

Техническое задание на автоматизированную систему BoardBox

Версия документа: 1.0

Дата составления: 11.03.2025

Разработчики: Команда 1.4

Подпись заказчика: _____

Оглавление

Термины и определения	5
1. Общие сведения	7
1.1 Полное наименование и условное обозначение	7
1.2 Организации	7
1.3 Основания для создания АС	7
1.4 Плановые сроки разработки	7
1.5 Финансирование	7
2. Цели и назначение создания втоматизированной системы	8
2.1 Цели создания АС	8
2.2 Назначение АС	8
3. Характеристика объекта автоматизации	9
3.1 Основные сведения	9
3.2 Условия эксплуатации	9
4. Требования к автоматизированной системе	10
4.1 Требования к структуре АС	10
4.1.1 Подсистемы АС	10
4.1.2 Информационное взаимодействие компонентов	10
4.1.3 Взаимосвязь с внешними системами	10
4.1.4 Режимы функционирования	10
4.1.5 Диагностика системы	11
4.1.6 Перспективы развития	11
4.2 Функциональные требования	11
4.3 Нефункциональные требования	11
4.3.1 Производительность	11
4.3.2 Надежность и доступность	11
4.3.3 Безопасность	12
4.3.4 Масштабируемость	12
4.3.5 Удобство использования (UX/UI)	12
4.3.6 Соответствие стандартам и лицензированию	12
4.4 Требования к видам обеспечения АС	12
4.4.1 Программное обеспечение	12
4.4.2 Информационное обеспечение	13
4.4.3 Техническое обеспечение	13

4.4.4 Лингвистическое обеспечение	13
4.4.5 Организационное обеспечение.....	13
5. Состав и содержание работ по созданию автоматизированной системы	14
5.1 Исследование рынка и формирование требований (11 марта 2025 г.)	14
5.2 Разработка архитектуры системы (25 марта 2025 г.).....	14
5.3 Разработка MVP (1 апреля 2025 г.)	15
5.4 Тестирование и отладка (май 2025 г.)	16
5.5 Разработка полной версии (май 2025 г.)	16
5.6 Внедрение и публикация (декабрь 2025 г.)	17
6. Порядок разработки автоматизированной системы	18
6.1 Порядок организации разработки АС	18
6.2 Перечень документов и исходных данных для разработки АС	18
6.3 Перечень документов, предъявляемых по окончании соответствующих этапов работ	18
6.4 Порядок проведения экспертизы технической документации	19
6.5 Перечень макетов и порядок их разработки, испытаний и документации	19
6.6 Порядок разработки, согласования и утверждения плана совместных работ по разработке АС.....	19
6.7 Порядок разработки, согласования и утверждения программы работ по стандартизации	20
6.8 Требования к гарантийным обязательствам разработчика	20
6.9 Порядок проведения технико-экономической оценки разработки АС.....	21
7. Порядок контроля и приемки автоматизированной системы.....	22
7.1 Виды, состав и методы испытаний АС и ее составных частей	22
7.2 Общие требования к приемке работ, порядок согласования и утверждения приемочной документации.....	22
7.3 Статус приемочной комиссии.....	23
7.4 Доработка системы в случае обнаружения критических ошибок	23
7.5 Подготовка приемочной документации	24
7.6 Критерии успешной приемки	24
8. Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу автоматизированной системы в действие	25
8.1 Создание условий функционирования объекта автоматизации	25
9. Требования к документированию	26
9.1 Перечень подлежащих разработке документов	26
9.2 Вид представления и количество документов	26

10. Источники разработки	27
10.1 Технико-экономическое обоснование.....	27
10.2 Отчёты о законченных научно-исследовательских работах	27
10.3 Информационные материалы на отечественные и зарубежные системы-аналоги ..	27
10.4 Другие информационные материалы	28

Термины и определения

API (Application Programming Interface) – интерфейс программирования приложений. Это набор правил и протоколов, которые позволяют различным программным приложениям взаимодействовать друг с другом.

AWS (Amazon Web Services) – облачная платформа, предоставляемая компанией Amazon, которая включает в себя широкий спектр облачных услуг, таких как вычислительные мощности, хранение данных и базы данных.

Agile – методология разработки программного обеспечения, основанная на гибкости, быстрой итерации и тесном сотрудничестве между командами разработки и заказчиками.

Автоматизированная система – система, в которой процессы управления и обработки информации выполняются автоматически с минимальным участием человека.

