

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ  
федеральное государственное бюджетное образовательное учреждение высшего образования  
«Тольяттинский государственный университет»

Кафедра Прикладная математика и информатика  
(наименование)

02.03.03 Математическое обеспечение и администрирование информационных систем  
(код и наименование направления подготовки / специальности)

Мобильные и сетевые технологии  
(направленность (профиль) / специализация)

## **ВЫПУСКНАЯ КВАЛИФИКАЦИОННАЯ РАБОТА (БАКАЛАВРСКАЯ РАБОТА)**

на тему: «Разработка мобильного приложения - помощника для обучающей настольной игры»

Обучающийся

Е.А.Буланов

(Инициалы Фамилия)

(личная подпись)

Руководитель

С.В.Митин

(ученая степень (при наличии), ученое звание (при наличии), Инициалы Фамилия)

Консультант

к.п.н., доцент, С.А. Гудкова

(ученая степень (при наличии), ученое звание (при наличии), Инициалы Фамилия)

Тольятти 2024

## Аннотация

Тема бакалаврской работы: «Разработка мобильного приложения - помощника для обучающей настольной игры»

В работе описан практический способ создания мобильного приложения помощника, а также самой настольной обучающей игры в жанре космос.

Проведено описание предметной области.

Рассмотрена игра как метод обучения для детей.

Выявлены проблемы современных настольных игр, а также предложены способы их решения.

Созданы логические модели программного обеспечения и настольной игры.

Поставлены требования к приложению помощнику.

Создан прототип интерфейса мобильного приложения.

Разработано программное обеспечение для Windows и Android на движке Renpy.

Интегрирован обучающий элемент в настольную игру.

Создана форма для проверки усвоения знаний игроков и сбора отзывов и пожеланий.

Разработан готовый интерфейс для приложения.

Разработан готовый дизайн для настольной игры.

Создана физическая копия настольной игры.

Проведено тестирование приложения и настольной игры на целевой аудитории.

Проведен анализ результатов тестирования.

Бакалаврская работа состоит из 50 страниц текста, 35 рисунков, 1 таблицы и 22 источников.

## **Abstract**

The title of the graduation work is «Development of mobile application - assistant for educational board game»

The paper describes a practical method for creating a mobile application assistant and a space-themed board game.

A detailed description of the subject area is provided.

The game is examined as a learning tool for children.

Problems with modern board games are identified, and potential solutions are proposed.

Logical models for the software and the board game are developed.

The requirements for the assistant application are established.

A prototype of the mobile application interface is created.

Software for Windows and Android is developed using the Renpy engine.

A learning element is integrated into the board game.

A form to check players' knowledge and collect feedback is created.

The interface for the application and the design for the board game are finalized.

A physical copy of the board game is produced.

The application and board game are tested with the target audience, and the results of the testing are analyzed.

The senior paper consists of 50 pages of text, 35 figures, 1 table and 22 sources.

## Оглавление

Введение .....	5
Глава 1 Функциональное моделирование предметной области .....	7
1.1 Техничко-экономическая характеристика предметной области .....	7
1.2 Анализ текущего рынка обучающих настольных игр .....	8
1.3 Постановка задачи на разработку мобильного приложения помощника .....	10
1.4 Улучшения настольной игры с помощью цифровых технологий....	11
Глава 2 Логическое проектирование настольной игры с мобильным приложением .....	13
2.1 Выбор технологии логического проектирования.....	13
2.2 Логическая модель настольной игры.....	14
2.3 Логическая модель программного обеспечения .....	22
2.4 Требования к настольной игре .....	25
2.5 Требования к программному лбеспечению .....	26
Глава 3 Реализация мобильного приложения и настольной игры .....	27
3.1 Выбор средства реализации программного обеспечения.....	27
3.2 Прототипирование интерфейса мобильного приложения .....	29
3.3 Разработка программного обеспечения.....	34
3.4 Интеграция обучающего элемента .....	39
3.5 Создание опроса для приложения .....	41
3.6 Готовый интерфейс приложения.....	43
3.7 Создание дизайна настольной игры .....	47
3.8 Создание физической копии игры .....	49
3.9 Тестирование настольной игры и приложения .....	50
3.10 Результаты проведения тестирования.....	52
Заключение .....	54
Список используемой литературы и используемых источников .....	56

## Введение

В современном мире технологии активно проникают в различные сферы жизни, включая образование и развлекательные индустрии. В связи с этим становится актуальной задача создания обучающих настольных игр с использованием цифровых помощников, которые могут улучшить процесс обучения и сделать его более увлекательным и комфортным.

Объектом исследования является процесс создания мобильного приложения-помощника для обучающей настольной игры на космическую тематику в жанре стратегия.

Предметом исследования является создание настольной игры с обучающим элементом, а также разработка и интеграция мобильного приложения-помощника на движке Ren'Py, которое облегчает игровой процесс, делает его удобнее и включает функционал для сбора обратной связи от игроков и проверки усвоенных знаний.

Целью данной работы является создание функционального мобильного приложения, которое автоматизирует и упрощает процесс игры, а также позволяет собирать данные о впечатлениях и знаниях игроков.

Для достижения поставленной цели необходимо решить следующие задачи:

- изучить рынок настольных обучающих игр;
- создать концепцию соревновательной игры с обучающим уклоном;
- спроектировать настольную игру в тематике «Космос» в жанре «Стратегия»;
- изучить движки для создания Android приложений;
- определить роль и функционал мобильного приложения в контексте настольной игры;
- спроектировать мобильное приложение;

- разработать мобильного помощника с помощью выбранных технологий;
- создать систему обратной связи и оценки закрепления знаний после игры;
- создать материальную часть настольной игры;
- проверить продукт на целевой аудитории;
- оценить эффективность настольной обучающей игры с использованием цифровых технологий.

Выпускная квалификационная работа состоит из введения, трёх глав, заключения, списка используемой литературы и приложения.

В первой главе проводится анализ и моделирование предметной области, исследуются проблемы и их решения. Построены функциональные модели «Как есть» и «Как должно быть».

Во второй главе проводится логическое проектирование настольной игры и приложения. Определяется внешний вид и структура. Проводится анализ функций и разработка требований к программному обеспечению и настольной игре.

В третьей главе выполнена разработка программного программного обеспечения и настольной игры, проведено тестирование на целевой аудитории и анализ усвоения знаний игроками.

## **Глава 1    Функциональное моделирование предметной области**

### **1.1    Техничко-экономическая характеристика предметной области**

#### **1.1.1    Игра как метод обучения**

Игра представляет собой не просто развлечение для детей, но и естественный способ обучения, который позволяет им исследовать мир, развивать навыки и усваивать информацию. Эффективность игры как метода обучения обусловлена рядом факторов.

Во-первых, внутренняя мотивация: игра увлекает детей и побуждает их к познанию и пробам нового. Во-вторых, практический опыт: в процессе игры дети учатся на собственном опыте, исследуя и экспериментируя без страха неудачи. Эмоциональная вовлеченность также играет важную роль: игра погружает детей в процесс обучения, повышая концентрацию внимания и способствуя запоминанию информации. Кроме того, социальное взаимодействие: многие игры имеют коллективный характер, что учит детей сотрудничать, общаться и решать проблемы вместе. Наконец, развитие креативности: игра стимулирует творческое мышление и воображение, позволяя детям придумывать новые правила, создавать истории и использовать различные подходы к решению задач.

Многочисленные исследования подтверждают преимущества игры как метода обучения. Дети, обучающиеся через игру, демонстрируют лучшие результаты в учебе. Уровень развития социальных навыков у таких детей выше, а их творческий потенциал более развит.

#### **1.1.2    Настольные игры: польза и влияние на обучение детей**

Настольные игры обладают значительным образовательным и развивающим потенциалом для детей, превращая процесс обучения в увлекательное занятие. Они способствуют когнитивному развитию, стимулируя логическое мышление, память и внимание. Дети учатся планировать свои

действия, принимать решения и справляться с последствиями своих выборов, развивая критическое мышление и способность к решению проблем.

Кроме того, настольные игры способствуют социальному развитию. Взаимодействуя во время игры, дети учатся сотрудничать, следовать правилам и уважать мнения других, что важно для успешной адаптации в школьной и социальной среде.

Игры могут быть эффективным инструментом в обучении конкретным предметам, таким как математика, история, физика или языки, повышая мотивацию к обучению и улучшая запоминание информации.

Настольные игры также развивают эмоциональный интеллект. Дети учатся справляться с разными эмоциями, развивают терпение и самоконтроль, что необходимо для управления своими чувствами в различных ситуациях.

Таким образом, настольные игры помогают детям весело проводить время и развиваться в когнитивном, социальном и эмоциональном плане, что делает их незаменимым элементом в образовательном процессе.

## **1.2 Анализ текущего рынка обучающих настольных игр**

### **1.2.1 Рынок обучающих настольных игр: состояние и проблемы**

Рынок обучающих настольных игр в настоящее время находится в состоянии стагнации.

Хоть и спрос на настольные игры в целом растет. В связи с такими факторами, как поиск семьями способов совместного досуга, рост интереса к ретро-развлечениям и популярность настольных игр среди взрослых. У современных детей популярность настольных игр не показывает значительного роста.

Существует несколько причин падения популярности обучающих настольных игр у современных детей:

- конкуренция со стороны электронных устройств;
- ограниченный ассортимент;



- мнение о потере актуальности настольных игр;
- отсутствие оригинальных идей на рынке.

Несмотря на эти проблемы, рынок обучающих настольных игр имеет потенциал для роста. Особенно с внедрением в них цифровых технологий.

### **1.2.2 Обоснование интеграции мобильного приложения помощника в настольную игру**

Внедрение мобильного приложения-помощника в настольную игру имеет множество преимуществ, которые делают игру более привлекательной и эффективной, особенно для современных детей.

Во-первых, мобильное приложение упрощает игровой процесс, предоставляя удобные инструменты для подсчета очков, управления правилами и отслеживания прогресса. Это избавляет игроков от необходимости вручную выполнять рутинные задачи, что позволяет полностью сосредоточиться на стратегии и обучении.

Во-вторых, приложение предоставляет интерактивный опыт, который может включать анимации, звуковые эффекты и визуальные подсказки, делая игру более захватывающей и динамичной. Такие элементы помогают удерживать внимание детей, делая процесс обучения увлекательным и интересным.

Кроме того, мобильное приложение облегчает сбор обратной связи от игроков. После каждой игровой сессии дети могут оставить свои эмоции и пожелания, что позволяет разработчикам постоянно улучшать игру. Встроенные опросы и тесты помогают оценить, насколько хорошо были усвоены знания, полученные в ходе игры [13].

Дети, рожденные в цифровую эпоху, с легкостью осваивают мобильные устройства и технологии. Интеграция мобильного приложения в настольную игру соответствует их привычкам и увлечениям, делая обучение через игру естественным и привлекательным. Мобильное приложение-помощник не только повышает удобство и интерактивность игры, но и способствует более

эффективному усвоению знаний, привлекая внимание и интерес современных детей.

### **1.3 Постановка задачи на разработку мобильного приложения помощника**

Цели мобильного приложения-помощника:

- автоматизация процесса игры, включая чтение правил, просмотр магазина построек и техники, подсчет ходов и напоминание игрока о необходимости взять нужную карточку;
- обеспечение удобного интерфейса для подсчета ресурсов игрока, а также ресурсов его планет и звезд.

Требования функциональности:

- интерфейс для чтения правил игры и ознакомления с игровыми элементами через удобное мобильное приложение;
- просмотр построек и техники в игровом магазине;
- автоматизированный подсчет ходов и напоминания игрокам о действиях, которые необходимо выполнить, включая взятие карточек;
- подсчет и отображение ресурсов игрока, а также ресурсов его планет и звезд;
- показ вопросов по астрономии с возможностью получения ресурсов за правильные ответы;
- опросник для сбора обратной связи от игроков после завершения партии.

