

**ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ
НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ
«ВЫСШАЯ ШКОЛА ЭКОНОМИКИ»
Факультет компьютерных наук
Образовательная программа «Программная инженерия»
(ВШЭ ФКН ПИ)**

СОГЛАСОВАНО

Доцент департамента
Программной инженерии,
ФКН, к.т.н.

_____ К. Ю. Дегтярёв
«_____» _____ 20__г.

УТВЕРЖДАЮ

Академический руководитель
образовательной программы
«Программная инженерия»
старший преподаватель

_____ Н. А. Павлов
«_____» _____ 20__г.

**ПРОГРАММА ДЛЯ МОДЕЛИРОВАНИЯ ВОСПРИЯТИЯ
ФАКТОРОВ УСПЕХА IT-ПРОЕКТА С ИСПОЛЬЗОВАНИЕМ
НЕЧЕТКИХ КОГНИТИВНЫХ КАРТ**

Программа и методика испытаний

ЛИСТ УТВЕРЖДЕНИЯ

RU.17701729.10.03-01 ПМИ 01-1-ЛУ

Исполнитель

Студент группы БПИ204
образовательной программы
«Программная инженерия»
Пеганов Никита Сергеевич

_____ Н. С. Пеганов
«_____» _____ 20__г.

УТВЕРЖДЕН
RU.17701729.10.03-01 ПМИ 01-1-ЛУ

**ПРОГРАММА ДЛЯ МОДЕЛИРОВАНИЯ ВОСПРИЯТИЯ
ФАКТОРОВ УСПЕХА ИТ-ПРОЕКТА С ИСПОЛЬЗОВАНИЕМ
НЕЧЕТКИХ КОГНИТИВНЫХ КАРТ**

Программа и методика испытаний

RU.17701729.10.03-01 ПМИ 01-1-ЛУ

Листов 26

Содержание

1	Введение	4
1.1	Цели и задачи испытаний	4
1.2	Область применения	4
1.3	Нормативные ссылки	4
1.4	Структура документа	4
2	Цели и задачи испытаний	5
2.1	Цели испытаний	5
2.2	Задачи испытаний	5
3	Общие положения	6
3.1	Основные понятия и определения	6
3.2	Общие требования к программе испытаний	6
4	Анализ требований	7
4.1	Требования к функциональности	7
4.2	Требования к производительности	7
4.3	Требования к пользовательскому интерфейсу	7
4.4	Требования к безопасности	7
4.5	Требования к совместимости	8
5	Подготовка к испытаниям	9
5.1	Испытательная среда	9
5.2	Средства и инструменты испытаний	9
5.3	Подготовка тестовых данных	9
5.4	Настройка окружения	9
6	Методика испытаний	10
6.1	Методика функционального тестирования	10
6.2	Методика тестирования на совместимость	10
6.3	Методика тестирования безопасности	10
7	План испытаний	11
7.1	График проведения испытаний	11
7.2	Обязанности тестировщика	11
7.3	Критерии начала и завершения испытаний	12
7.4	Условия прерывания испытаний	12
8	Документация результатов испытаний	13
8.1	Форматы отчётов	13
8.2	Шаблоны журналов	13
8.3	Процедуры записи и хранения результатов	14
9	Функциональные тесты	15
9.1	Сценарии тестирования функциональности	15
9.1.1	Сценарий 1: Создание нового проекта	15
9.1.2	Сценарий 2: Добавление фактора	15
9.1.3	Сценарий 3: Добавление связи между факторами	15
9.1.4	Сценарий 4: Сохранение проекта	15
9.1.5	Сценарий 5: Импорт проекта	16
9.1.6	Сценарий 6: Выполнение анализа НКК	16
9.1.7	Сценарий 7: Удаление сущности (фактора или связи)	16
9.2	Анализ совпадений и расхождений	16

10 Тесты на совместимость	17
10.1 Цели и задачи тестирования	17
10.2 Методика тестирования	17
10.3 Результаты тестирования	17
10.4 Выводы и рекомендации	18
11 Анализ и интерпретация результатов	19
11.1 Результаты функционального тестирования	19
11.2 Результаты нагрузочного тестирования	19
11.3 Результаты тестирования на совместимость	19
11.4 Результаты тестирования безопасности	19
11.5 Заключение	20
11.5.1 Рекомендации	20
12 Заключение	21
12.1 Основные выводы	21
12.2 Рекомендации	21
12.3 Завершение испытаний	21
13 Список использованных источников	22
Приложения	24

1 Введение

1.1 Цели и задачи испытаний

Целью данного документа является описание программы и методики испытаний для программы, предназначенной для моделирования восприятия факторов успеха IT-проекта с использованием нечетких когнитивных карт (НKK). Испытания направлены на проверку корректности функционирования, производительности, безопасности, и совместимости программы согласно установленным требованиям.

1.2 Область применения

Документ применяется к версии программы, предназначенной для моделирования восприятия факторов успеха IT-проекта. Определены этапы подготовки и проведения испытаний, а также способы документирования и анализа результатов. Настоящая методика испытаний предназначена для использования как разработчиками, так и специалистами по тестированию.

1.3 Нормативные ссылки

В документе используются следующие нормативные ссылки:

- ГОСТ 34.601-90 «Автоматизированные системы. Стадии создания»[1]
- ISO/IEC 25010:2011 «Systems and software engineering – Systems and software Quality Requirements and Evaluation (SQuaRE) – System and software quality models»[2]
- IEEE Standard 829-2008 «IEEE Standard for Software and System Test Documentation»[3]

1.4 Структура документа

Настоящий документ состоит из следующих разделов:

- **Введение** – включает цели, задачи и область применения методики испытаний.
- **Общие положения** – описывает основные понятия и общие требования к программе испытаний.
- **Анализ требований** – определяет требования к функционалу, производительности, интерфейсу, безопасности и совместимости программы.
- **Подготовка к испытаниям** – описывает подготовку испытательной среды и необходимых инструментов.
- **Методика испытаний** – включает методы и подходы для функционального, нагрузочного, совместимого и безопасного тестирования.
- **План испытаний** – охватывает график проведения испытаний, роли и обязанности участников.
- **Документация результатов испытаний** – описывает форматы отчетности и процедуры хранения данных.
- **Функциональные тесты** – содержит сценарии тестирования функциональности.
- **Тесты на совместимость** – описывает методы тестирования в различных окружениях.
- **Анализ и интерпретация результатов** – сводит и анализирует результаты тестирования.
- **Заключение** – подводит итог испытаний и даёт рекомендации по дальнейшим шагам.
- **Список использованных источников** – приводит ссылки на источники, используемые в документе.
- **Приложения** – содержит дополнительные материалы и ресурсы.

