Учреждение образования

«Гродненский государственный политехнический колледж»

Специальность: 2-40 01 01 «Программное обеспечение информационных технологий»

Специализация: 2-40 01 01.35 ««Программное обеспечение обработки экономической и деловой информации»

Предмет «Конструирование программ и языки программирования»

Группа: ПЗТ-40

КУРСОВОЙ ПРОЕКТ

Тема: Разработка игрового приложения «Путь авантюриста»

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Разработала К.В.Савило

Руководитель проекта А.Г.Бабуль

Содержание

Введение	4
1 Анализ предметной области и формулировка требований к программе	5
1.1 Исследование предметной области	5
1.2 Инструменты разработки	5
2 Проектирование	5
2.1 Диаграмма вариантов использования	
2.2 Диаграмма деятельности	8
2.3 Описание тестов	10
3 Построение программы	11
4 Тестирование	16
5 Применение	17
5.1 Назначение и условия применения программы	17
5.2 Инсталляция	17
5.3 Выполнение программы	19
Заключение	24
Список использованных источников	25
Приложение А Листинг программы	
Приложение Б Описание тестов	
Приложение В Тестовый сценарий	

Изм.	Лист	№докум.	Подпись	Дата	КП 2-40 01 01.35.40	0.12.25	73	
Разр	σαδ.	Савило				Лит.	Лист	Листов
Пров	3.	Бабуль			Разработка игрового		3	<i>25</i>
					приложения «Путь			
Н. к	онтр.				авантюриста»			
Утв.	,				acanimopacina,			

Введение

На данном курсовом проекте была поставлена задача разработать игровое приложение «Путь авантюриста». Данный программный продукт позволит пользователю получить эстетическое удовольствие, создаст неповторимый игровой опыт, сочетающий приключения, исследование, торговлю и сражения. Игроки получат удовольствие от самого процесса путешествия, открытия новых локаций и достижения успеха.

Создаваемое приложение ориентировано на пользователей старше 12 лет. Далее приведено краткое описание разделов пояснительной записки.

Первый раздел носит название «Анализ предметной области и формулировка требований к программе». В нем можно ознакомиться с постановкой задачи, которая включает в себя: исследование предметной области поставленной задачи, функциональные и нефункциональные требования к программному продукту. В подразделе «Инструменты разработки» рассмотрена среда, в которой создается данный программный продукт.

В разделе «Проектирование задачи» рассмотрены основные аспекты разработки программного продукта. Здесь можно узнать об организации данных в контексте среды разработки. В данном разделе составлены диаграмма вариантов использования и диаграмма деятельности.

«Построение программы» — третий раздел отчета, в котором описываются все элементы и объекты, которые использованы при реализации данного приложения. В этом разделе описана диаграмма классов.

Четвертый раздел — «Тестирование». В нем представлен отчёт о проведении тестирования.

В разделе «Применение» описано назначение программы, область применения и среда функционирования программного обеспечения.

«Заключение» содержит краткую формулировку задачи, результаты проделанной работы, описание использованных методов и средств, описание степени автоматизации процессов на различных этапах разработки.

В разделе «Список использованных источников» приведен список используемых при разработке источников.

В приложении А приведен листинг программы.

В приложении Б представлено описание тестов.

В приложении В представлен тестовый сценарий.

В графической части представлена диаграмма классов

Изм.	Лист	№докум.	Подпись	Дата

1 Анализ предметной области и формулировка требований к программе

1.1 Исследование предметной области

Наименование задачи – игровое приложение «Путь авантюриста».

Цель разработки данного программного продукта заключается в создании игрового приложения, которое позволит пользователю окунуться в фэнтезийный мир увлекательных приключений, где он в роли мага-авантюриста, сможет выполнять интересные задания и исследовать игровой мир.

Игровое приложение выполнено в жанре RPG в стиле фентезисредневековья.

Назначение: данный программный продукт ориентирован на несколько категорий игроков: любителей аниме и фэнтези, поклонников ролевых игр (RPG), фанатов крафта и торговли, любителей экшена и приключений, креативных игроков. Таким образом, игра будет привлекательна для широкой аудитории, включая как казуальных игроков, так и более опытных любителей RPG, что делает её универсальной и многогранной.

В центре сюжета стоит авантюрист, чей жизненный путь состоит в поиске приключений. Игрок приходит в город, где ему предстоит вступить в гильдию авантюристов «Бесконечный путь», здесь его будут ждать разнообразные задания, от охоты на монстров до расшифровки древних текстов. Игра предлагает игрокам возможность развивать своего персонажа, повышая его ранг в гильдии и открывая новые уровни сложности и уникальные награды. Игрок может взаимодействовать с NPC, покупать и продавать различные предметы, а также украшать свой собственный дом.

Задача игрока - достичь наивысшего ранга, в гильдии, выполнив все задания и получить звание лучшего авантюриста, а также же наслаждаться увлекательными приключениями в мире фэнтези.

Периодичность использования данного программного продукта неограниченна. Игрок может в любой момент сбросить игровой прогресс, а так же имеет возможность даже после выполнения всех заданий продолжать игру.

Существует множество игр об авантюристах, однако абсолютных аналогов данного программного продукта не было найдено.

1.2 Инструменты разработки

Для разработки данного проекта была выбрана среда Unity, которая является межплатформенной средой для создания и разработки видеоигр и приложений, разработанной компанией Unity Technologies.

				·
Изм.	Лист	№докум.	Подпись	Дата

Разработка будет производиться на языке программирования С# — многофункциональный язык программирования, активно развивающийся на данный момент. Он часто используется как первый язык программирования начинающих программистов, так и для реализации крупных коммерческих проектов.

Иные инструменты, используемые при разработке и написании сопутствующей документации:

- WEB-ресурс DRAW.IO будет использоваться для создания графической части и разработки UML-диаграмм;
- Microsoft Office Word 2013 для написания документации к программному продукту;
 - Smart Install Maker утилита для создания инсталляторов;
 - Dr. Explain инструмент разработки пользовательской документации;
 - GitHub веб-сервис для хостинга IT-проектов;
- Google Drive сервис хранения, редактирования и синхронизации файлов, разработанный компанией Google.

При разработке данного программного продукта был использован компьютер со следующими характеристиками:

- процессор Intel Core i5-12500H (3.30 / 4.50 GHz);
- O3У: 8Gb;
- память: HDD 512Gb;
- OC: Windows 10.

Изм.	Лист	№докум.	Подпись	Дата

2 Проектирование

2.1 Диаграмма вариантов использования

Диаграмма вариантов использования — диаграмма, отражающая отношения между актерами и прецедентами и являющаяся составной частью модели прецедентов, позволяющей описать систему на концептуальном уровне.

Суть такой диаграммы состоит в следующем: проектируемая система представляется в виде множества сущностей или актеров, взаимодействующих с системой с помощью так называемых вариантов использования.

В данной диаграмме вариантов использования актером является игрок.

Игроку доступны следующие варианты использования: играть, читать справку, изменить музыку, выти из игры.

У варианта использования «Играть» наблюдаются следующие «extend»связи: посмотреть инвентарь, перейти на локации, взаимодействовать с персонажем, купить игровые предметы, продать добытые ресурсы, пройти тест на вступление в гильдию, украсит дом, выбрать задание соответствующее рангу.

У варианта использования «Изменить настройки» наблюдаются следующие «extend»-связи: выбрать мелодию, остановить воспроизведение музыки, продолжить воспроизведение музыки.

У варианта использования «Посмотреть инвентарь» наблюдаются следующие «extend»-связи: перейти к использованию некоторых предметов, посмотреть информацию о предметах.

У варианта использования «Взаимодействовать с персонажем» наблюдаются следующие «extend»-связи: начать диалог.

У варианта использования «Украсить дом» наблюдаются следующие «include»-связи: разместить предмет интерьера.

У варианта использования «Выбрать задание, соответствующее рангу» наблюдаются следующие «extend»-связи: собрать растения, расшифровать сообщение, собрать кристаллы, создать зелье, изловить белку.

У варианта использования «Перейти к использованию некоторых предметов» наблюдаются следующие «extend»-связи: создать зелье, посмотреть информацию в руководстве начинающего травника, посмотреть информацию в руководстве по зельеварению.

Диаграмма вариантов использования представлена на рисунке 1.

Изм.	Лист	№докум.	Подпись	Дата

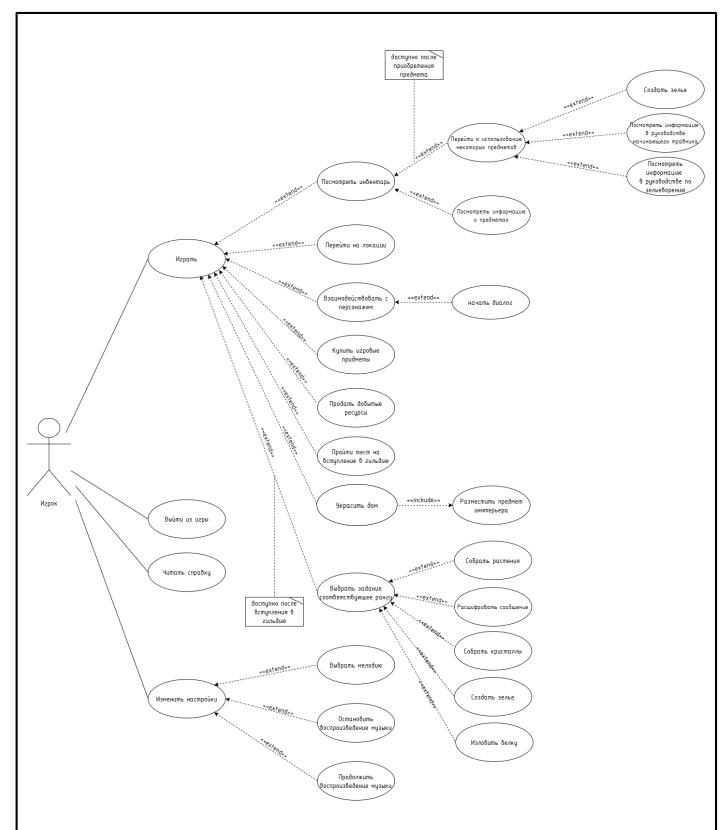


Рисунок 1 – Диаграмма вариантов использования

2.2 Диаграмма деятельности

Диаграмма деятельности — это поведенческая диаграмма, которая показывает поток работы или действий в рамках системы или процесса. Она отображает последовательность шагов и возможные варианты выполнения работы,

				·
Изм.	Лист	№докум.	Подпись	Дата

включая параллельные процессы и ветвления. Диаграмма деятельности включает в себя такие элементы, как начальные и конечные узлы, узлы действий, узлы решений, вилки и слияния, а также потоки управления, которые связывают эти узлы.

На диаграмме отображен процесс покупки предмета. Диаграмма деятельности представлена на рисунке 2.

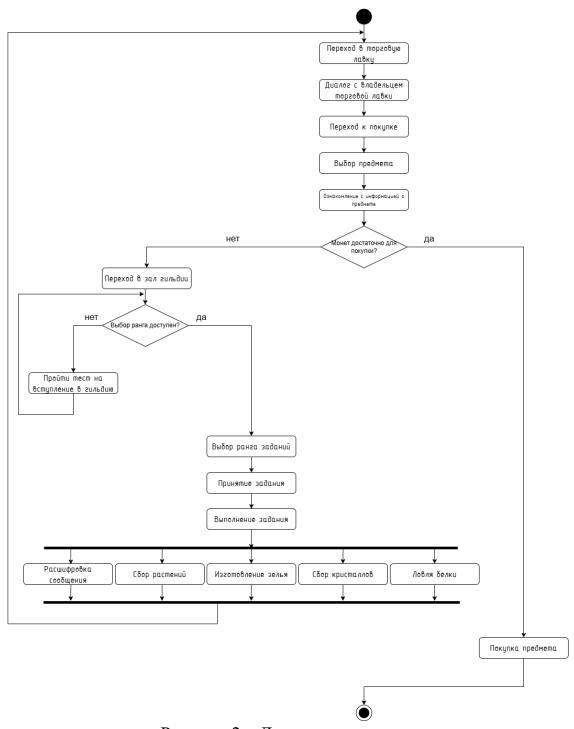


Рисунок 2 – Диаграмма деятельности

Изм.	Лист	№докум.	Подпись	Дата

2.3 Описание тестов

При разработке приложения необходимо будет провести тест по игровому процессу.

Тесты по игровому процессу проводятся для обеспечения высокого качества игры, выявления ошибок, улучшения геймплея и повышения удовлетворенности игроков. Они помогают разработчикам понять, насколько увлекательна и удобна игра, выявить баги и несоответствия требованиям, а также оценить общую игровую механику.

Описание тестового сценария представлено в приложении Б.

				·
Изм.	Лист	№докум.	Подпись	Дата

3 Построение программы

Диаграмма классов — это визуальное представление структуры системы, отображающее классы, их атрибуты, методы и взаимосвязи между ними. Она является одним из ключевых элементов объектно-ориентированного проектирования и используется для моделирования статической структуры приложения.

Основная цель диаграммы классов — показать, как элементы системы связаны друг с другом, какие свойства и функции они имеют, а также какие отношения существуют между ними.

Диаграмма помогает разработчикам и архитекторам понять архитектуру системы, определить связи между объектами и грамотно спроектировать базу данных или архитектуру приложения.

В данном разделе будут описаны основные 11 классов из 64.

Класс «playerControler» содержит:

Свойства:

- public float speed: скорость движения игрока;
- public float jumpForce: сила прыжка;
- public float moveInput: входные данные для движения;
- private Rigidbody2D rb: компонент Rigidbody2D для физики;
- private bool facing Right: направление, в котором смотрит игрок;
- private bool isGrounded: состояние, находится ли игрок на земле;
- public Transform freePos: позиция для проверки касания земли;
- public float checkRadius: радиус проверки касания;
- public LayerMask whatIsGround: слой, определяющий, что считается землей;
 - private Animator anim: компонент Animator для анимации.

Методы:

- private void Start(): инициализация компонентов;
- private void FixedUpdate(): обновление физики, движение игрока;
- private void Update(): обновление состояния игрока;
- void Flip(): переворот игрока в другую сторону.

Класс «Bullet_enemy» содержит:

Свойства:

- public float offset: угол поворота пули;
- public GameObject bullet: префаб пули;
- public Transform shotPoint: точка, откуда будет вылетать пуля;
- private float timeBtwShots: время между выстрелами;
- public float startTimeBtwShots: начальное время между выстрелами;

Изм.	Лист	№докум.	Подпись	Дата

– public Transform parentObject: объект-родитель для пуль.

Методы:

- private void Start(): инициализация корутины для стрельбы;
- private IEnumerator ShootBullets(): корутина для создания пуль с задержкой;
 - private void Update(): обновление положения пули.

Класс «Bullet» содержит:

Свойства:

- public float speed: скорость движения пули;
- public float lifetime: продолжительность жизни пули;
- public float lifetime1: время, через которое пуля будет уничтожена;
- public float distance: дистанция, на которую будет двигаться пуля;
- public int damage: урон, который наносит пуля;
- public LayerMask whatIsSolid: слой для коллизий с твердыми объектами;
- public LayerMask whatIsWrag: слой для коллизий с врагами;
- private Vector2 direction: направление движения пули;
- public GameObject bulletEffect: эффект, который появляется при столкновении.

Методы:

- private void Start(): уничтожение пули через заданное время;
- public void SetDirection(Vector2 dir): установка направления движения;
- private void Update(): обновление движения пули и проверка на столкновения.

Класс «нр» содержит:

Свойства:

- public int hp: здоровье игрока;
- public Image а: изображение первой полосы здоровья;
- public Image b: изображение второй полосы здоровья.

Методы:

– public void boy(): уменьшает здоровье игрока.

Класс «Rang» содержит:

Свойства:

- public GameObject pauseMenuUI: меню паузы;
- public Text text1: текст для отображения результата;
- public int heal: количество здоровья;
- public bool b: флаг состояния.

Методы:

- public void Start(): инициализация значений;
- void Update(): проверка состояния игры;

				·
Изм.	Лист	№докум.	Подпись	Дата

– private IEnumerator HandleGameOver(): обработка окончания игры.

Класс «Епету» содержит:

Свойства:

- public int health: здоровье врага;
- public GameObject deathEffect: эффект смерти врага.

Метолы:

- private void Update(): проверка состояния здоровья и уничтожение врага;
- public void TakeDamage(int damage): уменьшает здоровье врага.

Класс «Gan» содержит:

Свойства:

- public float offset: угол поворота;
- public GameObject bullet: префаб пули;
- public Transform shotPoint: точка выстрела;
- private float timeBtwShots: время между выстрелами;
- public float startTimeBtwShots: начальное время между выстрелами.
- public Transform parentObject: объект-родитель для пуль.

Методы:

– private void Update(): обновление состояния и стрельба.

Класс «hil» содержит:

Свойства:

- public Image а: изображение для первой полосы здоровья;
- public Image b: изображение для второй полосы здоровья;
- public Text text: текст для отображения количества зелий.

Методы:

- void Update(): проверка нажатия клавиш для использования зелий;
- private void hiler(): восстанавливает здоровье игрока.

Класс «casanije wrag» содержит:

Свойства:

- public int x = 0: используется для проверки нужно ли отталкивание игрока.

Методы:

– private void OnCollisionEnter2D(Collision2D collision): нанесении урона при касании игрока и его отталкивание при необходимости.

Класс «Draggable» содержит:

Свойства:

- private Canvas canvas: ссылка на родительский Canvas;
- private RectTransform rectTransform: прямоугольная трансформация объекта;
 - private Vector3 originalPosition: оригинальная позиция объекта.

Методы:

	·			
Изм.	Лист	№докум.	Подпись	Дата

- public void OnBeginDrag(PointerEventData eventData): сохраняет
 оригинальную позицию при начале перетаскивания;
- public void OnDrag(PointerEventData eventData): обновляет позицию объекта при перетаскивании;
 - public Vector3 GetCurrentPosition(): возвращает текущую позицию объекта. Класс «JewelryManager» содержит:

Свойства:

- public GameObject[] jewelryPrefabs: массив префабов украшений;
- public Button[] buttons: кнопки для создания украшений;
- private Dictionary<int, int> quantities: словарь для хранения количества украшений;
- public Transform parentTransform: объект-родитель для созданных украшений;
 - public Vector2 spawnPosition: позиция, куда будут создаваться украшения;
- private List<Draggable> draggableJewelry: список перетаскиваемых украшений.

Методы:

- void Start(): инициализация кнопок и загрузка данных;
- void Update(): проверка нажатий для управления украшениями;
- void LoadQuantities(): загрузка количества украшений из файла;
- void LoadJewelryPositions(): загрузка позиций украшений из файла;
- public void OnJewelryButtonClick(int index): создание экземпляра украшения;
- void SavePositions(): сохранение позиций перетаскиваемых украшений в файл.

Далее приведено описание связей в диаграмме классов.

Зависимость обозначает такое отношение между классами, что изменение спецификации класса-поставщика может повлиять на работу зависимого класса, но не наоборот.

Класс Bullet вызывает методы классов hp и Enemy, тем самым уменьшая переменные жизней игрока и врага при попадании.

Класс hill изменяет(увеличивает) переменную класса hp для увеличения единиц жизни игрока.

Класс casanije_wrag вызывает метод класса hp, для нанесения урона.

Класс Rang использует информацию из hp (значение переменной hp), чтобы обновлять свои поля.

Класс Enemy изменяет переменную класса Rang, определяя условие определения ранга.

Изм.	Лист	№докум.	Подпись	Дата

Отношение ассоциации показывает, что объекты одной сущности (класса) связаны с объектами другой сущности таким образом, что можно перемещаться от объектов одного класса к другому.

Классы Gan и Bullet_enemy создают пули используют метод Bullet, чтобы задавать ей направление.

Класс JewelryManager добавляет компонент Draggable для перетаскивания объектов, то есть управления ими.

Диаграмма классов представлена в графической части проекта.

Изм.	Лист	№докум.	Подпись	Дата

4 Тестирование

При разработке данной программы многие возникающие ошибки и недоработки были исправлены на этапе реализации проекта. После завершения испытания реализации программы было проведено тщательное функциональное тестирование. Функциональное тестирование должно гарантировать работу всех элементов программы в автономном режиме.

Элементы программы были проверены, и было установлено, что все они работают правильно и выполняют задачи, указанные в классах.

Разработанный тестовый сценарий и статус его выполнения представлен в приложении Б.

Расписание проведения и время, затраченное на тестирование, описано в таблице 2.

Таблица 2 – Расписание работ над проектом

Имя	Дата	Деятельность	Продолжительность, ч
Савило Ксения	30.05.2025	Разработка тестов	2
Савило Ксения	05.06.2025	Тестирование игрового приложения	2
Савило Ксения	10.06.2025	Исправление найденных ошибок	2
Савило Ксения	15.06.2025	Проведение регрессионного тестирования	1
Савило Ксения	18.06.2025	Составление отчета о результатах тестирования	2

Элементы программы были проверены, и было установлено, что все они работают правильно и выполняют задачи, указанные в процедурах.

Статистика по всем дефектам представлена в таблице 3.

Таблица 3 – Статистика по всем дефектам

Стотио	Количество		Важн	юсть	
Статус	Количество	Низкая	Средняя	Высокая	Критическая
Найдено	1	1	0	0	0
Исправлено	1	1	0	0	0
Проверено	1	1	0	0	0
Открыто	0	0	0	0	0
заново					
Отклонено	0	0	0	0	0

Исходя из данных по приведенным ранее таблицам, можно сделать вывод, что все возможные ошибки и недочеты были найдены и исправлены.

Изм.	Лист	№докум.	Подпись	Дата

5 Применение

5.1 Назначение и условия применения программы

Цель данного программного продукта заключается в разработке программного продукта, который позволит пользователю получить эстетическое удовольствие, создаст неповторимый игровой опыт, сочетающий приключения, исследование, торговлю и сражения.

Создаваемое приложение будет рассчитано на любого пользователя.

Качество и скорость работы приложения всегда зависит от самих характеристик персонального компьютера. Поэтому приложение должно было быть протестировано на разных машинах. Тестирование проводилось на разных персональных компьютерах и результаты были удовлетворительные.

Сама программа была разработана на программном устройстве со следующими характеристиками:

– процессор Intel Core i5-12500H (3.30 / 4.50 GHz);

- O3У: 8Gb;

– память: HDD 512Gb;

- OC: Windows 10.

5.2 Инсталляция

Для того, чтобы установить программу необходимо запустить файл Setup.exe. Появится мастер установки игрового приложения «Путь авантюриста», представленный на рисунке 4.

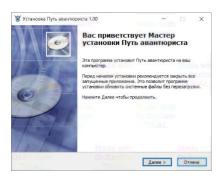


Рисунок 4 — Мастер установки игрового приложения «Путь авантюриста» После нажатия кнопки «Далее» появляется возможность выбора места для установки программного продукта, представленное на рисунке 5.

				·
Изм.	Лист	№докум.	Подпись	Дата

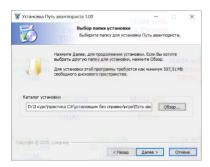


Рисунок 5 – Мастер установки игрового приложения «Путь авантюриста»

После нажатия на кнопку «Далее» появляется возможность выбрать создавать ли ярлыка игрового приложения, представленная на рисунке 6.

