

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

Некоммерческое акционерное общество
АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ

Факультет Информационных технологий
Специальность Вычислительная техника и программное обеспечение
Кафедра Компьютерных технологий

ЗАДАНИЕ

на выполнение дипломного проекта

Студент Есен Айман Естайхозов
(фамилия, имя, отчество)

Тема проекта Сайт дистанционного дистанционного обучения для кафедр компьютерных технологий

утверждена приказом ректора № 115 от «24» сентября 2013 г.

Срок сдачи законченной работы «29» мая 2014 г.

Исходные данные к проекту требуемые параметры результатов проектирования (исследования) и исходные данные объекта

Разработка сайта дистанционного обучения для кафедр компьютерных технологий

Перечень подлежащих разработке дипломного проекта вопросов или краткое содержание дипломного проекта:

При разработке необходимо учитывать функциональность. Определить стратегические цели и представить при разработке

Перечень графического материала (с точным указанием обязательных чертежей)

Макет при создании сайта на этапе построения задачи создается структура сайта. Под структурой сайта понимается навигационная структура, и она оформлена в виде пунктов меню

Рекомендуемая основная литература

1. Линк Бэйли, Майк Моррисон. Изучаем PHP и MySQL
 2. Веллинг Л., Томсон Л. Разработка веб-приложений с помощью PHP и MySQL

Консультанты по проекту с указанием относящихся к ним разделов

Раздел	Консультант	Сроки	Подпись
Курсовая	Зрессова З. Д.	11.04 - 13.05.14	Зрессова
БЖД	Тришова И. Б.	11.04 - 02.05.14	Тришова
Основная часть	Ертеман А. А.	11.04 - 20.05.14	Ертеман
Нормоконтроль	Тучнов Д. М.	28.05.2014	Тучнов

Аннотация

Дипломная работа написана на тему "Сайт бесплатного дистанционного обучения для кафедры КТ". Кафедра компьютерных технологий Алматинского университета энергетики и связи является объектом изучения, а именно процесс обучения. Предметом исследования в данной работе является внедрение дистанционного обучения в процесс обучения на данной кафедре.

Дипломная работа состоит из пяти разделов. Во введении раскрыта основная идея разработки данного сайта и актуальность этого метода в настоящее время. В пяти разделах описаны от этапа разработки до создания сайта. В заключении представлены выводы по проделанной работе.

Аңдатпа

Дипломдық жұмыстың тақырыбы "Компьютерлік технология кафедрасының тегін дистанциондық оқу сайты". Алматы энергетика және байланыс университетінің компьютерлік технология кафедрасының оқу үдірісі жобаның нысаны болып табылады. Зерттеу пәні болып, дистанциондық оқуды осы кафедраға еңгізу.

Дипломдық жоба бес тараудан тұрады. Кіріспе сайтың жобаланған негізгі ойын білдіреді және оның осы ауқыттағы маңыздылығы. Бес тарауда жобаның бас тапқы кезеңінен толық сайттың жасалғанына дейін. Ең соңында жұмыстың қортындысы көрсетілген.

Annotation

Diploma thesis written on the topic of "Free site for distance learning department CT". Department of Computer Technologies Almaty University of Power Engineering and Telecommunication is the object of study , namely the process of learning. Subject of study in this paper is the introduction of distance learning in the learning process in this department.

My thesis consists of five chapters. In the introduction, the basic idea is disclosed development of this site and the relevance of this method at the moment. The first section was analyzed purpose of this site and overview of modern technologies in developing websites . In conclusion, presented findings on the work done .

Содержание

Введение.....	14
1 Аналитическая часть.....	15
1.1 Современные технологии разработки сайтов	15
1.1.1 Особенности современных Web-редакторов	16
1.1.2 Классификация интернет-систем	17
1.2 Последовательности создания сайта бесплатного дистанционного обучения	20
1.2.1 Постановка задачи	20
1.2.2 Назначение и область применение сайта	21
1.2.3 Анализ предметной области	22
1.2.4 Модели и технологии дистанционного обучения	25
1.2.5 Дистанционное обучение с использованием сети Интернет.....	31
1.2.6 Дистанционное обучения в Казахстане	33
2 Программы используемые при разработке программы	41
2.1 История создания PHP	41
2.2 Преимущества PHP	43
2.3 Возможности PHP	45
2.4 Язык гипертекстовой разметки HTML	47
2.5 CSS – каскадные таблицы стилей.....	48
2.6 Объектно-ориентированный язык JavaScript	49
2.7 Основы XML.....	50
2.8 Система управления базами данных MySQL.....	52
3 Создание базы данных и приложения к ней.....	57
3.1 СУБД	57
3.2 Разграничение доступа	58
3.3 Создание базы данных на Wampserver	62
3.4 Описание программы.....	64
4 Техничко-экономическое обоснование	71
4.1 Описание работы и обоснование необходимости	71
4.2 Оборудование и программное обеспечение используемое в работе.....	71
4.3.Определение объема и трудоемкости разработки программного обеспечения.....	72
4.4 Расчет затрат на разработку информационных технологий.....	74
4.5 Расчет цены программного продукта	79
Вывод.....	79
5 Безопасность жизнедеятельности.....	81
5.1 Анализ условий труда.....	81
5.2 Рабочее место программиста	85
5.3 Расчёт естественного освещения.....	86
5.4 Расчёт искусственного освещения методом коэффициента использования	88

5.5 Средства пожаротушения.....	90
Вывод.....	91
Список литературы	93
Приложение А	94
Приложение Б.....	142
Приложение В.....	144
Приложение Г	145
Приложение Д.....	145
Приложение Ж.....	145
Приложение З	146

Введение

Новые технологии свойственны для современного человека, в ежедневной жизни технологии и новшество по всюду, куда бы не обратился человек и чем бы не воспользовался. Ведь технологии ускоряют и упрощают нашу жизнь. Инновации характерны для любой профессиональной деятельности человека и поэтому естественно становятся предметом изучения. Вот и в процесс получения образования, вводятся новые технологии, которые будут упрощать работу преподавателей, а для студентов выступать как дополнительная закрепление полученной информации в аудиториях. Для обеих сторон как для преподавателей так и для студентов дистанционное обучение будет ускоряющим механизмом в их профессиональной деятельности. Самостоятельное освоение материала приводит и учит самостоятельности студентов, они также должны осознавать что образование это не только положительные оценки, а самостоятельность и их будущее. Дистанционное освоение курсов будет закреплять полученные знания. Дает возможность проверить дополнительно свои знания.

Данный сайт дистанционного обучения для студентов кафедры компьютерных технологий предназначена для изучения основных языков программирования, а также других технологий связанные с их деятельностью. Хотелось бы отметить что сайт доступен на бесплатной основе, для студентов кафедры КТ.

Дистанционное обучение предлагаемое в данной работе не подразумевает замену стандартного очного обучения, а модернизация, улучшения, помощь и ускорения получения образования. На данный момент весь учебный материал имеется в электронном виде, вполне осуществимо ввести дистанционное обучение курсов.

Правительством РК утверждены Правила организации учебного процесса по дистанционным образовательным технологиям (Постановление Правительства Республики Казахстан от 19 января 2012 года № 112 «Об утверждении Правил организации учебного процесса по дистанционным образовательным технологиям»), отталкиваясь от данного постановления можно утверждать, что дистанционное обучения актуальна в наше время и не только в нашей стране, но и во всем мире.

Главная цель данного разработанного мною сайта улучшения и модернизация качества образования, а также ускорения усваивания информации.

1 Аналитическая часть

1.1 Современные технологии разработки сайтов

Создание сайтов – это далеко не такой легкий процесс, каким он может показаться на первый взгляд непрофессионалу. Создать эффективный сайт, который будет привлекать внимание пользователей в сети, при этом удовлетворяя самым высоким требованиями различных поисковых систем, по силам не каждому разработчику. Необходимо постоянно совершенствовать свои знания, овладевая новыми технологиями создания сайтов. Современные технологии создания сайтов позволяют создавать веб-ресурсы, которые быстро оправдывают вложенные в их разработку средства, предоставляя владельцу широкие возможности для использования сайта как инструмента маркетинговой политики, а также средства распространения необходимой информации. Но данный созданный сайт, не влечет за собой финансовую прибыль, а улучшения и модернизация процесса образования.

Безусловно, владеть различными технологиями создания веб-сайтов может только специалист, для которого создание сайтов – это каждодневный труд. Какие же технологии создания сайта необходимо в совершенстве изучить, чтобы получать результаты, которые будут удовлетворять всем требованиям заказчика:

- PHP-скрипт, являющийся одним из самых популярных среди технологий создания веб-сайтов;
- Java Script, который постоянно совершенствуется в процессе создания сайтов;
- HTML, необходимый для создания страниц текста, которые затем будут легко читаться в любом браузере;
- СУБД и MySQL, предназначенные для работы со структурированными базами данных;
- CSS – оформления внешнего вида веб-страниц, написанных с помощью языков разметки HTML и XHTML, но может также применяться к любым XML-документам, например, к SVG или XUL.

Каждый разработчик использует те технологии создания сайтов, которые считает наиболее подходящими для себя. Однако грамотный специалист выбирает технологию в соответствии с задачами, которые перед ним возникают в процессе разработки, и которые он должен максимально эффективно решить.

Технологии создания сайтов не стоят на месте, каждый день появляются новинки, которые можно использовать в процессе работы. Создание сайтов – творческий процесс, а для реализации всех своих идей необходимо владеть современной технологической базой, совершенствовать свои умения и получать новые знания в области веб-технологий.

1.1.1 Особенности современных Web-редакторов

Редактор источников кода дает не имеющие ограничений возможности контроля над HTML-тэгами и их форматированием. Для того чтобы переключиться из режима разметки страницы в режим редактирования кода, можно просто щелкнуть по специальной вкладке в окне документа. Внеся изменения в код, вы можете тут же просмотреть их, щелкнув по другой вкладке.

Контроль синтаксиса, поддержка HTML и функций JavaScript мало отличаются на различных платформах. Вот почему нынешние Web-дизайнеры располагают встроенной функцией проверки синтаксиса. Она разрешает узнать, какие браузеры, и на каких платформах поддерживают все те элементы, которые вы предусмотрели на своей странице, будь то DHTML или таблицы стилей. Вы можете построить страницу в HTML 4.0, преобразовать ее в стандартный формат HTML 3.2 или произвести ее в обоих вариантах, дав право выбора пользователю.

Динамические компоненты и шаблоны. Если на Web-сайте имеются такие общие элементы дизайна, как навигационные панели и заголовки, которые наличествуют на каждой странице, то вам особенно приглянутся динамические компоненты, разрешающие употреблять некоторые элементы страниц, сделанные раньше и уже снабженные текстами, картинками или другим содержимым. Проще говоря, можно организовывать шаблонные страницы и употреблять их снова и снова для сотворения сложных сайтов.

Встроенный FTP-клиент. Нынешние Web-сайты оснащены глубоким интегрированным FTP-клиентом, который разрешает загружать и обновлять Web-сайт. Когда появляется нужда обновить сайт, запускается процесс обновления, будут обновлены только помеченные файлы. Можно так же переносить отдельные страницы, просто "переносить" их с панели FTP или, напротив, на нее. Для удобства работы нынешние Web-дизайнеры высчитывают время, которое потребуется посетителям сайта для загрузки ваших страниц.

HTML-редактор – это программа, с помощью которой создаются HTML-документы, то есть те самые WEB-страницы. В настоящее время существуют редакторы двух типов: WYSIWYG-редакторы и редакторы тегов. К этому можно добавить, что большинство современных текстовых редакторов умеют сохранять документы в формате HTML. WYSIWYG-редакторы – это программы, с помощью которых можно создавать страницы, не имея никакого понятия о языке HTML. Самым распространенным из них является, вероятно, Microsoft FrontPage. При работе с этого редактора не видно самого процесса строительства страницы. Это все равно, как строить дом, собирая его из готовых, уже покрашенных и помытых кем-то стен, крыши и т.д. Удобно, конечно, быстро, но никогда не можешь быть уверенным, что дом будет теплым, крепким, надежным. И вторая причина, более важная, но, тем не менее, вытекающая из первой: у меня нет достаточного опыта работы с такими

редакторами. Если кто-то предпочтет работать с Microsoft FrontPage, например, я думаю, он найдет как минимум несколько учебников в интернете, плюс учебник в самой программе. К тому же ему можно пропустить те главы на этом сайте, в которых будет рассказываться о создании страницы с помощью редактора тегов и значит у него появится время для освоения Microsoft FrontPage.

Недостатком WYSIWYG-редакторов является громоздкость созданных с их помощью HTML-документов. Это получается оттого, что WYSIWYG-редакторы вставляют HTML-коды, как говорится, "на все случаи жизни". В результате объем документа, а значит и время загрузки страницы, увеличивается. Ведь серверу даже для того, чтобы рассказать браузеру о пустом месте, то есть пробеле, необходимо передать информацию, а значит затратить какое-то время. Второй вид HTML-редакторов – редакторы тегов. При работе с ними Вы видите непосредственно HTML-код документа, можете оперативно изменять его, контролируя результат – просматривая создаваемую страницу браузером. В этом случае, как правило, документ получается более компактным и изящным (здесь говорится именно о HTML-документе, а не о странице, которая будет видна в окне браузера. Красота и изящность страницы очень слабо связаны со способом ее создания).

1.1.2 Классификация интернет-систем

Цель любой информационной системы – обработка данных об объектах реального мира.

Интернет продукты можно классифицировать по ниже следующим пунктам.

По правам на доступность сервисов:

– Открытые – все сервисы полностью доступны для любых посетителей и пользователей.

– Полуоткрытые – для доступа необходимо зарегистрироваться (обычно бесплатно).

– Закрытые – полностью закрытые служебные сайты организаций (в том числе корпоративные сайты), личные сайты частных лиц. Такие сайты доступны для узкого круга пользователей. Доступ новым пользователям обычно даётся через приглашения (инвайты).

По физическому расположению:

– Общедоступные сайты сети Интернет.

– Локальные сайты – доступны только в пределах локальной сети.

Информационные ресурсы:

– Тематический сайт – сайт, предоставляющий специфическую узкоспециализированную информацию по какой-либо теме.

– Тематический портал – это большой веб-ресурс, который предоставляет исчерпывающую информацию по определённой тематике.

Порталы похожи на тематические сайты, но дополнительно содержат средства взаимодействия с пользователями и позволяют пользователям общаться в рамках портала (форумы, чаты) – это среда существования пользователя.

Интернет-представительства владельцев бизнеса (торговля и услуги, не всегда связанные напрямую с Интернетом):

- Сайт-визитка – содержит самые общие данные о владельце сайта (организация или индивидуальный предприниматель). Вид деятельности, история, прайс-лист, контактные данные, реквизиты, схема проезда. Специалисты размещают своё резюме, то есть подробная визитная карточка.

- Представительский сайт – так иногда называют сайт-визитку с расширенной функциональностью: подробное описание услуг, портфолио, отзывы, форма обратной связи и т.д.

- Корпоративный сайт – содержит полную информацию о компании/владельце, услугах/продукции, событиях в жизни компании. Отличается от сайта-визитки и представительского сайта полнотой представленной информации, зачастую содержит различные функциональные инструменты для работы с содержимым (поиск и фильтры, календари событий, фотогалереи, блоги, форумы). Может быть интегрирован с внутренними информационными системами компании-владельца (КИС, CRM, бухгалтерскими системами). Может содержать закрытые разделы для тех или иных групп пользователей – сотрудников, дилеров, контрагентов и пр.

- Каталог продукции – в каталоге присутствует подробное описание товаров/услуг, сертификаты, технические и потребительские данные, отзывы экспертов и т.д. На таких сайтах размещается информация о товарах/услугах, которую невозможно поместить в прайс-лист.

- Интернет-магазин – сайт с каталогом продукции, с помощью которого клиент может заказать нужные ему товары. Используются различные системы расчётов: от пересылки товаров наложенным платежом или автоматической пересылки счета по факсу до расчётов с помощью пластиковых карт.

- Промо-сайт – сайт о конкретной торговой марке или продукте, на таких сайтах размещается исчерпывающая информация о бренде, различных рекламных акциях (конкурсы, викторины, игры и т.п.).

- Сайт-квест – Интернет-ресурс, на котором организовано соревнование по разгадыванию последовательности взаимосвязанных логических загадок.

Веб-сервис – сайт, созданный для выполнения каких либо задач либо предоставления услуг в рамках глобальной сети:

- Доска объявлений.
- Каталог сайтов – например, OpenDirectoryProject.
- Поисковые сервисы – например, Yahoo!, Google.
- Почтовый сервис.
- Форумы.
- Блоговый сервис.
- Файл обменный Пиринговый сервис – например, Bittorrent.

- Датахостинговый (хранение данных) сервис – например, Skydrive.
- Датаэдиторинговый (редактирование данных) сервис – например, GoogleDocs.
- Фотохостинг – например, Picnik, ImageShack, Panoramio, Photobucket.
- Хранение видео – например, YouTube, Dailymotion.
- Социальные Медиа – например, Buzz.
- Комбинированные веб-сервисы (Социальные сети) – например, Facebook, Twitter, ВКонтакте.
- Комбинированные веб-сервисы (Специализированные социальные сети) - например, MySpace, Flickr, SpringMe.

Немного подробнее о досках объявлений. Доска объявлений – сайт предоставляющий площадку для размещения объявления. Это может быть сайт с конкретной узконаправленной тематикой, многопрофильный, платным или бесплатным, отдельным сайтом некоторого конкретного города либо региона, может быть модерлируемый либо администрируемый и т.д.

По технологии отображения:

- Статические – состоящие из статичных html (htm) страниц составляющих единое целое. Пользователю выдаются файлы в том виде, в котором они хранятся на сервере.

- Динамические – состоящие из динамичных html (htm, dhtml) страниц-шаблонов, информации, скриптов и прочего в виде отдельных файлов. Содержимое генерируется по запросу специальными скриптами (программами) на основе других данных из любого источника.

По типам макетов:

- Фиксированной ширины (англ.rigidfixed) – размеры элементов страницы имеют фиксированное значение, независящее от разрешения, размера, соотношения сторон экрана монитора и размеров окна обозревателя, задаётся в абсолютных значениях— PX (пиксели).

- Резиновый макет (англ.adaptablefluid) – размеры несущих элементов, значения ширины, задаются относительным значением – % (проценты), страницы отображаются во весь экран монитора по ширине . По сути данный тип макета является частным случаем фиксированного типа макета.

- Динамично эластичный (англ.dynamically expandable elastic) – размеры большинства элементов, задаются относительными значениями – EM и % . Все относительные пропорции размеров элементов всегда остаются неизменными, независимо от разрешения, размера, соотношения сторон экрана монитора, размеров окна и масштаба окна обозревателя, и всегда постоянны относительно окна обозревателя.

1.2 Последовательности создания сайта бесплатного дистанционного обучения

1.2.1 Постановка задачи

При постановке задачи необходимо решить следующие вопросы:

– целевая аудитория, на которую рассчитывается создание данного проекта;

– основные цели и задачи проекта.

Также при создании сайта на этапе построения задачи решается вопрос создания структуры сайта. Под структурой сайта подразумевается составление разделов сайта. Под структурой сайта понимается навигационная структура, и она обычно изображена в виде пунктов меню. К постановке задачи также относится разработка функциональности сайта и требования к дизайну.

Функциональность сайта значимый элемент, от которого зависит, приглянется ли посетителю система управления сайтом. Если Вы сделаете слишком тяжелую систему управления, тем меньшее количество посетителей захочет возвратиться на ваш сайт. В такой ситуации действует правило: «Будьте проще и люди к Вам потянутся».

Что относится к дизайну сайта, то здесь не существует каких-либо строгих правил. Дизайн сайта во многом зависит от целевого назначения сайта. Например, корпоративный сайт может отражать фирменный стиль корпорации. Таким образом, дизайн корпоративного сайта будет осуществлен в корпоративных цветах корпорации. Возьмем, простой пример. Банк имеет логотип, исполненный из нескольких элементов красного и желтого цвета. Т.е. будет логичным применять в дизайне сайта именно эти цвета. Кроме того, дизайнеры не советуют использовать в дизайне сайта более 2-3 тонов. Практика представляет, что во многом успех отдельных сайтов абсолютно не зависит от их дизайна. Здесь примитивным примером может быть поисковая система Яндекс, которая пользуется большой известностью, несмотря на достаточно простой дизайн.

Конечно, самым важным на этапе постановки задачи является определение стратегических целей.

Создание сайта может быть организовано на всевозможных масштабах бюджета, но независимо от этого, сайт создается для того, чтобы о нем узнали. Это значит что, при постановке задачи разработчики решают важные вопросы определения ключевых слов, назначенных для продвижения сайта в поисковых системах, установление перспективных рекламных площадок и т.д. Уже на этапе постановки задачи, сайтов владелец может примерно знать, чего ждать от реализации данного проекта.

1.2.2 Назначение и область применения сайта

Данный сайт предназначен для студентов кафедры компьютерных технологий. В дипломном проекте будут реализованы стандарты веб-сайта с применением системы динамического управления и наполнением на базе веб-интерфейса.

Разработка интерактивного web-сайта включает в себя выполнение следующего комплекса задач:

- разработка контента сайта;
- разработка дизайна сайта, включающая в себя создание графических элементов макета сайта и элементов навигации;
- создание форм, обеспечивающих интерактивное взаимодействие с клиентами корпорации;
- разработка панели управления содержимым сайта, для обеспечения возможности оперативного изменения и добавления информации на сайт;
- создание групп пользователей для организации безопасности работы в системе, путем распределения уровней доступа, модулями авторизации;
- разработка базы клиентов корпорации с возможностью просмотра данных обо всех клиентах корпорации, данных по предварительным заказам и данных по заключенным договорам для каждого из клиентов.

Контент сайта – это его информационное наполнение. Цель помещения контента на сайт состоит в том, чтобы предоставить клиентам корпорации максимально полную информацию о продуктах, предоставляемых сайтом, не утомяв его при этом слишком большими текстами и ненужной информацией.

Дизайн сайта – определяет визуальное оформление сайта и является характеристикой его внешнего вида. Дизайн предназначен для повышения эстетической ценности сайта от его содержимого.

Формы интерактивного взаимодействия с пользователями – позволяют получать информацию от клиента и отправлять ее на сервер для обработки. Служат для установления обратной связи с клиентами корпорации.

Панель управления содержимым сайта – используется для редактирования, удаления и добавления контента сайта, изменения структуры меню и способа навигации по сайту. Позволяет сотрудникам корпорации оперативно изменять информацию на сайте без привлечения сторонних специалистов, что позволяет избежать дополнительных затрат на сопровождение сайта и поддерживать актуальность представленной информации.

Организация безопасности – обеспечение безопасности работы в системе. Модуль авторизации позволяет ограничить доступ к панели управления сайтом, а также разграничить права пользователей. В частности, будет организовано две группы пользователей с предоставлением им различных возможностей доступа. Группе администраторов предоставляется полный доступ к панели управления с возможностью полного изменения web-сайта, группе менеджеров

отдела продаж предоставляется частичный доступ с возможностью изменения контента сайта и возможностью работы с базой клиентов корпорации.

Профили пользователя – содержат информацию из базы данных клиентов корпорации. Предоставляют возможность просмотра информации о клиентах корпорации. Содержат формы для ввода данных о договорах заключенных с клиентами. Введенная информация заносится в базу данных клиентов, что обеспечивает возможность ее последующего просмотра.

1.2.3 Анализ предметной области

Задача дистанционного обучения – учить, не имея прямого постоянного контакта с обучаемым. Можно было бы взять старую схему заочного обучения и перевести ее на новые технические рельсы. Но сейчас к обучению предъявляются новые высокие требования, диктуемые взрывным ритмом жизни и революционными техническими возможностями.

Можно сформулировать базовые принципы, на основе которых создаются системы дистанционного образования, являющиеся сегодня критерием удачных или неудачных решений.

Во-первых, доступность обучения. При хроническом недостатке времени хорошей альтернативой очному и заочному образованию является дистанционное обучение. Причем процесс обучения должен начинаться когда угодно – когда есть время; длиться сколько угодно – сколько есть времени и сил; он может быть внезапно прекращен или прерван, например срочный телефонный звонок, и может быть продолжен с любого места. А сверху еще накладываются субъективные особенности обучаемых: начальные знания, специфика освоения материала и т. д. Радикальный прорыв в этой области позволил осуществить известные результаты научно-технического прогресса: персональный компьютер и Интернет, а в ближайшей перспективе – мобильный Интернет и беспроводные локальные сети.

Во-вторых, радикально новые формы представления и организации информации, обеспечивающие максимальную степень ее восприятия. Среди них можно выделить:

- максимальное использование различных способов представления информации: текста, графики, видео, звукового сопровождения, анимации, т.е. то, что получило название "мультимедиа";

- нелинейную форму организации материала, при которой его единицы представлены не в линейной последовательности, а как система явно указанных возможных переходов, связей между ними, дающая возможность адекватного представления всей взаимосвязи ее различных аспектов. Такой подход позволяет в максимальной степени приблизить процесс передачи знаний к естественному общению и обеспечить адаптивность траектории обучения;

- присутствие большого количества справочной информации, причем именно в дополнительной, сопровождающей форме, когда пользователь видит основной предмет изучения в окружении каких-либо других узлов, т. е. любой

вопрос (тема, проблема, аспект, идея, документ) всегда оказывается связанным с другими вопросами. Пользователь может не учитывать эту информацию, но она ему предоставляется, причем именно как смежная, находящаяся в определенных связях с вопросом, непосредственно интересующим пользователя. В целом такая система заставляет учитывать, что интересующая его тема может иметь еще какие-то аспекты.

В настоящее время в мировой практике наиболее перспективным способом организации разнородной информации признана гипермедийная технология. Главными ее достоинствами являются, с одной стороны, ориентация на создание информационной среды, объективно отражающей практически любую предметную область, с другой – возможность адекватного представления всей взаимосвязи различных аспектов. Характерным примером ее использования является популярная World Wide Web – всемирная сеть, которую можно (в контексте задач обучения) рассматривать как интеллектуальную надстройку над информационным банком глобальной сети Интернет.

В–третьих, достоверность сертификации знаний. Оценка знаний предельно усложняется. Обычно сдача зачетов и экзаменов осуществлялась при непосредственном контакте с преподавателем, то в системах дистанционного обучения при сертификации знаний очного контакта, как правило, нет. Соответственно и методики сертификации знаний существенно отличаются от используемых при очном обучении. Наиболее распространенными здесь являются методики, основанные на тестировании, которые используются в практике очного образования скорее как дополнительные. Здесь они – основные. И сразу же встает другая проблема – при отсутствии очного контакта нет гарантий того, что экзамены не сдал кто–то другой. Именно поэтому на некоторых сертификатах, выдаваемых профессиональными организациями, есть подпись главного психометриста.

Таким образом, если к "дистанционным" сертификатам не будет доверия, как к сертификатам по очным формам обучения, то вся затея с дистанционным образованием теряет смысл, подрываются все экономические основы дистанционного обучения.

Многие не верят, что создание полноценной системы дистанционного обучения возможно. Системы, которая логически была бы вплетена в экономику труда и капитала и обеспечивала бы цепочку: есть спрос на специалистов – находятся желающие занять вакансии – они дистанционно обучаются – они дистанционно сертифицируют свои знания – их принимают на работу в организации, которым требуются специалисты.

По сути, дистанционное обучение с использованием информационных технологий можно охарактеризовать как форму заочного обучения, в которой недостаток очного общения преподавателя и студента компенсируется возможностью сделать это посредством телекоммуникаций. В развернутом описании дистанционное образование – комплекс образовательных услуг, предоставляемых с помощью специализированной информационной

образовательной среды, базирующейся на средствах обмена учебной информацией на расстоянии (спутниковое телевидение, радио, компьютерные сети и т. п.). Это и есть информационно-образовательная среда, представляющая собой совокупность средств приема и передачи данных, информационных ресурсов, протоколов взаимодействия, аппаратно-программного и организационно-методического обеспечения, получаемая пользователем в виде дистанционного курса.

Создание системы дистанционного обучения в учебном заведении предполагает целый комплекс работ, представляющих, по сути, кардинальную реорганизацию административной, методической и учебной деятельности. Такая работа требует:

- предварительной разработки концептуальных основ создания центра дистанционного обучения, с учетом специфики деятельности конкретного учебного заведения, включая подготовку программы создания системы ДО, с учетом существующей концепции дистанционного образования в Казахстане, действующей нормативной базы и примеров эффективного функционирования систем дистанционного обучения, как на базе классических учебных заведений, так и в форме самостоятельных "сетевых" центров;

- разработки кадровой структуры центра ДО, требований по техническому обеспечению, бизнес-плана создания такого центра и определения условий экономической целесообразности функционирования с учетом отраслевой специфики;

- структуризации учебных курсов по модульному принципу, позволяющей гибко компоновать пакеты учебно-методических материалов по специальностям и специализациям учебного заведения в соответствии с различными уровнями базового обучения, повышения квалификации, профессиональной подготовки и дополнительного образования;

- разработки комплексного плана подготовки учебно-методических материалов по специальностям и специализациям учебного заведения для использования в системе ДО;

- обучения авторов учебных курсов использованию инструментальных систем создания учебных курсов для системы ДО;

- подготовки и апробации учебно-методических материалов, включая электронные учебники и системы тестирования, для обучения, как преподавателей, так и студентов.

Эффективное функционирование системы дистанционного обучения предполагает наличие в учебном заведении мощной телекоммуникационной базы, имеющей выход по "быстрым" каналам во всероссийскую и международную информационную среду. Если учебные курсы, функционирующие на основе технологии Web-CD, допускают невысокие технические характеристики компьютера конечного пользователя и средств связи, используя которые он подключается к учебному центру, то компьютерная и телекоммуникационная база самого учебного центра должна

быть, без всяких сомнений, на высоте технических и эксплуатационных решений. Тем более что, как показывают расчеты, затраты на эти составляющие системы дистанционного обучения, равно как и на программное обеспечение, поддерживающее его функционирование, существенно ниже других необходимых расходов.

Таким образом, создание системы дистанционного обучения, подготовка и организация работы центра дистанционного обучения в учебном заведении – комплексная проблема, в которой тесно переплетаются вопросы учебно-методического обеспечения, организации учебного процесса, эксплуатации компьютерной и телекоммуникационной систем, подготовки и повышения квалификации преподавательского состава и др.

1.2.4 Модели и технологии дистанционного обучения

Достоинства распределенного дистанционного обучения обусловлены его новыми функциями, расширением возможностей и сервиса предоставления образовательных услуг обучающимся, распределенным по различным сегментам рынка и территориям, а также использованием системы гибкого непрерывного образования, гибких графиков, проведения занятий в синхронном и асинхронном режимах. Основной особенностью гибких режимов обучения является то, что они не так жестко регламентируют временные и пространственные рамки проведения занятий и общения между преподавателями и студентами. Это позволяет преподавателям и студентам более гибко планировать время обучения и учебный процесс. Особенно это важно для студентов-заочников и студентов, обучающихся в филиалах университетов в других городах.

Сегодня дистанционное обучение является одним из наиболее активно развивающихся направлений в образовании. Существующие возможности аппаратно-программного обеспечения современных компьютеров и Web-серверов позволяют разрабатывать интерактивные программы для получения образования посредством глобальной сети. Базируясь на Интернет-технологиях, дистанционное обучение расширяет свои возможности и рассматривается как перспективная система обучения, представляющая собой синтез технологий конкретного обучения, телевидения и международной сети Интернет. Данный новый вид обучения не только способствует интенсификации распространения знаний для широкого круга потенциальных обучаемых, но и активно формирует рыночную среду обучения из потенциальных потребителей новых знаний и технологий.

В наибольшей мере преимущества дистанционного обучения проявляются в преподавании гуманитарных дисциплин (социологии, политологии, экономики и др.), что обусловлено возможностью гибко сочетать теорию и практику, использовать свежую информацию для иллюстрации теоретических положений и анализа современной ситуации. Новая технология предполагает использование также электронной почты и телевидения как

формы контакта студентов с преподавателем в сочетании с очными учебно-консультационными занятиями.

Дистанционное образование – это средство, при котором учащиеся находятся на расстоянии от создателя учебных материалов; студенты могут учиться в любом месте по выбору (дом, работа, учебный центр) без непосредственного контакта с учителем. Дистанционное обучение на основе Интернет-технологий является современной универсальной формой профессионального образования, ориентированного на индивидуальные запросы обучаемых и их специализацию, а также предоставляет возможность обучаемым непрерывно повышать свой профессиональный уровень с учетом индивидуальных особенностей. В процессе такого обучения студент определенную часть времени самостоятельно осваивает в интерактивном режиме учебно-практические материалы, проходит тестирование, выполняет контрольные работы под руководством преподавателя, осуществляемого с помощью Интернета и других средств коммуникаций.

С развитием электронно-компьютерных сетей технология дистанционного обучения предполагает использование Интернет-телефонной связи, электронной почты и телевидения как формы контакта студентов с преподавателем в сочетании с очными учебно-консультационными занятиями на основе гибкого плана обучения. Это позволяет использовать разнообразные интерактивные формы обучения:

- проведение занятий и воспроизведение лекций для внешних удаленных слушателей;

- электронный доступ к гипертекстовым книгам, материалам интерактивного чтения и учебным упражнениям, программам лингвистического перевода;

- электронное дистантное взаимодействие обучаемых с профессорско-преподавательским составом, собеседования и учебно-консультационные занятия (тьюториалы);

- деловые игры и конкретные ситуации, ориентированные на специфику профессиональной деятельности и потребности реальной практики;

- телевизионные и компьютерные конференции и др.