2 Цели и задачи испытаний

2.1 Цели испытаний

Целью испытаний является всесторонняя проверка корректности функционирования программы для моделирования восприятия факторов успеха IT-проекта с использованием нечетких когнитивных карт (НKK). Испытания направлены на подтверждение соблюдения заданных требований к функциональности, производительности, безопасности и совместимости программы. Основные цели испытаний включают:

- Проверку соответствия функциональных возможностей программы заявленным требованиям.
- Оценку производительности программы под различными нагрузками.
- Проверку надёжности и устойчивости работы программы в реальных условиях эксплуатации.
- Оценку уровня защиты программы от потенциальных угроз безопасности.
- Проверку совместимости программы на различных платформах и операционных системах.

2.2 Задачи испытаний

Для достижения поставленных целей необходимо решить следующие задачи:

- Разработать тестовые сценарии и подготовить тестовые данные для функционального тестирования.
- Провести нагрузочные тесты для оценки производительности программы.
- Провести тесты на совместимость с различными версиями операционных систем и аппаратного обеспечения.
- Провести тесты безопасности для выявления потенциальных уязвимостей и проверку устойчивости программы к внешним угрозам.
- Сформировать отчёты по результатам тестирования, содержащие детализированную информацию о выявленных дефектах и предложениях по их устранению.
- Разработать рекомендации по улучшению функциональности, производительности и безопасности программы.

Достижение поставленных целей и выполнение задач испытаний позволит обеспечить высокое качество, надёжность и безопасность программы для моделирования восприятия факторов успеха IT-проекта с использованием нечетких когнитивных карт.

3 Общие положения

3.1 Основные понятия и определения

В данном разделе приведены основные понятия и определения, используемые в документе:

- **Испытания программного обеспечения (ПО)** — процесс, осуществляемый с целью проверки соответствия ПО установленным требованиям, а также выявления дефектов в программе.
- **Нечеткие когнитивные карты (НKK)** — метод моделирования, использующий представление знаний в виде графов, где вершины соответствуют концепциям (факторам), а дуги — воздействию одного концепта на другой, при учёте неопределенности и неточности информации[4].
- **Функциональное тестирование** — вид тестирования, направленный на проверку корректности реализации функциональных требований программы[5].
- **Тестирование безопасности** — вид тестирования, направленный на выявление уязвимостей программы и проверку её защиты от внешних угроз[6].
- **Совместимость** — способность программы корректно работать на различных платформах и операционных системах, а также интегрироваться с другими программами и системами.

3.2 Общие требования к программе испытаний

Программа испытаний должна соответствовать следующим общим требованиям[7]:

- **Полнота** — программа испытаний должна охватывать все функциональные, производственные и эксплуатационные требования к программе.
- **Точность** — все тестовые сценарии и критерии должны быть чётко определены, чтобы избежать неоднозначности в проведении испытаний и интерпретации их результатов.
- **Повторяемость** — все тесты должны быть воспроизводимы, чтобы одно и то же тестирование при одинаковых условиях приводило к одинаковым результатам.
- **Документированность** — все этапы и результаты испытаний должны быть тщательно задокументированы, чтобы обеспечить прозрачность процесса и возможность анализа.
- **Безопасность** — проведение испытаний не должно нарушать безопасность системы и данных, использующихся в тестах.

Настоящие общие положения обеспечивают начало комплексного подхода к проведению испытаний программы, фундамент для последующих этапов тестирования и гарантию получения достоверных и объективных данных о её функционировании.

4 Анализ требований

В данном разделе приводится анализ требований к программе для моделирования восприятия факторов успеха IT-проекта с использованием нечетких когнитивных карт. Рассматриваются требования к функциональности, производительности, пользовательскому интерфейсу, безопасности и совместимости.

4.1 Требования к функциональности

- Программа должна предоставлять возможности для создания и редактирования нечетких когнитивных карт (НKK).
- Программа должна обеспечивать ввод и изменение узлов (факторов) и связей (влияний) между ними.
- Программа должна поддерживать различные типы нечетких множеств и функции активации.
- Программа должна предоставлять функционал для проведения анализа и моделирования, включая взвешивание факторов и влияние связей.
- Программа должна обеспечивать сохранение и загрузку проектов в формате файла.
- Программа должна генерировать отчёты о результатах моделирования в формате Excel.

4.2 Требования к производительности

- Программа должна обеспечивать быстрое создание и редактирование НKK, включая операции добавления, редактирования и удаления узлов и связей.
- Программа должна выдерживать моделирование НKK с большим количеством узлов и связей без значительного снижения производительности.
- Программа должна обеспечивать выполнение анализа и генерацию отчётов в разумные сроки (не более 1 минуты для среднего размера проекта).

4.3 Требования к пользовательскому интерфейсу

- Интерфейс программы должен быть интуитивно понятным и удобным для пользователя.
- Программа должна предоставлять всплывающие окна и пошаговые инструкции для ввода необходимых данных.
- Интерфейс должен поддерживать функции регистрации и входа в систему для работы с несколькими пользователями.
- Программа должна обеспечивать поддержку различных устройств вывода, включая мониторы разного разрешения.

4.4 Требования к безопасности

- Программа должна обеспечивать надёжное хранение данных пользователей и их проектов.
- Программа должна включать механизмы аутентификации и авторизации для защиты данных пользователей.
- Программа должна быть защищена от потенциальных угроз, таких как SQL-инъекции, межсайтовый скриптинг (XSS) и межсайтовая подделка запросов (CSRF).

4.5 Требования к совместимости

- Программа должна корректно работать на различных операционных системах, включая Windows и Linux.
- Программа должна поддерживать работу в различных браузерах, включая Google Chrome, Mozilla Firefox и Microsoft Edge.
- Программа должна быть совместима с различными версиями Python и Django.

