Некоммерческое акционерное общество
«АЛМАТИНСКИЙ УНИВЕРСИТЕТ ЭНЕРГЕТИКИ И СВЯЗИ ИМЕНИ Г. ДАУКЕЕВА»
Кафедра космической инженерии

«ДОПУЩЕН К ЗАЩИТЕ»
Зав.кафедрой __доцент, Алипбаев К.А._____
(ученая степень, звание, Ф.И.О.)
_____«___» _____201___г.
(подпись)

ДИПЛОМНЫЙ ПРОЕКТ

На тему: Разработка системного проекта технологического обеспечения испытаний программных средств космического назначения
Специальность _5В074600___
Выполнил _____Утегенова Н.Д._____ Группа __КТТ-16-1
(Ф.И.О.)
Научный руководитель к.т.н., проф. Исмаил Е.Е._____
(ученая степень, звание, Ф.И.О.)
Консультанты:
по экономической части: ____профессор Ибришев Н.Н.
(ученая степень, звание, Ф.И.О.)
_____ «___» _____201___г.
(подпись)
по безопасности жизнедеятельности:__ст. преп. Тыщенко Е.М._____
(ученая степень, звание, Ф.И.О.)
_____ «___» _____201___г.
(подпись)
Нормоконтролер:_____доцент, Алипбаев К.А._____
(ученая степень, звание, Ф.И.О.)
_____ «____» _____201___г.
(подпись)
Рецензент: _____к.т.н. Инчин А.С._____
(ученая степень, звание, Ф.И.О.)
_____ «____» _____201___г.
(подпись)

Алматы, 2020

Аннотация

В дипломной работе рассматриваются аспекты и особенности разработки системного проекта технологического обеспечения испытаний программных средств космического назначения.

Целью дипломной работы является разработка системных требований и концепции построения технологического обеспечения испытаний программных средств, определяющей облик будущей системы, ее функции и структуру. Для этого были рассмотрены особенности программных средств космической техники, виды, процесс испытаний ПС и требования для их проведения.

В результате были определены перечень требований к средствам, исполнителям, их функциональная связь на основе нормативной документации и международных стандартов испытаний. Также, проведен общий расчет затрат времени, оценка ресурсов и условия процесса испытания.

Аңдатпа

Бұл дипломдық жұмыста ғарыш саласына арналған бағдарламалық құралдарды сынауды технологиялық қамтамасыз етудің жүйелік жобасын әзірлеу аспектілері мен ерекшеліктері қарастырылады.

Дипломдық жұмыстың мақсаты болашақ жүйенің бейнесін, оның функциялары мен құрылымын анықтайтын бағдарламалық құралдарды сынаудың технологиялық қамтамасыз етілуін құрастырудың жүйелік талаптары мен концепциясын әзірлеу болып табылады. Ол үшін ғарыштық техниканың бағдарламалық құралдарының ерекшеліктері, сынақ түрлері, процесі және оларды жүргізу үшін талаптар қарастырылды.

Нәтижесінде құралдарға, орындаушыларға қойылатын талаптар тізбесі, нормативтік құжаттама мен халықаралық сынақ стандарттары негізінде олардың функционалдық байланысы анықталды. Сонымен қатар, уақыт шығынының жалпы есебі, ресурстарды бағалау және сынау процесінің шарттары жүргізілді.

Abstract

The diploma project discusses aspects and features of the development of a system design of technological support for testing software for space applications.

The aim of the diploma project is the development of system requirements and the concept of building technological support for testing software, which determines the appearance of the future system, its functions and structure. To do this, we considered the features of the software for space technology, the types, the test process of the software and the requirements for their implementation.

As a result, there were determined a list of requirements for tools, performers, their functional relationship based on regulatory documentation and international test standards . Also, a general calculation of the spent time , resource estimation, and conditions of the test process was performed.

Содержание

Обозначения и сокращения 5

Введение 6

Ⅰ. АНАЛИЗ ТРЕБОВАНИЙ И ТЕХНОЛОГИИ ИСПЫТАНИЙ ПРОГРАММНЫХ СРЕДСТВ 7

1.1 Цели, задачи и особенности испытаний ПСКН 7

1.2 Виды и средства испытаний программных средств 10

1.3 Процесс испытаний программных средств и его технологическое обеспечение 18

1.4 Задача разработки системного проекта создания технологического обеспечения испытания ПСКН 22

Ⅱ. РАЗРАБОТКА КОНЦЕПЦИИ ПОСТРОЕНИЯ ТЕХНОЛОГИЧЕСКОГО ОБЕСПЕЧЕНИЯ ИСПЫТАНИЙ ПСКН 24

2.1 Требования к технологическому обеспечению испытаний ПСКН 24

2.2 Разработка функциональной модели и структуры технологического обеспечения испытаний ПСКН 37

Ⅲ. РАЗРАБОТКА РЕКОМЕНДАЦИЙ ПО РЕАЛИЗАЦИИ ПРОЕКТА СОЗДАНИЯ ТЕХНОЛОГИЧЕСКОГО ОБЕСПЕЧЕНИЯ ИСПЫТАНИЙ ПСКН 51

3.1 Анализ существующих программно-технических средств испытания ПС 51

3.2 Разработка рекомендаций по выбору отдельных элементов системы технологического обеспечения испытания ПСКН 56

Ⅳ. ЭКОНОМИЧЕСКАЯ ЧАСТЬ 59

4.1. Принципы оценки экономической эффективности испытания ПС 59

4.2. Анализ и расчет затрат на испытание ПС 61

V. САНИТАРНО-ГИГИЕНИЧЕСКАЯ ОЦЕНКА ОСВЕЩЕННОСТИ РАБОЧЕГО ПОМЕЩЕНИЯ 69
5.1 Расчет естественного освещения помещения 69
5.2. Расчет искусственного освещения помещения 72
ЗАКЛЮЧЕНИЕ 76
Список использованной литературы 78

Обозначения и сокращения

ЖЦ – жизненный цикл

КА – космический аппарат

КТ – космическая техника

ПМИ – программа испытания

ПО – программное обеспечение

ПП – программный продукт

ПС – программные средства

ПСКН – программные средства космического назначения

СИ – средства испытаний

ТЗ – техническое задание

Введение

Программные средства космического назначения (ПСКН) впервые были использованы на американском пилотируемом корабле Gemini в 1965 году. Инженеры для создания программных средств использовали такие языки программирования, как: C, C++, Java, ADA, Fortran, Lisp, Python и Assembler.

Программные средства позволяют регулировать и управлять процессами работы космической техники. Их назначение и виды варьируются в зависимости от целевого использования аппарата. NASA открыл для общего доступа каталог программного обеспечения (ПО), которое они используют для: конструирования дронов, связи между наземными и космическими станциями, обработки спутниковых данных, автоматизации работы станции и т.д.

Актуальность данной работы обусловлена необходимостью в обеспечении контроля надежности сложных систем, применяемых в ракетно-космической технике. Структурирование и детализация требований к средствам испытаний открывает возможности упрощения и увеличения точности процесса.

Цель дипломной работы заключается в разработке системного проекта технологического обеспечения испытания программных средств космического назначения.

Задачи, решаемые в рамках этой работы:

1. Исследование предметной области испытания ПСКН;
2. Анализ требований к технологическому обеспечению испытаний;
3. Разработка системного проекта технологического обеспечения испытаний;
4. Оценка реализации проекта;
5. Заключение по работе.

В качестве теоретической основы были использованы материалы лекций, учебные пособия, статьи и разработки по стандартизации и сертификации, литература по разработке программного обеспечения, а также международные и национальные стандарты.

1. АНАЛИЗ ТРЕБОВАНИЙ И ТЕХНОЛОГИИ ИСПЫТАНИЙ ПРОГРАММНЫХ СРЕДСТВ

1.1 Цели, задачи и особенности испытаний ПСКН

На сегодняшний день эталоны качества космической техники формируются Международной организацией по стандартизации (International Organization for Standardization, ISO) и Европейской кооперацией по стандартизации в области космической техники (European Cooperation for Space Standardization, ECSS). Организации позволяют контролировать уровень разработки стандартов в рамках европейской космической программы, однако, стандарты ECSS применяются и в казахстанской космической деятельности.

Чем характеризуется надежность ПС? Надежность ПС – это способность объекта безотказно выполнять определенный набор функций при условиях, заданных в течение какого-то периода времени.

Допустимого качества программных продуктов можно добиться двумя методами. Первый метод позволяет обеспечить качество путем контроля за процессом проектирования и производства ПО, предотвращая ошибки или дефекты на каждом этапе жизненного цикла. Второй метод – анализ качества уже готового программного продукта, которое удостоверяется при сертификационных испытаниях.

Испытания ПО проводятся для установления уровня качества, разрабатываемого или используемого программного продукта.

На начальном этапе исследования нас интересовали суть и особенности испытаний программных средств космического назначения.

Согласно стандартному определению, *испытания* – это экспериментальное нахождение качественных и/или количественных характеристик свойств продукта в реальной среде или ее модели.

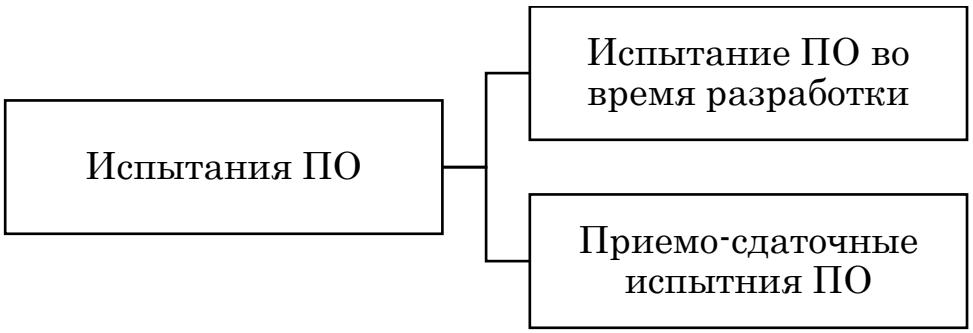


Рисунок 1.1.1 – Классификация испытаний ПО по назначению.

Особенность испытаний программных средств космического назначения заключается в том, что большинство объектов космического назначения не поддаются ремонту в процессе эксплуатации и это делает обеспечение качества и надежности более актуальным, чем в других секторах промышленности. Также расходы на миссии высоки, и их сложность возрастает по мере ее развития в течение всего жизненного цикла, т.е. чем позже обнаруживается проблема, тем больше усилий и средств понадобится для ее устранения.

Такие ПО относятся к критическим – это программное обеспечение, выполняющее критические функции, т.е. функции важные для безопасности.

Например, программное обеспечение, которое:

- непосредственно выполняет команды и осуществляет контроль за условиями или состоянием аппаратных компонентов, и его невыполнение, непоследовательное выполнение или неправильное выполнение может привести к некорректному выполнению функций управления, влияет на риск безопасности или может привести к возникновению опасных условий;

- осуществляет текущий контроль за состоянием аппаратных компонентов, результатами невыполнения, непоследовательного выполнения или неправильного выполнения, которого, могут стать данные, приводящие к ошибочным решениям операторов или вспомогательных систем, которые подвергают риску безопасности или могут привести к созданию условий возникновения рисков;

- непосредственно осуществляет управление и контроль за условиями или состоянием аппаратных компонентов и его (управление и контроль)

невыполнение, непоследовательное выполнение или выполнение, связанное с ошибками персонала, аппаратным отказом или отказом в результате воздействия окружающей среды, может быть опасным или может привести к созданию условий возникновения рисков [2].

Результатом положительных испытаний является *сертификат соответствия* – документ, изданный в соответствии с правилами системы сертификации, удостоверяющий, что обеспечивается необходимая уверенность в том, что должным образом идентифицированная продукция, процесс или услуга соответствует конкретным стандартам или другим нормативным документам.

1.2 Виды и средства испытаний программных средств

Испытание программных средств – это завершающий этап тестирования. Однако гораздо правильнее будет разделить испытание на его составляющие части (рис. 1.2.1).

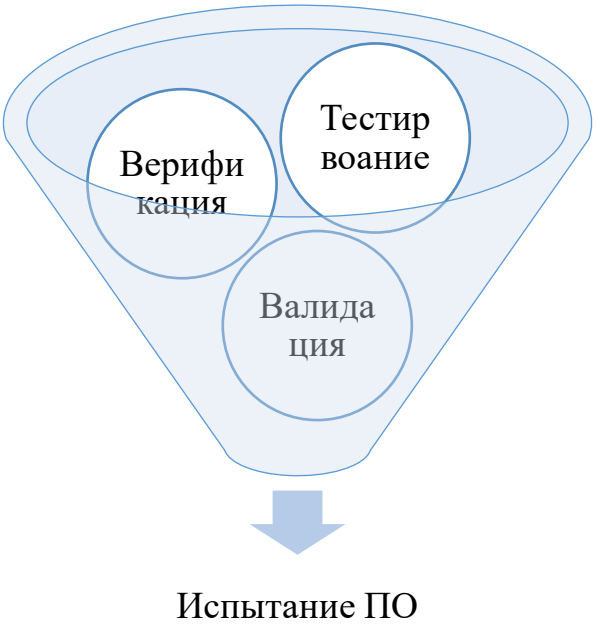


Рисунок 1.2.1 – Уровни испытаний.

Несмотря на то, что понятия «тестирование», «верификация» и «валидация» кажутся схожими по назначению, они представляют разные уровни проверки.

Тестирование программных средств направлено на доказательство наличия дефектов в ПО и обычно относится к проверке правильности реализации программы.

Целью *верификации* программных средств является установление соответствия ПО требованиям, удовлетворяет проектным спецификациям и стандартам. Процесс верификации включает в себя инспекции, тестирование кода, анализ результатов тестирования, формирование и анализ отчетов о проблемах. Процесс тестирования является составной частью процесса верификации.

Валидацией обычно называют процесс доказательства того, что ПО соответствует ожиданиям заказчика. То есть тем требованиям и желаниям, которые изначально устанавливались между исполнителем и заказчиком.

Как правило в области испытаний программных средств специалистами выделяются следующие: испытание компонентов ПС, системные, приемо-сдаточные, полевые и промышленные испытания (Рис. 1.2.2).