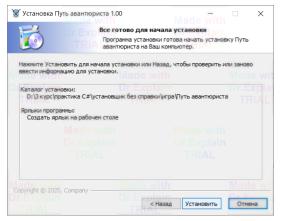


Рисунок 6 – Создание ярлыка

После нажатия на кнопку «Далее» открывается форма завершения установки, представленная на рисунке 7.

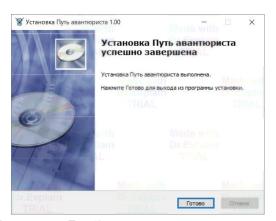


Рисунок 7 – Завершение установки

После установки на «Рабочем столе» появится ярлык для запуска приложения, который представлен на рисунке 7.

				·
Изм.	Лист	№докум.	Подпись	Дата



Рисунок 8 – Ярлык приложения

5.3 Выполнение программы

При запуске игры пользователя встречает экран загрузки. Экран загрузки представлен на рисунке 9.



Рисунок 9 – Экран загрузки

После игрок попадает на Главное меню, как представлено на рисунке 10.



Рисунок 10- Главное меню

У пользователя есть возможность кликнуть по кнопке «Выйти из игры», «Музыка», «Играть», «Справка».

При нажатии на кнопку в виде изображения листка с вопросительным знаком будет открыта справка.

При нажатии на кнопку «Выход из игры» приложение будет закрыто.

Изм.	Лист	№докум.	Подпись	Дата

При нажатии на кнопу играть попадаете на сцену «Входа в город», где вас встречает предисловие к игре. Сцена «Входа в город» представлена на рисунке 11.



Рисунок 11- Сцена «Входа в город»

Для перелистывания предисловия необходимо нажать на кнопки стрелки, расположенную в нижнем правом углу листа предисловия.

После прочтения предисловия, можно продолжить игру нажав на кнопку «В путь», или сбросить игровой прогресс к изначальному (если таковой был), нажав на кнопку «Начать с нуля».

Нажав на одну из данных кнопок вы попадаете на улицу города где расположены здания, в которые можно войти нажав на дверь.

Управление персонажем осуществляется при помощи стрелок клавиатуры, или же клавиш WASD: левая стрелка или клавиша А и правая стрелка или клавиша D это движения в лево и право соответственно, верхняя стрелка или W - прыжок, нижняя стрелка или клавиша S - приседание. Для атаки используется левая клавиша мыши, атака доступна только на сценах сражения.

В левом верхнем углу расположена кнопка перехода в инвентарь (изображена в виде сумки), а в правом верхнем углу кнопка «Меню».

На сцене инвентарь можно просмотреть количество добытых во время игры ресурсов, а также информацию о них, кликнув по значку предмета из левой части инвентаря. Для некоторых предметов, предусмотрена кнопка «Использовать», которая будет располагаться в нижнем правом углу, и нажав на которую будет произведён переход на сцену использования предмета. Покинуть инвентарь можно, а также другие аналогичные сцены можно нажав кнопку «Выход». Инвентарь представлен на рисунке 12.

Изм.	Лист	№докум.	Подпись	Дата



Рисунок 12 – Инвентарь

Нажав на кнопку «Меню» будет открыто меню, на котором расположены кнопки аналогичные кнопкам главного меню. Нажав на крестик меню будет закрыто. При покупке предмета «Карта» в меню будет добавлена кнопка «Карта», которая ведёт на сцену карты, где отмечаются посещённые локации, кликнув по названиям которых можно их посетить.

Нажав на дверь здания гильдии, располагающегося на улице города, будет произведён переход на сцену «Зал гильдии», представленный на рисунке 13.



Рисунок 13 – Сцена «Зал Гильдии»

На данной сцене располагается доска заданий, с доступными заданиями и администратор гильдии. Чтобы получить доступ к задания необходимо зарегистрироваться в гильдии, подойдя к администратору, и нажав на кнопку «Начать диалог» (изображена в виде трёх вопросительных знаков), вы получите необходимую информацию о дальнейших действиях. После будет отображена кнопка «Тест», нажав на которую вы попадёте на сцену для прохождения теста.

После прохождения теста вы будете зарегистрированы в гильдии, а ваш текущий ранг будет отображён в верхнем левом углу около кнопки «Инвентарь».

Также будет открыт доступ к заданиям вашего ранга, чтобы перейти к выполнению задания, нужно выбрать ранг на доске заданий, кликнув по нему, выбранный ранг обводится рамкой. Далее нужно кликнуть по листку задания, после чего вы можете ознакомиться с условиями выполнения и наградой за выполнение,

Изм.	Лист	№докум.	Подпись	Дата

и нажать кнопку «Принять», после чего вы будете направлены на сцену соответствующего задания.

Вернуться не выполнив задание, можно нажав на кнопку «Гильдия». При успешном выполнении задания вы получите сообщение о зачислении награды, провалив задание вы также получите соответствующее сообщение с возможностью начать заново нажав на соответствующую кнопку или вернуться в гильдию.

В игре присутствуют функции торговли, перейдя на сцены магазинов, подойдя к торговцам и начав диалог описанным выше способом будут отображены кнопки «К покупке», «К продаже». Нажав на каждую из которых вы перейдёте на соответствующую сцену.

Для покупки нудно нажать на предмет и в правом углу его описания нажать кнопку «Купить». Слева отображено количество ваших монет, справа цена предмета за единицу.

Аналогично устроена сцена продажи. Для продажи нужно нажать на предмет ввести количество в поле «Шт.» и нажать кнопку «Продать».

Также в игре присутствует возможность украшать дом персонажа. Нажав на кнопку в виде плюса, расположенного около инвентаря, будет открыт список украшений, нажав на плюс повторно, список будет скрыт, нажав на кнопку из списка на сцену будет добавлено соответствующее украшение. Украшения можно перемещать с помощью курсора, удалять нажав клавишу backspace, изменять размер нажимая на + и -, поворачивать в лево и право нажав клавишу Tab, а также поворачивать по окружности, нажав клавишу Q, все эти действия можно применять к украшению на которое наведён курсор.

В игре есть функция Крафта зелий. Приобретя котёл и нажав кнопку «Использовать» в инвентаре, будет произведён переход на сцену котла, где нажав на кнопу ингредиента, и расположив его с помощью мыши в ячейки в соответствии с рецептом будет создано зелье, которое необходимо забрать, кликнув по нему. Скена котелка отображена на рисунке 14.



Рисунок 14 – Сцена «Котелок»

Изм.	Лист	№докум.	Подпись	Дата

№докум.

Подпись Дата

Заключение

Целью данного курсового проекта была разработка программного продукта «Путь авантюриста». В ходе тестирования не было выявлено исключительных ситуаций. Проект работает без сбоев и ошибок. В процессе разработки поставленной задачи были получены знания по использованию платформы Unity и закреплены знания по языку программирования С# и закреплены умения создания собственных функций.

Поставленная задача выполнена в соответствии со всеми ранее задуманными требованиями, созданы и протестированы все необходимые компоненты проекта.

В ходе тестирования все исключительные ситуации были обработаны. Проект работает без сбоев и ошибок. В поставленной задаче был реализован простой и понятный пользовательский интерфейс.

Исходя из этого, можно сделать вывод, что программа реализована успешно.

Изм	. Лист	№докум.	Подпись	Дата

Список использованных источников

- 1. Для программистов, не знакомых с Unity [Электронный ресурс]. Режим доступа: https://unity.com/ru/how-to/programming-unity/ Дата доступа: 10.04.2025.
- 2. Unity с нуля: Занятие 2. С# и ООП [Электронный ресурс]. Режим доступа: https://aelit.by/unity-s-nulya-zanyatie-2-c-i-oop/D.html Дата доступа: 20.04.2025.
- 3. Полное руководство по языку программирования С# 13 и платформе .NET 9 [Электронный ресурс]. Режим доступа: https://metanit.com/sharp/tutorial/ Дата доступа: 25.04.2025.
- 4. Создаем 2D-игру на Unity: инструкция для новичка [Электронный ресурс]. Режим доступа: https://proglib.io/p/sozdaem-2d-igru-na-unity-instrukciya-dlya-novichka-2020-09-01 Дата доступа: 30.04.2025.
- 5. Руководство по С# [Электронный ресурс]. Режим доступа: https://vscode.ru/prog-lessons/sozdanie-redaktirovanie-i-udalenie-dannyih-xml-fayla-c-sharp.html Дата доступа: 01.05.2025.