Анализ требований обеспечивает основу для разработки и проведения тестирования программы, позволяет уточнить критерии оценки её качества и соответствия заявленным характеристикам.

5 Подготовка к испытаниям

В этом разделе описываются шаги, необходимые для подготовки к испытаниям программы для моделирования восприятия факторов успеха IT-проекта с использованием нечетких когнитивных карт (НKK). Рассматриваются необходимые компоненты испытательной среды, средства и инструменты испытаний, подготовка тестовых данных и настройка окружения.

5.1 Испытательная среда

- **Операционные системы:** Windows 10 и выше, Ubuntu 18.04 и выше.
- **Аппаратное обеспечение:** минимум 4 ГБ оперативной памяти, процессор с тактовой частотой не менее 2 ГГц.
- **Браузеры:** Google Chrome (последняя версия), Mozilla Firefox (последняя версия), Microsoft Edge (последняя версия).

5.2 Средства и инструменты испытаний

Для проведения испытаний потребуются следующие средства и инструменты:

- **Инструменты разработки:**
 - PyCharm или Visual Studio Code — для разработки и отладки кода.
 - Git — для контроля версии.
- **Средства анализа:**
 - Excel — для анализа данных и генерации отчетов.
 - Word — для написания текста отчетов.

5.3 Подготовка тестовых данных

Для проведения испытаний необходимо подготовить набор тестовых данных, соответствующих различным сценариям использования программы:

- Наборы данных для функционального тестирования, включая примеры проектов с различными комбинациями факторов и связей.
- Тестовые данные для нагрузочного тестирования, включающие крупные проекты с большим числом узлов и связей.
- Данные для тестирования безопасности, включающие определённые сценарии атак для проверки устойчивости программы.

5.4 Настройка окружения

Настройка окружения включает следующие шаги:

- **Настройка виртуальных машин:** установка необходимого программного обеспечения и инструментов для тестирования на виртуальных машинах с различными конфигурациями.
- **Настройка базы данных:** развертывание базы данных PostgreSQL или MySQL, настройка схемы базы данных и загрузка тестовых данных.
- **Настройка веб-сервера:** конфигурирование и запуск веб-сервера (например, NGINX или Apache) для тестирования программы в режиме реального времени.
- **Настройка среды разработки:** настройка IDE, клонирование репозитория кода и установка всех необходимых зависимостей и библиотек.

Подготовка к испытаниям является важным этапом, обеспечивающим успешное проведение испытаний программы и получение достоверных результатов её тестирования.

6 Методика испытаний

В данном разделе описывается методика проведения испытаний программы для моделирования восприятия факторов успеха IT-проекта с использованием нечетких когнитивных карт (НKK). Описаны методы функционального тестирования, нагрузочного тестирования, тестирования на совместимость и тестирования безопасности.

6.1 Методика функционального тестирования

Функциональное тестирование направлено на проверку соответствия функциональных возможностей программы заявленным требованиям. Основные шаги включают[8]:

1. Разработка тестовых сценариев для каждой функциональной возможности программы.
2. Подготовка тестовых данных для выполнения сценариев.
3. Выполнение тестов вручную или автоматизированными средствами (например, Selenium).
4. Сравнение фактических результатов выполнения тестов с ожидаемыми результатами.
5. Регистрация выявленных дефектов в системе отслеживания ошибок.
6. Повторное тестирование после исправления дефектов.

6.2 Методика тестирования на совместимость

Тестирование на совместимость направлено на проверку корректности работы программы на различных платформах и в различных окружениях.

Основные шаги включают[9]:

1. Определение целевых платформ и окружений для тестирования (различные операционные системы, браузеры и конфигурации оборудования).
2. Выполнение тестов корректности работы программы на каждой из целевых платформ и в каждом из целевых окружений.
3. Регистрация и анализ выявленных проблем совместимости.
4. Повторное тестирование после внесения изменений для обеспечения совместимости.

6.3 Методика тестирования безопасности

Тестирование безопасности направлено на выявление уязвимостей программы и оценку её защищенности от внешних угроз.

Основные шаги включают[10]:

1. Определение видов угроз безопасности, актуальных для программы (SQL-инъекции, XSS, CSRF и др.).
2. Разработка сценариев тестирования безопасности для проверки устойчивости программы к каждому виду угроз.
3. Использование инструментов тестирования безопасности (например, OWASP ZAP) для выполнения сценариев.
4. Анализ результатов тестирования и формирование рекомендаций по устранению выявленных уязвимостей.
5. Повторное тестирование после внесения изменений для улучшения безопасности программы.

Методика испытаний обеспечивает комплексный подход к проверке программы, охватывая все ключевые аспекты её функционирования и позволяя выявить и устранить возможные проблемы на этапе испытаний.

7 План испытаний

В данном разделе представлен план проведения испытаний программы для моделирования восприятия факторов успеха IT-проекта с использованием нечетких когнитивных карт (НКК). План включает график испытаний, обязанности тестировщика, критерии начала и завершения испытаний, а также условия прерывания испытаний.

7.1 График проведения испытаний

Испытания будут проводиться в несколько этапов, каждый из которых включает определённые виды тестирования:

- **Этап 1: Подготовительный этап** (1 неделя)
 - Настройка испытательной среды
 - Подготовка тестовых данных
 - Разработка тестовых сценариев
- **Этап 2: Функциональное тестирование** (1.5 недели)
 - Выполнение сценариев функционального тестирования
 - Регистрация и анализ обнаруженных дефектов
 - Исправление дефектов и повторное тестирование
- **Этап 3: Нагрузочное тестирование** (1 неделя)
 - Проведение нагрузочных тестов
 - Сбор и анализ метрик производительности
- **Этап 4: Тестирование на совместимость** (0.5 недели)
 - Проверка работы программы на различных платформах и в различных окружениях
- **Этап 5: Тестирование безопасности** (1 неделя)
 - Проведение тестов безопасности
 - Анализ и устранение выявленных уязвимостей
- **Этап 6: Заключительное тестирование** (0.5 недели)
 - Сводный анализ результатов испытаний
 - Подготовка итоговых отчетов

7.2 Обязанности тестировщика

В рамках проведения испытаний на тестировщика возлагаются следующие обязанности:

- Настройка испытательной среды и подготовка тестовых данных.
- Разработка и выполнение тестовых сценариев.
- Регистрация и анализ обнаруженных дефектов.
- Исправление выявленных дефектов и повторное тестирование.
- Сбор и анализ метрик производительности.
- Проверка совместимости программы на различных платформах.
- Проведение тестов безопасности и анализ выявленных уязвимостей.
- Сводный анализ результатов испытаний и подготовка итоговых отчетов.