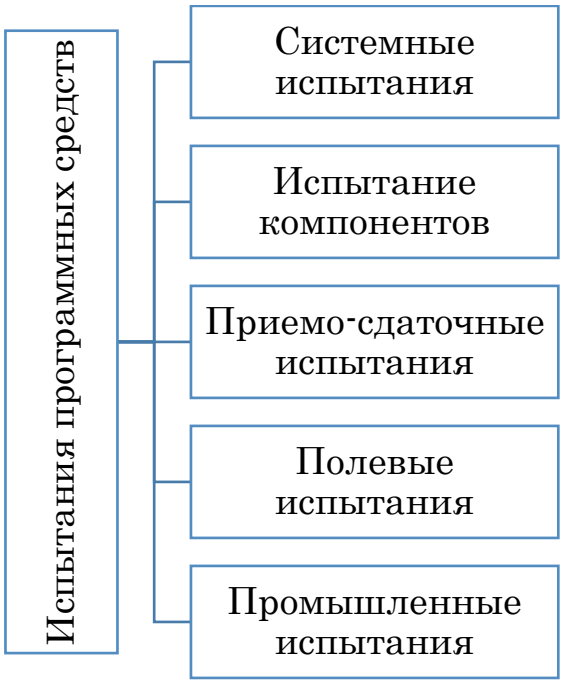


Рисунок 1.2.2 – Виды испытаний

Для более полной характеристики рассматриваемого вопроса опишем каждый вид подробнее.

Системные испытания ПС предназначены для установления соответствия полной системы программного продукта и проводятся по решению аттестационной комиссии. Обычно инициаторами являются заказчик и разработчик, редко – третья сторона.

Испытания компонентов ПС можно назвать выборочными, ведь подобные тесты проводятся для проверки отдельно взятых частей программы.

Назначение *приемо-сдаточных испытаний* понятно, такие операции проводятся для готового продукта с целью подведения итогов и принятия решения о соответствии продукта техническому заданию и другим регламентам.

Первоначально приемо-сдаточные испытания начинаются с тщательной проверки документации, результатов тестирования и отладке.

Полевыми испытаниями называются испытания, предназначенные для демонстрации разработанного ПО в реальной среде или же ее смоделированной версии. Такие испытания сложны ввиду технических возможностей и большой стоимости затрат.

Промышленные испытания проводятся пользователями продукта. Объект передается в пользование группе людей на период опытной эксплуатации.

Из практики известно, чтобы определить функционирует ли код требуется его запуск. Такой тип тестирования называют *динамическим.*

Статистический метод тестирования предполагает тестирования без запуска кода на исполнение. Тестированию подвергаются документы, сопровождающие ПО, эскизы, схемы интерфейса, параметры среды исполнения и уже готовые тестовые данные.

Существует классификация тестирования в зависимости от того, какой степени доступ имеют тестировщики. *Функциональное тестирование* (Functional Testing), которое также называют тестированием по принципу *«черного ящика»* («Black Box» Testing), заключается в проверке функций, выполняемых элементами ПО на соответствие требованиям проекта ПО и документации пользователя.

При функциональном тестировании испытания проводятся в контексте реальной среды и учитывая ее влияние. Тестировщик может не обладать особыми знаниями, но есть вероятность того, что при данном методе часть возможных вариантов поведения останется неиспытанной и для проведения необходима документация высокого качества.

Структурное тестирование (Structural Testing), которое также называют тестированием по принципу *«белого ящика»* или «стеклянной шкатулки» («White Box» Testing или «Glass Box» Testing), заключается в проверке внутренней структуры элементов ПО. Метод белого ящика позволяет:

- просмотреть скрытые проблемы, что значительно упрощает их диагностику;
- осуществлять простую автоматизацию;
- обладает развитой системой метрик, сбор и анализ которых легко автоматизируется;

Однако, структурное тестирование не допускает не обладающих достаточными навыками и знаниями тестировщиков. Также, при использовании данного метода, ПО исследуется в отрыве от реальной среды.

Выполнение функционального и структурного тестирования ПО может быть осуществлено независимо друг от друга. Основным видом тестирования является функциональное. Функциональное тестирование применяют для ПО в целом, а также для программных объектов любого уровня (процедуры, модуля, подсистемы, системы). Структурное тестирование дополняет функциональное. Этой вид тестирования возможен для уровня не выше уровня программного модуля.

Таблица 1.2.1 Виды функционального тестирования

Виды тестирования	**Цели тестирования**	**Объекты тестирования**
Тестирование транзакций	Верификация реализации прикладных функций	Транзакции (переходы) между режимами работы системы
Тестирование областей входных данных	Верификация реализации функциональных вычислений и условий для областей входных данных	Области значений входных данных, включая предельные и запрещенные значение
Тестирование синтаксиса	Верификация пользовательского интерфейса и сообщений	Элементы пользовательского интерфейса

Тестирование логических условий	Верификация реализации логических условий ПО	Комбинации логических условий реального мира
Тестирование состояний	Верификация реализации состояний ПО	Состояния и переходы между ними

Таблица 1.2.2. Виды структурного тестирования

Виды тестирования	**Цели тестирования**	**Объекты тестирования**
Тестирование маршрутов	Верификация потока управления внутри программных модулей, обеспечения полноты покрытия программных областей и маршрутов	Программные области и маршруты
Тестирование циклов	Верификация выполнения циклов	Программные циклы и их предельные значения
Тестирование обработки данных	Верификация использования данных	Программные данные

Выделяют также гибридный вид – *метод серого ящика*. Он представлен как различная комбинация методов белого и черного ящиков, в зависимости от того, к каким компонентам программы имеется открытый доступ.

Степень автоматизации так же является одним из путей классификации испытаний. Различают *ручное и автоматизированное* тестирования. *Ручное тестирование* выполняется человеком без использования средств автоматизации. *Автоматизированное тестирование* наиболее популярно и особенно необходимо для сложных систем.

Средства испытаний программных средств

Для оптимизации разработки и тестирования программных продуктов используются различные специальные пакеты *инструментов*. Исходя из потребности выполнения различных между собой процедур, инструменты испытания делят на классы:

1) Для регистрации параметров системы для сбора информации относительно обнаружения и исправления дефектов;
2) Для оценки вероятностных индикаторов надежности программного обеспечения;
3) Для статического анализа текста программного обеспечения.

Багтрекинг или система отслеживания ошибок – программа, разработанная с целью помочь разработчикам или тестировщикам учитывать и контролировать ошибки в программах, ожидания пользователей.

База данных системы отслеживания ошибок содержит сведения об обнаруженных дефектах. Эти сведения включают в себя:

- Идентификатор дефекта;
- Короткое описание дефекта;
- Источник информации о дефекте;
- Дата и время, когда был обнаружен дефект;
- Версия продукта, в которой обнаружен дефект;
- Серьёзность (критичность) дефекта и приоритет решения;
- Описание шагов для выявления дефекта (воспроизведения непреднамеренного поведения программы);
- Ожидаемый результат и фактический результат;
- Исполнитель, ответственный за устранение дефекта;
- Обсуждение возможных решений и их последствий;
- Текущее состояние (статус) дефекта;
- Версия продукта, в которой дефект исправлен.

Таблица 1.2.3 – Примеры систем отслеживания ошибок

Свободно распространяемые	Проприетарные (несвободные)	Разное
Redmine	Atlassian JIRA	BugTracker.NET
BUGS – the bug genie	Bontq	BugNet
Bugzilla	PVCS Tracker	ClearQuest
e Traxis	Project Kaiser	Intland CodeBeamer
GNATS	TrackStudio Enterprise	LifeTask.ru
Launchpad	YouTrack	FlySpray
Mantis bug tracking system	Яндекс.Трекер	StarTeam

В *ручном тестировании* для управления процессом используется множество разнообразных инструментов: Redmine, Jira, HP ALM, IBM Rational Quality Manager, MS Team Foundation Server, TestRail, TestLink и других. Если команда испытателей работает на территории заказчика, то выбор в пользу того или иного инструмента зависит от того, что использует сам заказчик. Однако здесь будут рассмотрены базовый набор инструментов.

Для поиска, конвертации и сравнения файлов удобно используется Notepad++ или PSPad. Первый — это почти тот же Блокнот для Windows, только с открытым исходным кодом и поддерживающий синтаксис большого числа языков программирования (написан на C++). Второй — редактор для текста и кода, позволяющий одновременно работать над несколькими документами и использовать несколько языков программирования, а также создавать шаблоны для автоматизации повторяющихся действий.

Хорошие XML-редакторы: Altova XML Spy (работа с XML и XSD), а также XMLPad (с отладчиком XSL). Среди файловых менеджеров можно выделить Total Commander (для Windows и Android, может соединяться с FTP-серверами), Free Commander (бесплатный, с возможностью группового

переименования и сравнения файлов), trolCommander (кроссплатформенный, написан на Java) и Far Manager (консольный).

Вспомогательные инструменты – средства, которые не относятся к тестированию ПО напрямую, но помогают в их организации.

Известно, что средне статистический тестировщик пользуется текстовыми файлами с шаблонными текстовыми данными. Это, к сожалению, может привести к эффекту пестицида, когда баги со временем перестают находиться. Устранить такую проблему можно используя специальные инструменты, которые умеют генерировать данные: Mockaroo. Это приложение умеет не только подбирать данные, но и генерировать SQL-запросы.

Для того, чтобы все проектные задачи были выполнены в срок, важно правильно их распределить. Ставить задачи, получать напоминания и вычеркивать из списка выполненное можно даже с помощью корпоративной почты в MS Outlook. Хотя гораздо эффективнее (особенно, когда задач больше, чем кажется) использовать для этого специальные инструменты, вроде Evernote. Для планирования командной работы отлично подходит Trello.

Поскольку наш мозг не всегда хорошо воспринимает информацию в виде текстов, списков и таблиц, ее можно визуализировать. Диаграмма связей или интеллект-карта – отличный метод структурирования задач (а также мыслей и планов). Если добавить в Mind Map данные о тестах, которые нужно провести, то сразу станет понятно, сколько их, какие между ними связи, есть ли среди них что-то лишнее или недостающее. С помощью интеллект-карт так же можно изобразить порядок своих действий, этапы работы и другое. В качестве примера можно выделить особо известные MindJet (добавляет информацию с разных сайтов, приложений, платформ и контактов), MindMup (бесплатный, легко сохранять карты и делиться), Coggle (простой в освоении), Bubbl.us (онлайн редактор и приложение) и XMind (серьезный софт для построения ментальных карт и различных диаграмм).

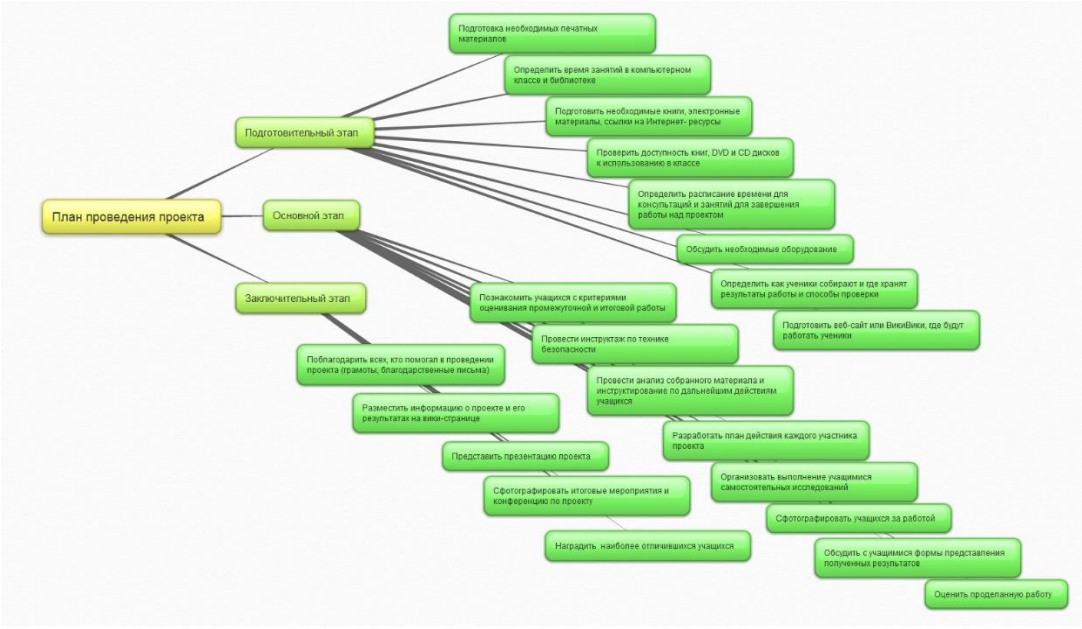


Рисунок 1.2.3 – Пример использования интеллект-карт

Чек-листы – это простой, но эффективный способ не только для упорядочивания повторяющихся действий, но и для самопроверки. Можно использовать уже готовые списки, составленные и любезно выгруженные другими тестировщиками в сеть, а можно подготовить свои собственные и использовать их, например, перед приемочным тестированием, для юзабилити-тестирования и т.д.

Один из самых удобных инструментов для создания чек-листов - Sitechco. Это бесплатный онлайн-сервис, в котором легко завести свои чек-листы, хранить результаты, делиться ими с командой, просматривать отчетность и статистику.

1.3 Процесс испытаний программных средств и его технологическое обеспечение

Последовательность и контроль испытаний основная задача специалистов. Условно процесс испытаний можно описать пятью этапами:

I. Обследование ПС, анализ предоставленной документации по программному продукту;

На данном этапе используется проектная документация о программном продукте. Она должна содержать подробную характеристику ПО, его назначение

и функции, т.е. техническое задание на основе которого будет совершаться определение соответствия результатов тестирования с ожидаемыми.

Объект испытаний – программный продукт, подвергаемый испытаниям. Результаты испытаний принимаются в отношении объекта: о его пригодности, о дефектах, о решении допустимости продукта к эксплуатации пользователями.

Объектами испытаний могут быть программное средство (системное тестирование), отдельные компоненты программы (тестирование компонентов).

II. Изучение структуры и компонентов ПС, анализ функций и путей.

Вторым этапом проводится анализ компонентов, функций ПО для дальнейшего установления метода испытаний.

III. Анализ показателей качества ПС и установление методов определения их значения.

Можно выделить основные принципы качества ПС. Они состоят из характеристик, которые условно объединяют по группам.

1. Функциональная пригодность:

- пригодность применения;
- корректность/точность;
- защищенность;
- взаимодействие.

2. Надежность:

- завершенность;
- устойчивость к ошибкам;
- восстанавливаемость;
- готовность.

3. Эффективность

- временная эффективность;
- использование ресурсов;

4. Практичность

- понятность;

- уровень сложности;
- легкость изучения;
- привлекательность.

5. Сопровождение

- удобство анализа;
- стабильность;
- тестируемость.

6. Переносимость

- адаптируемость;
- простота устанавливания;
- соответствие;
- защищаемость.

IV. Разработка или использование программно-технических средств, баз данных.

V. Проведение испытаний по разработанному плану испытаний, подведение итогов.

В процессе испытаний должно проверяться, что комплекс программ работает в соответствии со спецификацией требований и удовлетворяет следующим общим базовым критериям:

- при допустимых входных данных комплекс программ вырабатывает верный результат, соответствующий требованиям;
- при недопустимых входных данных комплекс программ отвергает входные данные и выдает соответствующее диагностическое сообщение;
- независимо от допустимости входных данных программы не зависают и не завершаются аварийно.