ИИ (Искусственный интеллект) – область компьютерных наук, изучающая создание систем, способных выполнять задачи, которые обычно требуют человеческого интеллекта, такие как восприятие, принятие решений и обучение.

Kanban – метод управления проектами, который использует визуальные доски (канбан-доски) для отслеживания состояния задач и их прогресса, обеспечивая эффективное распределение работы в команде.

MVP (Minimum Viable Product) – минимально жизнеспособный продукт. Это версия продукта с минимально необходимым набором функций, чтобы решить основную проблему пользователей и получить обратную связь для дальнейшего улучшения.

OAuth – открытый стандарт авторизации, который позволяет приложениям и веб-сайтам безопасно получать доступ к данным пользователя, не требуя от него предоставления пароля.

OpenAI – исследовательская организация в области искусственного интеллекта, которая занимается разработкой и применением технологий ИИ с целью создания полезных и безопасных решений для человечества.

Scrum – методология управления проектами в рамках Agile, которая основывается на работе в коротких итерациях (спринтах), нацеленных на создание инкрементальных улучшений продукта.

UI/UX – UI (User Interface) — пользовательский интерфейс, который включает в себя все элементы, с которыми взаимодействует пользователь в приложении. **UX (User Experience)** — опыт пользователя, общий подход к проектированию, направленный на обеспечение удобства и удовлетворенности пользователя при взаимодействии с продуктом.

GCP (Google Cloud Platform) – облачная платформа, предоставляемая компанией Google, которая предлагает широкий спектр облачных сервисов, таких как вычислительные

мощности, базы данных и хранение данных, а также инструменты для разработки и анализа данных.

1. Общие сведения

1.1 Полное наименование и условное обозначение

Полное наименование: Автоматизированная система "BoardBox"

Условное обозначение: АС "BoardBox"

1.2 Организации

Заказчик: Преподаватель Тарасов Вячеслав Сергеевич.

Разработчик: Команда BoardBox (Аверьянова Анастасия, Дзюба Дмитрий, Круглов Георгий, Частухин Даниил).

1.3 Основания для создания АС

АС "BoardBox" создается в соответствии с:

- анализом потребностей пользователей настольных игр;
- современными трендами в области автоматизации и рекомендательных систем;
- стандартами UI/UX для удобного взаимодействия с системой.

1.4 Плановые сроки разработки

- Начало разработки: март 2025 г.
- Окончание разработки MVP: май 2025 г.
- Окончание полного цикла разработки: декабрь 2025 г.

1.5 Финансирование

Проект реализуется на энтузиазме команды с возможностью дальнейшего привлечения инвестиций.

2. Цели и назначение создания втоматизированной системы

2.1 Цели создания АС

- Автоматизация процесса подбора настольных игр в зависимости от предпочтений пользователей.
- Упрощение освоения правил игр за счёт генерации кратких объяснений через ИИ.
- Обеспечение удобного хранения и управления личной коллекцией игр.
- Создание социальной платформы для взаимодействия между игроками.
- Минимизация времени на поиск подходящей игры для различных ситуаций (вечеринка, семейный вечер и т. д.).

Критерии оценки достижения целей:

- Внедрение ИИ-рекомендаций и фильтрации по игровым параметрам.
- Интеграция системы объяснения правил.
- Функциональность личных коллекций и взаимодействия между пользователями.
- Количество активных пользователей и удовлетворенность их потребностей.

2.2 Назначение АС

АС "BoardBox" предназначена для автоматизации процесса подбора, хранения и освоения настольных игр. Она будет использоваться игроками разного уровня, коллекционерами и новыми пользователями для удобного управления своей библиотекой игр.

3. Характеристика объекта автоматизации

3.1 Основные сведения

Объектом автоматизации является процесс подбора, хранения и изучения настольных игр. Включает в себя следующие компоненты:

- База данных настольных игр с характеристиками и описаниями.
- Система фильтрации и рекомендаций по играм.
- Механизм генерации кратких правил через ИИ.
- Личный профиль пользователя с возможностью управления коллекцией игр.

3.2 Условия эксплуатации

АС "BoardBox" разрабатывается как веб- и мобильное приложение. Ожидаемые условия эксплуатации:

- Совместимость с Android.
- Нагрузка – до 1 000 пользователей на старте.
- Интеграция с внешними сервисами (OpenAI API, OAuth-авторизация).
- Возможность дальнейшего масштабирования с ростом аудитории.

4. Требования к автоматизированной системе

4.1 Требования к структуре АС

В данном разделе определяются основные структурные элементы автоматизированной системы "BoardBox" и их взаимосвязи.