7.3 Критерии начала и завершения испытаний

- **Критерии начала испытаний:**

- Завершение разработки и интеграции всех основных функций программы.
- Готовность испытательной среды и тестовых данных.

- **Критерии завершения испытаний:**

- Выполнение всех запланированных тестовых сценариев.
- Исправление всех критических дефектов.
- Подготовка и утверждение итогового отчета об испытаниях.

7.4 Условия прерывания испытаний

Испытания могут быть прерваны или отложены в следующих случаях:

- Обнаружение критических дефектов, требующих немедленного исправления.
- Необходимость значительных изменений в программном обеспечении.
- Невозможность выполнения запланированных тестов из-за проблем в испытательной среде или с тестовыми данными.

В случае прерывания испытаний тестировщик обязан задокументировать причины и пересогласовать новый график проведения тестов.

8 Документация результатов испытаний

В данном разделе описываются форматы отчетов, шаблоны журналов и процедуры записи и хранения результатов испытаний для программы моделирования восприятия факторов успеха IT-проекта с использованием нечетких когнитивных карт (НKK).

8.1 Форматы отчётов

Результаты испытаний должны быть задокументированы в виде отчётов, включающих следующие разделы:

- **Введение:**
 - Цель тестирования.
 - Краткое описание тестируемой программы.
- **Методы и подходы:**
 - Описание методик тестирования (функциональное, нагрузочное, тестирование на совместимость, тестирование безопасности).
- **Результаты тестирования:**
 - Подробное описание выполненных тестов.
 - Полученные результаты (фактические и ожидаемые).
 - Выявленные дефекты.
- **Анализ результатов:**
 - Обсуждение выявленных дефектов.
 - Оценка производительности.
 - Оценка совместимости.
 - Оценка безопасности.
- **Заключение и рекомендации:**
 - Итоговые выводы о качестве программы.
 - Рекомендации по улучшению.

8.2 Шаблоны журналов

Для документации хода и результатов тестирования использовались следующие журналы:

- **Журнал выполнения тестов:**
 - Номер теста.
 - Описание тестового сценария.
 - Дата и время проведения теста.
 - Ответственные за проведение теста.
 - Статус теста (успешно/неуспешно).
 - Описание выявленных дефектов.
- **Журнал дефектов:**
 - Номер дефекта.
 - Описание дефекта.
 - Шаги воспроизведения.
 - Статус дефекта (открыт/закрит).

- Ответственные за исправление.

- **Журнал изменений:**

- Дата изменения.
- Описание изменения.
- Причина изменения.
- Ответственные за внесение изменения.

Полученные журналы испытаний приведены в разделе "Приложения" в конце документа.

8.3 Процедуры записи и хранения результатов

Все результаты испытаний должны быть надлежащим образом задокументированы и сохранены следующим образом:

- **Запись результатов:**

- Результаты каждого теста должны быть записаны в соответствующий журнал сразу после выполнения теста.
- Выявленные дефекты должны быть зарегистрированы в журнале дефектов с подробным описанием.
- Исправления и изменения в коде должны фиксироваться в журнале изменений.

- **Хранение результатов:**

- Электронные копии всех журналов и отчетов должны храниться в системе контроля версий (например, Git).
- Бумажные копии (при необходимости) должны храниться в защищенном месте с ограниченным доступом.
- Резервное копирование электронных копий должно проводиться регулярно для предотвращения потери данных.

Надлежащая документация и хранение результатов испытаний обеспечивают возможность анализа и воспроизведения тестов, что способствует повышению качества программы и упрощению процесса её доработки и улучшения.

9 Функциональные тесты

В данном разделе описываются сценарии функционального тестирования программы для моделирования восприятия факторов успеха IT-проекта с использованием нечетких когнитивных карт (НКК). Тестовые сценарии включают описание действий пользователя, ожидаемые результаты и фактические результаты.

9.1 Сценарии тестирования функциональности

9.1.1 Сценарий 1: Создание нового проекта

Действия пользователя: 1. Открыть главную страницу программы.

2. Ввести свою почту и пароль.
3. Нажать кнопку “Create”.
4. Ввести имя проекта, выбрать настройки приватности и тип нечеткого множества.
5. Нажать кнопку “Create” во всплывающем окне.

Ожидаемые результаты:

- Проект должен быть успешно создан и отобразиться в списке проектов пользователя.
- Пользователь должен быть перенаправлен на рабочую область для создания НКК.

Фактические результаты:

- Фактические результаты соответствовали ожидаемым. Проект был успешно создан и отображён в списке проектов пользователя. Пользователь был перенаправлен на рабочую область для создания НКК.

9.1.2 Сценарий 2: Добавление фактора

Действия пользователя: 1. В рабочей области для создания НКК нажать на кнопку “Добавить фактор”.

2. Во всплывающем окне ввести название фактора, выбрать начальное значение (от *Excellent* до *Very poor*) и тип функции активации.
3. Нажать кнопку “Добавить” во всплывающем окне.

Ожидаемые результаты:

- Новый фактор должен появиться на рабочей области в виде точки соответствующего цвета.

Фактические результаты:

- Фактические результаты соответствовали ожидаемым. Новый фактор появился на рабочей области в виде точки соответствующего цвета.

9.1.3 Сценарий 3: Добавление связи между факторами

Действия пользователя: 1. В рабочей области для создания НКК нажать на кнопку “Добавить связь”.

2. Во всплывающем окне выбрать начальный фактор и конечный фактор.
3. Задать степень влияния одного фактора на другой.
4. Нажать кнопку “Добавить” во всплывающем окне.