Основные этапы тестирования вручную:

1. Подготовительный

К этому этапу относится анализ исходных документов о системе, среди которых могут быть ТЗ, функциональные и бизнес-требования, паспорт проекта

и другие. Дальше разрабатываются и согласовываются план тестирования и тест-кейсы, сроки и число итераций, оцениваются возможные риски.

2. Основной

Тестирование ведется вручную по подготовленным заранее тестовым сценариям с фиксированием всех найденных ошибок в багтрекинговой системе.

3. Итоговый

На завершающем этапе разрабатываются и согласовываются отчеты о проведенном тестировании. Обычно они включают в себя список обнаруженных отклонений и рекомендации о возможности внедрения ПО.

Технологическое обеспечение – это комплекс организационных и технических средств, которые обеспечивают информационную, программную, методическую поддержку испытаний программных продуктов.

То есть, данный комплекс должен содержать информацию об объекте испытаний, инструментальным средствам, исполнителях испытаний и нормативно-техническую документацию.

Средства испытаний охватывают любые технические средства, применяемые при испытаниях. Туда входят испытательное оборудование, средства измерений, применяемые для измерений тех или иных характеристик объекта или контроля условий испытаний.

Исполнители испытаний – специалисты, участвующие в процессе испытаний. Безусловно, исполнителем не может быть неподготовленный человек, особенно при испытаниях критически важных объектов. Поэтому требованиями к исполнителям являются их квалификация, образование, опыт и стаж работы и др.

Нормативно-техническая документация устанавливается для введения общих правил и оформления испытаний, что обеспечивает согласованность нормативов и методической основы. При разработке или испытании сложных систем проблемой является взаимная увязка всех подсистем, из-за чего результаты испытаний искажаются.

В нормативно-техническую документацию входят:

- Эксплуатационная;
- Документация по техническим условиям;
- Программа и методика испытаний.

В качестве сопроводительных документов используются:

- Техническое задание;
- Технико-экономическое обоснование;
- Документация по организации и проведении контроля.

1.4 Задача разработки системного проекта создания технологического обеспечения испытания ПСКН

Изучив подробнее все аспекты и детали технологического обеспечения испытаний, несложно прийти к выводу, что для такой обширной системы необходима структура. И это в полной мере может обеспечить системный проект.

Системный проект понятие вовсе неновое. Проект используется во всех сферах деятельности для упрощения сложных систем, контроля за процессами и т.д.

В рамках этой работы рассматривается системный проект, как средство взаимной увязки совокупности мероприятий создания технологического обеспечения испытаний ПСКН.

Проект – это концепция построения технологического обеспечения испытаний ПС, который определяет требования и будущий облик системы, а также ее функции и структуру. Под этим следует понимать распределение ролей, функций, обязанностей между средствами испытаний и исполнителями.

Достоинства	Возможность описать будущую систему и внести поправки
	Уменьшение затрат на разработку и внедрение
	Оценка времени и результатов
	Наглядное представление для участников процесса
	Создание оптимальной структуры

Рисунок 1.4.1 – Преимущества использования системного проекта

Целью системного проекта является:

1) Обеспечение соответствия ожиданий заинтересованных лиц с целями испытаний;

2) Обеспечение системности и охват целого комплекса создания технологического обеспечения;

3) Обеспечение управляемости, контроля и прозрачности проекта.

Задачи системного проекта	определение внешних требований к технологическому обеспечению испытаний ПС, условий его функционирования
	определение функций и структуры технологического обеспечения испытаний ПС
	определение требований к компонентам технологического обеспечения испытаний ПС и их характеристикам;
	определение состава работ и ограничений (сроки, ресурсы и др.) по реализации проекта создания технологического обеспечения испытаний ПС
	оценка ресурсов, требуемых для реализации проекта создания технологического обеспечения испытаний ПС
	оценка проектных рисков

2. РАЗРАБОТКА КОНЦЕПЦИИ ПОСТРОЕНИЯ ТЕХНОЛОГИЧЕСКОГО ОБЕСПЕЧЕНИЯ ИСПЫТАНИЙ ПСКН

2.1 Требования к технологическому обеспечению испытаний ПСКН

Как было отмечено в главе 1, разработка технологического обеспечения начинается с определения основных требований. Для этого необходимы нормативные документы, в которых будут указаны эти данные.

Нормативная база обеспечивает решение задач:

- Регламентации общего порядка процесса испытаний;
- Регламентация общих требований к ПО;
- Формирование методических материалов, для использования исполнителями наиболее эффективных методов испытаний, единство схем;
- Регламентация состава и форм проектных материалов в программной документации.

В состав входят стандарты и другие типы документов, шаблоны проектных документов.

Республика Казахстан использует национальные и международные стандарты качества в области космической деятельности и информационных технологий.

Ниже приведен перечень стандартов, используемых в последние годы применительно к космической технике.

1) СТ РК ECSS-Q-ST-80C-2012 Гарантия космической продукции. Гарантия качества программного обеспечения [текст]. – Введ. 2013–07–01. - Астана, 2012. – 98 с.: ил.

2) ГОСТ Р ИСО/МЭК 25040-2014 Требования и оценка качества систем и программного обеспечения (SQuaRE). Процесс оценки. Определения [Текст]. – Введ. 2015-06-01. – М.: Стандартинформ, 2014.

3) ECSS-Q-HB-80-03A: 2012 Space product assurance. Software dependability and safety.

4) ECSS-Q-ST-80C Rev.1: 2017 Space product assurance. Software product assurance.

5) ISO/IEC 25001:2014 Systems and software engineering - Systems and software Quality Requirements and Evaluation (SQuaRE) - Planning and management

6) СТ РК ECSS-E-ST-31C-2016 Космический инжиниринг. Термический контроль. Общие требования.

7) СТ РК ECSS-E-ST-33-11C-2016 Космический инжиниринг. Взрывчатые системы и устройства.

8) СТ РК ECSS-Q-ST-40-02C-2016 Гарантия космической продукции. Анализ опасности.

9) СТ РК ECSS-Q-ST-60-12C-2016 Гарантия космической продукции. Проектирование, выбор, закупка и применение монолитных интегральных схем свч-диапазона.

10) СТ РК ECSS-Q-ST-70-01C-2016 Гарантия космической продукции. Контроль загрязнения и чистоты.

11) СТ РК 2843 - 2016 (ECSS Q-ST-70-02C:2008) Гарантия космической продукции. Термовакуумные испытания по выделению газа из экранирующих космических материалов.

12) СТ РК 2844 - 2016 (ECSS E-ST-20-08C:2012) Космический инжиниринг. Сборка фотоэлектрических компонентов.

13) СТ РК ECSS–Q–ST–30–09C–2016 Гарантия космической продукции. Анализ готовности.

14) СТ РК ECSS Q-ST-70-46C-2016 Гарантия космической продукции. Резьбовые соединения. Требования.

15) СТ РК ECSS Q-ST-70-71C-2016 Гарантия космической продукции. Материалы и процессы. Требования отбора.

16) СТ РК ECSS–Q–ST–60–05C–2016 Гарантия космической продукции. Общие требования к приобретению герметичных гибридных интегральных микросхем.

17) СТ РК EN 2424 – 2016 Авиация и космонавтика. Маркировка изделий для авиационно-космической техники.

18) СТ РК ECSS–Q–ST–60–14C–2016 Гарантия космической продукции. EEE компоненты. процедуры восстановления.

19) СТ РК 2842-2016 (ECSS–Q–ST–70–05C :2009) Гарантия космической продукции. Обнаружение органических загрязнений поверхности методами инфракрасной спектроскопии.

20) СТРК ECSS–Q–ST–30–02C–2016 Гарантия космической продукции. Анализ характера и последствий отказов

21) СТ РК ECSS–Q–ST–30–11C–2016 Гарантия космической продукции. Пониженный режим работы EEE компонентов.

22) СТ РК ГОСТ Р 56098 – 2016 Космические системы. Метрологическая экспертиза конструкторской документации. Организация и порядок проведения.

23) СТ РК 3054-2017 Геоинформационное картографирование Пространственные модели местности. Общие требования.

24) СТ РК ISO 10795-2017 Космические системы. Менеджмент и качество программ. Термины и определения.

25) СТ РК ISO 10786-2017 Космические системы. Детали и узлы конструкции.

26) СТ РК ISO 14623 – 2017 Космические системы. Сосуды и конструкции под давлением. Проектирование и функционирование.

27) СТ РК EN 9100-2017 Система менеджмента качества. Требования к авиационным, космическим и оборонным организациям.

28) СТ РК ECSS-E-ST-31-02C-2017 Космический инжиниринг. Оборудование двухфазной теплопередачи. Требования.

29) СТ РК ECSS Q-ST-60-13C-2017 Гарантия космической продукции. Требования к применению электрических, электронных и электромеханических изделий.

30) СТ РК ECSS-Q-ST-70-26C-2017 Гарантия космической продукции. Высокая надежность электрических соединений.

31) СТ РК ECSS Q-ST-70-13C-2017 Гарантия космической продукции. Испытания изоляционных слоев и покрытий.

32) СТ РК ECSS Q-ST-70-06C-2017 Гарантия космической продукции. Испытание космических материалов ультрафиолетовым излучением и элементарными частицами.

33) СТ РК ECSS Q-ST-70-21C-2017 Гарантия космической продукции. Испытания на воспламеняемость экранирующих космических материалов.

34) СТ РК ECSS E-ST-20-06C-2017 Космический инжиниринг. Влияние внешней среды на электрическую систему космических аппаратов.

35) СТ РК ECSS E-ST-10-04C-2017 Космический инжиниринг. космическое пространство.

36) СТ РК ECSS-Q-ST-70-22C-2017 Гарантия космической продукции. Контроль материалов с ограниченным сроком хранения.

37) СТ РК ECSS-Q-ST-70-09C-2017 Гарантия космической продукции. Измерения термооптических свойств теплоизолирующих материалов.

38) СТ РК ECSS-Q-ST-20-10C-2017 Гарантия космической продукции. Серийные изделия. Требования к применению в космических системах.

39) СТ РК ECSS-E-ST-35-01C-2017 Космический инжиниринг. Жидкостный и электрический двигатели космического аппарата.

40) СТ РК ECSS M-ST-10-01C -2017 Менеджмент космического проекта. Организация и проведение рассмотрений.

41) СТ РК ECSS-E-ST-70-31C-2017 Космический инжиниринг. Наземные системы и операции. Мониторинг и управление определением данных.

42) СТ РК ECSS-E-ST-10-12C -2017 Космический инжиниринг. Методы расчета излучения и его последствий. Правила обеспечения проектной надежности.

В IT наиболее целесообразно использовать следующие нормативные документы:

1) ГОСТ Р ИСО МЭК 12270-99 «Информационные технологии. Процессы жизненного цикла программного обеспечения»

2) ISO/IEC NR 15504 – «Оценка процессов жизненного цикла ПО»

3) IEEE 1063-1987 «Стандарт документации пользователя ПО»

4) IEEE 829 «Планирование тестирования ПС»

5) СТ РК ГОСТ Р ИСО/МЭК «Методы и средства обеспечения безопасности информационных технологий»

Требования к документации

Перейдем к подробному разбору *требований технологической документации* испытаний ПС.

Эксплуатационный нормативный документ содержит рекомендации по эксплуатации ПО. Перечень документации, разрабатываемый на продукцию, в основном, зависит от уровня ее технической сложности, однако можно выделить следующие:

- Руководство по эксплуатации (РЭ);
- инструкция по вправке;
- формуляр (ФО);
- паспорт (ПС);
- этикетка (ЭТ);
- каталог компонентов и состава ПС;
- специальные инструкции;
- ведомость эксплуатационных документов (ВЭ).

Условия испытаний ПС также должны быть подробно расписаны.

Программа и методика испытаний определяет технический данные объектов испытания и тестирования, порядок и их контроль. Этот документ предназначен для организации

Программы и методики испытаний (ПМИ) разрабатываются на основе ТЗ, конструкторской и программной документации с использованием, при необходимости, типовых программ и методик испытаний, для конкретной

категории или вида испытаний, с целью определения соответствия продукции ТЗ или требованиям НД (ГОСТ, РД, ОТТ и пр. на определенный вид продукции).

В программу испытаний включают:

- объект испытаний;
- цель испытаний;
- объем испытаний;
- условия и порядок проведения испытаний;
- материально-техническое обеспечение испытаний;
- метрологическое обеспечение испытаний;
- отчетность по испытаниям.

В программы испытаний включают перечни конкретных проверок (решаемых задач, оценок), которые следует проводить при испытаниях для подтверждения выполнения требований ТЗ, со ссылками на соответствующие методики испытаний.

В методику испытаний включают:

- оцениваемые характеристики (свойства, показатели) продукции;
- условия и порядок проведения испытаний;
- способы обработки, анализа и оценки результатов испытаний;
- используемые средства испытаний, контроля и измерений;
- отчетность.

Виды испытаний: Исследовательские испытания, Контрольные испытания, Сравнительные испытания, Определительные испытания, Государственные испытания, Межведомственные испытания, Ведомственные испытания, Доводочные испытания, Предварительные испытания, Приемочные испытания, Квалификационные испытания, Приемо-сдаточные испытания,

Периодические испытания, Инспекционные испытания, Типовые испытания, Натурные испытания и др.

Требования к средствам испытаний

Технологическое обеспечение включает также *средства испытаний*, которым необходимо определение требований.

Испытательные средства должны иметь высокую точность, исправность работы и визуальную простоту использования, но таким образом нельзя достаточно точно описать требования к инструментам испытаний.

Инструменты испытаний программного обеспечения:

- анализаторы исходных текстов программ;
- анализатор объектных кодов программ;
- программы модельных испытаний;
- средства измерения и оценки показателей качества ПС;
- средства измерения и оценки показателей безопасности ПС;
- программы экспертизы выполнения требований;
- средства визуализации результатов испытаний;
- программы принятия решений;
- программы расчета рисков

Анализаторы исходного кода – класс продуктов для разработчиков и заказчиков разработки программного обеспечения, веб-приложений и модулей расширения, с целью обнаружения ошибок, которые могут быть использованы для несанкционированного доступа к информации.

Методика модельных испытаний включает в себя огромное множество моделей надежности, которые имеют свои слабые и сильные стороны в зависимости от цели оценки.

Требования к средствам испытаний:

- Поддержка большого набора систем;
- Поддержка национальных, международных стандартов;
- Наличие базы уязвимостей и ее использование;

- Возможность визуализации программного кода;
- Приемлемая скорость сканирования;
- Глубокая специализация;
- Поиск уязвимостей нулевого дня;
- Крупная база перечня анализируемых языков программирования;
- Возможность интеграции с другими системами;
- Оценка свойств на каждом этапе;
- Возможность выбора параметров объекта;
- Возможность воспроизведения функциональной системы и т.д.