4.1.1 Подсистемы АС

АС "BoardBox" включает в себя следующие подсистемы:

- **Подсистема управления пользователями** – регистрация, аутентификация, управление профилем.
- **Подсистема рекомендаций** – подбор настольных игр с использованием алгоритмов машинного обучения.
- **Подсистема хранения коллекций** – возможность добавления, удаления и организации настольных игр в личную библиотеку.
- **Подсистема обучения** – генерация кратких правил игр с использованием ИИ.

Дополнительно возможно добавление новых подсистем в рамках модернизации системы.

4.1.2 Информационное взаимодействие компонентов

- Обмен данными между подсистемами осуществляется через API взаимодействие.
- Используются стандартизированные форматы данных (JSON).
- Поддерживается интеграция с внешними сервисами через REST API.
- Взаимодействие пользователей с системой осуществляется через мобильное приложение.

4.1.3 Взаимосвязь с внешними системами

- Интеграция с платежными системами для возможных внутриигровых покупок.
- Авторизация пользователей через сторонние сервисы (Google).
- Взаимодействие с внешними базами данных настольных игр.
- Возможность экспорта данных о коллекциях игр в сторонние сервисы.

4.1.4 Режимы функционирования

- **Обычный режим** – стандартная работа системы с доступом ко всем функциям.
- **Офлайн-режим** – ограниченный доступ к сохраненным данным без подключения к сети.
- **Режим администрирования** – доступен только администраторам системы, включает управление пользователями, модерацию контента и аналитические инструменты.

4.1.5 Диагностика системы

- Встроенная система логирования событий для мониторинга работы.
- Автоматическое оповещение администраторов о сбоях.
- Функционал самодиагностики и сбора метрик производительности.

4.1.6 Перспективы развития

- Ведение статистики пользователем и ее учет в рекомендациях.
- Расширение функционала социальной подсистемы (рейтинги, группы).
- Интеграция с голосовыми помощниками для быстрого подбора игр.
- Поддержка новых форматов контента (видеообзоры, интерактивные руководства).

4.2 Функциональные требования

АС "BoardBox" должна обеспечивать выполнение следующих функций:

- **Управление коллекцией** – добавление, удаление и сортировка игр в профиле пользователя.
- **Подбор игр** – анализ предпочтений пользователя и предложений оптимальных вариантов.
- **Генерация кратких правил** – автоматизированное создание кратких описаний правил для игр.
- **Авторизация и управление пользователями** – регистрация, вход, восстановление пароля, управление профилем.
- **Режим администрирования** – обновление базы данных настольных игр, модерация контента, управление пользователями.

4.3 Нефункциональные требования

4.3.1 Производительность

- Время отклика системы: не более 1 секунды для 90% запросов.
- Достоверность выдаваемых рекомендаций: не менее 80% совпадений с предпочтениями пользователей.
- Отказы в работе функций: не более 1% запросов в сутки.

4.3.2 Надежность и доступность

- Доступность сервиса: не менее 99,2% времени работы.
- Поддержка отказоустойчивости и резервирования данных.
- Возможность восстановления данных в случае сбоя.

4.3.3 Безопасность

- Аутентификация пользователей через OAuth (Google).
- Защита персональных данных пользователей.
- Механизмы предотвращения несанкционированного доступа.
- Политики доступа: разграничение прав пользователей (администраторы, модераторы, обычные пользователи).

4.3.4 Масштабируемость

- Возможность увеличения нагрузки за счет балансировки трафика.
- Поддержка горизонтального масштабирования серверов базы данных и API.

4.3.5 Удобство использования (UX/UI)

- Интерфейс должен быть интуитивно понятным и удобным для пользователей без специальных навыков.
- Поддержка русского языка интерфейса.
- Дизайн соответствует современным принципам UI/UX.

4.3.6 Соответствие стандартам и лицензированию

- Использование лицензированных решений и технологий.
- Применение современных стандартов веб-разработки и баз данных.
- Соответствие требованиям патентной чистоты.

4.4 Требования к видам обеспечения АС

4.4.1 Программное обеспечение

Программное обеспечение включает:

- **Серверную часть:** разработка с использованием современных технологий (Golang).
- **Клиентскую часть:** мобильные приложения (React Native).
- **Базу данных:** PostgreSQL, MongoDB
- **Взаимодействие с внешними сервисами:** OAuth-авторизация (Google), OpenAI API, внешние базы игр.
- **Систему кэширования** для быстрого доступа к популярным данным (Redis, Memcached).
- **Модуль аналитики:** сбор и обработка пользовательских данных.