Наибольшие перспективы дистанционного обучения связаны с интеграцией телекоммуникационных и компьютерных Интернет-технологий, одним из видов которой являются электронные интерактивные видеоконференции. Интерактивная обучающая видеоконференция представляет собой одну из наиболее передовых технологий дистантного обучения и обеспечивает распределенную онлайн-связь обучающихся и обучаемых, независимо от их пространственного и географического местоположения. Интерактивная мультимедийная видеоконференция обеспечивает технические условия организации глобальной образовательной сети распределенной аудитории обучающихся. Объединение сотрудников университетов универсальными цифровыми информационными магистралями ведет к формированию глобальной универсальной образовательной сети

гипермедиа, объединяющей обучающихся и обучаемых, потребителей и производителей знаний.

Исторически дистанционное образование прошло через следующие этапы:

1 Использование одной технологии и отсутствие непосредственного взаимодействия между преподавателем и студентом. Типичным примером является заочное обучение – обучение по переписке.

2 Интегрирование различных технологий, упор на односторонний поток информации и на самообучение; использование тьютора для взаимодействия со студентами. Типичный пример – открытые университеты.

3 Широкое использование двусторонних коммуникационных технологий для взаимодействия преподавателя со студентами и студентов между собой. Пример – распределенные системы дистанционного образования, использующие Интернет.

Существует несколько моделей дистанционного образования, представляющих его как систему взаимосвязанных компонент. Рассмотрим две наиболее популярные модели. Первая модель описана в книге Michael Moore «Distance Education. A Systems View» и может быть представлена в следующем виде (Таблица 1.1).

Т а б л и ц а 1.1 – Модель принятия решений в отношении разработки курсов от потребности студентов и специфики организации

Источники	Дизайн	Доставка	Взаимодействие	Учебная среда
Потребности	Разработка учебного процесса	Печатные материалы	Преподаватели	Рабочее место
Студента	Среда	Аудио/Видео записи	Тьюторы	Дом
Организация	Программа	Радио/Телевидение Компьютеры и программное обеспечение	Консультанты	Класс
Теория История Философия	Оценка	Аудиоконференции Видеоконференция Локальные и глобальные сети	Эксперты Администраторы Другие студенты	Учебный центр

В этой модели потребности студентов и специфика организации (миссия, экспертиза, философия) порождают решения в отношении разработки курсов. Дизайн курсов осуществляется специальными командами, объединяющими специалистов в области разработки учебного процесса, графики, продюсеров, программистов, специалистов по оценке эффективности обучения и т. д. Возможности студентов и педагогические принципы обуславливают выбор технологии. В учебные курсы встраивается система взаимодействия студентов

друг с другом, с преподавателями и экспертами, с администратором и т. д. Особое внимание уделяется природе учебной среды и предвидению потенциальных проблем, обусловленных ее спецификой, – например, эффект одинокого обучающегося.

Все элементы, представленные в описанной выше модели, являются взаимозависимыми и все они существенны для использования в дистанционной образовательной системе.

Вторая модель была разработана Tony Bates и описана в его книге «Technology, Open Learning and Distance Education». Эта модель называется ACTIONS и предназначена для принятия решений в области дистанционного образования, в особенности, для выбора технологий. Модель получила название по заглавным буквам слов, выражающих следующие критерии:

- Access (Доступ) – насколько используемая технология доступна для студентов;

- Costs (Затраты) – какова структура затрат на использование технологии, каковы затраты на одного обучающегося;

- Teaching and Learning (Преподавание и обучение) – какие виды обучения необходимы, какой педагогический подход реализует их наилучшим образом и каковы наиболее эффективные технологии для выбранной методики преподавания и обучения;

- Interactivity and User-friendliness (Интерактивность и легкость в использовании) – какова интерактивность выбранной технологии, как она может быть реализована;

- Organizational issues (Организационные вопросы) – каковы организационные барьеры на пути внедрения выбранной технологии, как они должны быть преодолены, какие изменения в организации потребуются для этого;

- Novelty (Новизна) – насколько нова выбранная технология;

- Speed (Скорость) – насколько быстро могут быть подготовлены курсы с использованием этой технологии, как быстро могут быть внесены в них изменения.

Доступность технологии является одним из важнейших факторов при ее выборе. Она зависит от выбранной целевой группы и учебной среды (обучаемые находятся дома, на рабочем месте, в университете, в учебном центре). Технология должна быть доступна для всех членов целевой группы, а не быть дополнительным средством для избранных. Часто доступность технологии дополняется ограничениями на место и время. Возникает возможность самостоятельно проанализировать доступность печатных материалов, аудио и видеокассет, телефона, телевизора, компьютера, Интернета для ваших условий.

Затраты также являются важнейшим фактором при выборе технологии. В структуре затрат важно выделить затраты на подготовку, производство, распределение и доставку материалов, административные расходы, коммуникационные расходы и др. Часть расходных статей напрямую зависит

от числа студентов, часть зависит от объема учебных материалов, некоторые расходы представляют собой накладные расходы, являющиеся фиксированными в рамках разработки и проведения учебного курса. Важной характеристикой затрат являются затраты на одного студента и на один час учебного процесса, которые, в свою очередь, определяются соотношением фиксированных и переменных издержек. Ясно, что эти затраты падают с увеличением числа студентов, что делает экономически выгодным обучение большого числа студентов как одновременно, так и путем многократного проведения разработанного курса.

Преподавание и обучение менее важно для выбора технологии, но важно для ее использования в учебном процессе, поскольку технологии различаются по своим презентационным характеристикам. Печатные материалы более подходят для передачи текста, диаграмм и графиков. Радио, аудиокассеты, телефон – для передачи звука. Телевидение и видеокассеты – для передачи образов и движения, компьютеры – для работы с базами данных, создания интерактивных программ и анимаций.

Интерактивность определяется как по отношению к учебным материалам, так и к обучаемым. Первый вид интерактивности лучше всего обеспечивается компьютерными программами, в особенности, с применением мультимедиа. Второй вид интерактивности обеспечивается двусторонними коммуникационными каналами: аудио или видеоконференция, телефон, факс, Интернет. Взаимодействие студентов и преподавателей может при этом быть как синхронным (одновременным), так и асинхронным (с задержкой во времени).

Организационные вопросы включают в себя техническую и человеческую инфраструктуру, создаваемую или используемую для разработки и проведения учебного курса. Это может быть наличие компьютеров, подключение к Интернету, наличие кабельной сети, аппаратуры для видеоконференций, занятость преподавателей в разработке дистанционных курсов и система вознаграждения, система технической поддержки преподавателей, наличие специалистов, необходимых для разработки и проведения дистанционных курсов и т. д.

Новизна технологии требует, с одной стороны, больших инвестиций в нее, с другой стороны, позволяет получать бесплатные продукты (в маркетинговых целях) или дополнительное финансирование. Новизна также тесно связана со временем, требуемым для внедрения и распространения технологии.

Скорость производства и обновления курсов связана с требованиями технического прогресса, устаревания технологии, жизненного цикла материалов курса, изменения спроса и т. д. Она также напрямую связана с затратами на разработку курсов. Например, использование видеоконференций позволяет за короткое время провести массу учебных мероприятий, в то время как для разработки Интернет-курса или мультимедийного курса требуется гораздо больше времени.

Организация дистанционно–образовательного курса начинается с формулировки задач, выбора методик, отвечающих заданным целям, выбора технологии, отвечающей заданным целям и методикам и создания разнообразных учебных ситуаций, таких как:

- педагогика сотрудничества;
- традиционное преподавание;
- учебные задачи и практика;
- интерактивные дискуссии;
- моделирование;
- демонстрации;
- открытия;
- игры;
- решение задач/проблем;
- наставничество.

Технологии дистанционного образования делятся на:

а) Интерактивные:

- аудиоконференции (audioconferencing);
- видеоконференции (videoconferencing);
- видеоконференции на рабочем столе (desktop videoconferencing);
- электронные конференции через:
 - e-mail,
 - online services;
- голосовая коммуникация (voice mail);
- двусторонняя спутниковая и СВЧ связь;
- виртуальная реальность (virtual reality).

б) Неинтерактивные:

- печатные материалы;
- аудиокассеты;
- видеокассеты;
- односторонняя спутниковая и СВЧ связь;
- телевизионные и радиопередачи;
- дискеты, CD-ROMы.

Часто дистанционное образование осуществляется не с помощью какого-то одного средства, а с помощью системы средств, например, телевизионный курс в сочетании с печатными материалами, рассылаемыми по почте и т. д. Интернет очень успешно используется при таком системном, смешанном подходе к дистанционному образованию, соединяя в себе все достоинства вышеперечисленных средств и используя и образы, и текст, и звук, и поиск, и реальное общение. Использование Интернета для дистанционного образования требует большего планирования и подготовки, чем все другие виды образования. Подготовка материалов и программ, использующих эти материалы, становится важнейшим компонентом образования.

1.2.5 Дистанционное обучение с использованием сети Интернет

Интернет – это международное объединение компьютерных сетей, использующих общий протокол семейства TCP/IP и единое адресное пространство. Среда WWW (World–Wide Web – Всемирная паутина) представляет собой унифицированную среду обмена информацией в глобальной компьютерной сети Интернет. В настоящее время это наиболее популярный вид доступа к данным. Он позволяет жителям всех континентов обмениваться информацией, совместно использовать компьютерные ресурсы при помощи того или иного прикладного сервиса Интернета (WWW, E–mail, FTP и др.) Благодаря развитию коммуникационных линий Интернет стремительно растет и развивается, охватывая все больше и больше пользователей в разных регионах и странах мира.

Существование многочисленных разнообразных программ для просмотра WWW–документов для всех популярных аппаратных и программных платформ, простой и понятный пользовательский интерфейс, возможность представления информации в форме мультимедиа в рамках единого стандартного языка форматирования документов HTML (HyperText Markup Language) обуславливают быстро растущую популярность среды WWW. Все это позволяет эффективно использовать среду WWW для дистанционного обучения и контроля знаний учащихся. Наличие развитого программного обеспечения, программ для создания и редактирования HTML–документов позволяет существенно упростить и сократить время подготовки учебных материалов показана в таблице 1.2.

Т а б л и ц а 1.2 – Характеристики дистанционного образования через Интернет

Название	Характеристики
Отсутствие временных ограничений	Почта может быть получена и прочитана в любое время. Преподаватель и студенты могут контактировать без привязки к строгому расписанию.
Отсутствие пространственных ограничений	Материалы могут быть получены или посланы в любое место
Синхронное общение	Использование диалоговых форм, требующих одновременного участия в процессе обучения студента и преподавателя
Асинхронное общение	Использование диалоговых форм, не требующих одновременного участия в процессе обучения студента и преподавателя
Линейная и нелинейная формы обучения	Обучения может быть структурировано как преподавателем, так и студентом. Использование как текстовых, так и гипертекстовых форм обучения

Ключом к получению информации в Интернете являются адреса ресурсов. Списки пользователей Интернета можно найти в адресных справочниках («Белые страницы» Интернета). Адреса ресурсов Интернета для получения необходимой информации по различным отраслям науки, искусству, бизнесу и т. д. можно найти в справочнике «Желтые страницы». Кроме того, существует целый ряд систем поиска ресурсов Интернета (Google, Yahoo, Yandex, Rambler и др.).

Интернет является хранилищем постоянно обновляемых информационных ресурсов. Здесь можно найти информацию об исследованиях, грантах, стипендиях, по наиболее актуальным проблемам экономики и преподавания экономических дисциплин. Если необходимо получить систематизированные знания в какой-либо области, то Интернет может быть самым эффективным (как с точки зрения минимизации затрат времени и денег, так и с точки зрения получения нужных знаний в нужном объеме) средством достижения этой цели.

Постоянно растет число периодических изданий, существующих только в электронной форме. Электронные версии печатных журналов легко доступны и постоянно обновляются. С помощью Интернета можно узнать о научных конференциях и семинарах, а также познакомиться с их материалами, опубликовать свои научные и учебно-методические разработки.

На основе Интернет-технологий, дистанционного обучения возникает реальная возможность создания единого распределенного он-лайн-образовательного сообщества (группы сообществ, сегментированных по профессиональным интересам или другим признакам, например, преподавание информатики для медицинских работников) образовательных учреждений России (а в перспективе, и стран ближнего и дальнего зарубежья) и их интеграции в единое образовательное Интернет-пространство (национальное и международное).

Для специалиста (или будущего специалиста) Интернет:

- является хранилищем ресурсов по экономическим наукам, которые постоянно обновляются;
- облегчает задачу получения необходимой информации с помощью поисковых систем;
- может быть самым эффективным средством приобретения новых знаний;
- может познакомить с новыми методиками преподавания дисциплин, позволяющими использовать обучающие игры по различным отраслям наук, интерактивные программы, моделирующие различные процессы, компьютерные упражнения, тестирование и самотестирование и т. д.;
- предоставляет электронные версии экономических журналов и газет. Имеется множество периодических изданий, существующих только в электронной форме, и их число постоянно увеличивается;

– знакомит с новостями в различных сферах и с оценками этих новостей ведущими специалистами;

– дает информацию о научных конференциях и семинарах, их материалах. Обмен информацией и публикации результатов исследований в Интернете создают возможность совместных научных исследований ученых, имеющих общие интересы, независимо от того, в какой стране расположены научные и учебные организации, в которых они работают;

– помогает найти ответы на возникающие вопросы в электронных дискуссиях в Usenet (Newsgroups) и т.п.

В целом Интернет-технологии, электронные средства коммуникаций позволяют перейти на новый этап развития распределенного сотрудничества и интеграции университетов в организационной, учебно-методической и научно-исследовательской областях как различных регионов Казахстана, так и стран СНГ. В этой связи можно говорить о формировании на локальном, региональном, межрегиональном, национальном и межгосударственном уровнях распределенного образовательного Интернет-сообщества.

1.2.6 Дистанционное обучения в Казахстане

Одним из наиболее привлекательных достоинств дистанционного обучения является возможность совмещения эффективности индивидуального обучения с экономичностью массового. По сути она представляет собой вариант автоматизированного обучения, в котором компьютер выполняет роль как учителя, так и администратора – организатора учебного процесса, с использованием локальных или глобальных коммуникационных сетей. Данная система образования позволяет начать обучение с любого момента в зависимости от подготовки обучаемого. Кроме того, она дает возможность повышения качества и интенсификации обучения. Каждый студент получает возможность самостоятельно определять сроки получения образования.

С развитием дистанционного обучения со временем необходимость в заочной форме обучения в казахстанских ВУЗах может отпасть, считает вице-министр образования и науки республики Тахир Балыкбаев, сообщает ИА Новости-Казахстан. "Есть такое задание, чтобы мы побольше привлекали рабочую молодежь для получения высшего образования. Сегодня его (заочное) закрыть полностью нельзя, но использование новых технологий, в том числе дистанционных, это позволит нам повысить качество образования в этом направлении. Поэтому это вопрос серьезный. Если будет хорошо развита система дистанционного обучения, может через какое-то время необходимость сама отпадет", – сказал Балыкбаев на брифинге в понедельник.

При этом он отметил, что пока заочная форма образования отменяться не будет. "Заочное образование нас очень сильно волнует, потому что там качество образования оставляет желать лучшего. Но с другой стороны полностью отменить заочное образование – объективно такой ситуации нет", – заверил вице-министр.

По его данным, сейчас порядка 150 тысяч казахстанских студентов из 520 тысяч обучается на заочном отделении. По заочному обучению выделяются места только на педагогические специальности, порядка 1 тысячи мест из 35 тысяч грантов, по госзаказу. Самостоятельно поступают на заочное около четверти абитуриентов.

"Поэтому полностью закрыть это направление пока такого не будет", – добавил Балыкбаев. В свою очередь глава Минобрнауки Аслан Саринжипов добавил, что основным критерием для министерства выступает не форма обучения в ВУЗах, а качество. "Здесь я хотел бы подчеркнуть, что мы не акцентируем внимание на форме обучения, мы будем акцентировать внимание на качестве обучения, будь то очное или заочное", – сказал министр. "Если качественное обучение – вопросов нет. Если есть вопросы некачественного образования, продажи дипломов и оценок, с этим мы мириться не будем. Ни в очной форме, ни в заочной. Другое дело, что отдельные вузы превращают это (заочное образование) в источник получения дохода. Здесь есть вопросы", – заключил Саринжипов.

На рубеже веков объективные потребности развития современного общества обусловили создание во многих странах мира системы дистанционного образования. Актуальность подобной формы обучения для Казахстана и стран СНГ очевидна: дистанционное образование призвано сыграть жизненно важную роль в сохранении и развитии единого образовательного пространства наших стран. Большой вклад в этот интеграционный процесс, на наш взгляд, может и должна внести Республика Казахстан. Во всем мире экспорт образовательных услуг несет не только социокультурную, но и экономическую нагрузку.

К примеру, по данным Института международного образования Соединенные Штаты Америки, занимая 5-е место среди крупнейших «экспортеров» образовательного продукта, ежегодно получают в казну 7 млрд. долл. Большая часть иностранных студентов – 57% представляют Азию, и лишь 15% – Европу. Сегодня высшее образование в Соединенных Штатах Америки называют «100-миллиардным бизнесом», составляющим 2,7% валового национального продукта.

Приведенные цифры красноречиво свидетельствуют о необходимости наращивать усилия в развитии отечественного образования и его экспорта. Несомненно, мощным средством достижения этого является дистанционное обучение, позволяющее предлагать образовательные услуги всем в них нуждающимся, невзирая на государственные, национальные, географические, временные и другие факторы.

Необходимы решительные и активные действия в реформировании отечественной школы, создании нового облика и инфраструктуры образовательной сферы. Требуется критическое переосмысление и некоторых основ педагогической науки, их переориентация на современные информационные технологии и средства обучения. Подобной логике развития следуют многие образовательные системы мира. Стратегическая цель

дистанционного образования, в этой связи, чрезвычайно актуальна – обеспечить гражданам право получения образования любого уровня на месте своего проживания или профессиональной деятельности. Эта цель достигается с учетом мировой тенденции мобильного распространения знаний за счет обмена образовательными ресурсами. Закономерно, что средством достижения такой цели должны быть высокотехнологичные и научно-обоснованные организационные формы, имеющие дистанционный характер. Вопросы развития дистанционного образования педагогическое сообщество обсуждает уже не первый год. В настоящее время эта проблема в нашей стране переходит из плоскости разговоров и теоретических поисков развития дистанционного обучения в плоскость реального воплощения.

Вместе с тем, для современного мирового сообщества идея дистанционного обучения не нова. С 1938 года существует Международный совет по заочному образованию – одна из старейших международных образовательных организаций, которая с 1982 года известна как Международный совет по дистанционному образованию (ICDE). Появился новый термин «глобальное образование», который подразумевает целостную систему международного высшего образования, включающую традиционные общие компоненты, но на новой технологической основе.

В 1969 году в Великобритании был открыт первый в мире университет дистанционного образования – Открытый Университет Великобритании, характеризующийся его доступностью за счет невысокой цены и отсутствия необходимости частого посещения аудиторные занятия.

В 1988 был реализован Советско–американский проект «Школьная электронная почта». Пионерами спутниковых технологий дистанционного обучения в 1990–х стали Международная ассоциация «Знание» и ее коллективный член Современная гуманитарная академия. За период своего развития технология дистанционного обучения прошла несколько этапов становления.

Первый этап – форма дистанционного образования, в которой обучение организуется по схеме «педагог – один или несколько учеников». Виды средств связи между преподавателем и учащимся здесь немногочисленны: обычная почта, телефон, компьютеры.

Второй этап условно можно обозначить: «педагог – множество учеников». Его оформление произошло благодаря организации в процессе дистанционного обучения односторонней связи, то есть без «обратного» компонента.

Третий этап характеризуется появлением с начала 80–х годов и дальнейшим ростом популярности «Internet». Его влияние стало столь огромным, что связь «многих со многими» и системы обмена знаниями стали общими.

Сегодня можно говорить о четвертом – интегрирующем этапе развития дистанционного образования, основанном на комплексной – виртуально-тренинговой технологии обучения. Его основу составляют развивающиеся

средства доставки информации благодаря комплексному внедрению современных коммуникационных систем, которые позволяют максимально быстро передавать любые формы информации в любую точку земного шара .

Актуализируя дистанционную форму обучения, на наш взгляд, следует отметить, основные преимущества дистанционного образования:

- возможность заниматься в удобное время, в удобном месте и темпе (нерегламентированный отрезок времени для освоения дисциплины);

- параллельное с профессиональной деятельностью обучение, т.е. без отрыва от производства;

- возможность обращения ко многим источникам учебной информации (электронным библиотекам, банкам данных, базам знаний и т.п.);

- общение через сеть Интернет и посредством электронной почты;

- концентрированное представление учебной информации и мультидоступ к ней;

- использование в образовательном процессе новейших достижений информационных и телекоммуникационных технологий, что обучает и работе с ними;

- равные возможности получения образования независимо от места проживания, состояния здоровья, элитарности и материальной обеспеченности обучаемого;

- экспорт и импорт мировых достижений на рынке образовательных услуг.

В целом, дистанционное обучение расширяет и обновляет роль преподавателя, который должен координировать познавательный процесс, постоянно совершенствовать преподаваемые им курсы, повышать творческую активность и квалификацию в соответствии с нововведениями и инновациями. Позитивное влияние оказывает дистанционное образование и на студента, повышая его творческий и интеллектуальный потенциал за счет самоорганизации, стремления к знаниям, умения взаимодействовать с компьютерной техникой и самостоятельно принимать ответственные решения. Таким образом, качество дистанционного образования не уступает в идеале качеству очной формы получения образования, а улучшается за счет привлечения выдающегося кадрового профессорско-преподавательского состава и использования в учебном процессе наилучших учебно-методических изданий и контролирующих тестов по дисциплинам.

В Республике Казахстан, принципы развития дистанционного образования были предусмотрены еще в Законе Республики Казахстан от 18 января 1992 года «Об образовании» (утратил силу – Законом Республики Казахстан от 7 июня 1999 года).

В частности, в указанном законодательном акте одним из принципов образования указывалось – «опережающее развитие образования», структура системы образования охватывало повышение квалификации и переподготовку кадров, субъекты образовательного процесса имели право на обучение

экстерном, на совмещение обучения с работой в свободное от учебы время, студентам высших и учащимся средних специальных и профессионально-технических учебных заведений, обучающихся без отрыва от производства, предоставлялось право на дополнительный оплачиваемый отпуск по месту работы, сокращенную рабочую неделю и другие льготы.

Помимо этого, одной из функций Министерства образования Республики Казахстан являлось изучение и распространение передового опыта и современных технологий обучения. В Законе Республики Казахстан от 7 июня 1999 года «Об образовании» (утратил силу – Законом Республики Казахстан от 27 июля 2007 года) регламентируется понятие дистанционного обучения (образование на расстоянии) – как одной из форм обучения, целенаправленное и методически организованное руководство учебно-познавательной деятельностью и развитием лиц, находящихся в отдалении от организаций образования, посредством электронных и телекоммуникационных средств.

В действующем на сегодняшний день Законе Республики Казахстан от 27 июля 2007 года «Об образовании» используется понятие «дистанционные образовательные технологии» – как обучение, осуществляемое с применением информационных и телекоммуникационных средств при опосредствованном (на расстоянии) или не полностью опосредствованном взаимодействии обучающегося и педагогического работника. Принятие данного нормативного правового акта послужило основанием для разработки соответствующих актов уполномоченных государственных органов Республики Казахстан по вопросам дистанционного образования.

Так, приказом Министра образования и науки Республики Казахстан от 13 апреля 2010 года № 169 утверждены Правила организации учебного процесса по дистанционным образовательным технологиям, где закрепляются основные задачи организации учебного процесса с использованием дистанционных образовательных технологий: 1) внедрение информационных технологий в образовании; 2) индивидуализация обучения; 3) повышение эффективности (качества) обучения; 4) предоставление образовательных услуг лицам, для которых традиционные формы обучения являются недоступными.

Порядок организации учебного процесса по дистанционным образовательным технологиям определены также Правилами организации учебного процесса по дистанционным образовательным технологиям, утвержденными постановлением Правительства Республики Казахстан от 19 января 2012 года № 112. Немаловажными актами, создающими условия для развития дистанционного обучения в Республике Казахстан являются Программа по развитию информационных и коммуникационных технологий в Республике Казахстан на 2010 – 2014 годы, утвержденная постановлением Правительства Республики Казахстан от 29 сентября 2010 года № 983, а также Государственная программа развития образования Республики Казахстан на 2011 – 2020 годы, утвержденная Указом Президента Республики Казахстан от 7 декабря 2010 года № 1118.

Таким образом, на мой взгляд, основополагающая законодательная база для развития и широкого распространения дистанционного образования в Республике Казахстан в достаточной степени имеется. Вместе с тем, развитие дистанционного обучения должно рассматриваться не как самоцель в системе современного образования, а как мощное средство решения актуальных цивилизационных проблем Казахстана и стран Содружества Независимых государств в целом. За такой технологией, на наш взгляд, будущее нашего единого образовательного пространства. Одной из наиболее сложных социальных проблем развития дистанционного образования в странах СНГ остается недостаточная информированность о возможностях дистанционного образования и еще невысокий уровень позитивного отношения к его качеству у широкого круга потенциальных потребителей услуг дистанционного образования. Так, из 11,3 млн. платежеспособных потенциальных потребителей дистанционного образования в трех крупнейших странах СНГ реально информировано о возможности дистанционного образования только 2,5 млн. чел. Отрицательно настроено к дистанционному образованию всего 3% платежеспособных потенциальных потребителей образовательных услуг дистанционного обучения в Республике Казахстан и по 6% – в Российской Федерации и в Украине. Уровень позитивно-нейтрального отношения к дистанционному образованию выше всего среди платежеспособных потенциальных потребителей образовательных услуг дистанционного обучения в Республике Казахстан – 55%. В Российской Федерации он составляет 43%, в Украине – только 24%.

Повышение качества образовательных услуг путем внедрения современных образовательных технологий и, соответственно, принятие мер для занятия лидирующего положения позволит предостеречь прихода их извне, как пришли из-за рубежа современные компьютеры, телевизоры, автомобили и товары народного потребления. Это должен быть непрерывный процесс обучения и воспитания всего населения страны, причем, не только с импортом, но и с экспортом образования, его высокой мобильностью и вариативностью, с использованием дистанционных, в том числе виртуальных, компьютерных, информационных и других современных коммуникационных технологий. Речь идет о формировании системы отечественного дистанционного образования, претендующей на передовую роль в центрально-азиатском регионе.

Положительным опытом в данном направлении, на наш взгляд, могут служить:

– проведенная 27 апреля 2012 года Федеральной службой государственной статистики по Белгородской области Международная научно-практическая конференция «Дистанционное образование XXI века: проблемы, опыт, перспективы», по итогам которой руководителем территориального органа Федеральной службы государственной статистики по Белгородской области О.С. Тарановой отмечено: «дистанционное образование – это шаг к завтрашней профессии, успешной карьере и стабильному социальному статусу»;

– принятие 22 февраля 2012 года Российской Федерации Закона о дистанционном образовании, позволяющего при реализации образовательных программ независимо от форм получения образования применять электронное обучение, дистанционные образовательные технологии.

По данным статистики, в 2008 году из 6,7 млрд. людей, живущих на планете, лишь 1,581 млрд. (порядка 23%) имеют такой доступ к виртуальному информационному пространству. Кроме того, владельцы доступа к Интернету распределены по континентам неравномерно. Если в Северной Америке трое из четырех жителей имели на тот момент выход в Интернет, то в Азии в начале 2009 года – менее одного из пяти человек и только один из 17 – в Африке. Однако, несмотря на низкий уровень сетевого развития в Азии, сегодня в данном регионе мира более 40% людей (примерно 650 млн.) обращаются к Интернету. Еще десятилетие назад такого доступа здесь не существовало. Потенциально дальнейшее развитие сети Интернет может сделать Азию планетарным центром электронного образования и дистанционного образования. Так, в Малайзии, Индии и Филиппинах уже успешно функционируют открытые и дистанционные университеты, которые нацелены на активную интеграцию в мировое образовательное пространство дистанционного обучения.

Открытый университет Филиппин (UPOU) определяет электронное образование как поставку образовательных услуг или учебных программ электронными средствами, в качестве которых может использоваться как компьютер, так и любое другое электронное устройство – например, мобильный телефон, расширяя, таким образом, диапазон технологий, используемых для обучения. Сегодня на Филиппинах аккредитовано порядка 20 университетов и академических учреждений для осуществления дистанционного обучения, но лишь Открытый университет, осуществляющий дистанционное обучение с 1995 года, владеет полным правом на присуждение ученой степени после освоения кандидатом соответствующих учебных программ.

Скачок в развитии ИТ в последнее десятилетие повлек за собой бурное развитие дистанционного образования, в том числе и в Арабских странах, в которых, как и в большинстве государств, система дистанционного обучения рассматривается как один из наиболее эффективных способов улучшения качества обучения при максимальном охвате населения образовательными услугами на основе новейших технологий.

Открытые университеты получили признание со стороны общественности дистанционных форм обучения как достойной по качеству получаемого образования альтернативы традиционному обучению вкупе с дополнительными преимуществами.

В Японии успешно реализуется программа дистанционного образования в Международном Христианском университете, где генерируется немало инноваций в области ИТ и происходит их успешное внедрение в дистанционные образовательные технологии. Сегодня на территории Азии работают 70

университетов, среди которых 7 из 11 самых больших университетов в мире, каждый с более чем 100 000 обучающихся. Стирание границ между Азиатскими государствами и другими уникальными общественными формациями Азии повлекло за собой активное экспериментирование в области образования и рост числа дистанционных программ и университетов. Здесь разрабатывают уникальные смешанные программы, эффективно синтезирующие самостоятельное обучение с виртуальным и традиционным.

Таким образом, опыт зарубежных стран может быть оперативно апробирован и применен для повышения эффективности управления качеством педагогических технологий, применяемых в казахстанской системе дистанционного обучения. Казахстан стал первым центрально-азиатским государством, которое вошло в европейское образовательное пространство. «Великую хартию университетов» подписали уже 60 казахстанских вузов. В сфере высшего образования произошли значительные качественные изменения по развитию академической мобильности, внедрению трехступенчатой модели подготовки кадров, а также по созданию условий для предоставления автономии вузам страны. Наряду с этим, проведена реструктуризация высшего образования: введена трехступенчатая модель подготовки кадров: (бакалавриат, магистратура, докторантура PhD).

Кроме того, заложены новые принципы финансирования высшего образования, адекватные рыночным условиям. В 38 вузах республики реализуется двудипломное образование, в 131 вузе ведется обучение по кредитной технологии, в 42 – дистанционное обучение. Вместе с тем, уникальным образовательным проектом в системе высшего образования Казахстана является вуз международного класса «Назарбаев Университет». Данный университет работает в партнерстве с университетами, входящими в 30-ку лучших вузов мира. Студенты первого набора уже приступили к обучению по программе бакалавриата в трех школах – инженерии, наук и технологий, социальных наук. Положительным моментом развития дистанционного образования в Республике Казахстан, на наш взгляд, является разработанная Академией государственного управления при Президенте Республики Казахстан – Концепция создания дистанционного обучения государственных служащих в Академии государственного управления при Президенте Республики Казахстан (далее – Концепция). В соответствии с данной Концепцией, существующая система повышения квалификации государственных служащих не в состоянии в полной степени охватить и удовлетворить потребности государственных служащих в переподготовке и повышении квалификации.

2 Программы используемые при разработке программы

2.1 История создания PHP

Название PHP – рекурсивная аббревиатура, означающая «PHP: Hypertext Preprocessor» (ранее акроним расшифровывался как «Personal Home Page Tools»). Изначально PHP создавался в качестве надстройки над Perl для облегчения разработки веб-страниц.

В 1994 году программист (ныне живущий в Канаде) Расмус Лердорф (Rasmus Lerdorf) написал набор скриптов на Perl/CGI для вывода и учёта посетителей его онлайн-резюме, обрабатывающий шаблоны HTML-документов. Лердорф назвал набор PersonalHomePage (Личная Домашняя Страница). Вскоре функциональности и быстроты Perl – интерпретатора скриптов – перестало хватать, и Лердорф написал на языке C новый интерпретатор шаблонов PHP/FI (англ. PersonalHomePage / FormsInterpreter – «Личная Домашняя Страница / Интерпретатор форм»). PHP/FI включал базовую функциональность сегодняшнего PHP: оформление переменных в стиле Perl (\$имя_переменной для вывода значения), автоматическую обработку форм и встраиваемость в HTML-текст и многое другое. Новорожденный язык отличался от своего прародителя более простым и ограниченным синтаксисом.

В 1997 году после длительного бета-тестирования вышла вторая версия обработчика, написанного на C – PHP/FI 2.0. Её использовали около 1% (приблизительно 50 тысяч) всех интернет-доменов мира.