Требования к архитектуре и реализации:

- нетребовательность к устройству, оптимизация программы;
- доступность приложения на разных операционных системах, таких как Windows, Linux, MacOS, Android, IOS, Web;

- гибкость и масштабируемость системы для легкого интегрирования новых карт и механик в игру.

Создание мобильного приложения-помощника для игры - это существенный шаг к автоматизации и улучшению игрового процесса. Такое приложение не только упростит управление ресурсами и ходами, но и предоставит пользователям удобный интерфейс для изучения игровых элементов и правил. С учетом требований к функциональности и архитектуре это приложение сможет эффективно помогать игрокам, обеспечивая удобство и гибкость использования на различных устройствах и операционных системах.

#### **1.4 Улучшения настольной игры с помощью цифровых технологий**

Сравнение различных элементов настольной игры до и после внедрения мобильного приложения помощника представлено в таблице 1. Из таблицы видно, что автоматизация некоторых процессов с помощью приложения принесла ряд улучшений. Теперь игроки могут с легкостью управлять ресурсами и ходами, а также быстро получать доступ к информации о игровых элементах и правилах. Кроме того, приложение обеспечивает удобство использования на различных устройствах и операционных системах, что делает игру более доступной для широкой аудитории.

Таблица 1 – Сравнение аспектов игры «Как было» и «Как будет».

Аспект	Как было	Как будет
Просмотр правил и магазина игры	Отдельные бумажные инструкции, которые занимают много игрового пространства, а также сбивают с темпа игры.	Вся текстовая информация будет доступна в мобильном приложении, что избавит игру от лишней бумаги и сделает игровой процесс более сконцентрированным.
Подсчет ходов	Ручной подсчет ходов и ресурсов может привести к ошибкам и спорам.	Автоматизированный подсчет ходов и ресурсов через мобильное приложение, что обеспечивает точность и минимизирует возможность ошибок.

Продолжение таблицы 1

Аспект	Как было	Как будет
Напоминания игрокам	Игрокам приходилось держать в уме, когда настанет их черед брать карту.	Персонализированные напоминания для игроков через мобильное приложение о важных моментах в игре, таких как взятие нужной карточки или использование специальных возможностей.
Карточки с вопросами и ответами	Содержание огромного количества бумажных карточек, а также отсутствие возможности добавления новых.	Карточки с вопросами хранятся в приложении, что позволяет избежать множества лишних карточек, а также оставляет возможность легкого добавления новых.
Подсчет ресурсов игроков	Запись на листочке. Каждый раз приходится обновлять данные и вести множественные записи.	Ресурсы игрока и его планет хранятся в приложении, легко обновляются и не забирают игровое пространство.
Цифровой опросник	Отсутствует.	Позволяет получить отзывы и пожелания по игре, а также проверяет знания, полученные во время партии.

В первой главе была представлена основная информация о методе обучения детей через игру, а также влияние настольных игр на учебный процесс. Также были выявлены существующие недостатки современных настольных игр и определен объект для дальнейшей автоматизации с учетом выявленных проблем. На основе проведенного анализа определены функциональные и архитектурные требования к программному обеспечению.

## **Глава 2 Логическое проектирование настольной игры с мобильным приложением**

### **2.1 Выбор технологии логического проектирования**

Для логического проектирования используются тип моделей UML.

UML – это язык моделирования, который помогает разработчикам программного обеспечения визуализировать, понимать и проектировать системы. Он предоставляет набор графических элементов и правил для создания моделей, которые могут быть использованы для документирования требований, проектирования архитектуры, разработки кода и тестирования системы [4], [11], [17].

Для реализации логического проектирования применяется инструмент `diagrams.net`.

`Diagrams.net` – это бесплатный онлайн-инструмент с открытым исходным кодом для создания различных диаграмм, таких как UML, блок-схемы, сетевые схемы и многое другое. Он обладает простым интерфейсом и богатым функционалом, делая его удобным как для новичков, так и для опытных пользователей. `Diagrams.net` интегрируется с облачными сервисами (Google Drive, OneDrive, Dropbox), позволяя легко работать над проектами совместно.

Одним из ключевых преимуществ `diagrams.net` является его бесплатность и открытый исходный код, что делает его доступным для широкого круга пользователей без необходимости приобретения лицензий или подписок. В отличие от многих других платных инструментов, `diagrams.net` обеспечивает полный набор функций без ограничений, таких как лимиты на количество создаваемых диаграмм или ограниченный доступ к инструментам. Кроме того, `diagrams.net` поддерживает интеграцию с различными облачными сервисами, что упрощает совместную работу и обмен диаграммами в командах. Пользовательский интерфейс интуитивно понятен и прост в освоении, что позволяет быстро начинать работу без необходимости длительного обучения.

## 2.2 Логическая модель настольной игры

Игра в жанре стратегия / выживание по теме космического будущего. Предполагает игровую сессию на 30-60 минут для 1-4 игроков в стартовом наборе. Возраст игроков – 7 лет и выше.

Схема, представляющая все типы материальных объектов в игре, представлена на рисунке 1.

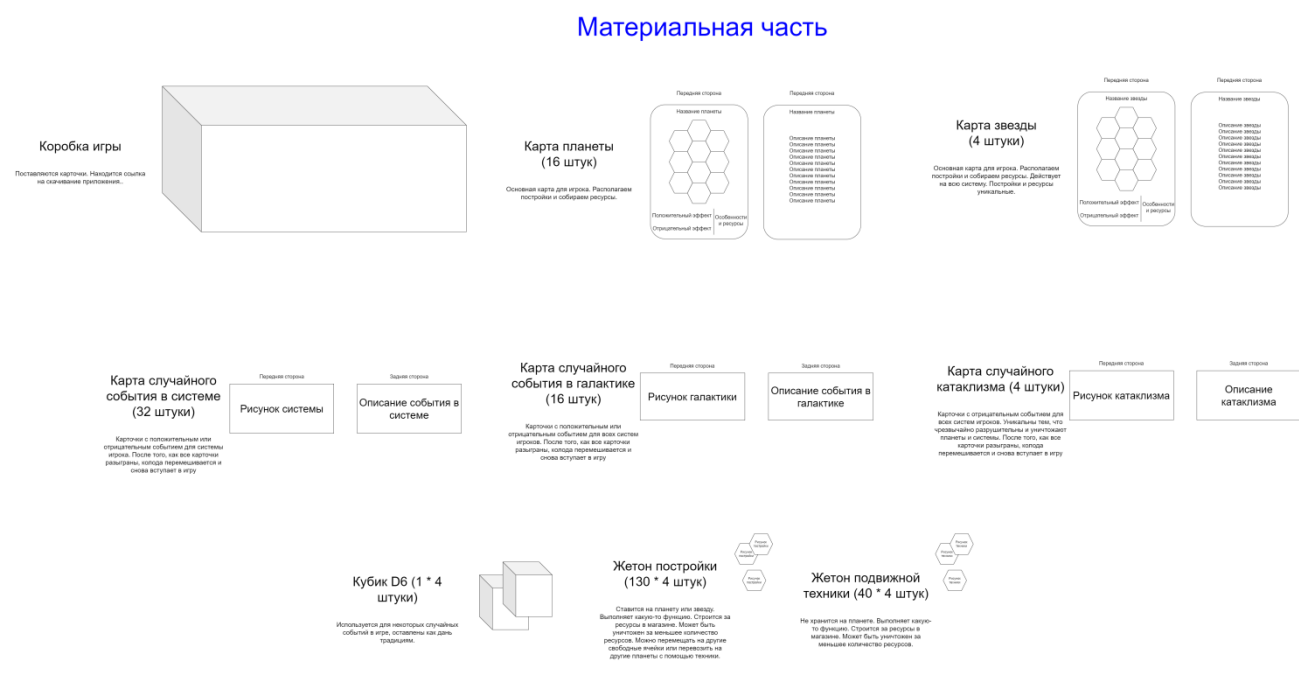


Рисунок 1 – Схема материальной части

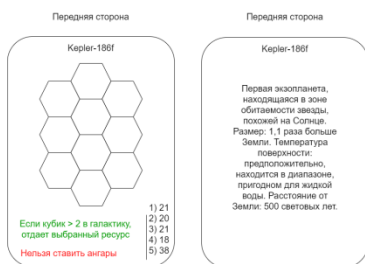
На схеме указаны названия объектов, их количество, примерный вид, а также роль в игре.

Для детального отображения всех объектов в игре используются схемы, включающие каждый тип карточек и жетонов.

Схема, показывающая тип карточек «планеты» представлена на рисунке 2.

## 2 Карта планеты (16 штук)

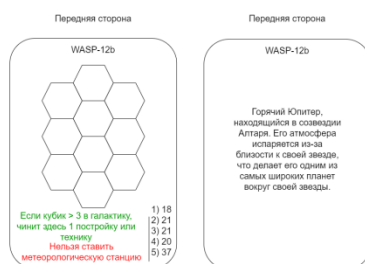
2.1



2.2



2.5



2.6

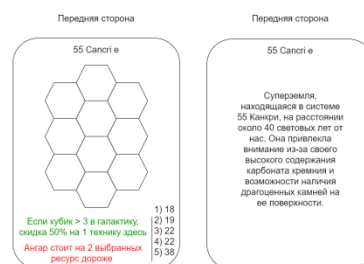


Рисунок 2 – Схема карт планет

На схеме указаны названия планет, их описание, ресурсы, положительные и негативные свойства. Также на карточке изображены 10 гексагонов, которые позволяют игроку на них размещать свои постройки.

Всего в игре используется 16 различных карточек реально существующих планет, каждая со своими уникальными свойствами и набором ресурсов.

По правилам игры, каждый игрок получает в начале партии по 4 случайных карточки планеты и 1 карточку звезды. Это будет его звездной системой.

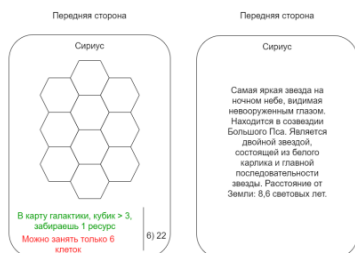
Игрок выбирает планету, на которой поставит свою базу и начнет развивать империю. Каждый ход будут применяться положительные и негативные свойства планеты, а также собираться ее ресурсы – древесина, каменные материалы, металл, химические материалы и топливо.

Ресурсы на планетах ограничены, что мотивирует игроков после определенного времени покинуть свои планеты в поисках новых для поддержания жизни цивилизации.

Схема, показывающая тип карточек «звезды» представлена на рисунке 3.

### 3 Карта звезды (4 штуки)

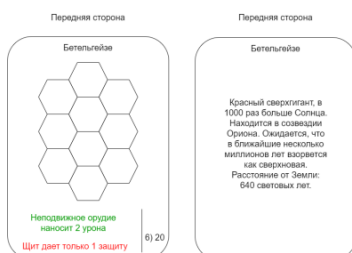
3.1



3.2



3.3



3.4



Рисунок 3 – Схема карт звезд

На схеме указаны названия звезд, их описание, ресурсы, положительные и негативные свойства. Также на карточке изображены 10 гексагонов, которые позволяют игроку на них размещать свои постройки.

Всего в игре используется 4 различных карточки реально существующих звезд, каждая со своими уникальными свойствами и набором ресурсов.

В стартовом наборе используются 4 карточки звезды, что позволяет играть 4 игрокам в партии, так как в одной системе не может быть больше 1 звезды.

Игрок не может изначально поставить базу на звезде, так как для нее нужна специальная постройка «солнцезащитная база». С помощью нее игрок может ставить постройки и создавать технику на звезде.

Звезды уникальны тем, что предоставляют только 1 вид ресурсов – солнечная энергия. Эксклюзивный тип, который позволяет создавать определенные виды построек и техники, а также использоваться как топливо.

Схема, показывающая тип карточек “случайные события в звездной системе” представлена на рисунке 4.



