

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

Некоммерческое акционерное общество
АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ

Кафедра Информационные системы

«Допущен к защите»

Заведующий кафедрой _____

(Ф.И.О., ученая степень, звание)

« _____ » 20__ г.
(подпись)

ДИПЛОМНЫЙ ПРОЕКТ

На тему: Анализ и моделирование информационной системы поддержки принятия решений в области сельского хозяйства

Специальность _____

Выполнил (а) Андрасбаева С.А. ИС-10-2
(Фамилия и инициалы) группа

Научный руководитель Или А.Т.
(Фамилия и инициалы, ученая степень, звание)

Консультанты:

по экономической части:

Беккучева А.У., к.т.н., доцент
(Фамилия и инициалы, ученая степень, звание)
А.У. « 26 » июль 2014 г.
(подпись)

по безопасности жизнедеятельности:

Самылова А.С., ст. преподаватель
(Фамилия и инициалы, ученая степень, звание)
А.С. « 11 » июль 2014 г.
(подпись)

по применению вычислительной техники:

Жаурбаева Б.К., к.т.н., ст. преподаватель
(Фамилия и инициалы, ученая степень, звание)
Б.К. « 05 » 06 2014 г.
(подпись)

Нормоконтролер: Или А.Т., к.т.н., доцент
(Фамилия и инициалы, ученая степень, звание)
И.А. « 09 » 06 2014 г.
(подпись)

Рецензент: _____
(Фамилия и инициалы, ученая степень, звание)
« _____ » 20__ г.
(подпись)

Алматы 2014 г.

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РЕСПУБЛИКИ КАЗАХСТАН

Некоммерческое акционерное общество
АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ

Факультет „Информационные технологии“
Специальность 58070300 – „Информационные системы“
Кафедра „Информационные системы“

ЗАДАНИЕ

на выполнение дипломного проекта

Студент Андабаева Сайра Ахмеджановна
(фамилия, имя, отчество)

Тема проекта Анализ и моделирование информационной системы поддержки принятия решений в области сельского хозяйства

утверждена приказом ректора № ___ от «___» сентября 20__ г.

Срок сдачи законченной работы «___» _____ 20__ г.

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

Исходные данные: задание к дипломному проекту. Требуемые параметры результатов проектирования: модель интеллектуальной информационной системы поддержки принятия решений, ER-диаграмма баз данных.

Исходные данные объекта: результаты обследования предметной области.

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

1. Анализ и исследование области сельского хозяйства
2. Проектирование и разработка интеллектуальной информационной системы поддержки принятия решений „Агроном“
3. Разработка программного обеспечения
4. Технико-экономическое обоснование проекта
5. Анализ условий труда

Перечень графического материала (с точным указанием обязательных чертежей)

1. Диаграмма прецедентов
2. Диаграмма компонентов
3. Диаграмма последовательности
4. Диаграмма развертывания
5. Схема интеллектуальной информационной системы.

Рекомендуемая основная литература

1. Горев А., Малашигин С., Ахалин Р. Формативная работа с СУБД, СПб.: Питер, 1997 г. - 425с.

Консультанты по проекту с указанием относящихся к ним разделов

Раздел	Консультант	Сроки	Подпись
БТД	Беломыслова А.С.	20.03 - 4.05.14	С.А.
Экспл. часть	Беломыслова А.С.		С.А.

Аннотация

В основе предлагаемой интеллектуальной системы поддержки принятия решений лежит технология Data Mining, математические модели и методы теории принятия решений. Архитектура создаваемой системы, включает в себя хранилище данных, базу знаний экспертов по сельскому хозяйству, базу математических моделей, геоинформационную систему (ГИС), датчики сбора данных, интерфейсы передачи данных между ИСППР и датчиками сбора данных.

Ключевые слова: сельское хозяйство, интеллектуальная система поддержки принятия решения, информационные технологии, информационные системы, технология Data Mining, теория принятия решений.

Аңдатпа

"Ауыл шаруашылық" интеллектуалдық ақпараттық шешім қабылдау және сүйеніш жүйесінің негізінде Data Mining технологиясы жатады, математикалық қалыптармен шешім қабылдау теория әдістері салынған.

Жасалатын жүйе сәулеті өзіне ауыл шаруашылыққа, математикалық қалыптың базасын, геоақпарат жүйені (ГАЗ), деректердің жиынының бергіштерін, интерфейс деректердің берілісінің ара интеллектуалдық шешім қабылдау және деректердің жиындарынан тұрады.

Бұлақты сөздер: ақпараттық технологиялар, ақпараттық жүйелер, Data Mining технологиясы, ауыл шаруашылық қағида, зияткерлік жүйе сүйеніш қағидасы.

Annotation

The intellectual system of support of making decision is "Agriculture" Offered intellectual system of support of making decision technology of Data Mining, mathematical models and methods of theory of making decision, is the basis of. Architecture of the created system, plugs in itself the depository of data, base of knowledge of experts on agriculture, base of mathematical models, geographic information system(HYBRID-TYPE DEVICE), sensors of capture of data, interfaces of communication of data between system and sensors of capture of data.

Keywords: agriculture, intellectual system of support of decision-making, information technologies, informative systems, technology of Data Mining, theory of making decision.

Содержание

Введение	9
1 Информационные технологии в сельском хозяйстве	11
1.1 Применение информационных технологий в сельском хозяйстве	11
1.2 Проблемы автоматизации ИИСППР «Агроном»	13
1.3 Обзор используемых программных и инструментальных средств	14
2 Проектирование и разработка ИИСППР «Агроном»	19
2.1 Системный анализ объекта исследования	19
2.2 Разработка UML-диаграмм ИИСППР «Агроном»	21
2.3 Разработка базы данных ИИСППР «Агроном»	26
3 Разработка программного обеспечения ИИСППР «Агроном»	31
3.1 Назначение и условия выполнения программы	31
3.2 Разработка интерфейса ИИСППР «Агроном»	41
4 Техничко-экономическое обоснование проекта	55
4.1 Описание работы и обоснование необходимости	55
4.2 Техничко – экономическое обоснование дипломной работы	56
4.3 Расчет затрат на разработку ИИСППР «Агроном»	57
4.4 Определение возможной (договорной) цены ИИСППР «Агроном»	64
4.5 Оценка социально - экономических результатов функционирования ИИСППР «Агроном»	65
5 Безопасность жизнедеятельности	67
5.1 Анализ условий труда в рабочем помещении	67
5.2 Особенности вынужденного движения людей	70
5.3 Параметры движения людей	70
5.4 Эвакуационные выходы и пути	74
5.5 Время эвакуации людей из отдельных помещений и зданий в целом	76
Заключение	80
Список литературы	81
Приложение А Техническое задание	82
Приложение Б Листинг программы	100

Введение

Национальный доход Казахстана – это нефть и сельское хозяйство. 38% процентов дохода страны приходится именно на сельское хозяйство. К нашему счастью наша страна богата различными ресурсами, но особенно она богата ресурсами земельными. Земельные площади Казахстана составляют двести двадцать два с половиной миллионов гектар. Восемьдесят два процента из его общих земельных площадей используется для сельскохозяйственного угодья. Более тридцати шести миллионов гектар составляют пашни, по их площадям Казахстан занимает шестое место в мире. Казахстан всегда занимал не плохие места по производству сельскохозяйственных культур, так допустим, в 2001 году мы заняли третье место по производству зерновых и зернобобовых культур.

Сельское хозяйство – это одна из отраслей экономики, к состоянию которого трепетно относятся как граждане нашей страны, так и сам президент. Оно играет важную роль, но многие факторы тормозят развитие сельского хозяйства и самый важный фактор – это недостаток квалифицированных кадров, вследствие чего нет современной техники, не развита селекция, нет новых методов посадки и так далее, которые не дают возможность получения максимального результата.

При изучении данной области, проводились наблюдения в Енбекшиказахском районе Алматинской области, поселке Каратурык, где люди зарабатывают на хлеб в основном земледелием: выращиванием овощей, яблок и табака.

В ходе наблюдения было замечено, что весь процесс выращивания культуры состоит из следующих этапов:

- подготовительные работы;
- посев;
- уход;
- сбор.

Подготовительные работы – процесс, при котором идет подготовка земли для посева и подготовка саженцев. Именно здесь проводится анализ почвы, удобрение почвы, капельное орошение, анализ семян.

Посев - это внесение семян в специально подготовленную почву. Это один из сложных и важных процессов, который в свою очередь делится на:

- определение способа посева;
- вычисление сроков посева;
- вычисление глубины заделки семян.

Для определения способа посева, срока посева и глубины заделки семян фермеры три раза в сутки проверяют температуру почвы и влажность.

После посева, следующим процессом является уход, к которому относятся поливные работы, прополка, удобрения.

В зависимости от погодных условий и определяется тип ухода: боронование, культивация, прополка, полив. А количество удобрения необходимо рассчитывать, опираясь на анализ почвы.

Завершающим процессом является «Сбор», когда владелец получает и собирает урожай для реализации и получения прибыли.

После наблюдения, была замечена следующая проблема. С каждым годом люди получают все лучше результаты посадки, опираясь на свои предыдущие опыты. Но фермеры при посадке культуры всегда идут на большой риск. Начинающие фермеры, от не знания, сажают культуру в той почве, в какой придется, а затем возмущаются, что культура не проросла; сажают то количество семян, которое смогли купить и при этом, даже не вычисляя сроки посадки семян, глубину заделки семян; удобряют в том количестве, в каком смогли, что не есть хорошо и все это в итоге приводит к тому, что фермер разочаровывается, и весь труд его пропадает даром. А опытные фермеры, для получения хорошего результата проводят огромную работу: постоянное слежение температуры и влажности, подсчет количества семян, подсчет необходимого количества удобрения, анализ земли, вычисление срока посева, сам посев, уход.

Таким образом, успех зависит от того, как фермер будет управлять всем процессом производства сельскохозяйственных продуктов. Управление подразумевает под собой принятие решений по выбору почвы, культуры, сортов, удобрения и т.д. с целью минимизации затрат и максимизации прибыли, то есть эффективное и рациональное использование материальных, трудовых и финансовых ресурсов.

Управление в сельском хозяйстве в значительной степени предполагает принятие решений в условиях неопределенности, обусловленной тремя основными причинами: отсутствие текущих данных о состоянии природы; недостаточность знаний о биологических и физических системах; случайный характер протекающих процессов.

Учитывая исключительную важность сельского хозяйства в вопросах обеспечения населения продуктами питания, актуальной является задача эффективного управления процессом производства сельскохозяйственных продуктов с тем, чтобы максимально повысить урожайность с целью получения максимальной прибыли.

В настоящее время принятие решений обычно основывается на отдельных расчетах и экспертных оценках, т.е. на «ручных» способах обработки информации.

В связи с вышесказанным в настоящем проекте рассматриваются вопросы автоматизации управления процессом производства сельскохозяйственных продуктов. Автоматизация достигается за счет создания информационной интеллектуальной системы поддержки принятия решения, которая поможет фермеру принять эффективное решение по выбору почвы, культуры, сортов, удобрения и т.д.

1 Информационные технологии в сельском хозяйстве

1.1 Применение информационных технологий в сельском хозяйстве

Сельскохозяйственное производство в Казахстане не отличается от уровня в 60-70-х годов прошлого столетия. Развитие агропромышленного комплекса тормозится из-за низкого уровня технологической оснащенности и недостатка квалифицированных кадров. На сегодняшний день мировой и европейский опыт ведения сельхозработ уже напрямую связан с информационными технологиями. В Казахстане и в странах СНГ данное направление осталось не тронутым.

Опыт ведущих стран с развитой аграрной сферой свидетельствует, что все они прошли своего рода «технологическую революцию». Обычное земледелие, которое высчитывалось благодаря опыту заменяется сейчас точными науками. Больше всего используются геоинформационные технологии, многооперационные энергосберегающие сельскохозяйственные агрегаты, селекция высокоурожайных сортов растений и выведение высокопродуктивных пород животных, создание биологически активных кормовых добавок, новых лекарственных средств для животных, современные методы борьбы с эпизоотиями, карантинными болезнями животных и растений.

Многие схожие задачи сейчас имеют примеры решений на территории Российской Федерации. Хозяйства, руководство которых своевременно и точно оценивает ситуацию и переходит на ресурсосберегающие инновационные технологии, начинают использовать различные доступные возможности информационных технологий. К сожалению, «заряжаются» идеями современных технологий сотни руководителей, но осмеливаются начать их внедрение только десятки.

На новом витке сельскохозяйственных реформ остро стоит необходимость производства и распространения технических и информационных средств модернизации агропромышленных предприятий.

На сегодняшний день есть специализированные системы, где формируются информационные ресурсы, происходит сбор, обобщение и адаптация баз данных, прикладных программных продуктов и рекомендаций по повышению эффективности сельскохозяйственного производства.

Таблица 1.1 - Примеры программного обеспечения сельскохозяйственной направленности в странах СНГ

Наименование	Краткая характеристика
«Ветеринария и животноводство»	Информационная система, где содержится информация о кормлении, ветеринарном обследовании, список животных
Комплекс «Ветеринария и животноводство»	Расчет нужного количества кормления животных. Автоматизация племенного учета
Программный продукт «Механизация»	База данных о сельскохозяйственной технике. Список оборудования перерабатывающей промышленности и предприятий, выпускающих и поставляющих данную технику
«Агрохимическое обслуживание и карантин растений»	Список обрабатывающих средств, содержащая информацию по минеральным, органическим и органоминеральным удобрениям и химическим мелиорантам, так же список поставщиков
Программа «Традиционные и перспективные технологии возделывания сельскохозяйственных культур»	Планирование основных технологий производств видов культур с учетом зональных, производственно-технических условий. Так же учет финансовых условий от реализации
Программный продукт «Планирование хозяйственной деятельности на основе принципа валового дохода»	Планирование и анализ хозяйственной деятельности сельскохозяйственных предприятий. Подсчет доходов, расходов и различной прибыли

Во многих странах большое внимание стали уделять деятельности по пропаганде и распространению новых сельскохозяйственных культур, сортов, технологий и техники. Теоретическая часть развития сельскохозяйственной сферы далеко ушла вперед, стали производиться новейшие технологии селекции, но информационные технологии в данной области особого развития не получили.

Все больше открываются, отвечающие современным требованиям, обучающие агроцентры. В данных центрах обучают как обычных фермеров так и руководителей, специалистов и работников сельскохозяйственных предприятий. Могут получить уроки специалисты региональных и районных администраций, фермеров путем организации курсов повышения

квалификации и семинаров, проведения встреч, совещаний, круглых столов[1].

1.2 Проблемы автоматизации ИИСППР «Агроном»

Данная система необходима для решения нижеперечисленных поставленных задач.

Выбор почвы под определенную культуру. Пользователь вводит данные относительно культуры: выбирает тип, вид и сорт культуры, а система, опираясь базу знаний, где содержатся данные относительно анализа почвы, предлагает область или регион, где данная культура точно даст результат. Либо пользователь вводит область, регион, где он проживает и система предлагает список культур, которые могут произрасти.

Предложение необходимого количества удобрения, анализируя состав почвы. Для этого пользователь должен указать количество гектар, тип культуры и регион.

Расчет количества семян или саженцев для посадки в зависимости от площади гектар и типа культуры. Пользователь вводит данные о типе культуре и количестве гектар земли.

Предложение сроков посева и глубины заделки семян, учитывая температуру и относительную влажность почвы.

На данный момент информация о температуре и влажности почвы будут вводиться вручную, но в будущем планируется считывание информации с датчиков. То есть на данной территории будут установлены датчики, которые три раза в сутки проверяет влажность и температуру почвы, затем, используя беспроводные технологии, данные передаются в систему. После считывания и анализа данных система предлагает решение.

Выбор оптимального сорта, учитывая погодные и климатические условия.

Всем известно, что в каждом регионе преобладает свой климат, который очень сильно влияет как на почву так и на саму культуру. Система будет проводить анализ по погодным условиям: жаркая и сухая погода, нормальная погода, холодная погода, дождливая. То есть системой будет предложен тот сорт, который во всех природных условиях даст наибольший урожай. Для решения данного вопроса будет использована теория игр, где противниками являются погода и пользователь.

Предложение по выгодности: выгодно ли данную почву удобрять под определенную культуру либо выгоднее выбрать другой сорт, который можно посадить, не удобряя почву.

То есть имеются две культуры. Допустим реализация первой культуры за килограмм равно 200 тенге, а вторая культура 50 тенге. Но для посадки первой культуры, стоимость которой 200 тенге, почву необходимо постоянно подпитывать и удобрять, то есть будут дополнительные затраты, а для второй культуры, стоимость которой 50 тенге, удобрения не нужны, состав почвы

итак благоприятен для посадки данного типа. Что же в итоге выгоднее посадить? Данная система будет давать ответ на этот вопрос.

Выбор процесса, учитывая температуру и влажность почвы, определять сроки полива, боронования, прополки.

Составление готового документа план.

Так же имеются свои фишки данного проекта. Будет задействована ГИС система, где в проекте будет выводиться карта Казахстана и по областям можно будет посмотреть состав, относительную влажность и температуру почвы, список культур, которые могут произрасти в данной области. На данный момент ГИС система предоставляет снимок поверхности земли, а для пользователя ИИСППР лишь снимок земли недостаточен, предлагается установление датчиков, которые показывают и внутреннее строение почвы, ее особенности. Также будет иметься база знаний, которая постоянно обновляется при помощи интернет, где хранится полная информация о различной культуре, почве, удобрениях, то есть краткий справочник землепользователя.

На данный момент ИИСППР реализована не полностью. Реализован интерфейс программы, база знаний и некоторые вышеперечисленные задачи. Представим такую ситуацию, когда фермер - новичок решил заниматься сельским хозяйством. У него есть участок земли, и он должен определиться, какую культуру и какой сорт будет выращивать, но до этого момента он никогда не занимался земледелием и не представляет, что можно взрастить на данном регионе. Появляются задачи, которые сейчас реализованы в ИИСППР:

- выбор культуры для посадки в зависимости от региона, опираясь на статистические данные (предыдущий опыт), используя технологию Data Mining;

- выбор оптимального сорта, учитывая погодные и климатические условия. Для решения данного вопроса будет использована теория игр, где противниками являются погода и пользователь.

1.3 Обзор используемых программных и инструментальных средств

На сегодняшний день существует множество различных программных и инструментальных средств, с помощью которых можно создать информационные, различные интеллектуальные системы для автоматизирования определенных областей жизнедеятельности.

Сейчас рынок информационных технологий предлагает очень много инструментальных средств для разработки информационных систем, которые позволяют скоростно решать проблемы управления кадрами. Так же при помощи таких систем можно улучшать эффективность бизнес-процессов в организациях любых уровней и различных направлений деятельности. Такой широкий выбор разнообразия программных продуктов позволяют подобрать

единственно верное инструментальное средство, которое поможет удовлетворить потребности пользователя в требуемом объеме [2].

Также, при создании баз данных необходимым является выбор такого инструментального средства разработки, который бы удовлетворял такие потребности конечных пользователей как: актуальность информации для организации; производительность БД, обеспечивающая получение требуемых данных за короткое время; гибкость и возможность подстройки под пользователя; масштабируемость при реорганизации и расширении предметной области; кроссплатформенность; целостность базы данных; удобство использования.

В данном дипломном проекте была использована MySQL. MySQL является решением для малых и средних приложений. Входит в состав серверов WAMP, AppServ, LAMP и в портативные сборки серверов Денвер, XAMPP. 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.

Кроме СУБД MySQL весь программный код написан на языке C#.

C# — это один из объектно-ориентированных языков программирования. Его разработали в 1998—2001. Разработкой занималась группа инженеров, а руководил ими Андерс Хейлсберг. Его разрабатывали в Microsoft как язык разработки приложений для платформы Microsoft .NET Framework.

C# это член семейства с C-подобным синтаксисом. Его синтаксис очень похож на C++ и Java.

Данный язык поддерживает статическую типизацию, полиморфизм, перегрузку операторов (в том числе операторов явного и неявного приведения типа), делегаты, атрибуты, события, свойства, обобщенные типы и методы, итераторы. Так же он поддерживает анонимные функции с поддержкой замыканий, исключения, комментарии в формате XML.

Реализация программы осуществлялась на VisualStudio.

Microsoft Visual Studio — это один из продуктов компании Майкрософт. Она включает в себя интегрированную среду разработки и инструментальные средства. Visual Studio позволяют разрабатывать консольные приложения, приложения с графическим интерфейсом, в том числе с поддержкой технологии Windows Forms. Так же

здесь можно разрабатывать веб-сайты, веб-приложения, веб-службы как в родном, так и в управляемом кодах для всех платформ, поддерживаемых Microsoft Windows, Windows Mobile, Windows CE, .NET Framework, Xbox, Windows Phone .NET Compact Framework и Microsoft Silverlight.

Visual Studio поддерживает технологию IntelliSense, кроме того еще имеется возможность простейшего рефакторинга кода. Так же имеется встроенный отладчик исходного кода и отладчик машинного уровня. Имеются специальные инструменты, которые включают в себя редактор форм для упрощения создания графического интерфейса приложения. Для создания сайтов имеется веб - редактор, дизайнер классов и дизайнер схемы базы данных. В любой момент времени Visual Studio позволяет создавать и подключать сторонние дополнения (плагины). Их подключения расширяют функциональность практически на каждом уровне, включая добавление поддержки систем контроля версий исходного кода (как например, Subversion и Visual SourceSafe), добавление новых наборов инструментов (например, для редактирования и визуального проектирования кода на предметно-ориентированных языках программирования или инструментов для прочих аспектов процесса разработки программного обеспечения (например, клиент Team Explorer для работы с Team Foundation Server).

Модели данных были построены с помощью BPwin. Это мощный и удобный инструмент, который предлагает Logic Works. Так же Logic Works предлагает механизм двунаправленной связи BPwin – Erwin, так как не до конца автоматизирован процесс перехода модели данных на логическую модель.

ERwin состоит из двух уровней модели - логический и физический. На логическом уровне данные представляются безотносительно конкретной СУБД. На данном уровне вся информация представлена в простой и наглядной форме, их может понять неспециалист. На физическом уровне данных отображается системный каталог. Этот каталог зависит от конкретной реализации СУБД. ERwin позволяет проводить процессы прямого и обратного проектирования БД. Это означает, что по модели данных можно сгенерировать схему БД или автоматически создать модель данных на основе информации системного каталога. Кроме того, ERwin позволяет выравнивать модель и содержимое системного каталога после редактирования того, либо другого. ERwin используется совместно с популярными средствами разработки как PowerBuilder, SQLWindows, Visual Basic, Delphi. Это позволяет автоматически генерировать код приложения, который готов к компиляции и выполнению.

CASE-технология предназначена для проектирования автоматизированной системы (АС). В нем имеется набор инструментальных средств, при помощи которой можно моделировать предметную область для наглядного представления, анализировать эту модель на всех этапах

разработки и сопровождения АС и разрабатывать приложения в соответствии с информационными потребностями пользователей.

Большинство существующих CASE-средств основано на методологиях структурного (в основном) или объектно-ориентированного анализа и проектирования, использующих спецификации в виде диаграмм или текстов для описания внешних требований, связей между моделями системы, динамики поведения системы и архитектуры программных средств.

Визуальное моделирование в Rational Rose – это процесс, при котором всю модель можно представить при помощи специально встроенных графиков. Наличие стандарта жизненно необходимо для реализации одного из преимуществ визуального моделирования — коммуникации. Общение между пользователями, разработчиками, аналитиками, тестировщиками, менеджерами и всеми остальными участниками проекта является основной целью графического визуального моделирования.

Data Mining – это технология, сам термин переводится как "добыча" или "раскопка данных". Часто со словом Data Mining встречаются слова "обнаружение знаний в базах данных" (knowledge discovery in databases) и "интеллектуальный анализ данных" – это одно и то же, их можно считать синонимами Data Mining. Возникновение очень больших информационных массивов, которые требуют обработки и вызвали появление данной технологии.

В условиях жесткой ограниченности ресурсов успешное ведение сельского хозяйства зависит от их эффективного использования. Стратегическое планирование – это одно из важных способов повышения эффективности управления хозяйством. Успешность проведения стратегического планирования во многом зависит от квалификации исполнителя и качества используемых результатов аналитических исследований. Любая ошибка в процессе анализа имеющихся данных (особенно методологическая) может привести к получению результатов ненадлежащего качества. Разработка средств автоматизированного контроля результатов, полученных исполнителем, является важной задачей данной предметной области.

Анализ данных – это одна из сложных задач, требующая высоких навыков и квалификации исследователя. Анализируя данные, предполагается получение практически значимых результатов, полученные знания должны как логически верными, так и отвечать поставленным целям исследования. В случае, когда исследователь сам является потребителем результатов анализа данных, проблема получения «не тех» результатов не так остра, как в случае выполнения анализа на заказ. Когда пользователь результатов и человек, получающий результаты (аналитик) разные люди, возникает проблема не понимания, снижение оценки корректности полученных результатов, так как обычный простой потребитель не имеет специализированной квалификации для проведения самостоятельной оценки.

Аналитик, не имеющий опыт, или же недобросовестный аналитик может допустить ошибки как при выборе метода и технологии анализа, при проведении анализа, так и при получении результатов.

На данном этапе разработки системы для осуществления анализа и прогнозирования данных используется пакет STATISTICA.

Система STATISTICA производится фирмой StatSoft Inc., (США), основанной в 1984 г.