PHP 3.0 была первой версией, напоминающей PHP, каким мы знаем его сегодня. В 1997 году два израильских программиста, Энди Гутманс (AndiGutmans) и Зэв Сураски (ZeevSuraski), разработчики из израильского технологического института, расположенного в Хайфе, Израиль, переписали код с нуля: PHP/FI 2.0 был сочтён ими непригодным для разработки приложения электронной коммерции, над которым они в то время работали. Для совместной работы над PHP 3.0 с помощью базы разработчиков PHP/FI 2.0 Энди, Расмус и Зив решили объединиться и объявить PHP 3.0 официальным преемником PHP/FI, разработка же PHP/FI была практически полностью прекращена.

Одной из сильнейших сторон PHP 3.0 была возможность расширения ядра. Впоследствии интерфейс написания расширений привлек к PHP множество сторонних разработчиков, работающих над своими модулями, что дало PHP возможность работать с огромным количеством баз данных, протоколов, поддерживать большое число API. Фактически, это и был главный ключ к успеху, но стоит добавить, что немаловажным шагом оказалась разработка нового, намного более мощного и полного синтаксиса с поддержкой объектно ориентированного программирования.

Абсолютно новый язык программирования получил новое имя. Разработчики отказались от дополнения о персональном использовании, которое имелось в аббревиатуре PHP/FI. Язык был назван просто PHP.

К концу 1998 года PHP использовался десятками тысяч пользователей. Сотни тысяч веб-сайтов сообщали о том, что они работают с использованием этого языка. В то время PHP 3.0 был установлен приблизительно на 10 % веб-серверов Интернета.

PHP 3.0 был официально выпущен в июне 1998 года после 9 месяцев публичного тестирования.

К зиме 1998 года, практически сразу после официального выхода PHP 3.0, Энди Гутманс и Зив Сураски начали переработку ядра PHP. В задачи входило увеличение производительности сложных приложений и улучшение модульности базиса кода PHP. Расширения дали PHP 3.0 возможность успешно работать с набором баз данных и поддерживать большое количество различных API и протоколов, но PHP 3.0 не имел качественной поддержки модулей и приложения работали неэффективно.

Новый движок, названный ZendEngine (от имён создателей, Зива и Энди, также основателей ZendTechnologies), успешно справлялся с поставленными задачами и впервые был представлен в середине 1999 года. PHP 4.0, основанный на этом движке и принёсший с собой набор дополнительных функций, официально вышел в мае 2000 года, почти через два года после выхода своего предшественника PHP 3.0. В дополнение к улучшению производительности, PHP 4.0 имел ещё несколько ключевых нововведений, таких как поддержка сессий, буферизация вывода, более безопасные способы обработки вводимой пользователем информации и несколько новых языковых конструкций.

Прекращение выпуска обновлений PHP 4 было запланировано на конец 2007 года. Однако вплоть до 8 августа 2008 года выпускались критические обновления безопасности. С 9 августа 2008 года всякая поддержка версии PHP 4.x была прекращена.

Пятая версия PHP была выпущена разработчиками 13 июля 2004 года. Изменения включают обновление ядра Zend (ZendEngine 2), что существенно увеличило эффективность интерпретатора. Введена поддержка языка разметки XML. Полностью переработаны функции ООП, которые стали во многом схожи с моделью, используемой в Java. В частности, введён деструктор, открытые, закрытые и защищённые члены и методы, окончательные члены и методы, интерфейсы и клонирование объектов. Нововведения, однако, были сделаны с расчётом сохранить наибольшую совместимость с кодом на предыдущих версиях языка. На данный момент стабильной и наиболее популярной является версии PHP 5.xx.

Шестая версия PHP находится в стадии разработки с октября 2006 года. В ней уже сделано множество нововведений, как, например, исключение из ядра регулярных выражений POSIX и «длинных» суперглобальных массивов, удаление директив `safe_mode`, `php_magic_quotes` и `register_globals` из

конфигурационного файла `php.ini`. Также много внимания уделено поддержке Юникода.

2.2 Преимущества PHP

Главным фактором языка PHP является практичность. PHP должен предоставить программисту средства для быстрого и эффективного решения поставленных задач. Практический характер PHP обусловлен пятью важными характеристиками: традиционностью, простотой, эффективностью, безопасностью, гибкостью.

Существует еще одна «характеристика», которая делает PHP особенно привлекательным: он распространяется бесплатно. Причем, с открытыми исходными кодами (OpenSource).

Язык PHP будет казаться знакомым программистам, работающим в разных областях. Многие конструкции языка позаимствованы из Си, Perl.

Код PHP очень похож на тот, который встречается в типичных программах на С или Pascal. Это заметно снижает начальные усилия при изучении PHP. PHP – язык, сочетающий достоинства Perl и Си и специально нацеленный на работу в Интернете, язык с универсальным (правда, за некоторыми оговорками) и ясным синтаксисом.

И хотя PHP является довольно молодым языком, он обрел такую популярность среди web-программистов, что на данный момент является чуть ли не самым популярным языком для создания web-приложений (скриптов).

Сценарий PHP может состоять из 10 000 строк или из одной строки – все зависит от специфики вашей задачи. Вам не придется подгружать библиотеки, указывать специальные параметры компиляции или что-нибудь в этом роде. Механизм PHP просто начинает выполнять код после первой экранирующей последовательности (`<?>`) и продолжает выполнение до того момента, когда он встретит парную экранирующую последовательность (`?>`). Если код имеет правильный синтаксис, он исполняется в точности так, как указал программист.

PHP – язык, который может быть встроен непосредственно в html – код страниц, которые, в свою очередь будут корректно обрабатываться PHP – интерпретатором. Мы можем использовать PHP для написания CGI-сценариев и избавиться от множества неудобных операторов вывода текста. Мы можем привлекать PHP для формирования HTML-документов, избавившись от множества вызовов внешних сценариев.

Большое разнообразие функций PHP избавят вас от написания многострочных пользовательских функций на С или Pascal.

Эффективность является исключительно важным фактором при программировании для многопользовательских сред, к числу которых относится и web.

Очень важное преимущество PHP заключается в его «движке». «Движок» PHP не является ни компилятором, ни интерпретатором. Он является

транслирующим интерпретатором. Такое устройство «движка» РНР позволяет обрабатывать сценарии с достаточно высокой скоростью.

По некоторым оценкам, большинство РНР–сценариев (особенно не очень больших размеров) обрабатываются быстрее аналогичных им программ, написанных на Perl. Однако, чтобы не делали разработчики РНР, откомпилированные исполняемые файлы будут работать значительно быстрее – в десятки, а иногда и в сотни раз. Но производительность РНР вполне достаточна для создания вполне серьезных web-приложений.

РНР предоставляет в распоряжение разработчиков и администраторов гибкие и эффективные средства безопасности, которые условно делятся на две категории: средства системного уровня и средства уровня приложения.

В РНР реализованы механизмы безопасности, находящиеся под управлением администраторов; при правильной настройке РНР это обеспечивает максимальную свободу действий и безопасность. РНР может работать в так называемом безопасном режиме (safemode), который ограничивает возможности применения РНР пользователями по ряду важных показателей. Например, можно ограничить максимальное время выполнения и использование памяти (неконтролируемый расход памяти отрицательно влияет на быстродействие сервера). По аналогии с cgi-bin администратор также может устанавливать ограничения на каталоги, в которых пользователь может просматривать и исполнять сценарии РНР, а также использовать сценарии РНР для просмотра конфиденциальной информации на сервере (например, файла passwd).

В стандартный набор функций РНР входит ряд надежных механизмов шифрования. РНР также совместим с многими приложениями независимых фирм, что позволяет легко интегрировать его с защищенными технологиями электронной коммерции (e-commerce). Другое преимущество заключается в том, что исходный текст сценариев РНР нельзя просмотреть в браузере, поскольку сценарий компилируется до его отправки по запросу пользователя. Реализация РНР на стороне сервера предотвращает похищение нетривиальных сценариев пользователями, знаний которых хватает хотя бы для выполнения команды ViewSource.

Поскольку РНР является встраиваемым (embedded) языком, он отличается исключительной гибкостью по отношению к потребностям разработчика. Хотя РНР обычно рекомендуется использовать в сочетании с HTML, он с таким же успехом интегрируется и в JavaScript, WML, XML и другие языки. Кроме того, хорошо структурированные приложения РНР легко расширяются по мере необходимости (впрочем, это относится ко всем основным языкам программирования).

Нет проблем и с зависимостью от браузеров, поскольку перед отправкой клиенту сценарии РНР полностью компилируются на стороне сервера. В сущности, сценарии РНР могут передаваться любым устройствам с браузерами, включая сотовые телефоны, электронные записные книжки, пейджеры и портативные компьютеры, не говоря уже о традиционных ПК. Программисты,

занимающиеся вспомогательными утилитами, могут запускать PHP в режиме командной строки.

Поскольку PHP не содержит кода, ориентированного на конкретный web-сервер, пользователи не ограничиваются определенными серверами (возможно, неизвестными для них). Apache, Microsoft IIS, Netscape Enterprise Server, Stronghold и Zeus – PHP работает на всех перечисленных серверах. Поскольку эти серверы работают на разных платформах, PHP в целом является платформенно-независимым языком и существует на таких платформах, как UNIX, Solaris, FreeBSD и Windows 95/98/NT/2000/XP/2003.

Наконец, средства PHP позволяют программисту работать с внешними компонентами, такими как EnterpriseJavaBeans или COM-объекты Win32. Благодаря этим новым возможностям PHP занимает достойное место среди современных технологий и обеспечивает масштабирование проектов до необходимых пределов.

Стратегия OpenSource, и распространение исходных текстов программ в массах, оказало, несомненно, благотворное влияние на многие проекты, в первую очередь – Linux, хотя и успех проекта Apache сильно подкрепил позиции сторонников OpenSource. Сказанное относится и к истории создания PHP, поскольку поддержка пользователей со всего мира оказалась очень важным фактором в развитии проекта PHP.

Принятие стратегии OpenSource и бесплатное распространение исходных текстов PHP оказало неоценимую услугу пользователям. Вдобавок, отзывчивое сообщество пользователей PHP является своего рода «коллективной службой поддержки», и в популярных электронных конференциях можно найти ответы даже на самые сложные вопросы.

2.3 Возможности PHP

Возможности PHP очень большие. Главным образом, область применения PHP сфокусирована на написание скриптов, работающих на стороне сервера; таким образом, PHP способен выполнять всё то, что выполняет любая другая программа CGI. Например, обрабатывать данные форм, генерировать динамические страницы, отсылать и принимать cookies. Но PHP способен выполнять и множество других задач.

Существуют три основных области, где используется PHP:

– Создание скриптов для выполнения на стороне сервера. PHP наиболее широко используется именно таким образом. Все, что вам понадобится, это парсер PHP (в виде программы CGI или серверного модуля), веб сервер и браузер. Чтобы Вы могли просматривать результаты выполнения PHP-скриптов в браузере, вам нужен работающий вебсервер и установленный PHP. За более подробными сведениями обратитесь к подразделу установка PHP.

– Создание скриптов для выполнения в командной строке. Вы можете создать PHP-скрипт, способный запускаться вне зависимости от веб сервера и браузера. Все, что вам потребуется – парсер PHP. Такой способ использования

PHP идеально подходит для скриптов, которые должны выполняться регулярно, например, с помощью cron (на платформах Unix или Linux) или с помощью планировщика задач (TaskScheduler) на платформах Windows. Эти скрипты также могут быть использованы в задачах простой обработки текстов. Дополнительная информация находится здесь.

– Создание приложений GUI, выполняющихся на стороне клиента. Возможно, PHP является не самым лучшим языком для создания подобных приложений, но, если вы очень хорошо знаете PHP и хотели бы использовать некоторые его возможности в своих клиент-приложениях, вы можете использовать PHP-GTK для создания таких приложений. Подобным образом вы можете создавать и кросс-платформенные приложения. PHP-GTK является расширением PHP и не поставляется вместе с дистрибутивом PHP. Если вы заинтересованы, посетите сайт PHP-GTK.

PHP доступен для большинства операционных систем, включая Linux, многие модификации Unix (такие, как HP-UX, Solaris и OpenBSD), Microsoft Windows, Mac OS X, RISC OS, и многих других. (Существует даже версия PHP для OS/2. Неизвестно, правда, насколько соответствующая нынешним реалиям). Также в PHP включена поддержка большинства современных веб-серверов, таких, как Apache, Microsoft InternetInformationServer, PersonalWebServer, серверов Netscape и iPlanet, сервера OreillyWebsitePro, Caudium, Xitami, OmniHTTPd и многих других. Для большинства серверов PHP поставляется в качестве модуля, для других, поддерживающих стандарт CGI, PHP может функционировать в качестве процессора CGI.

Таким образом, выбирая PHP, вы получаете свободу выбора операционной системы и веб-сервера. Кроме того, у вас появляется выбор между использованием процедурного или объектно-ориентированного программирования или же их сочетания.

PHP способен не только выдавать HTML. Возможности PHP включают формирование изображений, файлов PDF и даже роликов Flash (с использованием libswf и Ming), создаваемых "на лету". PHP также способен выдавать любые текстовые данные, такие, как XHTML и другие XML-файлы. PHP способен осуществлять автоматическую генерацию таких файлов и сохранять их в файловой системе вашего сервера, вместо того, чтобы отдавать клиенту, организуя, таким образом, кеш динамического содержания, расположенный на стороне сервера.

Одним из значительных преимуществ PHP является поддержка широкого круга баз данных. Создание скрипта, использующего базы данных, – очень просто. В настоящее время PHP поддерживает следующие базы данных: Adabas, dBase, Empress, FilePro, Hyperwave, IBMDB2, Informix, Ingres, InterBase, FrontBase, mSQL, DirectMS-SQL, MySQL, ODBC, Oracle, Ovrimos, PostgreSQL, Solid, Sybase, Velocis, Unixdbm.

Также в PHP включена поддержка DBX для работы на абстрактном уровне, так что вы можете работать с любой базой данных, использующих DBX. Кроме того, PHP поддерживает ODBC (OpenDatabaseConnectionstandard),

таким образом, вы можете работать с любой базой данных, поддерживающей этот всемирно признанный стандарт.

RНР также поддерживает "общение" с другими сервисами с использованием таких протоколов, как LDAP, IMAP, SNMP, NNTP, POP3, HTTP, COM (на платформах Windows) и многих других. Кроме того, вы получаете возможность работать с сетевыми сокетами "напрямую". RНР поддерживает стандарт обмена сложными структурами данных WDDX. Обращая внимание на взаимодействие между различными языками, следует упомянуть о поддержке объектов Java и возможности их использования в качестве объектов RНР. Для доступа к удаленным объектам вы можете использовать расширение CORBA.

HP включает средства обработки текстовой информации, начиная с регулярных выражений Perl или POSIX Extended и заканчивая парсером документов XML. Для парсинга XML используются стандарты SAX и DOM. Для преобразования документов XML вы можете использовать расширение XSLT.

Последним по порядку, но не по значению, является поддержка многих других расширений, таких, как функции поисковой машины mnoGoSearch, функции IRC Gateway, функции для работы со сжатыми файлами (gzip, bz2), функции календарных вычислений, функции перевода и многое другое.

2.4 Язык гипертекстовой разметки HTML

HTML (Hyper Text Markup Language) – это язык гипертекстовой разметки. С его помощью можно оформлять различными способами интернет страницы, их дизайн и структуру. HTML является лучшим инструментом для включения элементов мультимедиа в документы. Изучая этот язык, я понял, что можно делать сложные вещи простыми способами, а главное быстро и эффективно, так как в компьютерном мире современности скорость имеет большое значение.

HTML–страницы не требуют разработки дополнительных программных продуктов, так как все инструменты необходимые для работы с данными в WEB–браузере уже давно стали частью стандартного программного обеспечения персональных компьютеров. Сам HTML интерпретируется браузером и отображается в виде документа, в удобной для человека форме. И поэтому требуется только подготовить тексты, нарисовать рисунки, создать HTML–страницы и составить связь между ними.

HTML – файл это и есть самый тот документ, который создан в виде гипертекста в текстовом редакторе, на основе языка HTML. Такие файлы при сохранении имеют расширение .html(.htm). В браузерах эти файлы имеют общее название «документ».

HTML – является теговым языком разметки документов. То есть любой документ имеющий расширение .html представляет собой набор элементов

называемых тегами, и начало и конец каждого элемента обозначается специальными символами.

На рисунке 2.1 показан пример HTML страницы.



Рисунок 2.1 – Пример HTML кода страницы.

```
<html>// начало документа  
<head>// начало заголовков документа  
<title>Заголовок документа</title>  
</head>// конец заголовков документа  
<body>//начало тело документа  
<p>Текст документа</p>  
</body>//конец тело документа  
</html>
```

И для создания простой Web-страницы достаточно изучить HTML, иметь на своем компьютере любой текстовый редактор для создания HTML-документа и браузер для его тестирования.

2.5 CSS – каскадные таблицы стилей

Если HTML используется для структурирования содержимого гипертекстовой страницы, то CSS используется для форматирования этого структурированного содержимого.

CSS – это каскадные таблицы стилей, которые являются поразительным изобретением для улучшения дизайна web-страниц. Он экономит время при разработке. Требуется меньше ресурсов, так как код может использоваться на разных элементах страницы и предоставляет новые возможности для оформления сайта.

CSS работает с форматированием и отображением HTML-документов. Корректирует шрифты, цвета, поля, рисунки, позиции элементов и многое другое. На сегодняшний день CSS поддерживается всеми браузерами.

Преимущества CSS таблиц стилей:

- управление отображением множества документов с помощью одной таблицы стилей;
- более точный контроль над внешним видом страниц.

Существует шесть типов таблиц стилей:

1 Стиль браузера – это стандартная таблица стилей, используемая браузером. Если никакие правила стиля не объявлены, применяются эти стандартные стили.

2 Стиль пользователя – пользователь может написать свою таблицу стилей и переопределить любые стили, созданные вами, изменив настройки браузера. Такой подход используется редко.

3 Встроенный стиль – стиль, который определяется непосредственно в элементе и применяется с использованием атрибута style. Такой подход полезен для стилей, одновременно применяемых к одному элементу, однако он не считается идеальным.

4 Внедренный стиль – этот стиль управляет представлением одного документа и размещается внутри элемента style HTML–документа.

5 Связанный стиль – это таблица стилей, которая связана с HTML–документом при помощи элемента link, размещенного в разделе документа head. Любой документ, связанный с данным типом таблицы стилей, получает все стили, определенные в ней, в чем и заключается преимущество управления языком CSS.

6 Импортированный стиль – этот тип похож на связанные стили, однако позволяет импортировать стили в связанную таблицу стилей или непосредственно в документ. Он полезен для реализации обходных путей и для управления множеством документов.

2.6 Объектно-ориентированный язык JavaScript

JavaScript – является объектно-ориентированным скриптовым языком программирования. Он используется в составе WEB–страниц и предназначен для написания сценариев, работающих как на стороне клиента, так и на стороне сервера, позволяя обеспечивать двустороннее взаимодействие между сайтом и пользователем. Таким образом увеличивает привлекательность сайта. JavaScript не является «полноценным» языком программирования, так как ориентирован на возможности той среды в которой он используется.

JavaScript имеет доступ к таким объектам как окна, меню, диалоги, текстовые области, фреймы, ввод и вывод в Web–страницу. А так же сценарии JavaScript можно присоединить к событиям как нажатие клавиш и движение мыши, загрузка и выгрузка страниц и графических образов. Одна из возможностей JavaScript является обеспечение запросов к базам данных.

Язык JavaScript по сравнению с родственными объектно-ориентированными языками Java и C++ не содержит в своем составе классы объектов. Он поддерживает конструкторы, которые создают объекты путем

выделения для них памяти. Объекты создаются путем вызова конструктора в операции `new` название элемента.

Как и в других языках программирования, программа на языке JavaScript – это текст, который состоит из операторов, блоков и комментариев. Операторы могут содержать переменные, константы и выражения. Для добавления JavaScript – кода на страницу, в редакторе можно использовать теги `<script></script>`. Ниже приведен пример простого вывода модального окна с сообщением через JavaScript.

```
<script type="text/javascript"> -- началоскрипта
Alert('Привет!!!'); -- модальное окно с сообщением
</script> -- закрытие скрипта
На html странице этот код будет выглядеть таким образом:
<html>
<head>
    <title>Пример JavaScript</title>
    <script type="text/javascript">
Alert('Привет!!!'); </script>
</head>
<body>
<input type=button name="p_button" value="Сообщение"
onclick="form1">
</body>
</html>
```

2.7 Основы XML

На сегодняшний день уже всем специалистам в области Web–технологий стало очевидно, что существующих стандартов передачи данных по Internet недостаточно. Формат HTML, став в свое время прорывом в области отображения содержимого узлов Internet, уже не удовлетворяет всем необходимым на данный момент требованиям. Он позволяет описать то, каким образом должны быть отображены данные на экране конечного пользователя, но не предоставляет никаких средств для эффективного описания передаваемых данных и управления ими.

Кроме того, камнем преткновения для многих компаний, занимающихся разработкой программного обеспечения, является необходимость совместного использования различных компонент, обеспечения их взаимодействия, возможности обмена данными между ними.

До недавнего времени не существовало стандарта, предоставляющего средства для интеллектуального поиска информации, обмена данными, адаптивной обработки получаемых данных.

Решением всех описанных выше проблем стал утвержденный в 1998 году международной организацией W3C язык XML. XML (eXtensible Markup Language) – это расширяемый язык разметки, предназначенный для описания в текстовой форме структурированных данных. Этот текстовый (text-based)

формат, во многом схожий с HTML, разработан специально для хранения и передачи данных.

XML позволяет описывать и передавать такие структурированные данные, как:

- отдельные документы;
- метаданные, описывающие содержимое какого-либо узла Internet;
- объекты, содержащие данные и методы работы с ними (например, элементы управления ActiveX или объекты Java);
- отдельные записи (например, результаты выполнения запросов к базам данных);
- всевозможные Web-ссылки на информационные и людские ресурсы Internet (адреса электронной почты, гипертекстовые ссылки и пр.).

Данные, описанные на языке XML, называются XML-документами. Язык XML легко читаем и достаточно прост для понимания. Если Вы были знакомы с HTML, то научиться составлять XML-документы не составит для Вас никакого труда.

Исходный текст XML-документа состоит из набора XML-элементов, каждый из которых содержит начальный и конечный тэги. Каждая пара тэгов представляет часть данных. То есть, как и HTML, язык XML для описания данных использует тэги. Но, в отличие от HTML, XML позволяет использовать неограниченный набор пар тэгов, каждая из которых представляет не то, как заключенные в нее данные должны выглядеть, а то, что они означают.

Любой элемент XML-документа может иметь атрибуты, уточняющие его характеристики. Атрибут – это пара имя = "значение", которая задается при определении элемента в начальном тэге.

Принцип расширяемости языка XML состоит в возможности использования неограниченного количества пар тэгов, определяемых создателем XML-документа.

Принцип независимости определения внутренней структуры документа от способов представления этой информации состоит в отделении данных от процесса их обработки и отображения. Таким образом, полученные данные можно использовать в соответствии с нуждами клиента, то есть выбирать нужное оформление, применять необходимые методы обработки.

Управлять отображением элементов в окне программы-клиента (например, в окне браузера) можно с помощью специальных инструкций – стилевых таблиц XSL (eXtensible Stylesheet Language). Эти таблицы XSL позволяют определять оформление элемента в зависимости от его месторасположения внутри документа, то есть к двум элементам с одинаковым названием могут применяться различные правила форматирования. Кроме того, языком, лежащим в основе XSL, является XML, а это означает, что таблицы XSL более универсальны, а для контроля корректности составления таких стилевых таблиц можно использовать DTD-описания или схемы данных, рассмотренные ниже.

Формат XML, по сравнению с HTML, имеет небольшой набор простых правил разбора, который позволяет разбирать XML-документы, не прибегая к каким-либо внешним описаниям используемых XML-элементов. В общем случае XML-документы должны удовлетворять следующим требованиям:

- Каждый открывающий тэг, определяющий некоторую часть данных в документе, обязательно должен сопровождаться закрывающим, то есть, в отличие от HTML, нельзя опускать закрывающие тэги.

- Вложенность тэгов в XML строго контролируется, поэтому необходимо следить за порядком следования открывающих и закрывающих тэгов.

- В XML учитывается регистр символов.

- Вся информация, располагающаяся между начальным и конечным тэгами, рассматривается в XML как данные, и поэтому учитываются все символы форматирования (то есть пробелы, переводы строк, табуляции не игнорируются, как в HTML).

В XML существует набор зарезервированных символов, которые должны быть заданы в XML-документе только специальным образом. Многие специалисты рассматривают XML как новую технологию интеграции программных компонент. Основными преимуществами использования XML являются:

- Интеграция данных из различных источников. XML можно использовать для объединения разнородных структурированных данных на среднем уровне трехуровневых Web-систем, баз данных.

- Локальная обработка данных. Полученные данные в формате XML можно разбирать, обрабатывать и отображать непосредственно на клиенте без дополнительных обращений к серверу.

- Просмотр и манипулирование данными в различных разрезах. Полученные данные могут обрабатываться и просматриваться клиентом различными способами в зависимости от нужд конечного пользователя.

- Возможность частичного обновления данных. С помощью XML можно обновлять только ту часть структурированных данных, которая была изменена, а не всю структуру целиком.

Все эти преимущества делают XML незаменимым инструментом для разработки гибких средств поиска информации в базах данных, мощных трехуровневых Web-приложений, а также приложений, поддерживающих транзакции. Другими словами, с помощью XML можно формировать запросы к базам данных различных структур, что позволяет осуществлять поиск информации в многочисленных несовместимых друг с другом базах данных.

2.8 Система управления базами данных MySQL

В наши дни люди часто говорят о базах данных. Компьютеры составляют неотъемлемую часть современного общества, поэтому нередко можно

услышать фразы вроде "Я поищу твою запись в базе данных". И речь идет не о больших ящиках, где хранятся груды папок, а о компьютерных системах, предназначенных для ускоренного поиска информации.

Компьютеры так прочно вошли в нашу жизнь, потому что их можно запрограммировать на выполнение утомительных, повторяющихся операций и решение задач, которые нам самим было бы не под силу решить без их вычислительной скорости и емкости информационных носителей. Помещение информации на бумагу и разработка схемы хранения бумаг в папках и картотеках – достаточно четко отработанный процесс, но многие вздохнули с облегчением, когда задача свелась к перемещению электронных документов в папки на жестком диске.

Одной из функций баз данных является упорядочение и индексация информации. Как и в библиотечной картотеке, не нужно просматривать половину архива, чтобы найти нужную запись. Все выполняется гораздо быстрее.

Не все базы данных создаются на основе одних и тех же принципов, но традиционно в них применяется идея организации данных в виде записей. Каждая запись имеет фиксированный набор полей. Записи помещаются в таблицы, а совокупность таблиц формирует базу данных.

Для работы с базой данных необходима СУБД (система управления базами данных), т.е. программа, которая берет на себя все заботы, связанные с доступом к данным. Она содержит команды, позволяющие создавать таблицы, вставлять в них записи, искать и даже удалять таблицы.

MySQL – это быстрая, надежная, открыто распространяемая СУБД. MySQL, как и многие другие СУБД, функционирует по модели "клиент/сервер". Под этим подразумевается сетевая архитектура, в которой компьютеры играют роли клиентов либо серверов. На рисунке 2.2 изображена схема передачи информации между компьютером клиента и жестким диском сервера.

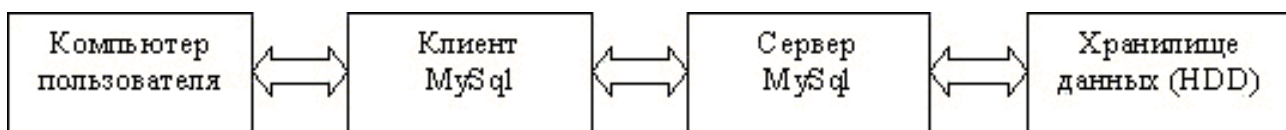


Рисунок 2.2 – Схема передачи данных в архитектуре "клиент/сервер"

СУБД управляет одной или несколькими базами данных. База данных представляет собой совокупность информации, организованной в виде множеств. Каждое множество содержит записи унифицированного вида. Сами записи состоят из полей. Обычно множества называют таблицами, а записи – строками таблиц.

Такова логическая модель данных. На жестком диске вся база данных может находиться в одном файле. В MySQL для каждой базы данных создается

отдельный каталог, а каждой таблице соответствуют три файла. В других СУБД могут использоваться иные принципы физического хранения данных.

Строки таблиц могут быть связаны друг с другом одним из трех способов. Простейшее отношение – "один к одному". В этом случае строка первой таблицы соответствует одной единственной строке второй таблицы. На диаграммах такое отношение выражается записью 1:1.

Отношение "один ко многим" означает ситуацию, когда строка одной таблицы соответствует нескольким строкам другой таблицы. Это наиболее распространенный тип отношений. На диаграммах он выражается записью 1:N.

Наконец, при отношении "многие ко многим" строки первой таблицы могут быть связаны с произвольным числом строк во второй таблице. Такое отношение записывается как N:M.

Программист, работающий с базой данных, не заботится о том, как эти данные хранятся, и приложения, взаимодействующие с СУБД, не знают о способе записи данных на диск. "Снаружи" виден лишь логический образ данных, и это позволяет менять код СУБД, не затрагивая код самих приложений.

Подобная обработка данных осуществляется посредством языка четвертого поколения (4GL), который поддерживает запросы, записываемые и исполняемые немедленно. Данные быстро утрачивают свою актуальность, поэтому скорость доступа к ним важна. Кроме того, программист должен иметь возможность формулировать новые запросы. Они называются нерегламентированными (ad hoc), поскольку не хранятся в самой базе данных и служат узкоспециализированным целям.

Язык четвертого поколения позволяет создавать схемы – точные определения данных и отношений между ними. Схема хранится как часть базы данных и может быть изменена без ущерба для данных.

Схема предназначена для контроля целостности данных. Если, к примеру, объявлено, что поле содержит целочисленные значения, то СУБД откажется записывать в него числа с плавающей запятой или строки. Отношения между записями тоже четко контролируются, и несогласованные данные не допускаются. Операции можно группировать в транзакции, выполняемые по принципу "все или ничего".

СУБД обеспечивает безопасность данных. Пользователям предоставляются определенные права доступа к информации. Некоторым пользователям разрешено лишь просматривать данные, тогда как другие пользователи могут менять содержимое таблиц.

СУБД поддерживает параллельный доступ к базе данных. Приложения могут обращаться к базе данных одновременно, что повышает общую производительность системы. Кроме того, отдельные операции могут "распараллеливаться" для еще большего улучшения производительности.

Наконец, СУБД помогает восстанавливать информацию в случае непредвиденного сбоя, незаметно для пользователей создавая резервные копии

данных. Все изменения, вносимые в базу данных, регистрируются, поэтому многие операции можно отменять и выполнять повторно.

MySQL – свободная реляционная система управления базами данных. Разработку и поддержку MySQL осуществляет корпорация Oracle, получившая права на торговую марку вместе с поглощённой Sun Microsystems, которая ранее приобрела шведскую компанию MySQL AB. Продукт распространяется как под GNU General Public License, так и под собственной коммерческой лицензией. Помимо этого, разработчики создают функциональность по заказу лицензионных пользователей. Именно благодаря такому заказу почти в самых ранних версиях появился механизм репликации.

MySQL является решением для малых и средних приложений. Входит в состав серверов WAMP, AppServ, LAMP и в портативные сборки серверов Денвер, XAMP. Обычно MySQL используется в качестве сервера, к которому обращаются локальные или удалённые клиенты, однако в дистрибутив входит библиотека внутреннего сервера, позволяющая включать MySQL в автономные программы.

Гибкость СУБД MySQL обеспечивается поддержкой большого количества типов таблиц: пользователи могут выбрать как таблицы типа MyISAM, поддерживающие полнотекстовый поиск, так и таблицы InnoDB, поддерживающие транзакции на уровне отдельных записей. Более того, СУБД MySQL поставляется со специальным типом таблиц EXAMPLE, демонстрирующим принципы создания новых типов таблиц. Благодаря открытой архитектуре и GPL-лицензированию, в СУБД MySQL постоянно появляются новые типы таблиц.

26 февраля 2008 года Sun Microsystems приобрела MySQL AB за \$1 млрд, 27 января 2010 года Oracle приобрела Sun Microsystems за \$7,4 млрд и включила MySQL в свою линейку СУБД.

Сообщество разработчиков MySQL созданы различные ответвления кода, такие как Drizzle (англ.), OurDelta, Percona Server, и MariaDB. Все эти ответвления уже существовали на момент поглощения компании Sun корпорацией Oracle.

MySQL портирована на большое количество платформ: AIX, BSDi, FreeBSD, HP-UX, Linux, MacOSX, NetBSD, OpenBSD, OS/2 Warp, SGI IRIX, Solaris, SunOS, SCO OpenServer, UnixWare, Tru64, Windows 95, Windows 98, Windows NT, Windows 2000, Windows XP, Windows Server 2003, WinCE, Windows Vista и Windows 7. Существует также порт MySQL к OpenVMS. Важно отметить, что на официальном сайте СУБД для свободной загрузки предоставляются не только исходные коды, но и откомпилированные и оптимизированные под конкретные операционные системы готовые исполняемые модули СУБД MySQL.