## 5 Карты случайного события в системе (32 штук)



Рисунок 4 – Схема карт случайных событий в звездной системе

На схеме указаны названия случайных событий в звездной системе и сами события.

Всего в игре используется 32 различных карточки событий в системе. Они могут иметь как негативные, так и положительные эффекты для игроков.

Особенность этих карточек заключается в том, что действуют они только на систему игрока и его захваченные территории.

Карточки берутся от старшего игрока к младшему каждые 4 хода и применяются в начале следующего хода.

Пример карточки случайного события в звездной системе – «Рост популяции. Если кубик > 3, на этом ходу вам нужно отдавать на 1 ресурс больше.». В некоторых событиях необходимо кинуть кубик, чтобы понять сработает ли эффект, в данном случае если кубик показал значения 4, 5 или 6, игрок отдает в этом ходу 1 свой дополнительный ресурс любого типа.

После того как все 32 карточки были разыграны, колода перемешивается и снова вступает в игру.

Схема, показывающая тип карточек «случайные события в галактике» представлена на рисунке 5.

## 6 Карты случайного события в галактике (16 штук)



Рисунок 5 – Схема карт случайных событий в галактике

На схеме указаны названия случайных событий в галактике и сами события.

Всего в игре используется 16 различных карточки событий в галактике. Они могут иметь как негативные, так и положительные эффекты для всех игроков.

Особенность этих карточек заключается в том, что действуют они уже на всю галактику, то есть на все системы игроков сразу.

Карточки берутся от старшего игрока к младшему каждые 16 ходов и применяются в начале следующего хода после карточки случайного события в звездной системе.

Пример карточки случайного события в галактике – «Глобальная перестройка. На этом ходу 3 выбранных игроками постройки или техники вернут при сноске 100% ресурсов.». В данном примере игрокам выпал

положительный эффект, и они могут без потери ресурсов снести 3 ненужные постройки или единицы техники.

После того как все 16 карточек были разыграны, колода перемешивается и снова вступает в игру.

Схема, показывающая тип карточек «катаклизмы» представлена на рисунке 6.



Рисунок 6 – Схема карт катаклизмов

На схеме указаны названия случайных катаклизмов в галактике. Их работа указана в правилах игры, которые хранятся в мобильном приложении.

Всего в игре используется 4 различных карточки катаклизма. Все они имеют только крайне негативные эффекты для всех игроков.

Особенность этих карточек заключается в том, что действуют они уже на всю галактику, то есть на все системы игроков сразу, а также могут привести к концу игры, если не будут своевременно уничтожены.

Карточки берутся от старшего игрока к младшему каждые 32 хода и применяются в начале следующего хода после карточки случайного события в галактике.

Пример карточки катаклизма – «Сверхмассивная черная дыра - с помощью кубика решается в какой системе она появится, а затем опять же с помощью кубика какая планета или звезда. Черная дыра не дает через себя пролетать, а также поглощает следующую планету или звезду справа от нее, каждую карточку системы. Чтобы ее уничтожить нужно выстрелить из

лазерного орудия в нее 2 раза.». Черная дыра будет поглощать планеты и звезды справа от нее каждую карточку системы, пока не уничтожит все на своем пути. Чтобы ее остановить игрокам нужно выстрелить 2 раза в нее из лазерного орудия.

После того как все 4 карточки были разыграны, колода перемешивается и снова вступает в игру.

Схема, показывающая тип жетонов «постройки» представлена на рисунке 7.



Рисунок 7 – Схема жетонов построек

На схеме указаны названия построек, их количество, назначение, цена и обслуживание.

Всего в игре используется 16 уникальных типов построек. У каждого игрока есть лимит покупки построек определенного типа. Например, любой игрок не может построить больше 16 лесопилок в своей системе. Если ему необходимо больше, он может захватить чужие постройки.

Постройки могут быть созданы на той планете или звезде, где игрок имеет свою базу.

Жетоны построек покупаются за ресурсы игрока на этапе действий и сразу же ставятся на свободный гексагон на планете или звезде. Работать же они начнут уже только в следующем ходу.

Игроки могут освобождать клетки путем сноски построек. При сноски возвращается только половина ресурсов на выбор игрока (Округляется математически). Постройки можно переносить с помощью грузовых кораблей. При переноске базы, все постройки планеты становятся ничейными.

Постройки могут быть разных видов – добывающие ресурсы, защищающие, атакующие, транспортные и т.д.

Схема, показывающая тип жетонов «подвижная техника» представлена на рисунке 8.



Рисунок 8 – Схема жетонов подвижной техники

На схеме указаны названия типа техники, их количество, назначение и цена.

Всего в игре используется 8 уникальных типов подвижной техники. У каждого игрока есть лимит покупки техники определенного типа. Например, любой игрок не может построить больше 5 торговых кораблей у себя в ангарах. Если ему необходимо больше, он может захватить ангарах других игроков и создать больше техники.

Техника может быть создана на той планете, где игрок имеет свой ангар.

Жетоны подвижной техники покупаются за ресурсы игрока на этапе действий и сразу же ставятся на орбиту планеты, где у игрок расположил свой ангар, не занимая гексагоны. Работать же они начнут уже только в следующем ходу.

Игроки могут также сносить и подвижную технику. При сноски возвращается только половина ресурсов на выбор игрока (Округляется математически). Если были захвачены все базы какого-либо игрока, его подвижная техника, также переходит под контроль захватившего.

Подвижная техника может быть разных видов – торговая, грузовая, боевая, ремонтная и т.д.

### **2.3 Логическая модель программного обеспечения**

Для проектирования логической работы мобильного приложения используется диаграмма активности UML, представленная на рисунке 9.

Диаграммы активности используются для моделирования динамического поведения системы. Они показывают, как потоки управления перемещаются между действиями в системе, а также условия, при которых эти переходы происходят [4], [19].

Она демонстрирует полный жизненный цикл программы от начала партии до ее конца.

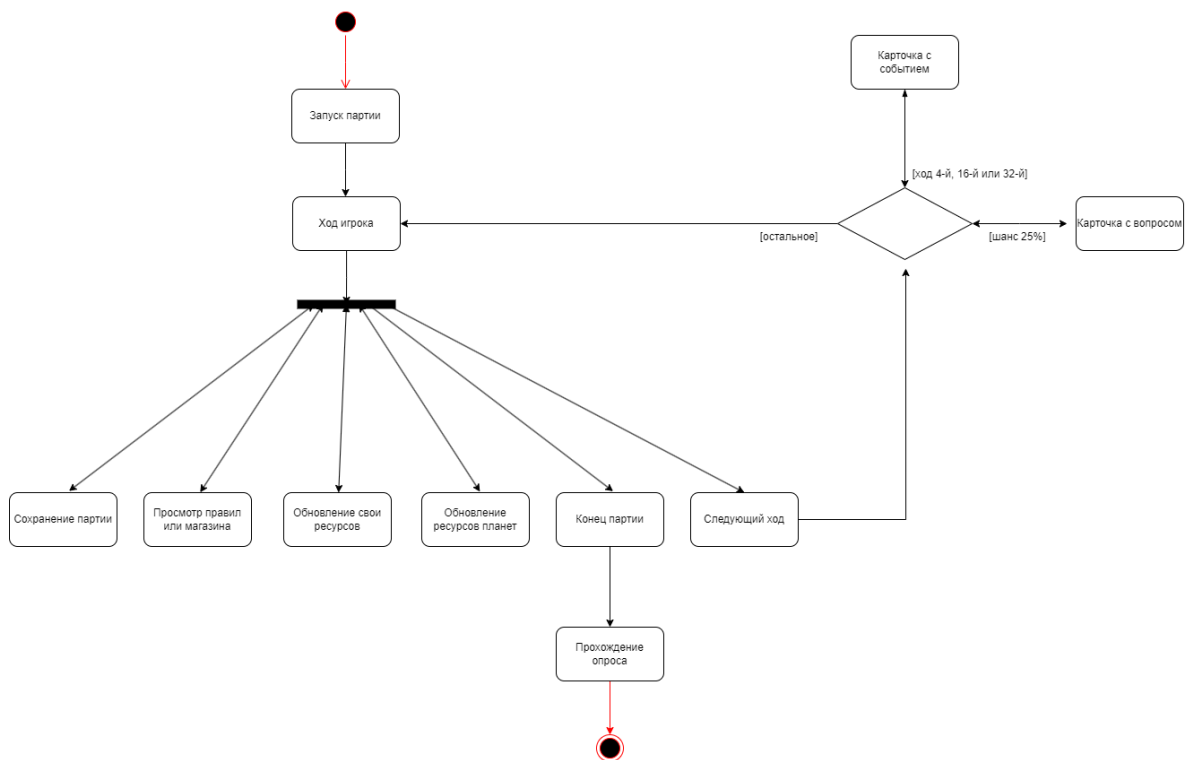


Рисунок 9 – Диаграмма активностей мобильного приложения

Программа запускается в начале партии и функционирует непрерывно, пока игрок не завершит игру. В каждом ходе игроку предоставляется возможность совершать различные действия, такие как переход на следующий ход. При переходе на следующий ход игрок может получить случайную карточку с вопросом, что добавляет элемент неожиданности в игровой процесс. Кроме того, если номер хода делится на 4, 16 или 32, игрок получает карточку случайного события, что также разнообразит и усложняет игру.

Для обеспечения использования мобильного приложения игроком применяется диаграмма прецедентов, которая иллюстрирует все возможные действия и взаимодействия игрока с приложением. Эта диаграмма представлена на рисунке 10, и она помогает разработчикам понять и реализовать все функциональные возможности, которые могут понадобиться игроку.

Данная диаграмма содержит такие варианты использования игроком:

- окончание партии – завершает программу и предлагает игроку пройти опрос на Google Forms;

- просмотр правил и магазина – игрок в специальном меню просматривает нужную ему информацию;
- обновление своих ресурсов – уменьшение или увеличение своих личных ресурсов;
- обновление ресурсов своих планет и звезд – игрок может добавлять планеты, удалять их, а также менять ресурсы на них;
- окончание хода – завершение текущего хода и показ карточек случайных событий и карточек вопросов, если они есть.

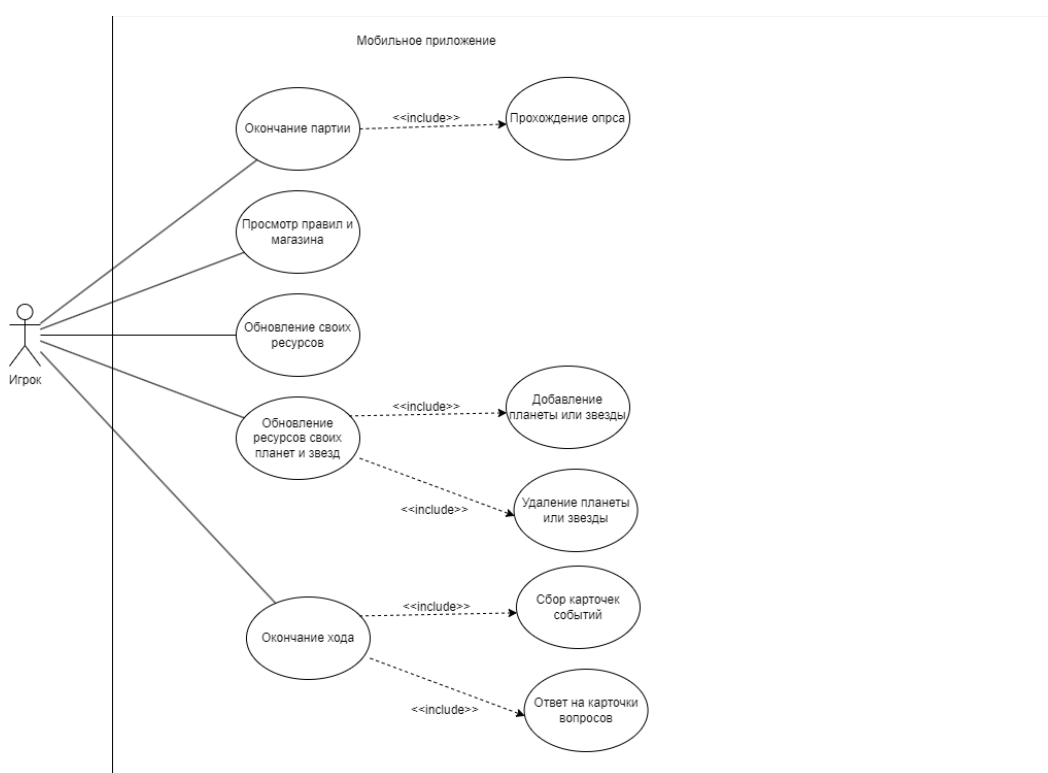


Рисунок 10 – Диаграмма прецедентов мобильного приложения

UML-диаграммы прецедентов являются мощным инструментом для описания функциональных требований системы. Они помогают аналитикам и разработчикам понять, что система должна делать с точки зрения пользователей, и служат основой для дальнейшего проектирования системы [4].



