|  |  |  |
| --- | --- | --- |
|  |  |  |
| МИНОБРНАУКИ РОССИИ | | |
| Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | |

Институт Информационных технологий

Кафедра Математического обеспечения и стандартизации информационных технологий

**Отчет по практическим работам №5-8**

по дисциплине «Системная и программная инженерия»

|  |  |
| --- | --- |
| **Выполнили:**  Студенты группыИКБО-13-19 | Изотина А. А.  Малютина В. А.  Назаркин В. И.  Тузов И. В. |
| **Проверил:** | доцент Туманова М.Б. |

МОСКВА 2022 г.

**Содержание**

[1 Практическая работа №5 4](#_Toc100037551)

[1.1 Ход работы: 4](#_Toc100037552)

[1.1.1 Структурные диаграммы 4](#_Toc100037553)

[1.1.2 Нотация BPMN 7](#_Toc100037554)

[1.2 Вывод: 9](#_Toc100037555)

[2 Практическая работа №6 10](#_Toc100037556)

[2.1 Ход работы: 10](#_Toc100037557)

[2.1.1 Диаграмма DFD 10](#_Toc100037558)

[2.1.2 Хранение данных 12](#_Toc100037559)

[2.2 Вывод: 12](#_Toc100037560)

[3 Практическая работа №7 13](#_Toc100037561)

[3.1 Ход работы: 13](#_Toc100037562)

[3.1.1 Концептуальная архитектура 13](#_Toc100037563)

[3.2 Вывод: 26](#_Toc100037564)

[4 Практическая работа №8 27](#_Toc100037565)

[4.1 Ход работы: 27](#_Toc100037566)

[4.1.1 Введение 27](#_Toc100037567)

[4.1.2 Основание для разработки 27](#_Toc100037568)

[4.1.3 Назначение разработки 27](#_Toc100037569)

[4.1.4 Требования к программе или программному изделию 28](#_Toc100037570)

[4.1.4.1 Требования к функциональным характеристикам 28](#_Toc100037571)

[4.1.4.2 Требования к надежности 28](#_Toc100037572)

[4.1.4.3 Условия эксплуатации 29](#_Toc100037573)

[4.1.4.4 Требования к составу и параметрам технических средств 29](#_Toc100037574)

[4.1.4.5 Требования к производительности 30](#_Toc100037575)

[4.1.4.6 Требования к информационной и программной совместимости 30](#_Toc100037576)

[4.1.4.7 Требования к маркировке и упаковке 30](#_Toc100037577)

[4.1.4.8 Требования к транспортированию и хранению 30](#_Toc100037578)

[4.1.4.9 Требования к доступности и локализации 30](#_Toc100037579)

[4.1.5 Требования к программной документации 31](#_Toc100037580)

[4.1.6 Стадии и этапы разработки 31](#_Toc100037581)

[4.1.7 Порядок контроля и приемки 33](#_Toc100037582)

[4.2 Вывод: 33](#_Toc100037583)

[5 Заключение 34](#_Toc100037584)

# Практическая работа №5

**Цель:** получить навыки по анализу бизнес-процессов, при помощи их моделирования в нотациях BPMN и IDEF0. А также навыки по проектированию структуры проекта посредством построения диаграммы классов и диаграммы объектов в нотации UML.

## Ход работы:

### Структурные диаграммы

Для демонстрации общей структуры иерархии классов системы, их кооперации, атрибутов, методов, интерфейсов и взаимосвязей между ними была построена структурная диаграмма классов. Диаграмма представлена на Рисунке 1.1, Рисунке 1.2 и Рисунке 1.3.