Требования к исполнителям проекта

Исполнители как участники системного проекта являются важнейшим элементом при испытаниях программных продуктов. Это люди с различным уровнем профессионализма, а также с опытом и знаниями, для которых, конечно же, важен стимул и удовлетворение потребностей.

Так же, как к средствам испытаний, в соответствие с ростом требований к любым продуктам на сегодняшний день, наблюдается увеличение требований и ответственности за испытания к специалистам, который этот процесс обеспечивают.

Специалисты в соответствии со своей квалификацией и ролью в проекте вносят в него положительный вклад, а также специфические дефекты и ошибки, что целесообразно учитывать при прогнозировании эффективности процессов испытаний.

Интеллектуальный труд специалистов в настоящее время высоко оплачивается, потому от уровня сложности проекта растет затраты на оплату труда. Разработка сложных комплексов, особенно, на начальных и завершающих этапах, характеризуется высокой долей творческого труда. Дефекты, трудоемкость и длительность отдельных операций и частных работ существенно

зависят от индивидуальных особенностей их исполнителей и характеристик конкретного проекта. В перспективе, несмотря на автоматизацию и повышение инструментальной оснащенности технологии разработки программ, доля интеллектуального труда при испытаниях версий крупных программных продуктов возрастает. Даже при сокращении суммарных затрат на разработку программных компонентов за счет автоматизации нетворческого труда, все более определяющей для технико-экономических показателей испытаний становится доля затрат на творческий труд и возрастают требования к творческим способностям испытателей.

Не менее важным считается четкое обоснование выбора ресурсов со стороны заказчика. Недостаточное внимание к этому процессу приводит к возникновение еще более острых проблем, таких как, конфликты между заказчиком и разработчиком.

Поэтому целесообразно активно привлекать заказчиков к управлению испытаниями, чтобы обеспечить своевременность разработки версий программных продуктов в условиях ограниченных ресурсов. Безусловно, нельзя исключить различие уровней квалификации. Последствием допущения подобной ошибки является то, что при этом первоначальное техническое задание зачастую оказывается недостаточно квалифицированным и подвергается в дальнейшем многократным изменениям. Этому же может способствовать различие между заказчиком и разработчиком в квалификации, уровне понимания целей продукта и необходимых затрат на реализацию дополнительных требований. Извечной проблемой также является желание заказчика форсировать сроки испытаний.

Даже квалифицированные заказчики вынуждены иногда корректировать техническое задание на любых этапах управления проектом, что может влиять на увеличение затрат на 10 – 20%. Вследствие этого необходима переработка требований и испытания готовых программ, что отражается большим ущербом вследствие дефектов исходных требований заказчика.

При испытаниях высококачественных комплексов программ, прежде всего, необходима организация и тесное взаимодействие представителей заказчика и разработчиков продукта. Взгляды и требования заказчика, в основном, отражаются в функциональных и потребительских характеристиках версий программного продукта. Устремления разработчиков направлены на возможность и способы их реализации с требуемым качеством. Эти различия исходных точек зрения на проект приводят к тому, что некоторые неформализованные представления тех и других имеют зоны неоднозначности и взаимного непонимания требований, что может приводить к конфликтам. Организация четкого взаимодействия и сокращение этих зон неоднозначностей требует проведения определенных мероприятий и контактов по обмену знаниями, взаимному повышению квалификации и обучению. При испытаниях комплексов программ коллективами велика роль квалификации руководителей разработки, что непосредственно отражается на успехе проекта.

Лидером испытаний программного продукта может быть: менеджер продукта, менеджер проектирования, руководитель проекта. Поэтому здесь перечислены основные требования к руководителям:

- руководить процессом выявления и формирования требований к функциям программного продукта, подлежащим испытаниям;
- рассматривать конфликтующие пожелания, поступающие от различных участников проекта комплекса программ и находить компромиссы, необходимые для определения приоритетов набора функций, представляющих наибольшую ценность для максимального числа участников проекта, прежде всего, заказчика и пользователей;
- вести переговоры с руководством системы, пользователями и разработчиками и поддерживать равновесие между тем, чего хочет заказчик, и тем, что может предоставить команда разработчиков и испытателей за ресурсы и время, отведенные для их реализации;

- осуществлять проверку спецификаций требований программного комплекса, чтобы удостовериться, что они соответствуют базовой концепции проекта и функций программного продукта;
- осуществлять управление изменением приоритетов испытаний задач и функций, а также добавлением и исключением новых функций комплекса программ.

Коллектив испытателей должен состоять из квалифицированных, проблемно-ориентированных аналитиков и системных архитекторов, способных переводить функциональные требования заказчика в конкретные спецификации и технические требования к испытаниям комплекса программ и его компонентов.

Системные аналитики сложных ПС должны иметь, прежде всего, хорошую подготовку по системному анализу алгоритмов и комплексов программ, по методам оценки эффективности, организации и планированию крупных разработок, и испытаниям программ. Им необходима высокая квалификация по архитектурному построению, и испытаниям версий программных продуктов определенных классов, умение организовать коллектив для решения общей целевой задачи реализации продукта. Это позволит на ранних этапах исключать или сокращать дефекты, обусловленные различием представления ими целей и задач комплекса программ, а также показателей качества.

Особую роль для обеспечения высокого качества программных продуктов играют *испытатели – тестировщики.* Они должны обеспечивать проверку реализации требований функциональных спецификаций, пользовательских интерфейсов, разрабатывать стратегию, выполнять и документировать тестирование для каждого компонента и версии программного продукта, должны быть административно независимыми от программистов и спецификаторов требований. Организационная структура испытателей и распределение ответственности за определенные уровни корректировок, имеют важное значение для постоянного совершенствования процесса и качества

результатов. Различные виды работ, проводимые по Программе испытаний, обычно включаются в состав декомпозиции плана работ всего проекта. Необходимо определять и фиксировать в плане испытаний, распределение основных ролей и ответственности между участниками процесса тестирования. Испытатели должны обладать специфической квалификацией для разработки эффективной стратегии тестирования, а также для проведения работ по испытаниям в необходимом и достаточном объеме.

Хороший испытатель должен обладать аналитическим складом ума, быть внимателен к деталям, организован, проявлять творческий подход и способности к прогнозированию, быть твердым и гибким при обсуждении спорных вопросов с разработчиками компонентов и требований к программному продукту. Ему целесообразно иметь опыт работы с различными платформами, операционными системами, интерфейсами других продуктов и систем, базами данных и языками разработки комплексов программ.

Для эффективного проведения тестирования требований необходимо, чтобы испытатели обладали достаточным опытом осуществления принятой стратегии тестирования и применения выбранных инструментов.

Личностные характеристики разработчиков программного обеспечения и отношения, сложившиеся внутри группы, являются важным фактором и резервом для повышения эффективности разработки и безопасности программных продуктов. Человеческий фактор оказывает гораздо большее влияние на эффективность и качество работы, чем любой другой отдельно взятый фактор в их жилом комплексе. Ни один специалист не может быть носителем всех необходимых высоких качеств, но группа испытателей, есть одна команда, может представлять собой максимально возможное количество требуемых качеств [7]:

- умение описывать последовательность событий и конфигурацию системы, которые приводят к возникновению дефекта, это включает способность четко документировать процедуры и результаты тестирования требований,

умение передать такую информацию разработчикам, другим испытателям и руководству;

- умение критиковать и корректно воспринимать критику, так объяснять разработчикам компонентов суть дефектов, чтобы с его слов их можно было устранить;
- умение противостоять внешнему давлению руководства, так как испытания являются завершающей стадией любого процесса разработки и, как правило, протекает в стрессовых обстоятельствах недостатка ресурсов и времени;
- терпеливым и готовым выполнять исполнение тестов столько раз, сколько нужно для того, чтобы выявить и локализовать дефект, после чего повторно выполнить регрессионные тесты, чтобы убедиться в корректном его устранении;
- обладать гибким мышлением – быть способным быстро переключаться на испытания нового программного компонента или продукта, или даже отказываться от испытания одного продукта в пользу другого, обладающего более высоким приоритетом;
- обладать способностью видеть общую панораму развития проекта и уметь при необходимости сосредоточиваться на деталях, – иметь широкий, и динамичный кругозор.

Учитывая сложность тестирования на клиент-сервере или нескольких пользователей в среде, тест должен обладать большим количеством технических знаний и навыков. Он должен иметь опыт работы с различными платформами, операционными системами, поддерживающими приложения, интерфейсами других продуктов и систем, базами данных и языками разработки приложений.

Для сокращения и устранения дефектов разработки посредством адекватных контрмер необходима четкая организация коллектива специалистов и автоматизация процессов исправления дефектов, которые позволяют избегать множества вторичных ошибок, обусловленных недостаточной координацией

проводимых корректировок и формирования новых версий сложных программных продуктов. Этому должна способствовать утвержденная дисциплина и иерархия принятия решений на координированные изменения компонентов и комплекса в целом, должностными лицами, поддержанная методами и средствами защиты от несанкционированного доступа при выполнении корректировок документов специалистами различной квалификации и права доступа к модификациям компонентов на разных уровнях проекта. Для утверждения документов и выполнения изменений и корректировок на каждом уровне разработчиков изменений и дополнений необходимо определить полномочия специалистов или групп. Контрмеры включают в себя перечень работ, выполняемых для выполнения: запрос на разрешение изменения; запрос на изменение; отслеживание изменения; разделение изменений к предыдущим версиям программного продукта и документации. Изменения, влияющие на действующий программный продукт, должны быть представлены заказчику в соответствии с формами и процедурами, установленными контрактом.

2.2 Разработка функциональной модели и структуры технологического обеспечения испытаний ПСКН

Схема проведения процедуры испытаний ПС представлены на рисунке 2.2.1. Блок-схема описывает процесса документирования испытаний и этот метод является наиболее удобным в использовании.

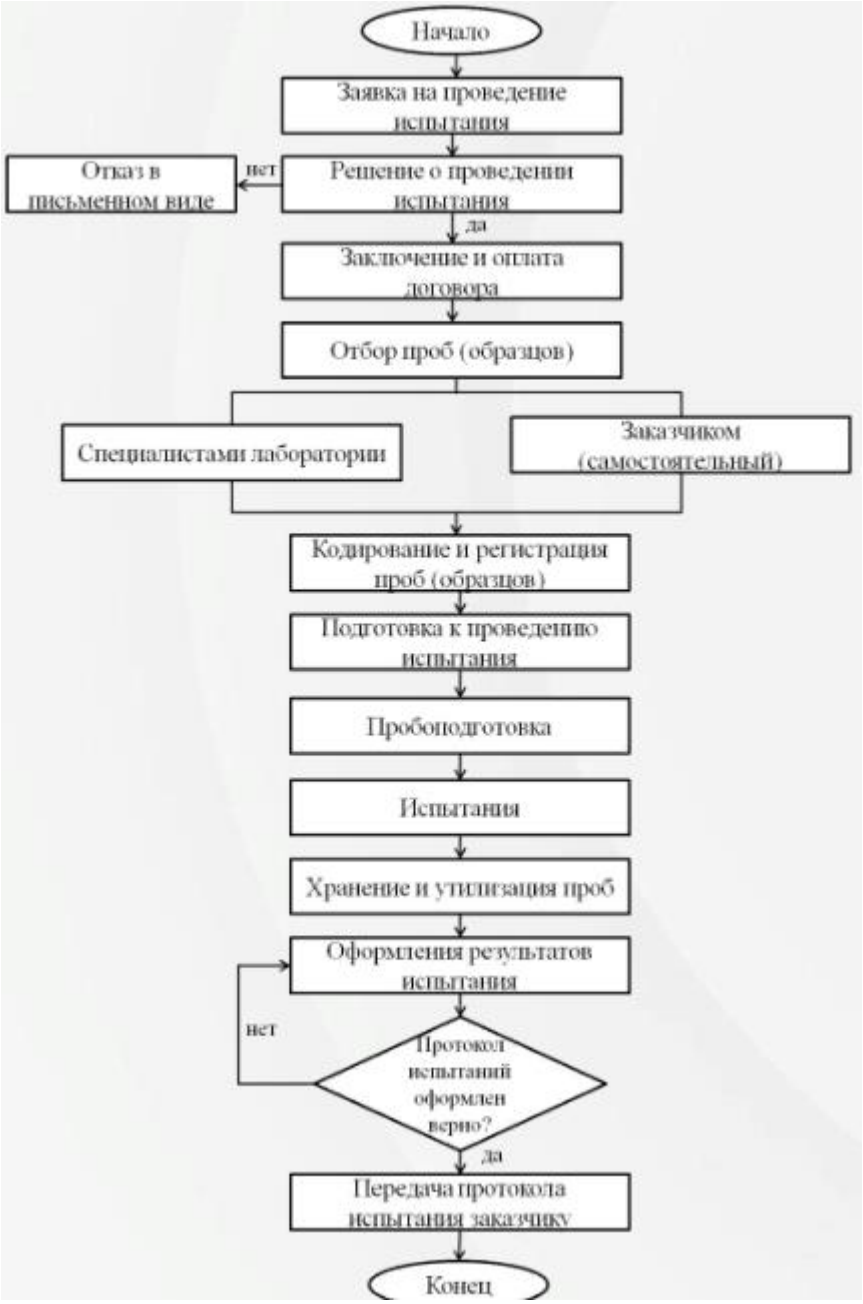


Рисунок 2.2.1 – Блок-схема испытаний

Структура технологического обеспечения должна включать в себя:

1. Средства измерения;
2. Средства оценки;
3. Рабочие программные продукты;
4. Средства визуализации;
5. Процесс испытаний;
6. Базу данных результатов.