Средства анализа STATISTICA Data Miner состоит из пяти основных классов:

- general Slicer/Dicer and Drill-Down Explorer – это класс, при помощи которого все данные разбиваются и проводится углубленный анализ. Данная процедура позволяет разбивать, группировать переменные, вычислять описательные статистики, строить исследовательские графики и т.д;

- general Classifier – специальный пакет классификации. STATISTICA Data Miner включает в себя полный пакет процедур классификации: обобщенные линейные модели, деревья классификации, регрессионные деревья, кластерный анализ и т.д;

- general Modeler/Multivariate Explorer – это специализированные модели, которые могут быть обобщенными линейными, нелинейными и регрессионными;

- general Forecaster – переводится как прогнозирование. Включает в себя модели АРПСС, сезонные модели АРПСС, экспоненциальное сглаживание, спектральный анализ Фурье, сезонная декомпозиция, прогнозирование при помощи нейронных сетей и т.д;

- general Neural Networks Explorer – анализ по типу нейросети. В данном пакете содержится глубокий статистический анализ, который рассматривает данные как нейросети [3].

2 Проектирование и разработка ИИСППР «Агроном»

2.1 Системный анализ объекта исследования

Создание ИИСППР «Агроном» состоит из 4 стадий.

Стадия 1 – стадия технического задания. Сроки: 01.09.13 – 01.11.13.

Данная стадия состоит из следующих этапов:

- предварительная работа – обследование и анализирование предметной области, объекта в целом и выбор модели для разработки;
- анализ требований к интеллектуальной информационной системе, анализ функциональных возможностей системы, требование к интерфейсу;
- построение архитектуры системы.

Стадия 2 – стадия эскизного проекта. Сроки: 01.11.13 – 01.01.14.

Данная стадия состоит из следующих этапов:

- анализ, требование к программному обеспечению;
- проектирование архитектуры программного обеспечения;
- детальное проектирование программирования;
- выбор технологий.

Стадия 3 – стадия технический проект. Сроки: 01.01.14 – 01.04.14.

Данная стадия состоит из следующих этапов:

- кодирование и тестирование ПО;
- интеграция ПО (сборка всех компонентов);
- квалификационное тестирование системы.

Стадия 4 – стадия рабочий проект или же сдача проекта. Сроки 01.04.14 – 20.05.14.

Стадия состоит из этапов: установка и приемка ПО.

Цели создания системы – автоматизация и сокращение времени работы по разработке производственного плана, учету выращиваемой агрокультуры, калькуляции необходимого количества саженцев, участка земли, удобрения; анализу урожайности, подсчету прибыльности агрокультуры, учитывая метеорологические условия и плодородность земельного участка.

Основными целями создания ИИСППР «Сельское хозяйство» являются:

- облегчение создания плана лицу, занимающемуся агрономической деятельностью;
- оптимизация трудового ресурса и времени. Замена ручных, непроизводительных и постоянно повторяющихся процессов для повышения оптимизации ресурсов труда;
- повышение качества принятия управленческих решений за счет оперативности представления, полноты, достоверности и удобства форматов отображения информации;
- содержание базы знаний, которой может воспользоваться лицо, не имеющее никакие навыки о данной отрасли.

Основные критерии достижения целей. Для реализации поставленных целей система должна решать следующие задачи:

- ввод первоначальных данных (количество гектар посаженной продукции) для реализации калькуляции;
- учет метеорологических и агрономических условий;
- построение аналитических отчетов;
- разработка плана.

Для достижения данной цели система должна решать следующие задачи:

- выбор почвы под определенную культуру. Пользователь вводит данные относительно культуры: выбирает тип, вид и сорт культуры, а система, опираясь базу знаний, где содержатся данные относительно анализа почвы, предлагает область или регион, где данная культура точно даст результат. Либо пользователь вводит область, регион, где он проживает и система предлагает список культур, которые могут произрасти;

- предложение необходимого количества удобрения, анализируя состав почвы. Для этого пользователь должен указать количество гектар, тип культуры и регион;

- расчет количества семян или саженцев для посадки в зависимости от площади гектар и типа культуры. Пользователь вводит данные о типе культуры и количестве гектар земли;

- предложение сроков посева и глубины заделки семян, учитывая температуру и относительную влажность почвы. На данный момент информация о температуре и влажности почвы будут вводиться вручную, но в будущем планируется считывание информации с датчиков. То есть на данной территории будут установлены датчики, которые три раза в сутки проверяет влажность и температуру почвы, затем, используя беспроводные технологии, данные передаются в систему. После считывания и анализа данных система предлагает решение;

- выбор оптимального сорта, учитывая погодные и климатические условия. Всем известно, что в каждом регионе преобладает свой климат, который очень сильно влияет как на почву так и на саму культуру. Система будет проводить анализ по погодным условиям: жаркая и сухая погода, нормальная погода, холодная погода, дождливая. То есть системой будет предложен тот сорт, который во всех природных условиях даст наибольший урожай. Для решения данного вопроса будет использована теория игр, где противниками являются погода и пользователь;

- предложение по выгодности: выгодно ли данную почву удобрять под определенную культуру либо выгоднее выбрать другой сорт, который можно посадить, не удобряя почву. То есть имеются две культуры. Допустим реализация первой культуры за килограмм равно 200 тенге, а вторая культура 50 тенге. Но для посадки первой культуры, стоимость которой 200 тенге, почву необходимо постоянно подпитывать и удобрять, то есть будут дополнительные затраты, а для второй культуры, стоимость которой 50 тенге,

удобрения не нужны, состав почвы итак благоприятен для посадки данного типа. Что же в итоге выгоднее посадить? Данная система будет давать ответ на этот вопрос;

- выбор процесса, учитывая температуру и влажность почвы, определять сроки полива, боронования, прополки;
- составление готового документа план.

Так же имеются свои фишки данного проекта. Будет задействована ГИС система, где в проекте будет выводиться карта Казахстана и по областям можно будет посмотреть состав, относительную влажность и температуру почвы, список культур, которые могут произрасти в данной области. На данный момент ГИС система предоставляет снимок поверхности земли, а для пользователя ИИСППР лишь снимок земли недостаточен, предлагается установление датчиков, которые показывают и внутреннее строение почвы, ее особенности. Также будет иметься база знаний, которая постоянно обновляется при помощи интернет, где хранится полная информация о различной культуре, почве, удобрениях, то есть краткий справочник землепользователя.

На данный момент ИИСППР реализована не полностью. Реализован интерфейс программы, база знаний и некоторые вышеперечисленные задачи. Представим такую ситуацию, когда фермер - новичок решил заниматься сельским хозяйством. У него есть участок земли, и он должен определиться, какую культуру и какой сорт будет выращивать, но до этого момента он никогда не занимался земледелием и не представляет, что можно взрастить на данном регионе. Появляются задачи, которые сейчас реализованы в ИИСППР:

- выбор культуры для посадки в зависимости от региона, опираясь на статистические данные (предыдущий опыт), используя технологию Data Mining;
- выбор оптимального сорта, учитывая погодные и климатические условия. Для решения данного вопроса будет использована теория игр, где противниками являются погода и пользователь.

2.2 Разработка UML-диаграмм ИИСППР «Агроном»

2.2.1 Вид с точки зрения поведения

На диаграмме прецедентов (вариантов использования) показано как осуществляется связь между вариантами использования и действующими лицами.

Она отражает требования к системе с точки зрения пользователя, показывает всю системы со стороны пользователя. Итого, варианты использования – это функции и задачи, выполняемые системой, а пользователь – это заинтересованные лица по отношению к создаваемой системе.

Самая главная задача диаграммы вариантов использования – показать один объединенный вид, дающей возможность заказчику, конечному пользователю и разработчику совместно обсуждать и видеть функциональность и поведение системы.

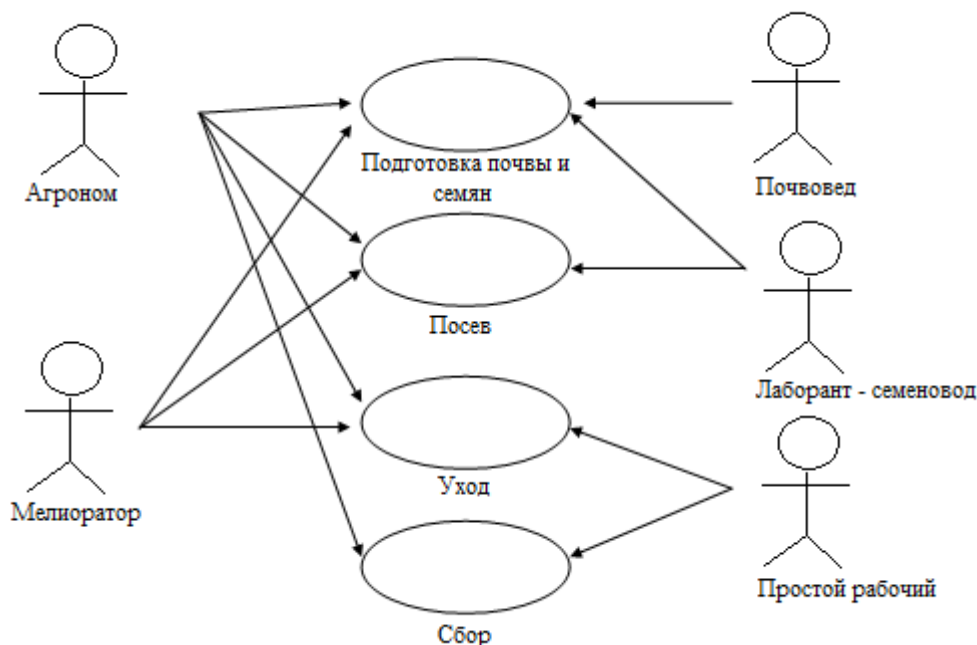


Рисунок 2.1 – Диаграмма прецедентов

На этой диаграмме показаны все действующие лица, которые непременно принимают участие в процессе выращивания культуры. Существует всего 4 основных процесса: подготовительные работы, посев, уход и сбор.

2.2.2 Вид с точки зрения процесса

Диаграммы видов деятельности - это один из пяти самых важных видов диаграмм, которые используются в UML для модели движущихся аспектов поведения системы. Диаграмма видов деятельности – это блок-схема, которая показывает, как идет движение потока, по какому направлению одно действие перетекает на другое.

Диаграммы деятельности используют для построения движущихся аспектов поведения системы. Они применяются, чтобы показать движение последовательных, а иногда и параллельных процессов вычислительной системы.

Основными элементами диаграмм видов деятельности являются обозначения состояния («начало», «конец»), действия (овал) и момента синхронизации действий (линейка синхронизации, на которой сходятся или разветвляются несколько стрелок).

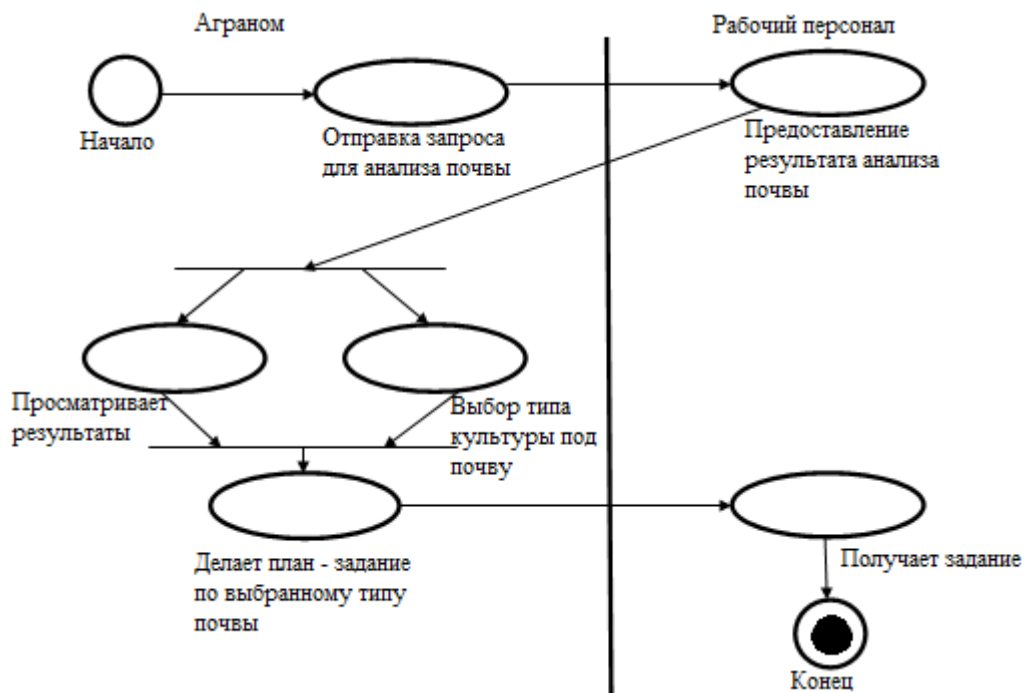


Рисунок 2.2 - Диаграмма видов деятельности для прецедента «Планирование»

Чтобы представить физические сущности в языке UML применяется специальный термин - компонент (component). Компонент объединяет некоторый набор интерфейсов и служит для общего обозначения элементов физического представления модели. Для графического представления компонента используется прямоугольник со вставленными слева двумя более мелкими прямоугольниками. В прямоугольнике записывается имя компонента и, если есть, некоторая дополнительная информация. Изображение этого символа может незначительно варьироваться в зависимости от характера ассоциируемой с компонентом информации.

В метамодели языка UML компонент является дочерним потомком классификатора. Он предоставляет организацию в рамках физического пакета ассоциированным с ним элементам модели. Как классификатор, компонент может иметь также свои собственные свойства, такие как атрибуты и операции.

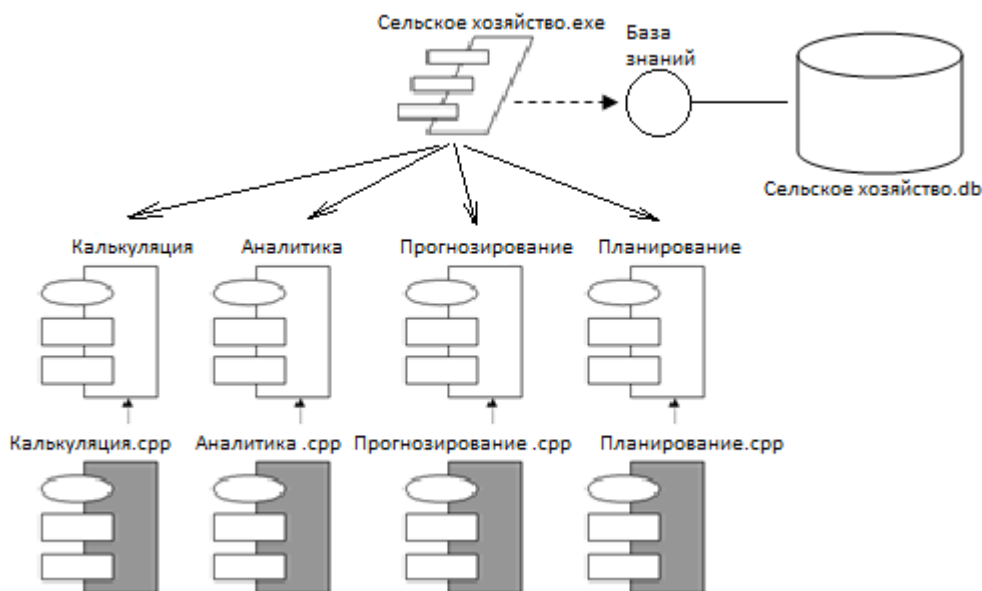


Рисунок 2.3 – Диаграмма компонентов

Основной компонент, который практически управляет остальными это Application(приложение).

Каждый компонент состоит из двух частей:

- спецификация – это заголовочный файл для сведений о прототипах функций для класса (не закрашенная часть);
- тело пакета – часть, которая содержит код операции класса (закрашенная часть).

Диаграмма компонентов показывает, как выглядит модель на физическом уровне.

2.2.3 Вид с точки зрения развертывания

Физическое представление программной системы не может быть полным, если отсутствует информация о том, на какой платформе и на каких вычислительных средствах она реализована. Для представления всей конфигурации, настройки и топологии распределенной программной системы в UML предназначены диаграммы размещения.

Диаграммы развертывания используются для моделирования статического вида системы с точки зрения развернутой системы. В основном под этим понимается моделирование топологии аппаратных средств, на которых выполняется система. По существу, диаграммы развертывания – это классовые диаграммы на всех узлах системы.

Диаграмма размещения показывает всю структуру в период его исполнения. При этом представляются только компоненты-экземпляры программы, являющиеся исполняемыми файлами или динамическими библиотеками. Те компоненты, которые не используются на этапе

исполнения, на диаграмме развертывания не показываются. Так, компоненты с исходными текстами программ могут присутствовать только на диаграмме компонентов. На диаграмме размещения они не указываются.

Диаграмма размещения отражает физические взаимосвязи между программными и аппаратными компонентами системы. Она является хорошим средством для того, чтобы показать маршруты перемещения объектов и компонентов в распределенной системе. Каждый узел на диаграмме размещения представляет собой некоторый тип вычислительного устройства – в большинстве случаев, часть аппаратуры. Эта аппаратура может быть простым устройством или датчиком, а может быть и мэйнфреймом.



Рисунок 2.4 – Диаграмма развертывания

Диаграмма размещения содержит графические изображения отделов и связей между ними. В отличие от диаграмм логического представления, диаграмма размещения является единой для системы в целом, поскольку должна всецело отражать особенности ее реализации. Разработка диаграммы размещения, как правило, является последним этапом спецификации модели программной системы.

2.3 Разработка базы данных ИИСППР «Агроном»

Методология IDEF1X (IDEF1 Extended) – язык для логического моделирования данных, основанных на концепции «сущность-связь». Является дополнением для стандарта IDEF1.

Диаграмма «сущность-связь» ERD (Entity-Relationship Diagram) нужна для разработки общей модели баз данных и дает возможность на стандартный способ распределения данных и отношений между ними.

Теоретической базой построения информационной модели является теория баз данных типа «сущность-связь».

Согласно стандарту, основными составляющими модели IDEF1X являются:

- люди, предметы, явления, о которых хранится информация (сущности);
- связи между этими элементами (отношения);
- характеристики этих элементов (атрибуты).

Каждая сущность в диаграмме IDEF1x описывается графическим объектом, который выглядит как прямоугольник. Каждый прямоугольник, который показывает собой сущность, отделяются горизонтальной линией на две части, в которой расположены ключевые и не ключевые поля.

Часть, которая расположена выше, называется ключевой областью, а нижняя часть областью данных.

Ключевая область содержит первичный ключ, который определяет сущность.

Первичный ключ - это набор атрибутов, который указывает на индивидуальности выбранной сущности. Все, связанные с сущностью атрибуты первичного ключа располагаются над линией в ключевой области. Неключевой атрибут - это атрибут, который не был выбран ключевым, то есть второстепенный и уникальность его не так значима как ключевого атрибута. Неключевые атрибуты располагаются под чертой, в области данных.

2.3.1 Связи между сущностями

Связи между сущностями бывают трех типов:

- один-к-одному;
- один-ко-многим;
- многие-ко-многим.

В ERWin используются связи один-ко-многим и многие-ко-многим.

В данном случае вид культуры использует отношение с типом, сортом, почвой и площадью. Почва включает в себя регион и удобрение.

Вид культуры – ассоциативная сущность;

Тип и сорт культуры – характеристическая сущность. Полученная модель показана на рисунке 2.5.

2.3.2 Выделение сущностей и атрибутов

После изучения предметной области, были выделены основные сущности, которые представлены в таблице 2.1 и которые являются составной частью системы.

Таблица 2.1 – Сущности и их описание

Сущность	Атрибуты	Назначение
Вид культуры	Наименование культуры	Содержит в БД список все видов культур. Например: кукуруза, яблоня, картофель и т.д.
Тип культуры	Наименование типа	Содержит в БД список все типов культур. Например: зерновые, пасленовые, овощи, фрукты и т.д.
Сорт культуры	Наименование сорта	Содержит в БД список все сортов культур. Например: белороза, апорт и т.д.
Почва	Наименование почвы Состав почвы	Содержит в БД полный состав различного вида почвы. Для определения и выявления подходящего типа культуры.
Площадь	Количество (га)	Содержит информацию о количестве посаженной продукции. Для расчета прибыли.
Регион	Название	Содержит название региона, который выбирается в зависимости от почвы.
Удобрение	Название Состав Назначение	Содержит список всех используемых удобрений. Необходимо для подсчета нужного количества удобрений.

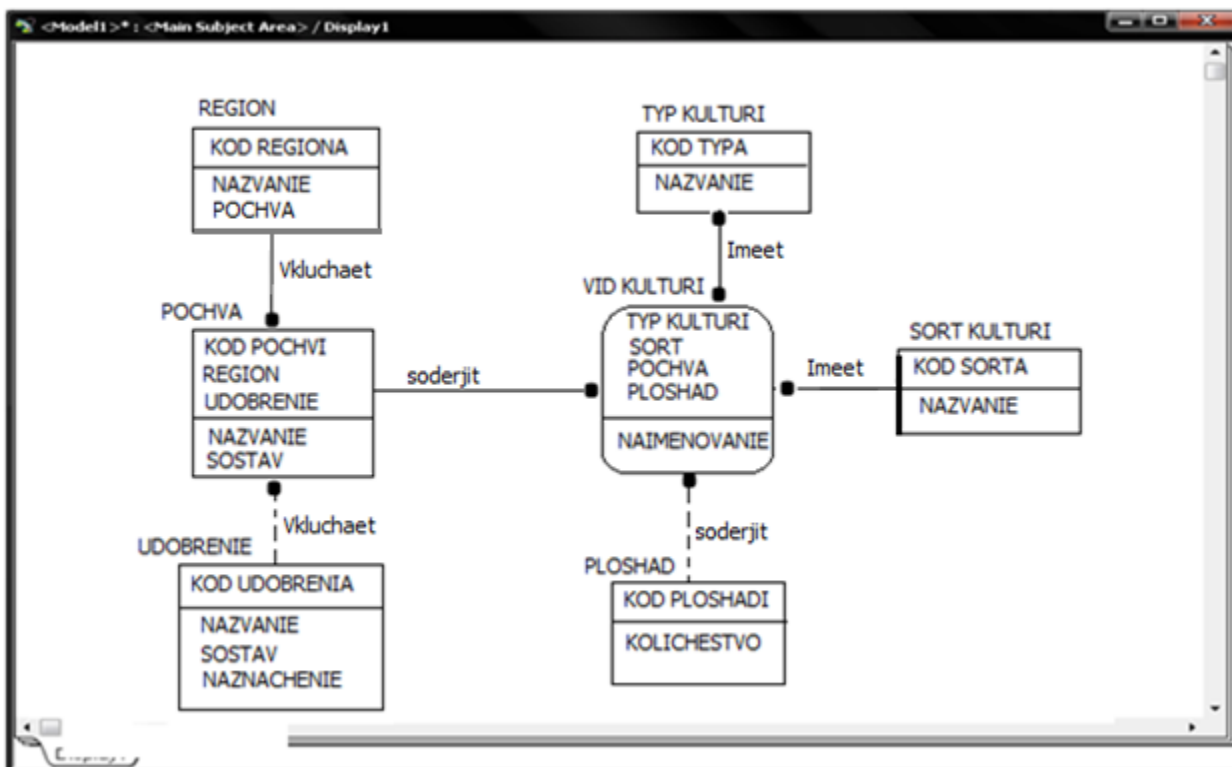


Рисунок 2.5 – Логический уровень

2.3.3 Переход на физический уровень

Создание интеллектуальной информационной системы поддержки принятия решений – очень сложный развивающийся процесс, который содержит фазы разработки двух моделей, логической и физической модели данных.

На логическом уровне всем объектам модели, сущностям и атрибутам даются имена, которые понятны большому кругу специалистов, в том числе и на кириллице.

На физическом уровне все объекты базы данных (таблицы, колонки и т.д.) должны быть названы так, как этого требует система управления базой данных (СУБД). Физическая модель на прямую зависит от конкретной СУБД, поэтому может быть одна логическая модель, которой могут соответствовать несколько физических моделей.

Такое описание в два уровня позволяет, с одной стороны, достаточно свободно обсуждать структуру данных со знатоками предметной области, а с другой стороны – лучше приспособить модель к требованиям хранения и обработки данных.

На основе разработанной физической модели можно сгенерировать системный каталог СУБД или соответствующий SQL-скрипт. Этот процесс преобразования называется прямым проектированием (Forward Engineering). Таким образом, достигается масштабируемость и изменяемость проекта. То есть, создав одну логическую модель данных, можно собрать физические

модели под любую поддерживаемую СУБД. С другой стороны, можно по содержимому системного каталога или SQL-скрипту воссоздать физическую и логическую модель данных (Reverse Engineering). После получения логической модели данных инструментальное средство проектирования способно объединить физическую модель для другой СУБД, а затем сгенерировать ее системный каталог. Именно по такому типу последовательности может быть решена задача переноса структуры данных с одного сервера на другой.

Физическая модель данных зависит от конкретной СУБД, фактически являясь отображением системного каталога. Именно в физической модели содержится информация обо всех объектах БД, начиная с атрибутов и так далее. Поскольку стандартов на объекты БД не существует (например, нет стандарта на типы данных), физическая модель зависит от конкретной реализации СУБД. Одной и той же логической модели могут соответствовать множество различных друг от друга физических моделей. Если в логической модели не имеет значения, какой конкретно тип данных имеет атрибут, то в физической модели важно описать всю информацию о конкретных физических объектах - таблицах, колонках, индексах, процедурах и т.д. Разделение модели данных на логические и физические позволяет решить несколько важных задач.

Связь многие-ко-многим возможна только на уровне логической модели данных, поэтому при переходе к физическому уровню ERWin автоматически преобразует связь многие-ко-многим, добавляя новую, ассоциативную сущность и устанавливая две новые связи один-ко-многим от старых к новой сущности.

При создании проекта были выявлены следующие ограничения:

- в базе описывается множество видов культуры;
- количество площади измеряется целым числом;
- каждый вид культуры может иметь несколько сортов, а тип культуры
- может иметь несколько видов культуры;
- в одном регионе описывается вид почвы;
- для одной почвы применимы несколько видов удобрения.