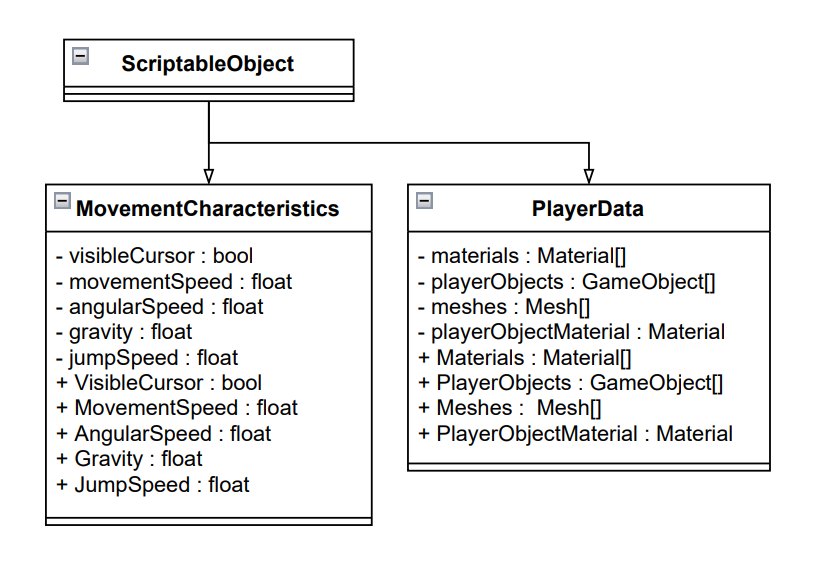


Рисунок 1.1 - Временное хранение данных

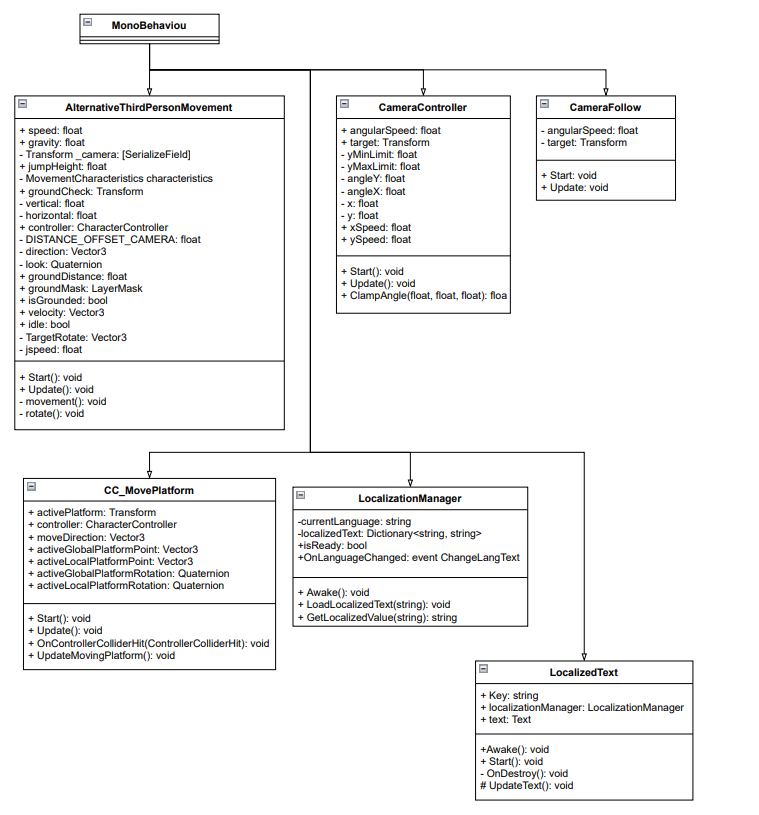


Рисунок 1.2 – MonoBehaviou начало

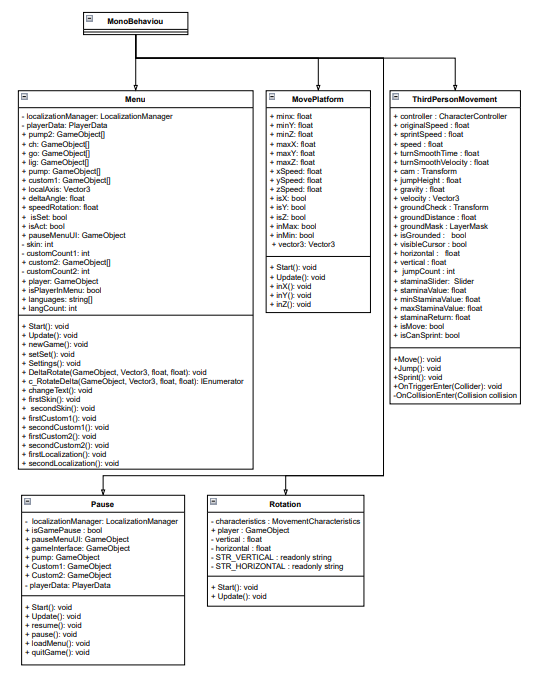


Рисунок 1.3 – MonoBehaviou продолжение

Далее выполнено построение диаграммы объектов для демонстрации совокупности моделируемых объектов и связей между ними в фиксированный момент времени. Диаграмма представлена на Рисунке 1.4. На ней отражена структура объектов игрового меню.

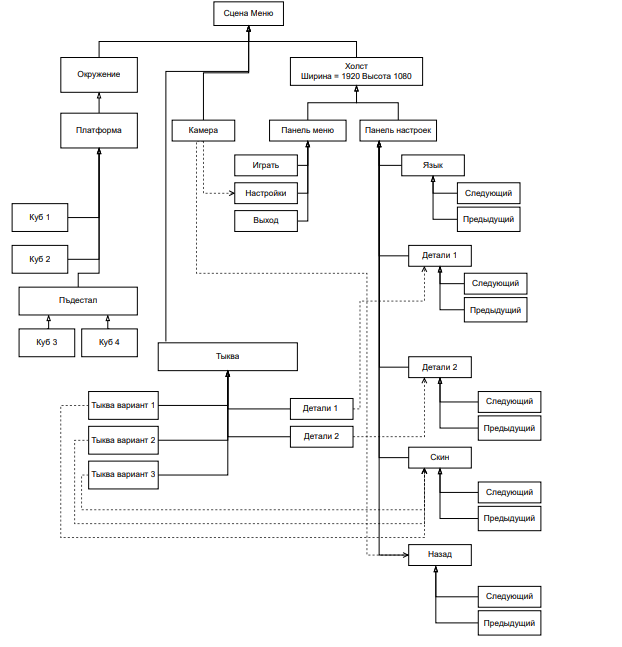


Рисунок 1.4 – Диаграмма объектов Сцены «Меню»

### Нотация BPMN

Была построена диаграмма бизнес-процесса «Jumpkin» В нотации BPMN. Диаграмма представлена на Рисунке 1.5 и Рисунке 1.6.