4.4.2 Информационное обеспечение

Информационное обеспечение включает:

- Базу данных настольных игр с характеристиками, правилами и пользовательскими оценками.
- Каталоги и классификаторы игр по жанрам, сложности, количеству игроков и среднему времени.
- Форматы хранения данных (JSON, SQL) и стандартизированные API для взаимодействия с внешними сервисами.
- Механизмы обновления, резервного копирования и восстановления данных.
- Инструменты мониторинга данных для выявления неточностей и актуализации информации.

4.4.3 Техническое обеспечение

Техническое обеспечение включает:

- Вычислительные серверы и облачные сервисы (AWS, GCP, Azure) для хостинга и обработки данных.
- Серверы баз данных с отказоустойчивостью и резервированием.
- Системы балансировки нагрузки для обеспечения высокой доступности.
- Серверы хранения данных для мультимедийного контента (изображений, видеообзоров).

4.4.4 Лингвистическое обеспечение

Лингвистическое обеспечение включает:

- Поддержку русского языка интерфейса.
- Развитие специализированного словаря терминов настольных игр.

4.4.5 Организационное обеспечение

- Регламенты работы АС, инструкции для пользователей и администраторов.
- Процедуры обновления контента и базы данных игр.
- Политики доступа и права пользователей (администраторы, модераторы, обычные пользователи)

5. Состав и содержание работ по созданию автоматизированной системы

5.1 Исследование рынка и формирование требований (11 марта 2025 г.)

Цель:

Провести анализ рынка настольных игр и существующих аналогов, определить потребности целевой аудитории и сформировать детальные требования к системе.

Задачи:

- 1. Маркетинговое исследование**
 - Изучение конкурентов и аналогов (BoardGameGeek, Tabletopia, BGStats, GameFlix и др.).
 - Определение сильных и слабых сторон конкурентов.
 - Определение уникальных предложений для "BoardBox".
- 2. Анализ пользовательских потребностей**
 - Определение основных сценариев использования системы.
- 3. Формирование требований**
 - Составление списка функциональных и нефункциональных требований к системе.
 - Определение приоритетов разработки.
- 4. Подготовка документов**
 - Разработка технического задания на основе анализа.
 - Создание прототипов основных экранов и функций системы.

Результаты:

- Документ "Техническое задание на разработку автоматизированной системы "BoardBox".
- Прототипы основных экранов и функций системы.

5.2 Разработка архитектуры системы (25 марта 2025 г.)

Цель:

Спроектировать архитектуру системы, определить основные компоненты и их взаимодействие.

Задачи:

- 1. Архитектурное проектирование**
 - Определение общей архитектуры системы.
 - Разработка схемы взаимодействия основных компонентов.
- 2. Проектирование баз данных**

- Создание структуры баз данных для хранения информации о настольных играх, пользователях и коллекциях.
- Определение связей между таблицами и индексов для оптимизации запросов.
- 3. **Разработка API**
 - Спроектировать API для взаимодействия между клиентом и сервером.
 - Определение методов API для основных функций (регистрация, управление коллекциями, рекомендации).
- 4. **Интеграция с внешними сервисами**
 - Подготовка к интеграции с внешними API (например, OpenAI для генерации правил).
 - Определение протоколов и форматов обмена данными (JSON, REST).

Результаты:

- Архитектурная схема системы.
- Документация API.
- Схемы баз данных.

5.3 Разработка MVP (1 апреля 2025 г.)

Цель:

Разработать минимально жизнеспособный продукт (MVP) с основными функциями для тестирования и получения обратной связи от пользователей.

Задачи:

1. **Back-end разработка**
 - Реализация основных функций серверной части:
 - Регистрация и аутентификация пользователей.
 - Управление коллекциями игр (добавление, удаление, просмотр).
 - Простые фильтры поиска игр (жанр, количество игроков, длительность).
 - Настройка баз данных и хранение данных.
 - Подключение к внешним сервисам (OpenAI).
2. **Front-end разработка**
 - Создание мобильного приложения с основными экранами:
 - Экран регистрации и входа.
 - Экран управления коллекцией игр.
 - Экран поиска и фильтрации игр.
 - Реализация интерфейса для просмотра рекомендаций и кратких правил.
3. **Интеграция**
 - Подключение клиентской части к серверной через API.
 - Тестирование основных функций в интеграционной среде.

Результаты:

- Рабочая версия MVP с основными функциями.

- Документация по использованию API и интеграции с внешними сервисами.

5.4 Тестирование и отладка (май 2025 г.)