Физический уровень базы данных системы представлен на рисунке 2.6.

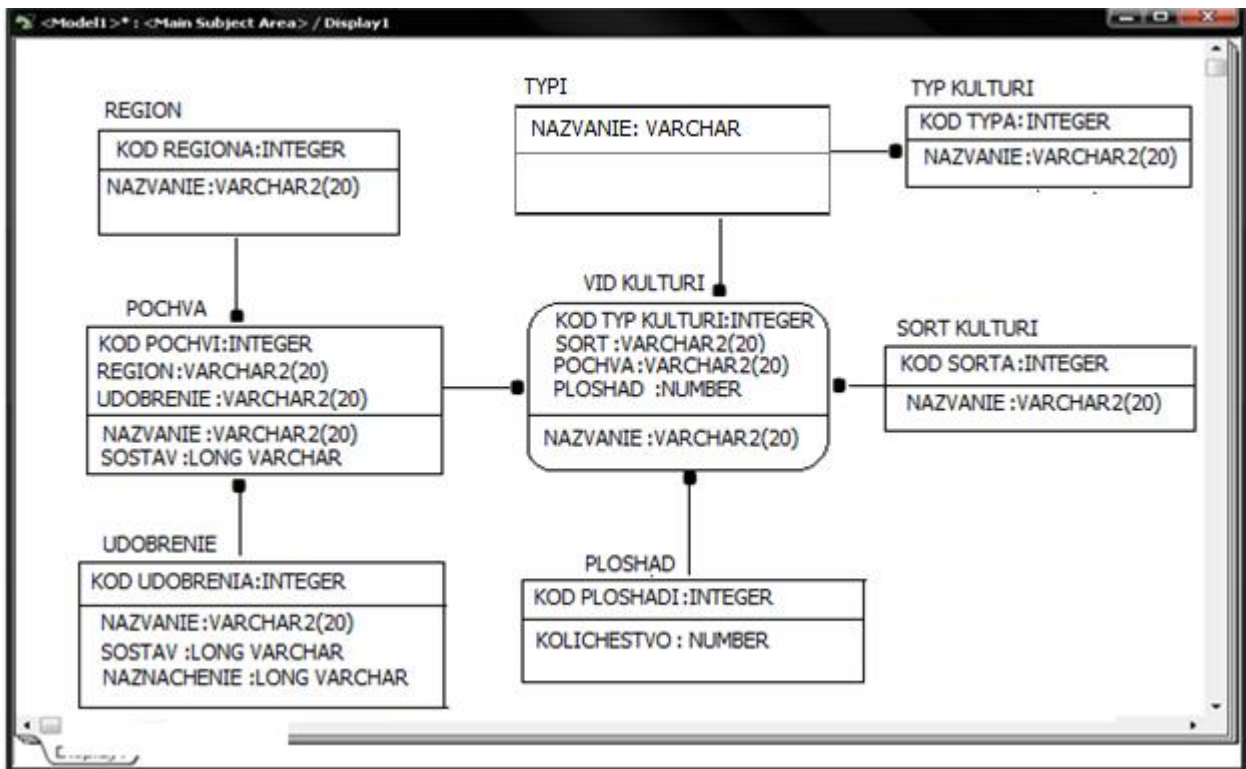


Рисунок 2.6 – Физическая модель данных

3 Разработка программного обеспечения ИИСППР «Агроном»

3.1 Назначение и условия выполнения программы

Назначение системы – разработка производственного плана, учет выращиваемой агрокультуры, калькуляция необходимого количества саженцев, участка земли, удобрения; анализ урожайности, подсчет прибыльности агрокультуры, учитывая погодные условия и плодородность земельного участка.

Цели создания системы – автоматизация и сокращение времени работы по разработке производственного плана, учету выращиваемой агрокультуры, калькуляции необходимого количества саженцев, участка земли, удобрения; анализу урожайности, подсчету прибыльности агрокультуры, учитывая метеорологические условия и плодородность земельного участка.

ИСППР «Сельское хозяйство» предназначена для комплексного информационно-аналитического обеспечения процессов выращивания агрокультуры в части исполнения следующих процессов:

- калькуляция саженцев и необходимого количества удобрения по виду культуры;
- аналитический вывод прибыльности агрокультуры с учетом метеорологических и агрономических условий;
- сравнительный анализ и построение диаграммы по убыткам и прибыли;
- разработка технического плана на предыдущий год.

Основными целями создания ИСПР «Сельское хозяйство» являются:

- облегчение создания плана лицу, занимающемуся агрономической деятельностью;
- повышение эффективности исполнения процессов, перечисленных выше, путем сокращения непроизводительных и дублирующих операций, операций, выполняемых "вручную", оптимизации трудового ресурса и времени;
- повышение качества принятия управленческих решений за счет оперативности представления, полноты, достоверности и удобства форматов отображения информации;
- содержание базы знаний, которой может воспользоваться лицо, не имеющее никаких навыков о данной отрасли.

Для реализации поставленных целей система должна решать следующие задачи:

- ввод первоначальных данных (количество гектар посаженной продукции) для реализации калькуляции;
- учет метеорологических и агрономических условий;
- построение аналитических отчетов;
- разработка плана.

Процессы по управлению ИИСППР «Агроном» (п. 2.1), а также контроль эффективности выполнения указанных процессов. Данные процессы осуществляются следующими специалистами:

- агроном - специалист сельского хозяйства, обладающий всесторонними знаниями в области агрономии;
- владелец крестьянского хозяйства;
- рабочий персонал в области сельского хозяйства.

В настоящий момент деятельность планирования и составления анализа в Казахстане в отрасли сельское хозяйство не автоматизирована. Вся деятельность осуществляется вручную, замер ведется примерный, а не точный.

Телекоммуникационная инфраструктура отсутствует. Персональный компьютер имеет агроном или же владелец крестьянского хозяйства.

Существующее нормативно-правовое обеспечение составляют федеральные и областные нормативные правовые акты:

- конституция РК;
- гражданский кодекс РК.

Требования к системе в целом:

- требования к структуре и функционированию системы;
- требования к персоналу системы;
- показатели назначения;
- требования к надежности; безопасности; эргономике и технической эстетике; эксплуатации, техническому обслуживанию, ремонту и хранению компонентов системы; защите информации; сохранности информации при авариях; защите от влияния внешних воздействий;
- дополнительные требования;
- требования к функциям (задачам), выполняемым системой; перечень функций, задач или их комплексов, подлежащих автоматизации (по каждой подсистеме); очередность ввода в эксплуатацию, временной регламент реализации и требования к качеству реализации каждой функции, задачи (или комплекса задач), к форме представления выходной информации, перечень и критерии отказов для каждой функции, по которой задаются требования по надежности;
- требования к видам обеспечения в зависимости от вида системы приводят требования к математическому, информационному, лингвистическому, программному, техническому, метрологическому, организационному, методическому и другим видам обеспечения системы.

В ИИСППР «Сельское хозяйство» должны решаться следующие задачи:

- занесение информации в БД;
- учет метеорологических и агрономических условий;
- формирование сравнительных диаграмм;
- разработка плана на предыдущий год.

Занесение информации в БД предназначена для проведения калькуляции. Вводятся основные данные (количество гектар крестьянского хозяйства и вид выращиваемой агрокультуры). На основании этих данных подсчитывается количество необходимых саженцев или семян и необходимое количество удобрения.

Учет метеорологических условий предназначена для получения выводов об урожайности на предыдущий год. Метеорологические данные будут учитываться из глобальной сети интернет, а агрономические данные будут содержаться в базе знаний.

Формирование сравнительных диаграмм предназначена для анализа и получения данных, которые используются при разработке плана на предыдущий год. Сравнительные диаграммы строятся на основании данных по прибыли и расходу за предыдущий год для выявления наиболее оптимального продукта, который более прибыльный и требует меньше расхода.

Разработка плана на предыдущий год предназначена для создания документа, на основании которого будут вестись посевные работы.

ИСПР «Сельское хозяйство» должна обеспечивать доступ в первую очередь агроному, руководителям различных крестьянских хозяйств для сравнения урожайности по различным годам и получения плана.

Входящие в состав ИСПР «Сельское хозяйство» задачи в процессе функционирования должны обмениваться информацией на основе открытых форматов обмена данными, используя для этого входящие в их состав модули информационного взаимодействия.

Форматы данных будут разработаны и утверждены на этапе технического проектирования.

В состав передаваемых данных входят:

- метеорологический отчет;
- почвоведческий отчет;
- бухгалтерский отчет.

Смежные системы отсутствуют.

Для ИСПР «Сельское хозяйство» определены следующие режимы функционирования:

- нормальный режим функционирования;
- аварийный режим функционирования.

Основным режимом функционирования ИИСПР является нормальный режим.

В нормальном режиме функционирования системы:

– программное обеспечение и технические средства обеспечивают возможность круглосуточного функционирования, с перерывами на обслуживание;

- исправно работает оборудование;
- исправно функционирует программное обеспечение системы.

Для обеспечения нормального режима функционирования системы необходимо выполнять требования и выдерживать условия эксплуатации программного обеспечения и технических средств системы.

Аварийный режим функционирования системы характеризуется отказом одного или нескольких компонент программного и (или) технического обеспечения.

Аварийный режим может возникнуть в ходе отсутствия или неисправности интернета, так как метеорологические данные будут отсутствовать.

В случае перехода системы в аварийный режим необходимо:

- завершить работу всех приложений, с сохранением данных;
- выполнить резервное копирование БД.

После этого необходимо выполнить комплекс мероприятий по устранению причины перехода системы в аварийный режим.

ИИСППР «Сельское хозяйство» должна реализовывать возможность дальнейшей модернизации как программного обеспечения.

Также необходимо предусмотреть возможность увеличения производительности системы путем её масштабирования и возможность увеличения базы данных.

Для эксплуатации ИИСППР «Сельское хозяйство» определены следующие роли:

- администратор баз данных;
- пользователь.

Основными обязанностями администратора баз данных являются:

– установка, модернизация, настройка параметров программного обеспечения СУБД;

– оптимизация прикладных баз данных по времени отклика, скорости доступа к данным;

– разработка, управление и реализация эффективной политики доступа к информации, хранящейся в прикладных базах данных;

– ввод дополнительных данных в БД.

Администратор баз данных должен обладать высоким уровнем квалификации и практическим опытом выполнения работ по установке, настройке и администрированию используемых в ИИСППР СУБД.

Пользователи системы должны иметь опыт работы с персональным компьютером на базе операционных систем Microsoft Windows на уровне квалифицированного пользователя и свободно осуществлять базовые операции в стандартных Windows.

Рекомендуемая численность для эксплуатации ИУС «Деятельность склада»:

- администратор - 1 штатная единица;
- пользователь - число штатных единиц определяется структурой предприятия.

ИИСППР «Агроном» должны обеспечивать возможность исторического хранения данных с глубиной не менее 10 лет.

Система должна обеспечивать:

- навигационные операции по экранным формам системы - не более 10 сек;
- формирование диаграмм и документации плана - не более 40 сек.

Время формирования аналитических отчетов определяется их сложностью и может занимать продолжительное время.

Система должна давать возможность масштабировать по производительности и объему обрабатываемой информации без модификации ее программного обеспечения путем усовершенствования всех используемых технических средств.

Интеллектуальная система должна хранить работоспособность и обеспечивать полное восстановление своих функций при возникновении следующих аварийных ситуаций:

- при выходе из строя системы электроснабжения аппаратной части, которое приведет к перезагрузке ОС, все операции должны быть восстановлены после перезапуска ОС и запуска исполняемого файла системы;
- при ошибках в работе аппаратных средств (кроме носителей данных и программ) восстановление функции системы возлагается на ОС;
- при выявлении ошибок, которые связаны с программным обеспечением (ОС и драйверы устройств), восстановление работоспособности возлагается на ОС;
- чтобы защитить данные в БД необходимо провести мероприятия по защите, установить пароль, ключ доступа и т.д;
- для того, чтобы программное обеспечение не переходило в аварийный режим, необходимо вести слежение за интернет ресурсами.

Чтобы защитить аппаратуру от резких бросков напряжения и коммутационных помех должны применяться сетевые фильтры.

Все внешние и наружные элементы технических средств системы, которые находятся под напряжением, должны быть изолированы от случайного прикосновения. Все технические средства должны иметь зануление или защитное заземление в соответствии с ГОСТ 12.1.030-81 и ПУЭ.

Система электропитания должна обеспечивать защитное отключение при перегрузках и коротких замыканиях в цепях нагрузки, а также аварийное ручное отключение.

Общие требования пожарной безопасности должны соответствовать нормам на бытовое электрооборудование. В случае возгорания должна быть защита от выделения ядовитых газов и дымов. После снятия электропитания должно быть допустимо применение любых средств пожаротушения.

Факторы, оказывающие вредные воздействия на здоровье со стороны всех элементов системы (в том числе инфракрасное, ультрафиолетовое, рентгеновское и электромагнитное излучения, вибрация, шум,

электростатические поля, ультразвук строчной частоты и т.д.), не должны превышать действующих норм.

Взаимодействие пользователей с прикладным программным обеспечением, входящим в состав системы должно осуществляться посредством визуального графического интерфейса (GUI). Интерфейс системы должен быть понятным и удобным, не должен быть перегружен графическими элементами и должен обеспечивать быстрое отображение экранных форм. Навигационные элементы должны быть выполнены в удобной для пользователя форме. Средства редактирования информации должны удовлетворять принятым соглашениям в части использования функциональных клавиш, режимов работы, поиска, использования оконной системы. Ввод-вывод данных системы, прием управляющих команд и отображение результатов их исполнения должны выполняться в интерактивном режиме. Интерфейс должен соответствовать современным эргономическим требованиям и обеспечивать удобный доступ к основным функциям и операциям системы.

Интерфейс должен быть рассчитан на преимущественное использование манипулятора типа "мышь", то есть управление системой должно осуществляться с помощью набора экранных меню, кнопок, значков и т. п. элементов. Клавиатурный режим ввода должен использоваться главным образом при заполнении и/или редактировании текстовых и числовых полей экранных форм. Система должна иметь всплывающие подсказки для неопытного пользователя и возможность отключения этих подсказок для опытного пользователя.

Система должна обеспечивать корректную обработку аварийных ситуаций, вызванных неверными действиями пользователей, неверным форматом или недопустимыми значениями входных данных. В указанных случаях система должна выдавать пользователю соответствующие сообщения, после чего возвращаться в рабочее состояние, предшествовавшее неверной (недопустимой) команде или некорректному вводу данных.

Система должна соответствовать требованиям эргономики и профессиональной медицины при условии комплектования высококачественным оборудованием (ПЭВМ, монитор и прочее оборудование), имеющим необходимые сертификаты соответствия и безопасности ГОСТов РК.

Техническая и физическая защита аппаратных компонентов системы, носителей данных, бесперебойное энергоснабжение, резервирование ресурсов, текущее обслуживание реализуется техническими и организационными средствами, предусмотренными в ИТ инфраструктуре Заказчика.

Чтобы обеспечить нормальную работу разрабатываемой системы необходимо бесперебойное электрическое питание. При эксплуатации система должна быть обеспечена соответствующая стандартам хранения носителей и эксплуатации ПК температура и влажность воздуха.

Периодическое техническое обслуживание используемого ПК должно проводиться в соответствии с требованиями технической документации изготовителей, но не реже одного раза в год.

Периодическое техническое обслуживание и тестирование ПК должны включать в себя обслуживание и тестирование ПК, кабельной системы, устройств бесперебойного питания.

В процессе проведения периодического технического обслуживания должны проводиться внешний и внутренний осмотр и чистка ПК, проверка контактных соединений, проверка параметров настроек работоспособности ПК.

Размещение ПК должно исключать возможность бесконтрольного проникновения в них посторонних лиц и обеспечивать сохранность находящихся в этих помещениях конфиденциальных документов и технических средств.

Размещение ПК должно соответствовать требованиям техники безопасности, санитарным нормам и требованиям пожарной безопасности. Все пользователи системы должны соблюдать правила эксплуатации электронной вычислительной техники.

Квалификация персонала и его подготовка должны соответствовать технической документации.

ИИСППР должна обеспечивать защиту от несанкционированного доступа (НСД).

Компоненты подсистемы защиты от НСД должны обеспечивать идентификацию пользователя.

Защищённая часть системы должна использовать "слепые" пароли (при наборе пароля его символы не показываются на экране либо заменяются одним типом символов; количество символов не соответствует длине пароля).

Программное обеспечение ИИСППР «Сельское хозяйство» должно восстанавливать свое функционирование при корректном перезапуске аппаратных средств. Должна быть предусмотрена возможность организации резервного копирования данных СУБД.

Помещение должно соответствовать санитарно-гигиеническим нормам.

Установка системы в целом, как и установка отдельных частей системы не должна предъявлять дополнительных требований к покупке лицензий на программное обеспечение сторонних производителей.

Экранные формы должны проектироваться с учетом требований унификации:

- все экранные формы пользовательского интерфейса должны быть выполнены в едином графическом дизайне, с одинаковым расположением основных элементов управления и навигации;

- для обозначения сходных операций должны использоваться сходные графические значки, кнопки и другие управляющие (навигационные) элементы.

Подсистема хранения данных должна осуществлять хранение оперативных данных системы, данных для формирования сравнительных диаграмм и для разработки плана на предыдущий год.

База данных должна быть защищенной паролем.

Подсистема должна обеспечивать периодическое резервное копирование.

Подсистема должна решать задачу обеспечения информационными данными, которыми обмениваются отдельные компоненты системы между собой. В число функций подсистемы должны быть включены функции подсчета количества саженцев и семян по площади гектара, подсчет количества определенного типа удобрения требуемого посадочной культурой, учитывая вид и сорт самой культуры.

Подсистема должна предоставлять пользователю удобные инструменты для реализации данной задачи.

Функция калькуляция данных должна обладать следующей основной функциональностью:

- данные о виде культуры, сорт;
- база знаний об удобрениях;
- информация о площади крестьянского хозяйства;
- данные о видах удобрения.

Подсистема аналитический вывод прибыльности агрокультуры с учетом метеорологических и агрономических условий.

Данная подсистема должна выводить результаты прибыльности, урожайности агрокультуры с учетом метеорологических условий, то есть погодные данные, которые непременно будут поступать с интернета и агрономических условий, которая содержится в базе знаний. Расчеты будут проводиться на основании теории игр и с периодичностью месяц. Фишкой данной подсистемы является картография, то есть визуализация данных по областям РК. Данная подсистема требует соединения с интернет. При переходе в аварийный режим, первым делом необходимо проверить данный модуль.

Подсистема сравнительный анализ и построение диаграммы по убыткам и прибыли.

Данная подсистема поможет пользователю визуально определить, какой вид культуры прибыльнее выращивать при этом вложив наименьшие денежные ресурсы. Для реализации данной подсистемы необходимо:

- бухгалтерские отчеты предыдущих лет;
- разработка технического плана на предыдущий год.

Подсистема подготавливает документ в виде плана. Данный документ должен быть оформлен по всем стандартам и должен иметь выход либо на текстовый документ, либо уже в стандартной форме отправляться на печать.

Математические методы и алгоритмы, используемые для шифрования/дешифрования данных, а также программное обеспечение, реализующее их. Основной технологией является Data mining.

Данные должны содержаться в многомерном массиве, что лучше всего организует технология OLAP [6].

Состав, структура и способы организации данных в системе должны быть определены на этапе технического проектирования.

Хранение данных должно осуществляться на основе современных реляционных или СУБД. Для обеспечения целостности данных должны использоваться встроенные механизмы СУБД.

Технические средства, обеспечивающие хранение информации, должны использовать современные технологии, позволяющие обеспечить повышенную надежность хранения данных и оперативную замену оборудования (зеркалирование).

Все прикладное программное обеспечение системы для организации взаимодействия с пользователем должно использовать русский язык.

Требования к программному обеспечению системы

Базовой программной платформой должна являться операционная система MS Windows.

Техническое обеспечение системы должно максимально и наиболее эффективным образом использовать существующий ПК у персонала.

Метрологическое обеспечение измерений - это установление и применение научных и организационных основ, технических средств, правил и норм, которые необходимы для достижения точности измерений и единой записи. В данном программном обеспечении к основным метрологическим измерениям относятся:

- площадь замеряется в гектарах (га);
- количество саженцев в единице штук (шт), а семян в килограммах (Кгр);
- жидкие удобрения замеряются в кубических литрах (л);
- температура замеряется в градусах Цельсия (°C);
- денежные средства рассчитываются в тенге (тг);

Организационное обеспечение системы должно быть достаточным для эффективного выполнения персоналом возложенных на него обязанностей. Заказчиком должны быть определены должностные лица, ответственные за:

- пользование основными пользовательскими функциями;
- администрирование ИИСППР.

Прежде чем приступить к работе с системой, необходимо пройти обучение и системой может пользоваться сотрудник, который имеет навыки на персональном компьютере.

Методическое обеспечение системы состоит из законодательных актов, стандартов, нормативов, инструкций по работе с системой.

Виды, состав, объем, и методы испытаний ИСПР «Сельское хозяйство» должны быть изложены в программе и методике испытаний.

Сдача-приёмка работ производится поэтапно, в соответствии с рабочей программой и календарным планом.

Сдача-приемка осуществляется комиссией, в состав которой входят представители Заказчика и Исполнителя.

Все создаваемые в рамках настоящей работы программные изделия (за исключением покупных) передаются Заказчику.

Статус приемочной комиссии определяется Заказчиком до проведения испытаний.

В ходе выполнения проекта на объекте автоматизации требуется выполнить работы по подготовке к вводу системы в действие. При подготовке к вводу в эксплуатацию ИСПР «Сельское хозяйство» Заказчик должен обеспечить выполнение следующих работ:

- определить ответственных должностных лиц, ответственных за внедрение и проведение опытной эксплуатации ИСПР «Сельское хозяйство»;
- обеспечить присутствие пользователей на обучении работе с системой, проводимом Исполнителем;
- совместно с Исполнителем подготовить план развертывания системы на технических средствах Заказчика;
- провести опытную эксплуатацию ИСПР «Сельское хозяйство».

Данный проект сопровождается ТЗ и документацией на технический проект.

В техническом задании расписываются:

- основные цели, задачи, сроки и этапы разработки;
- список основных функций и требований;
- список функций интерфейса.

Документация на технический проект является инструкцией по использованию данного ПО. В данной документации будут расписываться:

- условия работы ПО;
- установка ПО;
- использование ПО, попунктное описание основных функций и функций интерфейса;
- устранение проблем при переходе в аварийный режим.

Для системы на различных стадиях создания должны быть выпущены следующие документы из числа предусмотренных в ГОСТ 34.201-«Информационная технология. Комплекс стандартов на автоматизированные системы».

Документы и информационные материалы (технико-экономическое обоснование, отчеты о законченных научно-исследовательских работах, информационные материалы на отечественные, зарубежные системы-аналоги и др.), на основании которых разрабатывалось ТЗ и которые должны быть использованы при создании системы.

Технико – экономическое обоснование. Данный документ содержит, финансовое описание системы, в котором содержится перечень используемых ресурсов и их ценовое описание. Стоимость системы, подсчет рентабельности. Минимизация используемых ресурсов для получения наиболее максимальной прибыли.

Данная система должна разрабатываться на основании ТК 34 по стандартизации Информационные технологии. Номер приказа и дата утверждения: от 27.07.01 г. № 274.

3.2 Разработка интерфейса ИИСППР «Агроном»

Интерфейс любой системы является одной из очень важной составляющей. Он ориентирован, прежде всего, на конечного пользователя.

От того, насколько хорошо спроектирован интерфейс зависят такие факторы как:

- скорость освоения (обучения пользователей) системы;
- затраты на внедрение (обучится сам или специалисты помогут);
- последующая успешная работа (интерфейс должен быть понятен всегда, с ним всегда найдется возможность поработать), т.е. снижение риска возникновения ошибок пользователей от работы с системой.

Сейчас ясно прослеживается тенденция отделения разработки пользовательского интерфейса от разработки остального приложения. Это связано с затратой большого количества времени именно на построение логики и формирование самого интерфейса.

Программное обеспечение для разработки пользовательского интерфейса разделяется на две основные группы - инструментарий для разработки пользовательского интерфейса (toolkits) и высокоуровневые средства разработки интерфейса (higher-level development tools).

Инструментарий для разработки пользовательского интерфейса, как правило, включает в себя библиотеку примитивов компонентов интерфейса (меню, кнопки, полосы прокрутки и др.) и предназначен для использования программистами.

Высокоуровневые средства разработки интерфейса могут быть использованы непрограммистами и снабжены языком, который позволяет специфицировать функции ввода-вывода, а также определять, используя технику непосредственного манипулирования, интерфейсные элементы.

Пользовательский интерфейс представляет несколько элементов управления. Каждый элемент может представлять следующую информацию:

- текстовую (текстовые поля ввода, метки, подписи);
- числовую (поля числового ввода, различные индикаторы прогресса, ползунки);
- графическую (изображения, индикаторы);
- звуковую (звуковые сообщения, команды);
- бинарную (флажки, индикаторы, радиокнопки) и другие.

Реагировать элементы управления могут на действия пользователя, произведенных с помощью средств, таких как: мышь, клавиатура, сенсорный экран, голос, другие.

Одной из основных характеристик элемента системы является его состояние. И в каждом состоянии элементы могут отображаться и реагировать

по-разному. Это необходимо, чтобы пользователь не мог привести систему в запрещенное состояние.

Полную логику пользовательского интерфейса интеллектуальной системы поддержки принятия решений представляют в объектно-ориентированном виде. Объект – это самостоятельная форма, которая обладает своими специализированными свойствами и методами. Например, форма – регистрации. Свойства – поля, данные. Методы – кнопки, или действия, которые можно совершить.