Изм.	Лист	№докум.	Подпись	Дата

Приложение А

Листинг программы

```
Файл DialogAnimator.cs:
using System.Collections;
using System.Collections.Generic;
                                                                hitInfo.collider.GetComponent<Enemy>().TakeDamag
using UnityEngine;
                                                                e(damage);
public class DialogAnimator: MonoBehaviour
                                                                       Instantiate(bulletEffect, transform.position,
  public Animator startAnim;
                                                                Quaternion.identity);
  public DialogManager dm;
                                                                       Destroy(gameObject);
  public void OnTriggerEnter2D(Collider2D other)
    startAnim.SetBool("startOpen",true);
                                                                     // Двигаем пулю в заданном направлении
                                                                     transform.Translate(direction * speed *
  public void OnTriggerExit2D(Collider2D other)
                                                                Time.deltaTime);
                                                                  }
    startAnim.SetBool("startOpen", false);
                                                                }
    dm.EndDialogue();
                                                              Файл DialogManager.cs:
                                                                using System.Collections;
                                                                using System.Collections.Generic;
Файл Bullet.cs:
                                                                using UnityEngine;
 using System;
                                                                using UnityEngine.UI;
 using System.Collections;
 using System.Collections.Generic;
                                                                public class DialogManager: MonoBehaviour
 using UnityEngine;
                                                                  public Text dialogueText;
 public class Bullet: MonoBehaviour
                                                                  public Text nameText;
    public float speed;
                                                                  public Animator boxAnim;
   public float lifetime;
                                                                  public Animator startAnim;
   public float lifetime1;
    public float distance;
                                                                  public int x;
    public int damage;
                                                                  public Button test;
   public LayerMask whatIsSolid;
                                                                  private Queue<string> sentences;
    private Vector2 direction; // Направление движения
                                                                  private void Start()
 пули
    public GameObject bulletEffect;
                                                                     sentences = new Queue<string>();
    private void Start()
                                                                  public void StartDialogue(Dialogue dialogue)
      // Уничтожаем пулю через заданное время
      Destroy(gameObject, lifetime1);
                                                                     boxAnim.SetBool("boxOpen", true);
                                                                     startAnim.SetBool("startOpen", false);
    public void SetDirection(Vector2 dir)
                                                                     nameText.text = dialogue.name;
                                                                     sentences.Clear();
      direction = dir; // Устанавливаем направление
                                                                     foreach(string sentence in dialogue.sentences)
    private void Update()
                                                                       sentences.Enqueue(sentence);
      RaycastHit2D hitInfo =
                                                                     DisplayNextSentence();
 Physics2D.Raycast(transform.position, direction,
 distance, whatIsSolid);
      if (hitInfo.collider != null)
                                                                  public void DisplayNextSentence()
                                                                     if (sentences.Count == 0)
        if (hitInfo.collider.CompareTag("Enemy"))
```

```
EndDialogue();
                                                                 using System.Collections.Generic;
        if (x == 1)
                                                                 using UnityEngine;
         { test.gameObject.SetActive(true); }
                                                                 using UnityEngine.UI;
                                                                 public class taxt : MonoBehaviour
      string sentence = sentences.Dequeue();
                                                                   public Text dialogueText;
      StopAllCoroutines();
                                                                   private Queue<string> sentences;
                                                                   public Dialogue dialogue;
      StartCoroutine(TypeSentence(sentence));
                                                                   public Button dalee;
                                                                   public Button Wputi;
    IEnumerator TypeSentence(string sentence)
                                                                   private void Start()
      dialogueText.text = "";
      foreach(char letter in sentence.ToCharArray())
                                                                      sentences = new Queue<string>();
                                                                      StartDialogue(dialogue);
        dialogueText.text += letter;
        yield return null;
                                                                   public void StartDialogue(Dialogue dialogue)
                                                                      foreach (string sentence in dialogue.sentences)
    public void EndDialogue()
                                                                        sentences.Enqueue(sentence);
      boxAnim.SetBool("boxOpen", false);
                                                                      DisplayNextSentence();
 }
                                                                   public void DisplayNextSentence()
Файл DialogTrigger.cs:
 using System.Collections;
                                                                      if (sentences.Count == 0)
 using System.Collections.Generic;
 using UnityEngine;
                                                                        dalee.gameObject.SetActive(false);
                                                                        Wputi.gameObject.SetActive(true);
 public class DialogTrigger: MonoBehaviour
                                                                        return;
    public Dialogue dialogue;
                                                                      string sentence = sentences.Dequeue();
    public void TriggerDialogue()
                                                                      StopAllCoroutines();
                                                                      StartCoroutine(TypeSentence(sentence));
 FindObjectOfType<DialogManager>().StartDialogue(d
                                                                   IEnumerator TypeSentence(string sentence)
 ialogue);
    }
                                                                      dialogueText.text = "";
                                                                      foreach (char letter in sentence.ToCharArray())
Файл Dialogue.cs:
 using System.Collections;
                                                                        dialogueText.text += letter;
 using System.Collections.Generic;
                                                                        yield return null;
 using UnityEngine;
 [System.Serializable]
 public class Dialogue
                                                                 }
                                                               Файл hil.cs:
    public string name;
                                                                 using System.Collections;
    [TextArea(3, 10)]
                                                                 using System.Collections.Generic;
    public string[] sentences;
                                                                 using UnityEngine;
 }
                                                                 using System.IO;
                                                                 using UnityEngine.UI;
Файл taxt.cs:
 using System.Collections;
                                                                 public class hil: MonoBehaviour
```

```
public GameObject prajenie2;
    public Image a;
    public Image b;
                                                                    public int kol_vo = 0;
    public Text text;
    void Update()
                                                                    bool par=false;
      if (Input.GetKeyDown(KeyCode.F))
                                                                    private void Update()
        hiler();
                                                                       string[] stroka = File.ReadAllLines("номер
                                                                  принятого задания.txt");
      string[] stroka1 = File.ReadAllLines("зелья.txt");
      text.text = stroka1[0];
                                                                       string[] stroka1 =
                                                                  File.ReadAllLines("предметы.txt");
    private void hiler()
                                                                       if ((FindObjectOfType<Hp>().hp <= 0
                                                                  )&&(par==false))
      string[] stroka1 = File.ReadAllLines("зелья.txt");
 ((int.Parse(stroka1[0])>0)&&(FindObjectOfType<нр>(
                                                                         stroka1[4]=(int.Parse(stroka1[4]) -
 ).hp<2))
                                                                 kol_vo).ToString();
         FindObjectOfType<Hp>().hp += 1;
                                                                         StreamWriter f1 = new
         stroka1[0]= (int.Parse(stroka1[0])-1).ToString();
                                                                  StreamWriter("предметы.txt", false);
         StreamWriter f1 = new
                                                                         for (int k = 0; k < \text{stroka1.Length}; k++)
 StreamWriter("зелья.txt", false);
         for (int k = 0; k < \text{stroka1.Length}; k++)
                                                                            f1.WriteLine(stroka1[k]);
           f1.WriteLine(stroka1[k]);
                                                                         f1.Close();
         f1.Close();
                                                                         if (int.Parse(stroka[0]) == 6)
         if (FindObjectOfType<Hp>().hp == 1)
                                                                            prajenie1.gameObject.SetActive(true);
         { a.gameObject.SetActive(true); }
         if (FindObjectOfType<Hp>().hp==2)
                                                                            string[] stroka5 =
         { b.gameObject.SetActive(true); }
                                                                 File.ReadAllLines("карта.txt");
                                                                           if (stroka5[0] == "0")
       }
    }
 }
                                                                              imgObg1.sprite = spriteImage1;
Файл porajenie.cs:
                                                                            else
 using System.Collections;
 using System.Collections.Generic;
                                                                              imgObg1.sprite = spriteImage2;
                                                                              karta1.gameObject.SetActive(true);
 using UnityEngine;
 using UnityEngine.UI;
 using System.IO;
                                                                            StartCoroutine(stop());
 public class porajenie: MonoBehaviour
                                                                         }
    public Image a;
    public Image b;
                                                                            prajenie2.gameObject.SetActive(true);
    public Image imgObg1;
                                                                            StartCoroutine(stop());
    public Sprite spriteImage1;
    public Sprite spriteImage2;
                                                                       par = true;
    public Button karta1;
    public GameObject prajenie1;
                                                                    private IEnumerator stop()
```

```
if (int.Parse(stroka[0]) == 6)
      yield return new WaitForSeconds(0f);
      Time.timeScale = 0f;
                                                                        text[0].text = stroka1[4] + "/10";
    public void zanovo()
                                                                     if ((int.Parse(stroka[0]) == 6) \&\&
      par = false;
                                                                   (int.Parse(stroka1[4]) >= 10))
      prajenie1.SetActive(false);
      prajenie2.SetActive(false);
      karta1.gameObject.SetActive(false);
      FindObjectOfType<Hp>().hp = 2;
                                                                        stroka1[4] = (int.Parse(stroka1[4]) -
       { a.gameObject.SetActive(true); }
                                                                   10).ToString();
      { b.gameObject.SetActive(true); }
      Time.timeScale = 1f;
                                                                        StreamWriter f1 = new
    }
                                                                   StreamWriter("предметы.txt", false);
 }
                                                                        for (int k = 0; k < \text{stroka1.Length}; k++)
Файл zad_po_sbor.cs:
                                                                          f1.WriteLine(stroka1[k]);
 using System.Collections;
 using System.Collections.Generic;
                                                                        f1.Close();
 using System.IO;
 using UnityEngine;
                                                                        stroka[0] = "0";
 using UnityEngine.UI;
                                                                        StreamWriter f = new StreamWriter("номер
                                                                   принятого задания.txt", false);
 public class zad_po_sbor : MonoBehaviour
                                                                        for (int k = 0; k < \text{stroka.Length}; k++)
  public GameObject z1;
                                                                          f.WriteLine(stroka[k]);
 public Text[] text;
                                                                        f.Close();
 public Image imgObg;
 public Sprite spriteImage1;
 public Sprite spriteImage2;
                                                                        ////////
                                                                        string[] stroka2 =
 public Button karta;
                                                                   File.ReadAllLines("манеты.txt");
                                                                        stroka2[0] = (int.Parse(stroka2[0]) +
 public GameObject pobeda;
                                                                   300).ToString();
                                                                        StreamWriter f2 = new
 void Start()
                                                                   StreamWriter("манеты.txt", false);
    string[] stroka = File.ReadAllLines("номер
                                                                        for (int k = 0; k < \text{stroka2.Length}; k++)
 принятого задания.txt");
                                                                          f2.WriteLine(stroka2[k]);
    if (int.Parse(stroka[0]) == 6)
                                                                        f2.Close();
      z1.gameObject.SetActive(true);
                                                                        string[] stroka3 =
                                                                   File.ReadAllLines("выполнение зд.txt");
 }
                                                                        stroka3[5] = (int.Parse(stroka3[5]) + 1).ToString();
                                                                        StreamWriter f3 = new
 void Update()
                                                                   StreamWriter("выполнение зд.txt", false);
                                                                        for (int k = 0; k < \text{stroka3.Length}; k++)
    string[] stroka = File.ReadAllLines("номер
                                                                          f3.WriteLine(stroka3[k]);
 принятого задания.txt");
    string[] stroka1 =
                                                                        f3.Close();
 File.ReadAllLines("предметы.txt");
                                                                        string[] stroka4 = File.ReadAllLines("ранг.txt");
```

```
stroka4[0] = (int.Parse(stroka4[0]) + 3).ToString();
                                                                          btn[7].gameObject.SetActive(false);
      StreamWriter f4 = new StreamWriter("ранг.txt",
 false);
                                                                          if
      for (int k = 0; k < \text{stroka4.Length}; k++)
                                                                  ((int.Parse(stroka[0])>=1)&&((int.Parse(stroka[0]) <
                                                                  5)))
         f4.WriteLine(stroka4[k]);
                                                                          imgObg.sprite = spriteImage1[0];
                                                                         btn[0].gameObject.SetActive(true);
      f4.Close();
                                                                       if ((int.Parse(stroka[0]) >= 5) \&\&
      pobeda.gameObject.SetActive(true);
                                                                  ((int.Parse(stroka[0]) < 12)))
      string[] stroka5 = File.ReadAllLines("карта.txt");
                                                                         imgObg.sprite = spriteImage1[1];
      if (stroka5[0] == "0")
                                                                          btn[0].gameObject.SetActive(true);
                                                                          btn[1].gameObject.SetActive(true);
         imgObg.sprite = spriteImage1;
                                                                       if ((int.Parse(stroka[0]) >= 12) \&\&
      else
                                                                  ((int.Parse(stroka[0]) < 22)))
                                                                          imgObg.sprite = spriteImage1[2];
         imgObg.sprite = spriteImage2;
         karta.gameObject.SetActive(true);
                                                                          btn[0].gameObject.SetActive(true);
                                                                          btn[1].gameObject.SetActive(true);
                                                                          btn[2].gameObject.SetActive(true);
      StartCoroutine(stop());
                                                                       if ((int.Parse(stroka[0]) >= 22) \&\&
                                                                  ((int.Parse(stroka[0]) < 35)))
                                                                          imgObg.sprite = spriteImage1[3];
                                                                          btn[0].gameObject.SetActive(true);
 }
                                                                          btn[1].gameObject.SetActive(true);
                                                                          btn[2].gameObject.SetActive(true);
                                                                          btn[3].gameObject.SetActive(true);
 private IEnumerator stop()
    yield return new WaitForSeconds(1f);
                                                                       if ((int.Parse(stroka[0]) >= 35) \&\&
    Time.timeScale = 0f;
                                                                  ((int.Parse(stroka[0]) < 51)))
                                                                         imgObg.sprite = spriteImage1[4];
                                                                          btn[0].gameObject.SetActive(true);
                                                                          btn[1].gameObject.SetActive(true);
Файл dostun rang.cs:
                                                                          btn[2].gameObject.SetActive(true);
 using System.Collections;
                                                                          btn[3].gameObject.SetActive(true);
                                                                          btn[4].gameObject.SetActive(true);
 using System.Collections.Generic;
 using UnityEngine;
 using UnityEngine.UI;
                                                                       if ((int.Parse(stroka[0]) >= 51) \&\&
 using System.IO;
                                                                  ((int.Parse(stroka[0]) < 70)))
 public class dostun_rang : MonoBehaviour
                                                                          imgObg.sprite = spriteImage1[5];
                                                                          btn[0].gameObject.SetActive(true);
    public Button[] btn;
                                                                          btn[1].gameObject.SetActive(true);
    public Image imgObg;
                                                                          btn[2].gameObject.SetActive(true);
                                                                          btn[3].gameObject.SetActive(true);
    public Sprite[] spriteImage1;
                                                                          btn[4].gameObject.SetActive(true);
                                                                          btn[5].gameObject.SetActive(true);
    void Update()
      string[] stroka = File.ReadAllLines("ранг.txt");
                                                                       if (int.Parse(stroka[0]) > 70)
      if (int.Parse(stroka[0]) > 0)
                                                                          imgObg.sprite = spriteImage1[5];
                                                                          btn[0].gameObject.SetActive(true);
```

```
btn[1].gameObject.SetActive(true);
                                                                     btn[2].gameObject.SetActive(false);
        btn[2].gameObject.SetActive(true);
        btn[3].gameObject.SetActive(true);
                                                                     r=2;
        btn[4].gameObject.SetActive(true);
                                                                     imgObg[0].gameObject.SetActive(false);
        btn[5].gameObject.SetActive(true);
                                                                     imgObg[1].gameObject.SetActive(true);
        btn[6].gameObject.SetActive(true);
                                                                     imgObg[2].gameObject.SetActive(false);
                                                                     imgObg[3].gameObject.SetActive(false);
                                                                     imgObg[4].gameObject.SetActive(false);
    }
 }
                                                                     imgObg[5].gameObject.SetActive(false);
                                                                     imgObg[6].gameObject.SetActive(false);
Файл wbr ranga.cs:
 using System.Collections;
                                                                     string[] stroka = File.ReadAllLines("выполнение
 using System.Collections.Generic;
                                                                зд.txt");
 using UnityEngine;
                                                                     if (int.Parse(stroka[3]) == 0)
 using UnityEngine.UI;
 using System.IO;
                                                                        btn[0].gameObject.SetActive(true);
                                                                     if (int.Parse(stroka[4]) == 0)
 public class wbr_ranga: MonoBehaviour
                                                                        btn[1].gameObject.SetActive(true);
    public int r;
   public Image[] imgObg;
                                                                     if (int.Parse(stroka[5]) == 0)
   public Button[] btn;
                                                                        btn[2].gameObject.SetActive(true);
    public void RangF_clik()
      btn[0].gameObject.SetActive(false);
                                                                   public void RangD_clik()
      btn[1].gameObject.SetActive(false);
      btn[2].gameObject.SetActive(false);
                                                                     btn[0].gameObject.SetActive(false);
                                                                     btn[1].gameObject.SetActive(false);
                                                                     btn[2].gameObject.SetActive(false);
      r = 1;
      imgObg[0].gameObject.SetActive(true);
      imgObg[1].gameObject.SetActive(false);
                                                                     r = 3;
      imgObg[2].gameObject.SetActive(false);
                                                                     imgObg[0].gameObject.SetActive(false);
      imgObg[3].gameObject.SetActive(false);
                                                                     imgObg[1].gameObject.SetActive(false);
      imgObg[4].gameObject.SetActive(false);
                                                                     imgObg[2].gameObject.SetActive(true);
      imgObg[5].gameObject.SetActive(false);
                                                                     imgObg[3].gameObject.SetActive(false);
      imgObg[6].gameObject.SetActive(false);
                                                                     imgObg[4].gameObject.SetActive(false);
                                                                     imgObg[5].gameObject.SetActive(false);
                                                                     imgObg[6].gameObject.SetActive(false);
      string[] stroka = File.ReadAllLines("выполнение
 зд.txt");
      if (int.Parse(stroka[0]) == 0)
                                                                     string[] stroka = File.ReadAllLines("выполнение
                                                                зд.txt");
        btn[0].gameObject.SetActive(true);
                                                                     if (int.Parse(stroka[6]) == 0)
      if (int.Parse(stroka[1]) == 0)
                                                                        btn[0].gameObject.SetActive(true);
        btn[1].gameObject.SetActive(true);
                                                                     if (int.Parse(stroka[7]) == 0)
      if (int.Parse(stroka[2]) == 0)
                                                                        btn[1].gameObject.SetActive(true);
         btn[2].gameObject.SetActive(true);
                                                                     if (int.Parse(stroka[8]) == 0)
                                                                        btn[2].gameObject.SetActive(true);
   public void RangE clik()
                                                                   public void RangC_clik()
      btn[0].gameObject.SetActive(false);
      btn[1].gameObject.SetActive(false);
                                                                     btn[0].gameObject.SetActive(false);
```

```
btn[0].gameObject.SetActive(false);
    btn[1].gameObject.SetActive(false);
    btn[2].gameObject.SetActive(false);
                                                                   btn[1].gameObject.SetActive(false);
                                                                   btn[2].gameObject.SetActive(false);
    r = 4;
    imgObg[0].gameObject.SetActive(false);
    imgObg[1].gameObject.SetActive(false);
                                                                   imgObg[0].gameObject.SetActive(false);
    imgObg[2].gameObject.SetActive(false);
                                                                   imgObg[1].gameObject.SetActive(false);
    imgObg[3].gameObject.SetActive(true);
                                                                   imgObg[2].gameObject.SetActive(false);
    imgObg[4].gameObject.SetActive(false);
                                                                   imgObg[3].gameObject.SetActive(false);
    imgObg[5].gameObject.SetActive(false);
                                                                   imgObg[4].gameObject.SetActive(false);
    imgObg[6].gameObject.SetActive(false);
                                                                   imgObg[5].gameObject.SetActive(true);
                                                                   imgObg[6].gameObject.SetActive(false);
    string[] stroka = File.ReadAllLines("выполнение
                                                                   string[] stroka = File.ReadAllLines("выполнение
зд.txt");
    if (int.Parse(stroka[9]) == 0)
                                                               зд.txt");
                                                                   if (int.Parse(stroka[15]) == 0)
       btn[0].gameObject.SetActive(true);
                                                                      btn[0].gameObject.SetActive(true);
    if (int.Parse(stroka[10]) == 0)
                                                                   if (int.Parse(stroka[16]) == 0)
       btn[1].gameObject.SetActive(true);
                                                                      btn[1].gameObject.SetActive(true);
    if (int.Parse(stroka[11]) == 0)
                                                                   if (int.Parse(stroka[17]) == 0)
       btn[2].gameObject.SetActive(true);
                                                                      btn[2].gameObject.SetActive(true);
  public void RangB_clik()
    btn[0].gameObject.SetActive(false);
                                                                 public void RangS clik()
    btn[1].gameObject.SetActive(false);
    btn[2].gameObject.SetActive(false);
                                                                   btn[0].gameObject.SetActive(false);
                                                                   btn[1].gameObject.SetActive(false);
                                                                   btn[2].gameObject.SetActive(false);
    imgObg[0].gameObject.SetActive(false);
    imgObg[1].gameObject.SetActive(false);
                                                                   r = 7;
    imgObg[2].gameObject.SetActive(false);
                                                                   imgObg[0].gameObject.SetActive(false);
    imgObg[3].gameObject.SetActive(false);
                                                                   imgObg[1].gameObject.SetActive(false);
    imgObg[4].gameObject.SetActive(true);
                                                                   imgObg[2].gameObject.SetActive(false);
    imgObg[5].gameObject.SetActive(false);
                                                                   imgObg[3].gameObject.SetActive(false);
    imgObg[6].gameObject.SetActive(false);
                                                                   imgObg[4].gameObject.SetActive(false);
                                                                   imgObg[5].gameObject.SetActive(false);
     string[] stroka = File.ReadAllLines("выполнение
                                                                   imgObg[6].gameObject.SetActive(true);
зд.txt");
    if (int.Parse(stroka[12]) == 0)
                                                                   string[] stroka = File.ReadAllLines("выполнение
                                                               зд.txt");
       btn[0].gameObject.SetActive(true);
                                                                   if (int.Parse(stroka[18]) == 0)
    if (int.Parse(stroka[13]) == 0)
                                                                      btn[0].gameObject.SetActive(true);
       btn[1].gameObject.SetActive(true);
                                                                   if (int.Parse(stroka[19]) == 0)
    if (int.Parse(stroka[14]) == 0)
                                                                      btn[1].gameObject.SetActive(true);
                                                                   if (int.Parse(stroka[20]) == 0)
       btn[2].gameObject.SetActive(true);
                                                                      btn[2].gameObject.SetActive(true);
  public void RangA_clik()
```

```
perehod.levelToLoad = 5;
Файл Zadanija.cs:
 using System.Collections;
                                                                      if (FindObjectOfType<wbr_ranga>().r == 4)
 using System.Collections.Generic;
 using UnityEngine;
                                                                        perehod.levelToLoad = 5;
 using UnityEngine.UI;
 using System.IO;
                                                                      if (FindObjectOfType<wbr_ranga>().r == 5)
 public class Zadanija: MonoBehaviour
                                                                        perehod.levelToLoad = 5;
                                                                      if (FindObjectOfType<wbr_ranga>().r == 6)
    public Image imgObg;
    public Sprite[] spriteImage1;
                                                                        perehod.levelToLoad = 5;
    public Переход perehod;
                                                                      if (FindObjectOfType<wbr_ranga>().r == 7)
    public void Z1_clic()
                                                                        perehod.levelToLoad = 5;
      if (FindObjectOfType<wbr_ranga>().r == 1)
         perehod.levelToLoad = 13;
                                                                      int s = 0;
                                                                      for (int i = 1; i < 8; i++)
        string[] stroka = File.ReadAllLines("номер
 принятого задания.txt");
                                                                        if (FindObjectOfType < wbr ranga > ().r == i)
        stroka[0]=1.ToString();
        StreamWriter f1 = new StreamWriter("номер
                                                                          imgObg.sprite = spriteImage1[s];
 принятого задания.txt", false);
        for (int k = 0; k < \text{stroka.Length}; k++)
                                                                        s += 3;
           f1.WriteLine(stroka[k]);
                                                                      imgObg.gameObject.SetActive(true);
        f1.Close();
        string[] stroka1 =
                                                                   public void Z2_clic()
 File.ReadAllLines("посещённые локации.txt");
        stroka1[0] = 1.ToString();
                                                                      if (FindObjectOfType<wbr_ranga>().r == 1)
        StreamWriter f11 = new
 StreamWriter("посещённые локации.txt", false);
                                                                        perehod.levelToLoad = 12;
        for (int k = 0; k < \text{stroka1.Length}; k++)
           f11.WriteLine(stroka1[k]);
                                                                      if (FindObjectOfType<wbr_ranga>().r == 2)
        f11.Close();
                                                                        perehod.levelToLoad = 14;
      if (FindObjectOfType<wbr_ranga>().r == 2)
                                                                        string[] stroka = File.ReadAllLines("номер
        perehod.levelToLoad = 14;
                                                                 принятого задания.txt");
        string[] stroka = File.ReadAllLines("номер
                                                                        stroka[0] = 5.ToString();
 принятого задания.txt");
                                                                        StreamWriter f1 = new StreamWriter("номер
        stroka[0] = 4.ToString();
                                                                 принятого задания.txt", false);
         StreamWriter f1 = new StreamWriter("номер
                                                                        for (int k = 0; k < \text{stroka.Length}; k++)
 принятого задания.txt", false);
        for (int k = 0; k < \text{stroka.Length}; k++)
                                                                          f1.WriteLine(stroka[k]);
                                                                        f1.Close();
           f1.WriteLine(stroka[k]);
                                                                      if (FindObjectOfType<wbr_ranga>().r == 3)
        f1.Close();
      if (FindObjectOfType<wbr_ranga>().r == 3)
                                                                        perehod.levelToLoad = 5;
```

```
if (FindObjectOfType<wbr_ranga>().r == 2)
    if (FindObjectOfType<wbr ranga>().r == 4)
                                                                      perehod.levelToLoad = 15;
       perehod.levelToLoad = 5;
                                                                      string[] stroka = File.ReadAllLines("номер
    if (FindObjectOfType<wbr_ranga>().r == 5)
                                                               принятого задания.txt");
                                                                      stroka[0] = 6.ToString();
       perehod.levelToLoad = 5;
                                                                      StreamWriter f1 = new StreamWriter("номер
                                                               принятого задания.txt", false);
    if (FindObjectOfType<wbr_ranga>().r == 6)
                                                                      for (int k = 0; k < \text{stroka.Length}; k++)
       perehod.levelToLoad = 5;
                                                                         f1.WriteLine(stroka[k]);
    if (FindObjectOfType<wbr_ranga>().r == 7)
                                                                      f1.Close();
       perehod.levelToLoad = 5;
                                                                      string[] stroka1 =
                                                               File.ReadAllLines("посещённые локации.txt");
                                                                      stroka1[1] = 1.ToString();
    int s = 1;
                                                                      StreamWriter f11 = new
    for (int i = 1; i < 8; i++)
                                                               StreamWriter("посещённые локации.txt", false);
                                                                      for (int k = 0; k < \text{stroka1.Length}; k++)
       if (FindObjectOfType<wbr_ranga>().r == i)
                                                                         f11.WriteLine(stroka1[k]);
         imgObg.sprite = spriteImage1[s];
                                                                      f11.Close();
       s += 3;
                                                                    if (FindObjectOfType<wbr_ranga>().r == 3)
    imgObg.gameObject.SetActive(true);
                                                                      perehod.levelToLoad = 5;
                                                                    if (FindObjectOfType<wbr_ranga>().r == 4)
  public void Z3 clic()
                                                                      perehod.levelToLoad = 5;
    if (FindObjectOfType<wbr_ranga>().r == 1)
                                                                    if (FindObjectOfType<wbr ranga>().r == 5)
       perehod.levelToLoad = 13;
                                                                      perehod.levelToLoad = 5;
       string[] stroka = File.ReadAllLines("номер
принятого задания.txt");
                                                                    if (FindObjectOfType<wbr_ranga>().r == 6)
       stroka[0] = 2.ToString();
       StreamWriter f1 = new StreamWriter("номер
                                                                      perehod.levelToLoad = 5;
принятого задания.txt", false);
                                                                    if (FindObjectOfType<wbr_ranga>().r == 7)
       for (int k = 0; k < \text{stroka.Length}; k++)
                                                                      perehod.levelToLoad = 5;
         f1.WriteLine(stroka[k]);
       f1.Close();
                                                                    int s = 2;
       string[] stroka1 =
                                                                    for (int i = 1; i < 8; i++)
File.ReadAllLines("посещённые локации.txt");
       stroka1[0] = 1.ToString();
                                                                      if (FindObjectOfType<wbr_ranga>().r == i)
       StreamWriter f11 = new
                                                                         imgObg.sprite = spriteImage1[s];
StreamWriter("посещённые локации.txt", false);
       for (int k = 0; k < \text{stroka1.Length}; k++)
                                                                      s += 3;
         f11.WriteLine(stroka1[k]);
       f11.Close();
                                                                    imgObg.gameObject.SetActive(true);
```

```
Файл Draggable1.cs:
   public void kr()
                                                                using System.Collections;
                                                                using System.Collections.Generic;
      imgObg.gameObject.SetActive(false);
                                                                using UnityEngine;
                                                                using System.IO;
                                                                using UnityEngine.EventSystems;
                                                                public class Draggable1: MonoBehaviour,
Файл kniga.cs:
                                                                IDragHandler, IBeginDragHandler
 using System.Collections;
 using System.Collections.Generic;
                                                                {
 using UnityEngine;
                                                                   private Canvas canvas;
                                                                   private RectTransform rectTransform;
 using UnityEngine.UI;
 using System.IO;
                                                                   private Vector3 originalPosition;
 public class kniga: MonoBehaviour
                                                                   private Vector2 offset;
                                                                   void Awake()
    public GameObject[] scrol;
   public Image img;
                                                                     rectTransform =
   public Sprite[] sprites;
                                                                GetComponent<RectTransform>();
   public Image fon;
                                                                     canvas = GetComponentInParent<Canvas>();
   public Sprite[] sprites_fon;
                                                                     originalPosition =
   public Button exit;
                                                                rectTransform.anchoredPosition;
   public Button exit1;
    void Start()
                                                                   }
      string[] stroka =
                                                                   public void OnBeginDrag(PointerEventData
 File.ReadAllLines("использовать.txt");
                                                                eventData)
      if (int.Parse(stroka[0])==0)
                                                                     // Сохранение оригинальной позиции при
                                                                начале перетаскивания
        fon.sprite = sprites fon[0];
        exit1.gameObject.SetActive(true);
        img.gameObject.SetActive(false);
                                                                     // Сохраняем смещение между курсором и
                                                                объектом
      if (int.Parse(stroka[0]) == 1)
                                                                     RectTransform rectTransform =
                                                                GetComponent<RectTransform>();
        fon.sprite = sprites_fon[1];
                                                                     originalPosition =
        exit.gameObject.SetActive(true);
                                                                rectTransform.anchoredPosition;
        scrol[0].gameObject.SetActive(true);
                                                                     offset = eventData.position -
        img.sprite = sprites[3];
                                                                Rect Transform Utility. World To Screen Point (event Data.\\
                                                                pressEventCamera, rectTransform.position);
      if (int.Parse(stroka[0]) == 2)
         fon.sprite = sprites_fon[2];
                                                                   public void OnDrag(PointerEventData eventData)
        exit.gameObject.SetActive(true);
        scrol[1].gameObject.SetActive(true);
                                                                     Vector2 position;
        img.sprite = sprites[0];
                                                                RectTransformUtility.ScreenPointToLocalPointInRecta
    }
                                                                ngle(canvas.transform as RectTransform,
                                                                eventData.position, canvas.worldCamera, out position);
    public void button_click(int i)
      img.sprite = sprites[i];
                                                                     Vector2 newPosition = eventData.position - offset;
                                                                     RectTransform rectTransform =
                                                                GetComponent<RectTransform>();
                                                                     rectTransform.position = newPosition;
 }
                                                                   }
```

```
public Vector3 GetCurrentPosition()
                                                                    Vector2 newPosition = eventData.position - offset;
      return rectTransform.anchoredPosition;
                                                                    RectTransform rectTransform =
                                                               GetComponent<RectTransform>();
 }
                                                                    rectTransform.position = newPosition;
Файл Draggable2.cs:
 using System.Collections;
                                                                 public Vector3 GetCurrentPosition()
 using System.Collections.Generic;
 using UnityEngine;
                                                                   return rectTransform.anchoredPosition;
 using System.IO;
 using UnityEngine.EventSystems;
 public class Draggable2: MonoBehaviour,
                                                             Файл JewelryManager1.cs:
 IDragHandler, IBeginDragHandler
                                                               using System;
                                                               using System.Collections;
 {
                                                               using System.Collections.Generic;
    private Canvas canvas;
                                                               using System.IO;
   private RectTransform rectTransform;
                                                               using UnityEngine;
                                                               using UnityEngine.EventSystems;
    private Vector3 originalPosition;
                                                               using UnityEngine.UI;
                                                               using static System.Net.Mime.MediaTypeNames;
    private Vector2 offset;
    void Awake()
                                                               public class JewelryManager1: MonoBehaviour
      rectTransform =
 GetComponent<RectTransform>();
                                                                 public GameObject[] jewelryPrefabs; // массив
      canvas = GetComponentInParent<Canvas>();
                                                               префабов
      originalPosition =
                                                                 public Button[] buttons;
                                                                 private Dictionary<int, int> quantities = new
 rectTransform.anchoredPosition;
                                                               Dictionary<int, int>();
                                                                 public Transform parentTransform;
    }
                                                                 public Vector2 spawnPosition;
    public void OnBeginDrag(PointerEventData
                                                                 public List<Draggable1> draggableJewelry = new
 eventData)
                                                               List<Draggable1>();
      // Сохранение оригинальной позиции при
 начале перетаскивания
                                                                 void Start()
                                                                    LoadQuantities();
                                                                    LoadJewelryPositions();
      // Сохраняем смещение между курсором и
                                                                    for (int i = 0; i < buttons.Length; i++)
 объектом
      RectTransform rectTransform =
 GetComponent<RectTransform>();
                                                                      int index = i;
                                                                      buttons[i].onClick.AddListener(() =>
      originalPosition =
 rectTransform.anchoredPosition:
                                                               OnJewelryButtonClick(index));
      offset = eventData.position -
 RectTransformUtility.WorldToScreenPoint(eventData.
                                                                 }
 pressEventCamera, rectTransform.position);
                                                                 void Update()
    public void OnDrag(PointerEventData eventData)
                                                                    // Проверяем, нажата ли клавиша Backspace
                                                                    if (Input.GetKeyDown(KeyCode.Backspace))
      Vector2 position;
                                                                      RemoveSelectedJewelry();
 Rect Transform Utility. Screen Point To Local Point In Recta \\
 ngle(canvas.transform as RectTransform,
```