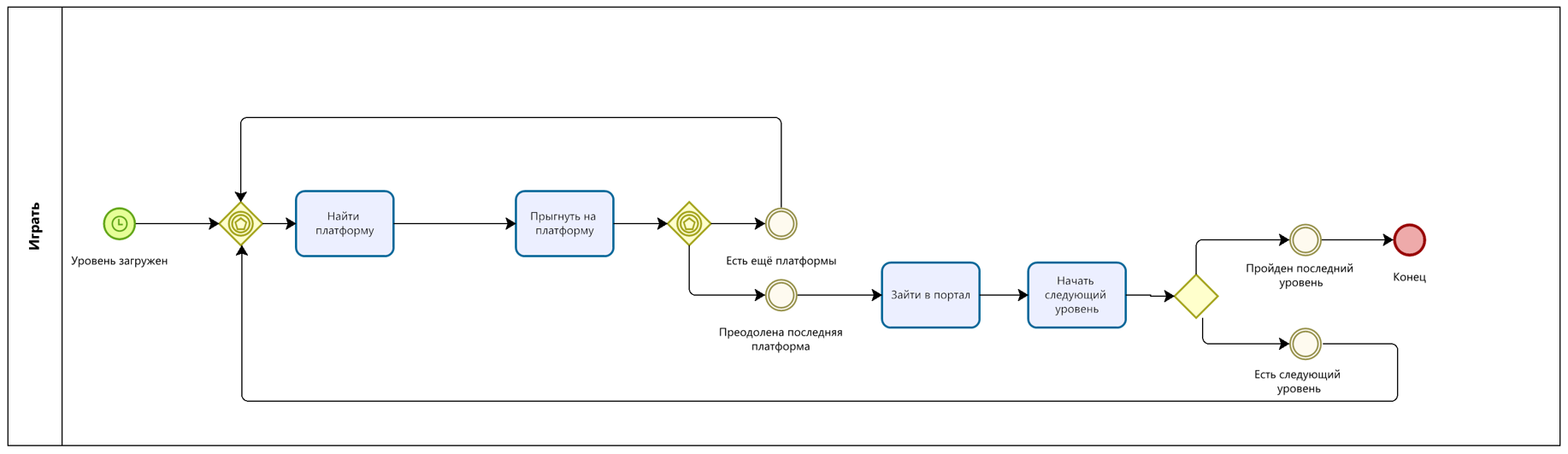
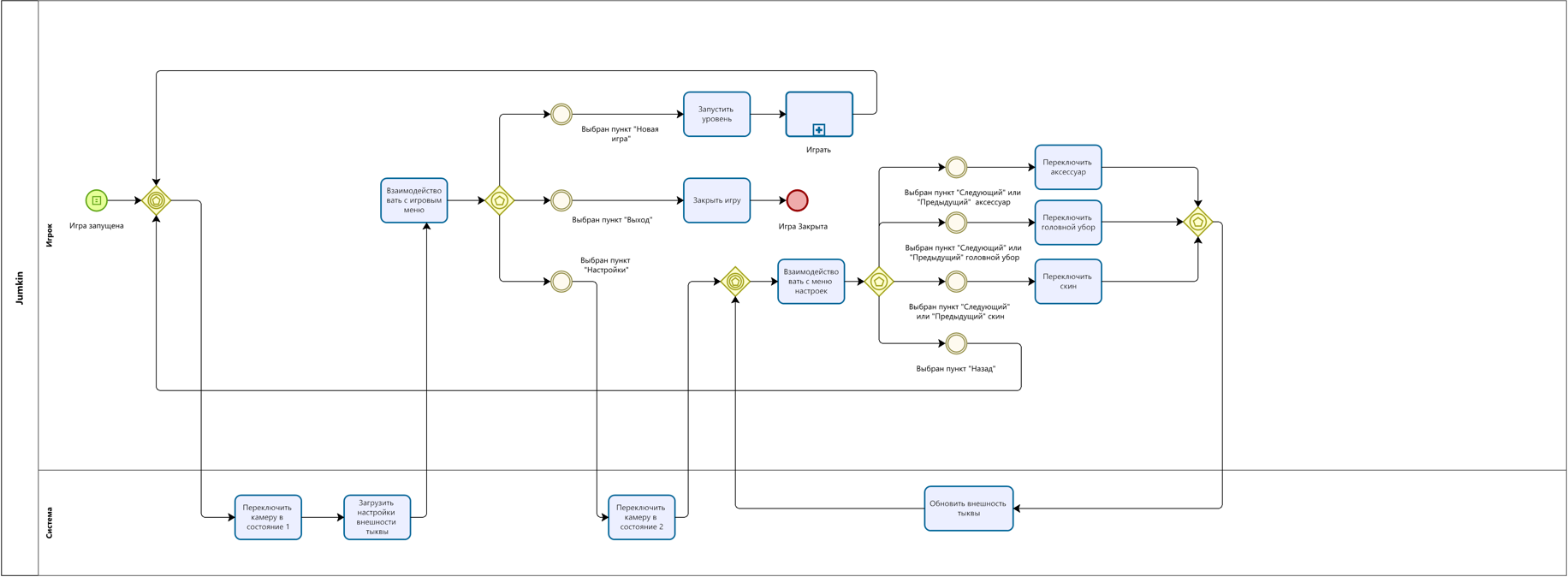


Рисунок 1.6 – Подпроцесс "Играть" в BPMN-диаграмме "Jumpkin"

Рисунок 1.5 – BPMN-диаграмма "Jumpkin"

## Вывод:

В ходе выполнения практической работы было построено две структурные диаграммы, а именно диаграмма классов и диаграмма объектов, которые описывают структуру проекта. Был разобран процесс, происходящий внутри проекта, разбор производился в нотации BPMN.

# **Практическая работа №6**

**Цель:**

Получить навыки по определению информационного взаимодействия системы, отражению процессов обработки, хранения и передачи информации между компонентами системы и внешней средой, а также получить навыки по проектированию логической модели базы данных для своей разработки.

## Ход работы:

### Диаграмма DFD

В проектируемой системе имеется одна внешняя сущность «Игрок». Система получает данные в виде информации о нажатых игроком клавишах клавиатуры. Далее она определяет, что значат для системы нажатые клавиши и выбирает тип дальнейших действий.

Если игроку необходимо настроить игровой процесс, то запускается подпроцесс «Настроить игровой процесс», после чего система возвращает игроку данные в виде информации на графическом интерфейсе.

Если игрок запускает игровой уровень, то запускается подпроцесс «Управлять игровым процессом», после чего система возвращает игроку данные в виде информации на графическом интерфейсе.

Если игрок выходит из игры, то запускается подпроцесс «Выключить игру», после чего система возвращает игроку данные в виде информации на графическом интерфейсе и завершает все свои процессы.

На основе описанной последовательности действий была построена диаграмма в нотации DFD Гейна-Сарсона. Диаграмма представлена на Рисунке 2.1 и Рисунке 2.2.