Экранные формы должны проектироваться с учетом требований унификации:

- все экранные формы пользовательского интерфейса должны быть выполнены в едином графическом дизайне, с одинаковым расположением основных элементов управления и навигации;

- для обозначения сходных операций должны использоваться сходные графические значки, кнопки и другие управляющие (навигационные) элементы.

3.2.1 Проектирование пользовательского интерфейса

Интеллектуальная информационная система поддержки и принятия решения ИИСППР «Агроном» содержит следующие окна:

- окно входа и регистрации;
- основная форма;
- окно «Калькуляция»;
- окно «Аналитика»;
- окно «Прогнозирование»;
- окно «План»;
- окно «База знаний».



Рисунок 3.1 – Главное окно

Так как в системе два типа доступа, необходимо окно входа или регистрации. В этом окне содержится поле для ввода логина и пароля. Логин всегда остается стандартным, это «Админ» и «Пользователь». В поле пароль заказчик установит пароль на каждую категорию по – своему усмотрению. Рисунок 3.2.

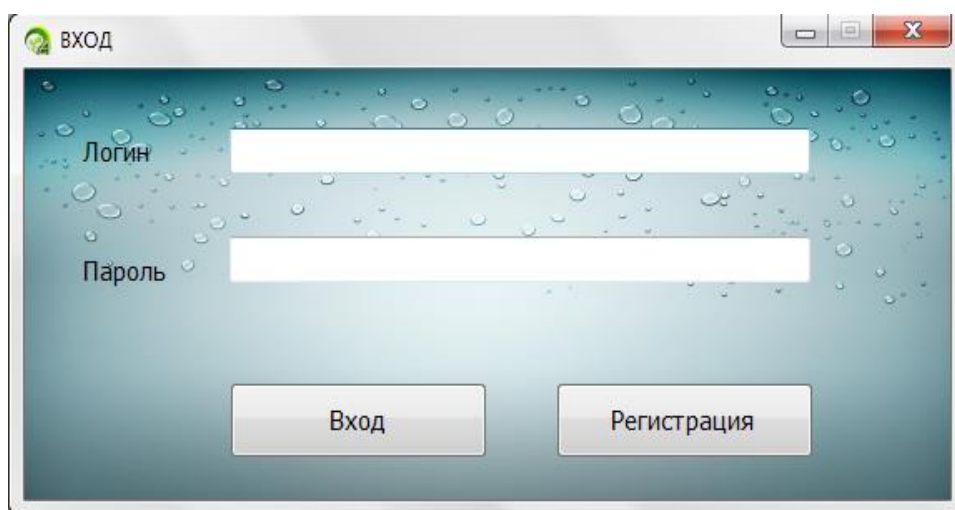


Рисунок 3.2 – Окно регистрации данных

Основная форма. В данном окне содержатся основные данные, благодаря которому можно пройти к различным функциям и процессам системы.

Имеется основное меню, где находятся:

- файл – в этой вкладке содержатся функции системы;
- вид – в этой вкладке можно включать и отключать подсказки;
- справка – в этой вкладке содержится справочный материал относительно системы и способы решения определенных задач;
- выход – для выхода из системы.

В основном окне имеется база, где для пользователя отображается список агрокультур, посаженных на данном хозяйстве.

Так же имеется функция вычисления наиболее подходящей почвы по выбору вида, типа, сорта культуры и вида удобрения.

Функции кнопок:

- инженерный калькулятор – кнопка открытия окна «Калькуляция»;
- аналитика – кнопка открытия окна «Аналитика»;
- база знаний – кнопка открытия окна «База знаний»;
- data Mining – кнопка открытия окна «Data Mining»;
- план – кнопка открытия окна «План»;
- - поиск – кнопка для осуществления поиска по БД;
- сортировка – кнопка для сортировки базы данных, для получения необходимых данных с БД.
- запрос – кнопка для составления запроса;
- база данных – кнопка, которая открывает всю базу данных;
- импорт – кнопка, благодаря которой можно импортировать различные БД, для осуществления определенных операций;
- экспорт – кнопка, благодаря которой можно экспортировать данные в текстовый документ WORD, EXCEL и так далее;
- выход – кнопка для выхода из системы. Главное окно изображено на рисунке 3.3.

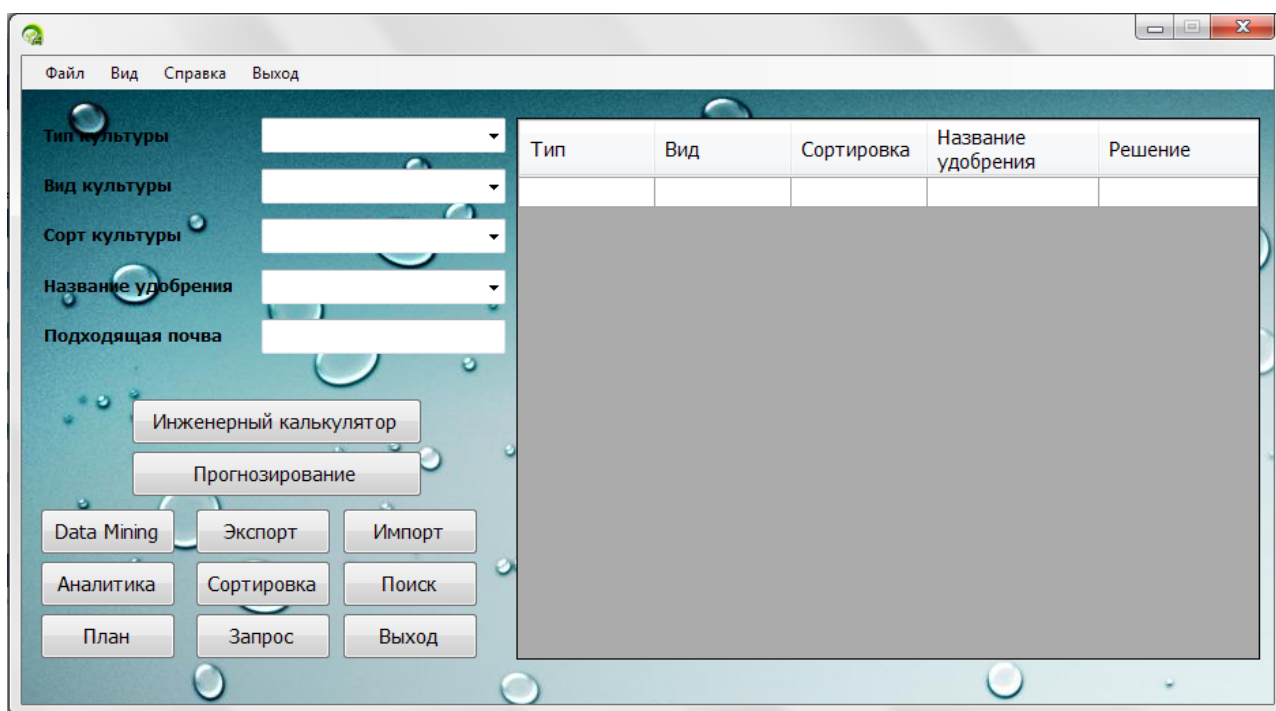


Рисунок 3.3– Основная форма

Форма «Калькуляция». В этой форме производится расчет количества саженцев, семян и необходимого количества удобрения в зависимости от стадии обработки.

Для калькуляции необходимо указать следующие данные:

- площадь (га);
- вид культуры;
- тип культуры;
- сорт культуры;
- вид удобрения;
- № процесса.

В подокне результаты можно получить расчеты. Данное окно изображено на рисунке 3.4.

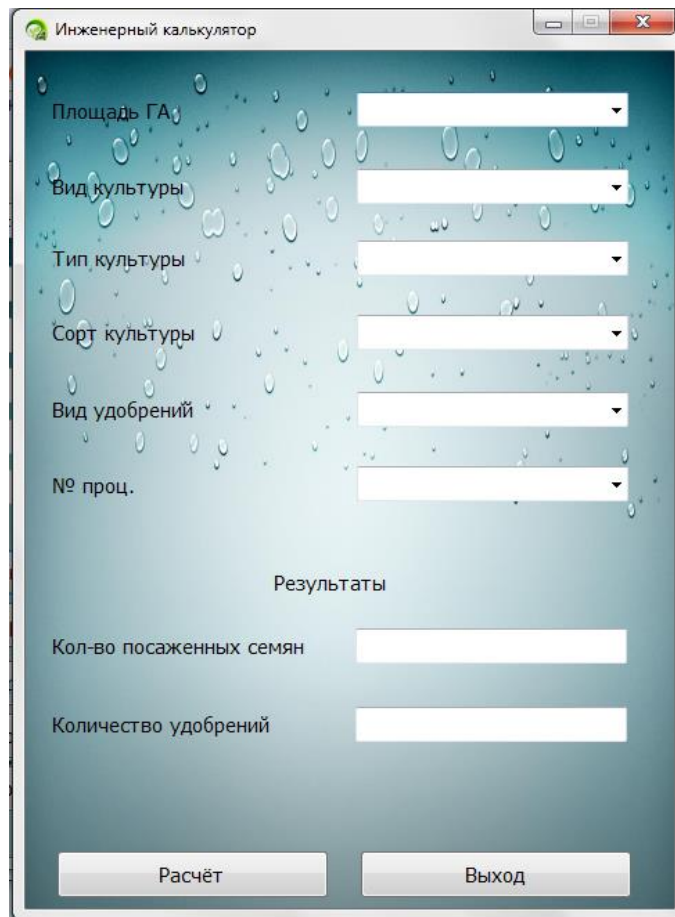


Рисунок 3.4 – Калькуляция

Форма «Data Mining». В данной форме можно получить сравнительные диаграммы, по которым можно сделать выводы, какую культуру выгоднее садить, сколько прибыли можно получить, сколько необходимо вложить для реализации данной культуры. Сравнительные диаграммы строятся за период: год, месяц, неделя.

Функции кнопок:

- импорт данных – данная кнопка осуществляет импорт данных, которые необходимо сравнить и построить диаграммы;
- экспорт данных – данная кнопка позволяет импортировать данные в текстовый документ;
- построить – данная кнопка позволяет построить сравнительные диаграммы;
- печать – кнопка для передачи результатов на печать;
- выход – закрытие формы.

Форма «Data Mining» изображен на рисунке 3.5 и на рисунке 3.6.

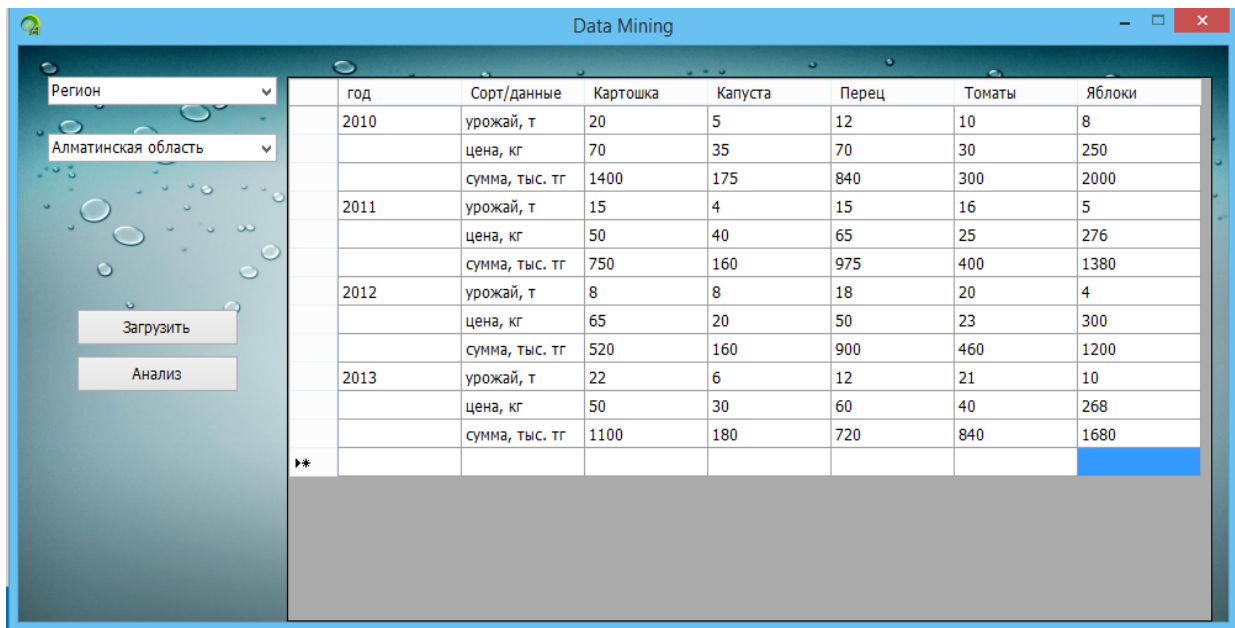


Рисунок 3.5 – Окно загрузки базы данных Data Mining

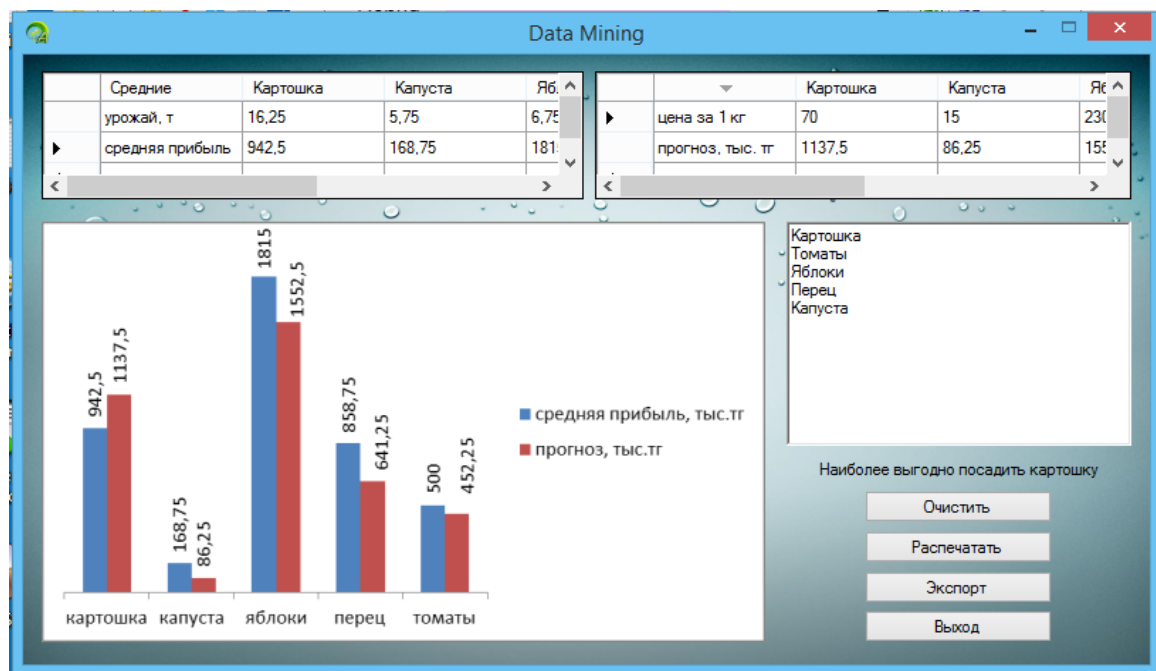


Рисунок 3.6 – Аналитика Data Mining

Форма «Прогнозирование». В данной форме осуществляется прогнозирование прибыльности агрокультуры с учетом метеорологических условий. Этот модуль имеет выход к глобальной сети интернет, где получает данные относительно погоды и даты. Далее система, опираясь на теорию игр и современные технологии интеллектуальных систем принятия решений, определяет наиболее вероятный результат. Система учитывает одну культуру, но разные сорта и опираясь на метеорологические данные определяет, какой

сорт данного вида культуры более выгоднее выращивать, какой вид более холодоустойчивый (Рисунок 3.7 – 3.8).

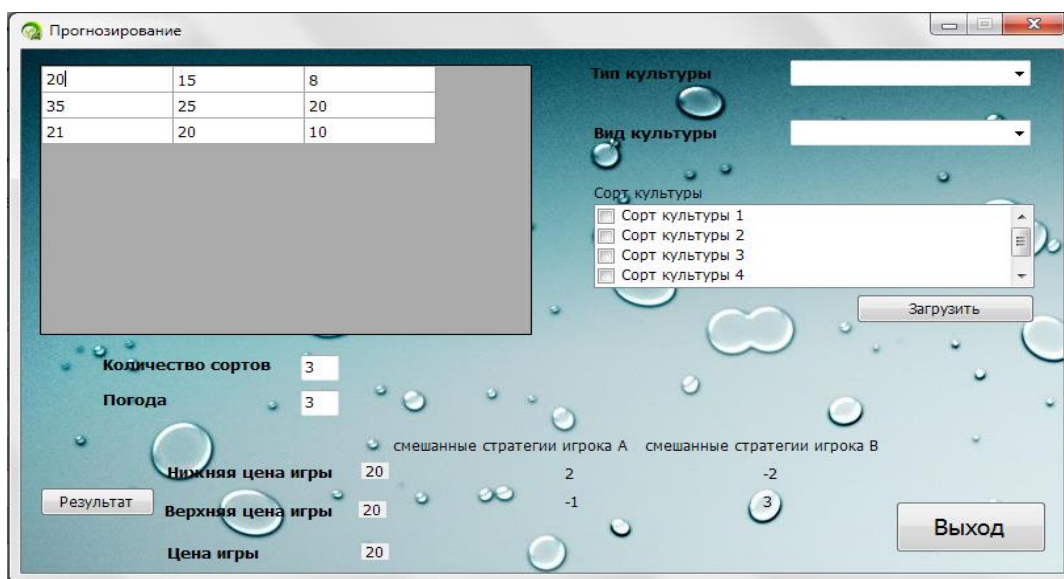


Рисунок 3.7 – Окно прогнозирования

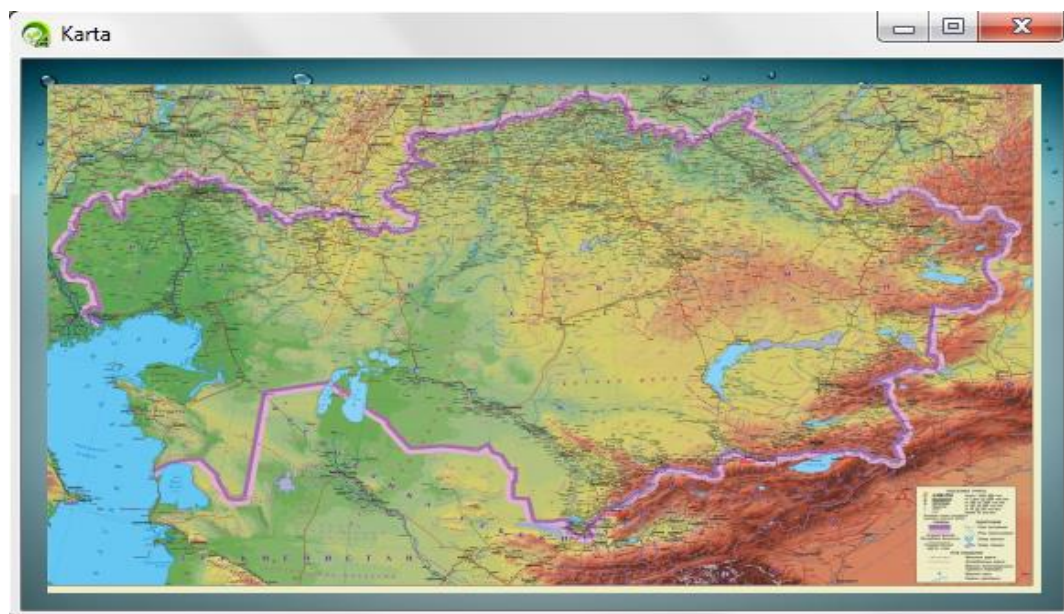


Рисунок 3.8 – ГИС Система

Окно «План». В данном окне осуществляется оформление плана на предыдущий год. Для этого необходимо ввести основные данные и отправить их на оформление.

К основным данным относятся:

- наименование предприятия;
- площадь (га);
- вид культуры;

- тип культуры;
- сорт культуры;
- сроки посева;
- сроки полива;
- сроки обработки;
- финансовые ресурсы;
- количество саженцев;
- количество удобрений.

Функции кнопок:

- оформить – кнопка, которая уже оформляет все данные в виде текстового документа, оформленного по определенному стандарту для вывода на печать.
- калькуляция – кнопка открытия окна «Калькуляция».
- импорт – Кнопка для импорта данных, необходимых для составления плана (Рисунок 3.9).

The image shows a software window titled "Plan" with a light blue background and a water droplet pattern. The window contains several input fields for data entry, each with a label in Russian. At the bottom, there are three buttons: "Оформить" (Format), "Калькулятор" (Calculator), and "Импорт" (Import).

Наименование предприятия	<input type="text"/>
Площадь, Га	<input type="text"/>
Вид культуры	<input type="text"/>
Тип культуры	<input type="text"/>
Сорт культуры	<input type="text"/>
Сроки посева	<input type="text"/>
Сроки полива	<input type="text"/>
Сроки обработки	<input type="text"/>
Финансовые ресурсы	<input type="text"/>
Количество саженцев	<input type="text"/>
Количество удобрений	<input type="text"/>
Дополнительная информация	<input type="text"/>

Buttons:

Рисунок 3.9 – План

Окно «База знаний». В данном окне содержится полная информация об агрокультуре, удобрениях и свойстве почв. Это некое подобие небольшого справочника для неопытных пользователей и людей, которые не имеют никакие знания относительно сельского хозяйства. В этом окне можно выбрать три направления:

- агрокультура;
- удобрения;

– почва.

Можно осуществить поиск. В данной системе будет реализован мягкий поиск. Необходимо просто вводить буквы в поле поиска. В левой колонке будут выводиться результаты поиска, а в правой содержится полная информация. Эти данные можно экспортировать при необходимости (Рисунок 3.10).

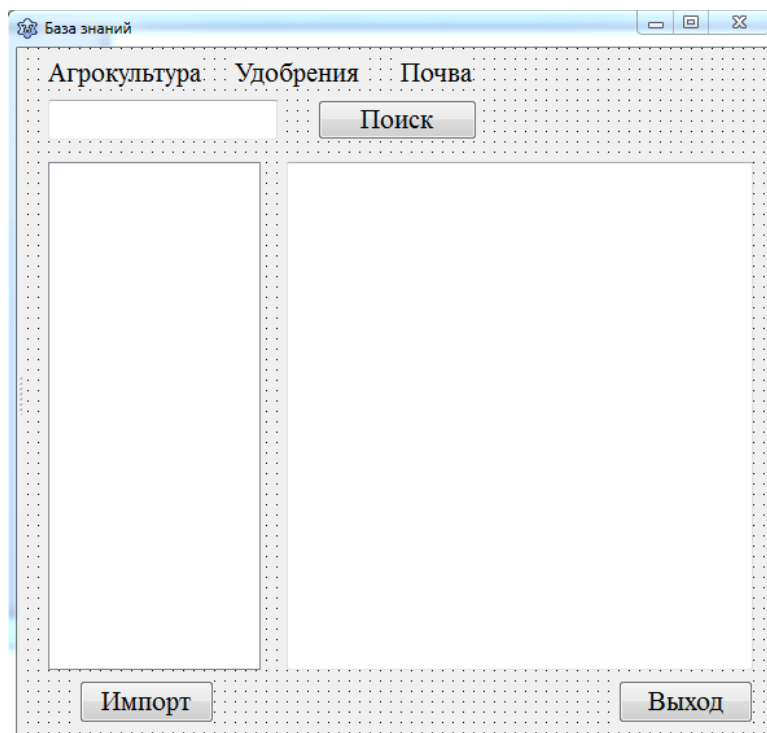


Рисунок 3.10 – База знаний

Перед агропредпринимателем стоит задача: сколько гектаров взять в аренду и что посадить, чтобы получить максимальную прибыль. На помощь агропредпринимателю приходит ИИСППР «Агроном», которая даст рекомендации по принятию эффективного решения по выбору культуры для посева и по оптимальному распределению площади посева под сорта культуры. Для формулировки таких рекомендаций ИИСППР «Агроном» решает следующие задачи:

- прогнозирование урожайности и ожидаемой прибыли;
- оптимальное планирование посева.

Решение задач основываются на технологии Data Mining и моделях и методах теории игр.

На следующем рисунке изображено окно «Прогнозирование ожидаемой прибыли»

На данном этапе разработки системы для осуществления анализа и прогнозирования данных используется пакет STATISTICA.

Система STATISTICA производится фирмой StatSoft Inc., (США), основанной в 1984 г.

На рисунке 3.11 изображено окно прогнозирования урожая ИИСППР «Агроном».

год	Сорт/данные	Картошка	Капуста	Перец	Томаты	Яблоки
2010	урожай, т	20	5	12	10	8
	цена, кг	70	35	70	30	250
	сумма, тыс. тг	1400	175	840	300	2000
2011	урожай, т	15	4	15	16	5
	цена, кг	50	40	65	25	276
	сумма, тыс. тг	750	160	975	400	1380
2012	урожай, т	8	8	18	20	4
	цена, кг	65	20	50	23	300
	сумма, тыс. тг	520	160	900	460	1200
2013	урожай, т	22	6	12	21	10
	цена, кг	50	30	60	40	268
	сумма, тыс. тг	1100	180	720	840	1680
**						

Рисунок 3.11 – Окно прогнозирования

В верхней левой части окна мы выбираем тип анализа (по региону или по культуре).

Далее при помощи кнопки «Загрузить» (Рисунок 3.11) загружается статистическая база данных, содержащая данные по цене, количеству собранных тонн и итоговой прибыли за предыдущие года.

Следующим шагом анализа является составление графика по имеющимся данным. Исходя из имеющихся данных, система составляет прогноз прибыли исходя из средней урожайности культуры и текущей цены. Для наглядного представления результатов программа строит график и выводит результат анализа в текстовом виде (Рисунок 3.12).