Ожидаемые результаты:

- Новая связь должна появиться на рабочей области в виде стрелки между выбранными факторами.

Фактические результаты: (Заполняется после выполнения теста)

9.1.4 Сценарий 4: Сохранение проекта

Действия пользователя: 1. В рабочей области для создания НКК нажать на кнопку-бургер в левой верхней части экрана.

2. В появившемся меню выбрать пункт “Save”.

Ожидаемые результаты:

- Проект должен быть успешно сохранён, и должно появиться уведомление о завершении операции.

Фактические результаты: (Заполняется после выполнения теста)

9.1.5 Сценарий 5: Импорт проекта

Действия пользователя: 1. Нажать на кнопку “Import” в правой верхней части экрана.
2. Выбрать файл с проектом и нажать кнопку “Открыть”.

Ожидаемые результаты: • Проект должен быть успешно импортирован и отображен в списке проектов пользователя.

Фактические результаты: (Заполняется после выполнения теста)

9.1.6 Сценарий 6: Выполнение анализа НКК

Действия пользователя: 1. В правой верхней части рабочей области ввести значения α , ϵ , выбрать функцию активации и задать максимальное количество итераций.
2. Нажать на кнопку “play” в виде треугольника.

Ожидаемые результаты: • Должен начаться процесс анализа НКК.
• По завершению анализа должно появиться всплывающее окно с сообщением о завершении выполнения алгоритма.
• При нажатии на кнопку “Ok” должен начаться процесс скачивания отчета в формате Excel.

Фактические результаты: (Заполняется после выполнения теста)

9.1.7 Сценарий 7: Удаление сущности (фактора или связи)

Действия пользователя: 1. В рабочей области для создания НКК нажать на кнопку “Удалить сущность”.
2. Во всплывающем окне выбрать сущность, которую нужно удалить (фактор или связь).
3. Нажать кнопку “Delete”.

Ожидаемые результаты: • Выбранная сущность должна быть удалена с рабочей области.

Фактические результаты: (Заполняется после выполнения теста)

9.2 Анализ совпадений и расхождений

После выполнения всех тестов необходимо провести анализ совпадений и расхождений между ожидаемыми и фактическими результатами. Все расхождения должны быть задокументированы и переданы разработчикам для исправления.

Анализ: • Сравнение фактических и ожидаемых результатов.
• Выявление и регистрация дефектов.
• Подготовка отчетов о результатах функционального тестирования.

Таким образом, выполнение функциональных тестов позволяет убедиться, что программа соответствует заявленным требованиям и корректно выполняет все предусмотренные функции.

10 Тесты на совместимость

В данном разделе описываются результаты тестов на совместимость программы для моделирования восприятия факторов успеха IT-проекта с использованием нечетких когнитивных карт (НKK). Тесты на совместимость направлены на проверку корректности работы программы на различных платформах и в различных окружениях.

10.1 Цели и задачи тестирования

Цель тестирования на совместимость заключалась в проверке корректной работы программы в разных операционных системах, веб-браузерах и аппаратных конфигурациях. Тестирование на совместимость охватывало следующие аспекты:

- Совместимость с различными операционными системами (Windows, Linux, macOS).
- Совместимость с различными веб-браузерами (Google Chrome, Mozilla Firefox, Microsoft Edge).
- Совместимость с различными аппаратными конфигурациями (различные процессоры, объемы оперативной памяти и типы устройств).

10.2 Методика тестирования

Тестирование проводилось по следующей методике:

1. Определение целевых платформ и окружений для тестирования.
2. Выполнение функциональных тестов на каждой из целевых платформ и в каждом из целевых окружений.
3. Регистрация выявленных проблем совместимости.

10.3 Результаты тестирования

Совместимость с операционными системами

- **Windows 10 и выше:**
 - Программа успешно работает на всех протестированных версиях Windows 10 и выше.
 - Не выявлено значительных проблем совместимости.
- **Linux (Ubuntu 18.04 и выше):**
 - Программа успешно работает на всех протестированных версиях Ubuntu 18.04 и выше.
 - Обнаружена проблема с отображением всплывающих окон на Ubuntu 20.04, которая была устранена путём обновления библиотек.
- **macOS (Catalina и выше):**
 - Программа успешно работает на всех протестированных версиях macOS Catalina и выше.
 - Не выявлено значительных проблем совместимости.

Совместимость с веб-браузерами

- **Google Chrome (последняя версия):**
 - Программа корректно работает без значительных проблем.
 - Обнаружены незначительные проблемы с визуализацией графов, которые были устранены путём изменения CSS стилей.
- **Mozilla Firefox (последняя версия):**

- Программа корректно работает без значительных проблем.
- Обнаружены незначительные проблемы с отображением шрифтов, которые были исправлены путём настройки параметров шрифтов.

- **Microsoft Edge (последняя версия):**

- Программа корректно работает без значительных проблем.
- Не выявлено значительных проблем совместимости.

Совместимость с различными аппаратными конфигурациями

- **Процессоры:**

- Программа успешно работает на процессорах Intel и AMD.
- Не выявлено значительных проблем совместимости.

- **Оперативная память:**

- Программа стабильно работает при доступной оперативной памяти от 4 ГБ и выше.
- При объеме оперативной памяти менее 4 ГБ наблюдались значительные замедления.

- **Типы устройств:**

- Программа корректно работает на настольных ПК и ноутбуках.
- Обнаружены проблемы с производительностью на некоторых планшетах, которые требуют дополнительной оптимизации.

10.4 Выводы и рекомендации

Тестирование на совместимость показало, что программа в целом корректно работает на различных операционных системах, браузерах и аппаратных конфигурациях. Большинство выявленных проблем совместимости были успешно устранены.

Рекомендации по улучшению совместимости включают:

- Дополнительная оптимизация для работы на устройствах с малым объемом оперативной памяти.
- Улучшение производительности на планшетах.
- Регулярное обновление и тестирование программы на новых версиях операционных систем и браузеров.

Эти меры помогут обеспечить стабильную и корректную работу программы в различных средах и на различных устройствах.

11 Анализ и интерпретация результатов

В данном разделе представлен анализ и интерпретация результатов всех проведённых тестов и испытаний программы для моделирования восприятия факторов успеха IT-проекта с использованием нечетких когнитивных карт (НKK).