## 2.4 Требования к настольной игре

Также игра должна соответствовать следующим критериям в рамках дизайна:

- изображения планет и звезд должны примерно походить на их реальные прототипы;
- карточки планет и звезд не должны быть перегружены и легко читаться;
- жетоны техники и построек должны легко определяться;
- карточки случайных событий должны легко читаться, а также на обратной стороне иметь изображение типа карточки – случайное событие в звездной системе, случайное событие в галактике или случайный катаклизм;
- игра должна выглядеть визуально привлекательно для детей от 7 до 18 лет.

Для комфортного использования, игра также должна удовлетворять следующим требованиям:

- большой размер карточек звезд и планет. На них должны уместиться 10 жетончиков;
- использование специальной бумаги для приятного тактильного взаимодействия с карточками, а также долговечности игры;
- использование обычной бумаги для жетонов в целях оптимизации расходов на игру;
- небольшой размер карточек случайных событий, использование специальной бумаги;
- использование обычных игровых кубиков D6;
- наличие коробки и пакетиков для карт для большего удобства.

## 2.5 Требования к программному обеспечению

Для эффективной работы приложения на Android, мобильное устройство должно соответствовать следующим техническим характеристикам:

- процессор с 2 ядрами;
- тактовая частота процессора не менее 1,5 ГГц;
- объем оперативной памяти не менее 1 Гб;
- свободное место на жестком диске не менее 100 Мб;
- подключение к сети;
- Android 10 и выше.

Для использования приложения на ПК, устройство должно соответствовать следующим техническим характеристикам:

- процессор Pentium® 4 1.5 GHz / Athlon® XP;
- видеокарта DirectX 9.0 и выше;
- объем оперативной памяти не менее 1 Гб;
- свободное место на жестком диске не менее 200 Мб;
- подключение к сети;
- Windows 7 и выше.

При проектировании будущего приложения и настольной игры были тщательно учтены все ключевые аспекты и стандарты, необходимые для успешной разработки. В рамках этого процесса с использованием UML были созданы диаграммы прецедентов и активности, которые помогли структурировать и визуализировать основные сценарии использования и последовательность действий в системе [6], [7].

Помимо этого, были разработаны детальные логические схемы каждого элемента настольной игры, включая карточки планет и звезд, карточки событий, а также жетоны построек и техники. Эти схемы обеспечили ясное понимание функциональности и взаимодействий всех компонентов игры, что позволило создать более целостную и проработанную концепцию проекта.

## Глава 3 Реализация мобильного приложения и настольной игры

### 3.1 Выбор средства реализации программного обеспечения

Android – это операционная система, экосистема для мобильных устройств. В основе Android лежит модифицированное ядро Linux, что делает ее открытой и доступной для всех. Разработанная консорциумом Open Handset Alliance под руководством Google, платформа Android быстро завоевала популярность благодаря своей гибкости, удобству и открытости [2], [8], [9].

Ключевыми особенностями Android являются:

- открытый исходный код;
- большее количество поддерживаемых устройств;
- простой и понятный интерфейс;
- большой выбор приложений;
- постоянное развитие операционной системы.

Под Android существует множество различных инструментов для создания приложений, таких как Android Studio, Unity, IntelliJ IDEA, Eclipse и т.д. Но в ходе изучения программ, был выбран движок для создания игр – RenPy, как самый подходящий под требования [1], [5], [12].

Ren'Py – это открытая платформа для разработки визуальных новелл и интерактивных игр, популярная среди независимых разработчиков и студий. Движок основан на языке программирования Python. Основное назначение Ren'Py – создание интерактивных историй с графикой, звуком и текстом, где игроки могут делать выборы, влияющие на развитие сюжета. Программа позволяет разработчикам создавать сложные и ветвящиеся повествования, обеспечивая пользователям захватывающий опыт взаимодействия [22].

Одним из главных преимуществ Ren'Py является его доступность. Благодаря простому синтаксису и интуитивно понятному интерфейсу, пользователи могут быстро освоить основы создания визуальных новелл. Использование языка Python для более сложных скриптов позволяет опытным

разработчикам внедрять собственные функции и механики, расширяя возможности движка.

Еще одним важным преимуществом Ren'Py является его мощная система управления ресурсами. Движок поддерживает различные мультимедийные форматы, включая изображения, звуковые эффекты, музыку и видео, что позволяет создавать насыщенные и атмосферные игровые миры. Гибкая система слоев и возможностей анимации помогает разработчикам визуализировать свои идеи с высокой степенью детализации. Кроме того, Ren'Py обладает встроенными инструментами для локализации, что значительно упрощает процесс перевода игр на другие языки и расширяет аудиторию разработчиков.

Кроссплатформенность – еще один значительный плюс Ren'Py. Игры, созданные на этом движке, можно запускать на различных операционных системах, включая Windows, macOS, Linux, а также мобильные платформы Android и iOS. Это обеспечивает широкий охват и доступность игр для самых разных пользователей. Наконец, активное сообщество разработчиков и обширная документация помогают быстро находить решения возникающих проблем и получать поддержку, что делает процесс разработки более гладким и эффективным.

Это выделяет движок RenPy на фоне других конкурентов. Сам движок написан и использует у себя язык Python.

Python - это популярный высокоуровневый язык программирования, созданный Гвидо ван Россумом и выпущенный в 1991 году. Его ключевая особенность - это лаконичный и выразительный синтаксис, который позволяет писать понятный и читаемый код. Python известен своей простотой и доступностью для начинающих программистов, но при этом является мощным инструментом для опытных разработчиков, так как позволяет решать сложные задачи с минимальным количеством кода. Синтаксис Python направлен на минимизацию кода, необходимого для выполнения задач, что делает его удобным и эффективным языком программирования [3], [9], [15].

Python также пользуется поддержкой активного и дружелюбного сообщества. Разработчики со всего мира вносят свой вклад в развитие языка, создавая и поддерживая библиотеки, составляя документацию и помогая друг другу на форумах и в группах. Это способствует быстрому решению проблем и обмену знаниями, что особенно полезно для начинающих [14], [15].

Python также отличается кроссплатформенностью, что является еще одним важным преимуществом. Программы, созданные на Python, могут запускаться на разных операционных системах, таких как Windows, macOS и Linux, без существенных изменений в коде. Это дает разработчикам возможность создавать приложения, которые легко переносятся и работают на различных платформах. Python является чрезвычайно гибким языком программирования, позволяющим создавать приложения, не ограниченные одной операционной системой [10], [16], [21].

### **3.2 Прототипирование интерфейса мобильного приложения**

Для представления пользовательского интерфейса в приложении был создан прототип в программе Figma.

Figma — это облачное приложение для дизайна и прототипирования пользовательских интерфейсов, которое завоевало популярность среди дизайнеров и команд разработчиков. Выпущенное в 2016 году, Figma быстро стало известным благодаря своим мощным функциям и возможностям для совместной работы в реальном времени. Основная цель Figma — помочь дизайнерам создавать привлекательные и функциональные интерфейсы для веб-сайтов, мобильных приложений и других цифровых продуктов. С Figma дизайнеры могут разрабатывать дизайн пользовательского интерфейса (UI) и опыта взаимодействия (UX), чтобы обеспечить удобство и привлекательность цифровых продуктов для пользователей. Одно из главных преимуществ Figma заключается в том, что это полностью облачное приложение [18].

В отличие от традиционных программ для дизайна, которые необходимо устанавливать на компьютер, Figma работает прямо в браузере, что делает его доступным на любой платформе без необходимости установки дополнительного программного обеспечения. Это делает процесс совместной работы более простым и эффективным, так как участники команды могут одновременно работать над одним проектом, независимо от их местонахождения или используемой ими операционной системы.

Figma обладает ещё одним важным преимуществом — это его мощный набор инструментов для дизайна и прототипирования. В Figma можно создавать векторную графику, компоненты и стили, что обеспечивает согласованность и повторное использование элементов дизайна. Функции автоматического макета и вариативные компоненты позволяют быстро адаптировать дизайн для различных устройств и экранов. Прототипирование в Figma позволяет создавать интерактивные макеты без необходимости писать код.

Прототип дизайна, представляющий два вида меню в приложении, представлен на рисунке 11.



Рисунок 11 – Прототип главного и игрового меню

В нем игрок сможет начать новую партию, продолжить старую, зайти в настройки, а также прочитать информацию об игре.

В меню, которое можно вызвать во время игры, доступны новые функции сохранения, загрузки, окончания партии, а также выхода в главное меню. Игровое меню включается по кнопке «Меню» во время партии.

Прототип дизайна раздела настроек представлен на рисунке 12.

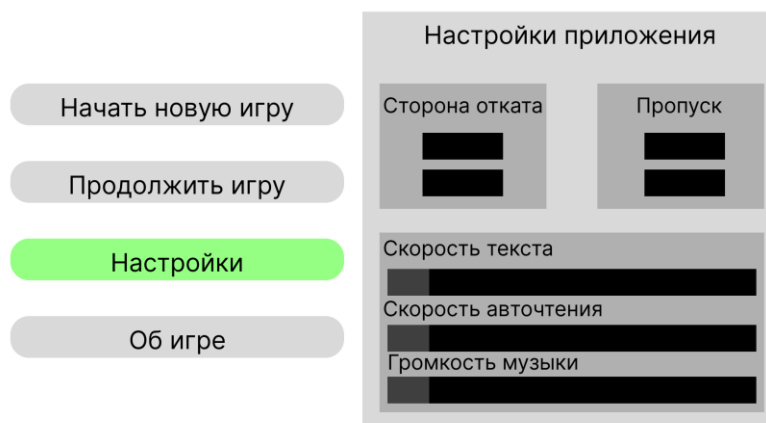


Рисунок 12 – Прототип раздела настроек

В настройках игрок может поменять сторону отката, включить пропуск текста, изменить скорость печати текста и настроить громкость музыки. Также для версии на ПК доступны режимы полного и оконного экрана. Раздел настроек доступен как в главном меню, так и игровом.

Прототип дизайна раздела «об игре» представлен на рисунке 13.

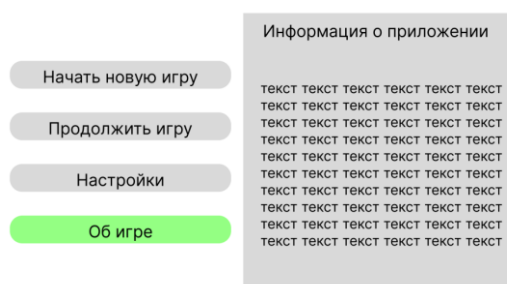


Рисунок 13 – Прототип раздела «об игре»

В этом разделе содержится информация о версии игры, движке на котором она разработана, а также об авторе программы.

Интерфейс игровой партии состоит из 3 основных элементов.

Экран ресурсов игрока, на котором отображается его собранное количество, а также который позволяет менять их количество. В левом верхнем углу показывается номер текущего хода.