Рисунок 3.12 – Окно анализа

Суть использования Data Mining в проекте заключается в следующем – обработка статистических данных и вывод результата после проведения анализа. Из рисунка 3.11 видно, что будут загружаться данные за разные года (2010-2014 гг) и все виды посаженных культур, количество сбора урожая с одного гектара и их стоимость за 1 кг и общая прибыль. С помощью технологии Data Mining программа обрабатывает данные и проводит сравнительный анализ таких переменных, как цена культуры и ее количество. Используя средства технологии, программа визуализирует результат для наглядного представления и сравнения. Как видно из рисунка 3.12, где красный столбец – прогноз, а синий – средняя прибыль, наибольшую величину прогноз имеет в случае яблок, но учитывая среднюю прибыль, выращивание яблок становится убыточным.

ИИСППР «Агроном» рекомендует агропредпринимателю принять решение поле посеять картофелем, так как его прогноз превышает среднюю прибыль.

В правой части программы выводится результат анализа в текстовом виде и расположены функциональные клавиши для работы с результатами анализа.

В следующем окне представлен пример задачи оптимального планирования посева

Урожайность культуры зависит от выбора сорта культуры и погодных условий. А прибыль еще зависит от цены сорта на рынке. Поэтому, чтобы получить максимальную прибыль с наименьшими затратами, необходимо

оптимальным образом распределить площадь посева под конкретные сорта культуры.

Для решения задачи в данном проекте используются модели и методы теории игр. Теория игр — это раздел математической экономики, изучающий решение конфликтов между игроками и оптимальность их стратегий.

Теория игр всегда применяется в таких задачах, где есть явно выраженные противники. Когда необходимо заранее понять кто же из этих противников выиграет и как можно сделать так, чтобы обыграть соперника.

Конфликтная ситуация — ситуация, в которой сталкиваются интересы двух и более лиц.

В нашем случае имеется конфликтная ситуация между агропредпринимателем и погодой. Предположим, что агропредприниматель может посеять три сорта культуры A_1, A_2, A_3 , которые являются его стратегиями. Погода в свою очередь тоже имеет три стратегии: может быть сухой B_1 , нормальной B_2 , дождливой B_3 .

Агропредприниматель имеет информацию о стоимости одного центнера каждого сорта культуры и статистические данные об урожайности каждого сорта культуры в зависимости от погодных условий. На рисунке 3.13 приведены статистические данные.

Погодные условия		Урожайность в центнерах		
		A_1	A_2	A_3
сухая погода	B_1	20	7,5	0
нормальная погода	B_2	5	12,5	7,5
дождливая погода	B_3	15	5	10
Цена за 1 центнер		2	4	8

Рисунок 3.13 – Статистические данные

В ИИСПП «Агроном» данная задача смоделирована в виде матричной игры, которая сводится к задаче линейного программирования и решается симплекс методом. На рисунке 3.14 приведены результаты решения задачи в виде рекомендации для агропредпринимателя.

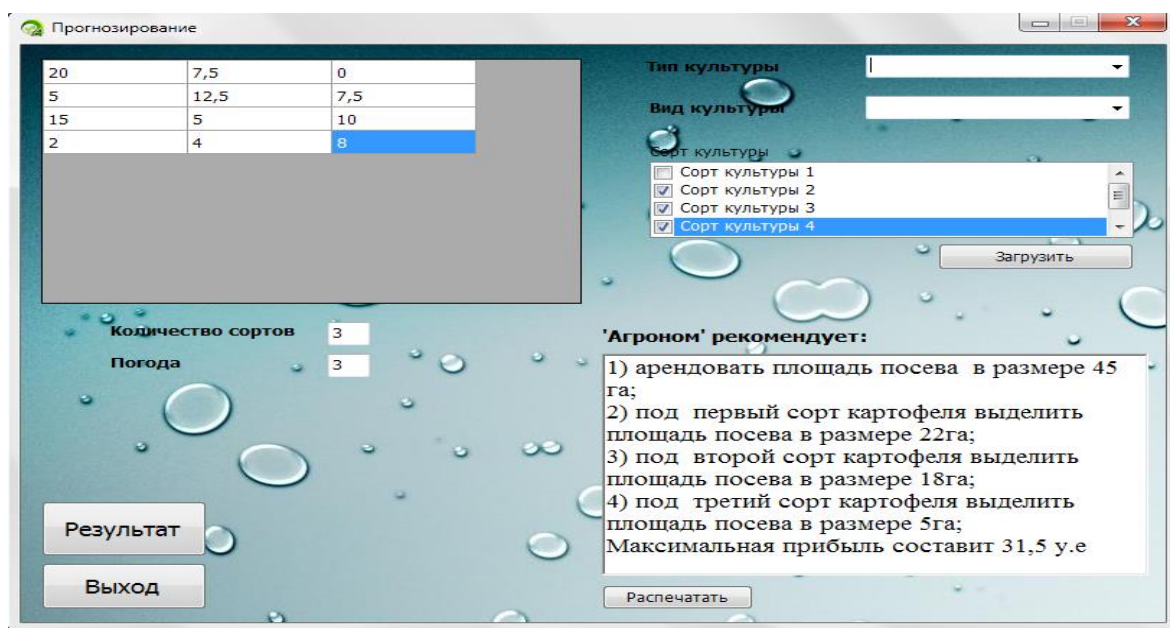


Рисунок 3.14 – Окно «Прогнозирование»

Итак, ИИСППР «Агроном» рекомендует агропредпринимателю принять следующие решения:

- арендовать площадь посева в размере 45 га;
- под первый сорт картофеля выделить площадь посева в размере 22га;
- под второй сорт картофеля выделить площадь посева в размере 18га;
- под третий сорт картофеля выделить площадь посева в размере 5га;

Если агропредприниматель примет рекомендации ИИСППР «Агроном», то он получит максимальную прибыль в размере 31,5 у.е.

4 Экономическая часть

4.1 Описание работы и обоснование необходимости

Для изучения данной области, проводились наблюдения в Енбекшиказахском районе Алматинской области, поселке Каратурык, где люди занимаются земледелием: выращиванием овощей, яблок и табака.

После наблюдения, была замечена следующая проблема. С каждым годом люди получают все лучше результаты посадки, опираясь на свои предыдущие опыты. Но фермеры при посадке культуры всегда идут на большой риск. Начинающие фермеры, от не знания, сажают культуру в той почве, в какой придется, а затем возмущаются, что культура не проросла; сажают то количество семян, которое смогли купить и при этом, даже не вычисляя сроки посадки семян, глубину заделки семян; удобряют в том количестве, в каком смогли, что не есть хорошо и все это в итоге приводит к тому, что фермер разочаровывается, и весь труд его пропадает даром. А опытные фермеры, для получения хорошего результата проводят огромную работу: постоянное слежение температуры и влажности, подсчет количества семян, подсчет необходимого количества удобрения, анализ земли, вычисление срока посева, сам посев, уход.

Таким образом, успех зависит от того, как фермер будет управлять всем процессом производства сельскохозяйственных продуктов. Управление подразумевает под собой принятие решений по выбору почвы, культуры, сортов, удобрения и т.д. с целью минимизации затрат и максимизации прибыли, то есть эффективное и рациональное использование материальных, трудовых и финансовых ресурсов.

Управление в сельском хозяйстве в значительной степени предполагает принятие решений в условиях неопределенности, обусловленной тремя основными причинами: отсутствие текущих данных о состоянии природы; недостаточность знаний о биологических и физических системах; случайный характер протекающих процессов.

Учитывая исключительную важность сельского хозяйства в вопросах обеспечения населения продуктами питания, актуальной является задача эффективного управления процессом производства сельскохозяйственных продуктов с тем, чтобы максимально повысить урожайность с целью получения максимальной прибыли.

В настоящее время принятие решений обычно основывается на отдельных расчетах и экспертных оценках, т.е. на «ручных» способах обработки информации.

В связи с вышесказанным в настоящем проекте рассматриваются вопросы автоматизации управления процессом производства сельскохозяйственных продуктов. Автоматизация достигается за счет создания информационной интеллектуальной системы поддержки принятия

решения, которая поможет фермеру принять эффективное решение по выбору почвы, культуры, сортов, удобрения и т.д.

4.2 Техничко - экономическое обоснование дипломной работы

4.2.1 Трудоемкость разработки ПП

Для определения трудоемкости разработки ПП прежде всего составляется перечень всех основных этапов и видов работ, которые должны быть выполнены. При этом особое внимание должно быть уделено логическому упорядочению последовательности отдельных видов работ и выявлению возможностей их параллельного выполнения, что позволяет существенно сократить общую длительность проведения разработки ПП [10].

Должны быть выделены такие этапы: постановка задачи, разработка алгоритма и блок - схемы, составление программы по готовой блок - схеме, отладка программы на ЭВМ, подготовка соответствующей документации.

Таблица 4.1 - Распределение работ по этапам и видам и оценка их трудоемкости

№ этап а	Содержание этапа	Время, затраченное на этап, час.	Дни, затраченн ые на этап, день	Рабочие часы в день, час.
1	Изучение предметной области, постановка задачи	108	27	4
2	Разработка алгоритма и блок-схемы	120	20	6
3	Требования к интерфейсу	50	10	5
4	Проектирование архитектуры системы	312	52	6
5	Проектирование БД	216	27	8
6	Кодирование ПО	416	52	8
7	Тестирование ПО	60	10	6
8	Интеграция ПО	90	15	6
9	Приемка ПО	40	5	8
10	Внедрение ПО	60	10	6
	Итого, затраченные дни, для создания проекта		228	

Для создания всего проекта, начиная от планирования и до внедрения, необходимо 228 дня, то есть 9 месяцев, учитывая, что в неделе будет 6 рабочих дней, а в месяц 25 рабочих дней ($228/25 = 9,12$). В течение 9 месяцев

программный продукт будет полностью готов, протестирован и внедрен в область применения.

4.3 Расчет затрат на разработку ПП

4.3.1 Расчет материальных затрат

Определение затрат на разработку ПП производится путем составления соответствующей сметы, которая включает следующие статьи:

- материальные затраты;
- затраты на оплату труда;
- социальный налог;
- амортизация основных фондов;
- прочие затраты.

В статью «Материальные затраты» включаются затраты на основные и вспомогательные материалы (бумага, картриджи и другие), энергию, необходимые для разработки ПП.

Общая сумма затрат на материальные ресурсы (Z_M) определяется по формуле:

$$Z_M = \sum_{i=1}^n P_i \times C_i, \quad (4.1)$$

- где P_i - расход i -го вида материального ресурса, натуральные единицы;
 C_i - цена за единицу i -го вида материального ресурса, тг;
 i - вид материального ресурса;
 n - количество видов материальных ресурсов.

Расчет затрат на материальные ресурсы приведены в таблице 4.2.

Таблица 4.2- Затраты на материальные ресурсы

Наименование материала	Марка	Количество	Цена за единицу в тенге	Сумма в тенге
Тетрадь формата А4	Bogheme	2	200	400
Общая тетрадь	ABDI	4	170	680
Ручки	Erich Crause	8	60	480
Листы А3	ABDI	3	310	930
Листы А4	ABDI	3	600	1800
Мышки	Samsung	3	1500	4500
Коврики под мышь	Samsung	3	500	1500
Итого		26	3340	10290

Всего на приобретение материалов понадобилось 10290 тенге.

Таблица 4.3 - ОС и ПО, необходимые для проекта

Наименование материала	Марка	Количество	Цена за единицу в тенге	Сумма в тенге
Ноутбук	Lenovo	3	215000	645000
Принтер	HP Laser Jet	1	27000	27000
Программное обеспечение для ноутбука	Windows, Kaspersky, и т.д.	1	80000	80000
Программное обеспечение для написания ИИСППР «Сельское хозяйство»	VisualStudio, SQL , Data Mining	1	250000	250000
Итого		11	346000	1002000

Интеллектуальная информационная система «Сельское хозяйство» будет разработан в среде VisualStudio, программный код будет написан на языке программирования С#. База данных проекта будет спроектирована на SQL, так же в проект будет сгенерирован Data Mining, для аналитики данных, содержащей в базе.

Всего на приобретение ОС и ПО, необходимые для проекта понадобится 1 002 000 тенге.

4.3.2 Расчет затрат на электроэнергию

Общая сумма затрат на электроэнергию (Z_3) рассчитывается по формуле:

$$Z_3 = \sum_{i=1}^n M_i \times K_i \times T_i \times C \quad (4.2)$$

где M_i - паспортная мощность i -го электрооборудования, кВт;

K_i - коэффициент использования мощности i -го электрооборудования (принимается $K_i=0.7, 0.9$);

T_i - время работы i -го оборудования за весь период разработки ПП ч;

C - цена электроэнергии, тг/кВт×ч;

i - вид электрооборудования;

n - количество электрооборудования.

Таблица 4.4- Затраты на электроэнергию

Наименование	Кол-во оборудования	Паспортная мощность, кВт	Коэффициент использования мощности	Время работы оборудования для разработки ПП, ч	Цена э/энергии, тг/кВт*ч	Сумма, тг
Ноутбук	3	0,12	0,9	1472	24	11446,27
Принтер	1	0,072	0,9	820	24	1275,26
ИТОГО затраты на электроэнергию						12721,53

4.3.3 Расчет заработной платы

В статью «Затраты на оплату труда» включаются расходы по оплате труда всех работников, занятых разработкой ПП (дипломника, руководителей и консультантов дипломной работы, привлеченных лиц).

Общая сумма затрат на оплату труда ($Z_{тр}$) определяется по формуле:

$$Z_{тр} = \sum_{i=1}^n ЧС_i \times T_i, \quad (4.3)$$

где $ЧС_i$ - часовая ставка i -го работника, тг;

T_i - трудоемкость разработки ПП, чел.×ч;

i - категория работника;

n - количество работников, занятых разработкой ПП.

Часовая ставка работника может быть рассчитана по формуле:

$$ЧС_i = \frac{ЗП_i}{ФРВ_i}, \quad (4.4)$$

где $ЗП_i$ - месячная заработная плата i -го работника, тг;

$ФРВ_i$ - месячный фонд рабочего времени i -го работника, час.

В статью «Социальный налог» включается сумма, которая рассчитывается как 11% от затрат на оплату труда всех работников ($Z_{тр}$),

занятых разработкой ПП. При расчете необходимо учесть, что пенсионные отчисления (10% от $Z_{тр}$) не облагаются социальным налогом.

Основная заработная плата за разработку ПП определяется в зависимости от тарифной ставки разработчика за час работы и времени затраченного на каждый этап работы.

Над программным продуктом работают три сотрудника:

- руководитель проекта – составление плана, изучение предметной области, разработка идеи, сбор данных, контроль каждого процесса и этапа, тестирование;
- IT разработчик- проектирование архитектуры системы, написание кода программы, интеграция каждой подсистемы, тестирование;
- администратор БД – ввод данных в базу хранилища, соединение БД с проектом, идея наилучшего вывода результата, составление запросов для проекта [9].

Таблица 4.5 - Расчёт основной заработной платы разработчиков ПП

Наименование и содержание этапов работ	Исполнитель	Рабочие часы в день, час	Заработная плата за час работы, тг	Сумма заработной платы в день, тг	Сумма платы за затраченные дни, тг
Изучение предметной области	Руководитель проекта	4	500	2000	54000
Анализ функциональных возможностей	Руководитель проекта	6	500	3000	60000
Требования к интерфейсу	IT Разработчик	5	500	2500	25000
Проектирование архитектуры системы	Руководитель проекта	4	500	2000	104000
	IT Разработчик	6	500	3000	156000
Проектирование БД	Администратор БД	8	400	3200	108000
Кодирование ПО	IT Разработчик	8	500	4000	208000
Тестирование ПО	IT Разработчик,	6	500	3000	30000
	Администратор БД	6	400	2400	24000
	Руководитель проекта	6	500	3000	30000
Интеграция ПО	IT Разработчик	6	500	3000	45000

Продолжение таблицы 4.5

	Администратор БД	6	400	2400	36000
	Руководитель проекта	6	500	3000	45000
Приемка ПО	Руководитель проекта	8	500	4000	20000
Внедрение ПО	Руководитель проекта	6	500	3000	30000
Итого					975000

Таблица 4.6 – Суммарная заработная плата за программный продукт

Должность	Зарботная плата, тг
Руководитель проекта	343000
Разработчик	464000
Программист БД	168000
Итого	975000

Таблица 4.7 – Начисление социального налога

	Кол – во, человек	Зарботная плата, тг	Пенсионные отчисления, тг	Социальный налог, тг
Администратор БД	1	168000	16800	16632
IT разработчик	1	464000	46400	45936
Руководитель проекта	1	343000	34300	33957
Фонд оплаты труда		975000	97500	96525
Итого	3	975000	97500	96525

4.3.4 Расчет амортизации

В статью «Амортизация основных фондов» включается сумма амортизационных отчислений от стоимости оборудования и программного обеспечения (ПО), используемых при разработке ПП. Амортизационные отчисления рассчитываются по форме, приведенной в формуле 1.5.

Общая сумма амортизационных отчислений определяется по формуле:

$$Z_{AM} = \sum_{i=1}^n \frac{\Phi_i \times H_{Ai} \times T_{НИРi}}{100 \times T_{ЭФi}}, \quad (4.5)$$

где Φ_i - стоимость i-го ОФ, тг;

H_{Ai} - годовая норма амортизации i-го ОФ, %;

$T_{НИРi}$ - время работы i-го ОФ за весь период разработки ПП, ч;

$T_{\text{Эфи}}$ - эффективный фонд времени работы i -го ОФ за год, ч/год;
 i - вид ОФ;
 n - количество ОФ.

Таблица 4.6 - Расчет амортизационных отчислений

Наименование оборудования	Первоначальная стоимость, тг	Норма амортизации, %	Сумма амортизации, тг
Ноутбук	645000	20	129000
Принтер	27000	20	5400
Программное обеспечение	80000	15	12000
Программное обеспечение для написания проекта	250000	15	37500
Сумма амортизации на 1 год			183900
Сумма амортизации на 9 месяцев			137925

Общая сумма амортизации на один год составляет 183900 тенге, а сумма амортизации на 9 месяцев = $(183900/12)*9$. В итоге сумма амортизации на 9 месяцев = 137925 тенге.

4.3.5 Арендная плата

Арендная плата рассчитывается в зависимости от стоимости аренды 1 кв.м площади и времени разработки ПП.

Таблица 4.7 – Плата за аренду, включает коммунальные услуги

Площадь, м ²	Цена за 1 м ²	Аренда за 1 месяц, тенге	срок, месяц	Арендная плата за 6 месяцев, тенге
20	3000	60000	9	540000

Для разработки ПП необходимо 8 месяцев, поэтому аренду помещения рассчитываем за 8 месяцев.

Таблица 4.8 – Расходы на интернет

Расходы на интернет за 1 месяц, тенге	Месяцев	Оплата за интернет, тенге
3500	9	31500

Все составляющие себестоимости разработки ПП отражены в таблице 4.9.

Т а б л и ц а 4.9 - Смета затрат на разработку ПП

Статья затрат	Сумма, тг
Заработная плата основных разработчиков	975000,00
Социальный налог	96525,00
Амортизационные отчисления	137925,00
Расходные материалы	10290,00
Расходы на интернет	31500,00
Арендная плата, включая коммунальные платежи	540000,00
Затраты на электроэнергию	12721,53
Затраты на сертификацию и лицензирование	200000,00
Расходы на рекламу и поиски покупателя	80000,00
Прочие производственные расходы,(2 % от всех затрат)	41679,34
Себестоимость программного продукта	2125640,87
Затраты на создание копий дисков с ПП	80000
Затраты на рекламу	100000
Итого	2305640,87

Из результатов видно, что себестоимость программного продукта = 2 305 640,87 тенге. Учитывая, что в Казахстане всего 4 основных сельскохозяйственных регионов (Северный Казахстан, Алматинская область, Жамбылская область, Южно-Казахстанская область), а в самой Алматинской области 2 крупных сельскохозяйственных комплексов, будет выпущено 6 копий программного продукта, стоимость каждой из которых равно $2\,305\,640,87/6 = 384\,273,478$ тенге.

4.4 Определение возможной (договорной) цены ПП

Величина возможной (договорной) цены ПП должна устанавливаться с учетом эффективности, качества и сроков ее выполнения на уровне, отвечающем экономическим интересам заказчика (потребителя) и исполнителя.

Договорная цена (C_d) для прикладных ПП рассчитывается по формуле:

$$C_d = Z_{НИР} \cdot \left(1 + \frac{P}{100}\right), \quad (4.6)$$

где $Z_{НИР}$ - затраты на разработку ПП (из таблицы 1.8), тг;

P - средний уровень рентабельности ПП = 20%, тогда

$$C_d = 384\,273,478 \times \left(1 + \frac{20\%}{100}\right) = 461\,128,1736 \text{ тг}$$

Согласно формуле расчета договорной цены, стоимость ПП составила 461 128, 1736 тенге.

Далее определяется цена реализации с учетом налога на добавленную стоимость (НДС), ставка НДС устанавливается законодательно Налоговым Кодексом РК. На 2014 год ставка НДС установлена в размере 12%.

Цена реализации с учетом НДС рассчитывается по формуле:

$$C_p = C_d + C_d * \text{НДС}, \quad (4.7)$$

$$C_p = 461\,128,1736 + 461\,128,1736 * 12\% = 516\,463,55 \text{ тг}$$

При установленной договорной цене в размере 461 128, 1736 тенге, рыночная цена одной копии ПП с учетом НДС составила 516 463, 55 тенге.

4.5 Оценка социально - экономических результатов функционирования ИИСППР «Сельское хозяйство»

Социально - экономические результаты функционирования ПП необходимо оценить по четырем основным критериям:

- качеству процессов управления;
- длительности процессов и срокам получения результатов;
- затратам на выполнение работ с использованием ПП;
- числу специалистов, занятых в автоматизируемом процессе, и

характеру их труда.

Производственный процесс полностью автоматизирован и не требует вмешательства со стороны, за исключением начального ввода данных, после чего программа произведет все необходимые расчеты.

Расчеты, в свою очередь, производятся моментально после введения необходимых данных, при условии наличия хорошего и быстрого соединения с интернет сетью.

Затраты касательно продукта относительно невелики и минимизированы. Под затратами на функционирование проекта понимается:

- наличие ПК или приобретение такового при неимении;
- оплата интернет соединения;
- оплата за потребление электроэнергии.

Простой и понятный интерфейс, использующийся в ПП, дает возможность ее использования любому уверенному пользователю ПК. Для выполнения всех работ необходима одна единица рабочей силы. Данный работник и ПП в сумме заменяют собой работу нескольких специалистов (почвовед, мелиоратор, аналитик) тем самым сокращая расходы на оплату труда, что зачастую необходимо на рынке.

Так как продукт несет за собой простое использование и снижение расходов предпринимателей, с большей вероятностью можно предположить перспективность программы, поскольку основной целью любого бизнеса является получение максимальной прибыли, вкладывая минимальные расходы.

ПП без каких либо сложностей можно использовать на среднестатистический ПК, так как он не требует больших ресурсов вычислительной машины (ОЗУ, ПЗУ, ЦПУ, ГП, ТЧ и т.д.).

Перспективность использования данного ПП очень высокая, так как сельское хозяйство широко развито и в нашей стране нет пока аналогов такого программного продукта. Учитывая, что в Казахстане всего 4 основных сельскохозяйственных регионов (Северный Казахстан, Алматинская область, Жамбылская область, Южно-Казахстанская область), а в самой Алматинской области 2 крупных сельскохозяйственных комплексов, будет выпущено 6 копий программного продукта, стоимость каждой из которых с учетом НДС составила 516 463, 55 тенге.

Для пробного использования ПП предлагается демо-версия сроком на 1 месяц бесплатного использования.

5 Безопасность жизнедеятельности

5.1 Анализ условий труда отдела планирования посадки культур

5.1.1 Анализ помещения

Помещение находится на втором этаже. Тип помещения: кабинет. Размеры помещения составляют 8×6×3 (соответственно длина, ширина, высота). В помещении работает 6-ть человек. Кабинет имеет 2 окна, расположенных с западной стороны. Размеры окон составляют 2×1,5 (соответственно длина, ширина). План помещения приведен на рис. 5.1.

5.1.2 Характеристики оборудования

В таблице 5.1 представлены характеристики используемого оборудования.

Таблица 5.1 - Оборудование и его характеристики

Название оборудования	Характеристики
Наличие компьютеров и серверов	6 сервера, ПК: Windows 7 профессиональная
Бесперебойные источники питания	4 – Back_UPS
Принтер	1 – HP LJ- Professional M1132 MFP
Лампы люминесцентные серии ЛБ 80-4	Мощность – 80 Вт; Световой поток - 5200лм; Срок службы – 12000 часов; Количество – 24 шт.
Телефон – аппараты CISCO	6 штук

5.1.3 Характеристики микроклиматических условий.

Работа всех 6-ти сотрудников производится сидя и не требует физического напряжения. Данная работа относится к лёгкой физической категории Ia. В таблице 5.2 приведены значения параметров микроклимата для рассматриваемого помещения.

Таблица 5.2 - Характеристики микроклиматических условий помещения

Период работы	Категория работы	Температура воздуха, °С	Относительная влажность воздуха, %	Скорость движения воздуха, м/с
Холодный	Ia	20-25	75 при 25 °С	0,2
Тёплый	Ia	20-26	60 при 26 °С	0,3

5.1.4 Характеристики освещения

Освещение кабинета осуществляется посредством совмещенного освещения: естественное освещение со стороны окон (боковое освещение) и искусственное местное освещение.

В качестве источника искусственного местного освещения используются 6 светильников, в каждом вставлены по 4 люминесцентных лампы.

5.1.5 Характеристики здания:

- в кабинете сидят 6 человек;
- всего на этаже 64 человека;
- на 1-ом этаже 68 человек.