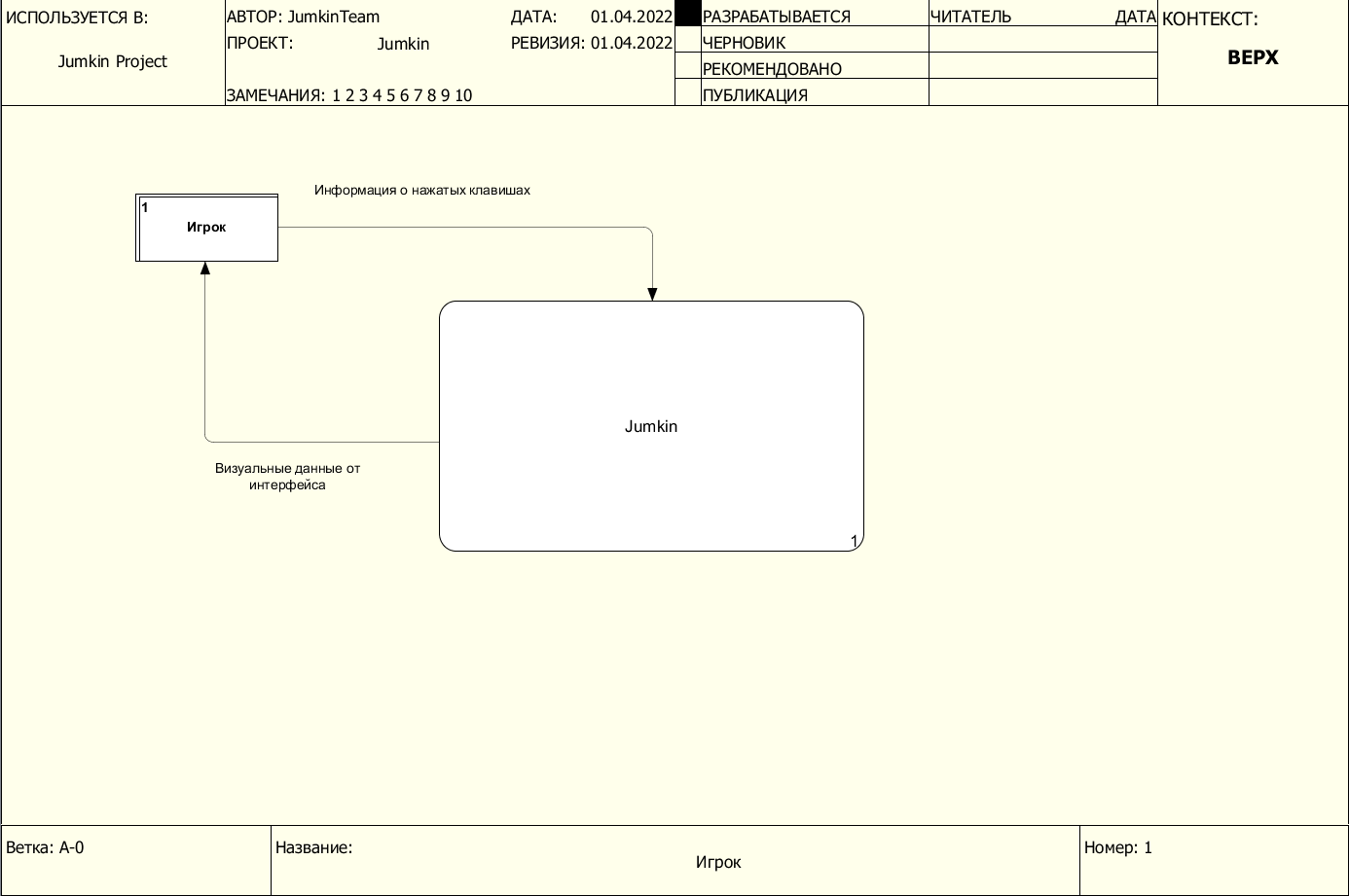


Рисунок 2.1 – Диаграмма DFD

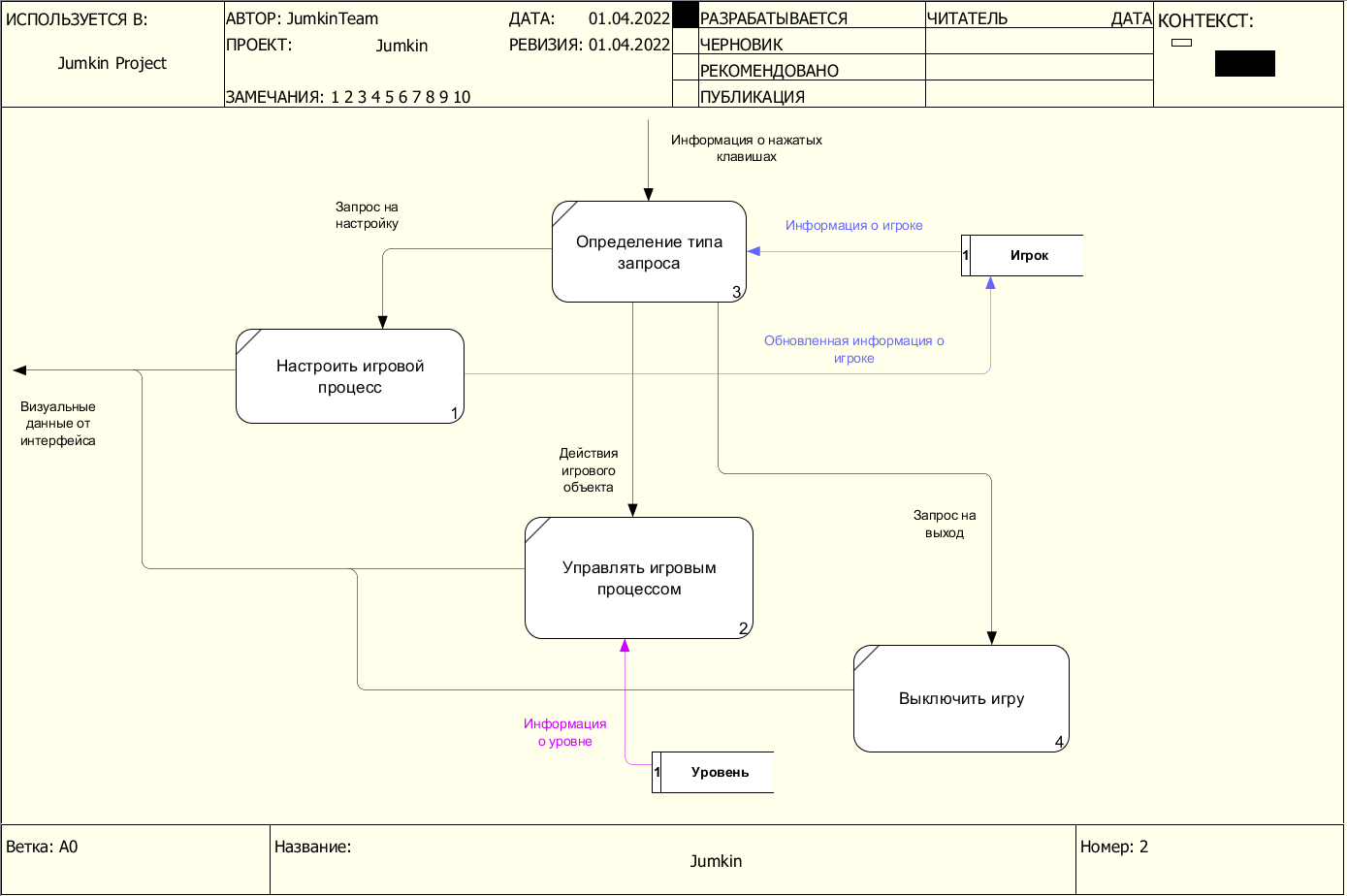


Рисунок 2.2 – Декомпозиция процесса Jumpkin

### Хранение данных

В данном проекте не планируется использование базы данных для хранения информации. Хранение пользовательской информации будет осуществляться локально в файловой системе посредством хэш-таблиц. Предварительная структура хранимых данных представлена в Таблице 2.1