Цель:

Провести комплексное тестирование системы, выявить и исправить ошибки, улучшить UX/UI на основе обратной связи от пользователей.

Задачи:

1. **Функциональное тестирование**
 - Проверка всех основных функций системы.
 - Выявление и исправление ошибок и неполадок.
2. **Интеграционное тестирование**
 - Проверка взаимодействия между подсистемами.
3. **Оптимизация и улучшение UX/UI**
 - Корректировка интерфейса на основе обратной связи и тестирования.
 - Оптимизация производительности системы.

Результаты:

- Отчет о функциональном и интеграционном тестировании.
- Улучшенная версия MVP с исправленными ошибками и оптимизированным UX/UI.

5.5 Разработка полной версии (май 2025 г.)

Цель:

Расширить функционал MVP и довести систему до полной версии с дополнительными возможностями и улучшенным UX/UI.

Задачи:

1. **Расширение функционала**
 - Добавление социальных функций (делиться коллекциями).
 - Расширение фильтров и персонализированных рекомендаций (по настроению, ситуации).
2. **Улучшение UX/UI**
 - Оптимизация интерфейса для различных устройств и платформ.
3. **Масштабирование и оптимизация**
 - Подготовка системы к увеличению нагрузки и расширению аудитории.
 - Оптимизация производительности и безопасности.

Результаты:

- Полностью функциональная версия системы с расширенными возможностями.

- Документация по использованию системы для администраторов и пользователей.

5.6 Внедрение и публикация (декабрь 2025 г.)

Цель:

Запустить систему в производственную эксплуатацию и обеспечить ее доступность для пользователей.

Задачи:

- 1. Техническая подготовка**
 - Деплой системы на серверы и облачные сервисы.
 - Настройка инфраструктуры (сети, безопасности, мониторинга).
- 2. Маркетинговая поддержка**
 - Запуск рекламных кампаний для привлечения пользователей.
 - Создание пресс-релизов и анонсов на социальных платформах.
- 3. Обучение пользователей**
 - Создание инструкций

6. Порядок разработки автоматизированной системы

6.1 Порядок организации разработки АС

Разработка системы "BoardBox" будет организована в соответствии с методологией Agile, с использованием принципов Scrum и Kanban для гибкого управления проектом. Проектная команда будет разделена на кросс-функциональные подгруппы, каждая из которых будет отвечать за определенные модули и задачи. Регулярные спринты (1-2 недели) будут использоваться для инкрементного развития продукта, с обязательными еженедельными встречами для обсуждения прогресса и корректировки планов.

6.2 Перечень документов и исходных данных для разработки АС

1. **Техническое задание (ТЗ):** Документ, определяющий функциональные и нефункциональные требования к системе.
2. **Архитектурная документация:** Схемы и диаграммы, описывающие структуру системы и взаимодействие компонентов.
3. **Документация API:** Описание методов и протоколов для взаимодействия между клиентом и сервером.
4. **Макеты интерфейса:** Прототипы основных экранов и функций системы, созданные с использованием инструментов Figma.
5. **Исследование рынка и пользовательских потребностей:** Отчеты и данные, полученные в результате анализа конкурентов и опросов целевой аудитории.

6.3 Перечень документов, предъявляемых по окончании соответствующих этапов работ

1. **Этап "Исследование и формирование требований":**
 - Техническое задание (ТЗ).
 - Отчет о рынке и анализе конкурентов.
 - Прототипы основных экранов.
2. **Этап "Разработка архитектуры системы":**
 - Архитектурная схема системы.
 - Схемы баз данных.
 - Документация API.
3. **Этап "Разработка MVP":**
 - Демонстрационная версия MVP.
 - Отчет о функциональном тестировании MVP.
4. **Этап "Тестирование и отладка":**
 - Сводка по исправленным ошибкам и улучшениям.
5. **Этап "Разработка полной версии":**

- Полностью функциональная версия системы.
 - Документация для администраторов и пользователей.
6. Этап "Внедрение и публикация":
- Отчет о техническом деплое.
 - Маркетинговые материалы и пресс-релизы.

6.4 Порядок проведения экспертизы технической документации

1. **Внутренний аудит:** перед выпуском каждой версии документации, она будет проверена внутри команды на предмет соответствия ТЗ и технической корректности.
2. **Внешний аудит:** при необходимости, привлечение внешних экспертов для оценки архитектуры, безопасности и функциональности системы.
3. **Исправление замечаний:** Все выявленные замечания и рекомендации будут учтены и исправлены до выпуска следующей версии документации.