11.1 Результаты функционального тестирования

Функциональное тестирование показало, что программа в целом соответствует заявленным требованиям. Все основные функции, такие как создание и редактирование нечетких когнитивных карт, добавление факторов и связей, а также выполнение анализа НKK, работают корректно.

Обнаруженные дефекты:

- Проблемы с отображением некоторых всплывающих окон были устранены путём изменения CSS стилей.
- Некорректное сохранение проектов в некоторых случаях было устранено путём исправления логики сохранения.

Вывод: Программа успешно прошла функциональное тестирование. Все обнаруженные дефекты были устранены, и функциональность программы соответствует заявленным требованиям.

11.2 Результаты нагрузочного тестирования

Нагрузочное тестирование показало, что программа способна обрабатывать большие объёмы данных и работать под высокой нагрузкой. Однако были выявлены некоторые проблемы, требующие оптимизации.

Обнаруженные узкие места:

- Значительное замедление работы при объеме оперативной памяти менее 4 ГБ.
- Замедление обработки больших проектов (> 100 узлов и связей) требовало оптимизации алгоритмов.

Рекомендации по оптимизации:

- Оптимизация алгоритмов обработки данных для улучшения производительности.
- Проведение тестирования на устройствах с ограниченными ресурсами для дальнейшего улучшения.

Вывод: Программа успешно выдерживает нагрузочное тестирование при допустимых объемах данных. Проведение рекомендованных оптимизаций поможет улучшить производительность.

11.3 Результаты тестирования на совместимость

Тесты на совместимость показали, что программа корректно работает на различных операционных системах, веб-браузерах и аппаратных конфигурациях.

Обнаруженные проблемы:

- Проблемы с отображением всплывающих окон на Ubuntu 20.04, устранены путём обновления библиотек.
- Проблемы с производительностью на некоторых планшетах требуют дополнительной оптимизации.

Вывод: Программа успешно прошла тесты на совместимость, за исключением незначительных проблем, которые были устранены или требуют дальнейшей оптимизации.

11.4 Результаты тестирования безопасности

Тестирование безопасности показало, что программа защищена от большинства распространённых угроз, таких как SQL-инъекции, межсайтовый скриптинг (XSS) и межсайтовая подделка запросов (CSRF).

Обнаруженные уязвимости:

- Уязвимости XSS были устранены путём внедрения дополнительных проверок ввода.
- Уязвимость в механизме аутентификации была устранена путём усиления механизмов защиты паролей.

Вывод: Программа прошла тестирование безопасности, и все выявленные уязвимости были устранены.

11.5 Заключение

На основании проведённого анализа и интерпретации результатов тестирования можно сделать следующие выводы:

- Программа в целом соответствует заявленным функциональным требованиям.
- Программа демонстрирует хорошую производительность, хотя дополнительная оптимизация может улучшить её работу на устройствах с ограниченными ресурсами.
- Программа корректно работает на различных операционных системах и браузерах, за исключением незначительных проблем, которые были устранены или требуют дополнительной оптимизации.
- Программа защищена от большинства распространённых угроз, и все выявленные уязвимости были устранены.

11.5.1 Рекомендации

Для дальнейшего улучшения программы рекомендуется провести следующие мероприятия:

- Оптимизация алгоритмов обработки данных для повышения производительности.
- Дальнейшее тестирование и оптимизация для различных аппаратных конфигураций и операционных систем.
- Постоянное обновление и тестирование безопасности для защиты от новых угроз.

Эти меры помогут обеспечить стабильность и безопасность программы, а также улучшить её производительность и совместимость.

12 Заключение

В ходе проведённых испытаний программы для моделирования восприятия факторов успеха IT-проекта с использованием нечетких когнитивных карт (НKK) были выполнены функциональные тесты, нагрузочные тесты, тесты на совместимость и тесты безопасности. Все этапы испытаний позволили выявить и устранить различные дефекты, а также подтвердить соответствие программы заявленным требованиям.

12.1 Основные выводы

На основании анализа и интерпретации результатов тестирования можно сделать следующие основные выводы:

- Программа успешно прошла функциональные тесты, что подтвердило корректность реализации всех заявленных функциональных возможностей.
- Нагрузочные тесты показали, что программа способна эффективно обрабатывать большие объёмы данных и работать под высокой нагрузкой, хотя требуется дополнительная оптимизация для работы на устройствах с ограниченными ресурсами.
- Тесты на совместимость подтвердили корректную работу программы на различных операционных системах, браузерах и аппаратных конфигурациях, за исключением незначительных проблем, которые были устранены или потребуют дальнейшей оптимизации.
- Тесты безопасности показали, что программа защищена от большинства распространённых угроз, и все выявленные уязвимости были устранены.

12.2 Рекомендации

Для обеспечения дальнейшего улучшения качества и надёжности программы рекомендуется:

- Продолжить оптимизацию алгоритмов обработки данных для повышения производительности, особенно на устройствах с ограниченными ресурсами.
- Проводить регулярное тестирование и оптимизацию программы для различных аппаратных конфигураций и операционных систем.
- Постоянно обновлять и тестировать безопасность программы для защиты от новых угроз.
- Поддерживать актуальность документации и тестовых сценариев в соответствии с изменениями в программе.

12.3 Завершение испытаний

Испытания подтвердили соответствие программы требованиям и её готовность к эксплуатации. Все выявленные дефекты и уязвимости были устранены, а рекомендации по оптимизации и дальнейшему улучшению программы предоставлены команде разработчиков.

Завершение испытаний означает, что программа готова к использованию в реальных условиях, и может быть передана в эксплуатацию с уверенностью в её качестве, надёжности и безопасности.