Экран планет и звезд игрока, позволяет просматривать колонизированные объекты, а также добавлять новые.

Навигационная панель снизу дает возможность перейти на ход назад или вперед, а также показать инструкцию к игре или магазин построек и подвижной техники. С помощью этой панели или кнопки «назад» на устройстве, можно открыть экран игрового меню.

Прототип дизайна экрана игровой партии представлен на рисунке 14.

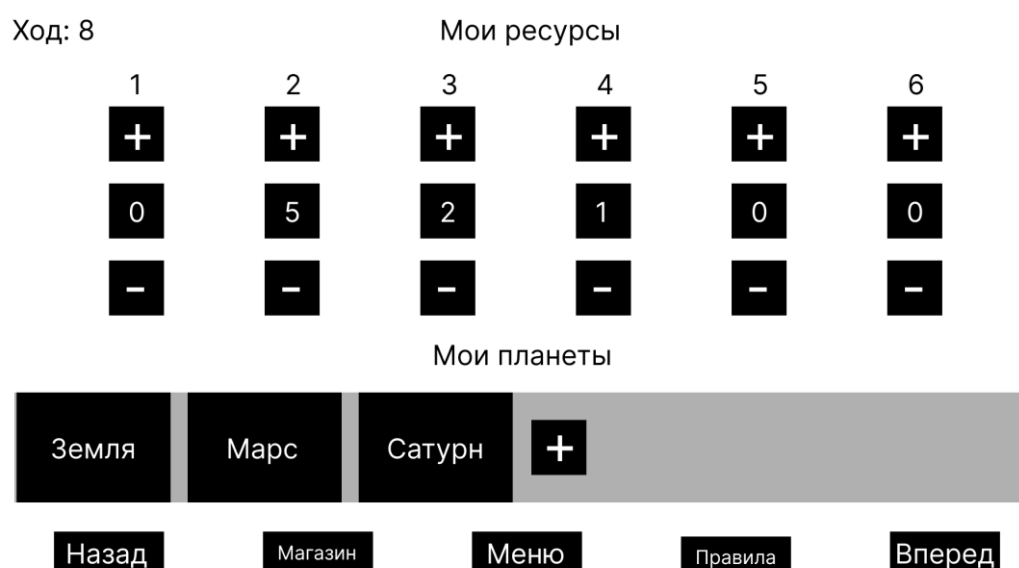


Рисунок 14 – Прототип основного игрового экрана

Интерфейс раздела планеты или звезды схож с интерфейсом основного игрового экрана за исключением нескольких функций.

Экран ресурсов планеты или звезды, на котором отображается его оставшиеся количество, а также который позволяет менять их количество. Вверху показывается название текущего объекта.



Экран удаления планеты или звезды, на котором также можно прочитать описание объекта.

Навигационная панель такая же, как и в основном игровом экране.

Прототип дизайна экрана редактирования планет и звезд представлен на рисунке 15.

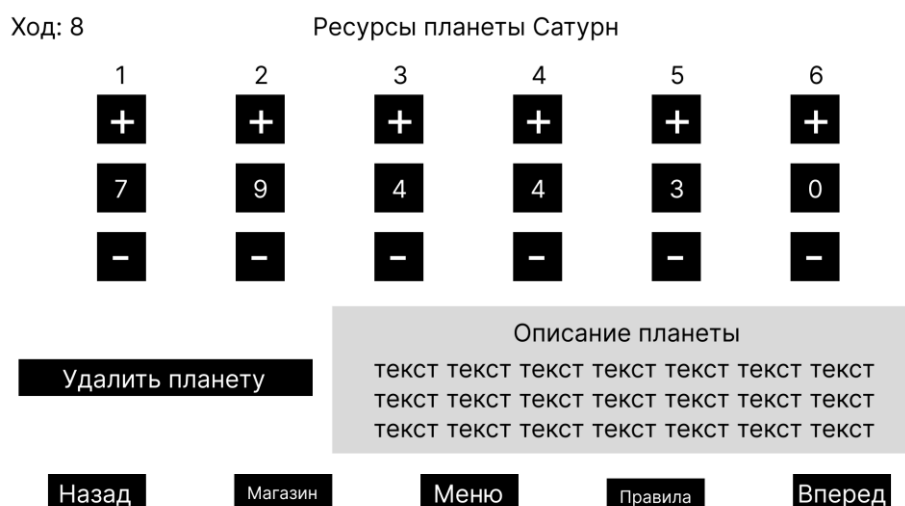


Рисунок 15 – Прототип экрана редактирования планет и звезд

Таким образом, были разработаны основные экраны, используемые в приложении, что значительно упрощает процесс дальнейшей разработки. Эти экраны служат каркасом, на который можно нанизывать функциональные элементы и данные, обеспечивая согласованность интерфейса и улучшая пользовательский опыт.

### 3.3 Разработка программного обеспечения

Фрагмент кода для импортирования необходимых библиотек, а также инициализации переменных и запуска основных экранов имеет вид, представленный на рисунке 16.

```

#нужные модули
init python:
    import random as r
    import copy
    from copy import deepcopy
    import pickle
    import time

#запуск игры
label start:
    stop music fadeout 3
    scene space with fade
    queue music [game_music]
    $ turn = 1
    $ res_1 = 10
    $ res_2 = 10
    $ res_3 = 10
    $ res_4 = 10
    $ res_5 = 0
    $ res_6 = 0
    $ player_planets = []
    $ shop_copy = copy.deepcopy(planets_list)
    show screen display_turn
    show screen players_planets
    while True:
        pause
    return

```

Рисунок 16 – Код, реализующий запуск игрового цикла

Библиотека «random» для генерации случайных чисел, «copy» для полного копирования, «pickle» для хранения данных, а «time» для работы со временем.

При запуске игры каждому типу ресурсов присваиваются стартовые значения, что задает начальные условия для всех игроков. Создается пустой список планет игрока, что позволяет в дальнейшем добавлять новые планеты в ходе игры. Одновременно отображаются экраны для учета ходов и отображения планет игрока, обеспечивая игроку всю необходимую информацию для управления своими ресурсами и стратегиями. После инициализации всех данных запускается бесконечный цикл, который поддерживает непрерывный игровой процесс и реагирует на действия игроков, обеспечивая динамичность и интерактивность игры.

Код, реализующий переход на следующий ход, а также показы экранов, представлен на рисунке 17.

```

#переход на следующий ход
label next_turn:
    $ turn += 1
    if turn % 64 == 0:
        show screen notify_card("Возьмите карточки:\n1) Катаклизм\n2) Случайное
событие в галактике\n3) Случайное событие в системе.")
    elif turn % 16 == 0:
        show screen notify_card("Возьмите карточки:\n1) Случайное событие в
галактике\n2) Случайное событие в системе.")
    elif turn % 4 == 0:
        show screen notify_card("Возьмите карточки:\n1) Случайное событие в
системе.")
    if r.randint(1, 100) > 75:
        $ random_boost_question = r.choice(list(boost_cards_list_test.keys()))
        $ random_boost_answer = boost_cards_list_test[random_boost_question]
        $ random_prize_res = r.choice(res_list)
        $ random_prize_num = r.randint(1, 3)
        show screen boost_card(random_boost_question)
    return

#показ экрана с ответом
label show_screen_with_answer:
    hide screen boost_card
    show screen boost_card_answer(random_boost_answer, random_prize_res,
random_prize_num)
    return

#показать ресурсы планеты
label show_screen_with_planet_res(item):
    hide screen display_turn
    hide screen players_planets
    show screen planet_reses(item)
    return

#назад к ресурсам игрока
label back_to_player_res:
    hide screen planet_reses
    show screen display_turn
    show screen players_planets
    return

```

Рисунок 17 – Код, реализующий переход на следующий ход, а также показы экранов

При переходе на следующий ход, происходит проверка на выпадение карточек случайных событий, а также с шансом 25% выпадение карточки с вопросом. Карточка с вопросом берется случайная.

Показы экранов и скрывтия экранов в зависимости, где сейчас находится игрок.

Код, реализующий экран ресурсов планеты или звезды, представлен на рисунке 18.

```

#экран ресурсов планеты
screen planet_rese(item):
    frame:
        xsize 1800
        ysize 400
        xanchor -60
        yanchor -600

        textbutton ("Вернуться") action Call("back_to_player_res") xalign 0.1
yalign 0.33
        textbutton ("Удалить планету") action
[Function(remove_planet_from_player, item), Call("back_to_player_res")] xalign
0.1 yalign 0.66
        text "[item[описание]]" xalign 0.9 yalign 0.33 xysize (600, 400)

text "Текущий ход - [turn]"
text "Ресурсы [item[название]]" xalign 0.5 yalign 0.05
text "[item[res1]]" xalign 0.1 yalign 0.3
image "gui/l_res.jpg" xalign 0.1 yalign 0.1
imagebutton xysize(64, 64):
    xalign 0.1
    yalign 0.2
    idle "gui/plus.png"
    hover "gui/plus_hover.png"
    action Call('change_res', item, True, 1)
imagebutton xysize(64, 64):
    xalign 0.1
    yalign 0.4
    idle "gui/minus.png"
    hover "gui/minus_hover.png"
    action Call('change_res', item, False, 1)

```

Рисунок 18 – Неполный код, реализующий экран ресурсов планеты или звезды

Экран состоит из основного кадра размера 1800 на 400 пикселей с отступом -60 по x и -600 по y. В кадре расположены текстовые кнопки, а также название и описание объекта.

Кнопки обновления ресурсов сделаны с помощью картинок 64 на 64 пикселя. Имеет светлую версию картинки при наводке курсором. При нажатии вызывают функцию изменения ресурсов со своими параметрами.

Показы экранов и скрывания экранов в зависимости, где сейчас находится игрок.

Код, реализующий прокручиваемый экран всех планет и звезд игрока, представлен на рисунке 19.

```

#показ планет и звезд игрока
screen players_planets:
    frame:
        xsize 1800
        ysize 200
        xanchor -60
        yanchor -600

        vpgrid:
            cols 20
            draggable True
            mousewheel True
            scrollbars "horizontal"
            xysize (1800, 200)

            $ j = 0

            for item in player_planets:
                imagebutton:
                    idle (item["image"])
                    xysize (200, 200)
                    action Call("show_screen_with_planet_res", item)
                #text "[item[название]]" xysize (200, 200)
            $ j += 1

            for i in range(20-j):
                imagebutton:
                    idle ("gui/no_item_buttn.png")
                    xysize (200, 200)
                    action Show("add planet to player")

```

Рисунок 19 – Код, реализующий прокручиваемый экран всех планет и звезд игрока

Экран приложения состоит из 20 колонок размером 200 на 200 пикселей, обеспечивая возможность горизонтальной прокрутки.

Алгоритм начинает с обработки списка планет игрока, добавляя соответствующие изображения этих планет в ячейки экрана. После того как все планеты игрока размещены, оставшиеся пустые клетки заполняются специальным изображением, которое служит кнопкой для добавления новой планеты или звезды в коллекцию игрока. При нажатии на это изображение пользователи могут легко расширить свою коллекцию, добавляя новые небесные тела.

Код, реализующий экран добавления планеты или звезды игроку, представлен на рисунке 20.

```

# Экран для добавления планеты игроку
screen add_planet_to_player:
    modal True
    frame:
        xsize 800
        ysize 1040
        xalign 0.5
        yalign 0.5
        textbutton _("Отмена") action Hide("add_planet_to_player") xalign 1.0
yalign 1.0
vbox:
    for planet_name, planet_info in shop_copy.items():
        if planet_info not in player_planets:
            textbutton planet_name:
                action Function(add_planet_to_player, planet_info)

```

Рисунок 20 – Код, реализующий экран добавления планеты или звезды игроку

Экран приложения представляет собой кадр размером 800 на 1040 пикселей, расположенный строго по центру экрана устройства, обеспечивая удобство взаимодействия для пользователей. Этот кадр является основным рабочим пространством, в котором отображаются все элементы интерфейса и игровые данные. В зависимости от текущего состояния игры и уже имеющихся объектов, приложение автоматически генерирует и обновляет список доступных для добавления звезд или планет, предоставляя пользователю актуальные опции для дальнейшего игрового процесса. Это позволяет динамично реагировать на изменения в игре и обеспечивать интуитивное управление игровыми элементами.