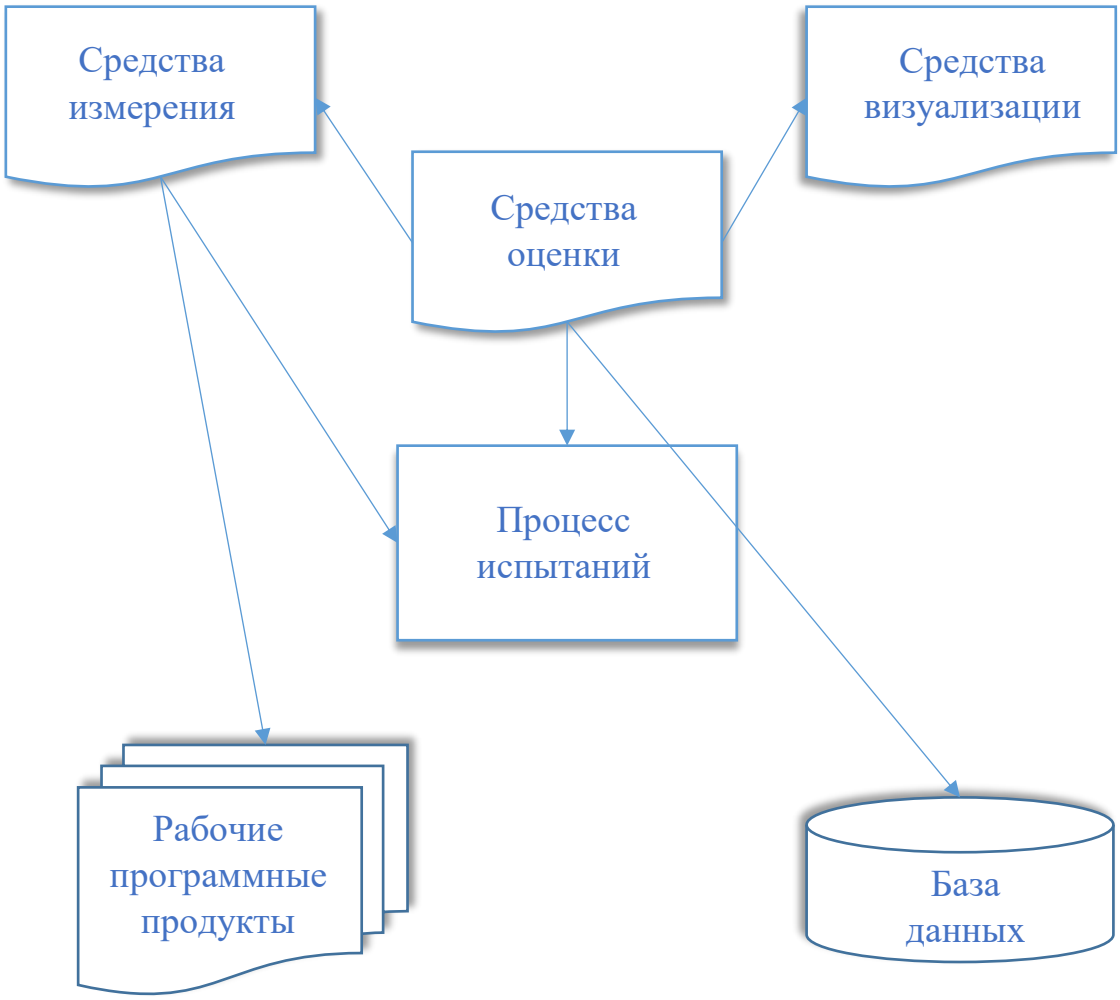


Рисунок 2.2.2 – Структура технологического обеспечения испытаний

В нашем случае идет оценка технологического обеспечения испытаний, поэтому в функциональной модели присутствуют такие элементы проекта как исполнители и их роли в проекте, а также материалы документации.

Таблица 2.2.1 – Распределение ролей исполнителей проекта

ROLES & RESPONSIBILITIES/РОЛИ И ОТВЕТСТВЕННОСТЬ

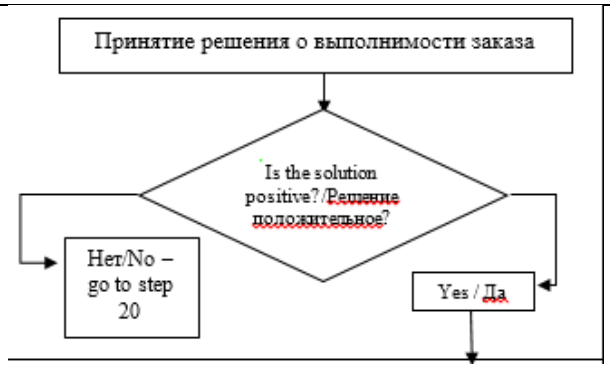
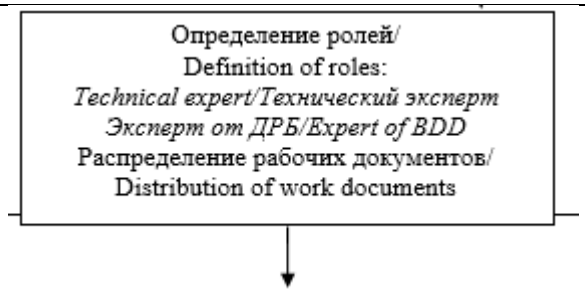
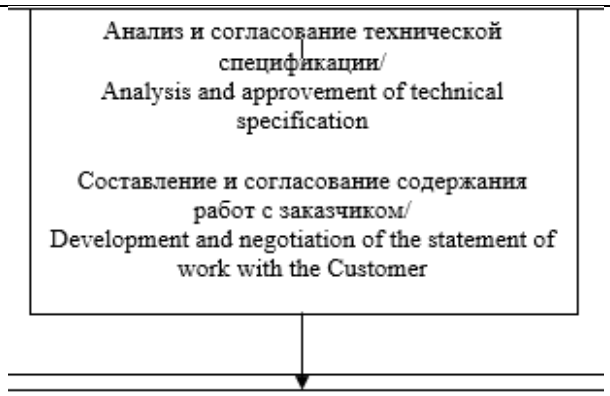
Business Development Director/ Business development bloc Manager The person responsible for the offer and an employee of the business development bloc, who responsible for: - transfer of technical requirements - setting deadlines for preparing a response - transfer of the Customer's contacts for working out technical requirements	**Директор по РБ/ Менеджер блока по развитию бизнеса** Лицо, ответственное за предложение, является сотрудником блока по развитию бизнеса и отвечает за: - передачу технических требований - постановку сроков подготовки ответа - передачу контактов Заказчика для отработки тех. требований
Business Development Department Expert responsible for: - approval of the content of works with the customer - approval of the technical specification with the customer - development of TND - transfer of the signed contract and the order to start the project	**Эксперт** отвечает за: - согласование содержания работ с заказчиком - согласование технической спецификации с заказчиком - разработку TND - передачу подписанного договора и приказа о начале проекта
Technical Director responsible for: - making a decision on the feasibility of the order - defining project roles - distribution of working documents - coordinating roles after the project starts - review of detailed design results - review of the results of the build and integration stage - review of the results of the test stage - closing the project	**Технический директор** отвечает за: - принятие решения о выполнимости заказа - определение ролей по проекту - распределение рабочих документов - согласование ролей после начала проекта - рассмотрение результатов детального проектирования - рассмотрение результатов этапа сборки и интеграции - рассмотрение результатов этапа испытаний - закрытие проекта

Acting head of design office responsible for: - making a decision on the feasibility of the order	**Начальник** отвечает за: - принятие решения о выполнимости заказа
Technical Expert responsible for: - analysis of the technical specification from the customer - preparation of the project content - assessment of the necessary equipment/components and human resources including the time frame for the project implementation	**Технический эксперт** отвечает за: - анализ технической спецификации, поступившей от заказчика - составление содержания работ по проекту - оценку необходимого оборудования/комплектующих и человеческих ресурсов, в том числе и сроков для реализации проекта
Project Manager responsible for: - work packages distribution - preparation of document package for preliminary and detailed design stages - execution of works according to the work schedule - coordination of design results and work at each stage with customer - accounting and elimination of comments from the customer - creating purchase requests and internal orders for the project - signing of acts of acceptance and transfer of products/services and closing the project	**Менеджер проекта** отвечает за: - распределение пакетов работ - подготовку пакета документов этапов предварительного и детального проектирования - выполнение работ согласно графику работ - согласование результатов проектирования и работ на каждом этапе с заказчиком - учет и устранение замечаний от заказчика - создание запросов на закуп и внутренний заказ по проекту - подписание актов приема-передачи продукции/услуг и закрытие проекта

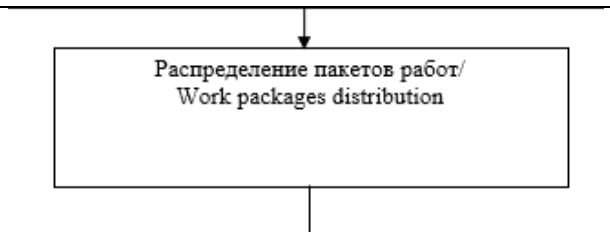
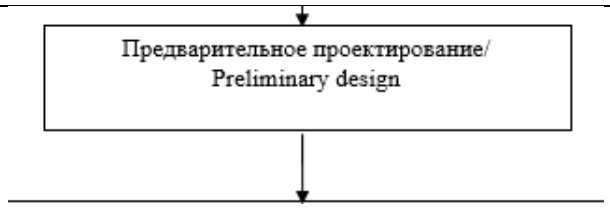
Work package Managers responsible for: - preliminary and detailed design - implementation of works according to the work schedule - preparation of project documentation - elimination of customer comments	**Менеджеры комплексных работ** отвечают за: - предварительное и детальное проектирование - выполнение работ согласно графику работ - подготовку проектной документации - устранение замечаний заказчика
System engineer responsible for: - work packages distribution - preliminary and detailed engineering - review of detailed design results - review of the results of the build and integration stage - review of the results of the test stage	**Системный инженер** отвечает за: - распределение пакетов работ - предварительное и детальное проектирование - рассмотрение результатов детального проектирования - рассмотрение результатов этапа сборки и интеграции - рассмотрение результатов этапа испытаний
Test Manager responsible for: - preparation of test plan - testing - creating a test report	**Менеджер по испытаниям** отвечает за: - подготовку плана испытаний - проведение испытаний - создание отчета о проведении испытаний

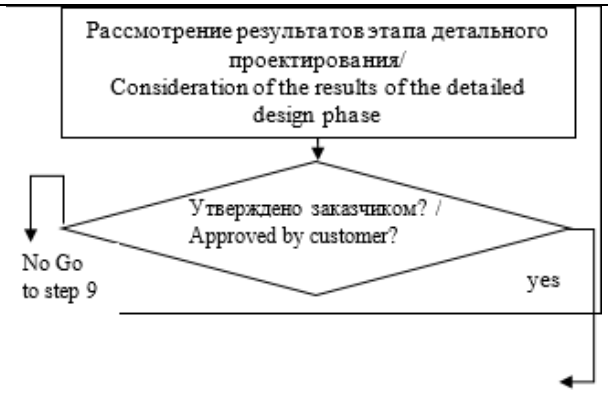
Таблица 2.2.2 – Структура проведения работ

Description/Описание				
Workflow / Порядок проведения работ				
	/Ответственный	**Input/Входные данные**	**Task/Задача**	**Output/Выходные данные**

	Директор Менеджер Начальник Технический директор/ *Business Development Director* *Business Development Department Manager* *Acting head of design office* *Technical director*	Запрос на получение информации (включающий описание тех. требований или предварительную спецификацию)/ Request for Information (-RFI) Preliminary technical specification Контакты заказчика		Уведомление ДРБ о проблемах, препятствующих выполнению заказа/возможности выполнения заказа Notifying the DRB of problems that prevent the order from being completed/the ability to complete the order
	Технический директор Директор по РБ / *Technical director* *Business Development Director*			
	Технический эксперт Эксперт от ДРБ Заказчик/ *Technical expert* *Business Development Expert*	Запрос на получение информации/ Request for Information (SKTB-RFI)		Согласованная тех. спецификация/Approved TS Первая согласованная версия содержания работ/Statement of Work– Draft

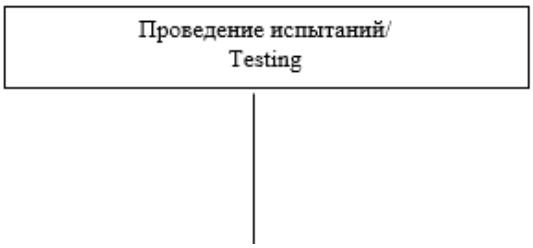
	Customer	Предварительная тех.спецификация (при наличии) / Preliminary technical specification Матрица соответствия/ Compliance matrix		
	Технический эксперт Эксперт от ДРБ / *Technical expert Business Development Expert*	Запрос от BDD на предоставление данных для расчета TND / Request from BDD to provide data for TND calculating	Оценка необходимого оборудования/комплектующих и человеческих ресурсов, в том числе и сроков для реализаии проекта / Assessment of the necessary equipment/components and human resources including the time frame for the project implementation Предоставление данных для расчета TND / Providing data for calculating TND	Предварительная оценка необходимого оборудования/комплектующих и человеческих ресурсов, в том числе и сроков для реализаии проекта / Preliminary estimate of the required equipment and human resources including timing of the project
	Технический директор Эксперт от ДРБ/ *Technical director Business Development Expert*	Подписанный договор/ Signed contract Приказ о начале проекта / The order to start the project Утвержденная базовая версия проекта в 1С-ERP / Approved base version of the project in 1C-ERP	Контакт подписан? /Решение Нет/No – go to step 20 Yes / Да Согласование ролей/ Definition of roles: *Менеджер проекта, Технические эксперты/ Менеджеры комплексных работ, Системный инженер, Менеджер по испытаниям* *Project Manager, Work package Managers/Technical experts, System engineer, Test Manager*	Приказ о назначении менеджера проекта/ Project management assignment order Электронное письмо с поручением работы или поручение составленное через 1С-ERP на каждую роль

	Менеджер проекта Системный инженер Менеджеры по пакетам работ/ *Work package managers Project Manager System engineer*	Техническая спецификация Содержание работ / *Technical specification Statement of Work*		Описание комплексных работ Структура разбивки работ/ Work Package Description Work Breakdown Structure (WPD, WBS)
	Менеджер проекта Системный инженер Менеджеры комплексных работ/ *Work package managers Project Manager System engineer*	Техническая спецификация / Technical specification		Пакет документов этапа предварительного проектирования/ The package of documents of the preliminary design phase (Design docs, Drawings, Design report etc.)
	Менеджер проекта Системный инженер Менеджеры комплексных работ Заказчик/ *Work package managers Project Manager*	Пакет документов этапа предварительного проектирования/ The package of documents of the preliminary design phase (Design docs, Drawings,		Замечания по пакету док-в Акты согласования RIDs Acts of approval

	System engineer Customer	Design report etc.)		
	Менеджер проекта Системный инженер Менеджеры комплексных работ/ *Work package managers Project Manager System engineer*	Замечания по пакету док-в Акты согласования RIDs Acts of approval		Пакет документов этапа детального проектирования/ The package of documents of the detailed design phase (Detailed Design docs, Drawings, Design report etc.)
	Менеджер проекта Системный инженер Менеджеры комплексных работ Заказчик/Технический директор/ Technical director *Work package managers Project Manager System engineer*	Пакет документов этапа детального проектирования/ The package of documents of the detailed design phase (Detailed Design docs, Drawings, Design report etc.)		Замечания по пакету док-в Акты согласования RIDs Acts of approval