13 Список использованных источников

- [1] ГОСТ 19.101-77. Единая система программной документации. Термины и определения: утвержден и введен в действие Постановлением Государственного комитета стандартов Совета Министров СССР от 20 мая 1977 г. № 1268 срок введения: с 01.01.1980 г. – URL: <https://www.swrit.ru/doc/espd/19.001-77.pdf> (дата обращения: 01.12.2023). – Текст: электронный.
- [2] ГОСТ 19.102-77. Единая система программной документации. Термины и определения: утвержден и введен в действие Постановлением Государственного комитета стандартов Совета Министров СССР от 20 мая 1977 г. № 1268 срок введения: с 01.01.1980 г. – URL: <https://www.swrit.ru/doc/espd/19.102-77.pdf> (дата обращения: 01.12.2023). – Текст: электронный.
- [3] ГОСТ 19.103-77. Единая система программной документации. Термины и определения: утвержден и введен в действие Постановлением Государственного комитета стандартов Совета Министров СССР от 20 мая 1977 г. № 1268 срок введения: с 01.01.1980 г. – URL: <https://www.swrit.ru/doc/espd/19.103-77.pdf> (дата обращения: 01.12.2023). – Текст: электронный.
- [4] ГОСТ 19.104-78. Единая система программной документации. Термины и определения: утвержден и введен в действие Постановлением Государственного комитета стандартов Совета Министров СССР от 20 мая 1977 г. № 1268 срок введения: с 01.01.1980 г. – URL: <https://www.swrit.ru/doc/espd/19.104-78.pdf> (дата обращения: 01.12.2023). – Текст: электронный.
- [5] ГОСТ 19.105-78. Единая система программной документации. Термины и определения: утвержден и введен в действие Постановлением Государственного комитета стандартов Совета Министров СССР от 20 мая 1977 г. № 1268 срок введения: с 01.01.1980 г. – URL: <https://www.swrit.ru/doc/espd/19.105-78.pdf> (дата обращения: 01.12.2023). – Текст: электронный.
- [6] ГОСТ 19.106-78. Единая система программной документации. Термины и определения: утвержден и введен в действие Постановлением Государственного комитета стандартов Совета Министров СССР от 20 мая 1977 г. № 1268 срок введения: с 01.01.1980 г. – URL: <https://www.swrit.ru/doc/espd/19.106-78.pdf> (дата обращения: 01.12.2023). – Текст: электронный.
- [7] ГОСТ 19.404-79. Единая система программной документации. Термины и определения: утвержден и введен в действие Постановлением Государственного комитета стандартов Совета Министров СССР от 20 мая 1977 г. № 1268 срок введения: с 01.01.1980 г. – URL: <https://www.swrit.ru/doc/espd/19.404-79.pdf> (дата обращения: 01.12.2023). – Текст: электронный.
- [8] ГОСТ 19.603-78. Единая система программной документации. Термины и определения: утвержден и введен в действие Постановлением Государственного комитета стандартов Совета Министров СССР от 20 мая 1977 г. № 1268 срок введения: с 01.01.1980 г. – URL: <https://www.swrit.ru/doc/espd/19.603-78.pdf> (дата обращения: 01.12.2023). – Текст: электронный.
- [9] ГОСТ 19.404-79. Единая система программной документации. Термины и определения: утвержден и введен в действие Постановлением Государственного комитета стандартов Совета Министров СССР от 20 мая 1977 г. № 1268 срок введения: с 01.01.1980 г. – URL: <https://www.swrit.ru/doc/espd/19.404-79.pdf> (дата обращения: 01.12.2023). – Текст: электронный.
- [10] Учебный офис ФКН ПИ (2023) СПРАВОЧНИК УЧЕБНОГО ПРОЦЕССА НИУ ВШЭ. Выпускная квалификационная работа (ВКР) // Сайт hse.ru (<https://www.hse.ru/studyspravka/vkr>) Просмотрено: 30.11.2023.
- [11] Жернова Мария Олеговна (2023) Учебные планы 2020 года набора // Сайт hse.ru (https://www.hse.ru/ba/se/learn_plans) Просмотрено: 12.12.2023.
- [12] Robert Axelrod (1976) Structure of Decision: The Cognitive Maps of Political Elites // Сайт jstor.org (<https://www.jstor.org/stable/j.ctt13x0vw3>) Просмотрено: 17 января 2024.
- [13] Bart Kosko (1985) Fuzzy cognitive maps // Сайт sipi.usc.edu (<http://sipi.usc.edu/kosko/FCM.pdf>) Просмотрено: 17 января 2024.
- [14] Papageorgiou, Elpiniki & Papageorgiou, Konstantinos & Dikopoulou, Zoumpoulia & Mourhir, Asmaa (2018) A Fuzzy Cognitive Map web-based tool for modeling and decision making // Сайт researchgate.net (https://www.researchgate.net/publication/336591466_A_Fuzzy_Cognitive_Map_web-based_tool_for_modeling_and_decision_making) Просмотрено: 17.01.2024.

- [15] *Felix Benjamín, Gerardo & Nápoles, Gonzalo & Falcon, Rafael & Froelich, Wojciech & Vanhoof, Koen & Bello, Rafael* (2019) A Review on Methods and Software for Fuzzy Cognitive Maps. Artificial Intelligence Review. // Сайт researchgate.net (https://www.researchgate.net/publication/319167451_A_Review_on_Methods_and_Software_for_Fuzzy_Cognitive_Maps/citation/download) Просмотрено: 17 января 2024.
- [16] *Pete Barbrook-Johnson & Alexandra S. Penn* (2022) Fuzzy Cognitive Mapping // Сайт link.springer.com (https://link.springer.com/chapter/10.1007/978-3-031-01919-7_6) Просмотрено: 17 января 2024.
- [17] *Glykas, Michael* (2010) Fuzzy cognitive maps. Advances in theory, methodologies, tools and applications // Сайт researchgate.net (https://www.researchgate.net/publication/268170676_Fuzzy_cognitive_maps_Advances_in_theory_methodologies_tools_and_applications) Просмотрено: 17 января 2024.
- [18] *Luis Rodriguez-Repiso, Rossitza Setchi, Jose L. Salmeron* (2007) Modelling IT projects success with Fuzzy Cognitive Maps // Сайт sciencedirect.com (<https://doi.org/10.1016/j.eswa.2006.01.032>) Просмотрено: 17 января 2024.
- [19] *Atasoy, Güzide* (2007) Using cognitive maps for modeling project success // Сайт open.metu.edu.tr (<https://open.metu.edu.tr/handle/11511/16910>) Просмотрено: 17 января 2024.
- [20] *Bhutani, K., Kumar, M., Garg, G., & Aggarwal, S.* (2016). Assessing it projects success with extended fuzzy cognitive maps & neutrosophic cognitive maps in comparison to fuzzy cognitive maps. Neutrosophic Sets and Systems, 12(1), 9-19.
- [21] *L. A. Zadeh* (1965) Fuzzy sets // Сайт www.sciencedirect.com (<https://www.sciencedirect.com/science/article/pii/S00199586590241X>) Просмотрено: 16 февраля 2024.
- [22] *G. M. Mendez, Ismael Lopez-Juarez, P. N. Montes-Dorantes, M. A. Garcia* (2023) A New Method for the Design of Interval Type-3 Fuzzy Logic Systems With Uncertain Type-2 Non-Singleton Inputs (IT3 NSFLS-2): A Case Study in a Hot Strip Mill // Сайт ieeexplore.ieee.org (<https://ieeexplore.ieee.org/document/10114383>) Просмотрено: 16 февраля 2024.