Максимальный размер таблиц в MySQL 3.22 до 4 Гб, в последующих версиях максимальный размер ограничивается максимальным размером файла используемой операционной системы. Размер таблицы ограничен её типом. В общем случае тип MyISAM ограничен предельным размером файла в файловой

системе операционной системы. Например в NTFS этот размер теоретически может быть до 32 эксабайт. В случае InnoDB одна таблица может храниться в нескольких файлах, представляющих единое табличное пространство. Размер последнего может достигать 64 терабайт. В отличие от MyISAM, в InnoDB имеется значительное ограничение на количество столбцов, которое можно добавить в одну таблицу. Размер страницы памяти по умолчанию составляет 16 килобайт, из которых под данные отведено 8123 байта. Размер указателя на динамические поля составляет 20 байт. Таким образом, в случае использования динамического формата строки (ROW_FORMAT=DYNAMIC), одна таблица может вместить максимум 409 столбцов типа blob или text.

MySQL имеет API для языков Delphi, C, C++, Эйфель, Java, Лисп, Perl, PHP, Python, Ruby, Smalltalk, Компонентный Паскаль и Tcl, библиотеки для языков платформы .NET, а также обеспечивает поддержку для ODBC посредством ODBC-драйвера MyODBC.

3 Создание базы данных и приложения к ней

3.1 СУБД

Для управления и хранения информации на сайте, необходима выбрать и создать базу данных. MySQL является решением для малых и средних приложений. Входит в состав серверов WAMP, для создания данного сайта использовался сервер WAMP. Что приводит к естественному решению использования реляционной базы данных и клиент серверной архитектуры MySQL. WampServer обладает также “Иконой в трее” (“TrayIcon”), которая позволяет легко управлять и конфигурировать сервера, не прибегая к файлам конфигурации.

Щелчок левой клавишей мышки на икону WampServer позволит:

- Управлять настройками Apache и MySQL.
- Менять статус в сети/не в сети (доступно каждому пользователю или же лимитировано на локальном хосте).
- Устанавливать или менять версию Apache, MySQL и PHP.
- Управлять параметрами конфигураций серверов.
- Получать доступ к логам.
- Получать доступ к файлам конфигураций.
- Создавать псевдонимы и др.

Клик правой клавишей мышки позволит поменять язык меню WampServer и перейти сразу на эту страницу.

База данных должна содержать следующие информации:

- информацию о зарегистрированных пользователях;
- лекциях, курсах, тестах;
- компонентах;
- модулях;
- правах доступа;
- галерея, видео;
- настройки системы.

В таблице 3.1 представлено полный список необходимой информации и ее краткое содержание.

Т а б л и ц а 3.1 – Перечень хранимой информации

Название выходной формы	Содержание
Сведения о пользователях	Уникальный номер, логин, пользовательская группа, электронная почта, дата регистрации
Книги	Уникальный номер, название
Лекции	Уникальный номер, название
Курсы	Уникальный номер, название
Тесты	Уникальный номер, название
Фотографии	Уникальный номер, название

Название выходной формы	Содержание
Видео	Уникальный номер, название
Форум	Уникальный номер, название
Чат (Преподаватель-Студент)	Уникальный номер, название
Настройки	Уникальный номер, название, описание, тип

3.2 Разграничение доступа

Система управления базами данных MySQL очень часто применяется для хранения важной информации на веб-сайтах. Если это обычный сайт или форум – в базе могут храниться сообщения пользователей, данные для динамических страниц, данные о посещениях, если это какой-либо интерактивный сервис, то кроме данных про доступ (конфиденциальных), там хранится и другая информация о пользователе и его действиях. Все это приводит к тому, что общая безопасность сайта, вернее всей веб-системы, зависит от того, насколько защищен именно сервер базы данных.

На обычных виртуальных хостингах каждый клиент получает свой логин и пароль, и ему доступна только одна база, в которой он может создавать произвольное количество таблиц. Один и тот же физический сервер БД используют разные клиенты, каждый из которых имеет доступ только до одной определенной базы данных. Ситуация, когда у пользователя одна база, за владение которой сражаются и "движок" форума (которому требуется создать сотню и больше таблиц), и скрипты списков рассылки, новостей, поисковый скрипт, а если еще установлена система управления контентом (CMS) или электронный магазин – тогда в этой базе возникает такое огромное количество различных таблиц, порой с очень странными и ничего не обозначающими названиями (хорошо, если два скрипта не используют таблицы с одинаковыми названиями, но разной структурой). В таких случаях очень желательно иметь возможность создать несколько отдельных баз данных, и выделить их для разных приложений (к примеру, одна база для форума, другая для электронного магазина).

Кроме сложностей с управлением несколькими сотнями таблиц в одной базе, вы столкнетесь с необходимостью ограничивать доступ разных пользователей к таблицам, базам и даже отдельным столбцам конкретной таблицы. Спешим вас успокоить – разработчики СУБД MySQL уже позаботились о подобной ситуации – в MySQL есть очень гибкий и мощный механизм управления и разграничения доступа пользователей к базам и таблицам.

Работает этот механизм, естественно, через служебные таблицы. В списке баз данных есть одна служебная база под названием "mysql", в которой хранятся в нескольких таблицах все служебные данные, необходимые для

работы сервера. Пока нас интересует только управление правами доступа на рисунке 3.1, за которые отвечают следующие таблицы:

- columns_priv – доступ на уровне столбцов таблицы;
- db – доступ к отдельным базам данных;
- host – доступ к базам с конкретных хостов/IP;
- tables_priv – доступ к отдельным таблицам базы данных;
- user – глобальные настройки доступа для конкретных пользователей.

таблица	Действие	Записи	Тип	Размер
<input type="checkbox"/> columns_priv		0	MyISAM	1.0 KB
<input type="checkbox"/> db		3	MyISAM	3.4 KB
<input type="checkbox"/> func		0	MyISAM	1.0 KB
<input type="checkbox"/> host		0	MyISAM	1.0 KB
<input type="checkbox"/> tables_priv		0	MyISAM	1.0 KB
<input type="checkbox"/> user		6	MyISAM	2.6 KB
6 таблиц(ы)	Всего	9	--	10.1 KB

Рисунок 3.1 – Администрирование правами доступа

Давайте детально рассмотрим таблицу user и специфику работы с ней, работа же с остальными таблицами аналогична и не будет сложнее.

Права, определяемые в таблице user, являются глобальными, это значит, что они применяются ко всем базам данных, к которым имеет доступ конкретный пользователь. Также в этой таблице хранятся сами имена пользователей и их пароли. Причем имена пользователей хранятся в виде открытого текста, а пароли зашифрованы функцией PASSWORD(). Сервер MySQL применяет свой метод шифрования вместо системного (если такая функция предоставляется ОС, к примеру, Linux/UNIX).

Поскольку MySQL предназначена для работы через сеть, то к серверу могут подключаться клиенты с различных хостов. Поэтому в таблице привилегий есть столбец host, в котором можно задать конкретный компьютер, IP или диапазон адресов, с которых определенный пользователь (или все, любые пользователи) могут подключаться, или же им полностью запрещен доступ. К примеру, если ваш тестовый сервер находится в локальной сети, то можно выборочно разрешить доступ к нему с отдельных компьютеров, а для всех остальных полностью заблокировать доступ.

Если для одного и того же пользователя надо разрешить доступ с различных компьютеров, то в таблице привилегий надо создать две (три или больше – сколько понадобится) записей, определяя в каждой отдельный параметр доступа. К примеру, для доступа пользователя root с компьютеров localhost и 192.168.1.138 надо в таблицу user внести две одинаковые записи, которые различаются только полем host. Хотя можно, конечно, и не делать их одинаковыми – тогда тот же пользователь, но в зависимости от места, откуда он подключился, будет иметь разные привилегии.

Значение столбца `host` можно задавать в различных форматах. Это может быть как имя хоста – `localhost`, так и IP-адрес – `192.168.1.138`. При необходимости, можно применять символы подстановки. Они аналогичны тем же. Что применяются в синтаксисе SQL-запросов. Символ `"_"` означает любой символ, а `"%"` – любое количество символов. К примеру, для разрешения доступа с любого компьютера, содержащего в имени `hostinfo.ru` надо прописать `%.hostinfo.ru` в столбце `host`. Таким же способом можно задавать и диапазоны IP-адресов. Для задания маски сети доступен также вариант написания через слеш – `192.168.1/255`. Для глобального задания доступа можно в столбце `host` прописать только символ `"%"` или оставить его пустым, что равнозначно представлен на рисунке 3.2.

←T→			Host	User	Password	Select_priv	Insert_priv	Update_priv
<input type="checkbox"/>			localhost	root		Y	Y	Y
<input type="checkbox"/>			%	root		Y	Y	Y
<input type="checkbox"/>			localhost			Y	Y	Y
<input type="checkbox"/>			%			N	N	N
<input type="checkbox"/>			localhost	s		N	N	N
<input type="checkbox"/>			localhost	__		N	N	N

Рисунок 3.2 – Глобальное задание доступа

Следующие поля – `user` и `password`, определяют имя и пароль пользователя. Эти поля уже, в отличие от `host`, различают регистр введенных символов, поэтому пользователь `Root` и `root` – это два разных пользователя. Знаки подстановки в этом поле уже не работают – имя пользователя `"%"` означает не "любой пользователь", а именно "пользователь с именем %". Для задания анонимного пользователя надо просто оставить поле пустым, иначе любой символ будет считаться именем пользователя. Так же и в поле `password` – пароль хранится в зашифрованном виде, знаки подстановки недействительны, а пустое поле означает, что пользователь вовсе не должен указывать пароль, а не то, что пароль может быть произвольным. Напомним, что просто пароль (пусть и зашифрованный) нельзя помещать в поле `password`, предварительно его надо шифровать функцией `PASSWORD("пароль_открытым_текстом")`. Например:

```
INSERT INTO user SET host="192.168.1.138", user="root",
password=PASSWORD("my_password")
```

Следующим полем является поле базы данных – `db`. В других таблицах оно также присутствует, поэтому во всех таблицах правила написания имен баз

данных должны быть одинаковыми. Разрешается использовать символы подстановки "%" и "_", пустое значение соответствует "все базы". Поле чувствительно к регистру.

Далее идет целый набор столбцов, разрешающие те или иные действия для пользователя. К примеру, `Select_priv`, `Insert_priv`, `Update_priv`, `Delete_priv`, `Index_priv` и `Alter_priv` стоит установить в "Y" (разрешено) для обычного пользователя, так как такие запросы чаще всего встречаются в обычных приложениях. Остальные возможности для большей безопасности надо отключить для всех пользователей (особенно, если есть учетная запись анонимного пользователя). Поэтому в следующих столбцах – `Create_priv`, `Drop_priv`, `Reload_priv`, `Shutdown_priv`, `Process_priv`, `File_priv`, `Grant_priv` и `References_priv` надо выставить значение "N"(запрещено), а при необходимости разрешить действия только администратору и максимально ограничить возможности подключения такого пользователя – указать конкретный IP-адрес или имя компьютера, выбрать длинный и сложный для подбора пароль.

Аналогична таблице `user` и таблица `db`, в которой можно задавать отдельные привилегии для конкретных баз данных. Таким образом, можно разрешить пользователям создавать новые таблицы только в своих базах данных и запретить даже простой просмотр чужих баз. Синтаксис всех полей полностью аналогичный таблице `user`.

При подключении пользователя сервер MySQL сначала обрабатывает таблицу `user` (записанные там правила доступа глобальные), а потом, если прав недостаточно или нельзя определить права доступа к конкретной базе – тогда уже поиск продолжается в таблице `db`. Если же и во второй таблице нет соответствующих привилегий, сервер пробует найти их в таблицах `tables_priv` и `columns_priv`. И только когда на основании информации со всех доступных таблиц сервер определяет, что клиенту не разрешено запрошенное действие, он отвергает клиентский запрос.

Таблица `host` также аналогична описанным выше, только вместо имен и паролей пользователей для идентификации привилегий служит имя хоста или IP-адрес. На первый взгляд, есть существенное дублирование информации. Ведь таблицы `db` и `host` почти равнозначны и содержат одинаковые данные. На самом деле это немного не так. Таблица `host` недоступна для изменения через операторы SQL `GRANT` и `REVOKE`, ее можно только править непосредственно. Когда сервер принимает решение о допуске клиента к базам, он сначала ищет упоминание имени хоста или IP в таблице `host`. Если там присутствует соответствующая запись, то тогда права из таблиц `host` и `db` объединяются и клиент получает некоторые привилегии. Если же в таблице `host` нет указаний на счет хоста или IP, то сервер даже не рассматривает дальше таблицу `db`, так как если клиенту не разрешено подключение с хоста, то тогда бессмысленно искать права на доступ к отдельным базам.

И, напоследок, давайте остановимся на одном нюансе при определении привилегий. Сервер MySQL, когда ищет и сравнивает привилегии, не сравнивает их напрямую. К примеру, символьное обозначение параметра имеет

больший приоритет, чем использование символов подстановки. Пример: если есть две записи – `%.hostinfo.ru` и `admin.hostinfo.ru`, то вторая запись имеет больший приоритет. Поэтому если для одного пользователя существуют разные привилегии в зависимости от хоста или других параметров, то будет выбираться первой та запись, где более точно и однозначно указан параметр. Это же касается и IP-адресов.

3.3 Создание базы данных на Wampserver

Для начала необходимо установить Wampserver. Скачать WampServer можно по адресу <http://www.wampserver.com/ru/> в разделе загрузки. Распространяется он по свободной лицензии, бесплатно и для всех желающих. Скачанный файл – обычный экзешник. Двойной клик и установка началась.

После установки в трее появится значок wampserver. Кликнув по нему правой кнопкой можно выбрать язык представлено на рисунке 3.3.



Рисунок 3.3 – Выбор языка

Далее необходимо кликнуть по `localhost` и получим всю информацию. Нам необходимо ссылка `phpmyadmin` в разделе `tools` и раздел `Your Aliases` представлено на рисунке 3.4.

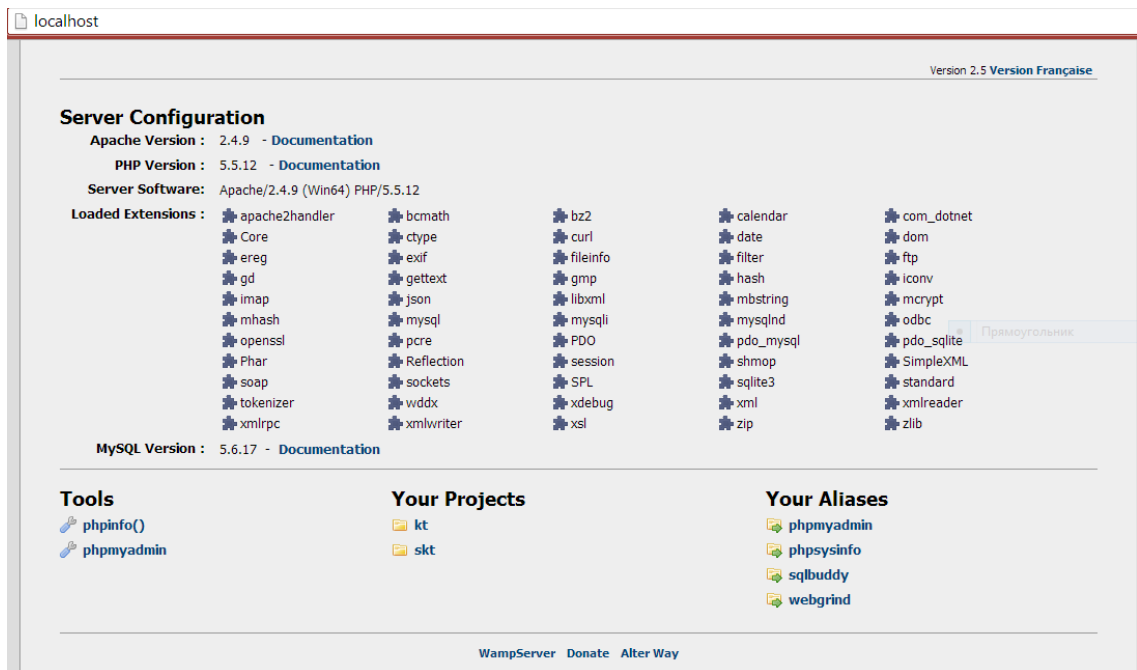


Рисунок 3.4 – Установка базы данных

После того как мы установили сервер создаем базу данных. Но для начала вамп сервере есть папка WWW, создаем в ней папку для хранения всех файлов сайта, в моем случае SKT.

Ниже представлено на рисунке 3.5 создание базы данных.

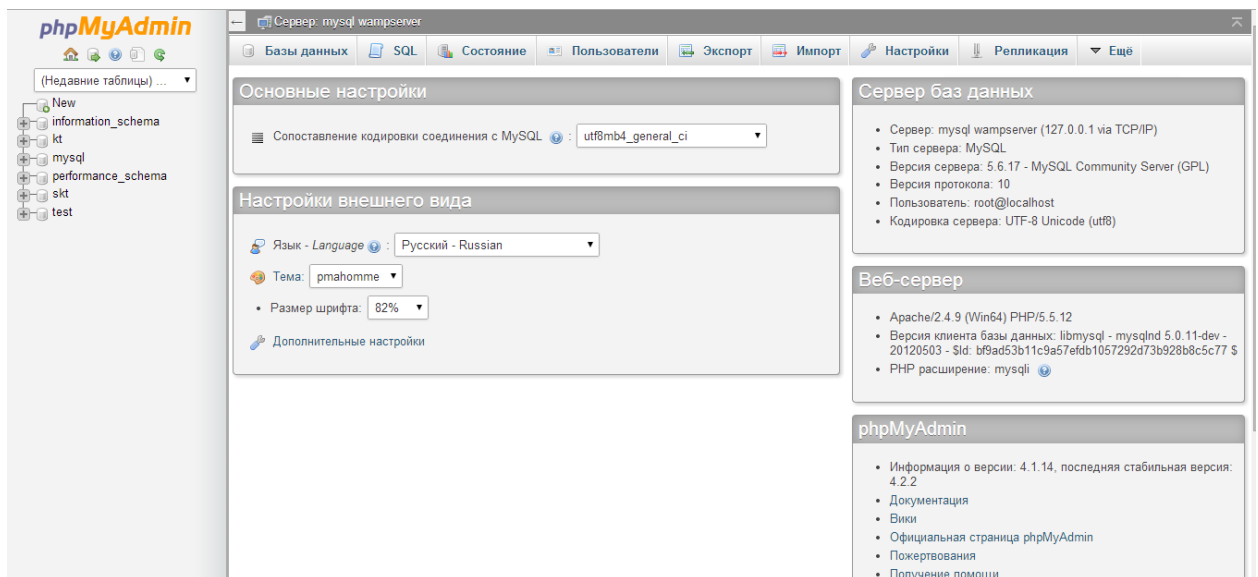


Рисунок 3.5 – Создание базы данных

После создание базы данных создаем приложение и заполняем базу данных всей необходимой информацией.

3.4 Описание программы

Приложение создано на языке PHP с применением HTML, XHTML, таблицы стилей CSS и JavaScript. Структура сайта для пользователя представляет собой информационные вкладки и категории, которая позволяет им перемещаться по страницам сайта для получения нужной информации. В зависимости от прав доступа пользователи получают информацию.

Весь программный код данной страницы представлен из файла index.php (Приложение 3.A).

По центру главной страницы располагается форма входа представлено на рисунке 3.6.

Вход

Логин

Пароль

Запомнить логин

[Забыли логин или пароль?](#)

Рисунок 3.6 – Форма "Входа"

В верхнем левом углу располагается 3 блока:

1 Основное меню, включающее в себя (представлено на рисунке 3.7):

- Форум;
- Чат.

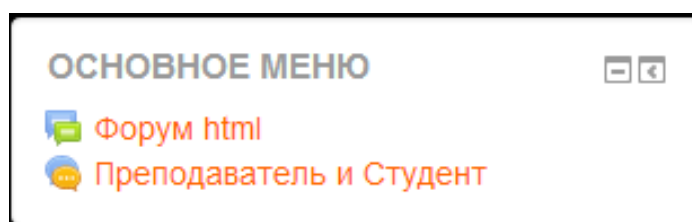


Рисунок 3.7 – Форма "Основное меню"

2 Навигация включает в себя (представлено на рисунке 3.8):

- Форум html;
- Чат Преподаватель и Студент;
- Курсы.

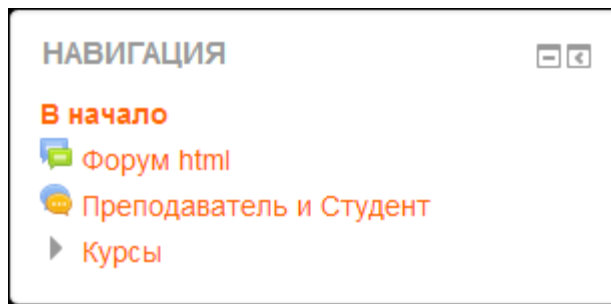


Рисунок 3.8 – Форма "Навигация"

3 Доступные курсы показывает доступные курсы на сегодняшний день представлено на рисунке 3.9.

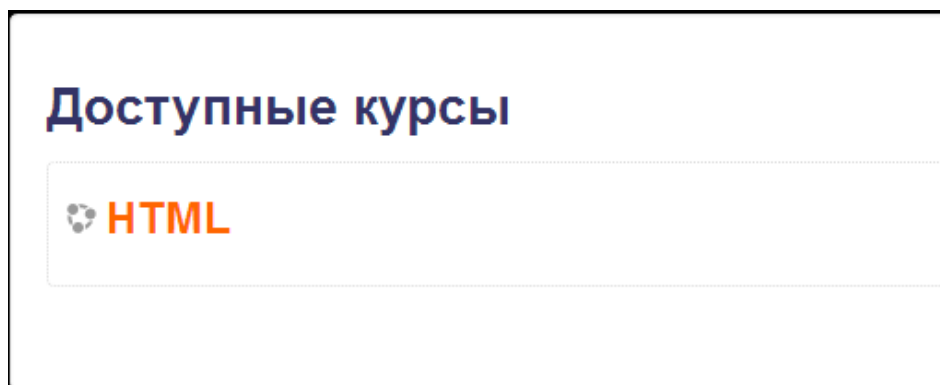


Рисунок 3.9 – Форма "Доступные курсы"

Ниже представлен Рисунок "Общей картины главной страницы" (Рисунок 3.10).

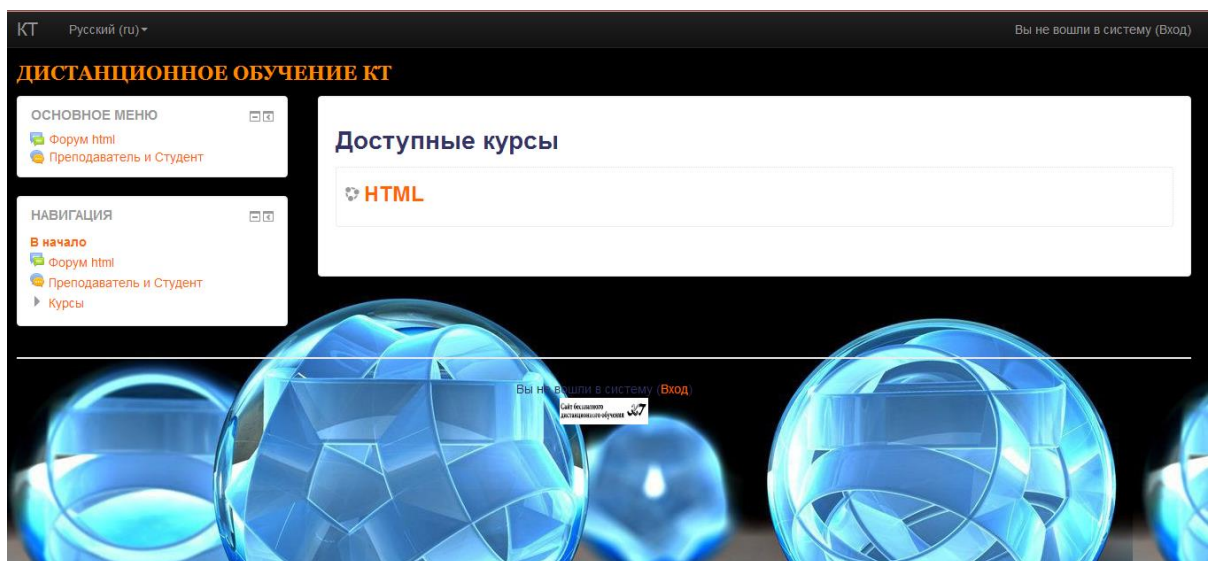


Рисунок 3.10 – Форма "Главная страница"

При входе на сайт с логином пользователя доступны следующие блоки информации представлено на рисунке 3.11.

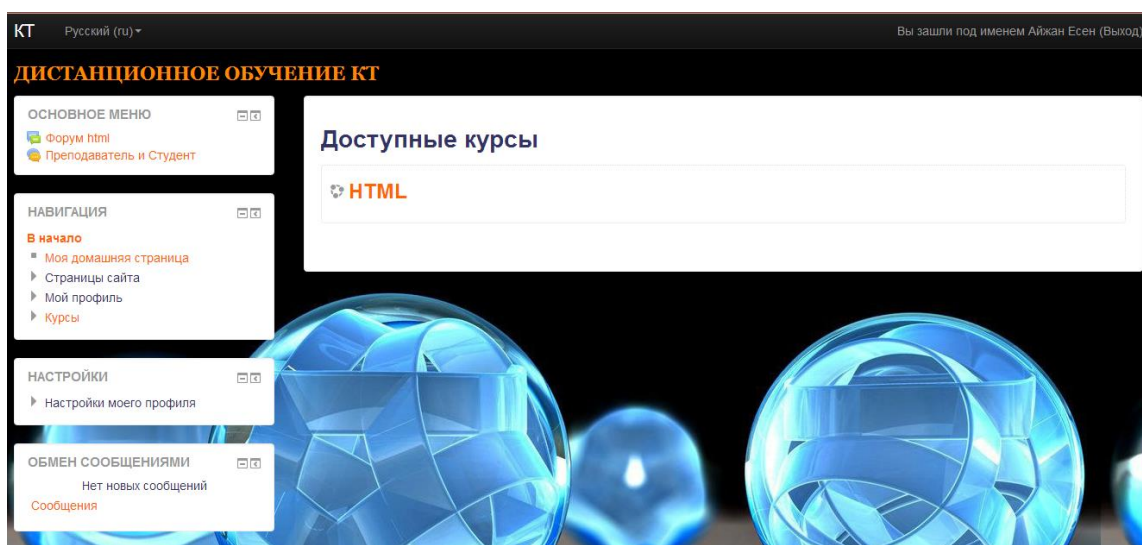


Рисунок 3.11 – Окно пользователя

В разделе Основное меню студент может участвовать в форуме открытом Преподавателем и в чате индивидуально только между преподавателем и студентом представлены на рисунке 3.12, 3.13.

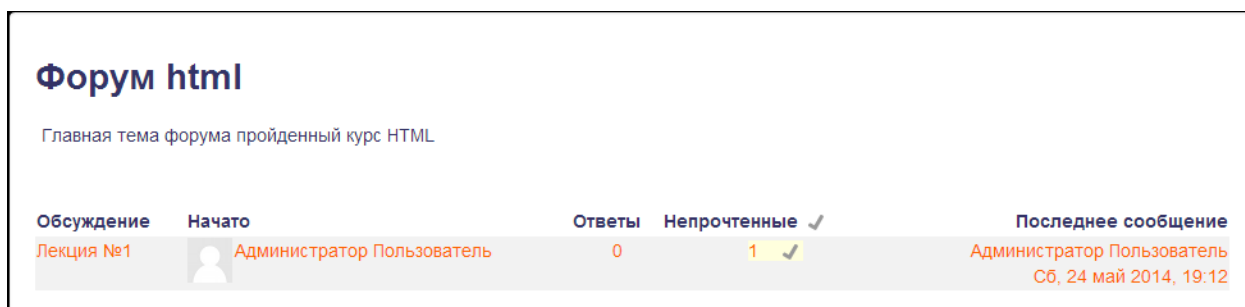


Рисунок 3.12 – Форма "Форум"



Рисунок 3.13 – Форма "Чат"

Студент может пройти активные курсы , на который его записал преподаватель, представлено на рисунке 3.14.

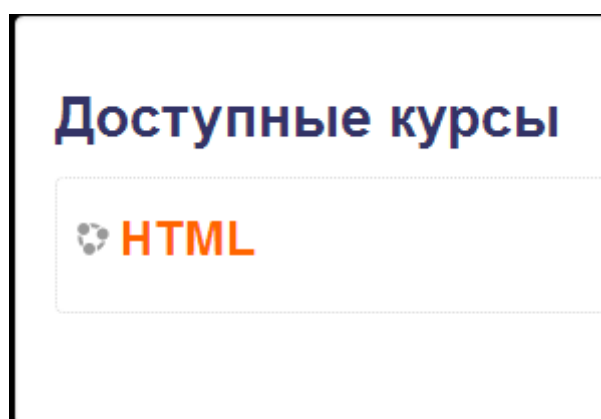


Рисунок 3.14 – Форма "Доступные курсы"

В курсы входит лекции представлено на рисунке 3.15.

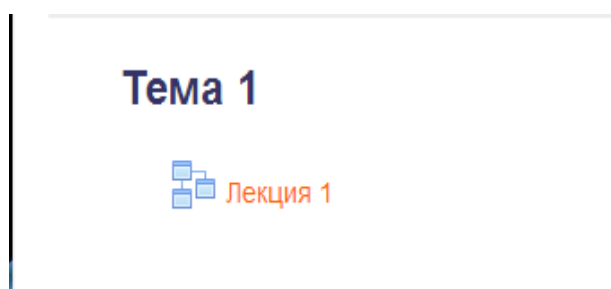


Рисунок 3.15 – Лекции

По каждому курсу можно пройти тесты представлено на рисунке 3.16.

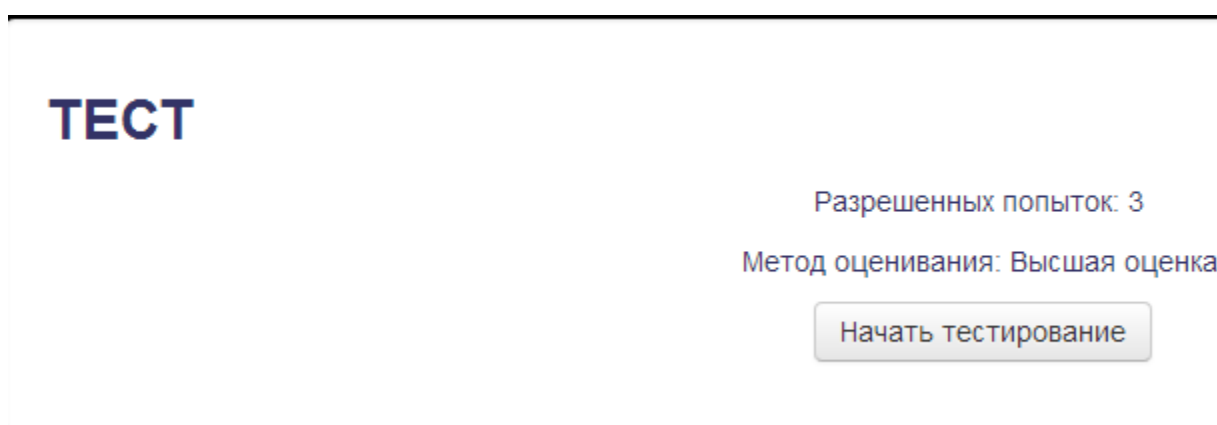


Рисунок 3.16 – Форма "Тесты"

Представлена страница теста на рисунке 3.17.

The image shows a test interface with two questions. Each question is in a light blue box with a grey sidebar on the left containing the question number, status, and score. Question 1 asks how to combine table cells, with options for CELLpadding, CELLspacing, ROWspan, and COLspan. Question 2 asks when frame scrolling is always disabled, with options for no, auto, and yes.

Вопрос 1
Пока нет ответа
Балл: 1,00
Отметить вопрос

Как объединить несколько ячеек таблицы?

Выберите один или несколько ответов:

- a. с помощью атрибута CELLpadding
- b. с помощью атрибута CELLspacing
- c. с помощью атрибута ROWspan
- d. с помощью атрибута COLspan

Вопрос 2
Пока нет ответа
Балл: 1,00
Отметить вопрос

В каких случаях возможность прокрутки окна фрейма будет всегда недоступна?

Выберите один или несколько ответов:

- a. <FRAME scrolling="no">
- b. <FRAME scrolling="auto">
- c. <FRAME scrolling="yes">

Рисунок 3.17 – Страница теста

Представлен блок обмена сообщениями Рисунок 3.18.