eventData.position, canvas.worldCamera, out position);

```
if (quantities.ContainsKey(index) &&
                                                               quantities[index] > 0)
  void LoadQuantities()
                                                                      GameObject jewelryInstance =
                                                               Instantiate(jewelryPrefabs[index], parentTransform);
    string[] lines = File.ReadAllLines("растения.txt");
    for (int i = 0; i < lines.Length; i++)
                                                               jewelryInstance.GetComponent<RectTransform>().anc
                                                               horedPosition = spawnPosition;
                                                                      quantities[index]--; // Уменьшаем количество
       if (int.TryParse(lines[i], out int quantity))
                                                                      UpdateQuantityFile();
         quantities[i] = quantity;
                                                               jewelryInstance.AddComponent<Draggable1>();
                                                               draggableJewelry.Add(jewelryInstance.GetComponent
  }
                                                               <Draggable1>());
                                                                    }
  void LoadJewelryPositions()
    if (File.Exists("positions_kotel.txt"))
                                                                 void UpdateQuantityFile()
                                                                    using (StreamWriter writer = new
       string[] lines =
File.ReadAllLines("positions_kotel.txt");
                                                               StreamWriter("растения.txt"))
       foreach (string line in lines)
                                                                      foreach (var quantity in quantities)
         string[] parts = line.Split('|');
         if (parts.Length == 3)
                                                                         writer.WriteLine(quantity.Value);
            string name = parts[0].Substring(0,
parts[0].Length - 7);
            float x = \text{float.Parse(parts[1])};
            float y = float.Parse(parts[2]);
                                                                 private void OnApplicationQuit()
                                                                    SavePositions(); // Сохраняем позиции при
            GameObject prefab =
                                                               выходе из приложения
System.Array.Find(jewelryPrefabs, p => p.name ==
name);
            if (prefab != null)
                                                                 public void SavePositions()
              GameObject jewelryInstance =
                                                                    File.WriteAllText("positions_kotel.txt",
Instantiate(prefab, parentTransform);
                                                               string.Empty);
                                                                    foreach (var draggable in draggableJewelry)
jewelryInstance.GetComponent<RectTransform>().anc
horedPosition = new Vector2(x, y);
                                                                      if (draggable != null)
                                                                         string positionData =
                                                               $"{draggable.gameObject.name}|{draggable.GetCurren
jewelryInstance.AddComponent<Draggable1>(); //
добавляем компонент для перетаскивания
                                                               tPosition().x}|{draggable.GetCurrentPosition().y}\n"; //
                                                               Добавляем размер
draggableJewelry.Add(jewelryInstance.GetComponent
                                                                         File.AppendAllText("positions_kotel.txt",
<Draggable1>()); // добавляем в список
                                                               positionData); // Сохраняем позицию в файл
                                                                    draggableJewelry.Clear(); // Очищаем список
                                                               после сохранения
  public void OnJewelryButtonClick(int index)
                                                                 private void RemoveSelectedJewelry()
```

```
PointerEventData pointerData = new
PointerEventData(EventSystem.current) { position =
                                                                          if
Input.mousePosition };
                                                              ((Math.Round(draggable.GetCurrentPosition().x, 2) ==
    List<RaycastResult> results = new
                                                              528.8) &&
List<RaycastResult>();
                                                              (Math.Round(draggable.GetCurrentPosition().y, 2) ==
    EventSystem.current.RaycastAll(pointerData,
                                                              23.3))
results);
                                                                             stroka1[3] = 0.ToString();
    foreach (RaycastResult result in results)
                                                              ((Math.Round(draggable.GetCurrentPosition().x, 2) ==
       Draggable1 draggable =
result.gameObject.GetComponent<Draggable1>();
                                                              712.8) &&
       if (draggable != null)
                                                              (Math.Round(draggable.GetCurrentPosition().y, 2) ==
                                                              23.4))
         int index =
GetJewelryIndex(draggable.gameObject.name);
                                                                             stroka1[4] = 0.ToString();
         if (index !=-1)
            // Увеличиваем количество в словаре
                                                              ((Math.Round(draggable.GetCurrentPosition().x, 2) ==
            quantities[index]++;
                                                              437) &&
                                                              (Math.Round(draggable.GetCurrentPosition().y, 2) == -
            // Обновляем файл после изменения
                                                              107.5))
количества
            UpdateQuantityFile();
                                                                             stroka1[5] = 0.ToString();
            // Удаляем объект из списка
                                                                          if
draggableJewelry и уничтожаем его
                                                              ((Math.Round(draggable.GetCurrentPosition().x, 2) ==
            draggableJewelry.Remove(draggable);
                                                              627) &&
                                                              (Math.Round(draggable.GetCurrentPosition().y, 2) == -
                                                              107.6))
            string[] stroka1 =
File.ReadAllLines("крафт.txt");
                                                                             stroka1[6] = 0.ToString();
            if
                                                                          StreamWriter f = new
((Math.Round(draggable.GetCurrentPosition().x,2)==
                                                              StreamWriter("крафт.txt", false);
                                                                          for (int i = 0; i < stroka1.Length; i++)
437)&&
(Math.Round(draggable.GetCurrentPosition().y,2) ==
146))
                                                                             f.WriteLine(stroka1[i]);
              stroka1[0] = 0.ToString();
                                                                           f.Close();
                                                                          Destroy(draggable.gameObject);
((Math.Round(draggable.GetCurrentPosition().x, 2) ==
625.2) &&
(Math.Round(draggable.GetCurrentPosition().y, 2) ==
144.8))
                                                                 private int GetJewelryIndex(string name)
              stroka1[1] = 0.ToString();
                                                                   for (int i = 0; i < jewelryPrefabs.Length; <math>i++)
                                                                      if (name.StartsWith("pact" + (i +
            if
((Math.Round(draggable.GetCurrentPosition().x, 2) ==
                                                              1).ToString()+ "(Clone)"))
351.3) &&
(Math.Round(draggable.GetCurrentPosition().y, 2) ==
                                                                        return i;
23.9))
              stroka1[2] = 0.ToString();
                                                                   return -1;
```

```
string[] lines =
                                                                File.ReadAllLines("positions kotel krist.txt");
                                                                        foreach (string line in lines)
 }
                                                                          string[] parts = line.Split('|');
                                                                          if (parts.Length == 3)
Файл JewelryManager2.cs:
                                                                            string name = parts[0].Substring(0,
 using System;
                                                                parts[0].Length - 7); // обрезаем "(Clone)"
 using System.Collections;
 using System.Collections.Generic;
                                                                            float x = \text{float.Parse(parts[1])};
                                                                            float y = \text{float.Parse(parts[2])};
 using System.IO;
 using UnityEngine:
                                                                            if (jewelryPrefab.name == name)
 using UnityEngine.EventSystems;
 using UnityEngine.UI;
                                                                               GameObject jewelryInstance =
 using static System.Net.Mime.MediaTypeNames;
                                                                Instantiate(jewelryPrefab, parentTransform);
 public class JewelryManager2: MonoBehaviour
                                                                jewelryInstance.GetComponent<RectTransform>().anc
                                                                horedPosition = new Vector2(x, y);
    public GameObject jewelryPrefab; // единственный
 префаб
                                                                jewelryInstance.AddComponent<Draggable2>();
    public Button button; // единственная кнопка
    private int quantity; // количество
                                                                draggableJewelry.Add(jewelryInstance.GetComponent
   public Transform parentTransform;
                                                                <Draggable2>());
   public Vector2 spawnPosition;
    private List<Draggable2> draggableJewelry = new
 List<Draggable2>();
    void Start()
                                                                   }
      LoadQuantity();
                                                                   public void OnJewelryButtonClick()
      LoadJewelryPositions();
                                                                     if (quantity > 0)
 button.onClick.AddListener(OnJewelryButtonClick);
                                                                        GameObject jewelryInstance =
                                                                Instantiate(jewelryPrefab, parentTransform);
    void Update()
                                                                jewelryInstance.GetComponent<RectTransform>().anc
      if (Input.GetKeyDown(KeyCode.Backspace))
                                                                horedPosition = spawnPosition;
                                                                        quantity--;
                                                                        UpdateQuantityFile();
         RemoveSelectedJewelry();
                                                                jewelryInstance.AddComponent<Draggable2>();
    void LoadQuantity()
                                                                draggableJewelry.Add(jewelryInstance.GetComponent
                                                                <Draggable2>());
      string[] lines =
 File.ReadAllLines("предметы.txt");
      if (lines.Length >= 5 && int.TryParse(lines[4],
 out quantity)) // берем 5 строку
                                                                   void UpdateQuantityFile()
         Debug.Log($"Quantity loaded: {quantity}");
                                                                     var lines = File.ReadAllLines("предметы.txt");
                                                                     if (lines.Length >= 5)
                                                                        lines[4] = quantity.ToString(); // обновляем 5
    void LoadJewelryPositions()
                                                                строку
                                                                       File.WriteAllLines("предметы.txt", lines);
      if (File.Exists("positions_kotel_krist.txt"))
```

```
private void OnApplicationQuit()
                                                               625.2) &&
                                                               (Math.Round(draggable.GetCurrentPosition().y, 2) ==
     SavePositions();
                                                               144.8))
                                                                           stroka1[1] = 0.ToString();
  public void SavePositions()
     using (StreamWriter writer = new
                                                               ((Math.Round(draggable.GetCurrentPosition().x, 2) ==
StreamWriter("positions_kotel_krist.txt"))
                                                               351.3) &&
                                                               (Math.Round(draggable.GetCurrentPosition().y, 2) ==
       foreach (var draggable in draggableJewelry)
                                                               23.9))
          if (draggable != null)
                                                                           stroka1[2] = 0.ToString();
                                                                         if
            string positionData =
$"{draggable.gameObject.name}|{draggable.GetCurren
                                                               ((Math.Round(draggable.GetCurrentPosition().x, 2) ==
tPosition().x}|{draggable.GetCurrentPosition().y}\n";
                                                               528.8) &&
            writer.Write(positionData);
                                                               (Math.Round(draggable.GetCurrentPosition().y, 2) ==
                                                               23.3))
                                                                           stroka1[3] = 0.ToString();
     draggableJewelry.Clear();
                                                               ((Math.Round(draggable.GetCurrentPosition().x, 2) ==
  private void RemoveSelectedJewelry()
                                                               712.8) &&
                                                               (Math.Round(draggable.GetCurrentPosition().y, 2) ==
     PointerEventData pointerData = new
                                                               23.4))
PointerEventData(EventSystem.current) { position =
Input.mousePosition };
                                                                           stroka1[4] = 0.ToString();
     List<RaycastResult> results = new
List<RaycastResult>();
     Event System. current. Ray cast All (pointer Data,\\
                                                               ((Math.Round(draggable.GetCurrentPosition().x, 2) ==
results);
     foreach (RaycastResult result in results)
                                                               (Math.Round(draggable.GetCurrentPosition().y, 2) == -
                                                               107.5))
       Draggable 2 draggable =
result.gameObject.GetComponent<Draggable2>();
                                                                           stroka1[5] = 0.ToString();
       if (draggable != null)
          quantity++; // увеличиваем количество
                                                               ((Math.Round(draggable.GetCurrentPosition().x, 2) ==
          UpdateQuantityFile();
                                                               627) &&
          draggableJewelry.Remove(draggable);
                                                               (Math.Round(draggable.GetCurrentPosition().y, 2) == -
                                                               107.6))
          string[] stroka1 =
File.ReadAllLines("крафт.txt");
                                                                           stroka1[6] = 0.ToString();
          if
                                                                         StreamWriter f = new
((Math.Round(draggable.GetCurrentPosition().x, 2) ==
                                                               StreamWriter("κραφτ.txt", false);
                                                                         for (int i = 0; i < \text{stroka1.Length}; i++)
(Math.Round(draggable.GetCurrentPosition().y, 2) ==
146))
                                                                           f.WriteLine(stroka1[i]);
            stroka1[0] = 0.ToString();
                                                                         f.Close();
                                                                         Destroy(draggable.gameObject);
                                                                         break;
          if
((Math.Round(draggable.GetCurrentPosition().x, 2) ==
```

```
== "Slot4") && (int.Parse(stroka[3]) ==0)) || ((y ==
                                                                                                                                "Slot5") && (int.Parse(stroka[4]) == 0)) \| ((y ==
                                                                                                                                "Slot6") && (int.Parse(stroka[5]) == 0)) \| ((y ==
Файл scitkolrastenij.cs:
                                                                                                                                "Slot7") && (int.Parse(stroka[6]) == 0)))
   using System.Collections;
                                                                                                                                              {
   using System.Collections.Generic;
   using UnityEngine;
   using UnityEngine.UI;
                                                                                                                                                   // Устанавливаем позицию элемента в
   using System.IO;
                                                                                                                               заданные координаты (targetX, targetY)
                                                                                                                                                   RectTransform itemRectTransform =
   public class scitkolrastenij: MonoBehaviour
                                                                                                                                item.GetComponent<RectTransform>();
                                                                                                                                                   item Rect Transform. anchored Position = new \\
       public Text[] text;
                                                                                                                                Vector2(targetX, targetY);
        void Update()
                                                                                                                                                   if (y == "Slot1")
                                                                                                                                                       if (item.name == "pact9(Clone)")
            string[] stroka =
   File.ReadAllLines("растения.txt");
            string[] stroka1 =
                                                                                                                                                            x = 1;
  File.ReadAllLines("предметы.txt");
                                                                                                                                                       else if (item.name == "pact13(Clone)")
            for (int i = 0; i < \text{stroka.Length}; i++)
                                                                                                                                                        \{ x = 2; \}
                                                                                                                                                       else if (item.name == "pact2(Clone)")
                 text[i].text = stroka[i];
                                                                                                                                                        \{ x = 3; \}
                                                                                                                                                       else
            text[15].text = stroka1[4];
                                                                                                                                                        \{ x = -1; \}
   }
                                                                                                                                                        stroka[0] = x.ToString();
Файл Slot.cs:
                                                                                                                                                   if (y == "Slot2")
   using System.Collections;
  using System.Collections.Generic;
   using System.IO:
                                                                                                                                                       if (item.name == "pact14(Clone)")
   using Unity. Visual Scripting;
   using UnityEngine;
                                                                                                                                                            x = 1;
   using UnityEngine.EventSystems;
   using static UnityEngine.RuleTile.TilingRuleOutput;
                                                                                                                                                       else if (item.name == "pact4(Clone)")
                                                                                                                                                        \{ x = 2; \}
   public class Slot: MonoBehaviour, IDropHandler
                                                                                                                                                       else if (item.name == "pact15(Clone)")
                                                                                                                                                        \{ x = 3; \}
       public float targetX; // Задайте значение х
                                                                                                                                                       else
        public float targetY; // Задайте значение у
                                                                                                                                                       \{ \mathbf{x} = -1; \}
       public string y;
       public int x;
                                                                                                                                                       stroka[1] = x.ToString();
        public void OnDrop(PointerEventData eventData)
            string[] stroka = File.ReadAllLines("κραφτ.txt");
                                                                                                                                                  if (y == "Slot3")
            Draggable1 item =
                                                                                                                                                       if (item.name == "pact3(Clone)")
   eventData.pointerDrag.GetComponent<Draggable1>();
                                                                                                                                                            x = 1;
             Draggable2 item2 =
   eventData.pointerDrag.GetComponent<Draggable2>();
                                                                                                                                                       else
                                                                                                                                                       \{ x = -1; \}
            if (item != null)
                                                                                                                                                        stroka[2] = x.ToString();
                 if (((y == "Slot1") && (int.Parse(stroka[0]) ==
   0)) \| ((y == "Slot2") && (int.Parse(stroka[1]) == 0)) \|
   ((y == "Slot3") \&\& (int.Parse(stroka[2]) == 0)) || ((y == "Slot3") \&\& (int.Parse(stroka[2]) == 0)) || ((y == "Slot3") && (int.Parse(stroka[2]) == 0) || ((y == "Slot3") && (int.Parse(stroka[2]) == 0) || ((y == "Slot3") && (int.Parse(stroka[2]) == 0) || ((y == "Slot3") && ((int.Parse(stroka[2]) && ((i
                                                                                                                                                  if (y == "Slot4")
```

```
if (item.name == "pact(Clone)")
                                                                                                                                                                             if (((y == "Slot1") && (int.Parse(stroka[0]) ==
                                                                                                                                                           0)) \| ((y == "Slot2") && (int.Parse(stroka[1]) == 0)) \|
                                   x = 1;
                                                                                                                                                           ((y == "Slot3") && (int.Parse(stroka[2]) == 0)) || ((y == "Slot3") && (int.Parse(stroka[2]) == 0) || ((y == "Slot3") && ((y == "Slot
                                                                                                                                                           == "Slot4") && (int.Parse(stroka[3]) == 0)) || ((y ==
                              else
                              \{ x = -1; \}
                                                                                                                                                           "Slot5") && (int.Parse(stroka[4]) == 0)) \| ((y ==
                                                                                                                                                           "Slot6") && (int.Parse(stroka[5]) == 0)) \| ((y ==
                                                                                                                                                           "Slot7") && (int.Parse(stroka[6]) == 0)))
                              stroka[3] = x.ToString();
                                                                                                                                                                             {
                        if (y == "Slot5")
                                                                                                                                                                                   // Устанавливаем позицию элемента в
                              if (item.name == "pact7(Clone)")
                                                                                                                                                           заданные координаты (targetX, targetY)
                                                                                                                                                                                   RectTransform itemRectTransform2 =
                                   x = 1;
                                                                                                                                                           item2.GetComponent<RectTransform>();
                                                                                                                                                                                   itemRectTransform2.anchoredPosition = new
                              else if (item.name == "pacт6(Clone)")
                                                                                                                                                           Vector2(targetX, targetY);
                              else if (item.name == "pact1(Clone)")
                                                                                                                                                                                   if (y == "Slot1")
                              \{ x = 3; \}
                              else
                                                                                                                                                                                          x = -1;
                              \{ \mathbf{x} = -1; \}
                                                                                                                                                                                         stroka[0] = x.ToString();
                              stroka[4] = x.ToString();
                        if (y == "Slot6")
                                                                                                                                                                                   if (y == "Slot2")
                              if (item.name == "pact10(Clone)")
                                                                                                                                                                                        x = -1;
                                                                                                                                                                                         stroka[1] = x.ToString();
                                   x = 1;
                              else
                              \{ x = -1; \}
                                                                                                                                                                                   if (y == "Slot3")
                              stroka[5] = x.ToString();
                                                                                                                                                                                        x = -1;
                        if (y == "Slot7")
                                                                                                                                                                                         stroka[2] = x.ToString();
                              if (item.name == "pact8(Clone)")
                                                                                                                                                                                   if (y == "Slot4")
                                   x = 1;
                                                                                                                                                                                              x = 1;
                              else
                              \{ x = -1; \}
                              stroka[6] = x.ToString();
                                                                                                                                                                                         stroka[3] = x.ToString();
                        StreamWriter f = new
                                                                                                                                                                                   if (y == "Slot5")
StreamWriter("крафт.txt", false);
                        for (int i = 0; i < stroka.Length; i++)
                                                                                                                                                                                        x = -1;
                              f.WriteLine(stroka[i]);
                                                                                                                                                                                         stroka[4] = x.ToString();
                                                                                                                                                                                   if (y == "Slot6")
                        f.Close();
            }
                                                                                                                                                                                       x = -1;
```