Таблица 2.1 – Данные игрока

|  |  |
| --- | --- |
| PlayerData | |
| materials | Material[] |
| playerObjects | GameObject[] |
| meshes | Mesh[] |
| playerObjectMaterial | Material |
| Materials | Material[] |
| PlayerObjects | GameObject[] |
| Meshes | Mesh[] |
| PlayerObjectMaterial | Material |

## Вывод:

В ходе выполнения практической работы было составлено словесное описание информационного взаимодействия компонентов системы. На основе описания построена DFD-диаграмма. Составлено описание хранилища данных.

# **Практическая работа №7**

**Цель:**

Получить навыки по описанию архитектуры своего решения, проектированию архитектурной диаграммы, распределению сформированных требований по различным компонентам системы с последующим обновлением матрицы требований.

## Ход работы:

### Концептуальная архитектура

Для представления архитектуры разрабатываемого проекта была построена диаграмма слоёв будущей игры (Рисунок 3.1).

Unity — это непревзойденная расширяемость и готовность к адаптации в стремительно меняющихся условиях благодаря мощной системе программирования на C#, богатым API и документации. Удобные инструменты и система визуального программирования Unity позволяют избежать затрат времени на низкоуровневое программирование в ходе разработки рабочего прототипа. Изменения вносятся быстро в режиме реального времени, упрощая творческую и техническую работу.

Исходя из вышеперечисленных фактов был сделан выбор технических средств для разработки.

Проект основан на игровом движке Unity, следовательно будет реализован при помощи одноименного ПО на универсальном языке программирования C#.

Создание моделей, текстур и анимации будет выполнено при помощи программного обеспечения Blender 3.0.1. Данное решение было принято на основе следующих фактов:

* Blender это бесплатное программное обеспечение, что является идеальным вариантом для проектов с малым бюджетом;
* Приложение не уступает по количеству возможностей и функционалу более продвинутым пакетам 3D графики;

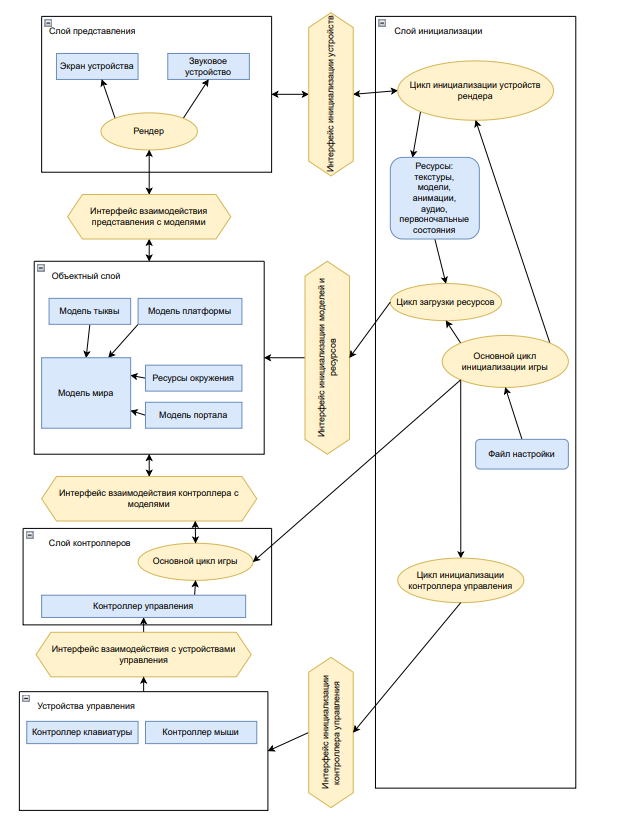


Рисунок 3.1 – Концептуальная архитектура

Отдельно выполнено построение диаграммы слоя игровой логики в модифицированной нотации ERD (Рисунок 3.2).

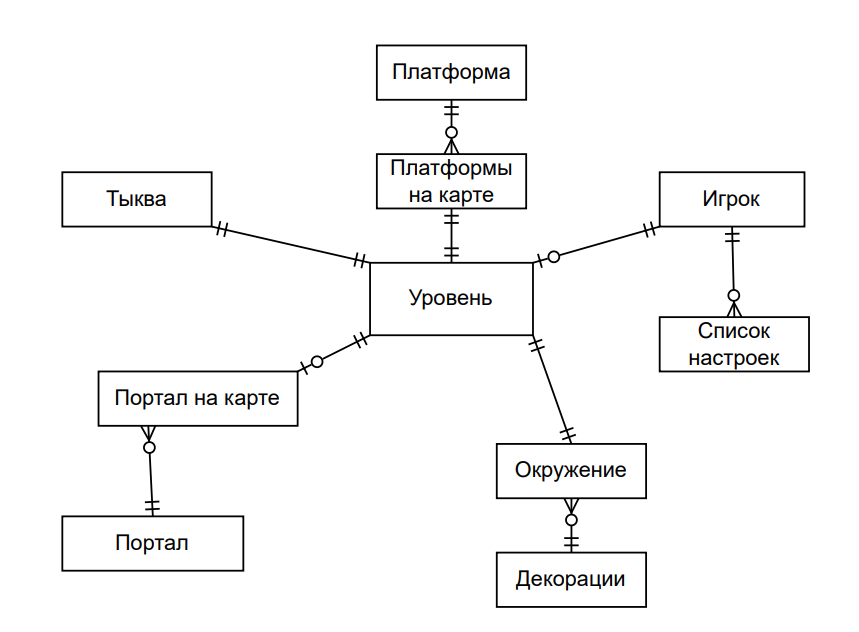


Рисунок 3.2 – Слой игровой логики

Описанную выше архитектуру системы следует отразить в ранее разработанной (в 4 практической работе) матрице требований. Изменённая таблица представлена ниже.