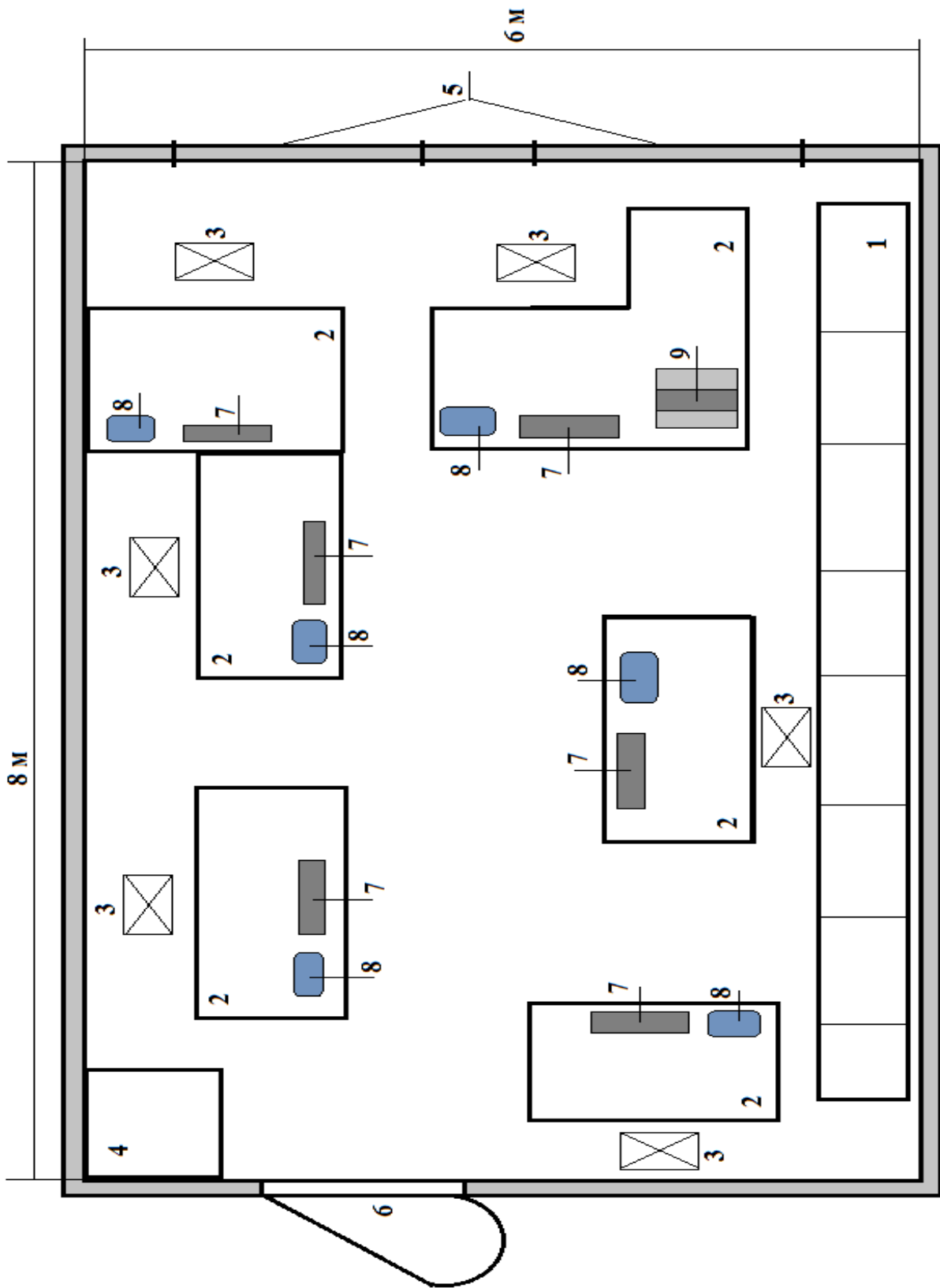


Рисунок 5.1 – План помещения: 1 — стеллаж с документами; 2 — компьютерные столы; 3 — стулья; 4 — шкаф для продуктов; 5 — окна; 6 — дверь; 7 — персональный компьютер; 8 — Cisco аппарат; 9 — принтер, сканер.

5.2 Особенности вынужденного движения людей

В отличие от эвакуации, спасение людей при пожаре представляет собой вынужденное перемещение людей наружу при воздействии на них опасных факторов пожара или возникновении непосредственной угрозы этого воздействия.

Спасение осуществляется самостоятельно, с помощью пожарных подразделений или специально обученного персонала, с использованием спасательных средств, через эвакуационные или аварийные выходы.

В качестве аварийных выходов могут использоваться выходы, не отвечающие требованиям, предъявляемым к эвакуационным. К аварийным выходам могут относиться: выход на открытый балкон или лоджию с глухим простенком не менее 1,2 м от торца балкона (лоджии) до оконного проема (остекленной двери) или не менее 1,6 м между остекленными проемами, выходящими на балкон (лоджию); выход на открытый переход шириной не менее 0,6 м, ведущий в смежную секцию здания класса Ф.1.3 или в смежный пожарный отсек через пожарную зону; выход на балкон или лоджию, оборудованные наружной лестницей, поэтажно соединяющей балконы или лоджии; выход на кровлю здания через окно, дверь или люк. Коллективное спасательное устройство – средство спасения при пожаре, которым одновременно может воспользоваться группа людей. Это специальные помещения внутри здания, предназначенные для размещения людей при пожаре в случае невозможности их эвакуации по имеющимся в здании эвакуационным путям и выходам.

При оснащении фасадов зданий подъемными устройствами для ремонта и очистки фасадов указанные устройства должны рассчитываться на использование пожарными подразделениями, в том числе для спасения людей. На стадии проектирования высотных зданий должна быть также рассмотрена возможность использования вертолетов для спасения людей. Индивидуальное спасательное устройство – средство для защиты органов дыхания от продуктов горения. В чрезвычайных ситуациях весьма эффективным средством спасения людей являются средства индивидуальной защиты в виде самоспасателей.

5.3 Параметры движения людей

Основными параметрами, характеризующими процесс эвакуации из зданий и сооружений, является: плотность D , скорость движения v людского потока, пропускная способность пути (выходов) Q и интенсивность движения q . Кроме того, эвакуационные пути, как горизонтальные, так и наклонные, характеризуются свободной длиной l и шириной δ движения.

Люди, двигаясь по эвакуационным путям, образуют людской поток. Размещение людей в потоке случайное и может меняться в процессе движения.

Плотность людского потока D , состоящая из N людей, равна:

$$D = N/A \quad (5.1)$$

где A — площадь пути эвакуационного участка, m^2 ;

$$A = \delta l \quad (5.2)$$

Путь движения людского потока состоит из нескольких участков:

1) расстояние в кабинете до дверного проёма с выходом в коридор $l = 4$ метра, ширина перехода $\delta = 5$ метра:

$$A_1 = 4 \times 5 = 20 \text{ м}^2;$$

2) наибольшее расстояние до дверного проема с выходом на лестничную клетку $l = 45$ метров, ширина перехода $\delta = 3$ метров:

$$A_2 = 45 \times 3 = 135 \text{ м}^2;$$

3) расстояние до лестничной клетки $l = 4$ метров, ширина перехода $\delta = 6$ метров:

$$A_3 = 6 \times 4 = 24 \text{ м}^2;$$

4) длина лестничной клетки $l = 12$ метров, ширина перехода $\delta = 3$ метра (всего 2-е лестничных клетки):

$$A_4 = 12 \times 3 = 36 \text{ м}^2;$$

5) расстояние до дверного проема с выходом наружу $l = 4$ метра, ширина перехода $\delta = 10$ метров:

$$A_5 = 4 \times 10 = 40 \text{ м}^2;$$

Отсюда общая площадь пути эвакуационного движения:

$$A = 20 + 135 + 24 + 36 + 40 = 255 \text{ м}^2$$

Плотность людского потока D на 1-ом участке, состоящем из $N = 6$ людей, равна:

$$D = 20 \div 6 = 4 \text{ чел/м}^2;$$

Плотность людского потока D на 2-ом участке, состоящем из $N=64/2$ человека, равна:

$$D = 32 \div 135 = 0,282 \text{ чел/ м}^2;$$

Плотность людского потока D на 3-ем участке, состоящем из $N=64/2$ человека, равна:

$$D = 64 \div 24 = 2,67 \text{ чел/м}^2;$$

Плотность людского потока D на 4-ом участке первой лестничной клетки, состоящем из $N=64/2$ людей, равна:

$$D = 32 \div 36 = 0,89 \text{ чел/м}^2;$$

Плотность людского потока D на 4-ом участке второй лестничной клетки, состоящем из $N=(64+68)/2$ людей, равна:

$$D = 66 \div 40 = 1,65 \text{ чел/м}^2;$$

Если площадь горизонтальной проекции человека обозначить через f (таблица 5.3), то формула определения плотности, $\text{м}^2/\text{м}^2$ примет вид:

$$D = N \times f / A \quad (5.3)$$

При неоднородном составе людей формула определения плотности может быть записана в следующем виде:

$$D = \frac{N_1 f_1 + N_2 f_2 + \dots + N_n f_n}{A} = \frac{\sum N_n f_n}{A} \quad (5.4)$$

где N_n — количество людей, имеющих площади горизонтальной проекции f_n .

Плотность людского потока D на 1-ом участке, состоящем из $N=6$ людей в демисезонной одежде, равна:

$$D = \frac{6 \times 0.113}{20} = 0.0339$$

Плотность людского потока D на 2-ом участке, состоящем из $N=64/2$ людей в демисезонной одежде, равна:

$$D = \frac{32 \times 0.113}{135} = 0.027$$

Плотность людского потока D на 3-ем участке, состоящем из $N= 64/2$ людей в демисезонной одежде, равна:

$$D = \frac{32 \times 0.113}{24} = 0.15$$

Плотность людского потока D на 4-ом участке первой лестничной клетки, состоящем из $N= 64/2$ людей в демисезонной одежде, равна:

$$D = \frac{32 \times 0.113}{36} = 0.1$$

Плотность людского потока D на 2-ом участке второй лестничной клетки, состоящем из $N= (64+68)/2$ людей в демисезонной одежде, равна:

$$D = \frac{32 \times 0.113}{40} = 0.373$$

Таблица 5.3- Площади горизонтальной проекции человека

Возраст, одежда человека и вид груза	Площадь горизонтальной проекции человека f , m^2
Взрослый человек:	
в летней одежде	0,10
в демисезонной одежде	0,113,
в зимней одежде	0,125
с ребенком на руках	0,285
с рюкзаком	0,315
с легким свертком	0,235
Подросток	0,07
Ребенок	0,04—0,05

В том случае, когда необходимо подойти к оценке плотности в чел/ m^2 , полученную плотность делят на f .

При плотности до $0,05 m^2/m^2$ человек имеет полную свободу движения как по направлению, так и при желаемой ему скорости. При плотности в интервале $0,05 < D < 0,15$ человек не может свободно менять направление своего движения; при плотности выше $0,15 m^2/m^2$ люди практически начинают двигаться слитно. Верхней границей слитного движения следует считать $D < 0,92 m^2/m^2$. Поэтому этой величиной ограничивается плотность при проектировании эвакуационных путей.

Образование высоких плотностей наиболее вероятно и наиболее опасно в случае вынужденной эвакуации, когда люди стремятся быстрее уйти от

источника опасности. Естественное стремление людей уйти от источника опасности диктует необходимость сократить до минимума время их пребывания в опасной зоне.

Пути эвакуации должны обеспечивать возможность людям покинуть помещение за время $t_{\text{нб}}$, не превышающее t_p , являющееся критерием безопасности для людей. Это условие $t_{\text{нб}} \leq t_p$ — еще одно расчетное предельное состояние путей движения.

5.4 Эвакуационные выходы и пути

Эвакуационные выходы - это дверные проемы, которые ведут из помещений непосредственно на улицу; в лестничную клетку с выходом наружу непосредственно или через вестибюль. Все эвакуационные выходы отделены от коридоров перегородками с дверьми, в проход или в коридор с непосредственным выходом наружу или в лестничную клетку; в соседние помещения того же этажа, обладающие огнестойкостью не ниже III степени, не содержащие производств, относящихся по пожарной опасности к категориям А, Б и Е, а также имеющие непосредственный выход наружу или на лестничную клетку (рисунок 5.2).

Если имеются проемы, которые не подлежат вышеуказанным признакам, то они не считаются эвакуационными и их при расчете не учитывают.

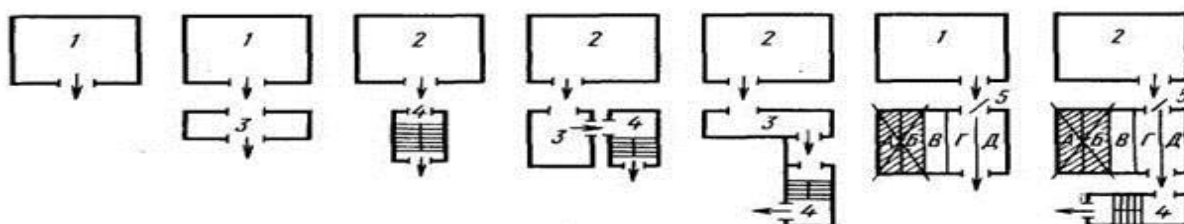


Рисунок 5.2 - Допустимые варианты устройства эвакуационных выходов: 1 — помещения в первом этаже; 2 — помещения в любом этаже, кроме первого; 3 — проходные коридоры или вестибюль; 4 — лестничные клетки; 5 — проходные помещения, которые при содержании категорий

Все эвакуационные пути должны вести к выходу и должны обеспечить защиту и безопасность людям в течение определенного времени. Самые распространенные пути эвакуации — это проходы, коридоры, фойе и лестницы. Лифты, эскалаторы и другие пути передвижения, которые требуют источники энергии, не относятся к эвакуационным путям, так как они могут вызвать пожары, возгорания и различные аварийные ситуации. Незадымляемая лестничная клетка в зданиях с высокой этажностью и различные тоннельные переходы относятся к эвакуационным выходам. Так же имеются запасные выходы, которые при нормальном движении не используются, но их использование при аварийных ситуациях является вынужденным. В помещениях с массовым пребыванием

людей запасные выходы в расчет эвакуации не принимаются. Чтобы результативно, быстро и безопасно выволить людей из помещения, устанавливают не менее двух эвакуационных путей, но в многоэтажных зданиях, где маленькая площадь, и одного эвакуационного выхода достаточно. Для безопасности от пожаров при аварийном случае, обязательно устанавливают выход на пожарную лестницу. Сам процесс эвакуации людей из зданий делится на три этапа. Первый этап — это эвакуация самых отдаленных от выхода людей, к эвакуационному выходу ведут людей с самих удаленных рабочих мест. К таким помещениям относят производственные цехи, мастерские, зрительные залы в театрах, кинотеатрах, клубах, классы и аудитории в учебных заведениях, палаты лечебных учреждений и др.

Второй этап это движение людей по коридору, то есть выход сотрудник на наружу. Такое движение происходит по коридорам, проходам к фойе и лестничным клеткам и по лестницам — через вестибюль наружу. Если здание одноэтажное с простым функциональным процессом, второй этап эвакуации может представлять собой движение от эвакуационного выхода из помещений по коридору непосредственно наружу. Иногда в производственных и жилых одноэтажных зданиях выход из помещений совпадает с выходом наружу. В различных ситуациях может получиться так, что второй этап пропускается, так как за частую выход наружу и совпадает с эвакуационном выходом.

Третий этап — после выхода людей на улицу, наступает третий этап, когда сотрудники сливаются с толпой и уже начинается оказание первой медицинской помощи.

«Строительные нормы и правила» допускают для большинства помещений, за исключением театров и дворцов спорта, определять суммарную ширину маршей лестничных клеток, а также ширину дверей, коридоров или проходов на путях эвакуации из расчета не менее 0,6 м на 100 человек.

Самая маленькая ширина коридора или прохода высчитывается расчетом, но они не должны быть менее 1,0 м. Исключение составляют проходы, нижняя часть высоты которых может быть частично занята оборудованием (например, проходы между креслами зрелищных помещений, между столами в учебных аудиториях, классах, в проектно-конструкторских бюро и т. п.). Их ширина может быть менее 0,5 м.

Двери, выходящие в коридор из примыкающих помещений, могут открываться по ходу людского потока или внутрь помещения. Минимальная ширина лестничных маршей определяется расчетом, но не должна быть меньше установленной по условиям одиночного перемещения людей. Максимальная ширина марша допускается 2,4 м.

Если имеются лестничные площадки с распашными дверьми и входами в лифты, то их ширина должна быть не менее 1,6 м. Двери из помещений и коридоров в лестничные клетки в открытом положении не должны уменьшать расчетную ширину пути эвакуации.

5.5 Время эвакуации людей из отдельных помещений и зданий в целом

Для расчета времени эвакуации людей полностью рассчитывают время движения маленьких и больших потоков людей по всему зданию, начиная с самых дальних мест размещения до выходов.

Чтобы определить путь движения людских групп, все участки, которые проходят люди (проход, коридор, дверной проем, лестничный марш, тамбур) подразделяются на более мелкие участки длиной l_i , и шириной δ_i . С самого начала разделение идет с рабочего места, проходы между столами, оборудованием и так далее. При определении расчетного времени длина и ширина каждого участка пути эвакуации принимаются по проекту. Различные лестничные длины и длина пандуса измеряется маршом. Такие длины пути как дверной проем не учитываются, их приравнивают к нулю. Проем, расположенный в стене толщиной более 0,7 м, а также тамбур следует считать самостоятельными участками горизонтального пути, имеющими конечную длину l_i .

Расчетное время эвакуации людей t_p определяется как сумма времени движения людского потока по отдельным участкам пути:

$$t_p = t_1 + t_2 + \dots + t_i, \quad (5.4)$$

где t_1 — время движения людского потока на первом (начальном) участке, мин; t_2, \dots, t_i — время движения людского потока на каждом из следующих после первого участка пути, МИН
Время движения людского потока по первому участку пути:

$$t_1 = l_1 / v_1, \quad (5.5)$$

где v_1 — скорость движения людского потока по горизонтальному пути на первом участке определяется по таблице 5.5.1 в зависимости от плотности D_1 , м/мин.

Плотность людского потока D_1 на первом участке пути, имеющем длину l_1 , и ширину δ_1 , равна:

$$D_1 = N_1 f / l_1 \delta_1, \quad (5.6)$$

где N_1 — количество людей на первом участке;
 f — средняя площадь горизонтальной проекции человека, принятая по таблице 5.3.

Плотность людского потока D на 1-ом участке, состоящем из $N=6$ людей в демисезонной одежде, равна:

$$D = \frac{6 \times 0.113}{20} = 0.0339$$

$$t_1 = 4/100 = 0,04 \text{ мин};$$

Плотность людского потока D на 2-ом участке, состоящем из $N= 64/2$ людей в демисезонной одежде, равна:

$$D = \frac{32 \times 0.113}{135} = 0.027$$

$$t_2 = 45/100 = 0,45 \text{ мин};$$

Плотность людского потока D на 3-ем участке, состоящем из $N= 64/2$ людей в демисезонной одежде, равна:

$$D = \frac{32 \times 0.113}{24} = 0.15$$

$$t_3 = 6/80 = 0,075 \text{ мин};$$

Плотность людского потока D на 4-ом участке первой лестничной клетки, состоящем из $N= 64/2$ людей в демисезонной одежде, равна:

$$D = \frac{32 \times 0.113}{36} = 0.1$$

$$t_{4.1} = 12/80 = 0,15 \text{ мин};$$

Плотность людского потока D на последнем участке, состоящем из $N= (64+68)$ людей демисезонной одежде, равна:

$$D = \frac{132 \times 0.113}{40} = 0.373$$

$$t_5 = 10/40 = 0,25 \text{ мин};$$

Расчетное время эвакуации людей t_p :

$$t_p = 0,04 + 0,45 + 0,075 + 0,15 + 0,25 = 0,965 \text{ мин}$$

Таблица 5.4 - Скорость движения людского потока

Плотность потока D м ² /м ²	Горизонтальный путь		Дверной проем	Лестница вниз		Лестница вверх	
	скорость v, м/мин	интенсивность q, м/мин	интенсивность q, м/мин	скорость v, м/мин	интенсивность q, м/мин	скорость v, м/мин	интенсивность q, м/мин
0,01	100,0	1,0	1,0	100	1,0		0,6
0,05	100,0	5,0	5,0	100	5,0	60,0 60,0	3,0
0,10	80	8,0	8,7	95,0 68,0	9,5	53,0 40,0	5,3
0,20	60,0	12,0	13,4	52,0	13,6	32,0	8,0
0,30	47,0 40,0	14,1	16,5	40,0	15,6	26,0	9,6
0,40	33,0	16,0	18,4	31,0	16,0	22,0	10,4
0,50	27,0	16,5	19,6	24,0	15,5	18,0	11,0
0,60	23,0	16,2	19,0	18,0	14,4	15,0	10,8
0,70	19,0	16,1	18,5	13,0	12,6	13,0 11,0	10,5
0,80	15,0	15,2	17,3	8,0	10,4		10,4
0,90 и более		13,5	8,5		7,2		9,9

Примечание. Табличное значение интенсивности движения в дверном проеме при плотности потока 0,9 и более, равное 8,5 м/мин, установлено для дверного проема шириной 1,6 м и более, а при дверном проеме меньшей ширины (б) интенсивность движения следует определять по формуле

$$q = 2,5 + 3,75 \quad (5.7)$$

Значение скорости v, движения людского потока на участках пути, следующих после первого, принимается по таблице 5.5.1 в зависимости от значения интенсивности движения людского потока по каждому из этих участков пути, которые следует определять для всех участков пути, в том числе и для дверных проемов:

$$q_i = q_{i-1} \delta_{i-1} / \delta_i \quad (5.8)$$

где δ_i δ_{i-1} — ширина рассматриваемого (i) и предшествующего ему (i - 1) участка пути, м;

q_i , q_{i-1} — значения интенсивности движения людского потока по рассматриваемому (i) и предшествующему ему (i-1) участкам пути, м/мин; значение интенсивности движения людского потока на первом участке пути $q = q_{i-1}$, определяется по таблице 5.5.1 по значению D_x , установленному по вышеприведенной формуле. Если полученное значение q_i будет меньше или равно значению q_{max} , то время движения по участку пути: $t_i = l_i / v_i$.

При этом значение q_{\max} следует принимать равным: для горизонтальных путей 16,5 м/мин, для дверных проемов 19,6 м/мин, при движении по лестницам вниз 16,0 м/мин и при движении по лестницам вниз 11 м/мин.

Если значение q_i будет больше q_{\max} , то ширину δ_i , данного участка пути следует увеличивать на такую величину, чтобы соблюдалось условие $q_i < q_{\max}$. При невозможности выполнения этого условия интенсивность и скорость движения людского потока по участку пути (i) определяют по таблице 5.5.1 при значении $D=0,9$ и более.

При слиянии в начале участка (i) двух и более людских потоков интенсивность движения людей равна:

$$q_i = \sum q_{i-1} \delta_{i-1} / \delta_i, \quad (5.9)$$

где q_{i-1} — интенсивность движения людских потоков, сливающихся в начале участка (i), м/мин; δ_{i-1} — ширина участка пути до слияния, м; δ_i — ширина рассматриваемого (i) участка пути, м.

$$q = 1 + 5 \times 3/4 + 8 \times 4/6 + 8 \times 6/3 + 16 \times 3/4 = 82,908 \text{ м/мин}$$

Вывод: эвакуация людей из кабинета осуществится ~ за 1 минут, интенсивность людского потока $q = 38,05$ м/мин.

Заключение

Данная система огромная и очень полезная.

Ценность с точки зрения новизны. Это абсолютно новая идея, такой системы в Казахстане и в странах СНГ пока нет.

Ценность с точки зрения автоматизации. Всеми этими подсчетами, анализом занимается большое количество людей, это агроном, экономист, лаборант. Данная система позволит автоматизировать процесс и получать нужный результат нажатием кнопки.

Ценность с точки зрения прибыльности. Любая автоматизация приводит к прибыли, что и является основной целью. Данная система постоянно будет подсчитывать и предлагать такие решения, которая приносит наибольшую прибыль.

ИИСППР «Сельское хозяйство» имеет приятный и понятный интерфейс, то есть данной системой может воспользоваться даже самый не опытный пользователь ПК и ему не нужно будет вводить формулы либо логические конструкции.

В данном дипломном проекте было произведено технико-экономическое обоснование разработки интеллектуальной информационной системы поддержки принятия решения «Сельское хозяйство».

Из результатов видно, что себестоимость программного продукта, учитывая все прямые и косвенные затраты равно 2 305 640,87 тенге.

Учитывая, что в Казахстане всего 4 основных сельскохозяйственных регионов (Северный Казахстан, Алматинская область, Жамбылская область, Южно-Казахстанская область), а в самой Алматинской области 2 крупных сельскохозяйственных комплексов, будет выпущено 6 копий программного продукта, стоимость каждой из которых с учетом НДС составила 516 463, 55 тенге.

Так же был проведен анализ условий труда. Подсчитывалось время эвакуации людского потока при пожаре. В итоге получилось, что эвакуация людей из кабинета осуществится ~ за 1 минут, интенсивность людского потока $q = 38, 05$ м/мин.

На сегодняшний день реализован только один модуль из всей системы, которая называется «Прогнозирование», то есть предварительный подсчет прибыли посаженной культуры и предложение наиболее эффективного варианта посадки культуры. Интеллектуальная система поддержки принятия решения состоит из трех модулей: прогнозирование, планирование, слежение. В будущем планируется развитие данного ПП: установка датчиков слежения температуры, влажности почвы; ГИС система по строению почвы.