if (item2 != null)

```
stroka[5] = x.ToString();
                                                                        n += 1:
           if (y == "Slot7")
             x = -1;
                                                                   if (n == 7)
                                                                      dt[0].gameObject.SetActive(true);
             stroka[6] = x.ToString();
                                                                     stroka2[0]=(int.Parse(stroka2[0])+1).ToString();
                                                                     n = 0;
           StreamWriter f = new
                                                                     GameObject[]
 StreamWriter("крафт.txt", false);
                                                                                                 objects
                                                                GameObject.FindObjectsOfType<GameObject>();
           for (int i = 0; i < \text{stroka.Length}; i++)
                                                                      foreach (GameObject obj in objects)
             f.WriteLine(stroka[i]);
                                                                        if (obj.name == "pact9(Clone)")
           f.Close();
                                                                           // Получаем координаты объекта
    }
                                                                           double
 }
                                                                Math.Round(obj.transform.position.x, 2);
                                                                           double
Файл sosdanie.cs:
                                                                Math.Round(obj.transform.position.y, 2);
using System.Collections;
                                                                           // Проверяем, совпадают ли координаты
                                                                           if (objX == 1397 \&\& objY == 686)
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;
                                                                             // Удаляем первый найденный объект
using System.IO;
                                                                             Destroy(obj);
using System;
                                                                             break; // Выходим из цикла после
                                                                удаления
public class sosdanie: MonoBehaviour
  public Button[] dt;
                                                                      foreach (GameObject obj in objects)
                                                                        if (obj.name == "pact14(Clone)")
  private void Start()
                                                                           // Получаем координаты объекта
    string[]
                             stroka2
                                                                           double
                                                                                                 objX
 File.ReadAllLines("зелья_кнопки.txt");
                                                                Math.Round(obj.transform.position.x, 2);
                                                                           double
    for (int i = 0; i < stroka2.Length; i++)
                                                                Math.Round(obj.transform.position.y, 2);
       if (int.Parse(stroka2[i]) >0)
                                                                           // Проверяем, совпадают ли координаты
                                                                           if (objX == 1585.2 \&\& objY == 684.8)
         dt[i].gameObject.SetActive(true);
                                                                             // Удаляем первый найденный объект
                                                                             Destroy(obj);
                                                                             break; // Выходим из цикла после
                                                                удаления
  public void Sosdanie()
    string[] stroka = File.ReadAllLines("κραφτ.txt");
                                                                      foreach (GameObject obj in objects)
                             stroka2
    string[]
 File.ReadAllLines("зелья_кнопки.txt");
                                                                        if (obj.name == "pact3(Clone)")
    int n = 0:
    for (int i = 0; i < stroka.Length; i++)
                                                                           // Получаем координаты объекта
       if (int.Parse(stroka[i]) == 1)
```

```
double
                               obiX
Math.Round(obj.transform.position.x, 2);
                                                                       // Получаем координаты объекта
          double
                               obiY
                                                                       double
                                                                                            obiX
Math.Round(obj.transform.position.y, 2);
                                                             Math.Round(obj.transform.position.x, 2);
                                                                       double
                                                                                            obiY
          // Проверяем, совпадают ли координаты
                                                             Math.Round(obj.transform.position.y, 2);
          if (objX == 1311.3 \&\& objY == 563.9)
                                                                       // Проверяем, совпадают ли координаты
                                                                       if (objX == 1397 \&\& objY == 432.5)
            // Удаляем первый найденный объект
            Destroy(obj);
            break; // Выходим из цикла после
                                                                         // Удаляем первый найденный объект
                                                                         Destroy(obj);
удаления
                                                                         break; // Выходим из цикла после
                                                             удаления
     foreach (GameObject obj in objects)
                                                                  foreach (GameObject obj in objects)
       if (obj.name == "pact(Clone)")
                                                                    if (obj.name == "pact8(Clone)")
          // Получаем координаты объекта
          double
                                                                       // Получаем координаты объекта
                               obiX
Math.Round(obj.transform.position.x, 2);
                                                                       double
                                                                                            obiX
          double
                                                             Math.Round(obj.transform.position.x, 2);
Math.Round(obj.transform.position.y, 2);
                                                                       double
                                                                                            obiY
                                                            Math.Round(obj.transform.position.y, 2);
          // Проверяем, совпадают ли координаты
          if (objX == 1488.8 \&\& objY == 563.3)
                                                                       // Проверяем, совпадают ли координаты
                                                                       if (objX == 1587 \&\& objY == 432.4)
            // Удаляем первый найденный объект
                                                                         // Удаляем первый найденный объект
            Destroy(obj);
            break; // Выходим из цикла после
                                                                         Destroy(obj);
удаления
                                                                         break; // Выходим из цикла после
                                                             удаления
     foreach (GameObject obj in objects)
                                                                  StreamWriter f = new StreamWriter("kpa<math>\phi T.txt",
       if (obj.name == "pact7(Clone)")
                                                             false);
                                                                  for (int i = 0; i < \text{stroka.Length}; i++)
          // Получаем координаты объекта
                                                                    stroka[i] = "0";
          double
                               objX
Math.Round(obj.transform.position.x, 2);
                                                                    f.WriteLine(stroka[i]);
          double
Math.Round(obj.transform.position.y, 2);
                                                                  f.Close();
          // Проверяем, совпадают ли координаты
                                                                        ((int.Parse(stroka[0])
          if (objX == 1672.8 \&\& objY == 563.4)
                                                                                                           2)&&
                                                             (int.Parse(stroka[1]) == 2) && (int.Parse(stroka[4]) ==
            // Удаляем первый найденный объект
                                                             2) && (int.Parse(stroka[3]) == 1))
            Destroy(obj);
            break; // Выходим из цикла после
                                                                  dt[1].gameObject.SetActive(true);
                                                                  stroka2[1] = (int.Parse(stroka2[1]) + 1).ToString();
удаления
                                                                  GameObject[]
                                                                                            objects
                                                            GameObject.FindObjectsOfType<GameObject>();
     foreach (GameObject obj in objects)
                                                                  foreach (GameObject obj in objects)
       if (obj.name == "pact10(Clone)")
```

```
if (obj.name == "pact13(Clone)")
                                                                    if (obj.name == "pact6(Clone)")
         // Получаем координаты объекта
          double
                               objX
                                                                      // Получаем координаты объекта
Math.Round(obj.transform.position.x, 2);
                                                                      double
                                                                                           objX
          double
                                                            Math.Round(obj.transform.position.x, 2);
Math.Round(obj.transform.position.y, 2);
                                                                      double
          // Проверяем, совпадают ли координаты
                                                            Math.Round(obj.transform.position.y, 2);
          if (objX == 1397 \&\& objY == 686)
                                                                      // Проверяем, совпадают ли координаты
            // Удаляем первый найденный объект
                                                                      if (objX == 1672.8 \&\& objY == 563.4)
            Destroy(obj);
            break; // Выходим из цикла после
                                                                        // Удаляем первый найденный объект
                                                                        Destroy(obj);
удаления
                                                                        break; // Выходим из цикла после
                                                            удаления
     foreach (GameObject obj in objects)
        if (obj.name == "pact4(Clone)")
                                                                 StreamWriter f = new StreamWriter("κραφτ.txt",
          // Получаем координаты объекта
                                                                  for (int i = 0; i < \text{stroka.Length}; i++)
          double
                               obiX
Math.Round(obj.transform.position.x, 2);
                                                                    stroka[i] = "0";
          double
                                                                    f.WriteLine(stroka[i]);
                               obiY
Math.Round(obj.transform.position.y, 2);
                                                                 f.Close();
                                                               if
                                                                      ((int.Parse(stroka[0])
         // Проверяем, совпадают ли координаты
                                                                                                             &&
          if (objX == 1585.2 \&\& objY == 684.8)
                                                            (int.Parse(stroka[1]) == 3) && (int.Parse(stroka[4]) ==
                                                            3) && (int.Parse(stroka[3]) == 1))
            // Удаляем первый найденный объект
                                                                 dt[2].gameObject.SetActive(true);
            Destroy(obj);
            break; // Выходим из цикла после
                                                                 stroka2[2] = (int.Parse(stroka2[2]) + 1).ToString();
удаления
                                                                 GameObject[]
                                                                                           objects
                                                            GameObject.FindObjectsOfType<GameObject>();
                                                                 foreach (GameObject obj in objects)
     foreach (GameObject obj in objects)
                                                                    if (obj.name == "раст2(Clone)")
       if (obj.name == "pact(Clone)")
          // Получаем координаты объекта
                                                                      // Получаем координаты объекта
          double
                               objX
                                                                      double
                                                                                           objX
Math.Round(obj.transform.position.x, 2);
                                                            Math.Round(obj.transform.position.x, 2);
          double
                                                                      double
Math.Round(obj.transform.position.y, 2);
                                                            Math.Round(obj.transform.position.y, 2);
                                                                      // Проверяем, совпадают ли координаты
          // Проверяем, совпадают ли координаты
                                                                      if (objX == 1397 \&\& objY == 686)
          if (objX == 1488.8 \&\& objY == 563.3)
                                                                         // Удаляем первый найденный объект
            // Удаляем первый найденный объект
                                                                        Destroy(obj);
                                                                        break; // Выходим из цикла после
            Destroy(obj);
            break; // Выходим из цикла после
                                                            удаления
удаления
                                                                 foreach (GameObject obj in objects)
     foreach (GameObject obj in objects)
```

```
if (obj.name == "pact15(Clone)")
                                                                    StreamWriter f = new StreamWriter("κραφτ.txt",
                                                              false):
          // Получаем координаты объекта
                                                                    for (int i = 0; i < \text{stroka.Length}; i++)
          double
                                objX
Math.Round(obj.transform.position.x, 2);
                                                                      stroka[i] = "0";
          double
                                                                      f.WriteLine(stroka[i]);
Math.Round(obj.transform.position.y, 2);
                                                                    f.Close();
          // Проверяем, совпадают ли координаты
          if (objX == 1585.2 \&\& objY == 684.8)
            // Удаляем первый найденный объект
            Destroy(obj);
            break; // Выходим из цикла после
                                                                  StreamWriter
удаления
                                                                                                                new
                                                              StreamWriter("зелья кнопки.txt", false);
                                                                  for (int i = 0; i < stroka2.Length; i++)
     foreach (GameObject obj in objects)
                                                                    f2.WriteLine(stroka2[i]);
        if (obj.name == "pact(Clone)")
                                                                  f2.Close();
          // Получаем координаты объекта
                                                               public void sele clik(int n)
          double
                                obiX
Math.Round(obj.transform.position.x, 2);
          double
                                                                  string[] stroka1 = File.ReadAllLines("зелья.txt");
Math.Round(obj.transform.position.y, 2);
                                                                                          stroka2
                                                              File.ReadAllLines("зелья кнопки.txt");
          // Проверяем, совпадают ли координаты
          if (objX == 1488.8 \&\& objY == 563.3)
                                                                  if (n == 0) {
                                                                    stroka1[0] = (int.Parse(stroka1[0]) + 1).ToString();
            // Удаляем первый найденный объект
                                                                    stroka2[0] = 0.ToString();
            Destroy(obj);
            break; // Выходим из цикла после
                                                                  if(n == 1)
удаления
                                                                    stroka1[1] = (int.Parse(stroka1[1]) + 1).ToString();
                                                                    stroka2[1] = 0.ToString();
     foreach (GameObject obj in objects)
        if (obj.name == "pact1(Clone)")
                                                                  if (n == 2)
          // Получаем координаты объекта
                                                                    stroka1[2] = (int.Parse(stroka1[2]) + 1).ToString();
          double
                                                                    stroka2[2] = 0.ToString();
                                obiX
Math.Round(obj.transform.position.x, 2);
          double
                                                                  StreamWriter f1 = new StreamWriter("зелья.txt",
Math.Round(obj.transform.position.y, 2);
                                                                  for (int i = 0; i < \text{strokal.Length}; i++)
          // Проверяем, совпадают ли координаты
          if (objX == 1672.8 \&\& objY == 563.4)
                                                                    f1.WriteLine(stroka1[i]);
            // Удаляем первый найденный объект
                                                                  f1.Close();
            Destroy(obj);
                                                                  StreamWriter
            break; // Выходим из цикла после
                                                                                                                 new
                                                              StreamWriter("зелья кнопки.txt", false);
удаления
                                                                  for (int i = 0; i < stroka2.Length; i++)
                                                                    f2.WriteLine(stroka2[i]);
```

```
text[i+stroka.Length+ stroka1.Length+
     f2.Close();
                                                                    stroka2.Length].text = stroka3[i];
     dt[n].gameObject.SetActive(false);
                                                                         for (int i = 0; i < stroka4.Length; i++)
                                                                            text[i+ stroka.Length+ stroka1.Length+
                                                                    stroka2.Length+ stroka3.Length].text = stroka4[i];
Файл invent.cs:
 using System.Collections;
 using System.Collections.Generic;
 using UnityEngine;
                                                                         for (int i = 0; i < \text{stroka5.Length}; i++)
 using UnityEngine.UI;
                                                                            text[i + stroka.Length + stroka1.Length +
 using System.IO;
                                                                    stroka2.Length + stroka3.Length+ stroka4.Length].text
 public class invent: MonoBehaviour
                                                                    = stroka5[i];
                                                                      }
    public Text[] text;
    public Image img;
                                                                      public void button_clic(int i)
    public Sprite[] sprites;
    public Переход perehod;
                                                                         //для смены номера сцены на кнопке
    public Button ispols;
                                                                    использовать
    private void Start()
                                                                         ispols.gameObject.SetActive(false);
                                                                         img.sprite=sprites[i];
      img.sprite = sprites[0];
      ispols.gameObject.SetActive(false);
                                                                         string[] stroka1 =
                                                                    File.ReadAllLines("предметы.txt");
                                                                         string[] stroka3 = File.ReadAllLines("карта.txt");
    void Update()
                                                                         if ((i == 1) \&\& (int.Parse(stroka3[0])>0))
                                                                            ispols.gameObject.SetActive(true);
       string[] stroka =
 File.ReadAllLines("растения.txt");
                                                                            perehod.levelToLoad = 16;
       string[] stroka1 =
 File.ReadAllLines("предметы.txt");
                                                                         img.sprite = sprites[i];
                                                                         if ((i == 2) \&\& (int.Parse(stroka1[0]) > 0))
      string[] stroka2 =
 File.ReadAllLines("манеты.txt");
      string[] stroka3 = File.ReadAllLines("карта.txt");
                                                                            ispols.gameObject.SetActive(true);
      string[] stroka4 = File.ReadAllLines("зелья.txt");
                                                                            perehod.levelToLoad = 10;
       string[] stroka5 =
 File.ReadAllLines("количество.txt");
                                                                            string[] stroka =
                                                                    File.ReadAllLines("использовать.txt");
      for (int i = 0; i < \text{stroka.Length}; i++)
                                                                            stroka[0]=0.ToString();
                                                                            StreamWriter f1 = new
         text[i].text=stroka[i];
                                                                    StreamWriter("использовать.txt", false);
                                                                            for (int k = 0; k < \text{stroka.Length}; k++)
      for (int i = 0; i < \text{stroka1.Length}; i++)
                                                                              f1.WriteLine(stroka[k]);
         text[i+ stroka.Length].text = stroka1[i];
                                                                            f1.Close();
      for (int i = 0; i < stroka2.Length; i++)
                                                                         img.sprite = sprites[i];
                                                                         if ((i == 4) \&\& (int.Parse(stroka1[1]) > 0))
         text[i+ stroka.Length+ stroka1.Length].text =
 stroka2[i];
                                                                            ispols.gameObject.SetActive(true);
                                                                            perehod.levelToLoad = 14;
       }
      for (int i = 0; i < stroka3.Length; i++)
                                                                         img.sprite = sprites[i];
                                                                         if ((i == 5) \&\& (int.Parse(stroka1[2]) > 0))
```

```
Файл magas.cs:
         ispols.gameObject.SetActive(true);
                                                                   using System.Collections;
         perehod.levelToLoad = 10;
                                                                   using System.Collections.Generic;
                                                                   using UnityEngine;
         string[] stroka =
                                                                   using UnityEngine.UI;
 File.ReadAllLines("использовать.txt");
                                                                   using System.IO;
         stroka[0] = 1.ToString();
         StreamWriter f1 = new
                                                                   public class magas: MonoBehaviour
 StreamWriter("использовать.txt", false);
         for (int k = 0; k < \text{stroka.Length}; k++)
                                                                     public GameObject[] scrol;
                                                                     public Text text1;
                                                                     public Image img;
           f1.WriteLine(stroka[k]);
                                                                     public Sprite[] sprites;
         f1.Close();
                                                                     public Button[] bt;
                                                                     void Start()
      img.sprite = sprites[i];
      if ((i == 6) \&\& (int.Parse(stroka1[3]) > 0))
                                                                        string[] stroka = File.ReadAllLines("Номер
                                                                   сцены.txt");
         ispols.gameObject.SetActive(true);
                                                                        string[] stroka1 = File.ReadAllLines("Купля-
         perehod.levelToLoad = 10;
                                                                   прдажа.txt");
                                                                        string[] stroka2 = File.ReadAllLines("цена в
         string[] stroka =
                                                                   торговой лавке.txt");
 File.ReadAllLines("использовать.txt");
                                                                        string[] stroka3 = File.ReadAllLines("цена
         stroka[0] = 2.ToString();
                                                                   продажи.txt");
         StreamWriter f1 = new
                                                                        string[] stroka4 = File.ReadAllLines("цена в
 StreamWriter("использовать.txt", false);
                                                                   магазине магии.txt");
         for (int k = 0; k < \text{stroka.Length}; k++)
                                                                        if ((int.Parse(stroka[0]) == 6) \&\&
           f1.WriteLine(stroka[k]);
                                                                   (int.Parse(stroka1[0]) == 0))
         f1.Close();
                                                                          scrol[0].SetActive(true);
                                                                          img.sprite=sprites[0];
    }
                                                                          bt[0].gameObject.SetActive(true);
                                                                          text1.text= stroka2[0];
Файл kupl_prod.cs:
                                                                        if ((int.Parse(stroka[0]) == 6) \&\&
                                                                   (int.Parse(stroka1[0]) == 1))
 using System.Collections;
 using System.Collections.Generic;
 using UnityEngine;
                                                                          scrol[1].SetActive(true);
 using System.IO;
                                                                          img.sprite = sprites[1];
                                                                          bt[1].gameObject.SetActive(true);
                                                                          text1.text = stroka3[0];
 public class kupl_prod : MonoBehaviour
    public void k p(int i)
                                                                        if ((int.Parse(stroka[0]) == 7))
      string[] stroka = File.ReadAllLines("Купля-
                                                                          scrol[2].SetActive(true);
 прдажа.txt");
                                                                          img.sprite = sprites[2];
      stroka[0]=i.ToString();
                                                                          bt[2].gameObject.SetActive(true);
      StreamWriter f1 = new StreamWriter("Купля-
                                                                          text1.text = stroka4[0];
 прдажа.txt", false);
      for (int k = 0; k < \text{stroka.Length}; k++)
                                                                     }
         f1.WriteLine(stroka[k]);
                                                                   }
      f1.Close();
                                                                 Файл pokupca.cs:
                                                                   using System.Collections;
                                                                   using System.Collections.Generic;
                                                                   using UnityEngine;
```

```
using UnityEngine.UI;
using System.IO;
                                                                          stroka[0] = (int.Parse(stroka[0]) -
                                                                  int.Parse(text1.text)).ToString();
public class pokupca: MonoBehaviour
                                                                          text0.text = stroka[0];
  public int num;
                                                                          StreamWriter f1 = new
  public Text text0;
                                                                  StreamWriter("манеты.txt", false);
  public Text text1;
                                                                          for (int k = 0; k < \text{stroka.Length}; k++)
  public Image img;
  public Sprite[] sprites;
                                                                            f1.WriteLine(stroka[k]);
                                                                          f1.Close();
                                                                          if (num == 0)
  private void Start()
                                                                            stroka1[0] = (int.Parse(stroka1[0]) +
     num = 0;
                                                                  1).ToString();
                                                                            StreamWriter f = new
                                                                  StreamWriter("предметы.txt", false);
                                                                            for (int k = 0; k < \text{stroka1.Length}; k++)
  void Update()
                                                                               f.WriteLine(stroka1[k]);
     string[] stroka = File.ReadAllLines("манеты.txt");
     text0.text = stroka[0];
                                                                            f.Close();
  public void button_click(int i)
                                                                          if (num == 1)
     num = i;
                                                                            stroka1[2] = (int.Parse(stroka1[2]) +
                                                                  1).ToString();
     img.sprite = sprites[i];
     string[] stroka1 = File.ReadAllLines("Номер
                                                                            StreamWriter f = new
сцены.txt");
                                                                  StreamWriter("предметы.txt", false);
     if ((int.Parse(stroka1[0]) == 6))
                                                                            for (int k = 0; k < \text{stroka1.Length}; k++)
       string[] stroka = File.ReadAllLines("цена в
                                                                               f.WriteLine(stroka1[k]);
торговой лавке.txt");
       text1.text = stroka[i];
                                                                            f.Close();
     if ((int.Parse(stroka1[0]) == 7))
                                                                          if ((num >= 2) && (num <= 27))
       string[] stroka = File.ReadAllLines("цена в
                                                                            stroka2[num - 2] = (int.Parse(stroka2[num -
магазине магии.txt");
                                                                  2]) + 1).ToString();
       text1.text = stroka[i];
                                                                            StreamWriter f = new
  }
                                                                  StreamWriter("количество.txt", false);
                                                                            for (int k = 0; k < \text{stroka2.Length}; k++)
  public void pokur()
                                                                               f.WriteLine(stroka2[k]);
     string[] stroka = File.ReadAllLines("манеты.txt");
     string[] stroka1 =
                                                                            f.Close();
File.ReadAllLines("предметы.txt");
     string[] stroka2 =
File.ReadAllLines("количество.txt");
     if (int.Parse(stroka[0])>= int.Parse(text1.text))
                                                                     public void pokur_magii()
```

```
string[] stroka = File.ReadAllLines("манеты.txt");
                                                                               f.WriteLine(stroka1[k]);
     string[] stroka1 =
File.ReadAllLines("предметы.txt");
                                                                             f.Close();
     string[] stroka2 = File.ReadAllLines("kapta.txt");
     string[] stroka3 = File.ReadAllLines("зелья.txt");
                                                                          if ((num == 3))
     if (int.Parse(stroka[0]) >= int.Parse(text1.text))
                                                                             stroka1[3] = (int.Parse(stroka1[3]) +
                                                                  1).ToString();
       stroka[0] = (int.Parse(stroka[0]) -
                                                                             StreamWriter f = new
int.Parse(text1.text)).ToString();
                                                                  StreamWriter("предметы.txt", false);
                                                                             for (int k = 0; k < \text{stroka1.Length}; k++)
       text0.text = stroka[0];
                                                                               f.WriteLine(stroka1[k]);
       StreamWriter f1 = new
                                                                             f.Close();
StreamWriter("манеты.txt", false);
       for (int k = 0; k < \text{stroka.Length}; k++)
          f1.WriteLine(stroka[k]);
       f1.Close();
       if (num == 0)
                                                                 Файл prodaja.cs:
                                                                  using System.Collections;
          stroka2[0] = (int.Parse(stroka2[0]) +
1).ToString();
                                                                  using System.Collections.Generic;
                                                                  using UnityEngine;
          StreamWriter f = new
                                                                  using UnityEngine.UI;
StreamWriter("карта.txt", false);
                                                                  using System.IO;
          for (int k = 0; k < stroka2.Length; k++)
                                                                  public class prodaja: MonoBehaviour
            f.WriteLine(stroka2[k]);
                                                                     public Text[] text;
          f.Close();
                                                                     public int num;
       if (num == 1)
                                                                     public Text text0;
                                                                     public Text text1;
                                                                     public Text text2;
          stroka3[0] = (int.Parse(stroka3[0]) +
1).ToString();
                                                                     public Image img;
                                                                     public Sprite[] sprites;
          StreamWriter f = new
StreamWriter("зелья.txt", false);
                                                                     void Start()
          for (int k = 0; k < \text{stroka3.Length}; k++)
                                                                        num = 0;
            f.WriteLine(stroka3[k]);
          f.Close();
                                                                     void Update()
                                                                        string[] stroka =
       if ((num == 2))
                                                                  File.ReadAllLines("растения.txt");
          stroka1[1] = (int.Parse(stroka1[1]) +
                                                                        string[] stroka1 =
1).ToString();
                                                                  File.ReadAllLines("предметы.txt");
                                                                        string[] stroka2 = File.ReadAllLines("зелья.txt");
          StreamWriter f = new
StreamWriter("предметы.txt", false);
                                                                        for (int i = 0; i < stroka.Length; i++)
          for (int k = 0; k < \text{stroka1.Length}; k++)
                                                                          text[i].text = stroka[i];
```

```
text[stroka.Length].text = stroka1[4];
                                                                        if (num == 15)
     for (int i = 1; i < stroka2.Length; i++)//3елья в
                                                                           stroka2[4] = (int.Parse(stroka2[4]) -
файле со 2 строки
                                                                 int.Parse(text2.text)).ToString();
       text[i + stroka.Length + 1-1].text = stroka2[i];
                                                                           StreamWriter f = new
                                                                 StreamWriter("предметы.txt", false);
                                                                           for (int k = 0; k < \text{stroka2.Length}; k++)
  public void button_click(int i)
                                                                             f.WriteLine(stroka2[k]);
                                                                           f.Close();
     num = i;
     img.sprite = sprites[i];
     string[] stroka = File.ReadAllLines("цена
продажи.txt");
     text1.text = stroka[i];
                                                                        if ((num >= 16) \&\& (num <= 17))
                                                                           stroka3[num-15] = (int.Parse(stroka3[num-
  public void prod()
                                                                 15]) - int.Parse(text2.text)).ToString();
                                                                           StreamWriter f = new
     string[] stroka = File.ReadAllLines("манеты.txt");
     string[] stroka1 =
                                                                 StreamWriter("растения.txt", false);
File.ReadAllLines("растения.txt");
                                                                           for (int k = 0; k < stroka3.Length; k++)
     string[] stroka2 =
File.ReadAllLines("предметы.txt");
                                                                             f.WriteLine(stroka3[k]);
     string[] stroka3 = File.ReadAllLines("зелья.txt");
                                                                           f.Close();
     if (int.Parse(text[num].text) >=
int.Parse(text2.text))
       stroka[0] = (int.Parse(stroka[0]) +
(int.Parse(text1.text)* int.Parse(text2.text))).ToString();
                                                               Файл Draggable.cs:
       text0.text = stroka[0];
                                                                 using System.Collections;
                                                                 using System.Collections.Generic;
       StreamWriter f1 = new
                                                                 using UnityEngine;
StreamWriter("манеты.txt", false);
                                                                 using System.IO;
       for (int k = 0; k < \text{stroka.Length}; k++)
                                                                 using UnityEngine.EventSystems;
          f1.WriteLine(stroka[k]);
                                                                 public class Draggable: MonoBehaviour,
                                                                 IDragHandler, IBeginDragHandler
       f1.Close();
                                                                   private Canvas canvas;
                                                                   private RectTransform rectTransform;
       if ((num \ge 0) & (num < 14))
                                                                   private Vector3 originalPosition;
          stroka1[num] = (int.Parse(stroka1[num]) -
                                                                   void Awake()
int.Parse(text2.text)).ToString();
                                                                      rectTransform =
          StreamWriter f = new
                                                                 GetComponent<RectTransform>();
                                                                      canvas = GetComponentInParent<Canvas>();
StreamWriter("растения.txt", false);
          for (int k = 0; k < \text{stroka1.Length}; k++)
                                                                      originalPosition =
                                                                 rectTransform.anchoredPosition:
            f.WriteLine(stroka1[k]);
          f.Close();
```

```
public void OnBeginDrag(PointerEventData
                                                                     else
 eventData)
                                                                       plus.gameObject.SetActive(true);
      // Сохранение оригинальной позиции при
                                                                       pl = false;
 начале перетаскивания
      originalPosition =
 rectTransform.anchoredPosition;
                                                                }
    public void OnDrag(PointerEventData eventData)
                                                               Файл JewelryManager.cs:
                                                                using System.Collections;
                                                                using System.Collections.Generic;
      Vector2 position;
                                                                using System.IO;
 Rect Transform Utility. Screen Point To Local Point In Recta \\
                                                                using UnityEngine;
                                                                using UnityEngine.EventSystems;
 ngle(canvas.transform as RectTransform,
 eventData.position, canvas.worldCamera, out position);
                                                                using UnityEngine.UI;
      rectTransform.anchoredPosition = position;
                                                                using static System.Net.Mime.MediaTypeNames;
                                                                public class JewelryManager: MonoBehaviour
   public Vector3 GetCurrentPosition()
                                                                   public GameObject[] jewelryPrefabs; // массив
      return rectTransform.anchoredPosition;
                                                                префабов украшений
                                                                  public Button[] buttons;
                                                                   private Dictionary<int, int> quantities = new
                                                                Dictionary<int, int>();
Файл inwentari.cs:
                                                                   public Transform parentTransform;
 using System.Collections;
                                                                   public Vector2 spawnPosition;
 using System.Collections.Generic;
 using UnityEngine;
                                                                   private List<Draggable> draggableJewelry = new
 using UnityEngine.UI;
                                                                List<Draggable>();
 using System.IO;
                                                                   void Start()
 public class inwentari : MonoBehaviour
                                                                     LoadQuantities();
                                                                     LoadJewelryPositions();
   public Text∏ text;
                                                                     for (int i = 0; i < buttons.Length; i++)
   public GameObject plus;
    public bool pl=true;
                                                                       int index = i;
                                                                       buttons[i].onClick.AddListener(() =>
                                                                OnJewelryButtonClick(index));
    void Update()
      string[] stroka =
 File.ReadAllLines("количество.txt");
                                                                   void Update()
      for (int i = 0; i < \text{stroka.Length}; i++)
                                                                     // Проверяем, нажата ли клавиша Backspace
                                                                     if (Input.GetKeyDown(KeyCode.Backspace))
        text[i].text = stroka[i];
                                                                       RemoveSelectedJewelry();
    public void Plus()
                                                                     // Проверяем, нажата ли клавиша Тав для
                                                                поворота украшения
                                                                     if (Input.GetKeyDown(KeyCode.Tab))
      if (!pl)
        plus.gameObject.SetActive(false);
                                                                       RotateSelectedJewelry();
        pl = true;
```