Таблица 3.1 - Матрица требований

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| № | Требование | Суть | Автор | Ссылки | Критерий проверки | Компоненты архитектуры |
| 1 | Игровые функции | | | | | |
| 1.1 | Настройка внешности персонажа | Игрок должен иметь возможность кастомизировать своего персонажа | Назаркин В.И. | <https://igry.ru/article-3485-luchshie-igry-s-kastomizatsiey-personaja.html> | Игрок может кастомизировать своего персонажа | Выбор производит пользователь контроллером мыши, данные передаются через интерфейс взаимодействия с устройствами управления в контроллер управления, далее в основной цикл игры через интерфейс взаимодействия контроллера с моделями в модель тыквы.  Вывод производится экраном устройства через интерфейс взаимодействия представления с моделями.  Хранение данных производится в хеш-таблице. |
| 1.2 | Выбор уровня | Игрок должен иметь возможность выбирать уровень для комфортной игры | Малютина В.А. | <https://supermeatboy.fandom.com/ru/wiki/Обычные_уровни> | Игрок может выбирать уровень для комфортной игры | Выбор производит пользователь контроллером мыши, данные передаются через интерфейс взаимодействия с устройствами управления в контроллер управления, далее в основной цикл игры.  Вывод производится экраном устройства через интерфейс взаимодействия представления с моделями.  Хранение данных производится в хеш-таблице. |
| 1.3 | Настройка времени суток | Игрок должен иметь возможность настраивать окружение для удобной игры | Назаркин В.И. | <https://modmine.net/statii/585-kak-pomenyat-vremya-sutok-v-maynkraft.html> | Игрок может настраивать окружение для удобной игры | Выбор производит пользователь контроллером мыши, данные передаются через интерфейс взаимодействия с устройствами управления в контроллер управления, далее в основной цикл игры через интерфейс взаимодействия контроллера с моделями в ресурсы окружения.  Вывод производится экраном устройства через интерфейс взаимодействия представления с моделями.  Хранение данных производится в хеш-таблице. |
| 1.4 | Функция прыжка и бега | Персонаж игрока должен иметь возможность прыгать и бегать. | Назаркин В.И. | <https://www.youtube.com/watch?v=7b2x1hA3zA4&t=641s> | Персонаж игрока может прыгать и бегать | Пользователь вводит данные контроллером клавиатуры, данные передаются через интерфейс взаимодействия с устройствами управления в контроллер управления, далее в основной цикл игры через интерфейс взаимодействия контроллера с моделями в модель мира.  Вывод производится экраном устройства через интерфейс взаимодействия представления с моделями. |
| 1.5 | Начало и конец уровня | Игровой уровень должен иметь начало и конец | Малютина В.А. | <https://ru.wikipedia.org/wiki/Платформер> | Игровой уровень имеет начало и конец | Вывод производится экраном устройства через интерфейс взаимодействия представления с моделями.  Хранение данных производится в хеш-таблице. |
| 1.6 | Наполнение уровня | Уровень должен содержать платформы | Малютина В.А. | <https://gamersgate.ru/reviews/zhanr-kompyuternykh-igr-platfo/> | Уровень содержит платформы | Вывод производится экраном устройства через интерфейс взаимодействия представления с моделями. |
| 1.7 | Управление камерой | Игрок должен иметь возможность управлять камерой с помощью мыши | Тузов И.В. | <https://skillbox.ru/media/gamedev/kamera_v_igrakh_kak_rasskazat_istoriyu_i_pokazat_mir/> | Игрок может управлять камерой | Пользователь передаёт данные контроллеру мыши, данные передаются через интерфейс взаимодействия с устройствами управления в контроллер управления, далее в основной цикл игры через интерфейс взаимодействия контроллера с моделями в модель тыквы.  Вывод производится экраном устройства через интерфейс взаимодействия представления с моделями. |
| 1.8 | Препятствия при прохождении уровня | Игровой уровень должен содержать препятствия | Малютина В.А. | <http://aushestov.ru/как-сделать-хороший-платформер/> | Уровень содержит препятствия | Вывод производится экраном устройства через интерфейс взаимодействия представления с моделями. |
| 1.9 | Наличие «progress bar» | Пользователь должен видеть прогресс прохождения уровня | Назаркин В.И. | <https://www.youtube.com/watch?v=-0EbGvtaYec> | Игрок может следить з прогрессом прохождения уровня | Вывод производится экраном устройства через интерфейс взаимодействия представления с моделями.  Хранение данных производится в хеш-таблице. |
| 1.10 | Наличие текстовых подсказок во время игры | Игрок должен получать текстовые подсказки о доступных ему возможностях. | Назаркин В.И. | <http://aushestov.ru/как-сделать-хороший-платформер/> | Игрок получает текстовые подсказки од доступных ему возможностях | Вывод производится экраном устройства через интерфейс взаимодействия представления с моделями. |
| 1.11 | Секретные места на уровнях | Уровни должны содержать неприметные места с секретами. | Назаркин В.И. | <http://aushestov.ru/как-сделать-хороший-платформер/> | Уровни содержат неприметные места с секретами. | Вывод производится экраном устройства через интерфейс взаимодействия представления с моделями. |
| 2 | Инструменты разработчика | | | | | |
| 2.1 | Система логов | В программе должна быть реализована система логов | Изотина А.А. | <https://habr.com/ru/post/651637/> | В программе реализована система логов | Вывод производится экраном устройства. |
| 2.2 | Проверка входной информации | В программе должна быть реализована проверка входной информации на соответствие типов и диапазонов | Изотина А.А. | <https://habr.com/ru/post/651637/> | В программе организована проверка входной информации | Проверка производится файлом настройки в слое инициализации. |
| 2.3 | Игровые технические ограничения | Должны быть определены минимальные и рекомендованные требования для установки игры | Изотина А.А. | <https://vgtimes.ru/guides/34430-dlya-chego-nuzhno-znat-sistemnye-trebovaniya-igry.html> | Определены минимальные и рекомендованные требования для пользователей | Проверка производится файлом настройки в слое инициализации. |
| 2.4 | Требования к FPS | Компьютерная игра должна производить минимум 30 FPS, оптимально – 60 FPS | Тузов И.В. | <https://vgtimes.ru/guides/34430-dlya-chego-nuzhno-znat-sistemnye-trebovaniya-igry.html> | Компьютерная игра производит 30FPS | Проверка производится файлом настройки в слое инициализации. |
| 2.5 | Требование к ОЗУ | Количество ОЗУ, используемой во время игры не должно превышать 2 ГБ | Изотина А.А. | <https://vgtimes.ru/guides/34430-dlya-chego-nuzhno-znat-sistemnye-trebovaniya-igry.html> | ОЗУ потребляет менее 2 ГБ | Проверка производится файлом настройки в слое инициализации. |
| 2.6 | Требование к видеопамяти | Количество видеопамяти, используемой во время игры не должно превышать 2 ГБ | Изотина А.А. | <https://vgtimes.ru/guides/34430-dlya-chego-nuzhno-znat-sistemnye-trebovaniya-igry.html> | Видеопамять потребляет менее 2ГБ | Проверка производится файлом настройки в слое инициализации. |
| 3 | Пользовательский интерфейс | | | | | |
| 3.1. | Локализация игрового процесса | Игрок должен иметь возможность настраивать язык игры | Изотина А.А. | <https://ru.wikipedia.org/wiki/Локализация_компьютерной_игры> | Игрок может настраивать язык игры | Выбор производит пользователь контроллером мыши, данные передаются через интерфейс взаимодействия с устройствами управления в контроллер управления, далее в основной цикл игры через интерфейс взаимодействия контроллера с моделями в ресурсы окружения, где выбирается язык.  Вывод производится экраном устройства через интерфейс взаимодействия представления с моделями. |
| 3.2 | Запуск игры | Пользователь должен уметь запускать игру, кликая по иконке приложения | Тузов И.В. | <https://doma35.ru/computers/cherez-kakuyu-programmu-otkryt-igru-na-kompyutere/> | Игрок может запускать игру | Пользователь запускает основной цикл инициализации игры, далее начинается цикл загрузки ресурсов и цикл инициализации устройств рендера.  Вывод производится экраном устройства через интерфейс взаимодействия представления с моделями. |
| 3.3 | Установка настроек игры | Игрок должен иметь возможность настраивать игру под себя | Тузов И.В. | <https://igry.ru/article-3485-luchshie-igry-s-kastomizatsiey-personaja.html> | Игрок может настраивать игру под себя | Выбор производит пользователь контроллером мыши, данные передаются через интерфейс взаимодействия с устройствами управления в контроллер управления, далее в основной цикл игры через интерфейс взаимодействия контроллера с моделями в ресурсы окружения.  Вывод производится экраном устройства через интерфейс взаимодействия представления с моделями. |
| 3.4 | Вывод информационных окон пользователю | В случае возникновения ошибок игроку должны быть выведены информативные диагностические сообщения. | Изотина А.А, | <https://habr.com/ru/post/651637/> | Игрок видит информационные окна | Вывод производится экраном устройства через интерфейс взаимодействия представления с моделями. |
| 3.5 | Наличие текстовых подсказок | Элементы интерфейса должны быть снабжены тестовыми подсказками | Тузов И.В. | <https://vc.ru/flood/10629-visual-cues> | Элементы интерфейса снабжены тестовыми подсказками | Вывод производится экраном устройства через интерфейс взаимодействия представления с моделями. |
| 4 | Оформление игры | | | | | |
| 4.1 | Иконка игры | Пользователь должен легко отличать файл запуска игры от остальных файлов. | Малютина В.А. | <https://habr.com/ru/post/303526/> | Игрок может отличать файл игры от любого другого файла | Выполнение норм должны обеспечивать все компоненты программной системы. |
| 4.2 | Разнообразие уровней | Игровые уровни должны различаться по дизайну | Малютина В.А. | <http://aushestov.ru/как-сделать-хороший-платформер/> | Игровые уровни различаются по дизайну | Выполнение норм должны обеспечивать все компоненты программной системы. |
| 4.3 | Звуковое сопровождение | Пользователь должен иметь возможность проходить уровни под музыкальное сопровождение | Малютина В.А. | <https://www.youtube.com/watch?v=dkt48dDcVEU> | Пользователь может проходить уровни под музыкальное сопровождение | Выполнение норм должны обеспечивать все компоненты программной системы. |
| 4.4 | Интуитивно понятное прохождение уровня | Игровой уровень должен быть спроектирован так, чтобы игрок интуитивно понимал путь его прохождения | Назаркин В.И. | <http://aushestov.ru/как-сделать-хороший-платформер/> | Игрок может пройти уровень интуитивно | Выполнение норм должны обеспечивать все компоненты программной системы. |
| 4.5 | Лёгкое и приятное цветовое оформление | Игровой дизайн должен быть разработан с преобладанием светлой и позитивной цветовой гаммы. | Малютина В.А. | <http://aushestov.ru/как-сделать-хороший-платформер/> | Игровой дизайн составлен со светлой палитрой цветов | Выполнение норм должны обеспечивать все компоненты программной системы. |