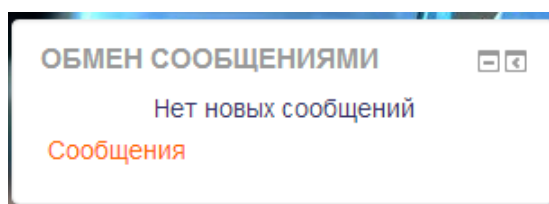


Рисунок 3.18 – Форма "Обмен сообщениями"

Представлен блок профиль, на нем указывается первый и последний вход студента Рисунок 3.19.



Рисунок 3.19 – Профиль

Представлен блок настройка, в настройке студент может менять профиль
Рисунок 3.20.

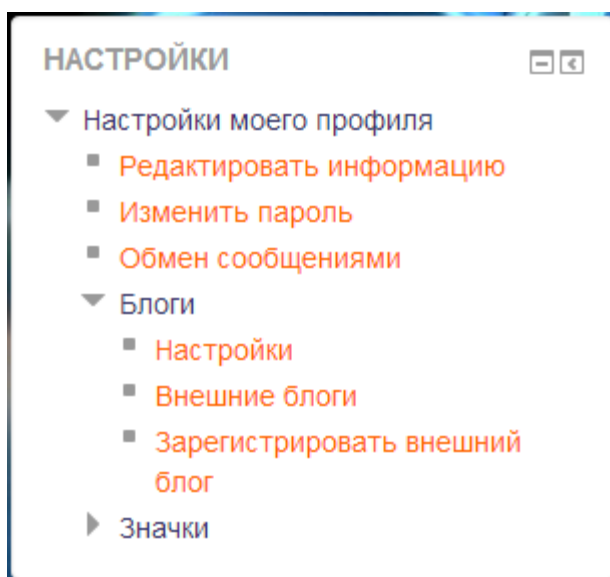


Рисунок 3.20 – Форма "Настройки"

Представлено окно администратора на рисунке 3.21.

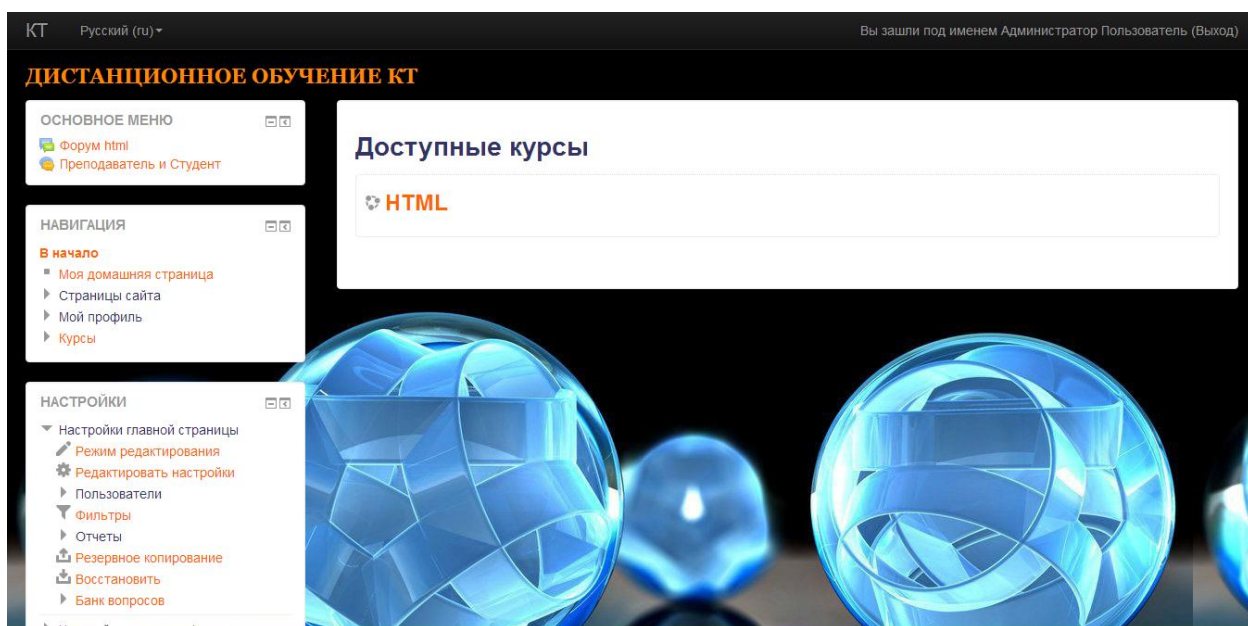


Рисунок 3.21 – Форма "Администрирование"

Представлено окно настройка, в котором администратор может настраивать главную страницу, профили пользователей, отчеты, резервное копирование, восстановление, банк вопросов, рисунок 3.22.

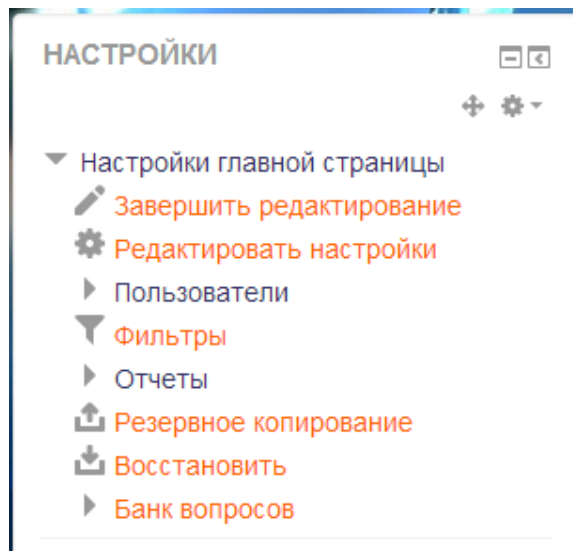


Рисунок 3.22 – Настройка

Представлено окно настройка профиля администратора на рисунке 3.23.

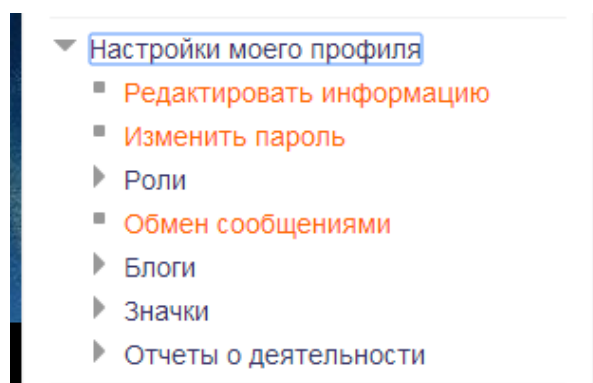


Рисунок 3.23 – Меню настройки профиля

Представлено меню администрирования на рисунке 3.24.

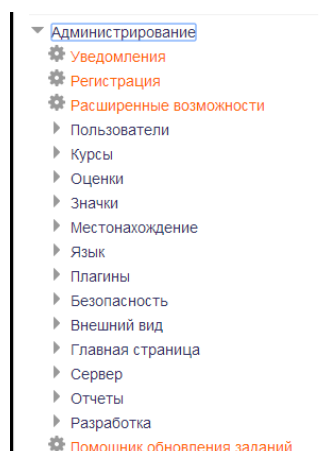


Рисунок 3.24 – Меню администрирования

4 Технико-экономическое обоснование

4.1 Описание работы и обоснование необходимости

Тема дипломной работы – «Сайт бесплатного дистанционного обучения для кафедры КТ». Данный сайт разрабатывается для всех студентов кафедры КТ, которые бесплатно могут проходить курсы на этом сайте. Сайт разрабатывается на бесплатной основе.

Цель работы – разработка сайта, позволяющего осуществлять дистанционное обучение студентов. Главным отличием будет расширенная функциональность. В данном разделе приводится рассмотрение экономической составляющей реализации сайта, отражающей временные, трудовые и финансовые затраты. В работе задействован: инженер-разработчик.

4.2 Оборудование и программное обеспечение используемое в работе

Оборудование, использованное при разработке сайта представлено в таблице 4.1.

Т а б л и ц а 4.1 – Перечень оборудования, необходимого для разработки сайта бесплатного дистанционного обучения

Наименование	Тип	Количество, Единиц	Цена за единицу, в тенге	Общая сумма, тенге
Ноутбук	ASUS S56C	1	145 000	145 000

Программное обеспечение, использованное при разработке сайта, представлено в таблице 4. 2.

Т а б л и ц а 4.2 – Перечень использованного программного обеспечения, необходимого для разработки сайта бесплатного дистанционного обучения

Программное обеспечение	Стоимость, тенге
Sublime Text	10000
NotePad++	10000
Dreamveawer	90000
Photoshop	35000
Итого	145000

4.3 Определение объема и трудоемкости разработки программного обеспечения

Базой для расчета плановой сметы затрат на разработку ПО является объем программного продукта. Общий объем (V_o) программного продукта определяется исходя из количества и объема функций, реализуемых программой

$$V_o = \sum_{i=1}^n V_i \quad (4.1)$$

где V_i – объем отдельной функции ПО;
 n – общее число функций.

$V_o=2200$ строк исходного кода

На основе информации о функциях разрабатываемого ПО определяется объем функций и общий объем ПО, который уточняется (корректируется) с учетом условий разработки ПО в организации в соответствии с каталогом функций программного обеспечения (Приложение Б.). Уточненный объем ПО (V_y) рассчитывается по формуле

$$V_y = \sum_{i=1}^n V_{yi} \quad (4.2)$$

где V_{yi} – уточненный объем отдельной функции ПО (LOC).

$V_{yi}=2450$

Общая трудоемкость небольших проектов рассчитывается по формуле

$$T_o = T_n \times K_c \times K_T \times K_H \quad (4.3)$$

где K_c – коэффициент, учитывающий сложность ПО (Приложение В).

$K_c=0,12$

K_T – поправочный коэффициент, учитывающий степень использования при разработке стандартных модулей в соответствии с значением поправочного коэффициента, учитывающего использование стандартных модулей типовых программ и ПО (K_T)т (Приложение Г).

$K_T=0,6$

K_n – коэффициент, учитывающий степень новизны ПО в соответствии с поправочным коэффициентом, учитывающие новизну ПО (K_n) (Приложение Д).

$$K_n=1$$

Нормативная трудоемкость разработки ПО (T_n) определяется на основании принятого к расчету объема (V_y) и категории сложности, которая уточняется с учетом сложности и новизны проекта и степени использования стандартных модулей при разработке. Данное ПО соответствует 3-й категории сложности ПО в соответствии с укрупненными нормами времени на разработку ПО в зависимости от уточненного объема ПО и группы сложности ПО (Приложение Б).

$$T_n=414$$

Таким образом общая трудоемкость проекта составляет

$$T_o=414*0,12*0,6*1=30,1$$

Численность исполнителей проекта ($Ч_p$) рассчитывается по формуле

$$Ч = T_o / (T_p \times \Phi_{эф}) \quad (4.4)$$

где $\Phi_{эф}$ – эффективный фонд времени работы одного работника в течение года (дн.);

T_o – общая трудоемкость разработки проекта (чел./дн.);

T_p – срок разработки проекта (лет).

Срок разработки проекта (T_p) определяется по формуле

$$T_p = T_o / (Ч_p \times \Phi_{эф}) \quad (4.5)$$

Эффективный фонд времени работы одного работника ($\Phi_{эф}$) рассчитывается по формуле

$$\Phi_{эф} = D_r - D_n - D_v - D_o \quad (4.6)$$

где D_r – количество дней в году;

D_n – количество праздничных дней в году;

D_v – количество выходных дней в году;

D_o – количество дней отпуска.

$$\Phi_{эф}=365-14-104-24=223 \text{ дней}$$

$$T_p = 30,1 / (1 * 223) = 0,15 \text{ лет}$$

$$Ч = 30,1 / (0,15 * 223) = 1 \text{ чел}$$

Результаты выполненных расчетов представлены в таблице 4.3

Т а б л и ц а 4.3 – Сводные результаты расчета трудоемкости

Название	Условное обозначение	Значение
Общий объем ПО	V_o	2200
Уточненный объем ПО V_v	V_v	2450
Нормативная трудоемкость разработки	T_n	414
Общая трудоемкость	T_o	30,1
Эффективный фонд времени работы	$\Phi_{эф}$	223 дня
Срок разработки проекта	T_p	0,15 лет
Численность исполнителей проекта	$Ч_p$	1 чел

4.4 Расчет затрат на разработку информационных технологий

Расчет полных затрат на разработку проектного решения в виде информационных технологий ($C_{пi}$) осуществляется по формуле

$$C_{пi} = Z_{фот} + Z_{сзи} + M_i + P_{ci} + P_{mi} + P_{нки} + П_{зи} + P_{ни} \quad (4.7)$$

где $Z_{фот}$ – общий фонд оплаты труда разработчиков, тенге;

$Z_{сзи}$ – отчисления по социальному налогу, тенге;

M_i – затраты на материалы, тенге;

P_{ci} – затраты на специальные программные средства, необходимые для разработки проектного решения, тенге;

P_{mi} – затраты, связанные с эксплуатацией техники, тенге;

$P_{нки}$ – затраты на научные командировки, тенге;

$П_{зи}$ – прочие затраты, тенге;

$P_{ни}$ – накладные расходы, тенге.

Размер фонда оплаты труда разработчиков ($Z_{фот}$) рассчитывается по формуле

$$Z_{фот} = Z_{oi} + Z_{di} \quad (4.8)$$

где Z_{oi} – основная заработная плата, тенге;

Z_{di} – дополнительная заработная плата, тенге.

Основная заработная плата исполнителей на конкретное ПО рассчитывается по формуле

$$Z_{oi} = \sum T_{ci} \times T_{ч} \times \Phi_{п} \times K \quad (4.9)$$

где n – количество исполнителей, занятых разработкой конкретного ПО;

T_{ci} – часовая тарифная ставка i -го исполнителя (тыс.тенге);

$\Phi_{п}$ – плановый фонд рабочего времени исполнителя (дней) (0,15лет или 440раб. часов);

$T_{ч}$ – количество часов работы в день (час) (8 часов);

K – коэффициент премирования (1,2).

По данным о специфике и сложности выполняемых функций составляется штатное расписание группы специалистов-исполнителей, участвующих в разработке ПО, с определением образования, специальности, квалификации и должности представлено в таблице 4.4.

Т а б л и ц а 4.4 – Сведения по работникам, задействованным в проекте

Специалист - Исполнитель	Количество, человек	Заработная плата в месяц, тенге
Программист	1	200 000
Итого		200 000

Часовая тарифная ставка рассчитывается путем деления месячной тарифной ставки на установленную при 40-часовой недельной норме рабочего времени расчетную среднемесячную норму рабочего времени в часах (Φ_p)

$$T_{ч} = T_{м} / \Phi_p \quad (4.10)$$

где $T_{ч}$ – часовая тарифная ставка (тыс.тенге);

$T_{м}$ – месячная тарифная ставка (тыс.тенге).

По формуле 4.10 можно определить часовую тарифную ставку исполнителя

$$T_{ч} = 200000 / 440 = 454,6 \text{ тг/час}$$

Рассчитываем основную заработную плату

$$Z_{oi} = 454,6 * 55 * 8 * 1,20 = 240000 \text{ тг}$$

Дополнительная заработная плата рассчитывается по формуле

$$З_{дi} = З_{oi} \times Н_{д}/100 \quad (4.11)$$

где $Н_{д}$ – коэффициент дополнительной заработной платы разработчиков=11.

$$З_{дi}=240000 *0,1=24000 \text{ тг.}$$

Рассчитаем фонд оплаты труда по формуле 4.8

$$З_{ФОТ}=240000+24000 =264000 \text{ тг.}$$

Социальный налог составляет 11% (ст. 358 п. 1 НК РК) от дохода работника, и рассчитывается по формуле

$$З_{сzi} = (ФОТ-ПО) \times 11\% \quad (4.12)$$

где ПО - пенсионные отчисления, которые составляют 10% от ФОТ и социальным налогом не облагаются

$$ПО = ФОТ \times 10\%. \quad (4.13)$$

$$ПО=264000 *0,1=26400 \text{ тг.}$$

Таким образом социальный налог составит

$$З_{сzi}=(264000 -26400)*0,11= 26136\text{тг.}$$

Величина затрат на материалы на основании исходных данных определяется по формуле

$$M_i = (З_{очн.} \times Н_{мз})/ 100\% \quad (4.14)$$

где $Н_{мз}$ – норма расхода материалов от основной заработной платы (3-5%).

$$M_i =(240000*3)/100=7200 \text{ тг}$$

Расходы по статье «Спецоборудование» (P_{ci}) включают затраты средств на приобретение вспомогательных специального назначения технических и программных средств, необходимых для разработки конкретного ПО, включая расходы на их проектирование, изготовление, отладку, установку и эксплуатацию

$$P_{ci} = \sum Ц_{ci} \quad (4.15)$$

где $Ц_{ci}$ – стоимость конкретного специального оборудования (тыс.тенге);

n – количество применяемого специального оборудования.

Первоначальная стоимость ПО указана в таблице 4.2 $P_{ci} = 145000$ тг

Расходы по статье «Машинное время» (P_{mi}) включают оплату машинного времени, необходимого для разработки и отладки ПО, которое определяется по нормативам (в машино-часах) на 100 строк исходного кода (H_{mb}) машинного времени в зависимости от характера решаемых задач и типа ПК

$$P_{mi} = C_{mi} \times (V_{oi}/100) \times H_{mb} \quad (4.16)$$

где C_{mi} – цена одного машино-часа (тыс.тенге) = 454,6 тг;

V_{oi} – общий объем ПО (строк исходного кода) = 2450;

H_{mb} – норматив расхода машинного времени на отладку 100 строк исходного кода (машино-часов) = 12 ч/100 строк кода в соответствии с оценкой значений среднего машинного времени на отладку 100 строк исходного кода без применения ПО (Приложение Б).

$$P_{mi} = 454,6 * (2450/100) * 12 = 133636,4 \text{ тг}$$

Расходы по статье «Научные командировки» (P_{nki}) на конкретное ПО определяются по нормативу, разрабатываемому в целом по организации, в процентах к основной заработной плате

$$P_{nki} = Z_{oi} \times H_{rnk} / 100 \quad (4.17)$$

где H_{rnk} – норматив расходов на командировки в целом по организации в (%), в дипломной работе нужно брать 30%.

Подставляем все данные в формулу 4.17 получаем

$$P_{nki} = 240000 * 0,3 = 72000 \text{ тг}$$

Расходы по статье «Прочие затраты» (P_{zi}) на конкретное ПО включают затраты на приобретение и подготовку специальной научно-технической информации и специальной литературы. Определяются по нормативу, разрабатываемому в целом по организации, в процентах к основной заработной плате

$$P_{zi} = Z_{oi} \times H_{pz} / 100 \quad (4.18)$$

где H_{pz} – норматив прочих затрат в целом по организации в (%), в дипломной работе нужно брать 20%.

Подставляем все данные в формулу 4.18 получаем

$$P_{zi} = 240000 * 0,2 = 48000 \text{ тенге}$$

Затраты по статье «Накладные расходы» ($P_{ни}$). Норматив устанавливается в целом по организации

$$P_{ни} = Z_{oi} \times N_{рн} / 100\% \quad (4.19)$$

где $P_{ни}$ – накладные расходы на конкретную ПО (тыс.тенге);

$N_{рн}$ – норматив накладных расходов в целом по организации в (%), в дипломной работе нужно брать 70%.

Подставляем все данные в формулу 4.19 получаем

$$P_{ни} = 240000 \times 0,7 = 168000 \text{ тг}$$

$$C_{пн} = 264000 + 26136 + 7200 + 145000 + 133636,4 + 72000 + 48000 + 168000 = 863972,4 \text{ тг}$$

Результаты расчета себестоимости разработки сайта и их структура представлены в таблице 4.5 и на рисунке 4.1.

Т а б л и ц а 4.5 - Сводные результаты расчета затрат на разработку сайта "Сайт бесплатного дистанционного обучения для кафедры КТ"

Затраты на разработку	Условное обозначение	Значение, тенге	В процентах от общей суммы
Фонд оплаты труда	$Z_{фот}$	264000	30,6
Социальный налог	$Z_{сзи}$	26 136	3,0
Материалы	M_i	7200	0,8
Спецоборудование	$P_{си}$	145 000	16,8
Машинное время	$P_{ми}$	133636,4	15,5
Научные командировки	$P_{нки}$	72000	8,3
Прочие затраты	$P_{зи}$	48 000	5,6
Накладные расходы	$P_{ни}$	168 000	19,4
Итого:		863972,4	100

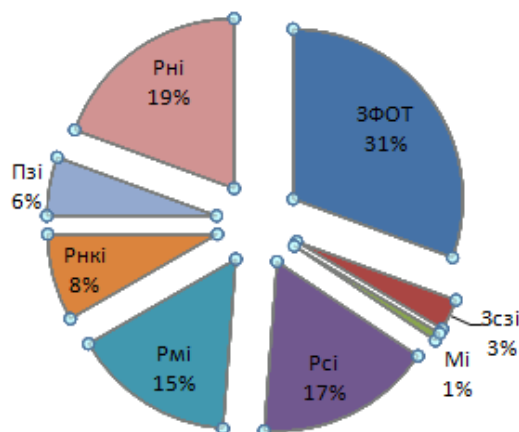


Рисунок 4.1 - Структура себестоимости сайта

4.5 Расчет цены программного продукта

Расчет цены ПП, который разработан одной организацией по заказу другой и не предназначен для тиражирования, осуществляется по формуле

$$C_{ПП} = Z_{РПР} + П_n + НДС \quad (4.25)$$

где $C_{ПП}$ – цена программного продукта, тенге;

$Z_{РПР}$ – затраты на разработку проектного решения, в данном случае программного продукта, тенге;

$П_n$ – планируемая прибыль, тенге;

$НДС$ – налог на добавленную стоимость, тенге.

Планируемая прибыль рассчитывается по формуле

$$П_n = Z_{РПР} \times R_{НПП} \quad (4.26)$$

где $R_{НПП}$ – нормативная рентабельность ПП, определяемая организацией.

$$R_{НПП} = 20\%.$$

Таким образом планируемая прибыль составляет

$$П_n = C_{пп} \times 0,2 = 863972,4 \times 0,2 = 172794,5 \text{ тг}$$

НДС, начисленный на ПП, определяется следующим образом

$$НДС = (Z_{РПР} + П_n) \times k_{НДС} \quad (4.27)$$

где $k_{НДС}$ – ставка налога на добавленную стоимость

$$Z_{РПР} = C_{пп} = 1047940 \text{ тг}$$

Таким образом НДС, начисленный на ПП составляет

$$НДС = (863972,4 + 172794,5) \times 0,12 = 124\,412 \text{ тг}$$

Цена ПП составляет

$$C_{ПП} = 863972,4 + 172794,5 + 124\,412 = 1\,161\,179 \text{ тг}$$

Вывод

Себестоимость сайта составляет 863972,4 тенге. Основную долю расходов составляют затраты на оплату труда -31%, накладные расходы -19%.

Цена реализации - 1 161 179 тенге(6 380\$). Стоимость аналогичных сайтов начинается от 1 820 000тенге (10 000\$). Данный сайт "Сайт бесплатного дистанционного обучения для кафедры КТ" является конкурентоспособным .

5 Безопасность жизнедеятельности

5.1 Анализ условий труда

Профессия программиста изначально связано с работой на компьютере, вся профессиональная деятельность осуществляется только посредством компьютера. В данной главе описаны основные вредные факторы при работе с компьютером.

К основным вредным факторам при работе с компьютером относят: длительное сидячее положение, нагрузка на зрение, перегрузка кистевых суставов, возможность заболеваний органов дыхания, аллергии, нарушение нормального течения беременности и др.

Длительное сидячее положение приводит к напряжению мышц шеи, головы, рук и плеч, остеохондрозу. Длительное сидячее положение еще приводит к застою крови в тазовых органах и, как следствие, к простатиту и геморрою. Не секрет, что малоподвижный образ жизни приводит к ожирению. Остеохондроз возникает при нарушении межпозвоночных дисков, которое приводит к выпячиванию в какую либо сторону (грыже межпозвоночного диска). Грыжа может повредит спинной мозг и нервные отростки. Последствия могут быть самыми разнообразными, от болей в спине и конечностях, до паралича конечностей и смерти. Одна из распространенных причин остеохондроза – дистрофия мышц спины. Программист, ведущий в основном сидячий образ жизни, вполне может заболеть остеохондрозом. Признаки начинающегося заболевания: дискомфорт в спине и болевые ощущения, головные боли, нарушения работы внутренних органов. К факторам риска заболевания геморроем относят: сидячий образ жизни, ожирение, чрезмерное употребление копченых, острых, соленых и пряных продуктов, воспалительные заболевания малого таза и др. Ожирение возникает из-за нерационального питания, малоподвижного и в том числе сидячего образа жизни, неадекватной реакции на стрессовые ситуации, чрезмерно долгий сон, применение гормональных препаратов, перегрузки организма пищевыми жирами и др. Ожирение приводит к увеличению нагрузки на сердце, изменению конфигурации и положения сердца в грудной полости, повышению содержания холестерина в крови, в результате он откладывается на стенках сосудов (атеросклероз). Повышенное скопление жира внутри грудной полости влияет на работу органов дыхания, что приводит к появлению одышки и гипоксии органов и тканей.

Нагрузка на зрение. Человеческий глаз реагирует на самую мелкую вибрацию текста и на мерцание экрана. Мышцы глаза, управляющие хрусталиком, находятся в постоянном напряжении, что обязательно приводит к потере остроты зрения. Немаловажное значение для профилактики зрительных дисфункций оказывают: правильный или рекомендуемый подбор цвета, шрифтов, компоновки окон в используемых приложениях, ориентация дисплея

монитора. Длительная работа за компьютером – это огромная нагрузка на глаза, поскольку изображение на мониторе складывается не из непрерывных линий, как на бумаге, а из отдельных точек, светящихся и мерцающих. У пользователя неизбежно ухудшается зрение, глаза начинают слезиться, появляется головная боль, утомление, изображение двоится и искажается.

Перегрузка суставов кистей рук приводит главным образом к такому явлению, как синдром запястного канала (см. выше).

Работа за компьютером и стрессы. Стресс – это эмоциональные переживания, внутреннее напряжение, вызванные событиями в жизни. Стресс возникает, в первую очередь, при потере или повреждении информации. Причины: отсутствие резервных копий, компьютерные вирусы, поломки жестких дисков, рабочие ошибки. Иногда стрессы являются причиной инфарктов. Стрессы бывают эмоционально положительными и эмоционально отрицательными, кратковременными и долгосрочными, острыми и хроническими, физиологическими и психологическими (информационными и эмоциональными). Работа за компьютером является одним из факторов, вызывающих стресс (стрессором). Реакция организма на стресс представляет собой запуск биохимических процессов, которые направлены на подавление экстремальной ситуации. Стрессовые ситуации и связанные с ними переживания вызывают в организме многочисленные негативные сдвиги.

Работа за компьютером и органы дыхания. Заболевания органов дыхания в данном контексте носят в основном аллергический характер. Это объясняется тем, что за время долгой работы компьютера корпус и платы последнего выделяют в воздух ряд вредных веществ, а так же компьютер создает вокруг себя электростатическое поле, притягивающее пыль, которая оседает в легких. Так же компьютер деионизирует окружающую среду и уменьшает влажность воздуха. Аллергия – это повышенная чувствительность организма к различным раздражителям, проявляющаяся в специфических реакциях при контакте с ними. Это вызывает такие симптомы аллергии как ринит, слезоточивость, кожная сыпь, анафилактический шок. Компьютер является довольно серьезным источником ряда аллергенов. К примеру, корпус монитора, нагреваясь до 45-50°C начинает выделять в воздух пары трифенилфосфата. Помимо монитора нагревается и материнская плата, блок питания, процессор, видеокарта, которые так же могут выделять в окружающую среду вредные органические и неорганические вещества (фтор, хлор, фосфорсодержащие). Кроме того, в компьютере есть очень много мест, где скапливается пыль и грязь, размножаются микробы и грибки. Пыль получает от экрана монитора слабый статический заряд, которого достаточно, что бы пыль оседала на теле пользователя и в его дыхательных путях. При аллергии повышается утомляемость, усиливается раздражительность и снижается иммунитет. Аллергия провоцирует ряд заболеваний: экзему, гемолитическую анемию, бронхиальную астму и др. Наиболее тяжким проявлением аллергии является анафилактический шок, который сопровождается затруднением дыхания,

судорогами, потерей сознания, понижением артериального давления и зачастую смерти.

Шум ПК часто мешает собраться с мыслями, раздражает - особенно при отсутствии других источников шума. К чести производителей ПК нужно отметить, что лучшие из них учитывают акустику при конструировании систем. Однако слишком много ПК до сих пор строится на дешевых охлаждающих компонентах, создающих источники шума

Главными источниками шума в компьютере являются моторы и вращающиеся детали, такие как жесткий диск, вентиляторы охлаждения корпуса, процессора и блока питания. Особенно много шума создают последние два. Кроме того, вентиляторы и жесткие диски являются источником вибрации, шум от которой усиливается конструкцией корпуса и поверхностью, на которой установлены устройства.

Чем выше быстродействие процессоров, жестких дисков и видеокарт, тем больше они выделяют тепла и тем сильнее нуждаются в охлаждении - а охлаждение, как правило, связано с шумом. Но есть ряд способов уменьшить шум, сохранив адекватное охлаждение.

Для уменьшения звука издаваемым принтером, необходимо включить функцию бесшумного режима (если принтер поддерживает данную функцию). Выберите этот вариант, чтобы сделать тише шум, например, при работе принтера ночью. При использовании этой функции может снизиться скорость печати.

Многие производители компьютерных комплектующих пытаются уменьшить шум компьютера различными способами: создают корпуса со звукоизоляционными материалами, устанавливают большие кулера с меньшими оборотами, делают пассивное охлаждение. В результате со временем шум все равно появляется. Первое, что необходимо сделать, это открыть боковую крышку системного блока и прислушаться, если:

1 Шумит кулер на процессоре

Основная причина шума кулера на процессоре - это скопление пыли на радиаторе. От пыли ухудшается теплоотдача радиатора, а кулер на процессоре управляется программно, в зависимости от температуры. Таким образом, от плохого отвода тепла, кулер начинает работать быстрее. Так же, со временем, от пыли, смазка кулера портится и возникает неприятный шум. В данной ситуации, что бы уменьшить шум надо его смазать. Ещё, если у вас штатный кулер 80 мм. или 10 мм. рекомендуем заменить на более тихий и эффективный 120 мм. кулер для процессора.

2 Шум кулера на системном корпусе

Уменьшить шум кулера можно так же чисткой, смазкой или заменой на более эффективные и тихие 120 - 140 мм. кулера. Далее, поставить вместо обычных болтов резиновые, или силиконовую прокладку между кулером и корпусом. А вообще к кулерам на корпусе надо подходить грамотней, так как они являются основным источником засорения пылью. Обратите внимание, если кулер работает на вдув в корпус, то защиту от пыли можно поставить

перед кулером и этого будет достаточно. Если кулер работает на выдув из корпуса, то пыль будет попадать из всех мелких отверстий, из-за отрицательного давления в корпусе. В данной ситуации, защиту надо ставить на всех отверстиях компьютера кроме кулера на выдув.

3 Шумит кулер на видеокарте

Шумит кулер на видеокарте в основном из-за пыли и плохого отвода тепла от радиатора, так как кулер на видеокарте так же управляется программно, в зависимости от температуры. В данной ситуации, что бы уменьшить шум кулера надо чаще смазывать его и чистить радиатор от пыли.

4 Шумит кулер в блоке питания

Кулер в блоке питания, начинает шуметь из-за отсутствия смазки и засорения пылью. Обратите внимание, перед смазкой кулера и чисткой блока питания необходимо отсоединить все провода и снять его с корпуса полностью. Еще, рекомендуем покупать блок питания отдельно от корпуса компьютера, так как у блока питания встроенного в корпус низкое качество кулера

5 Уменьшение шума от жестких дисков

По опыту могу сказать, что шум жесткого диска со временем увеличивается. А снизить шум можно одним способом, это в местах крепления поставить резиновые прокладки. Использовать звукоизоляционный материал не рекомендуем, так как жесткий диск при работе выделяет тепло.

6 Уменьшение шума от вибраций корпуса

Здесь шум зависит от качества сборки и толщины стали. Если у вас дешевый корпус, рекомендуем поставить следующие аксессуары: Звукоизоляционный материал AcoustiPack, это материал-наклейка устанавливается на корпуса для утолщения стенок. Этот материал вибрацию не уберет, но общий шум изнутри компьютера задержит. Далее, заменить пластиковые ножки на мягкие силиконовые.

Исходя из выше описанного можно уточнить, что из 6-и источников шума, 4-е по вине пыли. Из этого следует, что бы компьютер был тихий, нужно бесконечно долго чистить его от пыли, смазывать кулера и т.д. Но есть более легкий способ уменьшения шума, достаточно установить на корпус компьютера пылевой фильтр.

Вентиляция помещения

Норма пространства помещения на одного человека составляет 6 м². В офисе предполагается наличие воздуховодов, которые размещены за подвесными потолками. Воздух плавно подается и удаляется из офиса либо через вентиляционные решетки в стенах, либо через специальные розетки, установленные на потолке. Также установлен кондиционер, который освежает воздух.