}

```
// Проверяем, нажаты ли клавиши + и - для
изменения размера
    if (Input.GetKeyDown(KeyCode.Equals)) //
                                                                           GameObject prefab =
Клавиша + на большинстве раскладок
                                                               System.Array.Find(jewelryPrefabs, p => p.name ==
                                                               name);
       ChangeSelectedJewelrySize(1.1f); //
                                                                           if (prefab != null)
Увеличиваем размер
                                                                             GameObject jewelryInstance =
                                                              Instantiate(prefab, parentTransform);
    else if (Input.GetKeyDown(KeyCode.Minus))
       ChangeSelectedJewelrySize(0.9f); //
                                                               jewelryInstance.GetComponent<RectTransform>().anc
                                                               horedPosition = new Vector2(x, y);
Уменьшаем размер
                                                                             jewelryInstance.transform.rotation =
                                                               Quaternion.Euler(0, 0, angle); // Устанавливаем угол
                                                                             jewelryInstance.transform.localScale =
    // Проверяем, нажата ли клавиша Тав для
поворота украшения по окружности
                                                               new Vector3(size, size1, 1); // Устанавливаем размер
    if (Input.GetKeyDown(KeyCode.Q))
                                                              jewelryInstance.AddComponent<Draggable>(); //
       RotateSelectedJewelry1();
                                                               добавляем компонент для перетаскивания
                                                               draggableJewelry.Add(jewelryInstance.GetComponent
                                                               <Draggable>()); // добавляем в список
  void LoadQuantities()
    string[] lines =
File.ReadAllLines("количество.txt");
    for (int i = 0; i < lines.Length; i++)
                                                                 public void OnJewelryButtonClick(int index)
       if (int.TryParse(lines[i], out int quantity))
                                                                   if (quantities.ContainsKey(index) &&
                                                               quantities[index] > 0)
         quantities[i] = quantity;
                                                                      GameObject jewelryInstance =
                                                               Instantiate(jewelryPrefabs[index], parentTransform);
                                                               jewelryInstance.GetComponent<RectTransform>().anc
  void LoadJewelryPositions()
                                                              horedPosition = spawnPosition;
                                                                      quantities[index]--; // Уменьшаем количество
                                                                      UpdateQuantityFile();
    if (File.Exists("positions.txt"))
                                                                     jewelryInstance.AddComponent<Draggable>();
       string[] lines =
File.ReadAllLines("positions.txt");
                                                               draggableJewelry.Add(jewelryInstance.GetComponent
       foreach (string line in lines)
                                                               <Draggable>());
         string[] parts = line.Split('|');
         if (parts.Length == 5) // Изменили на 4 для
учета размера
                                                                 void UpdateQuantityFile()
            string name = parts[0].Substring(0,
                                                                   using (StreamWriter writer = new
parts[0].Length - 7);
                                                              StreamWriter("количество.txt"))
            float x = \text{float.Parse(parts[1])};
            float y = float.Parse(parts[2]);
                                                                      foreach (var quantity in quantities)
            float size = float.Parse(parts[3]);
                                                                        writer.WriteLine(quantity.Value);
            float angle = float.Parse(parts[4]);
            float size 1 = size:
            if (size < 0)
              size1 *=-1;
```

```
private void OnApplicationQuit()
                                                                 private int GetJewelryIndex(string name)
     SavePositions(); // Сохраняем позиции при
                                                                   for (int i = 0; i < jewelryPrefabs.Length; <math>i++)
выходе из приложения
                                                                      if (name.StartsWith("ykp" + (i + 1).ToString() +
                                                              "(Clone)"))
  public void SavePositions()
                                                                        return i;
     File.WriteAllText("positions.txt", string.Empty);
     foreach (var draggable in draggableJewelry)
                                                                   return -1;
       string positionData =
$"{draggable.gameObject.name}|{draggable.GetCurren
tPosition().x}|{draggable.GetCurrentPosition().y}|{dra
                                                                 private void RotateSelectedJewelry()
ggable.transform.localScale.x}|{draggable.transform.ro
tation.eulerAngles.z}\n"; // Добавляем размер
                                                                   PointerEventData pointerData = new
       File.AppendAllText("positions.txt",
                                                              PointerEventData(EventSystem.current) { position =
                                                              Input.mousePosition };
positionData); // Сохраняем позицию в файл
                                                                   List<RaycastResult> results = new
     draggableJewelry.Clear(); // Очищаем список
                                                              List<RaycastResult>();
после сохранения
                                                                   EventSystem.current.RaycastAll(pointerData,
                                                              results);
  private void RemoveSelectedJewelry()
                                                                   foreach (RaycastResult result in results)
     PointerEventData pointerData = new
PointerEventData(EventSystem.current) { position =
                                                                      Draggable draggable =
Input.mousePosition };
                                                              result.gameObject.GetComponent<Draggable>();
     List<RaycastResult> results = new
                                                                      if (draggable != null)
List<RaycastResult>();
                                                                        draggable.transform.localScale = new
     EventSystem.current.RaycastAll(pointerData,
                                                              Vector3(-draggable.transform.localScale.x,
results);
                                                              draggable.transform.localScale.v, 1); // Поворачиваем
     foreach (RaycastResult result in results)
                                                              в противоположную сторону
                                                                        break; // Поворачиваем только первое
       Draggable draggable =
                                                              найденное украшение
result.gameObject.GetComponent<Draggable>();
       if (draggable != null)
                                                                 }
         int index =
GetJewelryIndex(draggable.gameObject.name);
                                                                 private void RotateSelectedJewelry1()
         if (index !=-1)
                                                                   PointerEventData pointerData = new
                                                              PointerEventData(EventSystem.current) { position =
            // Увеличиваем количество в словаре
            quantities[index]++;
                                                              Input.mousePosition };
                                                                   List<RaycastResult> results = new
                                                              List<RaycastResult>():
            // Обновляем файл после изменения
количества
                                                                   EventSystem.current.RaycastAll(pointerData,
            UpdateQuantityFile();
                                                              results);
            // Удаляем объект из списка
                                                                   foreach (RaycastResult result in results)
draggableJewelry и уничтожаем его
            draggableJewelry.Remove(draggable);
                                                                      Draggable draggable =
            Destroy(draggable.gameObject);
                                                              result.gameObject.GetComponent<Draggable>();
                                                                     if (draggable != null)
            break;
                                                                        draggable.transform.Rotate(0, 0, 90); //
                                                              Поворачиваем на 90 градусов
     }
```