Код, реализующий функции добавления и изъятия планеты или звезды у игрока, представлен на рисунке 21.

```

# функция для добавления планеты игроку
init python:
    def add_planet_to_player(planet_info):
        global shop_copy
        shop_copy = copy.deepcopy(planets_list)
        if planet_info not in player_planets:
            player_planets.append(planet_info)
            # Скрываем экран добавления планеты
            renpy.hide_screen("add_planet_to_player")

    def remove_planet_from_player(item):
        if item in player_planets:
            player_planets.remove(item)

```

Рисунок 21 – Код, реализующий функции добавления и изъятия планеты или звезды у игрока

Функции проверяют наличие или отсутствие планеты в списке, а после этого производят свои операции.

Код, реализующий экран карточки со случайным вопросом, представлен на рисунке 22.

```
#экран с случайным вопросом
screen boost_card(random_boost_question):
    modal True
    zorder 200
    style_prefix "confirm"
    add "gui/overlay/confirm.png"
    frame:
        vbox:
            xalign .5
            yalign .5
            spacing 45

            label _("[random_boost_question]"):
                style "confirm_prompt"
                xalign 0.5

            hbox:
                xalign 0.5
                spacing 150

                textbutton _("Показать ответ") action
                Call("show_screen_with_answer")

        key "game_menu" action Hide("boost_card")
```

Рисунок 22 – Код, реализующий экран карточки со случайным вопросом

Модальный экран не позволяет переключаться на другие экраны. Добавляет пустой кадр и в нем показывает случайный вопрос.

### 3.4 Интеграция обучающего элемента

Разработка мобильного приложения помощника для игры представляется важным шагом в автоматизации и улучшении игрового процесса. Приложение не только облегчит управление ресурсами и выполнением ходов, но и обеспечит удобный интерфейс для ознакомления с игровыми элементами и правилами. Обучающим элементом, помимо самой игры, является наличие описания карт реально существующих планет и звезд, а также карточек с вопросами по

астрономии, физике и математике, что позволяет игрокам получать знания в этих областях в процессе игры. Учитывая требования к функциональности и архитектуре, данное приложение сможет эффективно поддерживать игроков, обеспечивая удобство и гибкость в использовании на различных устройствах и операционных системах.

Код, реализующий хранение звезд и планет с их описанием в виде словарей, представлен на рисунке 23.

```
define planets_list = {
  "Kepler-186f":{
    "название":"Kepler-186f",
    "описание":"Первая экзопланета, находящаяся в зоне обитаемости звезды,
    похожей на Солнце. Размер: 1,1 раза больше Земли. Температура поверхности:
    предположительно, находится в диапазоне, пригодном для жидкой воды. Расстояние
    от Земли: 500 световых лет.",
    "res1":21,
    "res2":20,
    "res3":21,
    "res4":18,
    "res5":38,
    "res6":0,
    "image":"gui/planet1.png"},
  "Проксима Центавра b":{
    "название":"Проксима Центавра b",
    "описание":"Эта планета находится в зоне обитаемости вокруг звезды Проксима
    Центавра, наиболее близкой к Солнечной системе. Она является объектом
    постоянного научного интереса в поисках жизни за пределами Земли.",
    "res1":21,
    "res2":21,
    "res3":20,
    "res4":17,
    "res5":44,
    "res6":0,
    "image":"gui/planet2.png"},
}
```

Рисунок 23 – Неполный код, реализующий хранение звезд и планет с их описанием в виде словарей

В словаре хранится название объекта, его описание, ресурсы, а также изображение.

Карточки с вопросами по смежным с космосом наукам помогут детям легче запоминать различные факты, формулы, а также научат их логически рассуждать.



Мотивацией для правильного ответа на вопрос помимо знаний являются ресурсы, которые может получить игрок. Они генерируются случайным образом и могут отличаться друг от друга. Вопросы и варианты ответа на него также хранятся в словаре.

Код, реализующий хранение вопросов и ответов в виде словаря, представлен на рисунке 24.

```
define boost_cards_list_test = {""Астрономия – наука, изучающая ...  
1) движение и происхождение небесных тел и их систем  
2) развитие небесных тел и их природу  
3) движение, природу, происхождение и развитие небесных тел и их систем"":"3)  
движение, природу, происхождение и развитие небесных тел и их систем",  
""Эклиптика – это ...  
1) зодиакальный пояс созвездий  
2) годичный путь Солнца по небесной сфере  
3) линия, вдоль которой движется Луна  
4) траектория движения планеты"":"2) годичный путь Солнца по небесной сфере",
```

Рисунок 24 – Неполный код, реализующий хранение вопросов и ответов в виде словаря

С помощью этих элементов, а также увлекательной игры можно заинтересовать детей изучением космоса.

### 3.5 Создание опроса для приложения

Для оценки запоминания знаний игроков и сбора обратной связи для улучшения игры используется специальная форма в Google Forms.

Google Forms – бесплатный сервис для создания тестов, проведения опросов и исследований, сбора отзывов о компании и продукте, сбора регистраций на мероприятия, а также поиска сотрудников [20].

Сервис прост в использовании, использует облачное хранилище, благодаря чему можно к нему можно получить доступ с любого устройства.

Имеет функции индивидуального оформления и удобного показа статистики по ответам.

Вид тестовых заданий и вопросов по мнению об игре представлен на рисунках 25 – 26.

Рисунок 25 – Вопросы на проверку знаний

Рисунок 26 – Вопросы для улучшения качества игры

Опросы в Google Forms являются мощным инструментом для проверки знаний игроков и сбора отзывов и пожеланий об настольной игре. Это поможет получить ценные сведения о том, какие аспекты игры нравятся игрокам, а какие нуждаются в улучшении. Использование Google Forms для сбора этой информации является эффективным и доступным способом получения ценных данных, которые могут помочь в совершенствовании игры и создании более

увлекательного игрового опыта для игроков. Сама форма не занимает много времени у игроков и не является обязательной.

Опрос будет доступен по qr коду вместе с настольной игрой, а также в качестве ссылки на форму после окончания партии в приложении. Опрос является анонимным и не заставляет игроков указывать свои персональные данные.

### 3.6 Готовый интерфейс приложения

На рисунках 27 – 30 представлен готовый интерфейс приложения для пользователя, демонстрирующий все ключевые элементы и экраны, которые игрок будет видеть при взаимодействии с игрой. Эти иллюстрации помогают визуализировать конечный продукт и дают представление о пользовательском опыте.

Вид главного меню игры на ПК представлен на рисунке 27.

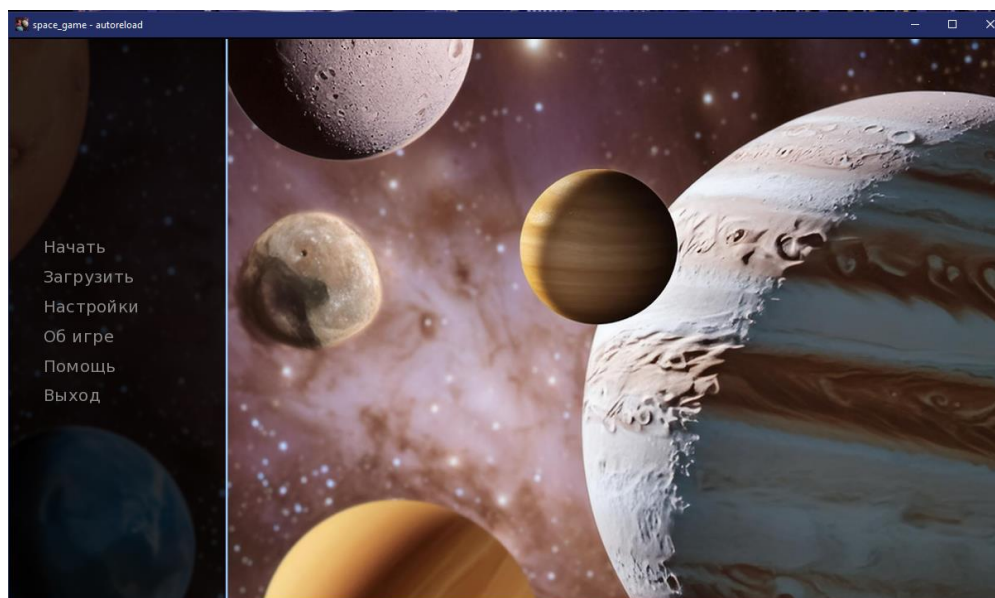


Рисунок 27 – Итоговое главное меню игры

Главный экран оформлен в стилистике игры, с использованием соответствующей цветовой гаммы и графики, что создает атмосферу и

настраивает пользователя на предстоящий игровой процесс. Навигация по главному меню интуитивно понятна и удобна, что позволяет пользователям легко ориентироваться и быстро находить нужные опции.

Обои для главного меню были созданы бесплатной нейросетью для генерации изображений – Adobe Firefly.

Вид игрового меню и раздела настроек на ПК представлен на рисунке 28.

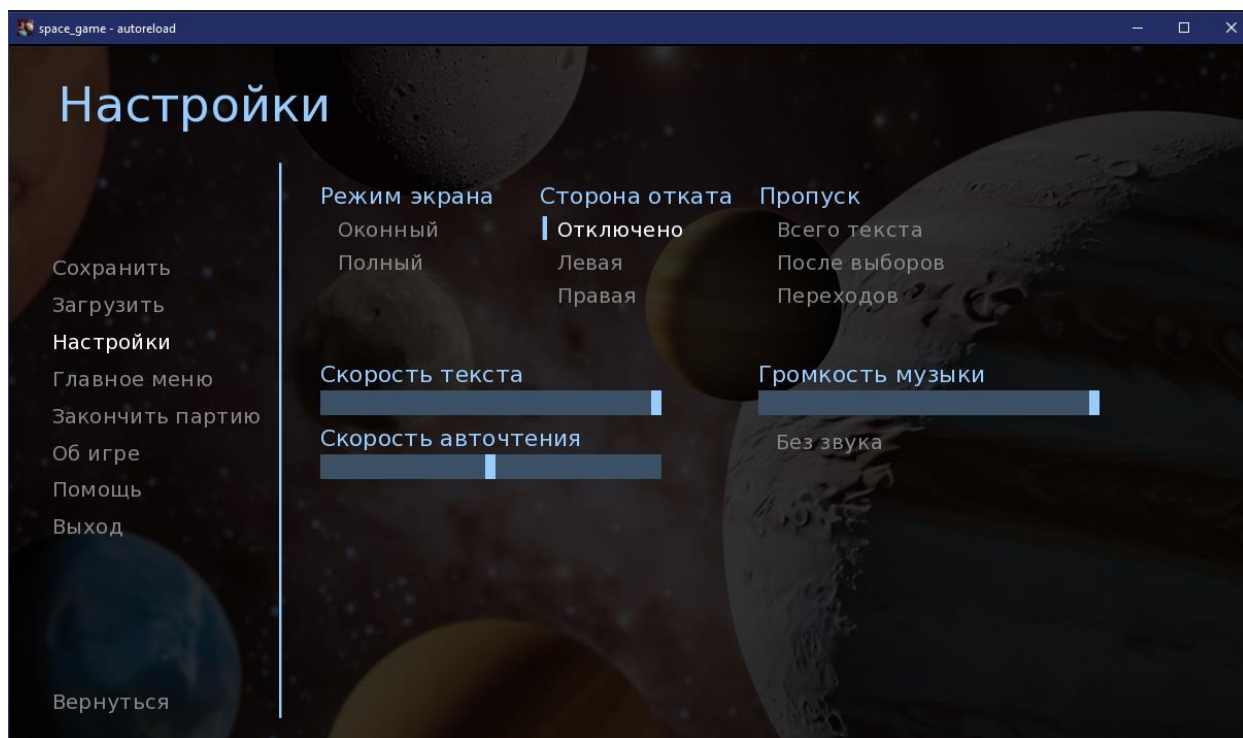


Рисунок 28 – Итоговое игровое меню и раздел настроек

Игровое меню вызывается во время партии и отличается от главного меню своим функционалом. Оно позволяет игрокам сохранять текущее состояние игры, загружать ранее сохраненные состояния и завершать текущую игровую сессию. Это меню предоставляет удобный доступ к основным функциям управления игровым процессом, что особенно важно в интерактивных играх, где сохранение прогресса и возврат к предыдущим точкам принятия решений являются ключевыми элементами игрового опыта.