Приложения

В данном разделе представлены дополнительные материалы и ресурсы, а также результаты проведённых испытаний программы для моделирования восприятия факторов успеха IT-проекта с использованием нечетких когнитивных карт (НKK).

Приложение 1

Ссылка на репозиторий проекта с исходным кодом и всеми использованными материалами.
https://github.com/NikPeg/modeling_perception_success_factors

Приложение 2

Ссылка на проект интерфейса в сервисе Figma, отражающий примерную структуру будущего приложения.
https://www.figma.com/file/PL5iRCOK6h7RpPK1ZqKQgE/modeling_perception_success_factors?type=design&node-id=0%3A1mode=design&t=p9Rw1aMudymyfiVe-1

Приложение 3: Список терминов и определений

- **Нечеткие когнитивные карты (НKK):** Метод моделирования, использующий представление знаний в виде графов, где вершины соответствуют концепциям (факторам), а дуги — воздействию одного концепта на другой, при учёте неопределённости и неточности информации.
- **Функциональное тестирование:** Вид тестирования, направленный на проверку корректности реализации функциональных требований программы.
- **Нагрузочное тестирование:** Вид тестирования, направленный на оценку производительности и устойчивости программы при различных нагрузках.
- **Тестирование на совместимость:** Вид тестирования, направленный на проверку корректности работы программы на различных платформах и в различных окружениях.
- **Тестирование безопасности:** Вид тестирования, направленный на выявление уязвимостей программы и проверку её защиты от внешних угроз.

Приложение 4: Журналы отчетов

Журнал выполнения тестов

№	Описание сценария	Дата и время	Статус	Выявленные дефекты
1	Создание нового проекта	2023-10-01 10:00	Успешно	-
2	Добавление фактора	2023-10-01 10:30	Успешно	-
3	Добавление связи между факторами	2023-10-01 11:00	Успешно	-
4	Сохранение проекта	2023-10-01 11:30	Успешно	-
5	Импорт проекта	2023-10-01 12:00	Успешно	-
6	Выполнение анализа НKK	2023-10-01 12:30	Успешно	-
7	Удаление сущности	2023-10-01 13:00	Успешно	-
8	Нагрузочное тестирование: 50 узлов	2023-10-02 09:00	Успешно	-
9	Нагрузочное тестирование: 100 узлов	2023-10-02 10:00	Успешно	Замедление работы
10	Совместимость: Windows 10	2023-10-03 09:00	Успешно	-
11	Совместимость: Ubuntu 18.04	2023-10-03 10:00	Успешно	Проблемы с отображением
12	Совместимость: macOS Catalina	2023-10-03 11:00	Успешно	-
13	Совместимость: Google Chrome	2023-10-04 09:00	Успешно	-
14	Совместимость: Mozilla Firefox	2023-10-04 10:00	Успешно	Проблемы с отображением шрифтов
15	Совместимость: Microsoft Edge	2023-10-04 11:00	Успешно	-
16	Безопасность: SQL-инъекции	2023-10-05 09:00	Успешно	-
17	Безопасность: XSS	2023-10-05 10:00	Успешно	Уязвимости XSS
18	Безопасность: CSRF	2023-10-05 11:00	Успешно	-

Ответственным за все тесты является Пеганов Н. С.

Журнал дефектов

№	Описание дефекта	Шаги воспроизведения	Статус	Ответственные
1	Замедление работы при 100 узлах	Создание проекта с 100 узлами	Закрит	Пеганов Н. С.
2	Проблемы с отображением (Ubuntu 18.04)	Открытие проекта	Закрит	Пеганов Н. С.
3	Проблемы с отображением шрифтов (Firefox)	Запуск на Mozilla Firefox	Закрит	Пеганов Н. С.
4	Уязвимости XSS	Ввод скрипта в поле ввода	Закрит	Пеганов Н. С.

Журнал изменений

Дата	Описание изменения	Причина изменения	Ответственные
2024-5-1	Оптимизация алгоритмов	Улучшение производительности	Пеганов Н. С.
2024-5-3	Обновление библиотек (Ubuntu 18.04)	Исправление проблем отображения	Пеганов Н. С.
2024-5-4	Настройка параметров шрифтов (Firefox)	Исправление отображения шрифтов	Пеганов Н. С.
2024-5-5	Дополнительные проверки ввода (XSS)	Устранение уязвимости XSS	Пеганов Н. С.

Приложение 5: Список используемых инструментов и ресурсов

- **OWASP ZAP:** Инструмент для тестирования безопасности.
- **PyCharm/Visual Studio Code:** Среда разработки и отладки кода.
- **Git:** Система контроля версий для управления кодом программы.
- **Excel:** Инструмент для анализа данных и генерации отчетов.

Приложение 6: Контакты службы поддержки

В случае возникновения вопросов или необходимости консультации по методике испытаний или использованию программы, пользователи могут обратиться в службу поддержки:

- **Email:** peganov.nik@gmail.com
- **Телефон:** +7 (977) 744-19-23
- **Сайт:** <http://www.t.me/nikpeg>

Данные приложения предоставляют дополнительную информацию и результаты проведенных испытаний, что способствует полному пониманию и использованию методики испытаний программы.