Можно утверждать, что система вентиляции соответствует нормам, следовательно мер по реконструкции вентиляции помещения применять не нужно.

5.2 Рабочее место программиста

Рассмотрим помещение, в котором ведется разработка:

- тип помещения: офис;
- размеры рабочего помещения: длина 4 м, ширина 3 м, высота 3 м;
- остекление помещения – двойное(одно окно размером 2000х2000 мм);
- искусственное освещение – светильники: 2 светильника, в каждом по 2 люминесцентные лампы (ПВЛМ-1х40);
- внутренняя отделка стен – светлая;
- помещение по зрительным условиям работы относится к V разряду, т. к. наименьший объект различения от 1 до 5 мм;
- вид работы – разработка интернет приложения для размещения бесплатных объявлений;
- количество рабочих мест – 2;
- категория работ – легкая.

План помещения представлен на рисунке 5.1.

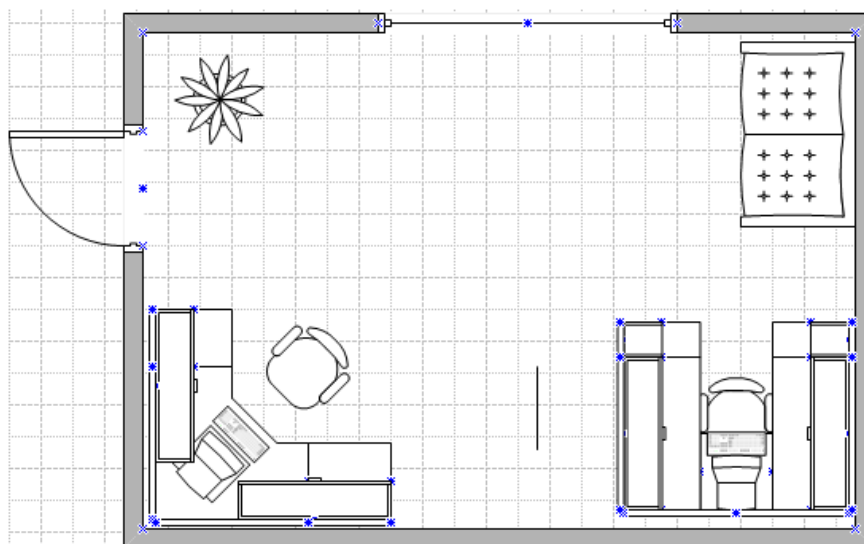


Рисунок 5.1 – План помещения

Характеристики используемого оборудования

Персональный компьютер. Технические характеристики устройства:

- габариты: 1600х700х1050мм (ноутбук + стол + кресло);
- электропитание: переменное напряжение 220-250 В, частотой 50 Гц, мощность 90 Вт;
- количество – 1 шт.

Модем:

- 4-х порт. коммутатором 10/100 Мбит/с;
- количество – 1 шт.;

Установлен 1 оконный кондиционер SamsungSMG4590, характеристики:

- режим охлаждения – 2400Вт;
- режим обогрева – 2400Вт;
- поток – 404 м3/ч;
- уровень шума 31Дб;
- габариты – 754x272x176 мм;
- вес – 39 кг;
- рассчитан на помещение площадью до 21 м².

5.3 Расчёт естественного освещения

Рассчитаем площадь боковых световых проёмов в помещении, необходимую для создания нормируемой освещённости на рабочем месте.

Помещение имеет размеры: длина $a = 4$ м, ширина $b = 4$ м, высота $h = 3$ м. Высота рабочей поверхности над уровнем пола – 0,7 м, окно начинается с высоты 0,8 м, высота окна 2 м.

Рабочее место расположено в 0,5 м от наружной стены помещения, где проектируем оконные проёмы. Минимальная освещённость будет в точке, отстоящей на расстояние 2,5 м от оконного проёма, в соответствии со СНиП РК 2.04-05-2002, указанными в таблице 5.1.

Общую требуемую площадь окон S_0 , м² определим по формуле

$$S_0 = \frac{S_n \cdot e_n \cdot \eta_0 \cdot k_{зд} \cdot k_3}{100 \cdot \tau_0 \cdot r_1} \quad (5.1)$$

где S_n – площадь помещения, м²

$$S_n = a \cdot b = 4 \cdot 3 = 12 \text{ м}^2$$

e_n – нормированное значение КЕО

$$e_n^{IV} = e_n \cdot m \cdot c \quad (5.2)$$

$m = 0,8$;

$c = 0,75$ – для IV часового пояса

Т а б л и ц а 5 . 1 – Значения коэффициентов m (СНиП РК 2.04-05-2002)

IV 50° северной широты и Южнее (Алматы и Караганда)	в наружных стенах зданий	в прямоугольных и трапециевидных фонарях	в фонарях типа шед.	При зенитных фонарях
	0,75	0,8	0,8	0,7

$e_n=1,2$ для работ средней точности V под разряда

$$e_n^{IV} = 1,2 \cdot 0,8 \cdot 0,75 = 0,72$$

k_3 – коэффициент запаса;

$k_3 = 1,2$ (учебные помещения, лаборатории, конструкторские бюро);

τ_0 – общий коэффициент светопропускания равный

$$\tau_0 = \tau_1 \cdot \tau_2 \cdot \tau_3 \cdot \tau_4 \quad (5.3)$$

В качестве светопропускающего материала используем:

– стекло оконное листовое, двойное: $\tau_1 = 0,8$;

– вид переплёта – двойной раздельный: $\tau_2 = 0,6$;

– вид несущей конструкции – железобетонные формы: $\tau_3 = 0,8$;

– солнцезащитные устройства – убирающиеся регулируемые жалюзи.

– $\tau_4 = 1$;

– общий коэффициент светопропускания:

$$\tau_0 = 0,8 \cdot 0,6 \cdot 0,8 \cdot 1 = 0,384$$

η_0 – световая характеристика окон.

Отношение длины комнаты к глубине наиболее удалённой точки от окна равно $\frac{4}{2,5} = 1,6$. Отношение ширины помещения к высоте от уровня рабочей

поверхности до верха окна равно $\frac{3}{2,1} = 1,43$. Отсюда $\eta_0 = 10,5$

r_1 – коэффициент, учитывающий повышение КЕО при боковом освещении благодаря свету, отражённому от поверхностей помещения и подстилающего слоя, прилегающего к зданию.

Отношение $\frac{a}{b} = \frac{4}{3} = 1,333$, средний коэффициент отражения в помещении

$\rho_{CP} = 0,5$, принимаем одностороннее боковое освещение

Тогда $r_1 = 1,3$.

$k_{зд}$ – коэффициент, учитывающий затенение окон противостоящими зданиями;

Поскольку затеняющих зданий поблизости нет, то $k_{зд} = 1$.

Вычислим общую площадь окон

$$S_0 = \frac{12 \cdot 0,72 \cdot 10,5 \cdot 1 \cdot 1,2}{100 \cdot 0,384 \cdot 1,3} = 2,18 \text{ м}^2$$

Так как в кабинете общая площадь окна составляет $2 \times 2 = 4 \text{ м}^2$, следовательно они соответствуют нормативам естественного освещения рабочего помещения. Реконструкция помещения не требуется.

5.4 Расчёт искусственного освещения методом коэффициента использования

В соответствии с требованиями СНиП РК 2.04-05-2002 «Естественное и искусственное освещение. Общие требования».

Разряд зрительной работы – V.

Нормируемая освещённость – 400 лк.

Используем систему общего освещения с люминесцентными лампами ЛБ с параметрами, указанными в таблице 5.2.

Т а б л и ц а 5 . 2 – Технические характеристики газоразрядных ламп ЛБ

Номинальная мощность, Вт	Номинальный световой поток ламп типа ЛБ, лм	Размеры ламп, мм	
		Диаметр	Длина по штырькам
40	3120	40	1213,6

Вычислим высоту подвеса светильника над рабочей поверхностью

$$H = h - h_p - h_c \quad (5.4)$$

где h_c – расстояние от светильника до перекрытия, $h_c = 0,05 \text{ м}$;

h_p – высота рабочей поверхности над полом, $h_p = 0,7 \text{ м}$;

h – высота помещения, $h = 3 \text{ м}$.

$$H = 3 - 0,7 - 0,05 = 2,25 \text{ м}$$

Наиболее выгодное расстояние между светильниками определяется как

$$L = \lambda \cdot H \quad (5.5)$$

где $\lambda = 1,2 \div 1,4$.

$$L = 1,2 \cdot 2,25 = 2,7 \text{ м};$$

Расстояние от стены до ближайшего светильника, когда работа у стены не проводится, определяем по формуле

$$l_1 = (0,4 \div 0,5) \cdot L \quad (5.6)$$

$$l_1 = 0,4 \cdot 2,7 = 1,08 \text{ м}.$$

Определяем индекс помещения

$$i = \frac{a \cdot b}{H \cdot (a + b)} \quad (5.7)$$

$$i = \frac{4 \cdot 3}{2,25 \cdot (4 + 3)} = 0,76$$

Коэффициенты отражения от потолка стен и пола соответственно равны

$$\rho_{\text{пот}} = 70\% ,$$

$$\rho_{\text{ст}} = 50\% ,$$

$$\rho_{\text{пол}} = 30\% .$$

Коэффициент использования в данном случае равен $\eta = 40\%$, коэффициент запаса равен $k_3 = 1,2$.

Определим количество люминесцентных ламп по формуле

$$N = \frac{E \cdot k_3 \cdot S_{\text{ос}} \cdot Z}{n \cdot \Phi_{\text{л}} \cdot \eta} \quad (5.8)$$

где $S_{\text{ос}}$ – площадь помещения;

k_3 – коэффициент запаса, $k_3 = 1,2$;

E – заданная минимальная освещённость, $E = 400$ лк.;

Z – коэффициент неравномерности освещения, $Z = 1,1$;

n – количество ламп в светильнике;

$\Phi_{\text{л}}$ – световой поток выбранной лампы, $\Phi_{\text{л}} = 3120$ лм;

η – коэффициент использования, $\eta = 40\%$.

$$N = \frac{400 \cdot 1,2 \cdot 12 \cdot 1,1}{2 \cdot 3120 \cdot 0,4} = 2,54 \approx 3$$

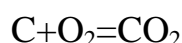
Всего для создания нормируемой освещенности 400 лк необходимо 3 светильника серии ЛБ с 2 лампами в светильнике, итого 6 люминесцентных ламп, мощность каждой лампы должна быть не меньше 40 Вт, что не соответствует действительности, а значит имеющегося в наличии освещения не достаточно для соответствия санитарным нормам. Так как в помещении установлено всего 2 светильника, а это нарушение санитарных норм, необходимо установить еще один светильник. В результате было реконструировано искусственное освещение, в итоге насчитывается 3

светильника серии ЛБ с 2 лампами в светильнике, что соответствует санитарным нормам.

5.5 Средства пожаротушения

Рассматриваемое помещение по взрывопожарной и пожарной опасности соответствует категории В (пожароопасное помещение) в соответствии со СНиП РК 3.02-04-2009 Административные и бытовые здания в разделе Противопожарные требования, так как в нем имеются горючие материалы (мебель, бумага, покрытие пола, шторы, пыль), которые горят при взаимодействии с кислородом воздуха.

Любой процесс горения сводится к взаимодействию



Количество горючих веществ составляет $G_{ГВ} = 130$ кг, а тепловой эффект реакции составляет $Q_{\text{тепл.углерода}} = 34,07$ МДж/кг.

Пожарная нагрузка определяется по формуле

$$Q=Q_{\text{тепл.углерода}} * G_{ГВ} \quad (5.9)$$

$$Q=130*34,07=4511 \text{ МДж}$$

Следовательно, пожарная нагрузка Q составляет 4511 МДж.

Удельная пожарная нагрузка определяется в соответствии санитарно эпидемиологическими требованиями

$$q=Q/S \quad (5.10)$$

где $S=12\text{м}^2$, площадь помещения

$$q=4511/12=375$$

Удельная пожарная нагрузка q составляет 375 МДж/м².

Проверяется условие

$$Q < 0,64 Q_{\text{н}}$$

Оно не выполняется, следовательно, помещение соответствует категории В2. В качестве средств пожаротушения применяется ручной углекислотный огнетушитель ОУ-2. Исправность огнетушителя периодически проверяется.

Вывод

В этом разделе был произведен анализ условий труда в данном офисном помещении, в частности, расчет естественного и искусственного освещения, небольшой анализ вентиляции.

Расчет показал, что для естественного освещения достаточно одного окна площадью 4 м².

Искусственное освещение рабочего места было недостаточным, что послужило поводом на реконструкцию освещения офиса. В итоге был добавлен еще один светильник. Таким образом, система искусственного освещения включает в себя 3 светильника по 2 лампы со световым потоком излучения 3120 лм каждая, поэтому в этом помещении можно работать и в темное время суток.

Пожарная безопасность не соответствовала норме, что было исправлено путем приобретения огнетушителя.

Вентиляция оборудована по всем стандартам и нормам, нет необходимости в реконструкции.

Заключение

Дистанционное обучение открывает новые возможности как для студентов так и для преподавателей. У студентов повышается эффективность самостоятельной работы, предоставляя абсолютно новые возможности для закрепления полученной информации. Повышается скорость усвоения и обработки информации. Преподаватели на лекциях и на семинарских часах могут уделять больше времени основным разделам курса. Ведь сейчас объем информации велик и увеличивается с каждым годом. Информация обновляется каждые два года по данным статистике РК.

В последнее десятилетие только развивается дистанционное образование в Казахстане. Но ведутся много исследовательских работ, а также частично в вузах вводится дистанционное информирование и дистанционные образовательные тесты. Все это зачатки большого будущего за дистанционным образованием. Считаю, что данный метод абсолютно совместим с нынешним поколением студентов. Так как большую информацию воспринимают с интернета, а дистанционное обучения осуществляется посредством интернета, в онлайн режиме. На данном сайте будет размещен открытый форум, который приблизит к естественной социальносетевой среде в которой привыкли находиться студенты.

Конечно же естественный контакт с преподавателем не заменит ни один сайт и ни одна программа. А данный образовательный сайт лишь помощь студентам и преподавателям.

Список литературы

- 1 Линн Бейли, Майкл Моррисон. Изучаем PHP и MySQL. – М.: Эксмо, 2010. – 786 с.
- 2 Веллинг Л., Томсон Л., Разработка web-приложений с помощью PHP и MySQL – М.:Вильямс, 2008. – 880 с.
- 3 Суэринг С., Конверс С., Парк Д. PHP и MySQL. Библия программиста. – СПб: Диалектика, 2010. – 912 с.
- 4 Котеров Д., Костарев А. PHP 5. Серия в подлиннике. – СПб: BHV. – 1120 с.
- 5 Сайт <http://php.com>
- 6 Томсон Л. Разработка Web-приложений на PHP и MySQL. /Пер. с англ./Томсон Л., Веллинг Л. -2-е изд. – СПб: ООО ДиаСофтЮП, 2003. Сайт <http://htmlbook.ru>
- 7 Сайт <http://javascript.com>
- 8 Харрис Э. PHP и MySQL для начинающих. – СПб: Кудиц-образ, 2005. – 384 с.
- 9 Суэринг С., Конверс С., Парк Д. PHP и MySQL. Библия программиста. – СПб: Диалектика, 2010. –912 с.
- 10 Сайт <http://mysql.com>
- 11 Законе Республики Казахстан от 27 июля 2007 года «Об образовании»
- 12 СНиП РК 2.04 – 05 – 2002 «Естественное и искусственное освещение. Общие требования».
- 13 СНиП РК 3.02 – 04 – 2009 Административные и бытовые здания раздел Противопожарная безопасность.
- 14 Безопасность жизнедеятельности. / Под ред. Арустамова Э.А. – Москва: Издательский дом «Дашков и К», 2002. - 676с.
- 15 З.Д. Еркешева, Г.Ш. Боканова. Методические указания к выполнению экономической части дипломных работ для студентов специальности 5В070400 – Вычислительная техника и программное обеспечение. – Алматы: АУЭС, 2013 – 40 с.
- 16 Экономика, организация и управление на предприятии Учебник/ Корсаков М.Н., Ребрин Ю.И., Федосова Т.В., Макареня Т.А., Шевченко И.К. и др.; Под ред. М.А.Боровской. – Таганрог: ТТИ ЮФУ, 2008. – 440с.
- 17 Аманбаев У.А. Экономика предприятия.-А.: «Бастау», 2012.
- 18 Буоров В.П. Бизнес-план фирмы – М.: «Инфра-М», 2011.
- 19 Куатова Д.Я. Экономика предприятия.-А.: «Экономика», 2011.

Приложение А

Листинг программы

```
<?php
/**
 * This page provides the Administration -> ... -> Theme selector
 UI.
 *
 * @package core
 * @license http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or
 later
 */

require_once(dirname(__FILE__) . '/../config.php');
require_once($CFG->libdir . '/adminlib.php');

$choose = optional_param('choose', '', PARAM_PLUGIN);
$reset = optional_param('reset', 0, PARAM_BOOL);
$device = optional_param('device', '', PARAM_TEXT);
$unsettheme = optional_param('unsettheme', 0, PARAM_BOOL);

admin_externalpage_setup('themeselector');

if (!empty($device)) {
    // Make sure the device requested is valid.
    $devices = core_useragent::get_device_type_list();
    if (!in_array($device, $devices)) {
        // The provided device isn't a valid device throw an
 error.
        print_error('invaliddevicetype');
    }
}
unset($SESSION->theme);
if ($reset and confirm_sesskey()) {
    theme_reset_all_caches();
} else if ($choose && $device && !$unsettheme &&
confirm_sesskey()) {
    // Load the theme to make sure it is valid.
    $theme = theme_config::load($choose);
    // Get the config argument for the chosen device.
    $themename =
core_useragent::get_device_type_cfg_var_name($device);
    set_config($themename, $theme->name);

    // Create a new page for the display of the themes readme.
    // This ensures that the readme page is shown using the new
 theme.
    $confirmpage = new univer_page();
    $confirmpage->set_context($PAGE->context);
    $confirmpage->set_url($PAGE->url);
}
```


Продолжение приложения А

```
$confirmpage->set_pagetype($PAGE->pagetype);
$confirmpage->set_title($PAGE->title);
$confirmpage->set_heading($PAGE->heading);

// Get the core renderer for the new theme.
$output = $confirmpage->get_renderer('core');

echo $output->header();
echo $output->heading(get_string('themesaved'));
echo $output->box_start();
echo format_text(get_string('choosereadme', 'theme_'. $theme-
>name), FORMAT_SKT);
echo $output->box_end();
echo $output->continue_button($CFG->wwwroot .
'/theme/index.php');
echo $output->footer();
exit;
} else if ($device && $unsettheme && confirm_sesskey() && ($device
!= 'default')) {
    // Unset the theme and continue.

unset_config(core_useragent::get_device_type_cfg_var_name($device)
);
    $device = '';
}

// Otherwise, show either a list of devices, or is
enableddevicedetection set to no or a
// device is specified show a list of themes.

$table = new html_table();
$table->data = array();
$heading = '';
if (!empty($CFG->enableddevicedetection) && empty($device)) {
    $heading = get_string('selectdevice', 'admin');
    // Display a list of devices that a user can select a theme
for.

    $strthemenoselected = get_string('themenoselected', 'admin');
    $strthemeselect = get_string('themeselect', 'admin');

    // Display the device selection screen.
    $table->id = 'admindeviceselector';
    $table->head = array(get_string('devicetype', 'admin'),
get_string('currenttheme', 'admin'), get_string('info'));

    $devices = core_useragent::get_device_type_list();
    foreach ($devices as $thedevice) {
```

Продолжение приложения А

```
$headingthemename = ''; // To output the picked theme name
when needed.
    $themename =
core_useragent::get_device_type_theme($thedevice);
    if (!$themename && $thedevice == 'default') {
        $themename = theme_config::DEFAULT_THEME;
    }

    $screenshotcell = $strthemename;
    $unsetthemebutton = '';
    if ($themename) {
        // Check the theme exists.
        $themename = clean_param($themename, PARAM_THEME);
        if (empty($themename)) {
            // Likely the theme has been deleted.

unset_config(core_useragent::get_device_type_cfg_var_name($thedevice));
        } else {
            $strthemename = get_string('pluginname',
'theme_'.$themename);
            // Link to the screenshot, now mandatory - the
image path is hardcoded because we need image from other themes,
            // not the current one.
            $screenshoturl = new
univer_url('/theme/image.php',
            array('theme' => $themename, 'image' =>
'screenshot', 'component' => 'theme'));
            // Contents of the screenshot/preview cell.
            $screenshotcell = html_writer::empty_tag('img',
array('src' => $screenshoturl, 'alt' => $strthemename));
            // Show the name of the picked theme.
            $headingthemename = $OUTPUT-
>heading($strthemename, 3);
        }
        // If not default device then show option to unset
theme.
        if ($thedevice != 'default') {
            $unsetthemestr = get_string('unsettheme',
'admin');
            $unsetthemeurl = new
univer_url('/theme/index.php',
            array('device' => $thedevice, 'sesskey' =>
sesskey(), 'unsettheme' => true));
            $unsetthemebutton = new
single_button($unsetthemeurl, $unsetthemestr, 'get');
            $unsetthemebutton = $OUTPUT-
>render($unsetthemebutton);
        }
    }
}
```

Продолжение приложения A

```
$deviceurl = new skt_url('/theme/index.php',
array('device' => $thedevice, 'sesskey' => sesskey()));
$select = new single_button($deviceurl, $strthemeselect,
'get');

$table->data[] = array(
    $OUTPUT->heading(ucfirst($thedevice), 3),
    $screenshotcell,
    $headingthemename . $OUTPUT->render($select) .
    $unsetthemebutton
);
}
} else {
    // Either a device has been selected or $CFG->
    >enabledevicedetection is off so display a list
    // of themes to select.
    $heading = get_string('selecttheme', 'admin', $device);
    if (empty($device)) {
        // If $CFG->enabledevicedetection is off this will return
        'default'.
        $device = core_useragent::get_device_type();
    }

    $table->id = 'adminthemeselector';
    $table->head = array(get_string('theme'), get_string('info'));

    $themes = core_component::get_plugin_list('theme');

    foreach ($themes as $themename => $themedir) {

        // Load the theme config.

        try {
            $theme = theme_config::load($themename);
        } catch (Exception $e) {
            // Bad theme, just skip it for now.
            continue;
        }
        if ($themename !== $theme->name) {
            // Obsoleted or broken theme, just skip for now.
            continue;
        }
        if (empty($CFG->themedesignermode) && $theme->
        >hidefromselector) {
            // The theme doesn't want to be shown in the theme
            selector and as theme
            // designer mode is switched off we will respect that
            decision.
            continue;
        }
    }
}
```

Продолжение приложения А

```
$strthemename = get_string('pluginname',
'theme_'.$themename);

// Build the table row, and also a list of items to go in
the second cell.
$row = array();
$infoitems = array();
$rowclasses = array();

// Set up bools whether this theme is chosen either main
or legacy.
$ischosentheme = ($themename ==
core_useragent::get_device_type_theme($device));

if ($ischosentheme) {
    // Is the chosen main theme.
    $rowclasses[] = 'selectedtheme';
}

// Link to the screenshot, now mandatory - the image path
is hardcoded because we need image from other themes,
// not the current one.
$screenshotpath = new univer_url('/theme/image.php',
    array('theme'=>$themename, 'image'=>'screenshot',
'component'=>'theme'));
// Contents of the first screenshot/preview cell.
$row[] = html_writer::empty_tag('img',
array('src'=>$screenshotpath, 'alt'=>$strthemename));
// Contents of the second cell.
$infocell = $OUTPUT->heading($strthemename, 3);
// Button to choose this as the main theme or unset this
theme for devices other than default.
if (($ischosentheme) && ($device != 'default')) {
    $unsetthemestr = get_string('unsettheme', 'admin');
    $unsetthemeurl = new skt_url('/theme/index.php',
        array('device' => $device, 'unsettheme' => true,
'sesskey' => sesskey()));
    $unsetbutton = new single_button($unsetthemeurl,
$unsetthemestr, 'get');
    $infocell .= $OUTPUT->render($unsetbutton);
} else if (!$ischosentheme) {
    $setthemestr = get_string('settheme');
    $setthemeurl = new skt_url('/theme/index.php',
        array('device' => $device, 'choose' => $themename,
'sesskey' => sesskey()));
    $setthemebutton = new single_button($setthemeurl,
$setthemestr, 'get');
    $infocell .= $OUTPUT->render($setthemebutton);
}
}
```

Продолжение приложения A

```
$row[] = $infocell;

$table->data[$themenam] = $row;
$table->rowclasses[$themenam] = join(' ', $rowclasses);
}
}
echo $OUTPUT->header('themeselector');
echo $OUTPUT->heading($heading);

$params = array('sesskey' => sesskey(), 'reset' => 1);
if (!empty($device)) {
    $params['device'] = $device;
}
echo $OUTPUT->single_button(new univer_url('index.php', $params),
get_string('themeresetcaches', 'admin'));

echo html_writer::table($table);

echo $OUTPUT->footer();
/* Custom CSS
-----*/
#page-header h1 {
    color:#000000;
    font-size:24px;
    font-family:Georgia;
    text-transform:uppercase;
    float:left;
    margin:-10px 0 5px;
}
a.logo {
    background: url([[setting:logo]]) no-repeat 0 0;
    display: block;
    float: left;
    height: 75px;
    margin: 0;
    padding: 0;
    width: 100%;
}

.dir-rtl a.logo {
    background: url([[setting:logo]]) no-repeat 100% 0;
    display: block;
    float: right;
}

/* Custom CSS Settings
-----*/
[[setting:customcss]]
<?php
// This file is part of SKT - http://
```

Продолжение приложения А

```
// comment out when debugging or better look into error log!
define('NO_DEBUG_DISPLAY', true);

// we need just the values from config.php and minlib.php
define('ABORT_AFTER_CONFIG', true);
require('../config.php'); // this stops immediately at the
beginning of lib/setup.php
require_once("$CFG->dirroot/lib/jslib.php");

if ($slashargument = min_get_slash_argument()) {
    $slashargument = ltrim($slashargument, '/');
    if (substr_count($slashargument, '/') < 2) {
        image_not_found();
    }
    // image must be last because it may contain "/"
    list($themename, $rev, $type) = explode('/', $slashargument,
3);
    $themename = min_clean_param($themename, 'SAFEDIR');
    $rev       = min_clean_param($rev, 'INT');
    $type      = min_clean_param($type, 'SAFEDIR');

} else {
    $themename = min_optional_param('theme', 'standard',
'SAFEDIR');
    $rev       = min_optional_param('rev', -1, 'INT');
    $type      = min_optional_param('type', 'head', 'RAW');
}

if ($type !== 'head' and $type !== 'footer') {
    header('HTTP/1.0 404 not found');
    die('Theme was not found, sorry.');
```

```
}

if (file_exists("$CFG->dirroot/theme/$themename/config.php")) {
    // exists
} else if (!empty($CFG->themedir) and file_exists("$CFG->
>themedir/$themename/config.php")) {
    // exists
} else {
    header('HTTP/1.0 404 not found');
    die('Theme was not found, sorry.');
```

```
}

$candidate = "$CFG->localcachedir/theme/$rev/$themename/javascript_{$type}.js";
$etag = sha1("$rev/$themename/$type");
```

```
if ($rev > 0 and file_exists($candidate)) {
    if (!empty($_SERVER['HTTP_IF_NONE_MATCH']) ||
!empty($_SERVER['HTTP_IF_MODIFIED_SINCE'])) {
```

Продолжение приложения А

```
        // we do not actually need to verify the etag value
because our files
        // never change in cache because we increment the rev
parameter
        js_send_unmodified(filemtime($candidate), $etag);
    }
    js_send_cached($candidate, $etag);
}

//=====
// ok, now we need to start normal script, we need to load all
libs and $DB
define('ABORT_AFTER_CONFIG_CANCEL', true);
define('NO_UPGRADE_CHECK', true); // Ignore upgrade check
require("$CFG->dirroot/lib/setup.php");

$theme = theme_config::load($themename);
$themerev = theme_get_revision();

if ($themerev <= 0 or $rev != $themerev) {
    // Do not send caching headers if they do not request current
revision,
    // we do not want to pollute browser caches with outdated JS.
    js_send_uncached($theme->javascript_content($type));
}

make_localcache_directory('theme', false);

js_write_cache_file_content($candidate,
core_minify::js_files($theme->javascript_files($type)));
// Verify nothing failed in cache file creation.
clearstatcache();
if (file_exists($candidate)) {
    js_send_cached($candidate, $etag);
}

js_send_uncached($theme->javascript_content($type));
<?php
// it under the terms of the GNU General Public License as
published by
// the Free Software Foundation, either version 3 of the License,
or
// (at your option) any later version.
// but WITHOUT ANY WARRANTY; without even the implied warranty of
// MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the
// GNU General Public License for more details.
//
// You should have received a copy of the GNU General Public
License
```

Продолжение приложения A

```
/**
 * jQuery serving script.
 *
 * Do not include jQuery scripts or CSS directly, always use
 * $PAGE->requires->jquery() or $PAGE->requires-
 >jquery_plugin('xx', 'yy').
 *
 * @package      core
 * @copyright    2013 Petr Skoda  {@link http://skodak.org}
 * @license      http://www.gnu.org/copyleft/gpl.html GNU GPL v3 or
 later
 */
// comment out when debugging or better look into error log!
define('NO_DEBUG_DISPLAY', true);
// We need just the values from config.php and minlib.php.
define('ABORT_AFTER_CONFIG', true);
require('../config.php'); // This stops immediately at the
beginning of lib/setup.php.

if ($slashargument = min_get_slash_argument()) {
    $path = ltrim($slashargument, '/');
} else {
    $path = min_optional_param('file', '', 'SAFEPATH');
    $path = ltrim($path, '/');
}

if (strpos($path, '/') === false) {
    jquery_file_not_found();
}

list($component, $path) = explode('/', $path, 2);

if (empty($path) or empty($component)) {
    jquery_file_not_found();
}

// Find the jQuery dir for this component.
if ($component === 'core') {
    $componentdir = "$CFG->dirroot/lib";
} else if (strpos($component, 'theme_')) {
    if (!empty($CFG->themedir)) {
        $componentdir = "$CFG->themedir/$component";
    } else {
        $componentdir = "$CFG->dirroot/theme/$component";
    }
} else {
    $componentdir =
core_component::get_component_directory($component);
}
```


Продолжение приложения А

```
if (!file_exists($componentdir) or
!file_exists("$componentdir/jquery/plugins.php")) {
    jquery_file_not_found();
}

$file = realpath("$componentdir/jquery/$path");

if (!$file or is_dir($file)) {
    jquery_file_not_found();
}

$etag = sha1("$component/$path");
$lifetime = 60*60*24*120; // 120 days should be enough.
$pathinfo = pathinfo($path);

if (empty($pathinfo['extension'])) {
    jquery_file_not_found();
}

$filename = $pathinfo['filename'].'.'.$pathinfo['extension'];

switch($pathinfo['extension']) {
    case 'gif' : $mimetype = 'image/gif';
                break;
    case 'png' : $mimetype = 'image/png';
                break;
    case 'jpg' : $mimetype = 'image/jpeg';
                break;
    case 'jpeg' : $mimetype = 'image/jpeg';
                break;
    case 'ico' : $mimetype = 'image/vnd.microsoft.icon';
                break;
    case 'svg' : $mimetype = 'image/svg+xml';
                break;
    case 'js' : $mimetype = 'application/javascript';
                break;
    case 'css' : $mimetype = 'text/css';
                break;
    case 'php' : jquery_file_not_found();
                break;
    default : $mimetype = 'document/unknown';
}

if (!empty($_SERVER['HTTP_IF_NONE_MATCH']) ||
!empty($_SERVER['HTTP_IF_MODIFIED_SINCE'])) {
    // We do not actually need to verify the etag value because
these files
    // never change, devs need to change file names on update!
    header('HTTP/1.1 304 Not Modified');
```

Продолжение приложения А

```
    header('Expires: '. gmdate('D, d M Y H:i:s', time() +
    $lifetime) .' GMT');
    header('Cache-Control: public, max-age='.$lifetime);
    header('Content-Type: '.$mimetype);
    header('Etag: "'.$etag.'"');
    die;
}

require_once("$CFG->dirroot/lib/xsendfilelib.php");

header('Etag: "'.$etag.'"');
header('Content-Disposition: inline; filename="'.$filename.'"');
header('Last-Modified: '. gmdate('D, d M Y H:i:s',
filemtime($file)) .' GMT');
header('Expires: '. gmdate('D, d M Y H:i:s', time() + $lifetime)
.' GMT');
header('Pragma: ');
header('Cache-Control: public, max-age='.$lifetime);
header('Accept-Ranges: none');
header('Content-Type: '.$mimetype);
if (xsendfile($file)) {
    die;
}
if ($mimetype === 'text/css' or $mimetype ===
'application/javascript') {
    if (!min_enable_zlib_compression()) {
        header('Content-Length: '.filesize($file));
    }
} else {
    // No need to compress images.
    header('Content-Length: '.filesize($file));
}
readfile($file);
die;
function jquery_file_not_found() {
    // Note: we can not disclose the exact file path here, sorry.
    header('HTTP/1.0 404 not found');
    die('File was not found, sorry.');
```