6.5 Перечень макетов и порядок их разработки, испытаний и документации

1. **Разработка макетов:**
 - Создание основных экранов (регистрация, управление коллекцией, поиск и рекомендации).
 - Использование инструментов Figma.
2. **Тестирование макетов:**
 - Проведение юзабилити-тестирования внутри команды.
 - Сбор обратной связи и корректировка UX/UI на основе результатов тестирования.
3. **Документация макетов:**
 - Описание основных экранов и функций.
 - Схемы навигации и взаимодействия пользовательских интерфейсов.

6.6 Порядок разработки, согласования и утверждения плана совместных работ по разработке АС

1. **Разработка плана:**
 - Создание детального плана проекта с распределением задач между членами команды.
 - Определение временных рамок и ключевых вех для каждого этапа разработки.
2. **Согласование плана:**
 - Проведение встреч внутри команды для обсуждения и корректировки плана.
 - Учет интересов всех заинтересованных сторон.
3. **Утверждение плана:**
 - Получение одобрения плана от руководства и заинтересованных сторон.

- Документирование утвержденного плана и его распределение среди команды.

6.7 Порядок разработки, согласования и утверждения программы работ по стандартизации

1. Разработка программы:

- Определение стандартов и протоколов, которые будут использоваться в проекте (например, REST API, OAuth для аутентификации).
- Создание плана по внедрению и соблюдению этих стандартов на всех этапах разработки.

2. Согласование программы:

- Обсуждение программы с командой и внешними экспертами для выявления потенциальных проблем и корректировки стандартов.

3. Утверждение программы:

- Получение одобрения программы от руководства проекта и заинтересованных сторон.
- Документирование утвержденной программы и ее распределение среди команды.

6.8 Требования к гарантийным обязательствам разработчика

1. Гарантийный срок

Разработчик обязуется предоставлять техническую поддержку и исправление критических ошибок в течение **3 месяцев** после выпуска полной версии системы.

2. Обязательства по устранению ошибок

- **Критические ошибки** (приводящие к полной неработоспособности системы) устраняются в течение **24 часов** с момента регистрации инцидента.
- **Серьёзные ошибки** (нарушающие основные функции системы, но не приводящие к полной неработоспособности) устраняются в течение **5 рабочих дней**.
- **Минорные ошибки** (не влияющие на основные функции системы) устраняются в плановом порядке в рамках выпуска обновлений.

3. Сервисное обслуживание

Разработчик предоставляет:

- Регулярные обновления и патчи для повышения функциональности и безопасности.
- Актуализацию документации в случае внесения значительных изменений в систему.

4. Консультационная поддержка

В течение гарантийного срока разработчик обязуется:

- Предоставлять консультации по установке, настройке и эксплуатации системы.
- Обеспечивать доступ к базе знаний и обучающим материалам.

- Отвечать на запросы пользователей в течение **2 рабочих дней**.

5. Ограничения гарантийных обязательств

Гарантия не распространяется на:

- Изменения, внесённые пользователем без согласования с разработчиком.
- Ошибки, вызванные аппаратными сбоями или сторонним программным обеспечением.
- Систему после окончания гарантийного срока, если не заключено дополнительное соглашение на продлённую поддержку.

6.9 Порядок проведения технико-экономической оценки разработки АС

1. Оценка затрат:

- Подготовка сметы с детализацией затрат на разработку, тестирование, инфраструктуру и маркетинг.
- Оценка трудозатрат для каждого этапа проекта.

2. Оценка рисков:

- Идентификация потенциальных рисков (технических, финансовых, маркетинговых) и разработка стратегий их минимизации.

3. Оценка экономической эффективности:

- Расчет ожидаемой ROI (возврата инвестиций) на основе прогнозных показателей использования системы и потенциальной аудитории.
- Подготовка отчета с рекомендациями по оптимизации затрат и повышению эффективности проекта.