В разделе настроек игрового меню пользователи могут настроить параметры, чтобы оптимизировать игровой процесс под свои предпочтения.

Здесь можно выбрать режим экрана, переключаясь между полноэкранным режимом и оконным отображением. Также доступна настройка стороны отката, позволяющая игрокам выбрать, насколько далеко назад они могут отмотать игровой процесс, чтобы пересмотреть или изменить принятые решения. Функция пропуска текста позволяет ускорить прохождение уже прочитанных или повторяющихся сцен, что особенно полезно при повторных прохождениях игры. Включить или отключить музыку.

Вид основного экрана игры на ПК представлен на рисунке 29.

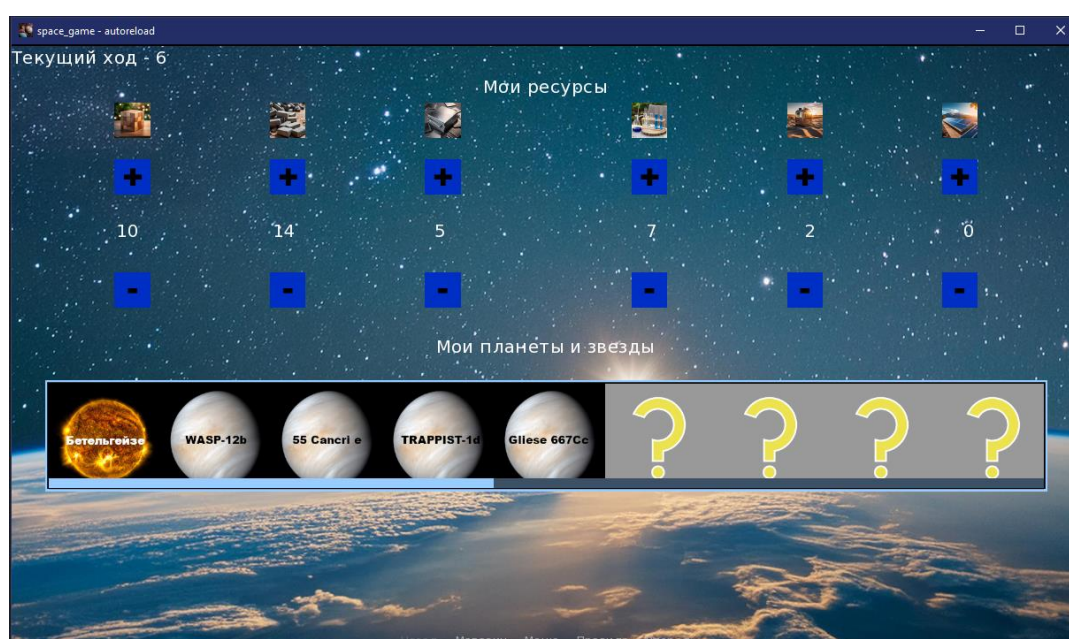


Рисунок 29 – Итоговый экран ресурсов и планет игрока

В верхней части экрана, по центру, размещены ресурсы игрока. Здесь можно увидеть количество доступных ресурсов, таких как древесина, металл, топливо и другие. Рядом с каждым типом ресурса находятся соответствующие иконки для наглядности. Игрок имеет возможность изменять количество своих ресурсов с помощью кнопок, расположенных рядом с иконками, что позволяет легко управлять своими запасами и принимать обоснованные решения в процессе игры. Слева показывается номер текущего хода для удобства.



Чуть ниже ресурсов расположен список планет и звезд, принадлежащих игроку. В самом низу экрана расположены кнопки управления и навигации, которые обеспечивают доступ к различным функциям игры. Кнопка «назад» позволяет вернуться к предыдущему экрану или этапу игры, а кнопка «вперед» используется для перехода к следующему этапу. Кнопка «магазин» открывает экран магазина, где игрок может прочитать описание и цену построек и техники, а кнопка «правила» предоставляет доступ к игровым правилам и инструкциям. Кнопка «меню» открывает игровое меню.

Вид экрана ресурсов построек представлен на рисунке 30.

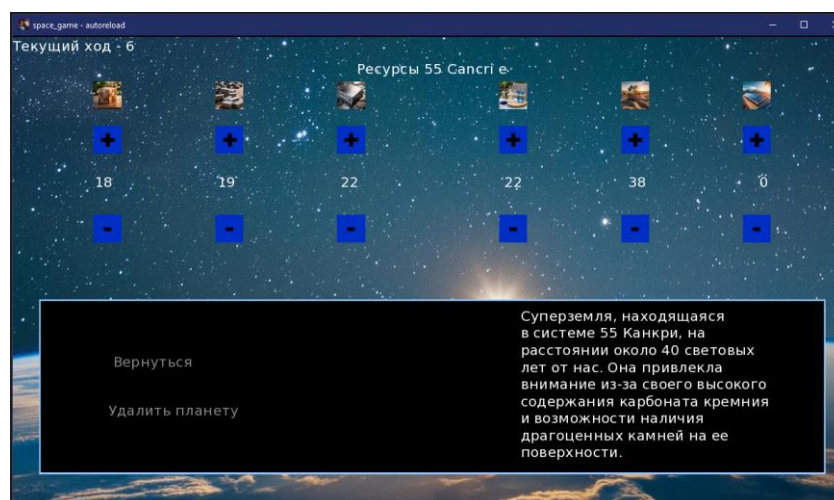


Рисунок 30 – Итоговый экран ресурсов планеты игрока

Экран ресурсов планеты схож с экраном ресурсов игрока, но предназначен для управления ресурсами конкретной планеты. В верхней части экрана указано название планеты, что позволяет игроку легко идентифицировать текущую планету. Чуть ниже расположено краткое описание планеты.

В нижней части экрана расположены кнопки управления, которые предоставляют дополнительные функции. Игрок может вернуться к экрану ресурсов игрока с помощью соответствующей кнопки, что позволяет быстро переключаться между общим управлением и управлением конкретными

планетами. Также предусмотрена возможность удаления выбранной планеты или звезды, что позволяет игроку оптимизировать свои владения и сконцентрироваться на наиболее перспективных объектах.

Интерфейс приложения получился интуитивно понятным и удобным для пользователя. Приятным и легко управляемым как на ПК, так и на Android.

### 3.7 Создание дизайна настольной игры

Внешний вид карточек планет представлен на рисунке 31.

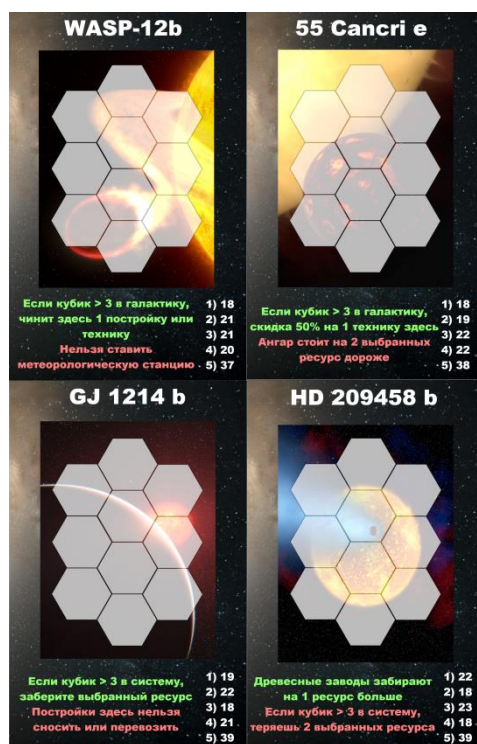


Рисунок 31 – Итоговый внешний вид карточек планет

В дизайне карточек использованы бесплатные художественные изображения планет и звезд, максимально приближенные к их реальным прототипам. Эти изображения были тщательно подобраны, чтобы обеспечить визуальное соответствие известных небесных объектов, таких как GJ 1214 b, HD 40307g, TOI-700d, а также различных типов звезд, включая красные гиганты и белые карлики.

В дизайне карточек планеты эффекты выделены зелёным и красным, чтобы подчеркнуть характер свойства, справа белым указано стартовое количество ресурсов объекта, а наверху карточки приведено название, что помогает игрокам легко идентифицировать планеты, оценивать их потенциал и стратегически планировать свои действия в игре.

Внешний вид карточек случайных событий в звездной системе, случайных событий в галактике и катаклизмов представлен на рисунке 32.

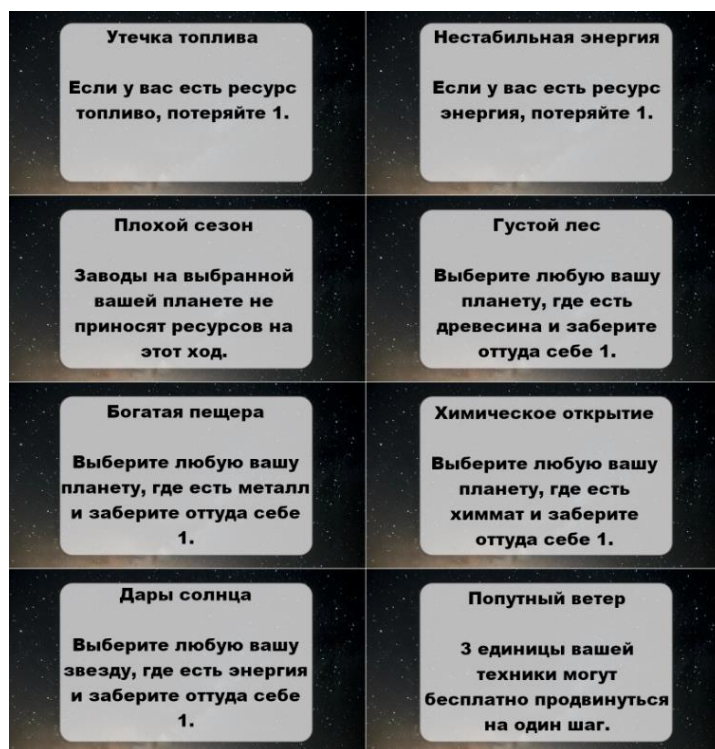


Рисунок 32 – Итоговый внешний вид карточек случайных событий

В дизайне карточек случайных событий особое внимание уделено читаемости и ясности информации. Крупный шрифт на белом фоне позволяет игрокам легко и быстро воспринимать название события и его действие без лишних усилий. Этот подход обеспечивает выделение важной информации и улучшает игровой опыт, особенно во время быстрых решений во время игры.

С обратной стороны каждой карточки находится иконка, отображающая тип события. Это позволяет игрокам быстро определить категорию события и



легко отличать одни карточки от других, даже без необходимости поворачивать их.

### 3.8 Создание физической копии игры

Карточки планет и случайных событий печатались на специальной фотобумаге, жетоны построек и техники печатались на обычной бумаге.

Распечатанный вариант карточек и жетонов представлен на рисунке 33.