Продолжение приложения А

```
if (strpos($slashargument, '_s/') === 0) {
    // Can't use SVG.
    $slashargument = substr($slashargument, 3);
    $usesvg = false;
} else {
    $usesvg = true;
}

$chunk = null;
if (preg_match('#/(chunk(\d+)(/|$))#', $slashargument,
$matches)) {
    $chunk = (int)$matches[2];
    $slashargument = str_replace($matches[1], '',
$slashargument);
}

list($themename, $rev, $type) = explode('/', $slashargument,
3);
$themename = min_clean_param($themename, 'SAFEDIR');
$rev       = min_clean_param($rev, 'INT');
$type     = min_clean_param($type, 'SAFEDIR');

} else {
    $themename = min_optional_param('theme', 'standard',
'SAFEDIR');
    $rev       = min_optional_param('rev', 0, 'INT');
    $type     = min_optional_param('type', 'all', 'SAFEDIR');
    $chunk    = min_optional_param('chunk', null, 'INT');
    $usesvg   = (bool)min_optional_param('svg', '1', 'INT');
}

if ($type === 'editor') {
    // The editor CSS is never chunked.
    $chunk = null;
} else if ($type === 'all') {
    // We're fine.
} else {
    css_send_css_not_found();
}

if (file_exists("$CFG->dirroot/theme/$themename/config.php")) {
    // The theme exists in standard location - ok.
} else if (!empty($CFG->themedir) and file_exists("$CFG->
themedir/$themename/config.php")) {
    // Alternative theme location contains this theme - ok.
} else {
    header('HTTP/1.0 404 not found');
    die('Theme was not found, sorry.');
```

```
$candidatedir = "$CFG->localcachedir/theme/$rev/$themename/css";
```

Продолжение приложения A

```
$etag = "$rev/$themenamе/$type";
$candidatename = $type;
if (!$usesvg) {
    // Add to the sheet name, one day we'll be able to just drop
    this.
    $candidatedir .= '/nosvg';
    $etag .= '/nosvg';
}

if ($chunk != null) {
    $etag .= '/chunk'.$chunk;
    $candidatename .= '.'.$chunk;
}
$candidatesheet = "$candidatedir/$candidatename.css";
$etag = sha1($etag);

if (file_exists($candidatesheet)) {
    if (!empty($_SERVER['HTTP_IF_NONE_MATCH']) ||
!empty($_SERVER['HTTP_IF_MODIFIED_SINCE'])) {
        // We do not actually need to verify the etag value
        because our files
        // never change in cache because we increment the rev
        counter.
        css_send_unmodified(filemtime($candidatesheet), $etag);
    }
    css_send_cached_css($candidatesheet, $etag);
}

// Ok, now we need to start normal univеr script, we need to load
all libs and $DB.
define('ABORT_AFTER_CONFIG_CANCEL', true);
define('NO_UPGRADE_CHECK', true); // Ignore upgrade check.

require("$CFG->dirroot/lib/setup.php");

$theme = theme_config::load($themenamе);
$theme->force_svg_use($usesvg);

$themerev = theme_get_revision();

$cache = true;
if ($themerev <= 0 or $themerev != $rev) {
    $rev = $themerev;
    $cache = false;

    $candidatedir = "$CFG-
>localcachedir/theme/$rev/$themenamе/css";
    $etag = "$rev/$themenamе/$type";
    $candidatename = $type;
    if (!$usesvg) {
```

Продолжение приложения А

```
// Add to the sheet name, one day we'll be able to just
drop this.
$candidatedir .= '/nosvg';
$setag .= '/nosvg';
}

if ($chunk !== null) {
    $setag .= '/chunk' . $chunk;
    $candidatename .= '.' . $chunk;
}
$candidatesheet = "$candidatedir/$candidatename.css";
$setag = sha1($setag);
}

make_localcache_directory('theme', false);

if ($type === 'editor') {
    $csscontent = $theme->get_css_content_editor();
    css_store_css($theme, "$candidatedir/editor.css", $csscontent,
false);
} else {

    $lock = null;

    // Lock system to prevent concurrent requests to compile LESS,
    which is really slow and CPU intensive.
    // Each client should wait for one to finish the compilation
    before starting a new compiling process.
    // We only do this when the file will be cached...
    if ($type === 'less' && $cache) {
        $lockfactory =
\core\lock\lock_config::get_lock_factory('core_theme_get_css_conte
nt');
        // We wait for the lock to be acquired, the timeout does
        not need to be strict here.
        $lock = $lockfactory->get_lock($themename, rand(15, 30));
        if (file_exists($candidatesheet)) {
            // The file was built while we waited for the lock, we
            release the lock and serve the file.
            if ($lock) {
                $lock->release();
            }
            css_send_cached_css($candidatesheet, $setag);
        }
    }

    // Older IEs require smaller chunks.
    $csscontent = $theme->get_css_content();
}
```

Продолжение приложения A

```
$relroot = preg_replace('|^http.?://[^\|+|', '', $CFG->wwwroot);
if (!empty($slashargument)) {
    if ($usesvg) {
        $chunkurl =
"{ $relroot }/theme/styles.php/{ $themename }/{ $rev }/all";
    } else {
        $chunkurl =
"{ $relroot }/theme/styles.php/_s/{ $themename }/{ $rev }/all";
    }
} else {
    if ($usesvg) {
        $chunkurl =
"{ $relroot }/theme/styles.php?theme={ $themename }&rev={ $rev }&type=all";
    } else {
        $chunkurl =
"{ $relroot }/theme/styles.php?theme={ $themename }&rev={ $rev }&type=all&svg=0";
    }
}

css_store_css($theme, "$candidatedir/all.css", $csscontent,
true, $chunkurl);

// Release the lock.
if ($lock) {
    $lock->release();
}
}

if (!$cache) {
    // Do not pollute browser caches if invalid revision
    requested,
    // let's ignore legacy IE breakage here too.
    css_send_uncached_css($csscontent);
} else if ($chunk !== null and file_exists($candidatesheet)) {
    // Greetings stupid legacy IEs!
    css_send_cached_css($candidatesheet, $etag);
} else {
    // Real browsers - this is the expected result!
    css_send_cached_css_content($csscontent, $etag);
}
define('NO_DEBUG_DISPLAY', true);
define('NO_UPGRADE_CHECK', true);
define('NO_UNIVER_COOKIES', true);

require('../config.php');
```

Продолжение приложения А

```
require_once($CFG->dirroot.'/lib/csslib.php');

$themename = optional_param('theme', 'standard', PARAM_SAFEDIR);
$type       = optional_param('type', '', PARAM_SAFEDIR);
$subtype    = optional_param('subtype', '', PARAM_SAFEDIR);
$sheet      = optional_param('sheet', '', PARAM_SAFEDIR);
$usesvg     = optional_param('svg', 1, PARAM_BOOL);
$chunk      = optional_param('chunk', null, PARAM_INT);

if (file_exists("$CFG->dirroot/theme/$themename/config.php")) {
    // The theme exists in standard location - ok.
} else if (!empty($CFG->themedir) and file_exists("$CFG->
themedir/$themename/config.php")) {
    // Alternative theme location contains this theme - ok.
} else {
    css_send_css_not_found();
}

$theme = theme_config::load($themename);
$theme->force_svg_use($usesvg);

if ($type === 'editor') {
    $csscontent = $theme->get_css_content_editor();
    css_send_uncached_css($csscontent);
}

$chunkurl = new univer_url($CFG->httpswwwroot .
'/theme/styles_debug.php', array('theme' => $themename,
    'type' => $type, 'subtype' => $subtype, 'sheet' => $sheet,
    'usesvg' => $usesvg));
// We need some kind of caching here because otherwise the page
navigation becomes
// way too slow in theme designer mode. Feel free to create full
cache definition later...
$key = "$type $subtype $sheet $usesvg";
$cache = cache::make_from_params(cache_store::MODE_APPLICATION,
'core', 'themedesigner', array('theme' => $themename));
if ($content = $cache->get($key)) {
    if ($content['created'] > time() -
THEME_DESIGNER_CACHE_LIFETIME) {
        $csscontent = $content['data'];

        // We need to chunk the content.
        if ($chunk !== null) {
            $chunks = css_chunk_by_selector_count($csscontent,
$chunkurl->out(false));
            $csscontent = ($chunk === 0) ? end($chunks) :
$chunks[$chunk - 1];
        }
    }
}
```

Продолжение приложения A

```
        css_send_uncached_css($csscontent);
    }
}

$csscontent = $theme->get_css_content_debug($type, $subtype,
$sheet);
$cache->set($key, array('data' => $csscontent, 'created' =>
time()));

// We need to chunk the content.
if ($chunk !== null) {
    // The chunks are ordered so that the last chunk is the one
    containing the @import, and so
    // the first one to be included. All the other chunks are set
    in the array before that one.
    // See {@link css_chunk_by_selector_count()} for more details.
    $chunks = css_chunk_by_selector_count($csscontent, $chunkurl-
>out(false));
    $csscontent = ($chunk === 0) ? end($chunks) : $chunks[$chunk -
1];
}

css_send_uncached_css($csscontent);
require('../config.php');

$url          = required_param('url', PARAM_LOCALURL);
$newdevice    = required_param('device', PARAM_TEXT);

require_sesskey();

core_useragent::set_user_device_type($newdevice);
redirect($url);
define('NO_DEBUG_DISPLAY', true);

// we need just the values from config.php and minlib.php
define('ABORT_AFTER_CONFIG', true);
require('../config.php'); // this stops immediately at the
beginning of lib/setup.php

// get special url parameters

list($parts, $slasharguments) = combo_params();
if (!$parts) {
    combo_not_found();
}

$etag = sha1($parts);
$parts = trim($parts, '&');

// find out what we are serving - only one type per request
```


Продолжение приложения А

```
$content = '';
if (substr($parts, -3) === '.js') {
    $mimetype = 'application/javascript';
} else if (substr($parts, -4) === '.css') {
    $mimetype = 'text/css';
} else {
    combo_not_found();
}

// if they are requesting a revision that's not -1, and they have
// supplied an
// If-Modified-Since header, we can send back a 304 Not Modified
// since the
// content never changes (the rev number is increased any time the
// content changes)
if (strpos($parts, '/-1/') === false and
(!empty($_SERVER['HTTP_IF_NONE_MATCH']) ||
!empty($_SERVER['HTTP_IF_MODIFIED_SINCE']))) {
    $lifetime = 60*60*24*360; // 1 year, we do not change YUI
versions often, there are a few custom yui modules
    header('HTTP/1.1 304 Not Modified');
    header('Expires: ' . gmdate('D, d M Y H:i:s', time() +
$lifetime) . ' GMT');
    header('Cache-Control: public, max-age='.$lifetime);
    header('Content-Type: '.$mimetype);
    header('Etag: "'.$etag.'"');
    die;
}

$parts = explode('&', $parts);
$cache = true;
$lastmodified = 0;

while (count($parts)) {
    $part = array_shift($parts);
    if (empty($part)) {
        continue;
    }
    $filecontent = '';
    $part = min_clean_param($part, 'SAFEPATH');
    $bits = explode('/', $part);
    if (count($bits) < 2) {
        $content .= "\n// Wrong combo resource $part!\n";
        continue;
    }

    $version = array_shift($bits);
    if ($version === 'rollup') {
        $yuipatchedversion = explode('_', array_shift($bits));
        $revision = $yuipatchedversion[0];
    }
}
```

Продолжение приложения A

```
$rollupname = array_shift($bits);

if (strpos($rollupname, 'yui-universimple') !== false) {
    if (substr($rollupname, -3) === '.js') {
        // Determine which version of this rollup should
be used.
        $filesuffix = '.js';
        preg_match('/(- (debug|min))?\.\.js/', $rollupname,
$matches);
        if (isset($matches[1])) {
            $filesuffix = $matches[0];
        }

        $type = 'js';
    } else if (substr($rollupname, -4) === '.css') {
        $type = 'css';
    } else {
        continue;
    }

    // Allow support for revisions on YUI between official
releases.
    // We can just discard the subrevision since it is
only used to invalidate the browser cache.
    $yuipatchedversion = explode('_', $revision);
    $yuiversion = $yuipatchedversion[0];

    $yuimodules = array(
        'yui',
        'oop',
        'event-custom-base',
        'dom-core',
        'dom-base',
        'color-base',
        'dom-style',
        'selector-native',
        'selector',
        'node-core',
        'node-base',
        'event-base',
        'event-base-ie',
        'pluginhost-base',
        'pluginhost-config',
        'event-delegate',
        'node-event-delegate',
        'node-pluginhost',
        'dom-screen',
        'node-screen',
        'node-style',
```

Продолжение приложения А

```
'querystring-stringify-simple',
'io-base',
'json-parse',
'transition',
'selector-css2',
'selector-css3',
'dom-style-ie',

// Some extras we use everywhere.
'escape',

'attribute-core',
'event-custom-complex',
'base-core',
'attribute-base',
'attribute-extras',
'attribute-observable',
'base-observable',
'base-base',
'base-pluginhost',
'base-build',
'event-synthetic',

'attribute-complex',
'event-mouseenter',
'event-key',
'event-outside',
'event-autohide',
'event-focus',
'classnamemanager',
'widget-base',
'widget-htmlparser',
'widget-skin',
'widget-uitableviews',
'widget-stdmod',
'widget-position',
'widget-position-align',
'widget-stack',
'widget-position-constrain',
'overlay',

'widget-autohide',
'button-core',
'button-plugin',
'widget-buttons',
'widget-modality',
'panel',
'yui-throttle',
'dd-ddm-base',
```

Продолжение приложения А

```
'dd-drag',
'dd-plugin',

// Cache is used by univer-core-tooltip which we
include everywhere.
'cache-base',
);

// We need to add these new parts to the beginning of
the $parts list, not the end.
if ($type === 'js') {
    $newparts = array();
    foreach ($yuimodules as $module) {
        $newparts[] = $yuiversion . '/' . $module .
        '/' . $module . $filesuffix;
    }
    $newparts[] = 'yuiuseall/yuiuseall';
    $parts = array_merge($newparts, $parts);
} else {
    $newparts = array();
    foreach ($yuimodules as $module) {
        $candidate = $yuiversion . '/' . $module .
        '/assets/skins/sam/' . $module . '.css';
        if (!file_exists("$CFG-
>libdir/yuilib/$candidate")) {
            continue;
        }
        $newparts[] = $candidate;
    }
    if ($newparts) {
        $parts = array_merge($newparts, $parts);
    }
}

// Handle the mcore rollup.
if (strpos($rollupname, 'mcore') !== false) {
    $yuimodules = array(
        'core/tooltip/tooltip',
        'core/popuphelp/popuphelp',
        'core/widget-focusafterclose/widget-
focusafterclose',
        'core/dock/dock-loader',
        'core/notification/notification-dialogue',
    );
}

// Determine which version of this rollup should be
used.
$filesuffix = '.js';
```

Продолжение приложения A

```
preg_match('/(-(debug|min))?.js/', $rollupname,
$matches);
    if (isset($matches[1])) {
        $filesuffix = $matches[0];
    }

    // We need to add these new parts to the beginning of
the $parts list, not the end.
    $newparts = array();
    foreach ($yuimodules as $module) {
        $newparts[] = 'm/' . $revision . '/' . $module .
$filesuffix;
    }
    $parts = array_merge($newparts, $parts);
}
continue;
}
if ($version === 'm') {
    $version = 'skt';
}
if ($version === 'skt') {
    if (count($bits) <= 3) {
        // This is an invalid skt load attempt.
        $content .= "\n// Incorrect skt module inclusion. Not
enough component information in {$part}.\n";
        continue;
    }
    $revision = (int)array_shift($bits);
    if ($revision === -1) {
        // Revision -1 says please don't cache the JS
        $cache = false;
    }
    $frankenstyle = array_shift($bits);
    $filename = array_pop($bits);
    $modulename = $bits[0];
    $dir =
core_component::get_component_directory($frankenstyle);

    // For shifted YUI modules, we need the YUI module name in
frankenstyle format.
    $frankenstylemodulename = join('-', array($version,
$frankenstyle, $modulename));
    $frankenstylefilename = preg_replace('/' . $modulename .
'/', $frankenstylemodulename, $filename);

    // Submodules are stored in a directory with the full
submodule name.
    // We need to remove the -debug.js, -min.js, and .js from
the file name to calculate that directory name.
```

Продолжение приложения A

```
$frankenstyledirectoryname = str_replace(array('-min.js',
'-debug.js', '.js', '.css'), '', $frankenstylefilename);

// By default, try and use the /yui/build directory.
$contentfile = $dir . '/yui/build/' .
$frankenstyledirectoryname;
if ($mimetype == 'text/css') {
    // CSS assets are in a slightly different place to the
JS.
    $contentfile = $contentfile . '/assets/skins/sam/' .
$frankenstylefilename;

    // Add the path to the bits to handle fallback for
non-shifted assets.
    $bits[] = 'assets';
    $bits[] = 'skins';
    $bits[] = 'sam';
} else {
    $contentfile = $contentfile . '/' .
$frankenstylefilename;
}

// If the shifted versions don't exist, fall back to the
non-shifted file.
if (!file_exists($contentfile) or !is_file($contentfile))
{
    // We have to revert to the non-minified and non-debug
versions.
    $filename = preg_replace('/-(min|debug)\./', '.',
$filename);
    $contentfile = $dir . '/yui/' . join('/', $bits) . '/'
. $filename;
}
} else if ($version === '2in3') {
    $contentfile = "$CFG->libdir/yuilib/$part";
} else if ($version == 'gallery') {
    if (count($bits) <= 2) {
        // This is an invalid module load attempt.
        $content .= "\n// Incorrect univer module inclusion.
Not enough component information in {$part}.\n";
        continue;
    }
    $revision = (int)array_shift($bits);
    if ($revision === -1) {
        // Revision -1 says please don't cache the JS
        $cache = false;
    }
    $contentfile = "$CFG->libdir/yuilib/gallery/" . join('/',
$bits);
```

Продолжение приложения A

```
} else if ($version == 'yuiuseall') {
    // Create global Y that is available in global scope,
    // this is the trick behind original SimpleYUI.
    $filecontent = "var Y = YUI().use('*');"

} else {
    // Allow support for revisions on YUI between official
releases.
    // We can just discard the subrevision since it is only
used to invalidate the browser cache.
    $yuipatchedversion = explode('_', $version);
    $yuiversion = $yuipatchedversion[0];
    if ($yuiversion != $CFG->yui3version) {
        $content .= "\n// Wrong yui version $part!\n";
        continue;
    }
    $newpart = explode('/', $part);
    $newpart[0] = $yuiversion;
    $part = implode('/', $newpart);
    $contentfile = "$CFG->libdir/yuilib/$part";
}
if (!file_exists($contentfile) or !is_file($contentfile)) {
    $location = '$CFG->dirroot'.preg_replace('/^'.preg_quote($CFG->dirroot, '/').'/',
'', $contentfile);
    $content .= "\n// Combo resource $part ($location) not
found!\n";
    continue;
}

if (empty($filecontent)) {
    $filecontent = file_get_contents($contentfile);
}
$fmodified = filemtime($contentfile);
if ($fmodified > $lastmodified) {
    $lastmodified = $fmodified;
}

$relroot = preg_replace('|^http.?://[^\/]+|', '', $CFG->wwwroot);
$sep = ($slasharguments ? '/' : '?file=');

if ($mimetype === 'text/css') {
    if ($version == 'univer') {
        // Search for all images in the file and replace with
an appropriate link to the yui_image.php script
        $imagebits = array(
            $sep . $version,
            $frankenstyle,
```

Продолжение приложения A

```
    $modulename,  
    array_shift($bits),  
    '$1.$2'  
);  
  
    $filecontent = preg_replace('/([a-z0-9_-  
]+)\.(png|gif)/', $relroot . '/theme/yui_image.php' . implode('/',  
$imagebits), $filecontent);  
    } else if ($version == '2in3') {  
        // First we need to remove relative paths to images.  
These are used by YUI modules to make use of global assets.  
        // I've added this as a separate regex so it can be  
easily removed once  
        // YUI standardise there CSS methods  
        $filecontent =  
preg_replace('#(\.\.\./\.\.\./\.\.\./\.\.\./assets/skins/sam/)?([a-z0-9_-  
]+)\.(png|gif)#', '$2.$3', $filecontent);  
  
        // search for all images in yui2 CSS and serve them  
through the yui_image.php script  
        $filecontent = preg_replace('/([a-z0-9_-  
]+)\.(png|gif)/', $relroot.'/theme/yui_image.php'.$sep.$CFG-  
>yui2version.'/$1.$2', $filecontent);  
  
    } else if ($version == 'gallery') {  
        // Replace any references to the CDN with a relative  
link.  
        $filecontent = preg_replace('#(' .  
preg_quote('http://yui.yahooapis.com/') . '(gallery-[^\/*/]*)#',  
'../..../..../', $filecontent);  
  
        // Replace all relative image links with the a link to  
yui_image.php.  
        $filecontent = preg_replace('#(' .  
preg_quote('../..../..../') . ') (gallery-[^\/*/]*/assets/skins/sam/[a-  
z0-9_-]+)\.(png|gif)#',  
  
            $relroot . '/theme/yui_image.php' . $sep .  
'/gallery/' . $revision . '/$2.$3', $filecontent);  
  
    } else {  
        // First we need to remove relative paths to images.  
These are used by YUI modules to make use of global assets.  
        // I've added this as a separate regex so it can be  
easily removed once  
        // YUI standardise there CSS methods  
        $filecontent =  
preg_replace('#(\.\.\./\.\.\./\.\.\./\.\.\./assets/skins/sam/)?([a-z0-9_-  
]+)\.(png|gif)#', '$2.$3', $filecontent);
```


Продолжение приложения A

```
        // search for all images in yui2 CSS and serve them
        through the yui_image.php script
        $filecontent = preg_replace('/([a-z0-9_-
]+)\.(png|gif)/',
$relroot.'/theme/yui_image.php'.$sep.$version.'/'.$1.$2',
$filecontent);
    }
}

$content .= $filecontent;
}

if ($lastmodified == 0) {
    $lastmodified = time();
}

if ($cache) {
    combo_send_cached($content, $mimetype, $etag, $lastmodified);
} else {
    combo_send_uncached($content, $mimetype);
}

/**
 * Send the JavaScript cached
 * @param string $content
 * @param string $mimetype
 * @param string $etag
 * @param int $lastmodified
 */
function combo_send_cached($content, $mimetype, $etag,
$lastmodified) {
    $lifetime = 60*60*24*360; // 1 year, we do not change YUI
versions often, there are a few custom yui modules

    header('Content-Disposition: inline; filename="combo"');
    header('Last-Modified: '. gmdate('D, d M Y H:i:s',
$lastmodified) .' GMT');
    header('Expires: '. gmdate('D, d M Y H:i:s', time() +
$lifetime) .' GMT');
    header('Pragma: ');
    header('Cache-Control: public, max-age='.$lifetime);
    header('Accept-Ranges: none');
    header('Content-Type: '.$mimetype);
    header('Etag: "'.$etag.'");
    if (!min_enable_zlib_compression()) {
        header('Content-Length: '.strlen($content));
    }

    echo $content;
}
```

Продолжение приложения A

```
    die;
}

/**
 * Send the JavaScript uncached
 * @param string $content
 * @param string $mimetype
 */
function combo_send_uncached($content, $mimetype) {
    header('Content-Disposition: inline; filename="combo"');
    header('Last-Modified: '. gmdate('D, d M Y H:i:s', time()) .'
GMT');
    header('Expires: '. gmdate('D, d M Y H:i:s', time() + 2) .'
GMT');
    header('Pragma: ');
    header('Accept-Ranges: none');
    header('Content-Type: '.$mimetype);
    if (!min_enable_zlib_compression()) {
        header('Content-Length: '.strlen($content));
    }

    echo $content;
    die;
}

function combo_not_found($message = '') {
    header('HTTP/1.0 404 not found');
    if ($message) {
        echo $message;
    } else {
        echo 'Combo resource not found, sorry.';
    }
    die;
}

function combo_params() {
    if (isset($_SERVER['QUERY_STRING']) and
strpos($_SERVER['QUERY_STRING'], 'file=/') === 0) {
        // url rewriting
        $slashargument = substr($_SERVER['QUERY_STRING'], 6);
        return array($slashargument, true);

    } else if (isset($_SERVER['REQUEST_URI']) and
strpos($_SERVER['REQUEST_URI'], '?') !== false) {
        $parts = explode('?', $_SERVER['REQUEST_URI'], 2);
        return array($parts[1], false);

    } else if (isset($_SERVER['QUERY_STRING']) and
strpos($_SERVER['QUERY_STRING'], '?') !== false) {
```

Продолжение приложения A

```
// note: buggy or misconfigured IIS does return the query
string in REQUEST_URI
return array($_SERVER['QUERY_STRING'], false);

} else if ($slashargument = min_get_slash_argument()) {
    $slashargument = ltrim($slashargument, '/');
    return array($slashargument, true);

} else {
    // unsupported server, sorry!
    combo_not_found('Unsupported server - query string can not
be determined, try disabling YUI combo loading in admin
settings.');
```

```
}
```

```
}
```

```
define('NO_DEBUG_DISPLAY', true);
```

```
// we need just the values from config.php and minlib.php
```

```
define('ABORT_AFTER_CONFIG', true);
```

```
require('../config.php'); // this stops immediately at the
beginning of lib/setup.php
```

```
if ($slashargument = min_get_slash_argument()) {
```

```
    $path = ltrim($slashargument, '/');
```

```
} else {
```

```
    $path = min_optional_param('file', '', 'SAFEPATH');
```

```
}
```

```
$etag = sha1($path);
```

```
$parts = explode('/', $path);
```

```
$version = array_shift($parts);
```

```
if ($version === 'm') {
```

```
    $version = 'univer';
```

```
}
```

```
if ($version == 'univer' && count($parts) >= 3) {
```

```
    $frankenstyle = array_shift($parts);
```

```
    $module = array_shift($parts);
```

```
    $image = array_pop($parts);
```

```
    $subdir = join('/', $parts);
```

```
    $dir = core_component::get_component_directory($frankenstyle);
```

```
// For shifted YUI modules, we need the YUI module name in
frankenstyle format.
```

```
$frankenstylemodulename = join('-', array($version,
$frankenstyle, $module));
```

```
// By default, try and use the /yui/build directory.
```

```
$imagepath = $dir . '/yui/build/' . $frankenstylemodulename .
'/assets/skins/sam/' . $image;
```

Продолжение приложения A

```
// If the shifted versions don't exist, fall back to the non-
shifted file.
    if (!file_exists($imagepath) or !is_file($imagepath)) {
        $imagepath = $dir . '/yui/' . $module .
'/assets/skins/sam/' . $image;
    }
} else if ($version == 'gallery' && count($parts) >= 3) {
    list($revision, $module, , , , $image) = $parts;
    $imagepath = "$CFG-
>dirroot/lib/yuilib/gallery/$module/assets/skins/sam/$image";
} else {
    // Allow support for revisions on YUI between official
releases.
    // We can just discard the subrevision since it is only used
to invalidate the browser cache.
    $yuipatchedversion = explode('_', $version);
    $yuiversion = $yuipatchedversion[0];
    if (count($parts) == 1 && ($yuiversion == $CFG->yui3version ||
$yuiversion == $CFG->yui2version)) {
        list($image) = $parts;
        if ($yuiversion == $CFG->yui3version) {
            $imagepath = "$CFG->dirroot/lib/yuilib/$CFG-
>yui3version/assets/skins/sam/$image";
        } else {
            $imagepath = "$CFG->dirroot/lib/yuilib/2in3/$CFG-
>yui2version/build/assets/skins/sam/$image";
        }
    } else {
        yui_image_not_found();
    }
}

if (!file_exists($imagepath)) {
    yui_image_not_found();
}

$pathinfo = pathinfo($imagepath);
$imagename = $pathinfo['filename'].'.'.$pathinfo['extension'];
switch($pathinfo['extension']) {
    case 'gif' : $mimetype = 'image/gif'; break;
    case 'png' : $mimetype = 'image/png'; break;
    case 'jpg' : $mimetype = 'image/jpeg'; break;
    case 'jpeg' : $mimetype = 'image/jpeg'; break;
    case 'ico' : $mimetype = 'image/vnd.microsoft.icon'; break;
    default: $mimetype = 'document/unknown';
}

// if they are requesting a revision that's not -1, and they have
supplied an
```

Продолжение приложения А

```
// If-Modified-Since header, we can send back a 304 Not Modified
since the
// content never changes (the rev number is increased any time the
content changes)
if (strpos($path, '/-1/') === false and
(!empty($_SERVER['HTTP_IF_NONE_MATCH']) ||
!empty($_SERVER['HTTP_IF_MODIFIED_SINCE']))) {
    $lifetime = 60*60*24*360; // 1 year, we do not change YUI
versions often, there are a few custom yui modules
    header('HTTP/1.1 304 Not Modified');
    header('Last-Modified: '. gmdate('D, d M Y H:i:s',
filemtime($imagepath)) .' GMT');
    header('Expires: '. gmdate('D, d M Y H:i:s', time() +
$lifetime) .' GMT');
    header('Cache-Control: public, max-age='.$lifetime.', no-
transform');
    header('Content-Type: '.$mimetype);
    header('Etag: "'.$etag.'");
    die;
}
```

```
yui_image_cached($imagepath, $imagename, $mimetype, $etag);
```

```
function yui_image_cached($imagepath, $imagename, $mimetype,
$etag) {
    global $CFG;
    require("$CFG->dirroot/lib/xsendfilelib.php");

    $lifetime = 60*60*24*360; // 1 year, we do not change YUI
versions often, there are a few custom yui modules

    header('Content-Disposition: inline;
filename="'.$imagename.'");
    header('Last-Modified: '. gmdate('D, d M Y H:i:s',
filemtime($imagepath)) .' GMT');
    header('Expires: '. gmdate('D, d M Y H:i:s', time() +
$lifetime) .' GMT');
    header('Pragma: ');
    header('Cache-Control: public, max-age=315360000, no-
transform');
    header('Accept-Ranges: none');
    header('Content-Type: '.$mimetype);
    header('Content-Length: '.filesize($imagepath));
    header('Etag: "'.$etag.'");

    if (xsendfile($imagepath)) {
        die;
    }
}
```

Продолжение приложения A

```
// no need to gzip already compressed images ;-)  
  
readfile($imagepath);  
die;  
}  
  
function yui_image_not_found() {  
    header('HTTP/1.0 404 not found');  
    die('Image was not found, sorry.');}  
define('NO_DEBUG_DISPLAY', true);  
  
define('ABORT_AFTER_CONFIG', true);  
require('../config.php');  
if ($slashargument = min_get_slash_argument()) {  
    $slashargument = ltrim($slashargument, '/');  
    if (substr_count($slashargument, '/') < 3) {  
        font_not_found();  
    }  
    list($themename, $component, $rev, $font) = explode('/',  
$slashargument, 4);  
    $themename = min_clean_param($themename, 'SAFEDIR');  
    $component = min_clean_param($component, 'SAFEDIR');  
    $rev        = min_clean_param($rev, 'INT');  
    $font       = min_clean_param($font, 'RAW');  
} else {  
    $themename = min_optional_param('theme', 'standard',  
'SAFEDIR');  
    $component = min_optional_param('component', 'core',  
'SAFEDIR');  
    $rev       = min_optional_param('rev', -1, 'INT');  
    $font      = min_optional_param('font', '', 'RAW');}  
  
if (!$font) {  
    font_not_found();  
}  
  
if (empty($component) or $component === 'univer' or $component ===  
'core') {  
    $component = 'core';  
}  
  
if (preg_match('/^[a-z0-9_-]+\.woff$/i', $font, $matches)) {  
    // This is the real standard!  
    $font = $matches[0];  
    $mimetype = 'application/font-woff';  
}
```

Продолжение приложения A

```
} else if (preg_match('/^[a-z0-9_-]+\.ttf$/i', $font, $matches)) {
    $font = $matches[0];
    $mimetype = 'application/x-font-ttf';

} else if (preg_match('/^[a-z0-9_-]+\.otf$/i', $font, $matches)) {
    $font = $matches[0];
    $mimetype = 'application/x-font-opentype';

} else if (preg_match('/^[a-z0-9_-]+\.eot$/i', $font, $matches)) {
    // IE8 must die!!!
    $font = $matches[0];
    $mimetype = 'application/vnd.ms-fontobject';
} else if (preg_match('/^[a-z0-9_-]+\.svg$/i', $font, $matches)) {
    $font = $matches[0];
    $mimetype = 'image/svg+xml';

} else {
    font_not_found();
}

if (file_exists("$CFG->dirroot/theme/$themenamename/config.php")) {
    // Normal theme exists.
} else if (!empty($CFG->themedir) and file_exists("$CFG->
>themedir/$themenamename/config.php")) {
    // Theme exists in alternative location.
} else {
    font_not_found();
}

$candidatelocation = "$CFG->
>localcachedir/theme/$rev/$themenamename/fonts/$component";
$etag = sha1("$rev/$themenamename/$component/$font");

if ($rev > 0) {
    if (file_exists("$candidatelocation/$font.error")) {
        font_not_found();
    }

    if (file_exists("$candidatelocation/$font")) {
        if (!empty($_SERVER['HTTP_IF_NONE_MATCH']) ||
!empty($_SERVER['HTTP_IF_MODIFIED_SINCE'])) {
            // We do not actually need to verify the etag value
because our files
            // never change in cache because we increment the rev
parameter.
            $lifetime = 60*60*24*60; // 60 days only - the
revision may get incremented quite often.
            header('HTTP/1.1 304 Not Modified');
            header('Expires: '. gmdate('D, d M Y H:i:s', time() +
$lifetime) .' GMT');
```