В модуле «Прогнозирование», пользователь вводит входные данные, необходимые условия, в результате получает список решений по предпочтительности.

Список литературы

1. Поспелов Г.С. Искусственный интеллект – основа новой информационной технологии. М.: Наука, 1988. - 524 с.
2. Наумов А.И. и др. Системы управления базами данных и знаний. М.: Финансы и статистика, 1991. - 378 с.
3. ред. Форсайт, Р. Экспертные системы. Принципы работы и примеры; М.: Радио и связь - Москва, 1987. - 224 с.
4. А.А. Барсегян, М.С. Куприянов, В.В. Степаненко, И.И. Холод Методы и модели анализа данных: OLAP и Data Mining, 2002. - 385 с.
5. Губко М.В., Новиков Д.А. — Теория игр в управлении организационными системами, 2000. - 457 с.
6. Путькина Л. В., Пискунова Т. Г. Интеллектуальные информационные системы; СПбГУП - Москва, 2008. - 228 с.
7. Усков А. А., Кузьмин А. В. Интеллектуальные технологии управления; Горячая Линия - Телеком - , 2004. - 144 с.
8. Чубукова И.А., Лекции по Data Mining, Курс лекций интернет-университета INTUIT, 2006 г. – 368 с.
9. Иванов А. А., Калинина М. И. Учебное пособие: Экономика предприятия. – М.: «Экономика и финансы», 2000. - 328 с.
10. Экономика предприятия: Учебник / Под ред. проф. О.И. Волкова. – М.: Инфра-М, 2000. - 325 с.
11. Грузинов В.П. Экономика предприятия М. Банки и биржи Юнити 1998 3. - 342 с.
12. 1. Горемыкин В.А. Планирование на предприятии: учеб. пособие / В.А. Горемыкин. – М.: Юрайт, 2012. – 704 с.

Приложение А

Техническое задание

А.1 Общие положения

А.1.1 Полное наименование системы и ее условное обозначение

Полное наименование системы: Интеллектуальная система принятия решения по подготовке плана в агрономической деятельности, учет урожайности агрономической культуры, анализ и прогноз прибыльности урожая, информационно-управляющая система в области сельского хозяйства.
Краткое наименование системы: ИСПР «Сельское хозяйство».

А.1.2 Шифр темы или шифр (номер) договора

Шифр темы: РГА/СИ - 180992
Номер контракта: 05022211

А.1.3 Наименование предприятий (объединений) разработчика и заказчика (пользователя) системы и их реквизиты

Заказчиком системы является крестьянское хозяйство «Сайра».
Адрес заказчика: Алматинская область, Енбекшиказахский район, село Каратурык, улица Жибек-Жолы.
Разработчиком системы является Андасбаева С.А.
Адрес разработчика: Город Алматы, улица Байтурсынова, Алматинский университет энергетики и связи.

А.1.4 Перечень документов, на основании которых создается система, кем и когда утверждены эти документы

Данная система разрабатывается на основании ТЗ и нормативных документов.

В техническом задании расписываются:

- основные цели, задачи, сроки и этапы разработки;
- список основных функций и требований;
- список функций интерфейса.

Оформление осуществляется на основании нормативных документов.

А.1.5 Плановые сроки начала и окончания работы по созданию системы

Плановый срок начала работ по созданию ИСПР «Сельское хозяйство» - 01.09.2013.

Плановый срок окончания работ ИСПР «Сельское хозяйство» - 20.05.2014.

А.1.6 Сведения об источниках и порядке финансирования работ

ИИСППР «Агроном» создается на бесфинансовой основе.

А.1.7 Порядок оформления и предъявления заказчику результатов работ

Система передается в виде функционирующего комплекса на базе средств вычислительной техники заказчика в сроки, установленные заказчиком. Приемка системы осуществляется комиссией в составе уполномоченных представителей заказчика.

Порядок предъявления системы, ее испытаний и окончательной приемки определен в п.6 настоящего ТЗ. Совместно с предъявлением системы производится сдача разработанного исполнителем комплекта документации согласно п.8 настоящего ТЗ.

А.1.8 Состав используемой нормативно-технической документации

При разработке автоматизированной системы и создании проектно-эксплуатационной документации Исполнитель должен руководствоваться требованиями следующих нормативных документов:

- ГОСТ 34.601-90. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Стадии создания;
- ГОСТ 34.201-89. Информационная технология. Комплекс стандартов на автоматизированные системы. Виды, комплексность и обозначение документов при создании автоматизированных систем;
- РД 50-34.698-90. Методические указания. Информационная технология. Комплекс стандартов на автоматизированные системы. Автоматизированные системы. Требования к содержанию документов.

Продолжение приложения А

А.2 Назначение и цели создания (развития) системы

– Назначение системы – разработка производственного плана, учет выращиваемой агрокультуры, калькуляция необходимого количества саженцев, участка земли, удобрения; анализ урожайности, подсчет прибыльности агрокультуры, учитывая погодные условия и плодородность земельного участка.

– Цели создания системы – автоматизация и сокращение времени работы по разработке производственного плана, учету выращиваемой агрокультуры, калькуляции необходимого количества саженцев, участка земли, удобрения; анализу урожайности, подсчету прибыльности агрокультуры, учитывая метеорологические условия и плодородность земельного участка.

А.2.1. Назначение

ИИСППР «Агроном» предназначена для комплексного информационно-аналитического обеспечения процессов выращивания агрокультуры в части исполнения следующих процессов:

- калькуляция саженцев и необходимого количества удобрения по виду культуры;
- аналитический вывод прибыльности агрокультуры с учетом метеорологических и агрономических условий;
- сравнительный анализ и построение диаграммы по убыткам и прибыли;
- разработка технического плана на предыдущий год.

А.2.2. Цели создания

Основными целями создания ИСППР «Сельское хозяйство» являются:

- облегчение создания плана лицу, занимающемуся агрономической деятельностью;
- повышение эффективности исполнения процессов, перечисленных выше, путем сокращения непроизводительных и дублирующих операций, операций, выполняемых "вручную", оптимизации трудового ресурса и времени;
- повышение качества принятия управленческих решений за счет оперативности представления, полноты, достоверности и удобства форматов отображения информации;
- содержание базы знаний, которой может воспользоваться лицо, не имеющее никаких навыков о данной отрасли.

Продолжение приложения А

Для реализации поставленных целей система должна решать следующие задачи:

- ввод первоначальных данных (количество гектар посаженной продукции) для реализации калькуляции;
- учет метеорологических и агрономических условий;
- построение аналитических отчетов;
- разработка плана.

Продолжение приложения А

А.3 Характеристика объекта автоматизации

Объект автоматизации

Процессы по управлению ИСПР «Сельское хозяйство» (п. 2.1), а также контроль эффективности выполнения указанных процессов. Данные процессы осуществляются следующими специалистами:

- агроном - специалист сельского хозяйства, обладающий всесторонними знаниями в области агрономии;
- владелец крестьянского хозяйства;
- рабочий персонал в области сельского хозяйства.

А.3.1 Существующее программное обеспечение

В настоящий момент деятельность планирования и составления анализа в Казахстане в отрасли сельское хозяйство не автоматизирована.

Вся деятельность осуществляется вручную, замер ведется примерный, а не точный.

А.3.2 Существующее техническое обеспечение

Телекоммуникационная инфраструктура отсутствует. Персональный компьютер имеет агроном или же владелец крестьянского хозяйства.

А.3.3 Существующее нормативно-правовое обеспечение

Существующее нормативно-правовое обеспечение составляют федеральные и областные нормативные правовые акты:

- конституция РК;
- гражданский кодекс РК.

Продолжение приложения А

А.4 Требования к системе

Требования к системе в целом:

- требования к структуре и функционированию системы;
- требования к персоналу системы;
- показатели назначения;
- требования к надежности; безопасности; эргономике и технической эстетике; эксплуатации, техническому обслуживанию, ремонту и хранению компонентов системы; защите информации; сохранности информации при авариях; защите от влияния внешних воздействий;
- дополнительные требования.

Требования к функциям (задачам), выполняемым системой; перечень функций, задач или их комплексов, подлежащих автоматизации (по каждой подсистеме); очередность ввода в эксплуатацию, временной регламент реализации и требования к качеству реализации каждой функции, задачи (или комплекса задач), к форме представления выходной информации, перечень и критерии отказов для каждой функции, по которой задаются требования по надежности.

Требования к видам обеспечения в зависимости от вида системы приводят требования к математическому, информационному, лингвистическому, программному, техническому, метрологическому, организационному, методическому и другим видам обеспечения системы.

А.4.1 Требования к системе в целом

А.4.1.1 Требования к структуре и функционированию системы

В состав ИСПР «Сельское хозяйство» должны решаться следующие задачи:

- занесение информации в БД;
- учет метеорологических и агрономических условий;
- формирование сравнительных диаграмм;
- разработка плана на предыдущий год.

Занесение информации в БД предназначена для проведения калькуляции. Вводятся основные данные (количество гектар крестьянского хозяйства и вид выращиваемой агрокультуры). На основании этих данных подсчитывается количество необходимых саженцев или семян и необходимое количество удобрения.

Учет метеорологических условий предназначена для получения выводов об урожайности на предыдущий год. Метеорологические данные будут учитываться из глобальной сети интернет, а агрономические данные будут содержаться в базе знаний.

Продолжение приложения А

Формирование сравнительных диаграмм предназначена для анализа и получения данных, которые используются при разработке плана на предыдущий год. Сравнительные диаграммы строятся на основании данных по прибыли и расходу за предыдущий год для выявления наиболее оптимального продукта, который более прибыльный и требует меньше расхода.

Разработка плана на предыдущий год предназначена для создания документа, на основании которого будут вестись посевные работы.

ИСПР «Сельское хозяйство» должна обеспечивать доступ в первую очередь агроному, руководителям различных крестьянских хозяйств для сравнения урожайности по различным годам и получения плана.

А.4.1.1.2 Требования к способам и средствам связи для информационного обмена между компонентами системы

Входящие в состав ИСПР «Сельское хозяйство» задачи в процессе функционирования должны обмениваться информацией на основе открытых форматов обмена данными, используя для этого входящие в их состав модули информационного взаимодействия.

Форматы данных будут разработаны и утверждены на этапе технического проектирования.

В состав передаваемых данных входят:

Метеорологический отчет;

Почвоведческий отчет;

Бухгалтерский отчет.

А.4.1.1.3 Требования к характеристикам взаимосвязей создаваемой системы со смежными системами

Смежные системы отсутствуют.

А.4.1.1.4 Требования к режимам функционирования системы

Для ИСПР «Сельское хозяйство» определены следующие режимы функционирования:

Нормальный режим функционирования;

Аварийный режим функционирования.

Основным режимом функционирования ИСПР является нормальный режим.

В нормальном режиме функционирования системы:

– программное обеспечение и технические средства обеспечивают возможность круглосуточного функционирования, с перерывами на обслуживание;

Продолжение приложения А

- исправно работает оборудование;
- исправно функционирует программное обеспечение системы.

Для обеспечения нормального режима функционирования системы необходимо выполнять требования и выдерживать условия эксплуатации программного обеспечения и технических средств системы.

Аварийный режим функционирования системы характеризуется отказом одного или нескольких компонент программного и (или) технического обеспечения.

Аварийный режим может возникнуть в ходе отсутствия или неисправности интернета, так как метеорологические данные будут отсутствовать.

В случае перехода системы в аварийный режим необходимо:

- завершить работу всех приложений, с сохранением данных;
- выполнить резервное копирование БД.

После этого необходимо выполнить комплекс мероприятий по устранению причины перехода системы в аварийный режим.

А.4.1.1.5 Требования по диагностированию системы

Заказчик требования по диагностированию системы не указывал.

А.4.1.1.6 Перспективы развития, модернизации системы

ИИСППР «Агроном» должна реализовывать возможность дальнейшей модернизации как программного обеспечения.

Также необходимо предусмотреть возможность увеличения производительности системы путем её масштабирования и возможность увеличения базы данных.

А.4.1.2 Требования к численности и квалификации персонала системы

Для эксплуатации ИСППР «Сельское хозяйство» определены следующие роли:

- администратор баз данных;
- пользователь.

Основными обязанностями администратора баз данных являются:

- установка, модернизация, настройка параметров программного обеспечения СУБД;
- оптимизация прикладных баз данных по времени отклика, скорости доступа к данным;
- разработка, управление и реализация эффективной политики доступа к информации, хранящейся в прикладных базах данных;

Продолжение приложения А

- ввод дополнительных данных в БД.

Администратор баз данных должен обладать высоким уровнем квалификации и практическим опытом выполнения работ по установке, настройке и администрированию используемых в ИСПР СУБД.

Пользователи системы должны иметь опыт работы с персональным компьютером на базе операционных систем Microsoft Windows на уровне квалифицированного пользователя и свободно осуществлять базовые операции в стандартных Windows.

Рекомендуемая численность для эксплуатации ИИСППР «Агроном»:

- администратор - 1 штатная единица;
- пользователь - число штатных единиц определяется структурой предприятия.

А.4.1.3 Показатели назначения

ИИСППР «Агроном» должны обеспечивать возможность исторического хранения данных с глубиной не менее 10 лет.

Система должна обеспечивать:

- для операций навигации по экранным формам системы - не более 10 сек;
- для операций формирования диаграмм и плана - не более 40 сек.

Время формирования аналитических отчетов определяется их сложностью и может занимать продолжительное время.

Система должна предусматривать возможность масштабирования по производительности и объему обрабатываемой информации без модификации ее программного обеспечения путем модернизации используемых технических средств.

А.4.1.4 Требования к надежности

Система должна сохранять работоспособность и обеспечивать восстановление своих функций при возникновении следующих внештатных ситуаций:

- при сбоях в системе электроснабжения аппаратной части, приводящих к перезагрузке ОС, восстановление программы должно происходить после перезапуска ОС и запуска исполняемого файла системы;
- при ошибках в работе аппаратных средств (кроме носителей данных и программ) восстановление функции системы возлагается на ОС;
- при ошибках, связанных с программным обеспечением (ОС и драйверы устройств), восстановление работоспособности возлагается на ОС;
- для защиты данных в БД необходимо провести мероприятия по защите, установить пароль, ключ доступа и т.д;

Продолжение приложения А

- для защиты программного обеспечения от перехода в аварийный режим необходимо вести слежение за интернет ресурсами;
- для защиты аппаратуры от бросков напряжения и коммутационных помех должны применяться сетевые фильтры.

А.4.1.5 Требования к безопасности

Все внешние элементы технических средств системы, находящиеся под напряжением, должны иметь защиту от случайного прикосновения, а сами технические средства иметь зануление или защитное заземление в соответствии с ГОСТ 12.1.030-81 и ПУЭ.

Система электропитания должна обеспечивать защитное отключение при перегрузках и коротких замыканиях в цепях нагрузки, а также аварийное ручное отключение.

Общие требования пожарной безопасности должны соответствовать нормам на бытовое электрооборудование. В случае возгорания не должно выделяться ядовитых газов и дымов. После снятия электропитания должно быть допустимо применение любых средств пожаротушения.

Факторы, оказывающие вредные воздействия на здоровье со стороны всех элементов системы (в том числе инфракрасное, ультрафиолетовое, рентгеновское и электромагнитное излучения, вибрация, шум, электростатические поля, ультразвук строчной частоты и т.д.), не должны превышать действующих норм.

А.4.1.6 Требования к эргономике и технической эстетике

Взаимодействие пользователей с прикладным программным обеспечением, входящим в состав системы должно осуществляться посредством визуального графического интерфейса (GUI). Интерфейс системы должен быть понятным и удобным, не должен быть перегружен графическими элементами и должен обеспечивать быстрое отображение экранных форм. Навигационные элементы должны быть выполнены в удобной для пользователя форме. Средства редактирования информации должны удовлетворять принятым соглашениям в части использования функциональных клавиш, режимов работы, поиска, использования оконной системы. Ввод-вывод данных системы, прием управляющих команд и отображение результатов их исполнения должны выполняться в интерактивном режиме. Интерфейс должен соответствовать современным эргономическим требованиям и обеспечивать удобный доступ к основным функциям и операциям системы.

Интерфейс должен быть рассчитан на преимущественное использование манипулятора типа "мышь", то есть управление системой должно

Продолжение приложения А

осуществляться с помощью набора экранных меню, кнопок, значков и т. п. элементов. Клавиатурный режим ввода должен использоваться главным образом при заполнении и/или редактировании текстовых и числовых полей экранных форм. Система должна иметь всплывающие подсказки для неопытного пользователя и возможность отключения этих подсказок для опытного пользователя.

Система должна обеспечивать корректную обработку аварийных ситуаций, вызванных неверными действиями пользователей, неверным форматом или недопустимыми значениями входных данных. В указанных случаях система должна выдавать пользователю соответствующие сообщения, после чего возвращаться в рабочее состояние, предшествовавшее неверной (недопустимой) команде или некорректному вводу данных.

Система должна соответствовать требованиям эргономики и профессиональной медицины при условии комплектования высококачественным оборудованием (ПЭВМ, монитор и прочее оборудование), имеющим необходимые сертификаты соответствия и безопасности ГОСТов РК.

А.4.1.7 Требования к эксплуатации, техническому обслуживанию, ремонту и хранению компонентов системы

Техническая и физическая защита аппаратных компонентов системы, носителей данных, бесперебойное энергоснабжение, резервирование ресурсов, текущее обслуживание реализуется техническими и организационными средствами, предусмотренными в ИТ инфраструктуре Заказчика.

Для нормальной эксплуатации разрабатываемой системы должно быть обеспечено бесперебойное питание ПК. При эксплуатации система должна быть обеспечена соответствующая стандартам хранения носителей и эксплуатации ПК температура и влажность воздуха.

Периодическое техническое обслуживание используемого ПК должно проводиться в соответствии с требованиями технической документации изготовителей, но не реже одного раза в год.

Периодическое техническое обслуживание и тестирование ПК должны включать в себя обслуживание и тестирование ПК, кабельной системы, устройств бесперебойного питания.

В процессе проведения периодического технического обслуживания должны проводиться внешний и внутренний осмотр и чистка ПК, проверка контактных соединений, проверка параметров настроек работоспособности ПК.

Размещение ПК должно исключать возможность бесконтрольного проникновения в них посторонних лиц и обеспечивать сохранность

Продолжение приложения А

находящихся в этих помещениях конфиденциальных документов и технических средств.

Размещение ПК должно соответствовать требованиям техники безопасности, санитарным нормам и требованиям пожарной безопасности.

Все пользователи системы должны соблюдать правила эксплуатации электронной вычислительной техники.

Квалификация персонала и его подготовка должны соответствовать технической документации.

А.4.1.8 Требования к защите информации от несанкционированного доступа

ИИСППР должна обеспечивать защиту от несанкционированного доступа (НСД).

Компоненты подсистемы защиты от НСД должны обеспечивать идентификацию пользователя.

Защищённая часть системы должна использовать "слепые" пароли (при наборе пароля его символы не показываются на экране либо заменяются одним типом символов; количество символов не соответствует длине пароля).

А.4.1.9 Требования по сохранности информации при авариях

Программное обеспечение ИИСППР «Агроном» должно восстанавливать свое функционирование при корректном перезапуске аппаратных средств. Должна быть предусмотрена возможность организации резервного копирования данных СУБД.

А.4.1.10 Требования к защите от влияния внешних воздействий

Помещение должно соответствовать санитарно-гигиеническим нормам.

А.4.1.11 Требования к патентной чистоте

Установка системы в целом, как и установка отдельных частей системы не должна предъявлять дополнительных требований к покупке лицензий на программное обеспечение сторонних производителей.

А.4.1.12 Требования по стандартизации и унификации

Экранные формы должны проектироваться с учетом требований унификации:

Продолжение приложения А

- все экранные формы пользовательского интерфейса должны быть выполнены в едином графическом дизайне, с одинаковым расположением основных элементов управления и навигации;
- для обозначения сходных операций должны использоваться сходные графические значки, кнопки и другие управляющие (навигационные) элементы.

А.4.2 Требования к функциям (задачам), выполняемым системой

А.4.2.1 Задача хранения данных

Подсистема хранения данных должна осуществлять хранение оперативных данных системы, данных для формирования сравнительных диаграмм и для разработки плана на предыдущий год.

База данных должна быть защищенной паролем.

А.4.2.2 Подсистема калькуляция данных

Подсистема должна решать задачу обеспечения информационными данными, которыми обмениваются отдельные компоненты системы между собой. В число функций подсистемы должны быть включены функции подсчета количества саженцев и семян по площади гектара, подсчет количества определенного типа удобрения требуемого посадочной культурой, учитывая вид и сорт самой культуры.

Подсистема должна предоставлять пользователю удобные инструменты для реализации данной задачи.

Функция калькуляция данных должна обладать следующей основной функциональностью:

- данные о виде культуры, сорт;
- база знаний об удобрениях;
- информация о площади крестьянского хозяйства;
- данные о видах удобрения.

А.4.2.3 Подсистема аналитический вывод прибыльности агрокультуры с учетом метеорологических и агрономических условий.

Данная подсистема должна выводить результаты прибыльности, урожайности агрокультуры с учетом метеорологических условий, то есть погодные данные, которые непременно будут поступать с интернета и агрономических условий, которая содержится в базе знаний. Расчеты будут проводиться на основании теории игр и с периодичностью месяц. Фишкой данной подсистемы является картография, то есть визуализация данных по областям РК. Данная подсистема требует соединения с интернет. При

Продолжение приложения А

переходе в аварийный режим, первым делом необходимо проверить данный модуль.

А.4.2.4 Подсистема сравнительный анализ и построение диаграммы по убыткам и прибыли

Данная подсистема поможет пользователю визуально определить, какой вид культуры прибыльнее выращивать при этом вложив наименьшие денежные ресурсы. Для реализации данной подсистемы необходимо:

- бухгалтерские отчеты предыдущих лет.

А.4.2.5 Разработка технического плана на предыдущий год

Подсистема подготавливает документ в виде плана. Данный документ должен быть оформлен по всем стандартам и должен иметь выход либо на текстовый документ, либо уже в стандартной форме отправляться на печать.

А.4.3 Требования к видам обеспечения

А.4.3.1 Требования к математическому обеспечению системы

Математические методы и алгоритмы, используемые для шифрования/дешифрования данных, а также программное обеспечение, реализующее их. Основной технологией является Data mining.

А.4.3.2 Требования к информационному обеспечению системы

Данные должны содержаться в многомерном массиве, что лучше всего организует технология OLAP.

Состав, структура и способы организации данных в системе должны быть определены на этапе технического проектирования.

Хранение данных должно осуществляться на основе современных реляционных или СУБД. Для обеспечения целостности данных должны использоваться встроенные механизмы СУБД.

Технические средства, обеспечивающие хранение информации, должны использовать современные технологии, позволяющие обеспечить повышенную надежность хранения данных и оперативную замену оборудования (зеркалирование).

А.4.3.3 Требования к лингвистическому обеспечению системы

Все прикладное программное обеспечение системы для организации взаимодействия с пользователем должно использовать русский язык.

Продолжение приложения А

А.4.3.4 Требования к программному обеспечению системы

Базовой программной платформой должна являться операционная система MS Windows.

А.4.3.5 Требования к техническому обеспечению

Техническое обеспечение системы должно максимально и наиболее эффективным образом использовать существующий ПК у персонала.

А.4.3.6 Требования к метрологическому обеспечению

Под метрологическим обеспечением измерений понимается установление и применение научных и организационных основ, технических средств, правил и норм, необходимых для достижения единства и требуемой точности измерений. В данном программном обеспечении к основным метрологическим измерениям относятся:

- площадь замеряется в гектарах (га);
- количество саженцев в единице штук (шт), а семян в килограммах (Кгр);
- жидкие удобрения замеряются в кубических литрах (л);
- температура замеряется в градусах Цельсия (°С);
- денежные средства рассчитываются в тенге (тг).

А.4.3.7 Требования к организационному обеспечению

Организационное обеспечение системы должно быть достаточным для эффективного выполнения персоналом возложенных на него обязанностей. Заказчиком должны быть определены должностные лица, ответственные за:

- пользование основными пользовательскими функциями;
- администрирование ИСПР.

К работе с системой должны допускаться сотрудники, имеющие навыки работы на персональном компьютере, ознакомленные с правилами эксплуатации и прошедшие обучение работе с системой.

А.4.3.8 Требования к методическому обеспечению

В состав методического обеспечения системы должны входить законодательные акты, стандарты, нормативы, инструкции по работе с системой.

Продолжение приложения А

А.4.3.9 Состав и содержание работ по созданию (развитию) системы

Создание ИИСППР состоит из 4 стадий.

Стадия 1 – стадия технического задания. Сроки: 01.09.13 – 01.11.13.

Данная стадия состоит из следующих этапов:

- подготовительная работа – обследование и анализирование объекта, выбор модели для разработки;
- анализ требований к системе, анализ функций возможности системы, требование к интерфейсу;
- проектирование архитектуры системы.

Стадия 2 – стадия прототипирования (эскизного проекта). Сроки: 01.11.13 – 01.01.14.

Данная стадия состоит из следующих этапов:

- анализ, требование к программному обеспечению;
- проектирование архитектуры программного обеспечения;
- детальное проектирование программирования;
- выбор технологий.

Стадия 3 – стадия технический проект. Сроки: 01.01.14 – 01.04.14.

Данная стадия состоит из следующих этапов:

- кодирование и тестирование ПО;
- интеграция ПО (сборка всех компонентов);
- квалификационное тестирование системы.

Стадия 4 – стадия рабочий проект или же сдача проекта. Сроки 01.04.14 – 20.05.14.

Стадия состоит из этапов установка и приемка ПО.

А.5 Порядок контроля и приемки системы

А.5.1 Виды, состав, объем и методы испытаний системы

Виды, состав, объем, и методы испытаний ИСППР «Сельское хозяйство» должны быть изложены в программе и методике испытаний.