}

```
using System.Collections.Generic;
           break; // Поворачиваем только первое
 найденное украшение
                                                               using UnityEngine;
                                                               public class Exit: MonoBehaviour
    private void ChangeSelectedJewelrySize(float
                                                                  public void ExitGame()
 scaleFactor)
                                                                    Application.Quit();
      PointerEventData pointerData = new
 PointerEventData(EventSystem.current) { position =
 Input.mousePosition };
                                                              Файл Gan.cs:
      List<RaycastResult> results = new
                                                               using System.Collections;
 List<RaycastResult>();
                                                               using System.Collections.Generic;
    EventSystem.current.RaycastAll(pointerData,
                                                               using UnityEngine;
 results);
      foreach (RaycastResult result in results)
                                                               public class Gan: MonoBehaviour
        Draggable draggable =
                                                                  public float offset;
 result.gameObject.GetComponent<Draggable>();
                                                                  public GameObject bullet;
        if (draggable != null)
                                                                  public Transform shotPoint;
                                                                  private float timeBtwShots;
           draggable.transform.localScale *=
                                                                  public float startTimeBtwShots;
 scaleFactor; // Изменяем размер
                                                                  // Укажите объект, к которому будут
           break; // Изменяем только первое
                                                               прикрепляться пули
 найденное украшение
                                                                  public Transform parentObject;
                                                                  private void Update()
      }
    }
                                                                    Vector3 difference =
 }
                                                               Camera.main.ScreenToWorldPoint(Input.mousePositio
Файл Enemy.cs:
                                                               n) - transform.position;
                                                                    float rotZ = Mathf.Atan2(difference.y,
 using System.Collections;
 using System.Collections.Generic;
                                                               difference.x) * Mathf.Rad2Deg;
 using UnityEngine;
                                                                    transform.rotation = Quaternion.Euler(0f, 0f, rotZ
                                                               + offset):
 public class Enemy: MonoBehaviour
                                                                    if (timeBtwShots <= 0)
                                                                       if (Input.GetMouseButtonDown(0))
   public int health;
   public GameObject deathEffect;
                                                                         // Создаем пулю с фиксированным
    private void Update()
                                                               вращением
                                                                         GameObject newBullet = Instantiate(bullet,
      if (health \leq 0)
                                                               shotPoint.position, Quaternion.Euler(0f, 0f, 0f));
        Instantiate(deathEffect, transform.position,
                                                                         // Устанавливаем родителя для новой пули
 Quaternion.identity);
                                                               newBullet.transform.SetParent(parentObject);
        FindObjectOfType<Rang>().heal = 0;
                                                                         // Задаем направление полета для пули
        Destroy(gameObject);
                                                                         Bullet bulletScript =
                                                               newBullet.GetComponent<Bullet>();
                                                               bulletScript.SetDirection(difference.normalized); //
    public void TakeDamage(int damage)
                                                               Передаем направление
    { health -= damage; }
 }
                                                                         timeBtwShots = startTimeBtwShots:
Файл Exit.cs:
 using System.Collections;
                                                                    else
```

```
if (int.Parse(stroka[i]) == 1)
        timeBtwShots -= Time.deltaTime:
                                                                        { z21++; }
                                                                      if ((z21==21) \&\& (prod == false))
Файл Karta.cs:
                                                                           endddd.gameObject.SetActive(true);
 using System.Collections;
                                                                          StartCoroutine(stop());
 using System.Collections.Generic;
                                                                           prod = true;
 using UnityEngine;
 using System.IO;
                                                                   private IEnumerator stop()
 using UnityEngine.UI;
                                                                      yield return new WaitForSeconds(3f);
 public class Karta: MonoBehaviour
                                                                      Time.timeScale = 0f:
    public Button[] bt;
    public Image imgObg1;
    public Sprite[] sprites;
                                                                   public void prodolj()
    void Start()
                                                                      Time.timeScale = 1f;
      string[] stroka = File.ReadAllLines("посещённые
                                                                      endddd.gameObject.SetActive(false);
 локации.txt");
      if (int.Parse(stroka[0])==1)
                                                               }
                                                               Файл Kristall.cs:
        bt[0].gameObject.SetActive(true);
                                                                 using System.Collections;
        imgObg1.sprite = sprites[0];
                                                                 using System.Collections.Generic;
                                                                 using System.IO;
      if (int.Parse(stroka[1]) == 1)
                                                                 using UnityEngine;
                                                                 public class Kristall: MonoBehaviour
        bt[1].gameObject.SetActive(true);
                                                                   public GameObject kristall_pref;
        imgObg1.sprite = sprites[1];
                                                                   public Transform parentTransform;
                                                                   private void Awake()
                                                                      // Получаем родителя текущего объекта и
                                                                 записываем его в переменную
Файл Konec_igr.cs:
                                                                      parentTransform = transform.parent;
 using System.Collections;
 using System.Collections.Generic;
                                                                   private void OnTriggerEnter2D(Collider2D other)
 using System.IO;
 using UnityEngine;
                                                                      if (other.CompareTag("Player"))
 using UnityEngine.UI;
                                                                        string[] stroka =
 public class Konec_igr : MonoBehaviour
                                                                 File.ReadAllLines("предметы.txt");
                                                                        stroka[4]=(int.Parse(stroka[4])+1).ToString();
                                                                        FindObjectOfType<porajenie>().kol_vo += 1;
    public GameObject endddd;
                                                                        StreamWriter f = new
    bool prod = false;
                                                                 StreamWriter("предметы.txt", false);
                                                                        for (int i = 0; i < \text{stroka.Length}; i++)
    private void Update()
                                                                          f.WriteLine(stroka[i]);
      string[] stroka = File.ReadAllLines("выполнение
                                                                        f.Close();
                                                                        // Задаем массив возможных позиций
 зд.txt");
                                                                        Vector2[] positions = new Vector2[]
      int z21 = 0;
                                                                        new Vector2(453f, 171f),
      for (int i = 0; i < stroka.Length; i++)
                                                                        new Vector2(-1104f, -553f),
                                                                        new Vector2(1533f, -540f)
```

```
};
                                                              public class MusicManager: MonoBehaviour
        // Генерируем случайный индекс
        int randomIndex = Random.Range(0,
                                                                 public AudioClip[] musicClips;
 positions.Length);
                                                                 public AudioSource audioSource;
                                                                 private void Awake()
                                                                   if (FindObjectsOfType<MusicManager>().Length
        GameObject prefab = kristall_pref;
        if (prefab != null)
                                                              > 1)
           GameObject jewelryInstance =
                                                                     Destroy(gameObject);
 Instantiate(prefab, parentTransform);
                                                                     return;
 jewelryInstance.GetComponent<RectTransform>().anc
                                                                   DontDestroyOnLoad(gameObject);
                                                                   audioSource =
 horedPosition = positions[randomIndex]; //
 Устанавливаем позицию
                                                              gameObject.AddComponent<AudioSource>();
                                                                   audioSource.loop = true;
                                                                   PlayMusic(0);
        Destroy(gameObject);
                                                                 public void PlayMusic(int index)
    }
                                                                   if (index < musicClips.Length)
 }
Файл LoadingScreen1.cs:
                                                                     audioSource.clip = musicClips[index];
 using System.Collections;
                                                                     audioSource.Play();
 using System.Collections.Generic;
 using UnityEngine;
 using UnityEngine.SceneManagement;
                                                                 public void SetVolume(float volume)
 using UnityEngine.UI;
                                                                   audioSource=GetComponent<AudioSource>();//
 public class LoadingScreen1: MonoBehaviour
                                                                   audioSource.volume = volume;
    public Slider progressBar;
   public string sceneToLoad;
                                                                 public void TogglePause()
   private void Start()
                                                                   if (audioSource.isPlaying) audioSource.Pause();
                                                                     else audioSource.UnPause();
      StartCoroutine(LoadSceneAsync() );
                                                             Файл MusicSelectionUI.cs:
    IEnumerator LoadSceneAsync()
                                                              using System.Collections;
                                                              using System.Collections.Generic;
      AsyncOperation operation =
                                                              using UnityEngine;
 SceneManager.LoadSceneAsync( sceneToLoad );
                                                              using UnityEngine.UI;
      while (!operation.isDone)
                                                              public class MusicSelectionUI: MonoBehaviour
        float progress = Mathf.Clamp01(
 operation.progress/0.9f);
                                                                 private MusicManager musicManager;
        progressBar.value = progress;
        yield return null;
                                                                 private void Start()
                                                                   musicManager =
                                                              FindObjectOfType<MusicManager>();
                                                                 public void SelectMusic(int index)
Файл MusicManager.cs:
 using System.Collections;
                                                                   if (musicManager != null)
 using System.Collections.Generic;
 using UnityEngine;
                                                                     musicManager.PlayMusic(index);
```

```
if ((int.Parse(stroka[0]) >= 70) \&\&
        public void TogglePause()
                                                                                                                                       (int.Parse(stroka[0]) \le 84))
             if (musicManager != null)
                                                                                                                                                     rang.text = "S";
                  musicManager.TogglePause();
                                                                                                                                                 if ((int.Parse(stroka[0]) == 1)||
                                                                                                                                       (int.Parse(stroka[0]) == 5) \parallel (int.Parse(stroka[0]) == 6)
                                                                                                                                       \parallel ((int.Parse(stroka[0]) >= 12) &&(int.Parse(stroka[0])
                                                                                                                                       <= 14)) || ((int.Parse(stroka[0]) >= 22) &&
Файл Oto rang.cs:
                                                                                                                                      (int.Parse(stroka[0]) \le 25)) \parallel ((int.Parse(stroka[0]) > 
   using System.Collections;
                                                                                                                                       35) && (int.Parse(stroka[0]) <= 39)) ||
   using System.Collections.Generic;
                                                                                                                                      ((int.Parse(stroka[0]) >= 51) \&\& (int.Parse(stroka[0]))
   using UnityEngine;
                                                                                                                                       <= 56) || ((int.Parse(stroka[0]) >= 70) &&
   using UnityEngine.UI;
                                                                                                                                       (int.Parse(stroka[0]) \le 76))
   using System.IO;
                                                                                                                                                      images[0].gameObject.SetActive(true);
   public class Oto_rang : MonoBehaviour
                                                                                                                                                 if ((int.Parse(stroka[0]) == 2) \parallel
        public GameObject[] images;
                                                                                                                                       (int.Parse(stroka[0]) == 7) \parallel (int.Parse(stroka[0]) == 8)
                                                                                                                                       \parallel ((int.Parse(stroka[0]) >= 15) && (int.Parse(stroka[0])
        public Text rang;
        void Update()
                                                                                                                                       <= 17) || ((int.Parse(stroka[0]) >= 26) &&
                                                                                                                                       (int.Parse(stroka[0]) \le 29)) \parallel ((int.Parse(stroka[0]) \ge 
                                                                                                                                      40) && (int.Parse(stroka[0]) \leq 44)) ||
             string[] stroka = File.ReadAllLines("ранг.txt");
                                                                                                                                       ((int.Parse(stroka[0]) >= 57) \&\& (int.Parse(stroka[0]))
                                                                                                                                       <= 62) || ((int.Parse(stroka[0]) >= 77) &&
             if (int.Parse(stroka[0]) > 0)
                                                                                                                                      (int.Parse(stroka[0]) \le 83)))
                  rang.gameObject.SetActive(true);
                                                                                                                                                     images[0].gameObject.SetActive(true);
                                                                                                                                                      images[1].gameObject.SetActive(true);
   ((int.Parse(stroka[0])>0)\&\&(int.Parse(stroka[0])<=3))
                                                                                                                                                 if ((int.Parse(stroka[0]) == 3) \parallel
                                                                                                                                      (int.Parse(stroka[0]) == 4) \parallel ((int.Parse(stroka[0]) >= 9)
                  rang.text = "F";
                                                                                                                                       && (int.Parse(stroka[0]) <= 11)) ||
                                                                                                                                       ((int.Parse(stroka[0]) >= 18) \&\& (int.Parse(stroka[0]))
                                                                                                                                       <= 21) || ((int.Parse(stroka[0]) >= 30) &&
             if ((int.Parse(stroka[0]) >= 5) \&\&
   (int.Parse(stroka[0]) \le 9))
                                                                                                                                       (int.Parse(stroka[0]) \le 34)) \parallel ((int.Parse(stroka[0]) \ge 34))
                                                                                                                                      45) && (int.Parse(stroka[0]) \leq 50)) ||
                  rang.text = "E";
                                                                                                                                       ((int.Parse(stroka[0]) \ge 63) \&\& (int.Parse(stroka[0]))
                                                                                                                                       <= 69)) \parallel (int.Parse(stroka[0]) >= 84))
             if ((int.Parse(stroka[0]) >= 12) \&\&
                                                                                                                                                      images[0].gameObject.SetActive(true);
   (int.Parse(stroka[0]) \le 18))
                                                                                                                                                      images[1].gameObject.SetActive(true);
                  rang.text = "D";
                                                                                                                                                      images[2].gameObject.SetActive(true);
             if ((int.Parse(stroka[0]) >= 22) \&\&
   (int.Parse(stroka[0]) \le 30))
                  rang.text = "C";
                                                                                                                                   Файл PausMenu.cs:
                                                                                                                                       using System.Collections;
             if ((int.Parse(stroka[0]) >= 35) \&\&
                                                                                                                                       using System.Collections.Generic;
   (int.Parse(stroka[0]) \le 45))
                                                                                                                                      using UnityEngine;
                                                                                                                                       using UnityEngine.UI;
             {
                  rang.text = "D";
                                                                                                                                       using System.IO;
             if ((int.Parse(stroka[0]) >= 51) \&\&
                                                                                                                                      public class PausMenu: MonoBehaviour
   (int.Parse(stroka[0]) \le 63))
                                                                                                                                           public Image imgObg;
                  rang.text = "A";
                                                                                                                                           public Sprite spriteImage1;
```

```
public Sprite spriteImage2;
                                                                    rb = GetComponent<Rigidbody2D>();
    public Button karta;
    public bool gameIsPause = false;
                                                                  private void FixedUpdate()
    public GameObject pauseMenuUI;
    public void Start()
                                                                    moveInput = Input.GetAxis("Horizontal");
                                                                    rb.velocity = new Vector2(moveInput*speed,
      string[] stroka = File.ReadAllLines("kapta.txt");
                                                                rb.velocity.y);
      if (stroka[0] == "0")
                                                                    if (facingRight==false && moveInput>0)
        imgObg.sprite = spriteImage1;
                                                                    else if (facingRight==true && moveInput<0) {
      else
                                                                Flip(); }
        imgObg.sprite = spriteImage2;
                                                                    if(moveInput==0) {
        karta.gameObject.SetActive(true);
                                                                       anim.SetBool("isRunning", false);
                                                                    else
    public void Button_menu()
                                                                       anim.SetBool("isRunning", true);
        Pause();
   public void Button_resum()
                                                                  private void Update()
      Resume();
                                                                    isGrounded =
                                                               Physics2D.OverlapCircle(freePos.position,
                                                                checkRadius, whatIsGround);
    void Resume()
                                                                    if (isGrounded == true &&
                                                                ((Input.GetKeyDown(KeyCode.Space)) ||
                                                               (Input.GetKeyDown(KeyCode.UpArrow)) ||
      pauseMenuUI.SetActive(false);
                                                               (Input.GetKeyDown(KeyCode.W))))
      Time.timeScale = 1f;
      gameIsPause = false;
                                                                       rb.velocity = Vector2.up * jumpForce;
    void Pause()
                                                                       anim.SetTrigger("taceOf");
    pauseMenuUI.SetActive(true);
                                                                    if (isGrounded==true)
      Time.timeScale = 0f;
      gameIsPause=true;
                                                                       anim.SetBool("isJumping",false);
                                                                    else
Файл playerControler.cs:
 using System.Collections;
                                                                       anim.SetBool("isJumping", true);
 using System.Collections.Generic;
 using UnityEngine;
 using System. Threading. Tasks;
                                                                    if (isGrounded == true &&
 public class playerControler: MonoBehaviour
                                                                ((Input.GetKeyDown(KeyCode.DownArrow) ||
                                                               (Input.GetKeyDown(KeyCode.S)))))
   public float speed;
   public float jumpForce;
                                                                       anim.SetBool("isPrised", true);
   private float moveInput;
   private Rigidbody2D rb;
                                                                    else { anim.SetBool("isPrised", false); }
   private bool facingRight=true;
   private bool isGrounded;
                                                                  void Flip()
   public Transform freePos;
    public float checkRadius;
                                                                    facingRight=!facingRight;
    public LayerMask whatIsGround;
                                                                    Vector3 Scaler = transform.localScale;
   private Animator anim;
                                                                    Scaler.x *= -1;
    private void Start()
                                                                    transform.localScale = Scaler;
      anim = GetComponent<Animator>();
```

```
pauseMenuUI.SetActive(true);
Файл Rang.cs:
 using System.Collections;
                                                               }
 using System.Collections.Generic;
                                                               Файл Restart_game.cs:
 using UnityEngine;
 using System.IO;
                                                                 using System.Collections;
 using UnityEngine.UI;
                                                                 using System.Collections.Generic;
 using UnityEngine.UIElements;
                                                                 using System.IO;
                                                                 using UnityEngine;
 public class Rang: MonoBehaviour
                                                                 using UnityEngine.UI;
                                                                 public class Restart game: MonoBehaviour
   public GameObject pauseMenuUI;
                                                                   public GameObject endddd;
   public Text text1;
   public int heal;
                                                                   public void restart_game()
   public bool b;
                                                                      string[] stroka3 = File.ReadAllLines("время
    public void Start()
                                                                 скрытия.txt");//
                                                                      string[] stroka4= File.ReadAllLines("выполнение
      heal = 10;
                                                                 зд.txt");//
      b=true;
                                                                      string[] stroka6 =
                                                                 File.ReadAllLines("зелья.txt");//
    void Update()
                                                                      string[] stroka7 =
                                                                 File.ReadAllLines("зелья кнопки.txt");//
      if (((FindObjectOfType<Hp>().hp <= 0) || (heal ==
                                                                      string[] stroka8 =
 0) & & (b==true))
                                                                 File.ReadAllLines("использовать.txt");//
                                                                      string[] stroka9 =
        b = false;
                                                                 File.ReadAllLines("карта.txt");//
        StartCoroutine(HandleGameOver());
                                                                      string[] stroka10 =
                                                                 File.ReadAllLines("количество.txt");//
                                                                      string[] stroka11 =
    private IEnumerator HandleGameOver()
                                                                 File.ReadAllLines("крафт.txt");//
                                                                      string[] stroka12 = File.ReadAllLines("Купля-
      // Ждем секунд перед остановкой времени и
                                                                 прдажа.txt");//
 отображением меню
                                                                      string[] stroka13 =
      yield return new WaitForSeconds(1f);
                                                                 File.ReadAllLines("манеты.txt");//
                                                                      string[] stroka14 = File.ReadAllLines("номер
      // Останавливаем время
                                                                 принятого задания.txt");//
      Time.timeScale = 0f;
                                                                      string[] stroka15 = File.ReadAllLines("Номер
      string[] stroka = File.ReadAllLines("ранг.txt");
                                                                 сцены для муз.txt");//
                                                                      string[] stroka16 = File.ReadAllLines("Номер
      if (FindObjectOfType<Hp>().hp < 2)
                                                                 сцены.txt");//
                                                                      string[] stroka17 =
      {
        text1.text = "F";
                                                                 File.ReadAllLines("посещённые локации.txt");//
        stroka[0] = "1";
                                                                      string[] stroka18 =
                                                                 File.ReadAllLines("предметы.txt");//
      }
      else
                                                                      string[] stroka19 =
                                                                 File.ReadAllLines("ранг.txt");//
        text1.text = "E";
                                                                      string[] stroka20 =
        stroka[0] = "5";
                                                                 File.ReadAllLines("растения.txt");//
                                                                      File.WriteAllText("positions.txt", string.Empty);
      using (StreamWriter f = new
                                                                      File.WriteAllText("positions_kotel.txt",
 StreamWriter("ранг.txt", false))
                                                                 string.Empty);
                                                                      File.WriteAllText("positions_kotel_krist.txt",
      {
         for (int k = 0; k < \text{stroka.Length}; k++)
                                                                 string.Empty);
           f.WriteLine(stroka[k]);
                                                                      StreamWriter f3 = new StreamWriter("время
                                                                 скрытия.txt", false);
                                                                      for (int k = 0; k < \text{stroka3.Length}; k++)
      // Отображаем меню паузы
```

```
stroka3[k] = "03.05.2025 17:41:54";
       f3.WriteLine(stroka3[k]);
                                                                          stroka11[k] = 0.ToString();
                                                                          f11.WriteLine(stroka11[k]);
     f3.Close();
     StreamWriter f4 = new
                                                                       f11.Close():
StreamWriter("выполнение зд.txt", false);
                                                                       StreamWriter f12 = new StreamWriter("Купля-
     for (int k = 0; k < \text{stroka4.Length}; k++)
                                                                  прдажа.txt", false);
                                                                       for (int k = 0; k < stroka12.Length; k++)
       if ((k>=0)\&\& (k<6))
       { stroka4[k] = 0.ToString(); }
                                                                          stroka12[k] = 0.ToString();
                                                                          f12.WriteLine(stroka12[k]);
       else
       { stroka4[k] = 1.ToString(); }
       f4.WriteLine(stroka4[k]);
                                                                       f12.Close();
                                                                       StreamWriter f13= new
     f4.Close();
                                                                  StreamWriter("манеты.txt", false);
     StreamWriter f6 = new StreamWriter("зелья.txt",
                                                                       for (int k = 0; k < \text{stroka13.Length}; k++)
false);
     for (int k = 0; k < \text{stroka6.Length}; k++)
                                                                          stroka13[k] = 0.ToString();
                                                                          f13.WriteLine(stroka13[k]);
       stroka6[k] = 0.ToString();
       f6.WriteLine(stroka6[k]);
                                                                       f13.Close();
                                                                       StreamWriter f14 = new StreamWriter("номер
     f6.Close();
                                                                  принятого задания.txt", false);
     StreamWriter f7 = new
                                                                       for (int k = 0; k < \text{stroka14.Length}; k++)
StreamWriter("зелья_кнопки.txt", false);
     for (int k = 0; k < \text{stroka7.Length}; k++)
                                                                          stroka14[k] = 0.ToString();
                                                                          f14.WriteLine(stroka14[k]);
       stroka7[k] = 0.ToString();
       f7.WriteLine(stroka7[k]);
                                                                       f14.Close();
                                                                       StreamWriter f15 = new StreamWriter("Номер
    f7.Close();
                                                                  сцены для муз.txt", false);
     StreamWriter f8 = new
                                                                       for (int k = 0; k < \text{stroka15.Length}; k++)
StreamWriter("использовать.txt", false);
     for (int k = 0; k < \text{stroka8.Length}; k++)
                                                                          stroka15[k] = 0.ToString();
                                                                          f15.WriteLine(stroka15[k]);
       stroka8[k] = 0.ToString();
       f8.WriteLine(stroka8[k]);
                                                                       f15.Close();
                                                                       StreamWriter f16 = new StreamWriter("Номер
     f8.Close();
                                                                  сцены.txt", false);
     StreamWriter f9 = new StreamWriter("карта.txt",
                                                                       for (int k = 0; k < \text{stroka16.Length}; k++)
false);
     for (int k = 0; k < \text{stroka9.Length}; k++)
                                                                          stroka16[k] = 0.ToString();
                                                                          f16.WriteLine(stroka16[k]);
       stroka9[k] = 0.ToString();
       f9.WriteLine(stroka9[k]);
                                                                       f16.Close();
                                                                       StreamWriter f17 = new
     f9.Close();
                                                                  StreamWriter("посещённые локации.txt", false);
                                                                       for (int k = 0; k < \text{stroka17.Length}; k++)
     StreamWriter f10 = new
StreamWriter("количество.txt", false);
                                                                          stroka17[k] = 0.ToString();
     for (int k = 0; k < stroka10.Length; k++)
                                                                          f17.WriteLine(stroka17[k]);
                                                                       f17.Close();
       stroka10[k] = 0.ToString();
       f10.WriteLine(stroka10[k]);
                                                                       StreamWriter f18 = new
                                                                  StreamWriter("предметы.txt", false);
     f10.Close();
                                                                       for (int k = 0; k < \text{stroka18.Length}; k++)
     StreamWriter f11 = new
                                                                          stroka18[k] = 0.ToString();
StreamWriter("крафт.txt", false);
     for (int k = 0; k < \text{strokal 1.Length}; k++)
                                                                          f18.WriteLine(stroka18[k]);
```

```
string[] stroka =
      f18.Close():
                                                                 File.ReadAllLines("растения.txt");
      StreamWriter f19 = new StreamWriter("ранг.txt",
 false);
                                                                      int x = 0;
      for (int k = 0; k < \text{stroka19.Length}; k++)
                                                                      for (int j = 0; j < \text{stroka.Length}; j++)
        stroka19[k] = 0.ToString();
                                                                         if ((buttonName == bt[0 + x].name) \parallel
        f19.WriteLine(stroka19[k]);
                                                                 (buttonName == bt[1 + x].name))
      f19.Close();
                                                                           stroka[j] = (int.Parse(stroka[j]) +
      StreamWriter f20 = new
                                                                 1).ToString();
 StreamWriter("растения.txt", false);
      for (int k = 0; k < stroka20.Length; k++)
                                                                         x += 2;
         stroka20[k] = 0.ToString();
                                                                       StreamWriter f = new
                                                                 StreamWriter("растения.txt", false);
        f20.WriteLine(stroka20[k]);
                                                                       for (int i = 0; i < stroka.Length; i++)
      f20.Close();
                                                                         f.WriteLine(stroka[i]);
      Time.timeScale = 1f;
      if (endddd.gameObject != null)
                                                                       f.Close();//счёт собранных растений
                                                                       string[] vremia = File.ReadAllLines("время
        endddd.gameObject.SetActive(false);
                                                                 скрытия.txt");
                                                                      DateTime d = DateTime.Now;
                                                                       for (int k=0; k < vremia.Length; k++)
 }
Файл sborrastenii.cs:
                                                                         if (buttonName == bt[k].name)
 using System.Collections;
 using System.Collections.Generic;
                                                                            vremia[k] = d.ToString();
 using UnityEngine;
                                                                           bt[k].gameObject.SetActive(false);;
 using UnityEngine.UI;
 using System.IO:
 using System;
                                                                       StreamWriter f1 = new StreamWriter("время
                                                                 скрытия.txt", false);
 public class sborrastenii: MonoBehaviour
                                                                       for (int i = 0; i < vremia.Length; i++)
   public Button[] bt;
                                                                         f1.WriteLine(vremia[i]);
    void Update()
                                                                      f1.Close();
      string[] vremia = File.ReadAllLines("время
 скрытия.txt");
                                                                Файл Spravka.cs:
      for (int i = 0; i < vremia. Length; i++)
                                                                 using System.Collections;
         DateTime d1 = DateTime.Now;
                                                                 using System.Collections.Generic;
        TimeSpan raznosti = d1 -
                                                                 using System. Diagnostics;
 DateTime.Parse(vremia[i]);
                                                                 using UnityEngine;
        if (raznosti.TotalMinutes > 15)
                                                                 public class Spravka: MonoBehaviour
           bt[i].gameObject.SetActive(true);
                                                                    public string chmFilePath;
                                                                    public void OpenCHM()
   public void ButtonClik(Button clickedButton)
                                                                      Process.Start(chmFilePath);
      string buttonName = clickedButton.name;//
 проверка какая из одинаковых кнопок нажата
                                                                Файл vosvr_vr.cs:
```

```
using System.Collections;
                                                                           stroka1[8] = (int.Parse(stroka1[8]) -
                                                                   2).ToString();
 using System.Collections.Generic;
 using UnityEngine;
                                                                           stroka1[9] = (int.Parse(stroka1[9]) -
 public class vosvr_vr : MonoBehaviour
                                                                   2).ToString();
                                                                           StreamWriter f1 = new
    public void Button_vosvr()
                                                                   StreamWriter("растения.txt", false);
      Time.timeScale = 1f;
                                                                           for (int k = 0; k < \text{stroka1.Length}; k++)
                                                                             f1.WriteLine(stroka1[k]);
Файл zadanieposborules.cs:
 using System.Collections;
                                                                           f1.Close();
 using System.Collections.Generic;
 using UnityEngine;
                                                                           stroka[0] = "0";
 using UnityEngine.UI;
                                                                           StreamWriter f = new StreamWriter("номер
 using System.IO;
                                                                   принятого задания.txt", false);
 public class zadanieposborules: MonoBehaviour
                                                                           for (int k = 0; k < \text{stroka.Length}; k++)
   public GameObject z1;
                                                                             f.WriteLine(stroka[k]);
   public GameObject z2;
    public Text[] text;
                                                                           f.Close();
   public Image imgObg;
                                                                           string[] stroka2 =
   public Sprite spriteImage1;
                                                                   File.ReadAllLines("манеты.txt");
   public Sprite spriteImage2;
                                                                           stroka2[0] = (int.Parse(stroka2[0]) +
    public Button karta;
                                                                   100).ToString();
    public GameObject pobeda;
    void Start()
                                                                           StreamWriter f2 = new
                                                                   StreamWriter("манеты.txt", false);
      string[] stroka = File.ReadAllLines("номер
                                                                           for (int k = 0; k < \text{stroka2.Length}; k++)
 принятого задания.txt");
      if (int.Parse(stroka[0]) == 1)
                                                                             f2.WriteLine(stroka2[k]);
         z1.gameObject.SetActive(true);
                                                                           f2.Close();
                                                                           string[] stroka3 =
      if (int.Parse(stroka[0]) == 2)
                                                                   File.ReadAllLines("выполнение зд.txt");
                                                                           stroka3[0] = (int.Parse(stroka3[0]) +
         z2.gameObject.SetActive(true);
                                                                   1).ToString();
                                                                           StreamWriter f3 = new
    void Update()
                                                                   StreamWriter("выполнение зд.txt", false);
                                                                           for (int k = 0; k < \text{stroka3.Length}; k++)
      string[] stroka = File.ReadAllLines("номер
                                                                             f3.WriteLine(stroka3[k]);
 принятого задания.txt");
      string[] stroka1 =
 File.ReadAllLines("растения.txt");
                                                                           f3.Close();
      if (int.Parse(stroka[0]) == 1)
                                                                           string[] stroka4 =
                                                                   File.ReadAllLines("ранг.txt");
         text[0].text = stroka1[8] + "/2";
                                                                           stroka4[0] = (int.Parse(stroka4[0]) +
         text[1].text = stroka1[9] + "/2";
                                                                   1).ToString();
      if (int.Parse(stroka[0]) == 2)
                                                                           StreamWriter f4 = new
                                                                   StreamWriter("ранг.txt", false);
         text[2].text = stroka1[7] + "/2";
                                                                           for (int k = 0; k < \text{stroka4.Length}; k++)
         text[3].text = stroka1[12] + "/2";
                                                                             f4.WriteLine(stroka4[k]);
      if ((int.Parse(stroka[0]) == 1) \&\&
 (int.Parse(stroka1[8]) >= 2) \&\& (int.Parse(stroka1[9])
                                                                           f4.Close();
 >= 2))
                                                                           pobeda.gameObject.SetActive(true);
```

```
string[] stroka5 =
File.ReadAllLines("карта.txt");
                                                                            f3.WriteLine(stroka3[k]);
       if (stroka5[0] == "0")
                                                                         f3.Close();
          imgObg.sprite = spriteImage1;
                                                                         string[] stroka4 =
                                                                  File.ReadAllLines("ранг.txt");
       else
                                                                         stroka4[0] = (int.Parse(stroka4[0]) +
          imgObg.sprite = spriteImage2;
                                                                  2).ToString();
          karta.gameObject.SetActive(true);
                                                                         StreamWriter f4 = new
                                                                  StreamWriter("ранг.txt", false);