Рисунок 33 – Распечатанные карточки и жетоны

Карточки планет и звезд были спроектированы таким образом, чтобы быть удобными в использовании во время игры. Они были распечатаны на четвертях листа формата А4, что обеспечило достаточное пространство для удобного размещения на них жетонов.

Сами жетоны были распечатаны на обычной черно-белой бумаге, что позволило снизить затраты на. Несмотря на отсутствие цвета, жетоны остаются четкими и легко различимыми благодаря ярко выраженным символам.

Карточки случайных событий, в свою очередь, были распечатаны на фотобумаге, обеспечивая высокое качество изображений и яркость цветов. Их компактный размер делает их удобными для перемещения и сортировки во время игры, а также обеспечивает легкость хранения и транспортировки.

### **3.9 Тестирование настольной игры и приложения**

Для тестирования игры и мобильного приложения на целевой аудитории была выбрана МБУ Школа №84 в городе Тольятти. Этот выбор был обоснован желанием получить обратную связь от учащихся, соответствующих возрастной группе и интересам целевой аудитории проекта. Был предоставлен 5 класс, то есть дети 10 – 12 лет. Проведение тестирования началось с организации двух уроков с участием школьников. На этих уроках им были объяснены правила игры и основные механики приложения. После этого дети начали игровую партию.

Школьники быстро освоились с игрой и погрузились в её процесс. Их быстрое понимание правил и активное участие в игре свидетельствуют о том, что игра была проста и увлекательна, что важно для привлечения внимания целевой аудитории. Участники тестирования проявили высокий уровень интереса к игре, задавали вопросы, а также отвечали на них. Высказывали свои мысли и предложения по улучшению проекта. Эта обратная связь послужит ценным материалом для доработки игры и мобильного приложения, чтобы сделать их ещё более привлекательными и увлекательными для целевой аудитории.

Фотография с проведения тестирования настольной игры и приложения в школе на рисунке 34.



Рисунок 34 – Дети играют в настольную космическую игру

Карточки случайных событий были распечатаны на фотобумаге, что обеспечило высокое качество изображений и яркость цветов, делающие их визуально привлекательными и легкими для восприятия. Компактный размер карточек делает их удобными для перемещения и сортировки во время игры, позволяя игрокам быстро и легко управлять игровыми элементами. Кроме того, такие размеры обеспечивают легкость хранения и транспортировки карточек, что особенно важно для игроков, которые часто берут игру с собой. Благодаря этим характеристикам, карточки не только улучшают игровой процесс, но и делают его более комфортным и организованным.

### **3.10 Результаты проведения тестирования**

После окончания игровой сессии участникам было предложено пройти опрос, целью которого было оценить усвоение знаний и собрать отзывы об игре. Результаты тестирования показали, что дети успешно усвоили материал, представленный в игре.

Большинство участников правильно ответили на все заданные вопросы, что свидетельствует о высокой информативности и образовательной ценности игры. Такие результаты подчеркивают эффективность игровой формы обучения и её способность передавать знания в увлекательной и понятной для детей форме.

Данный результат подтверждает, что игра эффективно способствует усвоению знаний о космосе в игровой форме, делая процесс обучения увлекательным и продуктивным. Полученные отзывы также указывают на высокий уровень удовлетворенности игроков, что подчеркивает удачное сочетание образовательных и развлекательных элементов в обучающей настольной игре.

Проценты ответов игроков приведены на рисунке 35.

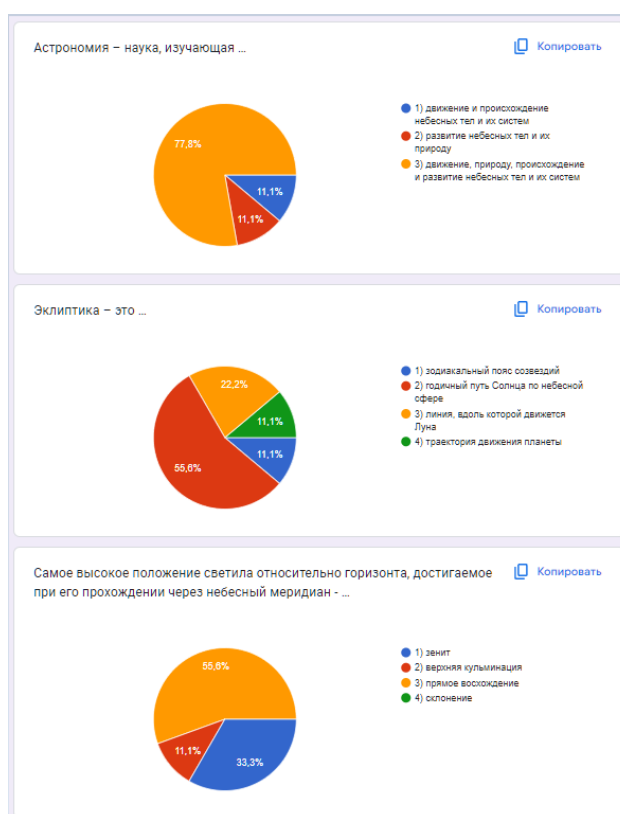


Рисунок 35 – Графики ответов игроков на вопросы по астрономии

Кроме того, игра получила положительные отзывы от всех участников тестирования. Все игроки высоко оценили процесс игры и визуальное оформление, отметив его привлекательность и интересность. Это подтверждает успешное сочетание образовательного и развлекательного аспектов игры, что делает её привлекательной для широкой аудитории.

В качестве пожеланий к игре самыми популярными были – добавить больше планет и звезд для расширения игрового мира и повышения его разнообразия. Также было высказано пожелание о создании версии игры для устройств iOS, что позволило бы расширить доступность игры для более широкой аудитории пользователей мобильных устройств.

Эти отзывы и пожелания игроков будут учтены при дальнейшей разработке игры, с целью её усовершенствования и создания наиболее удовлетворительного игрового опыта для всех пользователей.

## Заключение

Выпускная квалификационная работа посвящена разработке мобильного приложения помощника для настольной обучающей игры и выполнена на основе материалов преддипломной практики, проведенной в ООО «АльВиРити», которое является резидентом технопарка в сфере высоких технологий «Жигулевская долина».

Целью данной работы было создание функционального мобильного приложения, которое автоматизирует и упрощает процесс игры, а также позволяет собирать данные о впечатлениях и знаниях игроков. Оно должно было обеспечить удобный и простой интерфейс для взаимодействия с игрой, автоматизировать рутинные задачи, связанные с ведением счета и отслеживанием хода игры, а также предоставить дополнительные функции, которые улучшат игровой процесс и сделают его более увлекательным.

В результате проведенной работы были решены следующие задачи:

- изучено влияние настольных игр на обучение детей;
- выявлены проблемы современного рынка настольных игр, а также предложены способы их решения;
- спроектирована логическая модель настольной игры;
- спроектирована логическая модель программного обеспечения;
- разработан прототип интерфейса мобильного приложения;
- разработано программное обеспечение для Windows и Android;
- интегрирован обучающий элемент в приложение;
- создан опрос для проверки знаний, а также сбора отзывов и пожеланий для игры;
- разработан готовый интерфейс для приложения;
- разработан готовый дизайн для настольной игры;
- создана физическая копия настольной игры;

- проведено тестирование приложения и настольной игры у целевой аудитории;
- проведен анализ результатов тестирования игры.

Итогом бакалаврской работы стала разработка мобильного приложения-помощника для настольной обучающей игры. Приложение было создано с целью дополнения и улучшения игрового процесса настольной игры, а также для повышения уровня вовлеченности и интереса игроков.

Разработанное приложение и игра прошли успешное тестирование в МБОУ «Школа №84» Имени Александра Невского. Приложение получило положительные отзывы от игроков. Оно было признано эффективным инструментом, который делает игровой процесс более увлекательным и динамичным, а также способствует лучшему усвоению игрового материала и повышению общего уровня знаний игроков в области космоса.

Таким образом, выпускная квалификационная работа успешно достигла своей цели и предоставила полезный инструмент для улучшения опыта игры в настольные обучающие игры.

## Список используемой литературы и используемых источников

1. Баланов, А. Комплексное руководство по разработке. От мобильных приложений до веб-технологий / А. Баланов. – М.: Лань, 2024. – 238с.
2. Березовская Ю.В. Введение в разработку приложений для ОС Android. Учебное пособие / Ю.В. Березовская — Москва: Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2021. — 427 с. — ISBN 978-5-4497-0890-8. — Текст: электронный // Цифровой образовательный ресурс IPR SMART: [Электронный ресурс]. — URL: <https://www.iprbookshop.ru/102000.html>
3. Буйначев С.К. Основы программирования на языке Python. Учебное пособие / Буйначев С.К. — Екатеринбург : Уральский федеральный университет, ЭБС АСВ, 2014. — 92 с. — ISBN 978-5-7996-1198-9. — Текст: электронный // Цифровой образовательный ресурс IPR SMART: [Электронный ресурс]. — URL: <https://www.iprbookshop.ru/66183.html>
4. Буч, Гради Введение в UML от создателей языка / Гради Буч, Джеймс Рамбо, Ивар Якобсон. - М.: ДМК Пресс, 2015. - 496 с.
5. Введение в гибридные технологии разработки мобильных приложений. Учебное пособие / Н.П. Павлович, А.М. Зайцев. – М.: Лань, 2022. – 160 с.
6. ГОСТ 34.601-90. Информационная технология. Комплекс стандартов на автоматизированные системы.
7. ГОСТ Р ИСО/МЭК 12119-2000 - Информационная технология. Пакеты программ. Требования к качеству и тестирование.
8. Колисниченко, Д. Н. Программирование для Android. Самоучитель / Д. Колисниченко, Е. Рогов. – М.: БХВ, 2021. – 288 с.
9. Лучано, Р. Python. К вершинам мастерства – ДМК Пресс, 2016. – 768с.
10. Мартелли, А. Python. Справочник. Полное описание языка / А. Мартелли. — М: Диалектика, 2019. — 896 с.
11. Фаулер, Д. UML. Издание на русском языке / Д. Фаулер – М.: Бином-Пресс, 2018. – 672 с.



12. Филлипс, Б. Android. Программирование для профессионалов / Филлипс Б. — СПб: Питер, 2021. — 704 с.
13. Фрайман, З. Создание приложений для смартфонов и планшетов под ОС Android. Практический курс / З. Фрайман. — М: Едиториал УРСС, 2020. — 504 с.
14. Уилкс, М. Профессиональная разработка на Python / М. Уилкс. - М.: ДМК Пресс, 2021. - 502 с.
15. Хеллман, Д. Стандартная библиотека Python 3. Справочник с примерами / Д. Хеллман. - М.: Диалектика, 2020. - 1376 с.
16. Шелудько В.М. Основы программирования на языке высокого уровня Python. Учебное пособие / Шелудько В.М. — Ростов-на-Дону, Таганрог : Издательство Южного федерального университета, 2017. — 146 с. — ISBN 978-5-9275-2649-9. — Текст: электронный // Цифровой образовательный ресурс IPR SMART: [Электронный ресурс]. — URL: <https://www.iprbookshop.ru/87461.html>
17. Сьюзан, Давид Келлер UML 2 для начинающих. 3-е изд. Библиотека программиста / Сьюзан Келлер, Давид Келлер. – М.: ДМК Пресс, 2010. – 382 с.
18. Camilus Raynaldo, Android UI Design with XML / С. Raynaldo. – Webucator, 2017. - 232 с.
19. Develop for Android [Электронный ресурс]. URL: <https://developer.android.com/develop> (дата обращения 16.03.2024)
20. Herv Guihot. Pro Android Apps Performance Optimization / Н. Guihot. – Apress, 2022. - 194 с.
21. The Python Tutorial [Электронный ресурс]. URL: <https://docs.python.org/3/tutorial/index.html> (дата обращения 10.04.2024)
22. Welcome to Ren'Py's documentation! [Электронный ресурс]. URL: <https://www.renpy.org/doc/html/index.html> (дата обращения 10.04.2024)