А.5.2 Общие требования к приемке работ по стадиям

Сдача - приёмка работ производится поэтапно, в соответствии с рабочей программой и календарным планом.

Сдача-приемка осуществляется комиссией, в состав которой входят представители Заказчика и Исполнителя.

Все создаваемые в рамках настоящей работы программные изделия (за исключением покупных) передаются Заказчику.

Продолжение приложения А

А.5.3 Статус приемочной комиссии

Статус приемочной комиссии определяется Заказчиком до проведения испытаний.

А.6 Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу системы в действие

В ходе выполнения проекта на объекте автоматизации требуется выполнить работы по подготовке к вводу системы в действие. При подготовке к вводу в эксплуатацию ИСПР «Сельское хозяйство» Заказчик должен обеспечить выполнение следующих работ:

Определить ответственных должностных лиц, ответственных за внедрение и проведение опытной эксплуатации ИСПР «Сельское хозяйство»;

Обеспечить присутствие пользователей на обучении работе с системой, проводимом Исполнителем;

Совместно с Исполнителем подготовить план развертывания системы на технических средствах Заказчика;

Провести опытную эксплуатацию ИИСПР «Сельское хозяйство».

А.7 Требования к документированию

Данный проект сопровождается ТЗ и документацией на технический проект.

В техническом задании расписываются:

- основные цели, задачи, сроки и этапы разработки;
- список основных функций и требований;
- список функций интерфейса.

Документация на технический проект является инструкцией по использованию данного ПО. В данной документации будут расписываться:

- условия работы ПО;
- установка ПО;
- использование ПО, попутное описание основных функций и функций интерфейса;
- устранение проблем при переходе в аварийный режим.

Для системы на различных стадиях создания должны быть выпущены следующие документы из числа предусмотренных в ГОСТ 34.201-«Информационная технология. Комплекс стандартов на автоматизированные системы».

Продолжение приложения А

А.8 Источники разработки

Документы и информационные материалы (технико-экономическое обоснование, отчеты о законченных научно-исследовательских работах, информационные материалы на отечественные, зарубежные системы-аналоги и др.), на основании которых разрабатывалось ТЗ и которые должны быть использованы при создании системы.

Технико – экономическое обоснование. Данный документ содержит, финансовое описание системы, в котором содержится перечень используемых ресурсов и их ценовое описание. Стоимость системы, подсчет рентабельности. Минимизация используемых ресурсов для получения наиболее максимальной прибыли.

Данная система должна разрабатываться на основании ТК 34 по стандартизации Информационные технологии. Номер приказа и дата утверждения: от 27.07.01 г. № 274.

Приложение Б

Листинг программы

```
Program.cs
using System;
using System.Collections.Generic;
using System.Linq;
using System.Windows.Forms;

namespace BD
{
    static class Program
    {
        /// <summary>
        /// Главная точка входа для приложения.
        /// </summary>
        [STAThread]
        static void Main()
        {
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new Main());
        }
    }

    static class Vhod
    {
        public static bool Data { get ; set; }

        public static int Admin { get; set; }
    }
}
DataBase.cs

using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Collections;
using System.Data.SqlClient;
using System.Data.Common;

namespace BD
{
```

Продолжение приложения Б

```
class DataBase
{
    // Подключение к БД (string y – имя БД) и выполнение команды(string
x).
    static public ArrayList GetInfo(string x, string y)
    {
        string ConnectionString = @"Data
Source=.\SQLEXPRESS;AttachDbFilename=" + Environment.CurrentDirectory +
@"\" + y + @";Integrated Security=True;User Instance=True";

        ArrayList AllInfo = new ArrayList();
        SqlConnection con = new SqlConnection(ConnectionString);
        SqlCommand com = new SqlCommand(x, con);

        try
        {
            con.Open();
            SqlDataReader DatReader = com.ExecuteReader();
            if (DatReader.HasRows)
                foreach (DbDataRecord result in DatReader)
                    AllInfo.Add(result);
            else
            {
                con.Close();
                return null;
            }
        }
        catch { }
        finally
        {
            con.Close();
        }
        return AllInfo;
    }
}
```

Main.cs

```
using System;
```

Продолжение приложения Б

```
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.Data.SqlClient;
using System.Collections;

namespace BD
{
    public partial class Main : Form
    {
        //При загрузке Ширина экрана 500, а Высота 391
        public Main()
        {
            InitializeComponent();
            this.Width = 500;
            this.Height = 391;
        }

        //Нельзя изменять размеры окна, нельзя увеличить окно
        private void Form1_Load(object sender, EventArgs e)
        {
            this.FormBorderStyle =
System.Windows.Forms.FormBorderStyle.Fixed3D;
            this.MaximizeBox = false;
        }

        //Создаёт новое окно и открывает его. Текущее окно скрывается
        private void button1_Click(object sender, EventArgs e)
        {
            Login Log = new Login();
            Vhod.Data = false;
            Vhod.Admin = 0;
            this.Hide();
            Log.ShowDialog();
            this.Show();
        }
    }
}
```

```
}  
}
```

Login.cs

```
using System;  
using System.Collections.Generic;  
using System.ComponentModel;  
using System.Data;  
using System.Drawing;  
using System.Linq;  
using System.Text;  
using System.Windows.Forms;  
using System.Data.SqlClient;  
using System.Collections;  
  
namespace BD  
{  
    public partial class Login : Form  
    {  
        string x;  
        ArrayList Table1;  
  
        public Login()  
        {  
  
            InitializeComponent();  
        }  
  
        //label делаем прозрачными  
        private void registr_Load(object sender, EventArgs e)  
        {  
            label1.BackColor = Color.Transparent;  
            label2.BackColor = Color.Transparent;  
            this.FormBorderStyle =  
System.Windows.Forms.FormBorderStyle.Fixed3D;  
            this.MaximizeBox = false;  
            Pass_tb.PasswordChar = '*';  
  
        }  
  
        //открываем окно регистрации  
        private void button1_Click_1(object sender, EventArgs e)  
        {  
            registr registr = new registr();
```

Продолжение приложения Б

```
registr.ShowDialog();

    }

    //Проверка входа.
    private void button2_Click(object sender, EventArgs e)
    {
        if (Pass_tb.Text != "" && Log_tb.Text != "")
        {
            x = "SELECT name,password FROM enter where name = '" +
Log_tb.Text + "' AND password = '" + Pass_tb.Text + "'";
            Table1 = DataBase.GetInfo(x, "Database1.mdf");

            try
            {
                if (Table1.Count == 1)
                {
                    x = "UPDATE enter SET zahodil=zahodil+1 where name='" +
Log_tb.Text + "'";
                    DataBase.GetInfo(x, "Database1.mdf");

                    if (Log_tb.Text == "admin")
                    {
                        Adminka adm = new Adminka();
                        this.Hide();
                        adm.ShowDialog();
                    }
                    else
                    {
                        second sec = new second();
                        this.Hide();
                        sec.ShowDialog();
                    }

                    Log_tb.Text = "";
                    Pass_tb.Text = "";
                }
                else MessageBox.Show("Неправильно введён логин или пароль");
            }

            catch
            {
```

```
        finally
        {
            }
        }
    }

private void Login_FormClosing(object sender, FormClosingEventArgs e)
{
    }
}
}
Registr.cs
```

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.Collections;
```

```
namespace BD
```

```
{
    public partial class registr : Form
    {
        string sol;
        ArrayList ArrayL = new ArrayList();

        public registr()
        {
            InitializeComponent();
        }

        //Регистрация нового пользователя
        private void button1_Click(object sender, EventArgs e)
        {
            if (Pass_tb.Text != Pass1_tb.Text)
```

Продолжение приложения Б

```
{
    MessageBox.Show("Пароли не совпадают");
    return;
}
if (Pass_tb.Text != "" && Log_tb.Text != "" && FIO_tb.Text != "")
{
    try
    {
        sol = "SELECT Name FROM Enter where name = " +
Log_tb.Text.ToString() + """;
        ArrayL = DataBase.GetInfo(sol, "Database1.mdf");

        if (ArrayL.Count > 0)
        {
            MessageBox.Show("Данный Логин занят");
        }

    }

    catch (NullReferenceException)
    {
        DataBase.GetInfo("INSERT INTO enter (FIO, Name, password)
VALUES (" + FIO_tb.Text.ToString() + ", " + Log_tb.Text.ToString() + ", " +
Pass_tb.Text.ToString() + """, "Database1.mdf");
        MessageBox.Show("Успешная регистрация");
        this.Close();
    }

}

}

//Окно закрывается.
private void registr_FormClosing(object sender, FormClosingEventArgs e)
{
    this.Dispose();
}

//Нельзя изменять размеры окна, нельзя увеличить окно. label1-4
становятся прозрачными.
private void registr_Load(object sender, EventArgs e)
{
```


Продолжение приложения Б

```
this.FormBorderStyle = System.Windows.Forms.FormBorderStyle.Fixed3D;
    this.MaximizeBox = false;
    label1.BackColor = Color.Transparent;
    label2.BackColor = Color.Transparent;
    label3.BackColor = Color.Transparent;
    label4.BackColor = Color.Transparent;
    Pass_tb.PasswordChar = '*';
    Pass1_tb.PasswordChar = '*';
}

//Вызывается при изменении пароля
private void Pass1_tb_TextChanged(object sender, EventArgs e)
{
    toolTip1.SetToolTip(Pass1_tb, "Пароли должны совпадать");
    toolTip1.IsBalloon = true;
}
}
```

adminka.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.Data.SqlClient;
using System.Collections;

namespace BD
{
    public partial class Adminka : Form
    {
        int ID;

        public Adminka()
        {
            InitializeComponent();
        }
    }
}
```

Продолжение приложения Б

```
//Устанавливает все label пустыми.
private void Adminka_Load(object sender, EventArgs e)
{
    label3.Text = "";
    label5.Text = "";
    label7.Text = "";

    comboBox1.SelectedIndex = 0;
    comboBox2.SelectedIndex = 0;
    update_Click(null, null);
}

//Выбираем поле, по которому будет сортировка
private void update_Click(object sender, EventArgs e)
{
    string order = "id";
    switch (comboBox1.SelectedIndex)
    {
        case 0:
            order = "FIO";
            break;
        case 1:
            order = "name";
            break;
        case 2:
            order = "password";
            break;
        case 3:
            order = "Zahodil";
            break;
        default:
            order = "id";
            break;
    }

    string Command = "SELECT * FROM Enter order by " + order;
    if (button1.Text == "по возрастанию")
        Command += " DESC";
    ArrayList Table = DataBase.GetInfo(Command, "Database1.mdf");
    dataGridView1.DataSource = Table;
}
```

Продолжение приложения Б

```
dataGridView1.RowHeadersVisible = false;
    foreach (DataGridViewColumn column in dataGridView1.Columns)
    {
        column.SortMode = DataGridViewColumnSortMode.NotSortable;
    }

    dataGridView1.AutoSizeColumnsMode =
DataGridViewAutoSizeColumnsMode.Fill;
}

//Поиск по введенному значению
private void textBox1_TextChanged(object sender, EventArgs e)
{
    string Command;
    switch (comboBox2.SelectedIndex)
    {
        case 0:
            Command = "SEIECT * FROM Enter where FIO LIKE '%" +
textBox1.Text + "%'";
            break;
        case 1:
            Command = "SEIECT * FROM Enter where name LIKE '%" +
textBox1.Text + "%'";
            break;
        case 2:
            Command = "SEIECT * FROM Enter where password LIKE '%" +
textBox1.Text + "%'";
            break;
        case 3:
            Command = "SEIECT * FROM Enter where zahodil LIKE '%" +
textBox1.Text + "%'";
            break;
        default: Command = "SEIECT * FROM Enter where id LIKE '%" +
textBox1.Text + "%'";
            break;
    }
    ArrayList Table = DataBase.GetInfo(Command, "Database1.mdf");
    dataGridView1.DataSource = Table;

    dataGridView1.RowHeadersVisible = false;
```

Продолжение приложения Б

```
foreach (DataGridViewColumn column in dataGridView1.Columns)
    {
        column.SortMode = DataGridViewColumnSortMode.NotSortable;
    }

    dataGridView1.AutoSizeColumnsMode =
DataGridViewAutoSizeColumnsMode.Fill;
}

//Кликаем по таблице и информация о пользователе выводится в
отдельное место
private void dataGridView1_CellClick(object sender,
DataGridViewCellEventArgs e)
{
    if (e.RowIndex >= 0)
    {

        label3.Text =
dataGridView1.Rows[e.RowIndex].Cells[1].Value.ToString();
        label5.Text =
dataGridView1.Rows[e.RowIndex].Cells[2].Value.ToString();
        label7.Text =
dataGridView1.Rows[e.RowIndex].Cells[3].Value.ToString();
        ID =
Int32.Parse(dataGridView1.Rows[e.RowIndex].Cells[0].Value.ToString());
    }
}

//Удаляем строку
private void Delete_Click(object sender, EventArgs e)
{
    try
    {
        if (ID != 1)
        {
            string Command = "DELETE FROM Enter where id = " + ID + "";
            DataBase.GetInfo(Command, "Database1.mdf");
            update_Click(null, null);
        }
    }
    catch { }
```

Продолжение приложения Б

```
//Сортировка по убыванию, возрастанию.
private void button1_Click(object sender, EventArgs e)
{
    if (button1.Text == "по возрастанию")
        button1.Text = "по убыванию";
    else button1.Text = "по возрастанию";
    update_Click(null, null);
}

//Обновляет таблицу
private void comboBox1_SelectedIndexChanged(object sender, EventArgs e)
{
    update_Click(null, null);
}

private void comboBox2_SelectedIndexChanged(object sender, EventArgs e)
{
}
}
}
second.cs
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;

namespace BD
{
    public partial class second : Form
    {
        public second()
        {
            InitializeComponent();
        }

        private void button1_Click(object sender, EventArgs e)
        {
```

Продолжение приложения Б

//Окно нельзя изменить, свернуть

```
private void second_Load(object sender, EventArgs e)
{
    this.FormBorderStyle =
System.Windows.Forms.FormBorderStyle.Fixed3D;
    this.MaximizeBox = false;
    dataGridView1.RowHeadersVisible = false;
}

private void button3_Click(object sender, EventArgs e)
{
}

private void button4_Click(object sender, EventArgs e)
{
}

//закрывает форму
private void button11_Click(object sender, EventArgs e)
{
    this.Close();
}

private void second_FormClosed(object sender, FormClosedEventArgs e)
{
    this.Dispose();
}

private void выходToolStripMenuItem_Click(object sender, EventArgs e)
{
    this.Close();
}

private void button6_Click(object sender, EventArgs e)
{
}

//открывает новое окно
private void DM_Click(object sender, EventArgs e)
```

Продолжение приложения Б

```
{
    Romich Rom = new Romich();
    Rom.Show();
}

//открывает новое окно
private void button2_Click(object sender, EventArgs e)
{
    DataMining DM = new DataMining();
    DM.Show();
}

//открывает новое окно
private void calc_Click(object sender, EventArgs e)
{
    Calc calculyator = new Calc();
    calculyator.Show();
}

//открывает новое окно
private void button5_Click(object sender, EventArgs e)
{
    Plan Pl = new Plan();
    Pl.Show();
}

//открывает новое окно
private void schet_Click(object sender, EventArgs e)
{
    schet sch = new schet();
    sch.Show();
}
}
```

calc.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
```

Продолжение приложения Б

```
using System.Text;
using System.Windows.Forms;
using System.Collections;
using System.Data.Common;
using BD;

namespace BD
{
    public partial class Calc : Form
    {
        ArrayList al = new ArrayList();

        public Calc()
        {
            InitializeComponent();

            //закрытие формы
            private void button1_Click(object sender, EventArgs e)
            {
                this.Close();
            }

            //label прозрачные. форму нельзя изменить, развернуть.
            private void Calc_Load(object sender, EventArgs e)
            {
                this.FormBorderStyle =
System.Windows.Forms.FormBorderStyle.Fixed3D;
                this.MaximizeBox = false;
                label1.BackColor = Color.Transparent;
                label2.BackColor = Color.Transparent;
                label3.BackColor = Color.Transparent;
                label4.BackColor = Color.Transparent;
                label5.BackColor = Color.Transparent;
                label6.BackColor = Color.Transparent;
                label7.BackColor = Color.Transparent;
                label8.BackColor = Color.Transparent;
                label9.BackColor = Color.Transparent;

                al = BD.DataBase.GetInfo("Select * FROM kind_of_culture", "one.mdf");
            }
        }
    }
}
```


Продолжение приложения Б

//достаём данные из таблицы.

```
private void comboBox2_SelectedIndexChanged(object sender, EventArgs e)
{
    try
    {
        comboBox4.Items.Clear();
        foreach (DbDataRecord result in BD.DataBase.GetInfo("Select NameST,
NameNS FROM sort_of_culture where NameKD = '" + comboBox2.Text + "'",
"one.mdf"))
            comboBox4.Items.Add(result.GetString(0));
    }
    catch { }
}
```

//достаём данные из таблицы

```
private void comboBox2_DropDown(object sender, EventArgs e)
{
    try
    {
        comboBox2.Items.Clear();
        foreach (DbDataRecord result in BD.DataBase.GetInfo("Select * FROM
kind_of_culture", "one.mdf"))
            comboBox2.Items.Add(result.GetString(0));

        comboBox4.Text = null;
    }
    catch { }
}
}
```

Romich.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
using System.Collections;
using Conn;
```

Продолжение приложения Б

```
namespace BD
{
    public partial class Romich : Form
    {
        public Romich()
        {
            InitializeComponent();
        }

        //запускаем новую форму.
        private void button2_Click(object sender, EventArgs e)
        {
            Romich2 Romanich2 = new Romich2();
            Romanich2.Show();
        }

        //нельзя развернуть и изменить размер окна
        private void Romich_Load(object sender, EventArgs e)
        {
            this.FormBorderStyle =
System.Windows.Forms.FormBorderStyle.Fixed3D;
            this.MaximizeBox = false;
        }
    }
}
```

```
Romich2.cs
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
```

```
namespace BD
{
    public partial class Romich2 : Form
    {
        public Romich2()
        {
```

Продолжение приложения Б

```
InitializeComponent();
    }
    private void richTextBox1_TextChanged(object sender, EventArgs e)
    {

    }

    //Закрытие формы
    private void button1_Click(object sender, EventArgs e)
    {
        this.Close();
    }

    //Label прозрачен. Нельзя изменять размер формы и разворачивать
    private void Romich2_Load(object sender, EventArgs e)
    {
        label1.BackColor = Color.Transparent;
        this.FormBorderStyle =
System.Windows.Forms.FormBorderStyle.Fixed3D;
        this.MaximizeBox = false;

    }
}
}
```

DataMining.cs

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
namespace BD
{
    public partial class DataMining : Form
    {
        public DataMining()
        {
            InitializeComponent();
        }
    }
}
```

Продолжение приложения Б

```
//закрытие формы.
private void button7_Click(object sender, EventArgs e)
{
    this.Close();
}
//нельзя изменить размер окна и развернуть
private void DataMining_Load(object sender, EventArgs e)
{
    this.FormBorderStyle =
System.Windows.Forms.FormBorderStyle.Fixed3D;
    this.MaximizeBox = false;
}
//открытие новой формы
private void button3_Click(object sender, EventArgs e)
{
    Karta Karta = new Karta();
    Karta.Show();
}
}
Karta.cs
```

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Windows.Forms;
```

```
namespace BD
{
    public partial class Karta : Form
    {
        public Karta()
        {
            InitializeComponent();
        }

        private void Karta_Load(object sender, EventArgs e)
        {
```

```
}  
}  
Plan.cs
```

```
using System;  
using System.Collections.Generic;  
using System.ComponentModel;  
using System.Data;  
using System.Drawing;  
using System.Linq;  
using System.Text;  
using System.Windows.Forms;
```

```
namespace BD  
{  
    public partial class Plan : Form  
    {  
        public Plan()  
        {  
            InitializeComponent();  
        }  
  
        private void Plan_Load(object sender, EventArgs e)  
        {  
  
        }  
    }  
}  
schet.cs
```

```
using System;  
using System.Collections.Generic;  
using System.ComponentModel;  
using System.Data;  
using System.Drawing;  
using System.Linq;  
using System.Text;  
using System.Windows.Forms;  
using System.CodeDom;  
using System.Collections;
```

```
namespace BD  
{
```

Продолжение приложения Б

```
public partial class schet : Form
{
    int ksto, kstr;
    //float[] massR;
    //float[] massC;

    public schet()
    {
        InitializeComponent();
    }

    //форма не изменяет размеры, её нельзя развернуть.
    private void schet_Load(object sender, EventArgs e)
    {
        this.FormBorderStyle =
System.Windows.Forms.FormBorderStyle.Fixed3D;
        this.MaximizeBox = false;
        dataGridView1.RowHeadersVisible = false;
        dataGridView1.ColumnHeadersVisible = false;
        dataGridView1.AllowUserToAddRows = false;

        label3.Text = "Агроном' рекомендует:";

        richTextBox1.Font = new Font("Times New Roman", 14);
        richTextBox1.Text = "1) арендовать площадь посева в размере 45 га;";
        richTextBox1.Text += Environment.NewLine + "2) под первый сорт
картофеля выделить площадь посева в размере 22га;";
        richTextBox1.Text += Environment.NewLine + "3) под второй сорт
картофеля выделить площадь посева в размере 18га;";
        richTextBox1.Text += Environment.NewLine + "4) под третий сорт
картофеля выделить площадь посева в размере 5га;";
        richTextBox1.Text += Environment.NewLine + "Максимальная прибыль
составит 31,5 у.е";
    }

    //проверка на наличие значений
    private void sto_TextChanged(object sender, EventArgs e)
    {
        if (sto.Text != "" && str.Text != "")
            matrica();
    }
}
```

Продолжение приложения Б

```
//создаём таблицу
private void matrica()
{
    this.dataGridView1.Rows.Clear();
    int count = this.dataGridView1.Columns.Count;
    for (int i = 0; i < count; i++)
        this.dataGridView1.Columns.RemoveAt(0);
    kstr = Int32.Parse(str.Text);
    ksto = Int32.Parse(sto.Text);
    dataGridView1.ColumnCount = kstr;
    dataGridView1.Rows.Add(ksto); //+1
}

//находим минимальное и максимальное в таблице и цену игры
private void Maxmin()
{
    float[] stoimost = new float[] { 1, 2, 3 };
    float maxR, maxC, P1, P2, CenaIgr, Q1, Q2;
    int i, j;
    float[,] massiv = new float[ksto, kstr];
    float[] massR = new float[ksto];
    float[] massC = new float[kstr];
    float[] min = new float[kstr];
    float[,] zamena = new float[ksto, kstr];

    for (i = 0; i < ksto; i++)
        for (j = 0; j < kstr; j++)
        {
            massiv[i, j] = stoimost[j] *
Single.Parse(dataGridView1.Rows[i].Cells[j].Value.ToString());
        }

    for (i = 0; i < ksto; i++)
    {
        maxR = massiv[i, 0];
//Single.Parse(dataGridView1.Rows[i].Cells[0].Value.ToString());
        for (j = 0; j < kstr; j++)
        {
            //Single.Parse(dataGridView1.Rows[i].Cells[j].Value.ToString())
            //Single.Parse(dataGridView1.Rows[i].Cells[j].Value.ToString)
            maxR = maxR < massiv[i, j] ? maxR : massiv[i, j];
        }
    }
}
```

Продолжение приложения Б

```
};
massR[i] = maxR;

    MessageBox.Show("Минимальное в строке " + Convert.ToString(i + 1)
+ " это " + Convert.ToString(maxR));
    }

for (i = 0; i < kstr; i++)
{
    maxC = massiv[0, i];
    //Single.Parse(dataGridView1.Rows[0].Cells[i].Value.ToString());
    for (j = 0; j < ksto; j++)
    {
        //Single.Parse(dataGridView1.Rows[j].Cells[i].Value.ToString())
        //Single.Parse(dataGridView1.Rows[j].Cells[i].Value.ToString());
        maxC = maxC > massiv[j, i] ? maxC : massiv[j, i];
    };
    massC[i] = maxC;
    MessageBox.Show("Максимальное в столбце " + Convert.ToString(i +
1) + " это " + Convert.ToString(maxC));
    }

    //РЕШЕНИЕ МАТРИЦЫ РАЗМЕРОМ 2 НА 2 !

    {
        P1 = ( massiv[1,1]-massiv[1,0] )/ (massiv[0,0] - massiv[1,0] -
massiv[0,1] + massiv[1,1]);
        P2=1-P1;

        CenaIgri = massiv[0, 0] * P1 + massiv[1, 0] * P2;

        Q1 = ( CenaIgri - massiv[0,1] )/ (massiv[0,0] - massiv[0,1] );
        Q2 = 1 - Q1;

        float qwerty = massC.Min();
        float asdfg = massR.Max();

        /* label9.Text = Convert.ToString(P1);
        label10.Text = Convert.ToString(P2);
        label11.Text = Convert.ToString(Q1);
        label12.Text = Convert.ToString(Q2);
        niz.Text = Convert.ToString(massR.Max());
        ver.Text = Convert.ToString(massC.Min());
```


Продолжение приложения Б

```
*/
if (massC.Min() == massR.Max())
    {
        // cena.Text = Convert.ToString(massR.Max() );
    }
    //else
    //cena.Text = Convert.ToString(CenaIgri);
}

//если количество строк и столбцов больше 1, то находим минимальное и
максимальное в таблице
private void button1_Click(object sender, EventArgs e)
{
    if (Convert.ToInt32(str.Text) > 1 && Convert.ToInt32(sto.Text) > 1)
        Maxmin();
}

// вызывается при изменении текста.
private void str_TextChanged(object sender, EventArgs e)
{
    if (sto.Text != "" && str.Text != "")
        matrica();
}

//закрытие формы.
private void button3_Click(object sender, EventArgs e)
{
    this.Close();
}

}
}
```