	Customer			
	Менеджер проекта Системный инженер Менеджеры комплексных работ Менеджер производств.участка Менеджер по закупкам Финансовый контролер/ *Work package managers* *Project Manager* *System engineer* *Manufacturing manager* *Procurement Manager* *Financial controller*	Пакет документов этапа детального проектирования/ The package of documents of the detailed design phase (RIDs, Detailed Design docs, Drawings, Design report etc.)	Инициализация закупа комплектующих Создание внутреннего заказа на пр-во комплектующих Разработка плана сборки и интеграции/ Purchase of components or Internal order for the production of components Development of an assembly and integration plan	Запрос на закуп Запрос на внутренний заказ План сборки и интеграции/ Assembly and integration plan Request to purchase Internal order

	Менеджер проекта Системный инженер Менеджеры комплексных работ Менеджер по испытаниям Менеджер по закупкам/ *Test Manager Work package managers Project Manager System engineer Manufacturing manager Procurement Manager*	План сборки и интеграции/ Assembly and integration plan	Поставка комплектующих Сборка и интеграция Разработка плана испытаний/ *Supply of components* *Assembly and integration* *Test plan development*	Отчет по сборке и интеграции План испытаний / Assembly and integration report Test plan
	Менеджер проекта Системный инженер Менеджеры комплексных работ/Заказчик / Технический директор/ Technical director	Отчет по сборке и интеграции План испытаний/ Assembly and integration report Test plan	Рассмотрение этапа готовности к испытаниям/ Review of the test readiness phase No Go to step 11 Готов к испытаниям? / Ready for testing?	Замечания по пакету док-в Акты согласования RIDs Acts of approval

	Work package managers Project Manager System engineer/Customer			
	Менеджер проекта Системный инженер Менеджеры комплексных работ Менеджер по испытаниям / *Work package managers Project Manager System engineer Test Manager*	Замечания по пакету док-в Акты согласования План испытаний / RIDs Acts of approval Test plan		Test report MoMs of tests reviews
	Менеджер проекта Системный инженер Менеджеры комплексных работ Менеджер по испытаниям	Test report MoMs of tests reviews		Замечания по пакету док-в Акты согласования RIDs Acts of approval

	Заказчик/ Технический директор/ Technical director *Work package managers Project Manager System engineer Test Manager Customer*			
	Менеджер проекта Системный инженер Менеджеры комплексных работ Менеджер по испытаниям Заказчик/ *Work package managers Project Manager System engineer Test Manager Customer*	Пакет проектной документации/ Package of project documentation		Акты приема-передачи продукции/ Acts of receiving the products Акт приемки этапа (при необходимости)/ Act of acceptance of the stage

	Менеджер проекта Системный инженер Заказчик Технический директор/ Technical director *Project Manager System engineer Customer*	Акты приема-передачи продукции/ Acts of receiving the products	Закрытие проекта Closing the project	Закрытие проекта в 1С-ERP / Closing the project in 1С-ERP

3. РАЗРАБОТКА РЕКОМЕНДАЦИЙ ПО РЕАЛИЗАЦИИ ПРОЕКТА СОЗДАНИЯ ТЕХНОЛОГИЧЕСКОГО ОБЕСПЕЧЕНИЯ ИСПЫТАНИЙ ПСКН

3.1 Анализ существующих программно-технических средств испытания ПС

Испытания в *среде программно-аппаратного комплекса* наиболее естественно формируют реальные условия функционирования ПО, т.к. все испытания проводятся на реальных аппаратных средствах и результат работы виден сразу.

Но проблема возникает тогда, когда возникает необходимость в имитации неправильной работы аппаратного обеспечения так, чтобы было зафиксировано обнаружение отказа аппаратного обеспечения. Конечно, обеспечить это возможно путем создания специального аппаратного испытательного макета с возможностью внесения неисправностей в аппаратные средства.

Но даже создав подобные аппаратные средства испытания, невозможно смоделировать некоторые виды отказов в силу их природы крайне сложно.

Хорошим примером подобного устройства может служить сетевой адаптер, механическое воздействие на который может привести к его полному выходу из строя и мало полезно с точки зрения испытания ПО, поскольку такой грубый вид отказа не составляет труда обнаружить, а проектирование ПО с возможностью обнаружения лишь таких отказов является потенциально опасным и маскирует другие виды отказов.

Более вероятно, что адаптер переменного тока и другие аппаратные и программные средства выходят из строя в результате небольших изменений в логике их работы из-за электромагнитных и температурных воздействий, которые не могут быть обнаружены. Есть случаи, когда пропуск флага во внутреннем регистре в результате незначительного перегрева последнего приводил к бесконечной отправке одного и того же пакета данных вместо другого, и это не было обнаружено программой мониторинга. Такие случаи не являются чем-то необычным и потенциально опасным, а также серьезно влияют на надежность всей системы.

Самым распространенным методом испытаний с использованием программно-аппаратного комплекса является метод программных закладок.

При использовании этого метода в программное обеспечение вносятся некоторые специфические изменения, которые, в свою очередь, могут привести к небольшим корректировкам программного обеспечения логики, а также привести к большим возможным ошибкам или маскировке существующих.

Рис. 3.1.1 – Общая организация ПО и аппаратных средств

Общая роль программных закладок сводится к провокации механизмов самотестирования ПО, а контроль корректности работы этих самых механизмов по-прежнему зависит от человеческого фактора.

Исходя из общей организации ПО и аппаратных средств, приведенной на рис. 3.1.1, последовательность действий при испытаниях ПО с использованием программных закладок примерно следующая:

1) оценка архитектуры ПО и выбор мест для внесения закладок;
2) внесение изменений в исходный код ПО (установка закладок);
3) оценка влияния изменений на логику ПО;
4) проведение испытаний;
5) анализ результатов испытаний.

Как видно из процедур действий при тестировании программного обеспечения, этот подход не является гибким с точки зрения экспертизы и тестирования. При тестировании этого метода возникает много вопросов с точки зрения корректировки модифицированного кода, оценки его влияния на логику программного обеспечения и анализа результатов.

Отсутствие доказательств того, что происходит в программном обеспечении во время тестирования, а также не включает уверенность в правильном и правильном написании всех тестов.

Кроме того, этот метод не позволяет рассмотреть незапланированную ситуацию, появившуюся как следствие воздействия на ПО с программными закладками, - это приводит к повторному ходу действий, проделанных при подготовке к испытаниям, что означает использование дополнительного времени и задержку ввода ПО в эксплуатацию.

*Программно-испытательный компле*кс (ПИК) – это набор программных инструментов, позволяющих проводить экспертизу и испытания ПО без его исполнения на целевой аппаратной платформе. ПО подвергается испытаниям с использованием моделей аппаратного обеспечения или его симуляцией на уровне программных интерфейсов.

Концептуально ПИК состоит из следующих структурных блоков:

- средства визуализации (СВ)
- средства управления (СУ);
- средства регистрации результатов (СР);
- средства внесения искажений в код (СВИК);
- средства внесения искажений в данные (СВИД);
- программное обеспечение контрольно-испытательное (ПО КИ);
- адаптация элементов программного комплекса под конкретное ПО КИ (адаптация для ПО КИ);
- средства симуляции аппаратных средств.

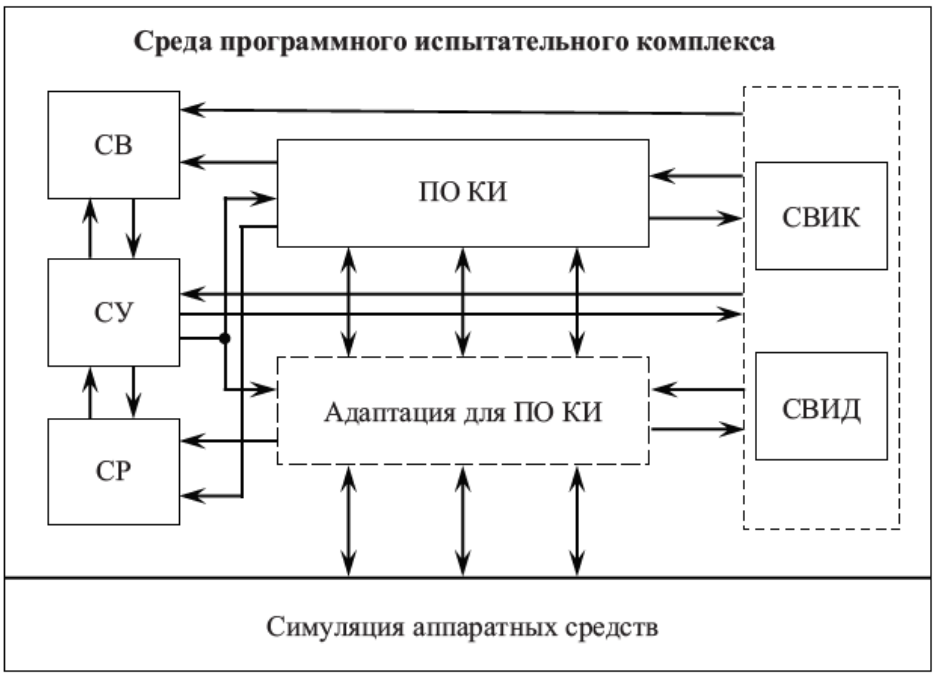


Рис. 3.1.2 – Средства ПИК

Архитектура программного обеспечения испытания группы позволяет испытать программное обеспечение, в полной готовности в лабораторных условиях. Это является очень важным фактором при изучении, так как оценка на различных уровнях удаления в условиях эксперт модульной организации.

При использовании этого подхода для решения проблем, значительно увеличивается качества компонент наблюдения. Дизайн испытания не трудоемкий процесс, который требует множество повторений, - все испытания проводятся в тестовой среде программного обеспечения, которое позволяет вам контролировать ряд точек испытания программного обеспечения и записывать необходимые аспекты работы, исходя из них, мы можем оценить правильность реализации механизмов безопасности.

Как видно, программно-испытательный комплекс позволяет более гибко подойти к испытаниям ПО и не вносить в него изменения, которые могли бы повлиять на логику его исполнения. Также нет необходимости проводить повторный анализ испытуемого ПО после адаптации его к исполнению в среде ПИКа, поскольку логика ПО не претерпевает никаких изменений.

3.2 Разработка рекомендаций по выбору отдельных элементов системы технологического обеспечения испытания ПСКН

Рекомендации по выбору отдельных элементов системы строятся на основе определения основных критериев выбора:

1. Качество;
2. Цена;
3. Надежность;
4. Эффективность.

Рекомендуется использовать средства измерения, которые были утверждены в государственном реестре.

Важно помнить о различиях требований к средствам испытаний, поэтому при приобретении импортных средств измерений нужно учитывать этот факт.

Выбирая средства оценки безопасности ПО необходимо обращать внимание на программы, обеспечивающие выполнение таких действий как:

- аппаратно-программной среды компьютера, на котором она запущена, формирование на основе этого анализа текущих характеристик своей среды выполнения;
- проверка подлинности среды выполнения путем сравнения ее текущих характеристик с эталонными, хранящимися на винчестере;
- блокирование дальнейшей работы программы при несовпадении текущих характеристик с эталонными.

Персонал должен разрабатывать варианты реализации изменений и документально оформлять:

- сообщение о каждом дефекте или заявку на внесение изменений;

- результаты их анализа и варианты реализации изменений;
- оценку их влияния на функциональную пригодность.

Следует согласовывать с заказчиком выбранные варианты изменений требований в соответствии с договором. Регистрация и учет истории этого процесса обеспечивает возможность его контроля и пошагового восстановления выполненных изменений (отката) документов для выявления вторичных дефектов, внесенных в процессе разработки очередной версии программного продукта. Такие дефекты обычно обусловлены одновременным, не скоординированным внесением групп изменений несколькими специалистами или потерей некоторых корректировок в определенной версии.

В процессах разработки и испытаний программного продукта может участвовать большое число специалистов различных направлений и квалификации, которые при необходимости целесообразно объединять в единый коллектив службу сертификационных испытаний на соответствие требованиям и управления конфигурацией программного продукта. Менеджер проекта является высшим должностным лицом, принимающим важнейшие решения по внесению изменений и корректировке требований и конфигурации сложных комплексов программ. Он взаимодействует с заказчиком и пользователями, определяющими модификации, для согласования изменений требований к системе. Заказчик системы должен оценивать и утверждать наиболее крупные изменения, заметно влияющие на условия контракта, технические требования или стоимость программного продукта. При организации сложного проекта следует идентифицировать инстанцию, уполномоченную утверждать версии программных продуктов и документов для поставки заказчику и пользователям, и любые изменения к ним.

Перечисленные выше специализации и квалификации персонала, участвующего в испытаниях программных продуктов, требуют соответствующего их отбора, подготовки и обучения. Должны быть разработаны и документированы в плане проекта, требования и цели обучения специалистов,

а также разработаны руководства, включая документы, используемые для обучения (см. ISO 19759 и ISO 15504). Персонал, ответственный за выполнение конкретных задач испытаний программных продуктов с высоким риском должен быть аттестован, если это необходимо, на основе соответствующего образования, подготовки и/или опыта работы. Может также стать необходимым включение в подготовку специалистов, ознакомление со специфической (проблемно-ориентированной) областью, в которой будет работать данный программный продукт, и повышение квалификации в этой области. Требования к квалификации и обучению персонала должны быть документально оформлены в контракте на проект.

4. ЭКОНОМИЧЕСКАЯ ЧАСТЬ

4.1. Принципы оценки экономической эффективности испытания ПС

В экономической части проекта целью является изложение содержания проекта для обоснования его возможностей, целесообразности и демонстрации основных преимуществ.

Обязательная сертификация необходима для программных продуктов и их производства, выполняющих особо ответственные функции, в которых недостаточное качество, ошибки или отказы могут нанести большой ущерб или опасны для жизни и здоровья людей.

Этот ущерб может определяться степенью безопасности применения комплексов программ в авиации, для управления в космосе, в атомной энергетике, в военных системах; или большими экономическими потерями в результате низкого качества функционирования ПС в системах государственного управления, в финансовых, банковских, транспортных системах. В подобных системах обязательная сертификация программных продуктов способствует значительному снижению риска заказчика и повышению безопасности функционирования программного продукта у потребителя до необходимого уровня. В этих случаях разработчики и поставщики программного продукта обязаны подвергать свои изделия сертификации на соответствие требованиям качества и безопасности для получения разрешения компетентных органов на их реальную эксплуатацию и применение по прямому назначению.