7. Порядок контроля и приемки автоматизированной системы

7.1 Виды, состав и методы испытаний АС и ее составных частей

- 1. Функциональное тестирование**
 - **Цель:** Проверка соответствия реализованных функций требованиям технического задания.
 - **Объекты тестирования:** Регистрация и аутентификация пользователей, управление коллекцией игр, система рекомендаций, генерация кратких правил через ИИ, социальные функции.
 - **Методы:** Тестирование по сценариям, проверка граничных условий, тестирование отрицательных сценариев.
 - **Инструменты:** Ручное тестирование, юнит тестирование на языке go lang.
- 2. Интеграционное тестирование**
 - **Цель:** Проверка взаимодействия между подсистемами и внешними сервисами.
 - **Объекты тестирования:** Взаимодействие клиента и сервера через API, интеграция с OpenAI для генерации правил, аутентификация через OAuth.
 - **Методы:** Тестирование последовательности вызовов, проверка обмена данными в формате JSON, нагрузочное тестирование.
 - **Инструменты:** Postman.
- 3. Пользовательское тестирование**
 - **Цель:** Оценка удобства использования системы и выявление скрытых ошибок.
 - **Объекты тестирования:** Управление коллекцией, поиск и фильтрация игр, ИИ-объяснение правил и подбора рекомендаций.
 - **Методы:** Бета-тестирование с привлечением группы пользователей, анкетирование, фокус-группы.
 - **Инструменты:** Обратная связь, дневники использования.

7.2 Общие требования к приемке работ, порядок согласования и утверждения приемочной документации

- 1. Требования к приемке работ**
 - Система должна соответствовать требованиям технического задания.
 - Все функции должны быть протестированы и документированы.
 - Критические ошибки должны быть исправлены.
 - Система должна обеспечивать заявленную производительность и надежность.
- 2. Порядок согласования и утверждения приемочной документации**

- **Подготовка документов:** Составление отчетов о тестировании, технической документации и инструкций для пользователей.
- **Внутренний аудит:** Проверка документов на соответствие требованиям и полноту.
- **Согласование с заказчиком:** Предоставление документов заказчику для ознакомления и утверждения.
- **Утверждение:** Получение подписи заказчика и заинтересованных сторон на приемочной документации.

7.3 Статус приемочной комиссии

1. Состав комиссии

- Руководитель проекта.
- Представитель заказчика.

2. Функции комиссии

- Оценка результатов тестирования и испытаний.
- Принятие решения о готовности системы к запуску.
- Формирование заключения о соответствии системы требованиям технического задания.
- Рекомендации по доработке в случае выявления недостатков.

7.4 Доработка системы в случае обнаружения критических ошибок

1. Критерии критических ошибок

- Невозможность выполнения основных функций (регистрация, рекомендации, управление коллекцией).
- Нарушение безопасности и конфиденциальности данных.

2. Процесс доработки

- **Анализ ошибок:** Сбор информации о выявленных проблемах и их причинах.
- **Планирование исправлений:** Составление плана по устранению ошибок и определение сроков.
- **Исправление и повторное тестирование:** Внесение изменений в код и повторное проведение тестов.
- **Утверждение изменений:** Подготовка отчета о доработке и предоставление его комиссии для повторного рассмотрения.

7.5 Подготовка приемочной документации

1. **Отчет о функциональном тестировании**
 - Список протестированных функций.
 - Результаты тестирования (успешно/неуспешно).
 - Описание выявленных ошибок и их исправление.
2. **Отчет об интеграционном тестировании**
 - Схема взаимодействия подсистем.
 - Результаты проверки интеграции с внешними сервисами.
 - Выявленные проблемы и их решение.
3. **Отчет о пользовательском тестировании**
 - Сводка обратной связи от тестировщиков.
 - Оценка удобства использования и предложений по улучшению.
 - Внесенные изменения на основе обратной связи.
4. **Техническая документация**
 - Архитектурная схема системы.
 - Документация API.
5. **Заключение приемочной комиссии**
 - Оценка соответствия системы требованиям технического задания.
 - Рекомендации по дальнейшему развитию и улучшению.

7.6 Критерии успешной приемки

Система считается успешно принятой при выполнении следующих условий:

1. **Соответствие техническому заданию:** Все функции и характеристики должны соответствовать требованиям ТЗ.
2. **Производительность и надежность:** Система должна обеспечивать показатели, указанные в разделе 4.5.1 и 4.5.2 (время отклика, точность рекомендаций, доступность сервиса).
3. **Безопасность:** Реализация механизмов защиты данных и аутентификации пользователей в соответствии с требованиями раздела 4.5.3.
4. **Удобство использования:** Интерфейс должен быть интуитивно понятным и соответствовать требованиям раздела 4.5.5.
5. **Отсутствие критических ошибок:** Все критические ошибки, влияющие на работу системы, должны быть устранены.
6. **Утверждение документации:** Все приемочные документы должны быть согласованы и утверждены заказчиком.