Продолжение приложения А

```
header('Cache-Control: public, max-age='.$lifetime);
    header('Content-Type: '.$mimetype);
    header('Etag: "'.$etag.'");
    die;
}
    send_cached_font("$candidatelocation/$font", $etag, $font,
$mimetype);
}
}

// Ok, now we need to start normal univer script, we need to load
all libs and $DB.
define('ABORT_AFTER_CONFIG_CANCEL', true);

define('NO_UNIVER_COOKIES', true); // Session not used here.
define('NO_UPGRADE_CHECK', true); // Ignore upgrade check.

require("$CFG->dirroot/lib/setup.php");

$theme = theme_config::load($themename);
$themerev = theme_get_revision();

$fontfile = $theme->resolve_font_location($font, $component);

if ($themerev <= 0 or $rev != $themerev) {
    // Do not send caching headers if they do not request current
revision,
    // we do not want to pollute browser caches with outdated
fonts.
    if (empty($fontfile) or !is_readable($fontfile)) {
        font_not_found();
    }
    send_uncached_font($fontfile, $font, $mimetype);
}

make_localcache_directory('theme', false);

if (empty($fontfile) or !is_readable($fontfile)) {
    if (!file_exists($candidatelocation)) {
        @mkdir($candidatelocation, $CFG->directorypermissions,
true);
    }
    // Make note we can not find this file.
    $cachefont = "$candidatelocation/$font.error";
    $fp = fopen($cachefont, 'w');
    fclose($fp);
    font_not_found();
}
```


Продолжение приложения A

```
$cachefont = cache_font($font, $fontfile, $candidatelocation);
if (connection_aborted()) {
    die;
}
// Make sure nothing failed.
clearstatcache();
if (file_exists($cachefont)) {
    send_cached_font($cachefont, $etag, $font, $mimetype);
}

send_uncached_font($fontfile, $font, $mimetype);

// Utility functions.

function send_cached_font($fontpath, $etag, $font, $mimetype) {
    global $CFG;
    require("$CFG->dirroot/lib/xsendfilelib.php");

    $lifetime = 60*60*24*60; // 60 days only - the revision may
    get incremented quite often.

    header('Etag: "'. $etag. '"');
    header('Content-Disposition: inline; filename="'. $font. '"');
    header('Last-Modified: '. gmdate('D, d M Y H:i:s',
filetime($fontpath)) . ' GMT');
    header('Expires: '. gmdate('D, d M Y H:i:s', time() +
$lifetime) . ' GMT');
    header('Pragma: ');
    header('Cache-Control: public, max-age='. $lifetime);
    header('Accept-Ranges: none');
    header('Content-Type: '. $mimetype);
    header('Content-Length: '. filesize($fontpath));

    if (xsendfile($fontpath)) {
        die;
    }

    // No need to gzip already compressed fonts.
    die;
}

function send_uncached_font($fontpath, $font, $mimetype) {
    header('Content-Disposition: inline; filename="'. $font. '"');
    header('Last-Modified: '. gmdate('D, d M Y H:i:s', time()) . '
GMT');
    header('Expires: '. gmdate('D, d M Y H:i:s', time() + 15) . '
GMT');
    header('Pragma: ');
}
```

Продолжение приложения А

```
readfile($fontpath);

header('Accept-Ranges: none');
header('Content-Type: '.$mimetype);
header('Content-Length: '.filesize($fontpath));

readfile($fontpath);
die;
}

function font_not_found() {
    header('HTTP/1.0 404 not found');
    die('font was not found, sorry.');
```

```
}

/**
 * Caches a given font file.
 *
 * @param string $font The name of the font that was requested.
 * @param string $fontfile The location of the font file we want
to cache.
 * @param string $candidatelocation The location to cache it in.
 * @return string The path to the cached font.
 */
function cache_font($font, $fontfile, $candidatelocation) {
    global $CFG;
    $cachefont = "$candidatelocation/$font";

    clearstatcache();
    if (!file_exists($candidatelocation)) {
        @mkdir($candidatelocation, $CFG->directorypermissions,
true);
    }

    // Prevent serving of incomplete file from concurrent request,
// the rename() should be more atomic than copy().
    ignore_user_abort(true);
    if (@copy($fontfile, $cachefont.'.tmp')) {
        rename($cachefont.'.tmp', $cachefont);
        @chmod($cachefont, $CFG->filepermissions);
        @unlink($cachefont.'.tmp'); // Just in case anything
fails.
    }
    return $cachefont;
}
define('NO_DEBUG_DISPLAY', true);

// we need just the values from config.php and minlib.php
define('ABORT_AFTER_CONFIG', true);
```

Продолжение приложения А

```
require('../config.php'); // this stops immediately at the
beginning of lib/setup.php

if ($slashargument = min_get_slash_argument()) {
    $slashargument = ltrim($slashargument, '/');
    if (substr_count($slashargument, '/') < 3) {
        image_not_found();
    }
    if (strpos($slashargument, '_s/') === 0) {
        // Can't use SVG
        $slashargument = substr($slashargument, 3);
        $usesvg = false;
    } else {
        $usesvg = true;
    }
    // image must be last because it may contain "/"
    list($themename, $component, $rev, $image) = explode('/',
$slashargument, 4);
    $themename = min_clean_param($themename, 'SAFEDIR');
    $component = min_clean_param($component, 'SAFEDIR');
    $rev       = min_clean_param($rev, 'INT');
    $image     = min_clean_param($image, 'SAFEPATH');

} else {
    $themename = min_optional_param('theme', 'standard',
'SAFEDIR');
    $component = min_optional_param('component', 'core',
'SAFEDIR');
    $rev       = min_optional_param('rev', -1, 'INT');
    $image     = min_optional_param('image', '', 'SAFEPATH');
    $usesvg    = (bool)min_optional_param('svg', '1', 'INT');
}

if (empty($component) or $component === 'univer' or $component ===
'core') {
    $component = 'core';
}

if (empty($image)) {
    image_not_found();
}

if (file_exists("$CFG->dirroot/theme/$themename/config.php")) {
    // exists
} else if (!empty($CFG->themedir) and file_exists("$CFG->
themedir/$themename/config.php")) {
    // exists
} else {
    image_not_found();
}
}
```

Продолжение приложения A

```
$candidatelocation = "$CFG-
>localcachedir/theme/$rev/$themename/pix/$component";
$etag = sha1("$rev/$themename/$component/$image");

if ($rev > 0) {
    if (file_exists("$candidatelocation/$image.error")) {
        // This is a major speedup if there are multiple missing
images,
        // the only problem is that random requests may pollute
our cache.
        image_not_found();
    }
    $cacheimage = false;
    if ($usesvg && file_exists("$candidatelocation/$image.svg")) {
        $cacheimage = "$candidatelocation/$image.svg";
        $ext = 'svg';
    } else if (file_exists("$candidatelocation/$image.png")) {
        $cacheimage = "$candidatelocation/$image.png";
        $ext = 'png';
    } else if (file_exists("$candidatelocation/$image.gif")) {
        $cacheimage = "$candidatelocation/$image.gif";
        $ext = 'gif';
    } else if (file_exists("$candidatelocation/$image.jpg")) {
        $cacheimage = "$candidatelocation/$image.jpg";
        $ext = 'jpg';
    } else if (file_exists("$candidatelocation/$image.jpeg")) {
        $cacheimage = "$candidatelocation/$image.jpeg";
        $ext = 'jpeg';
    } else if (file_exists("$candidatelocation/$image.ico")) {
        $cacheimage = "$candidatelocation/$image.ico";
        $ext = 'ico';
    }
    if ($cacheimage) {
        if (!empty($_SERVER['HTTP_IF_NONE_MATCH']) ||
!empty($_SERVER['HTTP_IF_MODIFIED_SINCE'])) {
            // we do not actually need to verify the etag value
because our files
            // never change in cache because we increment the rev
parameter
            $lifetime = 60*60*24*60; // 60 days only - the
revision may get incremented quite often
            $mimetype = get_contenttype_from_ext($ext);
            header('HTTP/1.1 304 Not Modified');
            header('Expires: '. gmdate('D, d M Y H:i:s', time() +
$lifetime) .' GMT');
            header('Cache-Control: public, max-age='.$lifetime.',
no-transform');
            header('Content-Type: '.$mimetype);
            header('Etag: "'.$etag.'"');
            die;
        }
    }
}
```

Продолжение приложения А

```
    }
    send_cached_image($cacheimage, $etag);
}
}

//=====
// ok, now we need to start normal univer script, we need to load
// all libs and $DB
define('ABORT_AFTER_CONFIG_CANCEL', true);

define('NO_UNIVER_COOKIES', true); // Session not used here
define('NO_UPGRADE_CHECK', true); // Ignore upgrade check

require("$CFG->dirroot/lib/setup.php");

$theme = theme_config::load($themename);
$themerev = theme_get_revision();

if ($themerev <= 0 or $rev != $themerev) {
    // Do not send caching headers if they do not request current
    // revision,
    // we do not want to pollute browser caches with outdated
    // images.
    $imagefile = $theme->resolve_image_location($image,
    $component, $usesvg);
    if (empty($imagefile) or !is_readable($imagefile)) {
        image_not_found();
    }
    send_uncached_image($imagefile);
}

make_localcache_directory('theme', false);

// We're not using SVG and there is no cached version of this file
// (in any format).
// As we're going to be caching a format other than svg, and
// because svg use is conditional we need to ensure that at the same
// time we cache a version of the SVG if it exists. If we don't do
// this other users who ask for SVG would not ever get it as
// there is a cached image already of another format.
// Remember this only gets run once before any candidate exists,
// and only if we want a cached revision.
if (!$usesvg) {
    $imagefile = $theme->resolve_image_location($image,
    $component, true);
    if (!empty($imagefile) && is_readable($imagefile)) {
        $cacheimage = cache_image($image, $imagefile,
    $candidatelocation);
        $pathinfo = pathinfo($imagefile);
    }
}
```

Продолжение приложения A

```
// There is no SVG equivalent, we've just successfully cached an
image of another format.
    if ($pathinfo['extension'] !== 'svg') {
        // Serve the file as we would in a normal request.
        if (connection_aborted()) {
            die;
        }
        // Make sure nothing failed.
        clearstatcache();
        if (file_exists($cacheimage)) {
            send_cached_image($cacheimage, $etag);
        }
        send_uncached_image($imagefile);
        exit;
    }
}

// Either SVG was requested or we've cached a SVG version and are
ready to serve a regular format.
$imagefile = $theme->resolve_image_location($image, $component,
$usesvg);
if (empty($imagefile) or !is_readable($imagefile)) {
    if (!file_exists($candidatelocation)) {
        @mkdir($candidatelocation, $CFG->directorypermissions,
true);
    }
    // Make note we can not find this file.
    $cacheimage = "$candidatelocation/$image.error";
    $fp = fopen($cacheimage, 'w');
    fclose($fp);
    image_not_found();
}

$cacheimage = cache_image($image, $imagefile, $candidatelocation);
if (connection_aborted()) {
    die;
}
// Make sure nothing failed.
clearstatcache();
if (file_exists($cacheimage)) {
    send_cached_image($cacheimage, $etag);
}

send_uncached_image($imagefile);

//=====
=====
//=== utility functions ===
```

Продолжение приложения А

```
// we are not using filelib because we need to fine tune all
header
// parameters to get the best performance.

function send_cached_image($imagepath, $etag) {
    global $CFG;
    require("$CFG->dirroot/lib/xsendfilelib.php");

    $lifetime = 60*60*24*60; // 60 days only - the revision may
get incremented quite often
    $pathinfo = pathinfo($imagepath);
    $imagename = $pathinfo['filename'].'.'.$pathinfo['extension'];

    $mimetype = get_contenttype_from_ext($pathinfo['extension']);

    header('Etag: "'.$etag.'"');
    header('Content-Disposition: inline;
filename="'.$imagename.'"');
    header('Last-Modified: '. gmdate('D, d M Y H:i:s',
filetime($imagepath)) .' GMT');
    header('Expires: '. gmdate('D, d M Y H:i:s', time() +
$lifetime) .' GMT');
    header('Pragma: ');
    header('Cache-Control: public, max-age='.$lifetime.', no-
transform');
    header('Accept-Ranges: none');
    header('Content-Type: '.$mimetype);
    header('Content-Length: '.filesize($imagepath));

    if (xsendfile($imagepath)) {
        die;
    }

    // no need to gzip already compressed images ;- )

    readfile($imagepath);
    die;
}

function send_uncached_image($imagepath) {
    $pathinfo = pathinfo($imagepath);
    $imagename = $pathinfo['filename'].'.'.$pathinfo['extension'];

    $mimetype = get_contenttype_from_ext($pathinfo['extension']);

    header('Content-Disposition: inline;
filename="'.$imagename.'"');
    header('Last-Modified: '. gmdate('D, d M Y H:i:s', time()) .'
GMT');
```

Продолжение приложения А

```
header('Expires: '. gmdate('D, d M Y H:i:s', time() + 15) .'
GMT');
header('Pragma: ');
header('Accept-Ranges: none');
header('Content-Type: '.$mimetype);
header('Content-Length: '.filesize($imagepath));

readfile($imagepath);
die;
}

function image_not_found() {
    header('HTTP/1.0 404 not found');
    die('Image was not found, sorry.');
```

```
}

function get_contenttype_from_ext($ext) {
    switch ($ext) {
        case 'svg':
            return 'image/svg+xml';
        case 'png':
            return 'image/png';
        case 'gif':
            return 'image/gif';
        case 'jpg':
        case 'jpeg':
            return 'image/jpeg';
        case 'ico':
            return 'image/vnd.microsoft.icon';
    }
    return 'document/unknown';
}

/**
 * Caches a given image file.
 *
 * @param string $image The name of the image that was requested.
 * @param string $imagefile The location of the image file we want
to cache.
 * @param string $candidatelocation The location to cache it in.
 * @return string The path to the cached image.
 */
function cache_image($image, $imagefile, $candidatelocation) {
    global $CFG;
    $pathinfo = pathinfo($imagefile);
    $cacheimage =
"$candidatelocation/$image.".$pathinfo['extension'];

    clearstatcache();
    if (!file_exists(dirname($cacheimage))) {
```


Продолжение приложения А

```
        @mkdir(dirname($cacheimage), $CFG->directorypermissions,
true);
    }

    // Prevent serving of incomplete file from concurrent request,
    // the rename() should be more atomic than copy().
    ignore_user_abort(true);
    if (@copy($imagefile, $cacheimage.'.tmp')) {
        rename($cacheimage.'.tmp', $cacheimage);
        @chmod($cacheimage, $CFG->filepermissions);
        @unlink($cacheimage.'.tmp'); // just in case anything
fails
    }
    return $cacheimage;
}
$THEME->name = 'more';
$THEME->parents = array('clean', 'bootstrapbase');

$THEME->doctype = 'html5';
$THEME->sheets = array('custom');
$THEME->lessfile = 'univer';
$THEME->parents_exclude_sheets = array('bootstrapbase' =>
array('univer'), 'clean' => array('custom'));
$THEME->lessvariablescallback = 'theme_more_less_variables';
$THEME->extralesscallback = 'theme_more_extra_less';
$THEME->supportscssoptimisation = false;
$THEME->yuicssmodules = array();
$THEME->enable_dock = true;
$THEME->editor_sheets = array();

$THEME->rendererfactory = 'theme_overridden_renderer_factory';
$THEME->csspostprocess = 'theme_more_process_css';

$THEME->blockrtlmanipulations = array(
    'side-pre' => 'side-post',
    'side-post' => 'side-pre'
);
function theme_more_extra_less($theme) {
    $content = '';
    $imageurl = $theme->setting_file_url('backgroundimage',
'backgroundimage');
    // Sets the background image, and its settings.
    if (!empty($imageurl)) {
        $content .= 'body { ';
        $content .= "background-image: url('$imageurl');";
        if (!empty($theme->settings->backgroundfixed)) {
            $content .= 'background-attachment: fixed;';
        }
        if (!empty($theme->settings->backgroundposition)) {
```

Продолжение приложения А

```
        $content .= 'background-position: ' . str_replace('_',
' ', $theme->settings->backgroundposition) . ';';
    }
    if (!empty($theme->settings->backgroundrepeat)) {
        $content .= 'background-repeat: ' . $theme->settings-
>backgroundrepeat . ';';
    }
    $content .= ' }';
}
// If there the user wants a background for the content, we
need to make it look consistent,
// therefore we need to round its borders, and adapt the
border colour.
if (!empty($theme->settings->contentbackground)) {
    $content .= '
        #region-main {
            .well;
            background-color: ' . $theme->settings-
>contentbackground . ';
            border-color: darken(' . $theme->settings-
>contentbackground . ', 7%);
        }';
}
return $content;
}

/**
 * Returns variables for LESS.
 *
 * We will inject some LESS variables from the settings that the
user has defined
 * for the theme. No need to write some custom LESS for this.
 *
 * @param theme_config $theme The theme config object.
 * @return array of LESS variables without the @.
 */
function theme_more_less_variables($theme) {
    $variables = array();
    if (!empty($theme->settings->bodybackground)) {
        $variables['bodyBackground'] = $theme->settings-
>bodybackground;
    }
    if (!empty($theme->settings->textcolor)) {
        $variables['textColor'] = $theme->settings->textcolor;
    }
    if (!empty($theme->settings->linkcolor)) {
        $variables['linkColor'] = $theme->settings->linkcolor;
    }
    if (!empty($theme->settings->secondarybackground)) {
```

Продолжение приложения A

```
        $variables['wellBackground'] = $theme->settings-
>secondarybackground;
    }
    return $variables;
}

/**
 * Parses CSS before it is cached.
 *
 * This function can make alterations and replace patterns within
the CSS.
 *
 * @param string $css The CSS
 * @param theme_config $theme The theme config object.
 * @return string The parsed CSS The parsed CSS.
 */
function theme_more_process_css($css, $theme) {

    // Set the background image for the logo.
    $logo = $theme->setting_file_url('logo', 'logo');
    $css = theme_more_set_logo($css, $logo);

    // Set custom CSS.
    if (!empty($theme->settings->customcss)) {
        $customcss = $theme->settings->customcss;
    } else {
        $customcss = null;
    }
    $css = theme_more_set_customcss($css, $customcss);

    return $css;
}

/**
 * Adds the logo to CSS.
 *
 * @param string $css The CSS.
 * @param string $logo The URL of the logo.
 * @return string The parsed CSS
 */
function theme_more_set_logo($css, $logo) {
    $tag = '[[setting:logo]]';
    $replacement = $logo;
    if (is_null($replacement)) {
        $replacement = '';
    }

    $css = str_replace($tag, $replacement, $css);

    return $css;
}
```

Продолжение приложения А

```
}

/**
 * Serves any files associated with the theme settings.
 *
 * @param stdClass $course
 * @param stdClass $cm
 * @param context $context
 * @param string $filearea
 * @param array $args
 * @param bool $forcedownload
 * @param array $options
 * @return bool
 */
function theme_more_pluginfile($course, $cm, $context, $filearea,
    $args, $forcedownload, array $options = array()) {
    if ($context->contextlevel == CONTEXT_SYSTEM && ($filearea ===
'logo' || $filearea === 'backgroundimage')) {
        $theme = theme_config::load('more');
        return $theme->setting_file_serve($filearea, $args,
$forcedownload, $options);
    } else {
        send_file_not_found();
    }
}

/**
 * Adds any custom CSS to the CSS before it is cached.
 *
 * @param string $css The original CSS.
 * @param string $customcss The custom CSS to add.
 * @return string The CSS which now contains our custom CSS.
 */
function theme_more_set_customcss($css, $customcss) {
    $tag = '[[setting:customcss]]';
    $replacement = $customcss;
    if (is_null($replacement)) {
        $replacement = '';
    }

    $css = str_replace($tag, $replacement, $css);

    return $css;
}
defined(SKT) || die;

if ($ADMIN->fulltree) {

    // @textColor setting.
    $name = 'theme_more/textcolor';
```

Продолжение приложения A

```
$title = get_string('textcolor', 'theme_more');
$description = get_string('textcolor_desc', 'theme_more');
$default = '#333366';
$setting = new admin_setting_configcolourpicker($name, $title,
$description, $default, null, false);
$setting->set_updatedcallback('theme_reset_all_caches');
$settings->add($setting);

// @linkColor setting.
$name = 'theme_more/linkcolor';
$title = get_string('linkcolor', 'theme_more');
$description = get_string('linkcolor_desc', 'theme_more');
$default = '#FF6500';
$setting = new admin_setting_configcolourpicker($name, $title,
$description, $default, null, false);
$setting->set_updatedcallback('theme_reset_all_caches');
$settings->add($setting);

// @bodyBackground setting.
$name = 'theme_more/bodybackground';
$title = get_string('bodybackground', 'theme_more');
$description = get_string('bodybackground_desc',
'theme_more');
$default = '';
$setting = new admin_setting_configcolourpicker($name, $title,
$description, $default, null, false);
$setting->set_updatedcallback('theme_reset_all_caches');
$settings->add($setting);

// Background image setting.
$name = 'theme_more/backgroundimage';
$title = get_string('backgroundimage', 'theme_more');
$description = get_string('backgroundimage_desc',
'theme_more');
$setting = new admin_setting_configstoredfile($name, $title,
$description, 'backgroundimage');
$setting->set_updatedcallback('theme_reset_all_caches');
$settings->add($setting);

// Background repeat setting.
$name = 'theme_more/backgroundrepeat';
$title = get_string('backgroundrepeat', 'theme_more');
$description = get_string('backgroundrepeat_desc',
'theme_more');
$default = 'repeat';
$choices = array(
    '0' => get_string('default'),
    'repeat' => get_string('backgroundrepeatrepeat',
'theme_more'),
```

Продолжение приложения А

```
'repeat-x' => get_string('backgroundrepeatrepeatx',
'theme_more'),
'repeat-y' => get_string('backgroundrepeatrepeaty',
'theme_more'),
'no-repeat' => get_string('backgroundrepeatnorepeat',
'theme_more'),
);
$setting = new admin_setting_configselect($name, $title,
$description, $default, $choices);
$setting->set_updatedcallback('theme_reset_all_caches');
$settings->add($setting);

// Background position setting.
$name = 'theme_more/backgroundposition';
$title = get_string('backgroundposition', 'theme_more');
$description = get_string('backgroundposition_desc',
'theme_more');
$default = '0';
$choices = array(
    '0' => get_string('default'),
    'left_top' => get_string('backgroundpositionlefttop',
'theme_more'),
    'left_center' =>
get_string('backgroundpositionleftcenter', 'theme_more'),
    'left_bottom' =>
get_string('backgroundpositionleftbottom', 'theme_more'),
    'right_top' => get_string('backgroundpositionrighttop',
'theme_more'),
    'right_center' =>
get_string('backgroundpositionrightcenter', 'theme_more'),
    'right_bottom' =>
get_string('backgroundpositionrightbottom', 'theme_more'),
    'center_top' => get_string('backgroundpositioncentertop',
'theme_more'),
    'center_center' =>
get_string('backgroundpositioncentercenter', 'theme_more'),
    'center_bottom' =>
get_string('backgroundpositioncenterbottom', 'theme_more'),
);
$setting = new admin_setting_configselect($name, $title,
$description, $default, $choices);
$setting->set_updatedcallback('theme_reset_all_caches');
$settings->add($setting);

// Background fixed setting.
$name = 'theme_more/backgroundfixed';
$title = get_string('backgroundfixed', 'theme_more');
$description = get_string('backgroundfixed_desc',
'theme_more');
```

Продолжение приложения A

```
$setting = new admin_setting_configcheckbox($name, $title,
$description, 0);
$setting->set_updatedcallback('theme_reset_all_caches');
$settings->add($setting);

// Main content background color.
$name = 'theme_more/contentbackground';
$title = get_string('contentbackground', 'theme_more');
$description = get_string('contentbackground_desc',
'theme_more');
$default = '#FFFFFF';
$setting = new admin_setting_configcolourpicker($name, $title,
$description, $default, null, false);
$setting->set_updatedcallback('theme_reset_all_caches');
$settings->add($setting);

// Secondary background color.
$name = 'theme_more/secondarybackground';
$title = get_string('secondarybackground', 'theme_more');
$description = get_string('secondarybackground_desc',
'theme_more');
$default = '#FFFFFF';
$setting = new admin_setting_configcolourpicker($name, $title,
$description, $default, null, false);
$setting->set_updatedcallback('theme_reset_all_caches');
$settings->add($setting);

// Invert Navbar to dark background.
$name = 'theme_more/invert';
$title = get_string('invert', 'theme_more');
$description = get_string('invertdesc', 'theme_more');
$setting = new admin_setting_configcheckbox($name, $title,
$description, 1);
$setting->set_updatedcallback('theme_reset_all_caches');
$settings->add($setting);

// Logo file setting.
$name = 'theme_more/logo';
$title = get_string('logo', 'theme_more');
$description = get_string('logodesc', 'theme_more');
$setting = new admin_setting_configstoredfile($name, $title,
$description, 'logo');
$setting->set_updatedcallback('theme_reset_all_caches');
$settings->add($setting);
```

Приложение Б
Каталог функций программного обеспечения

№ п/п	Наименование (содержание) функций	Объем функций (строк исходного кода)		
		С использованием среды разработки приложений		
		Delphi (Borland)	C++ Builder (Borland)	Visual C++ (Microsoft)
1. Ввод, анализ входной информации, генерация кодов и процессор входного языка				
101	Организация ввода информации	100	110	150
102	Контроль, предварительная обработка и ввод информации	550		
103	Анализ входного языка (синтаксический и семантический)	630	660	980
104	Преобразование операторов входного языка и команды другого языка	1050	1050	980
105	Обработка входного заказа и формирование таблиц	750	900	1340
106	Преобразование входного языка в машинные команды (транслятор, препроцессор, макрогенератор)	1300	4300	5100
107	Синтаксический и семантический анализ входного языка и генерация кодов команд	8700	5400	5400
108	Процессор языка	3000	2300	2300
109	Организация ввода/вывода информации в интерактивном режиме	220	220	320
110	Организация ввода/вывода информации с сети терминалов	3680	3340	3200
110	Управление вводом/выводом	2700	2900	2400
2. Формирование, введение и обслуживание баз данных				
201	Генерация структуры базы данных	3450	3950	4300
202	Генерация структуры базы данных	1540	1610	2060
203	Формирование баз данных	1700	1750	2180
204	Обработка наборов и записей базы данных	2050	2350	2670
205	Обслуживание базы данных в пакетном режиме	1030	1100	1260
206	Обслуживание базы данных в интерактивном режиме	3800	4400	6950
207	Манипулирование данными	8400	8670	9550
208	Организация поиска и поиск в базе данных	5230	5460	5480
209	Реорганизация базы данных	130	190	220
210	Загрузки базы данных	3150	2950	2780
3. Формирование и обработка файлов				
301	Формирование последовательного файла	340	360	290
302	Сортировка файла	340	360	290
303	Автоматическая сортировка файлов	1040	1150	930

304	Обслуживание файлов	520	540	420
305	Обработка файлов	750	800	720
306	Обработка файлов в диалоговом режиме	2400	2600	3050
307	Совместная обработка группы файлов	4900	5300	6180
308	Управление файлами	5130	5380	5750
309	Формирование файла	1100	1080	1020
4. Генерация программ и ПО, а также настройка ПО				
401	Генерация рабочих программ	3680	4120	3360
402	Генерация программ по описанию пользователей	10870	12330	9880
403	Формирование служебных таблиц	570	620	1070
404	Система генерации ПО	5120	5340	4980
405	Система генерации ПО	250	300	370
5. Управление ПО, компонентами ПО и внешними устройствами				
501	Монитор ПО (управление работой компонентов)	350	360	740
502	Монитор системы (управление работой комплекса ПО)	3750	3880	7740
503	Управление внешними устройствами и объектами	6850	7340	5900
504	Обработка прерываний	890	730	540
505	Управление внешней памятью	250	210	200
506	Обработка ошибочных и сбойных ситуаций	430	410	410
507	Обеспечение интерфейса между компонентами	730	750	970
6. Отладка прикладных программ, обмен информацией между магнитным диском и магнитной лентой, вспомогательные программные функции				
601	Отладка прикладных программ в интерактивном режиме	4500	4700	4300
602	Обмен информацией между магнитным диском и магнитной лентой	*	*	*
603	Копирование наборов данных на магнитной ленте и восстановление	*	*	*
604	Справка и обучение	680	680	720
605	Вспомогательные и сервисные программы	460	490	580
7. Расчетные задачи, формирование и вывод на внешние носители (АЦПУ) документов сложной формы и файлов				
701	Матстатистика и прогнозирование	8370	9570	9320
702	Расчетные задачи (расчет режимов обр-ки)	12600	15300	14800
703	Расчет показателей	410	500	460
704	Процессор отчетов	1070	1230	3200
705	Формирование и вывод на внеш.носители	2650	2850	3500
706	Предварит.обработка и печать файлов	540	560	470
707	Графический вывод результатов	300		480
708	Интерактивный редактор текста	3800	3910	4540

Приложение В

Укрупненные нормы времени на разработку ПО в зависимости от
уточненного объема ПО и группы сложности ПО

Объем ПО(строки исходного кода, LOC)	Категории сложности ПО			Категории сложности ПО
	1-я	2-я	3-я	
<i>I</i>	2	3	4	5
200	—	—	21	1
300	—	—	23	2
400	—	—	25	3
500	—	—	27	4
600	—	33	28	5
700	—	36	30	6
800	—	38	32	7
900	—	40	34	8
1000	51	43	36	9
1200	54	45	38	10
1400	57	48	40	11
1600	60	50	42	12
1800	64	54	45	13
2000	68	57	48	14
2200	73	61	51	15
2400	76	64	54	16
2600	81	68	57	17
2800	86	72	60	18
3000	91	76	64	19
3200	97	81	68	20
3400	103	86	72	21
3600	110	92	77	22
3800	117	98	82	23
4000	124	104	87	24
4200	133	111	93	25
4400	141	118	99	26
4600	151	126	105	27
4800	160	134	112	28
5000	170	142	119	29
5500	182	152	127	30
6000	194	162	135	31
6500	206	172	144	32
7000	220	184	154	33
7500	235	196	164	34
8000	252	210	175	35

Приложение Г
Дополнительные коэффициенты сложности ПО

Характеристика ПО	Значения K_c
1. Функционирование ПО в расширенной операционной среде (связь с другими ПО)	0,08
2. Интерактивный доступ	0,06
3. Обеспечение хранения, ведения и поиска данных в сложных структурах	0,07
4. Наличие у ПО одновременно нескольких характеристик по табл.Г4.1, приложение Г	
4.1 2 характеристики	0,12
4.2 3 характеристики	0,18
4.3 Свыше 3-х характеристик	0,26

Приложение Д
Значения поправочного коэффициента, учитывающего использование стандартных модулей типовых программ и ПО (K_T)

Степень охвата реализуемых функций разрабатываемого ПО стандартными модулями, типовыми программами и ПО	Значения K_T
1. От 60 % и выше	0,6
2. От 40 % до 60	0,7
3. От 20 % до 40 %	0,8
4. До 20 %	0,9
5. Типовые программы и ПО не используемые для реализации функций разрабатываемого ПО	1,0

Приложение Ж
Поправочные коэффициенты, учитывающие новизну ПО (K_H)

Категория Новизны	Степень новизны	Использование		Значение K_H
		На основе нового типа ПК	В среде новой ОС	
А	Принципиально новые ПО, не имеющие доступных аналогов	+	+	1,75
		-	+	1,6
		+	-	1,2
		-	-	1,0
Б	ПО, являющиеся развитием определенного параметрического ряда ПО	+	+	1,0
		-	-	0,9
		+	-	0,8
В	ПО, являющиеся развитием определенного параметрического ряда ПО, разработанных для ранее освоенных типов конфигурации ПК и ОС	-	-	0,7

Приложение 3

Оценка значений среднего машинного времени на отладку 100 строк
исходного кода без применения ПО

Наименование подсистемы АС и СОД	Средний расход машинного времени, ч/100 строк кода
1. Общесистемные задачи: ведение линейных файлов, информационно-поисковые системы и информационно-справочные системы, сбор информации, ввод информации, расширение возможностей средств обработки данных, организация вычислительного процесса	12
2. Задачи расчетного характера	15
3. Оперативное управление производством, расчеты по Ценообразованию	7
4. Техническая подготовка производства, транспортное, ремонтное, энергетическое и инструментальное обслуживание производства	8
5. Бухгалтерский учет, финансовые расчеты, учет пенсий и пособий, учет страховых операций, качество продукции	13
6. Управление кадрами	13
7. Техничко-экономическое планирование	13
8. Материально-техническое снабжение, реализация и сбыт готовой Продукции	13