Экономическую рентабельность обязательной сертификации количественно определить чаще всего невозможно. Ее эффект сосредоточен в повышении таких показателей качества систем, как адекватность функционирования, надежность, своевременность представления информации, полнота, достоверность, конфиденциальность, безопасность применения программного продукта, которые зачастую трудно представить и оценить конкретными экономическими категориями. Необходимость проведения

обязательной сертификации, как правило, определяет заказчик или потребитель программного продукта для получения формальных гарантий достижения производителем заданных значений показателей качества и безопасности продукта. Он же может выступать в качестве заявителя при обращении к сертификационной лаборатории на выполнение испытаний, а также участвовать в формулировании требуемых показателей и в контроле их измерений при испытаниях. Соответственно заявитель финансирует испытания и получает документы, регистрирующие их результаты, в том числе сертификат соответствия при положительных результатах. В конечном итоге эти затраты отражаются на цене продукта, однако они первоначально вкладываются заказчиком или потребителем как дополнительная часть стоимости создания заказанного продукта. Роль разработчика состоит в устранении выявленных недостатков для достижения заданного требованиями уровня качества и безопасности. Если результаты испытаний не позволяют сертифицирующей лаборатории и органу дать положительное заключение, то проверенная продукция возвращается разработчикам для доработки и предъявления на повторные испытания.

Ресурсы для сертификации программных средств и систем качества должны выделяться в зависимости от характеристик объекта та сертификации или процесса. В результате сложность комплексов программ, а также доступные ресурсы становятся косвенными критериями или факторами, влияющими на выбор методов испытаний и на достигаемое их качество. При этом следует учитывать, что каждый вид ресурсов в реальных условиях может варьироваться в некотором диапазоне и в той или иной степени ограничен. Наиболее общим видом ресурсов, используемых при испытаниях, являются допустимые временные и финансовые затраты или сметная стоимость сертификации. При анализе этот показатель может применяться или как вид ресурсных ограничений, или как оптимизируемый критерий. Определяющим ресурсом сертификации является совокупная численность и структура коллектива специалистов, а также его квалификация и подготовленность к коллективной проверке конкретного

типа ПС и компонентов или системы качества. Аппаратурная оснащенность испытателей конкретного ПС определяется, прежде всего, ресурсами и другими характеристиками реализующей и технологической ЭВМ, доступных для использования коллективу специалистов при сертификации.

Исследования ошибок 20 крупных завершенных и поставляемых программных продуктов, созданных в 13 различных организациях, показали, что коллективы специалистов добились среднего уровня 0,06 дефекта на тысячу строк нового и измененного программного кода. В пяти наибольших из них по размеру достигнуто 0,04 – 0,075 ошибок на тысячу строк. Таким образом, уровень ошибок около 0,05 на тысячу строк кода в разных публикациях считается близким к предельному уровню для высококачественных программных продуктов. Уровень ошибок критического проекта особенно высокого качества может служить программный продукт бортовых систем Шатла [19], созданный NASA. В нем содержится менее одной ошибки на 10000 строк кода. Однако стоимость программного продукта достигала 1000 $ за строку кода, что в среднем в сто раз больше, чем для административных систем, и в десять раз больше, чем для ряда критических систем реального времени.

4.2. Анализ и расчет затрат на испытание ПС

Реализация системного проекта ставит перед нами задачи:

1) Расчет трудоемкости стадий разработки;
2) Расчет времени на создание системного проекта;
3) Расходы на оплату труда разработчиков;
4) Расходы на социальные нужды;
5) Расходы на техническое обеспечение разработки и затраты на потребляемую энергию.

Общая трудоемкость подобных проектов рассчитывается по формуле:

$$T_0 = T_H \cdot K_c \cdot K_T \cdot K_H \qquad (4.1)$$

где K_c - коэффициенты, учитывающие, сложности ПО;

K_H – коэффициенты, учитывающие степени новизна ПО;

K_T – поправочная коэффициентов, учитывающие степени использования при тестировании;

T_H – нормативная трудоемкость испытаний ПО.

Для системных проектов:

$K_c = 0{,}12$, $K_T = 0{,}8$, $K_H = 0{,}8$, $\text{Т}_\text{Н} = 112$ чел./дн.

Из этого следует, что общая трудоемкость будет равна:

$$T_0 = 112 \cdot 0{,}12 \cdot 0{,}8 \cdot 0{,}8 = 8\,\frac{\text{чел.}}{\text{дн}}.$$

Эффективность фонда времени работы на одного сотрудника ($\Phi_{\text{эф}}$) рассчитывается по формуле:

$$\Phi_{\text{эф}} = \text{Д}_\text{г} - \text{Д}_\text{п} - \text{Д}_\text{в} - \text{Д}_\text{о} \tag{4.2}$$

где $\text{Д}_\text{г}$ – количество дней в году, 366 дней;

$\text{Д}_\text{п}$ – количество праздничных дней, 48 дней;

$\text{Д}_\text{в}$ – количество выходных;

Д_0 – отпускные дни.

Для этого можно воспользоваться производственным календарем на 2020 год. По плану в 2020 году ожидается 120 выходных дней для 5 дневной рабочей недели.

Стандартное количество отпускных дней – 24 дня.

$\Phi_{\text{эф}} = 366 - 120 - 24 = 222$ дня.

$$\Phi_{\text{эфj}} = 8 \cdot 222 = 1776 \text{ часов}$$

Исходя из численности исполнителей системного проекта и количества рабочих дней составляется численность группы работников. На разработку системного проект T_P потребуется 70 рабочих дней.

$$\text{Ч}_\text{Р} = \frac{T_\text{Н}}{T_P \cdot \Phi_{\text{эф}}} = \frac{112}{0{,}23 \cdot 222} = 2 \text{ чел.} \tag{4.3}$$

Расчет времени на разработку системного проекта

Таблица 4.1 – Установка времени на работу.

Этап работы	Время, ч
Определение требований к системе, анализ литературы	60
Создание технической спецификации	70
Разработка системы	100
Разработка рекомендации	60
Итого	290

Рассчитаем капитальные вложения на реализацию проекта:

$$K_P = K_o + K_{дд} + К_{пп} + К_{св} \quad (4.4)$$

где K_o – затраты на основное и вспомогательное оборудование, тенге;

$K_{дд}$ – затраты на помещение, тенге;

$К_{пп}$ – затраты на пакеты инструментов;

$К_{св}$ – затраты на связь и интернет;

Затраты на основные и вспомогательные оборудования включают балансовые стоимости всех видов оборудования (120000 тенге), количество единицы оборудования (6 шт.).

$$K_o = 120000 * 6 * 0{,}75 = 540\ 000 \text{ тенге}$$

$$K_p = 540000 + 6000 = 546000$$

Суммарные затраты на разработку системного проекта представлены в таблице ниже.

Таблица 4.2 - Затраты

Затраты	Сумма, тенге
Приобретение ПП	128900
Вспомогательное оборудование	220680
Установка и сопровождение	80890
Обучение персонала	105600
Комплектация ТС	70690
Итого	606760

Согласно расчетам обеспечение создания системного проекта технлогического обеспечения составит сумму затрат в 606760 тенге.

Расчет затрат на заработную плату исполнителей проекта.

Исполнители проекта:

- Руководители
- Тестировщики
- Специалисты по стандартизации/сертификации
- Специалисты IT
- IT инженеры

Таблица 4.3 – Средняя заработная плата исполнителей

Руководители проектов	600 000 тенге
Тестировщики	300 000 тенге
Специалисты по стандартизации/сертификации	150 000 тенге
Специалисты IT	250 000 тенге

Операторы	150 000 тенге

Расходы по оплате труда испытателей программного обеспечения ($ЗП_И$) Определяются путем умножения трудоемкости испытаний на среднечасовую зарплату разработчиков $СЧ_И$.

$$ЗП_И = Т \cdot СЧ_И, \quad (4.5)$$

$СЧ_И$ – среднечасовая оплата труда разработчика, тенге в час.

$$СЧ_И = \frac{\sum ЗП}{Ф_t} \quad (4.6)$$

где ЗП – это совокупность заработной платы всех исполнителей проекта.

$Ф_t$ – месячный фонд рабочего времени, при 8-ми часовом рабочем дне и 5-ти дневной рабочей неделе = 168 часов.

На основании трудоемкости по каждому исполнителю можно рассчитать расходы по оплате. Расчет приведен в таблице 3.

Таблица 4.4 – Затраты на оплату

Этап работы	Исполнители	Время, ч	З/п в час, тенге	Итого з/п за время
Определение требований к системе, анализ литературы	Руководитель	60	3571	214260
Создание технической спецификации	Специалист IT	70	1488	104160
Разработка системы	Специалист по стандартизации	80	893	89300
Испытания	Тестировщики	120	1786	214285
Разработка рекомендации	Оператор	60	893	53580

Итого		410	8631	675585

Косвенные затраты по заработной оплате труда составляют около 20% из суммы заработной оплаты.

$$З_К = ЗП \times 20\% = 675585 \times 0{,}2 = 135117 \text{ тенге} \quad (4.7)$$

Тогда фонд оплаты труда составит ФОТ = 675585+135117 = 810702 тенге.

Пенсионные отчисления в РК составляют 10% от ФОТ

Тогда Пенсионные отчисления = 810702*10% = 81070 тенге

Размер социальных отчислений в 2020 году составил 3,5% от ФОТ

Социальные отчисления = 810702*3,5% = 28374,6 тенге.

Расчет затрат на техническое обеспечение проекта

Испытания программного обеспечения требует наличие технических средств, таких как:

- Системный блок компьютера
- Мониторы компьютера
- Гарнитура: мышь, клавиатура
- Программный продукт
- Лицензионные программы

В соответствии с требованиями к параметрам компьютера наиболее приемлемой моделью системного блока является HP 590-P0074UR стоимостью 349 990 тенге [5 technodom.kz]. Для испытаний также были выбраны монитор Samsung LS24F356FHI 1920x1080 диагональю 23.5’’ и ценой 51600 тенге, и комплектующие части на сумму 10 000 тенге.

Рисунок 4.1 – Оборудование

На лицензионные продукты в среднем потребуется 50 000 тенге.

Итого затраты на техническое обеспечение испытаний составляет 461 590 тенге.

Теперь есть возможность рассчитать издержки на амортизационные отчисления, которые рассчитываются по следующей формуле:

$$З_а = С_{бал} * \frac{Н_а}{100} \quad (4.8)$$

Балансовая стоимость ($С_{бал}$):

$$С_{бал} = С_{рын} + З_{пр} \quad (4.9)$$

где $З_{пр}$ – затраты на установку, обычно составляют 3-6% от общей цены компьютера.

$$С_{бал} = С_{рын} + З_{пр} = 401590 + 20079{,}5 = 421669{,}5 \text{ тенге} \quad (4.10)$$

Каждое техническое оборудование имеет срок службы или срок полезного его использования. Его устанавливает организация самостоятельно, с учетом таких факторов, как ожидаемая производительность или мощность, физический износ и нормативные ограничения в использовании.

На практике и в соответствии с классификацией основных средств срок полезного использования компьютера составляет от 2 до 3 лет.

Для учета данных факторов используем линейный метод начисления. Устанавливаем срок в 3 года или 36 месяцев. Поэтому ежемесячная амортизация для организации будет составлять 11 713 тенге.

По результатам расчета можно сделать выводы, что затраты для испытаний ПС составят 1 948 576 тенге.

Цель экономического обоснования проекта достигнута, путем расчета затрат необходимых для организации процесса испытаний программных средств космического назначения. Расчетные данные предоставляют возможность анализа затрат и оценки экономической целесообразности испытаний.

5. САНИТАРНО-ГИГИЕНИЧЕСКАЯ ОЦЕНКА ОСВЕЩЕННОСТИ РАБОЧЕГО ПОМЕЩЕНИЯ

5.1 Расчет естественного освещения помещения

Испытания программных средств предполагает обеспечение благоприятных условий на рабочем месте для специалистов.

В плане учета данного положения работа специалистов ведется в помещении офисного типа здания предприятия. Параметры типового помещения представлены в таблице 5.1.1.

Таблица 5.1.1. Параметры помещения

Длина L, м	8,4
Ширина B, м	4,5
Высота H, м	3,4
Высота окна $h_{ок}$, м	2,6
Высота от пола до окна $l_{ок}$, м	0,8
Количество окон $n_{ок}$	4
Коэффициент отражения потолка $\rho_{пот}$	50
Коэф. отражения стен $\rho_{стен}$	50
Коэф. отражения пола $\rho_{пол}$	10
Световой пояс	Алматы
Высота здания $H_{ЗД}$, м	20
Расстояние до рядом стоящего здания P, м	50

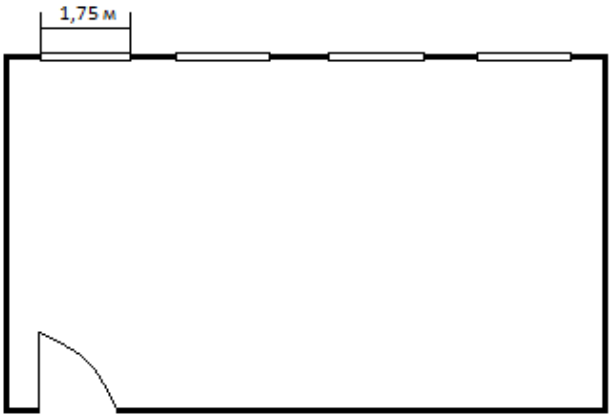


Рисунок 5.1.1 – Параметры оконных проемов помещения.

Как правило расчет начинается с определения площадей световых проемов исходя из нормативных параметров.

При боковом освещении формула для расчета выглядит следующим образом:

$$100 \cdot \frac{S_0}{S_п} = \frac{e_н \cdot K_з \cdot \eta_0}{\tau_0 \cdot r_1} \cdot K_{зд} \quad (5.1)$$

В данной формуле So – площадь световых проемов при боковом освещении, которую нам необходимо в итоге определить.

Площадь пола помещения $S_п = L \cdot B = 8{,}4 \cdot 4{,}5 = 37{,}8$ м^2

Коэффициент естественного освещения (КЕО) $e_Н = e \cdot m_N$

где N определяет разряд зрительных работ;

e - значение КЕО (при боковом равняется 1,5 [10]);

Оконные проемы в стенах с учетом местоположения и ориентации на стороны света дают коэффициент светового климата $m_N = 0{,}7$[11].

На экране компьютера самыми мелкими объектами различения являются буквы и цифры текста толщиной от 0,5 до 1 мм, что соответствует средней точности. Контраст объекта различения с фоном - средний. Фон можно определиь как средний. При таких условиях работа на компьютере относится к разряду зрительных работ Б-1 [3].

$$e_Н = e \cdot m_N = 1{,}5 \cdot 0{,}7 = 1{,}05 \quad (5.2)$$

В качестве светопропускающего материала используются органические прозрачные стекла, коэффициент светопропускания τ_1 которых составляет 0,9 [13].