## Вывод:

Были получены навыки по описанию архитектуры своего решения, проектированию архитектурной диаграммы, распределению сформированных требований по различным компонентам системы с последующим обновлением матрицы требований.

# Практическая работа №8

**Цель:**

Получение навыков по оформлению технического задания в соответствии с ГОСТ 34.602-2020 или ГОСТ 19.201-78.

## Ход работы:

### Введение

Компьютерные игры – это прекрасный способ провести время и отвлечься от реальности. Вот уже полстолетия игры прочно занимают место на торговом рынке и не теряют актуальности. За всю их историю развилось большое количество различных жанров, начиная с текстовых квестов, заканчивая инновационными проектами, использующими технологию виртуальной реальности. Свою актуальность не теряют и игры-платформеры, которые в начале своего пути были 2D, но в процессе эволюции перешли в 3D. В данных практических работах разрабатываемым программным обеспечением является игра-платформер “Jumpkin”.

### Основание для разработки

Тема проектируемого программного обеспечения «3D игра-платформер «Jumpkin»» и на основании данной темы производится разработка данного продукта.

### Назначение разработки

Разрабатываемое приложение должно выполнять функцию игры. Оно должно быть доступно любому пользователю на его персональном компьютере. Пользователь должен иметь возможность хорошо проводить время за игрой, быть погруженным в прохождение уровней от третьего лица, и иметь свободу кастомизации своего персонажа. Данное ПО предназначено для обеспечения пользователя приятным досугом и залипательным интерактивом.