8. Требования к составу и содержанию работ по подготовке объекта автоматизации к вводу автоматизированной системы в действие

8.1 Создание условий функционирования объекта автоматизации

1. Техническая инфраструктура

- Подготовка серверов и облачных ресурсов для размещения и функционирования системы.
- Установка и настройка необходимых программных и аппаратных средств.
- Настройка сетевой инфраструктуры для обеспечения высокой доступности и безопасности системы.

2. Интеграция с внешними сервисами

- Настройка и проверка интеграции с внешними API (OpenAI для генерации правил, OAuth для аутентификации).
- Обеспечение стабильного взаимодействия с внешними базами данных настольных игр и сервисами.

3. Системы мониторинга и аналитики

- Установка и настройка систем мониторинга производительности.
- Подготовка инструментов для сбора и анализа пользовательских данных для дальнейшей оптимизации системы.

9. Требования к документированию

9.1 Перечень подлежащих разработке документов

1. **Техническое задание (ТЗ)**
 - Основной документ, определяющий функциональные и нефункциональные требования к системе.
 - Содержит описание целей, назначения, характеристик объекта автоматизации и требований к системе.
2. **Архитектурная документация**
 - Схемы и диаграммы, описывающие структуру системы, взаимодействие компонентов и архитектурные решения.
 - Включает описание используемых технологий, протоколов и стандартов.
3. **Документация API**
 - Описание методов, параметров и форматов данных для взаимодействия между клиентом и сервером.
4. **Документация баз данных**
 - Схемы баз данных, описывающие структуру таблиц, индексы и связи.
5. **Макеты интерфейса и прототипы**
 - Визуальные представления основных экранов и функций системы.
6. **Тестовые документы**
 - Документация по выявленным ошибкам и их исправлению.

9.2 Вид представления и количество документов

1. **Текстовые документы**
 - ТЗ, архитектурная документация.
 - Формат: PDF или DOCX, количество: по одному на каждый тип документа.
2. **Графические документы**
 - Схемы архитектуры, схемы баз данных, макеты интерфейса.
 - Формат: PNG, JPG или векторные форматы (Figma, Adobe XD), количество: по числу компонентов и экранов.
3. **Тестовые документы**
 - Сценарии и отчеты о тестировании.
 - Формат: PDF или DOCX, количество: по одному на каждый этап тестирования.

10. Источники разработки

10.1 Технико-экономическое обоснование

1. Технико-экономическое обоснование проекта "BoardBox"

- Автор: Разработчики "BoardBox"
- Год: 2024
- Описание: Документ содержит анализ рынка настольных игр, оценку экономической эффективности проекта, прогнозные показатели и расчёты затрат.
- URL: <https://github.com/board-box/.github/upload/main/profile/documentation>

10.2 Отчёты о законченных научно-исследовательских работах

1. Настольные игры в России

- Автор: TAdviser
- Год: 2023
- Описание: Обзор современных алгоритмов рекомендательных систем, применение ИИ в генерации рекомендаций.
- URL: <https://www.tadviser.ru/index.php>

2. Анализ рынка настольных игр в России

- Автор: DISCOVERY RESEARCH GROUP
- Год: 2019
- Описание: Опросы и интервью с различными категориями игроков, сводка функциональных требований.
- URL: <https://drgroup.ru/>

10.3 Информационные материалы на отечественные и зарубежные системы-аналоги

1. BoardGameGeek

- Автор: BoardGameGeek LLC
- Год: 2024
- Описание: Крупнейшая база данных настольных игр, обзоры функциональности и пользовательские отзывы.
- URL: <https://boardgamegeek.com/>

2. Tabletopia и Board Game Arena

- Автор: Tabletopia Inc., Board Game Arena
- Год: 2024
- Описание: Онлайн-платформы для цифровых настольных игр, анализ преимуществ и ограничений.
- URL: <https://tabletopia.com/>, <https://boardgamearena.com/>

3. BGStats и аналогичные приложения

- Автор: BGStats
- Год: 2024
- Описание: Приложения для ведения статистики и управления коллекцией игр.
- URL: <https://www.bgstatsapp.com/>

10.4 Другие информационные материалы

1. Стандарты и методологии разработки

- **ГОСТ 34.602-2020** — «Техническое задание на создание автоматизированной системы»
 - Автор: Росстандарт
 - Год: 2020
 - URL: <https://docs.cntd.ru/document/1200181804>

2. Технические спецификации и архитектурные решения

- Автор: Разработчики "BoardBox"
- Год: 2024
- Описание: Используемые технологии (Golang, React Native, PostgreSQL, MongoDB, Redis), архитектурные схемы.
- URL: <https://github.com/board-box/.github/upload/main/profile/documentation>