Для учета потери света в переплетах светопроема принимается $\tau_2 = 0{,}75$, ввиду того, что переплет одинарные открывающиеся [13].

τ_3 - коэффициент, учитывающий потери света в несущих конструкциях ($\tau_3 = 1{,}0$) [13].

Для защиты от солнца используются регулируемые жалюзи. В таком случае коэффициент τ_4, учитывающий потери света в солнцезащитных устройствах, равен 1,0 [4].

Таким образом общий коэффициент светопропускания материала:

$$\tau_0 = \tau_1 \cdot \tau_2 \cdot \tau_3 \cdot \tau_4 = 0{,}9 \cdot 0{,}75 \cdot 1 \cdot 1 = 0{,}68 \quad (5.3)$$

Средневзвешенный коэффициент отражения потолка, стен и пола:

$$\frac{\rho_{пот}+\rho_{стен}+\rho_{пол}}{3} = \frac{50+50+10}{3} = 36{,}7\% \quad (5.4)$$

Следовательно, r_1=1,1 [2].

$К_З$ – коэффициент запаса ($К_З$ = 1.4 [2]).

$К_{ЗД}$ – коэффициент, учитывающий затемнение окон противостоящими зданиями = 1.0, т.к. Р/ НЗД больше 3 [10].

Световая характеристика окон определяется соотношением глубины комнаты к его высоте от уровня условной рабочей поверхности до верха окна

$$\frac{L}{B} = \frac{8{,}4}{4{,}5} = 1{,}87 \quad (5.5)$$

Уровень условной рабочей поверхности до верха окна составляет

$$h_{расч} = h_{ок} + h_{н.ок} - H_{стола} = 2{,}6 + 0{,}5 - 1{,}75 = 1{,}35 \text{ м} \quad (5.6)$$

Из этого следует

$$\frac{B}{h_{расч}} = \frac{4{,}5}{2{,}1} = 2{,}14 \quad (5.7)$$

Тогда световая характеристика окон по таблице будет равна 9,5 [11].

Таким образом, мы можем применять для расчета следующие нормы (таблица 1.2.1)

Таблица 5.1.2 - Показатели

Площадь пола помещения $S_п$, м^2	37,8
Нормируемое значение КЕО $e_Н$	1,05
Коэффициент запаса $К_З$	1,4
Коэффициент $К_{ЗД}$	1
Световая характеристика η_0	9,5
Общий коэффициент светопропускания τ_0	0,68
Коэффициент учета повышения КЕО r_1	1,1

Рассчитаем площадь боковых проемов при естественном освещении по следующей формуле:

$$S_0 = \frac{S_п \cdot e_н \cdot K_з \cdot \eta_0 \cdot K_{зд}}{100 \cdot \tau_0 \cdot r_1} \quad (5.8)$$

$$S_0 = \frac{37{,}8 \cdot 1{,}05 \cdot 1{,}4 \cdot 9{,}5 \cdot 1.0}{100 \cdot 0{,}68 \cdot 1{,}1} = 7{,}06 = 7{,}1 \text{ м}^2$$

Расчет показывает, что допустимое значение площади светового проема не должно быть меньше 7 м^2.

Помещение данного типа имеет 4 окна размерами 1,75×2,6 м. Ввиду этого для обеспечения нормируемой освещенности площадь светового проема будет 7 м^2, что означает приемлемость выбранных параметров световых проемов.

5.2. Расчет искусственного освещения помещения

Цель данного расчета состоит в определении количества средств обеспечения искусственного освещения на рабочем месте и их взаимное расположение.

Коэффициент использования светового потока зависит от типа выбранного светильника, индекса помещения и коэффициента отражения поверхностей помещения.

В качестве средств обеспечения искусственного освещения были использованы светильники для офисных помещений DS office 30.

В соответствии с разрядом работ Б-1, при среднем фоне и контрасте, значение общей искомой освещённости по СниП для ламп накаливания Ен = 300 лк [12].

Индекс помещения:

$$\mathrm{i} = \frac{\mathrm{L \cdot B}}{(\mathrm{h_p} \cdot (\mathrm{L+B})} = \frac{8{,}4 \cdot 4{,}5}{1{,}35 \cdot (8{,}4+4{,}5)} = 2{,}17 \qquad (5.9)$$

В таком случае коэффициент использования светового потока η = 40 % [11].

Светильники DS office 30 состоят из 4 светодиодов. Поэтому общий коэффциент F=3540 лм для каждого светильника будет равен 885 лм.

Тогда обеспечения освещенности помещения потребуется данное количество светодиодов:

$$\mathrm{N} = \frac{\mathrm{E \cdot S \cdot Z \cdot K_З}}{\mathrm{F \cdot \eta} \cdot n} = \frac{300 \cdot 37{,}8 \cdot 1{,}1 \cdot 1.4}{885 \cdot 0{,}4 \cdot 4} \approx 12 \qquad (5.10)$$

где S – площадь пола 37,8 м^2;

n – количество ламп в светильнике;

Z – коэффициент показывающий неравномерность освещения, определяется в пределах 1,1 ÷ 1,2 [11].

Высота расположения светильников над рабочими поверхностями определяется по формуле

$$\mathrm{Hc} = H - h_{\mathrm{CB}} - h_{\mathrm{P}} = 3{,}4 - 0{,}3 - 1{,}35 = 1{,}75 \text{ м} \qquad (5.11)$$

где H – общая высота помещения, м;

h_{CB} – высота от потолка до нижней части светильника, м;

h_{P} – высота от пола до освещаемой поверхности, м.

Высоту подвеса их над уровнем пола устанавливают не менее 3-6 м при лампах мощности большей 300 Вт, в целях уменьшения ослепляющего воздействия [11].

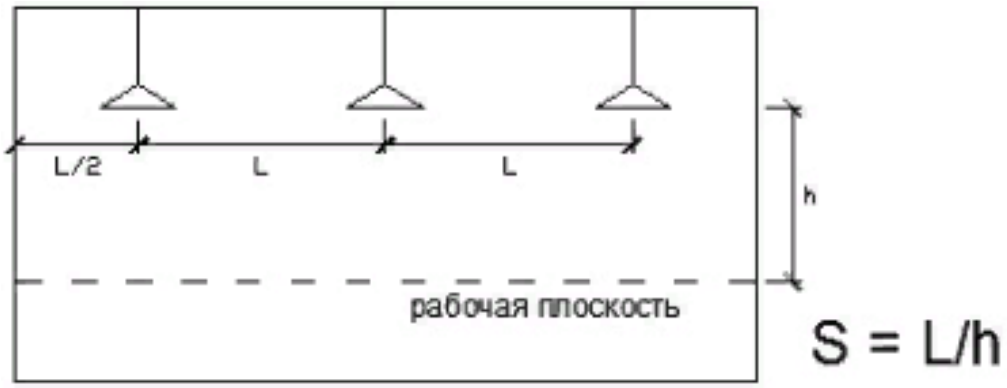


Рисунок 3 – Условная схема высоты подвеса средств освещения.

Теперь мы можем определить расстояние между светильниками L. Показатель λ при расположении светильников прямоугольником определяется в пределах 0,5÷2,0 [11].

$$L_A = \lambda \cdot h_1 = 1{,}5 \cdot 1{,}35 = 2{,}03 \text{ м}$$

$$L_B = \lambda \cdot h_1 = 1 \cdot 1{,}35 = 1{,}35 \text{ м}$$

$$l_A = 0.5 \cdot L_A = 0{,}5 \cdot 2{,}03 = 1{,}015 \text{ м}$$

$$l_B = 0.5 \cdot L_B = 0{,}5 \cdot 1{,}35 = 0{,}675 \text{ м}$$

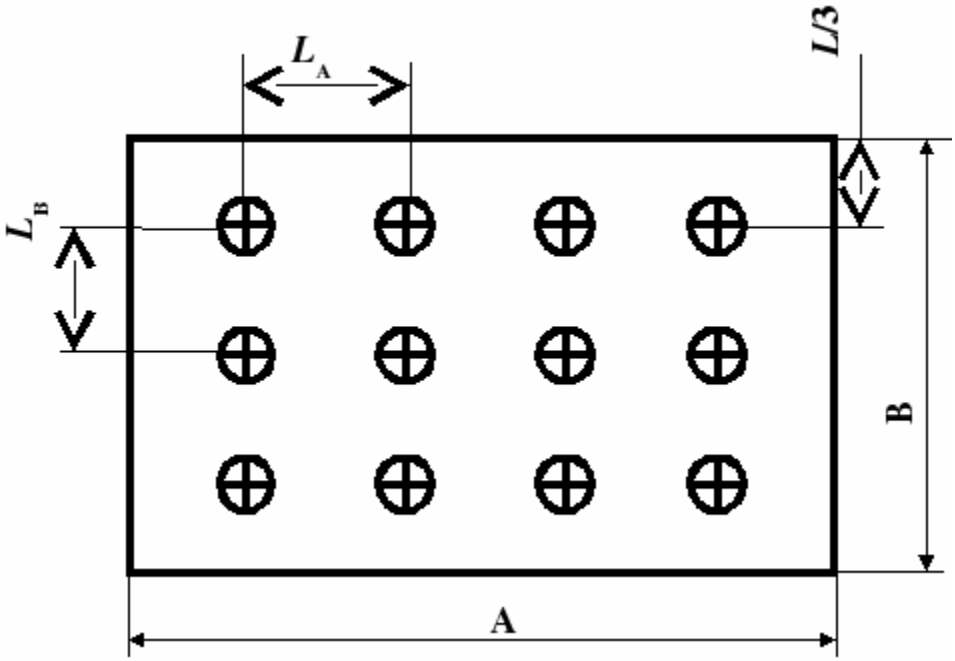


Рисунок 4 – Схема расположения светильников

Выбранное количество светильников установленным нормам обеспечения приемлемого уровня освещенности удовлетворяет.

ЗАКЛЮЧЕНИЕ

На основе проведенной работы можно сделать вывод о том, что испытания программных средств имеет актуальное значение в сфере космической техники в связи с особенностью их эксплуатации. В работе были рассмотрены:

- Особенности испытаний программных средств космического назначения;
- Виды испытаний;
- Средства испытаний;
- Процесс испытаний;
- Понятие системного проекта и его задачи.

В виду этого при выполнении работы были выполнены следующие задачи:

- Исследование предметной области испытания ПСКН;
- Анализ требований к технологическому обеспечению испытаний;
- Разработка системного проекта технологического обеспечения испытаний;
- Оценка реализации проекта.

Разработка системного проекта технологического обеспечения испытаний ПСКН представляется трудоемким процессом, который требует затрат многих видов ресурсов. Средства технологического обеспечения имеют большое многообразие в выборе и специфике в настоящее время, модели программно-технических комплексов в свою очередь предоставляют возможность автоматизации трудоемких процессов на этапах испытаний. Квалификация специалистов играет основную роль в обеспечении процесса, и потому требует соответственной суммы затрат.

Но нельзя отметить, что при должном выполнении всех требований и тщательной подготовки системного проекта затраты в полной мере оправдывают качество испытаний ПСКН.

Список использованной литературы

1. Липаев В.В. Тестирование компонентов и комплексов программ. Учебник. – М.: СИНТЕГ, 2010. – 400 с.

2. Липаев В.В. Программная инженерия. Методологические основы. Учебник. –М.: Издательство «ТЕИС», 2006. – 609 с.

3. Липаев В.В. Экономика производства программных продуктов. Издание второе. – М.: СИНТЕГ. 2011

4. О. А. Наседкин, А. М. Белоус Средства испытания программного обеспечения микропроцессорных СЖАТ. - (https://cyberleninka.ru/article/n/sredstva-ispytaniya-programmnogo-obespecheniya-mikroprotsessornyh-szhat/viewer)

5. http://www.protesting.ru/automation/functional/testtool.html

6. ECSS-Q-HB-80-03-12 Space product assurance. Software dependability and safety (Гарантия космического продукта. Надежность и безопасность программного обеспечения). - (http://www.dit.upm.es/~str/proyectos/upmsat2/documentacion/ECSS-Q-HB-80-03.pdf)

7. ECSS-Q-80-03-2006 Space Product Assurance – Methods and Techniques to Support the Assessment of Software Dependability and Safety. - (http://www.dit.upm.es/~str/proyectos/upmsat2/documentacion/ECSS-Q-HB-80-03.pdf).

8. СНиП РК 2.04-05-2002 «Естественное и искусственное освещение» - Астана, 2004 – 15 с.

9. Абдимуратов Ж.С., Мананбаева С.Е. Безопасность жизнедеятельности. Методические указания к выполнению раздела «Расчет производственного освещения» в выпускных работах для всех специальностей. Бакалавриат - Алматы: АИЭС, 2009. - 20 с.

10. «Гигиенические требования к персональным электронно-вычислительным машинам и организации работы». Санитарно-эпидемиологические правила и нормативы СанПиН 2/2.4.1340;

11. СН РК «Естественное и искусственное освещение», Астана 2011 – 23 с.

12. Е.Л. Кантора. Экономика промышленного предприятия. Учебник. – М. 2007

13. З.Д. Еркешева, Г.Ш. Боканова. Методические указания к выполнению экономической части дипломных работ для студентов специальности 5В070400 – Вычислительная техника и программное обеспечение. – Алматы: АУЭС, 2013 – 40 с.

14. ГОСТ ИСО/МЭК 12119-2000 «Информационная технология. Пакеты программ. Требование к качеству и тестирование».

15. «О системе оплаты труда гражданских служащих, работников организаций, содержащихся за счет средств государственного бюджета, работников казенных предприятий» Постановление Правительства Республики Казахстан от 29 декабря 2007 года N 1400.

16. Казарин О.В. Надежность программного обеспечения – 2018 г.

17. Исмаил Е.Е. Показатели надежности программных средств космического назначения //Вестник Алматинского университета энергетики и связи. - №3 (34), 2016. - с. 27-35

18. СТ РК ECSS M-ST-10-01C -2017 Менеджмент космического проекта. Организация и проведение рассмотрений.

19. СТ РК ECSS-Q-ST-80C-2012 Гарантия космической продукции. Гарантия качества программного обеспечения [текст]. – Введ. 2013–07–01. - Астана, 2012. – 98 с.: ил.

20. ГОСТ Р ИСО/МЭК 25040-2014 Требования и оценка качества систем и программного обеспечения (SQuaRE). Процесс оценки. Определения [Текст]. – Введ. 2015-06-01. – М.: Стандартинформ, 2014.

21. ECSS-Q-HB-80-03A: 2012 Space product assurance. Software dependability and safety.

22. ECSS-Q-ST-80C Rev.1: 2017 Space product assurance. Software product assurance.

23. ISO/IEC 25001:2014 Systems and software engineering - Systems and software Quality Requirements and Evaluation (SQuaRE) - Planning and management.