### Требования к программе или программному изделию

#### Требования к функциональным характеристикам

Разрабатываемая информационная должна выполнять следующие функции:

* Настройка внешности персонажа
* Выбор уровня
* Настройка времени суток
* Функция прыжка и бега
* Начало и конец уровня
* Наполнение уровня
* Управление камерой
* Препятствия при прохождении уровня
* Наличие «progress bar»
* Наличие текстовых подсказок во время игры
* Секретные места на уровнях
* Запуск игры
* Изменение настроек игры

#### Требования к надежности

Надежность должна обеспечиваться за счет:

* Проверки входной информации на соответствие типов, принадлежность диапазону допустимых значений и соответствие структурной корректности.
* Вывода информативных диагностических сообщений.
* Системы логов.
* Соблюдения правил эксплуатации и технического обслуживания программно-аппаратных средств;
* Предварительного обучения пользователей.

#### Условия эксплуатации

Для использования данной игры пользователю не нужно иметь специальных знаний, кроме базовых навыков работы с компьютером и небольшого опыта в играх.

#### Требования к составу и параметрам технических средств

Для обеспечения хорошей работы информационной системы необходимы следующие минимальные характеристики:

* ОС: Windows 10
* Процессор: Intel(R) Core(TM) i5-8300H CPU @ 2.30GHz
* Видеокарта: NVIDIA GeForce GTX 680
* Оперативная память: 4GB
* Жесткий диск: 7GB
* Видеопамять: 4GB
* Звуковая карта: Совместимая с DirectX;
* DirectX 12;
* Клавиатура, Мышь.

Рекомендованные характеристики для стабильной игры:

* ОС: Windows 10
* Процессор: Intel(R) Core(TM) i5-8300H CPU @ 2.30GHz
* Видеокарта: NVIDIA GeForce GTX 1050
* Оперативная память: 8GB
* Жесткий диск: 10GB
* Видеопамять: 4GB
* Звуковая карта: Совместимая с DirectX;
* DirectX 12;
* Клавиатура, Мышь.

#### Требования к производительности

Для обеспечения стабильной работы разрабатываемой игры к производительности предъявляются следующие требования:

* Компьютерная игра должна производить минимум 30 FPS, оптимально – 60 FPS
* Количество ОЗУ, используемой во время игры не должно превышать 2 ГБ
* Количество видеопамяти, используемой во время игры не должно превышать 2 ГБ

#### Требования к информационной и программной совместимости

Специальные требования не предъявляются.

#### Требования к маркировке и упаковке

Требования не предъявляются.

#### Требования к транспортированию и хранению

Требования не предъявляются.

#### Требования к доступности и локализации

Приложение не должно требовать обязательной регистрации пользователя, а также должно быть доступно для скачивания любому желающему. Для удобства скачивания приложение должно располагаться на игровой платформе.

Также приложение должно быть многоязычным для пользователей из разных стран, однако основным языком должен быть русский.

### Требования к программной документации

Проектная документация должна быть разработана в соответствии с ГОСТ 19.201-78 и ГОСТ 7.32-2017.

Отчетные материалы должны включать в себя текстовые материалы (представленные в виде бумажной копии или на цифровом носителе в формате MS Word) и графические материалы.

Предоставить документы:

1. Схемы:
   1. Диаграмма вариантов использования
   2. Процессно-событийная диаграмма
   3. Диаграмма классов
   4. BPMN-диаграмма
   5. Диаграмма объектов
   6. DFD-диаграмма
   7. Диаграмма архитектуры
   8. ERD-диаграмма
2. Техническое задание
3. Пояснительная записка
4. Руководство пользователя
5. Программа и методика испытаний

### Стадии и этапы разработки

Всего планом предусмотрено 3 этапа разработки продукта, которые представлены ниже в качестве таблицы.

Таблица 4.1 - План ведения разработки программного продукта

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Роли** | Дизайнер | Разработчик | Тестировщик-аналитик | Менеджер-тех.писатель |
| **Первый этап разработки** | Разработать концепты локаций | Создать тестовую локацию по проекту дизайнера | Анализ предметной области и разработка функциональных требований | | |
| Разработать концепт и образ персонажа | Создать тестовую версию персонажа по концептам дизайнера | Проектирование диаграмм и анализ целевой аудитории | Распределение задач между участниками команды |
| Спроектировать тестовую локацию | Внедрить тестовые локации в игровой проект | Создание матрицы требований | Написать план работ |
| Разработать локации по концептам | Настроить физику персонажа и его функции в готовых локациях | Моделирование информационной системы | Написать технические задания участникам команды |
| Назначить дедлайны по задачам |
| **Второй этап разработки** | Подбор или разработка музыкального сопровождения игры | Добавление визуальных эффектов и аудио-сопровождения на уровни игры | Подготовка к документированию разработки |
| Разработка пользовательского интерфейса игры | Добавление пользовательского интерфейса в игру | Тестирование пробных уровней | Сбор сопровождающей документации от членов команды |
| Разработка логотипа игры | \*Добавление дополнительных возможностей | Проверка кода на данном этапе |  |
| **Третий этап разработки** |  | Доработка проекта в зависимости от результата тестирования | Тестирование программного продукта | Подготовка методик и программы тестирования |
|  |  |  | Написание сопровождающей документации |
| Защита проекта | | | | |

### Порядок контроля и приемки

Для проверки пригодности разрабатываемого приложения необходимо по завершению основных работ протестировать готовый продукт. Для тестирования будет применятся методика BDD. Таким образом, через данную методику будет являться возможным полностью подготовить игру к выпуску за короткое время.

Приложение считается готовым к приемке, если оно:

* Работает
* Соответствует требованиям производительности
* Соответствует требованиям надёжности
* Удовлетворяет 70% заявленных требований

## Вывод:

В данной практической работе было составлено техническое задание в соответствии с ГОСТ 19.201-78. Данный ГОСТ был выбран для написания технического задания, так как он лаконичен, краток и подходит для описания любой разрабатываемой информационной системы (в данном компьютерной игры).

# Заключение

В ходе данных четырёх практических работ был и разработаны схемы проекта в нотациях DFD, BPMN и диаграммы классов и объектов, разработана и описана концептуальная архитектура системы и на основание ранее разработанных требований написано техническое задание. (на этом мои полномочия всё)