        StartCoroutine(stop());
                                                                         for (int k = 0; k < \text{stroka4.Length}; k++)
     if ((int.Parse(stroka[0]) == 2) \&\&
(int.Parse(stroka1[7]) >= 2) \&\& (int.Parse(stroka1[12])
                                                                            f4.WriteLine(stroka4[k]);
>= 2)
                                                                         f4.Close();
       stroka1[7] = (int.Parse(stroka1[7]) -
                                                                         pobeda.gameObject.SetActive(true);
2).ToString();
       stroka1[12] = (int.Parse(stroka1[12]) -
                                                                         string[] stroka5 =
                                                                  File.ReadAllLines("карта.txt");
2).ToString();
                                                                         if (stroka5[0] == "0")
       StreamWriter f1 = new
StreamWriter("растения.txt", false);
                                                                            imgObg.sprite = spriteImage1;
       for (int k = 0; k < \text{stroka1.Length}; k++)
                                                                         else
          f1.WriteLine(stroka1[k]);
                                                                            imgObg.sprite = spriteImage2;
                                                                            karta.gameObject.SetActive(true);
       f1.Close();
       stroka[0] = 0.ToString();
                                                                         StartCoroutine(stop());
       StreamWriter f = new StreamWriter("номер
принятого задания.txt", false);
       for (int k = 0; k < \text{stroka.Length}; k++)
                                                                    private IEnumerator stop()
          f.WriteLine(stroka[k]);
                                                                       yield return new WaitForSeconds(3f);
       f.Close();
                                                                       Time.timeScale = 0f;
       string[] stroka2 =
File.ReadAllLines("манеты.txt");
       stroka2[0] = (int.Parse(stroka2[0]) +
                                                                  }
100).ToString();
                                                                Файл zadani_sell.cs:
       StreamWriter f2 = new
                                                                 using System.Collections;
StreamWriter("манеты.txt", false);
                                                                  using System.Collections.Generic;
       for (int k = 0; k < \text{stroka2.Length}; k++)
                                                                  using UnityEngine;
                                                                  using UnityEngine.UI;
          f2.WriteLine(stroka2[k]);
                                                                  using System.IO;
       f2.Close();
                                                                  public class zadani_sell : MonoBehaviour
       string[] stroka3 =
                                                                    public GameObject z1;
File.ReadAllLines("выполнение зд.txt");
                                                                    public GameObject z2;
                                                                    public Text[] text;
       stroka3[2] = (int.Parse(stroka3[2]) +
1).ToString();
                                                                    public Image imgObg;
       StreamWriter f3 = new
                                                                    public Sprite spriteImage1;
                                                                    public Sprite spriteImage2;
StreamWriter("выполнение зд.txt", false);
       for (int k = 0; k < \text{stroka3.Length}; k++)
                                                                    public Sprite spriteImage3;
```

```
public Sprite spriteImage4;
                                                                         for (int k = 0; k < \text{stroka.Length}; k++)
  public Button karta;
                                                                            f.WriteLine(stroka[k]);
  public GameObject pobeda;
                                                                         f.Close();
  public Переход perehod;
                                                                         string[] stroka2 =
                                                                  File.ReadAllLines("манеты.txt");
  void Start()
                                                                         stroka2[0] = (int.Parse(stroka2[0]) +
                                                                  225).ToString();
     string[] stroka = File.ReadAllLines("номер
                                                                         StreamWriter f2 = new
принятого задания.txt");
                                                                  StreamWriter("манеты.txt", false);
     if (int.Parse(stroka[0]) == 4)
                                                                         for (int k = 0; k < stroka2.Length; k++)
       z1.gameObject.SetActive(true);
                                                                            f2.WriteLine(stroka2[k]);
       perehod.levelToLoad = 5;
                                                                         f2.Close();
     if (int.Parse(stroka[0]) == 5)
                                                                         string[] stroka3 =
       z2.gameObject.SetActive(true);
                                                                  File.ReadAllLines("выполнение зд.txt");
       perehod.levelToLoad = 5;
                                                                         stroka3[3] = (int.Parse(stroka3[3]) +
                                                                  1).ToString();
  }
                                                                         StreamWriter f3 = new
                                                                 StreamWriter("выполнение зд.txt", false);
                                                                         for (int k = 0; k < \text{stroka3.Length}; k++)
  void Update()
     string[] stroka = File.ReadAllLines("номер
                                                                            f3.WriteLine(stroka3[k]);
принятого задания.txt");
                                                                         f3.Close();
     string[] stroka1 = File.ReadAllLines("зелья.txt");
                                                                         string[] stroka4 =
     if (int.Parse(stroka[0]) == 4)
                                                                 File.ReadAllLines("ранг.txt");
                                                                         stroka4[0] = (int.Parse(stroka4[0]) +
       text[0].text = stroka1[1] + "/1";
                                                                  2).ToString();
                                                                         StreamWriter f4 = new
     if (int.Parse(stroka[0]) == 5)
                                                                  StreamWriter("ранг.txt", false);
                                                                         for (int k = 0; k < \text{stroka4.Length}; k++)
       text[1].text = stroka1[2] + "/1";
                                                                            f4.WriteLine(stroka4[k]);
     if ((int.Parse(stroka[0]) == 4) \&\&
                                                                         f4.Close();
(int.Parse(stroka1[1]) >= 1))
       stroka1[1] = (int.Parse(stroka1[1]) -
                                                                         pobeda.gameObject.SetActive(true);
1).ToString();
                                                                         string[] stroka5 =
       StreamWriter f1 = new
                                                                 File.ReadAllLines("карта.txt");
StreamWriter("зелья.txt", false);
                                                                         if (stroka5[0] == "0")
       for (int k = 0; k < \text{stroka1.Length}; k++)
                                                                            imgObg.sprite = spriteImage1;
          f1.WriteLine(stroka1[k]);
                                                                         else
       f1.Close();
                                                                            imgObg.sprite = spriteImage2;
       stroka[0] = "0";
                                                                            karta.gameObject.SetActive(true);
       StreamWriter f = new StreamWriter("номер
принятого задания.txt", false);
```

```
stroka4[0] = (int.Parse(stroka4[0]) +
        StartCoroutine(stop());
                                                                 2).ToString();
     }
                                                                         StreamWriter f4 = new
                                                                 StreamWriter("ранг.txt", false);
     if ((int.Parse(stroka[0]) == 5) \&\&
                                                                         for (int k = 0; k < \text{stroka4.Length}; k++)
(int.Parse(stroka1[2]) >= 1))
                                                                           f4.WriteLine(stroka4[k]);
                                                                         f4.Close();
                                                                         pobeda.gameObject.SetActive(true);
       stroka1[2] = (int.Parse(stroka1[2]) -
1).ToString();
                                                                         string[] stroka5 =
                                                                 File.ReadAllLines("карта.txt");
                                                                         if (stroka5[0] == "0")
       StreamWriter f1 = new
StreamWriter("зелья.txt", false);
       for (int k = 0; k < \text{stroka1.Length}; k++)
                                                                           imgObg.sprite = spriteImage3;
          f1.WriteLine(stroka1[k]);
                                                                         else
       f1.Close();
                                                                           imgObg.sprite = spriteImage4;
                                                                           karta.gameObject.SetActive(true);
       stroka[0] = 0.ToString();
       StreamWriter f = new StreamWriter("номер
принятого задания.txt", false);
       for (int k = 0; k < \text{stroka.Length}; k++)
                                                                         StartCoroutine(stop());
          f.WriteLine(stroka[k]);
       f.Close();
       string[] stroka2 =
                                                                    private IEnumerator stop()
File.ReadAllLines("манеты.txt");
                                                                       yield return new WaitForSeconds(3f);
       stroka2[0] = (int.Parse(stroka2[0]) +
200).ToString();
                                                                       Time.timeScale = 0f;
       StreamWriter f2 = new
StreamWriter("манеты.txt", false);
                                                                    public void start()
       for (int k = 0; k < \text{stroka2.Length}; k++)
                                                                       Time.timeScale = 1f;
          f2.WriteLine(stroka2[k]);
       f2.Close();
                                                                 }
                                                                Файл zagruska.cs:
       string[] stroka3 =
                                                                 using System.Collections;
File.ReadAllLines("выполнение зд.txt");
                                                                 using System.Collections.Generic;
       stroka3[4] = (int.Parse(stroka3[4]) +
1).ToString();
                                                                 using UnityEngine;
                                                                 using UnityEngine.SceneManagement;
       StreamWriter f3 = new
                                                                 using UnityEngine.UI;
StreamWriter("выполнение зд.txt", false);
       for (int k = 0; k < stroka3.Length; k++)
                                                                 public class zagruska: MonoBehaviour
          f3.WriteLine(stroka3[k]);
                                                                    private Animator anim;
       f3.Close();
                                                                    void Start()
       string[] stroka4 =
File.ReadAllLines("ранг.txt");
                                                                      StartCoroutine(gar1());
```

```
void Start()
    private IEnumerator gar1()
                                                                      player =
      yield return new WaitForSeconds(7.8f);
                                                                 GameObject.FindWithTag("Player").transform;
                                                                      initialPosition = transform.position;
      anim = GetComponent<Animator>();
                                                                      animator = GetComponent<Animator>(); //
      SceneManager.LoadScene(1);
                                                                 Получаем компонент Animator
      anim.SetTrigger("fade");
    }
                                                                   void Update()
  }
                                                                      if (player.position.x > x && player.position.y < y
                                                                 && player.position.x < x1)
Файл Переход.cs:
 using System.Collections;
                                                                        MoveTowardsPlayer();
 using System.Collections.Generic;
 using UnityEngine;
                                                                      else
 using UnityEngine.SceneManagement;
                                                                        ReturnToInitialPosition();
 public class Переход: MonoBehaviour
    private Animator anim;
    public int levelToLoad;
                                                                   void MoveTowardsPlayer()
    private void Start()
                                                                      Vector2 direction = (player.position -
      anim = GetComponent<Animator>();
                                                                 transform.position).normalized;
                                                                      Vector3 targetPosition = transform.position + new
                                                                 Vector3(direction.x * moveSpeed * Time.deltaTime, 0,
    public void FadeToLevel()
      anim.SetTrigger("fade");
                                                                      if (IsWithinZone(targetPosition))
                                                                        transform.position = targetPosition;
    public void OnFadeComplete()
                                                                        UpdateAnimationAndRotation(direction); //
                                                                 Обновляем анимацию и поворот
      SceneManager.LoadScene(levelToLoad);
}
                                                                   bool IsWithinZone(Vector3 position)
Файл враг.cs:
                                                                      float halfWidth = zoneSize.x / 2;
                                                                      float halfHeight = zoneSize.y / 2;
 using System.Collections;
 using System.Collections.Generic;
                                                                      return position.x >= initialPosition.x - halfWidth
                                                                 && position.x <= initialPosition.x + halfWidth &&
 using UnityEngine;
 using static UnityEngine.RuleTile.TilingRuleOutput;
                                                                          position.y >= initialPosition.y - halfHeight
                                                                 && position.y <= initialPosition.y + halfHeight;
 public class враг: MonoBehaviour
   [SerializeField] private float x;
                                                                   void ReturnToInitialPosition()
    [SerializeField] private float x1;
    [SerializeField] private float y;
                                                                      if (Vector3.Distance(transform.position,
    [SerializeField] private float moveSpeed = 2f;
                                                                 initialPosition) > 0.1f
    [SerializeField] private Vector2 zoneSize = new
                                                                        if (transform.position.x > initialPosition.x)
  Vector2(5f, 5f);
    private UnityEngine.Transform player;
    [SerializeField] private bool playerInRange = false;
                                                                           transform.localScale = new Vector3(1, 1, 1);
    private Vector3 initialPosition;
    private Animator animator; // Аниматор врага
                                                                        if (transform.position.x < initialPosition.x)
```

```
float knockbackX = knockbackDirection.x *
         transform.localScale = new Vector3(-1, 1, 1);
                                                             400f; // Настройте силу отталкивания по вашему
                                                             усмотрению
                                                                       float knockbackY =400f; // Сила
       transform.position =
                                                             отталкивания вверх
Vector3.MoveTowards(transform.position,
initialPosition, moveSpeed * Time.deltaTime);
                                                                       // Применяем силу отталкивания
                                                                       playerRigidbody.AddForce(new
                                                             Vector2(knockbackX, knockbackY),
       if (transform.position == initialPosition)
                                                             ForceMode2D.Impulse);
         animator.SetBool("isRunning", false); //
Остановка анимации бега
                                                             }
     }
  }
                                                            Файл нр.cs:
  private void UpdateAnimationAndRotation(Vector2
                                                             using System.Collections;
                                                             using System.Collections.Generic;
direction)
                                                             using UnityEngine;
    animator.SetBool("isRunning", true); // Запуск
                                                             using UnityEngine.UI;
анимации бега
    // Поворачиваем врага в сторону движения
                                                             public class нр: MonoBehaviour
    if (direction.x > 0)
                                                                public int hp;
                                                                public Image a;
       transform.localScale = new Vector3(-1, 1, 1); //
Поворот вправо
                                                                public Image b;
                                                                public void boy()
    else if (direction.x < 0)
       transform.localScale = new Vector3(1, 1, 1); //
                                                                  hp--;
Поворот влево
                                                                  if (hp == 1)
                                                                  { a.gameObject.SetActive(false); }
                                                                  if (hp == 0)
                                                                  { b.gameObject.SetActive(false); }
  private void OnCollisionEnter2D(Collision2D
collision)
                                                             }
    if (collision.collider.CompareTag("Player"))
                                                            Файл переходизмузыки.cs:
                                                             using System.Collections;
       нр playerController =
                                                             using System.Collections.Generic;
collision.collider.GetComponent<Hp>();
                                                             using UnityEngine;
       playerController.boy(); // Наносим урон
                                                             using UnityEngine.SceneManagement;
                                                             using System.IO;
игроку
                                                             public class переходизмузыки: MonoBehaviour
       // Получаем Rigidbody2D игрока
       Rigidbody2D playerRigidbody =
                                                                private Animator anim;
collision.collider.GetComponent<Rigidbody2D>();
       if (playerRigidbody != null)
                                                                private void Start()
         // Вычисляем направление от врага к
                                                                  anim = GetComponent<Animator>();
игроку
         Vector2 knockbackDirection =
(collision.transform.position -
                                                                public void FadeToLevel()
transform.position).normalized;
                                                                  anim.SetTrigger("fade");
         // Устанавливаем компонент Ү для
отбрасывания
                                                                public void OnFadeComplete()
```

```
using System.IO;
      //считывание номера сцены из файла
                                                              public class переходсзаписью: MonoBehaviour
      StreamReader f = new StreamReader("Homep
 сцены для муз.txt");
                                                                private Animator anim;
      int n = int.Parse(f.ReadToEnd());
                                                                public int levelToLoad;
      f.Close();
                                                                private void Start()
      SceneManager.LoadScene(n);
                                                                   anim = GetComponent<Animator>();
   public void OnFadeComplete1()
                                                                public void FadeToLevel()
                                                                   anim.SetTrigger("fade");
      //считывание номера сцены из файла
      StreamReader f = new StreamReader("Homep
                                                                   //получение индекса
                                                                   Scene currentScene
 сцены.txt");
      int n = int.Parse(f.ReadToEnd());
                                                              =SceneManager.GetActiveScene();
      f.Close();
                                                                   int sceneIndex =currentScene.buildIndex;
      SceneManager.LoadScene(n);
                                                                   StreamWriter f = new StreamWriter("Номер
                                                              сцены.txt",false);
                                                                   f.WriteLine(sceneIndex.ToString() );
                                                                   f.Close();
Файл переходизторговли.cs:
 using System.Collections;
 using System.Collections.Generic;
                                                                public void FadeToLevel1()
 using UnityEngine;
 using UnityEngine.SceneManagement;
                                                                   anim.SetTrigger("fade");
 using System.IO;
 public class переходизторговли: MonoBehaviour
                                                                   //получение индекса
                                                                   Scene currentScene =
    private Animator anim;
                                                              SceneManager.GetActiveScene();
                                                                   int sceneIndex = currentScene.buildIndex;
    private void Start()
                                                                   StreamWriter f = new StreamWriter("Номер
                                                              сцены для муз.txt", false);
      anim = GetComponent<Animator>();
                                                                   f.WriteLine(sceneIndex.ToString());
                                                                   f.Close();
    public void FadeToLevel()
                                                                public void OnFadeComplete()
      anim.SetTrigger("fade");
                                                                   SceneManager.LoadScene(levelToLoad);
   public void OnFadeComplete()
         //считывание номера сцены из файла
      StreamReader f = new StreamReader("Номер
 сцены.txt");
      int n = int.Parse(f.ReadToEnd());
                                                             Файл расшифровка.cs:
      f.Close();
                                                              using System.Collections;
      SceneManager.LoadScene(n);
                                                              using System.Collections.Generic;
                                                              using UnityEngine;
                                                              using UnityEngine.UI;
}
                                                              using System.IO;
Файл переходсзаписью.cs:
                                                              public class расшифровка: MonoBehaviour
 using System.Collections;
 using System.Collections.Generic;
                                                                public Text text1;
 using UnityEngine;
                                                                public Text text2;
 using UnityEngine.SceneManagement;
                                                                public Image imgObg;
```

```
if (stroka[0] == "0")
  public Sprite spriteImage1;
  public Sprite spriteImage2;
                                                                          imgObg.sprite = spriteImage1;
  public Image imgObg2;
  public Sprite spriteImage21;
                                                                        else
  public Sprite spriteImage22;
  public Button karta;
                                                                          imgObg.sprite = spriteImage2;
                                                                          karta.gameObject.SetActive(true);
  public GameObject pobeda;
  public GameObject porajenie;
  public void Gotovo()
                                                                     else {
                                                                        porajenie.SetActive(true);
     if ((text1.text == "Великие начинания даже не
надо обдумывать, надо взяться да дело, иначе,
                                                                        string[] stroka =
                                                                File.ReadAllLines("карта.txt");
заметив трудность отступишь.") && (text2.text == "-
Гай Юлий Цезарь"))
                                                                        if (stroka[0] == "0")
       string[] stroka1 =
                                                                          imgObg2.sprite = spriteImage21;
File.ReadAllLines("манеты.txt");
       stroka1[0] = (int.Parse(stroka1[0]) +
                                                                        else
100).ToString();
                                                                          imgObg2.sprite = spriteImage22;
       StreamWriter f1 = new
                                                                          karta.gameObject.SetActive(true);
StreamWriter("манеты.txt", false);
       for (int k = 0; k < \text{stroka1.Length}; k++)
          f1.WriteLine(stroka1[k]);
                                                                     Time.timeScale = 0f;
       f1.Close();
       string[] stroka2 =
                                                                   public void zanovo()
File.ReadAllLines("выполнение зд.txt");
       stroka2[1] = (int.Parse(stroka2[1]) +
                                                                     porajenie.SetActive(false);
                                                                     karta.gameObject.SetActive(false);
1).ToString();
                                                                     text1.text = "";
       StreamWriter f2 = new
                                                                     text2.text = "";
StreamWriter("выполнение зд.txt", false);
                                                                     Time.timeScale = 1f;
       for (int k = 0; k < \text{stroka2.Length}; k++)
          f2.WriteLine(stroka2[k]);
                                                                Файл Bullet_enemy.cs:
       f2.Close();
                                                                using System;
                                                                using System.Collections;
                                                                using System.Collections.Generic;
       string[] stroka3 =
File.ReadAllLines("ранг.txt");
                                                                using UnityEngine;
       stroka3[0] = (int.Parse(stroka3[0]) +
1).ToString();
                                                                public class Bullet: MonoBehaviour
       StreamWriter f3 = new
                                                                   public float speed;
StreamWriter("ранг.txt", false);
                                                                   public float lifetime;
       for (int k = 0; k < \text{stroka3.Length}; k++)
                                                                   public float lifetime1;
                                                                   public float distance;
          f3.WriteLine(stroka3[k]);
                                                                   public int damage;
                                                                   public LayerMask whatIsSolid; // Лейер для
                                                                "огонёк"
       f3.Close();
       pobeda.SetActive(true);
                                                                   public LayerMask whatIsWrag; // Лейер для
       string[] stroka =
                                                                 "кислота_паук"
File.ReadAllLines("карта.txt");
```

```
private Vector2 direction; // Направление движения
пули
                                                                   // Двигаем пулю в заданном направлении
                                                                   transform.Translate(direction * speed *
  public GameObject bulletEffect;
                                                              Time.deltaTime);
  private void Start()
    // Уничтожаем пулю через заданное время
                                                              Файл casanije_wrag.cs:
    Destroy(gameObject, lifetime1);
                                                              using System.Collections;
                                                              using System.Collections.Generic;
                                                              using UnityEngine;
  public void SetDirection(Vector2 dir)
                                                              public class casanije_wrag : MonoBehaviour
    direction = dir; // Устанавливаем направление
                                                                public int x = 0;
                                                                private void OnCollisionEnter2D(Collision2D
  private void Update()
                                                              collision)
                                                                   if (collision.collider.CompareTag("Player"))
     RaycastHit2D hitInfo =
Physics2D.Raycast(transform.position, direction,
                                                                     нр playerController =
distance);
                                                              collision.collider.GetComponent<+p>();
    if (hitInfo.collider != null)
                                                                     playerController.boy(); // Наносим урон
                                                              игроку
       string objectName = gameObject.name;
                                                                     if (x == 0) {
       // Проверяем, если это "огонёк(Clone)"
                                                                     // Получаем Rigidbody2D игрока
       if (objectName == "огонёк(Clone)" && ((1 <<
                                                                     Rigidbody2D playerRigidbody =
hitInfo.collider.gameObject.layer) & whatIsSolid) != 0)
                                                              collision.collider.GetComponent<Rigidbody2D>();
                                                                     if (playerRigidbody != null)
         if (hitInfo.collider.CompareTag("Enemy"))
                                                                       // Вычисляем направление от врага к
                                                              игроку
hitInfo.collider.GetComponent<Enemy>().TakeDamag
                                                                        Vector2 knockbackDirection =
                                                              (collision.transform.position -
e(damage);
                                                              transform.position).normalized;
                                                                       float knockbackX = knockbackDirection.x *
         Instantiate(bulletEffect, transform.position,
                                                              400f;
Quaternion.identity);
                                                                        float knockbackY = 400f;
                                                                       playerRigidbody.AddForce(new
         Destroy(gameObject);
                                                              Vector2(knockbackX, knockbackY),
                                                              ForceMode2D.Impulse);
       // Проверяем, если это
"кислота паук(Clone)"
       if ((objectName == "кислота_паук(Clone)"||
objectName == "огонь_сал(Clone)") && ((1 <<
hitInfo.collider.gameObject.layer) & whatIsWrag) != 0)
         if (hitInfo.collider.CompareTag("Player"))
                                                              Файл result z7.cs:
                                                              using System.Collections;
hitInfo.collider.GetComponent<Hp>().boy();
                                                              using System.Collections.Generic;
                                                              using UnityEngine;
                                                              using UnityEngine.UI;
         Instantiate(bulletEffect, transform.position,
                                                              using System.IO;
Quaternion.identity);
         Destroy(gameObject);
                                                              public class result_z7 : MonoBehaviour
                                                                public Button gild;
     }
```

```
public Button kart;
                                                                         StreamWriter f = new StreamWriter("номер
                                                                 принятого задания.txt", false);
  public GameObject pobeda;
                                                                         for (int k = 0; k < \text{stroka.Length}; k++)
  public GameObject belca;
                                                                           f.WriteLine(stroka[k]);
  void Start()
                                                                         f.Close();
     string[] stroka_v =
File.ReadAllLines("враги.txt");
                                                                         string[] stroka2 =
     if (stroka_v[2] == 1.ToString())
                                                                 File.ReadAllLines("манеты.txt");
                                                                         stroka2[0] = (int.Parse(stroka2[0]) +
       string[] stroka_z = File.ReadAllLines("номер
                                                                 400).ToString();
принятого задания.txt");
       stroka_z[0] = 7.ToString();
                                                                         StreamWriter f2 = new
       StreamWriter fz = new StreamWriter("номер
                                                                 StreamWriter("манеты.txt", false);
принятого задания.txt", false);
                                                                         for (int k = 0; k < \text{stroka2.Length}; k++)
       for (int k = 0; k < stroka_z.Length; k++)
                                                                           f2.WriteLine(stroka2[k]);
          fz.WriteLine(stroka_z[k]);
                                                                         f2.Close();
       fz.Close();
                                                                         string[] stroka3 =
     }
                                                                 File.ReadAllLines("выполнение зд.txt");
     else {
       Destroy(belca.gameObject);
                                                                         stroka3[6] = 1.ToString();
                                                                         StreamWriter f3 = new
                                                                 StreamWriter("выполнение зд.txt", false);
                                                                         for (int k = 0; k < \text{stroka3.Length}; k++)
     string[] stroka = File.ReadAllLines("номер
принятого задания.txt");
                                                                           f3.WriteLine(stroka3[k]);
     if( stroka[0] == 7.ToString())
                                                                         f3.Close();
       gild.gameObject.SetActive(true);
                                                                         string[] stroka4 =
                                                                 File.ReadAllLines("ранг.txt");
                                                                         stroka4[0] = (int.Parse(stroka4[0]) +
  }
                                                                 3).ToString();
  void Update()
                                                                         StreamWriter f4 = new
                                                                 StreamWriter("ранг.txt", false);
     string[] stroka_z = File.ReadAllLines("номер
                                                                         for (int k = 0; k < \text{stroka4.Length}; k++)
принятого задания.txt");
     string[] stroka_v =
                                                                           f4.WriteLine(stroka4[k]);
File.ReadAllLines("враги.txt");
     if ((stroka_v[2] == 0.ToString())&&(stroka_z[0])
                                                                         f4.Close();
== 7.ToString())
                                                                         StartCoroutine(stop());
     {
       pobeda.gameObject.SetActive(true);
       string[] stroka5 =
File.ReadAllLines("карта.txt");
       if (stroka5[0] == "1")
                                                                   private IEnumerator stop()
          kart.gameObject.SetActive(true);
                                                                      yield return new WaitForSeconds(3f);
                                                                      Time.timeScale = 0f;
       string[] stroka = File.ReadAllLines("номер
принятого задания.txt");
       stroka[0] = 0.ToString();
```

```
Файл result_z8.cs:
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
                                                                    if (FindObjectOfType<Enemy>()!= null)
using UnityEngine.UI;
using System.IO;
                                                                      text_jz.text =
using System;
                                                               (FindObjectOfType<Enemy>().health).ToString();
using static UnityEngine.Rendering.DebugUI.Table;
using UnityEngine.SceneManagement;
public class result z8: MonoBehaviour
                                                                    string[] vremia = File.ReadAllLines("z8-
  public Text text jz;
                                                               9 время.txt");
  public GameObject licor;
                                                                      DateTime d1 = DateTime.Now;
  public Text text_licor;
                                                                      TimeSpan raznosti = d1 -
                                                               DateTime.Parse(vremia[0]);
  public Button gild;
                                                                    if (raznosti.TotalMinutes > 30)
  public Button kart_pob;
  public Button kart_por;
                                                                      licor.gameObject.SetActive(true);
                                                                      text_licor.gameObject.SetActive(false);
  public GameObject pobeda;
  public GameObject porajenie;
                                                                    else
  public GameObject pauc;
  public Image serdca;
                                                                      licor.gameObject.SetActive(false);
                                                                      text_licor.gameObject.SetActive(true);
  bool par = false;
                                                                      DateTime d2 = DateTime.Now;;
  public Image a;
                                                                      text_licor.text = (TimeSpan.FromMinutes(30) -
  public Image b;
                                                               (d1 -
  void Start()
                                                               DateTime.Parse(vremia[0]))).ToString(@"hh\:mm\:ss")
     string[] stroka_v =
File.ReadAllLines("враги.txt");
    if (stroka_v[0] == 1.ToString())
                                                                    if ((FindObjectOfType<Hp>().hp <= 0) && (par
                                                               == false))
       string[] stroka z = File.ReadAllLines("номер
                                                                      porajenie.gameObject.SetActive(true);
принятого задания.txt");
       stroka_z[0] = 8.ToString();
       StreamWriter fz = new StreamWriter("номер
                                                                      string[] stroka5 =
принятого задания.txt", false);
                                                               File.ReadAllLines("карта.txt");
                                                                      if (stroka5[0] == "1")
       for (int k = 0; k < stroka_z.Length; k++)
         fz.WriteLine(stroka_z[k]);
                                                                         kart_por.gameObject.SetActive(true);
       fz.Close();
     }
    else
                                                                      par = true;
                                                                      StartCoroutine(stop());
       Destroy(pauc.gameObject);
       serdca.gameObject.SetActive(false);
    string[] stroka = File.ReadAllLines("номер
                                                                    string[] stroka_z = File.ReadAllLines("номер
принятого задания.txt");
                                                               принятого задания.txt");
                                                                    string[] stroka_v = File.ReadAllLines("z8-9.txt");
    if (stroka[0] == 8.ToString())
                                                                    if ((int.Parse(stroka_v[0]) > 0) && (stroka_z[0] ==
       gild.gameObject.SetActive(true);
                                                               8.ToString()))
                                                                      pobeda.gameObject.SetActive(true);
  void Update()
```

```
string[] stroka5 =
                                                                        StreamWriter f_lik = new StreamWriter("z8-
File.ReadAllLines("карта.txt");
                                                                9.txt", false);
       if (stroka5[0] == "1")
                                                                        for (int k = 0; k < stroka_lik.Length; k++)
          kart_pob.gameObject.SetActive(true);
                                                                          f lik.WriteLine(stroka lik[k]);
                                                                        f_lik.Close();
       string[] stroka = File.ReadAllLines("номер
принятого задания.txt");
                                                                        StartCoroutine(stop());
       stroka[0] = 0.ToString();
       StreamWriter f = new StreamWriter("номер
                                                                   }
принятого задания.txt", false);
                                                                   private IEnumerator stop()
       for (int k = 0; k < stroka.Length; k++)
                                                                      yield return new WaitForSeconds(3f);
                                                                      Time.timeScale = 0f;
          f.WriteLine(stroka[k]);
       f.Close();
                                                                   public void zanovo()
       string[] stroka2 =
File.ReadAllLines("манеты.txt");
                                                                     par = false;
       stroka2[0] = (int.Parse(stroka2[0]) +
                                                                      porajenie.SetActive(false);
500).ToString();
                                                                      FindObjectOfType<Hp>().hp = 2;
                                                                      FindObjectOfType<Enemy>().health = 10;
       StreamWriter f2 = new
                                                                      { a.gameObject.SetActive(true); }
StreamWriter("манеты.txt", false);
                                                                      { b.gameObject.SetActive(true); }
       for (int k = 0; k < \text{stroka2.Length}; k++)
                                                                      Time.timeScale = 1f;
                                                                      SceneManager.LoadScene(19);
          f2.WriteLine(stroka2[k]);
                                                                   }
       f2.Close();
       string[] stroka3 =
File.ReadAllLines("выполнение зд.txt");
       stroka3[7] = 1.ToString();
                                                                Файл result_z9.cs:
                                                                 using System.Collections;
       StreamWriter f3 = new
                                                                 using System.Collections.Generic;
                                                                 using UnityEngine;
StreamWriter("выполнение зд.txt", false);
       for (int k = 0; k < \text{stroka3.Length}; k++)
                                                                 using UnityEngine.UI;
                                                                 using System.IO;
                                                                 using System;
          f3.WriteLine(stroka3[k]);
                                                                 using static UnityEngine.Rendering.DebugUI.Table;
                                                                 using Unity. Visual Scripting;
       f3.Close();
                                                                 using UnityEngine.SceneManagement;
       string[] stroka4 =
File.ReadAllLines("ранг.txt");
                                                                 public class result_z9 : MonoBehaviour
       stroka4[0] = (int.Parse(stroka4[0]) +
3).ToString();
                                                                   public Text text jz;
                                                                   public GameObject licor;
       StreamWriter f4 = new
                                                                   public Text text_licor;
StreamWriter("ранг.txt", false);
       for (int k = 0; k < \text{stroka4.Length}; k++)
                                                                   public Button gild;
                                                                   public Button kart_pob;
          f4.WriteLine(stroka4[k]);
                                                                   public Button kart_por;
       f4.Close();
                                                                   public GameObject pobeda;
                                                                   public GameObject porajenie;
                                                                   public GameObject pauc;
       string[] stroka_lik = File.ReadAllLines("z8-
                                                                   public Image serdca;
9.txt");
       stroka_lik[0]=0.ToString();
```

```
bool par = false;
                                                                        text\_licor.text = (TimeSpan.FromMinutes(45) -
                                                                (d1 -
  public Image a;
  public Image b;
                                                                DateTime.Parse(vremia[1]))).ToString(@"hh\:mm\:ss")
  void Start()
     string[] stroka_v =
                                                                     if ((FindObjectOfType<Hp>().hp <= 0) && (par
File.ReadAllLines("враги.txt");
                                                                == false))
     if (stroka_v[1] == 1.ToString())
                                                                     {
                                                                        porajenie.gameObject.SetActive(true);
       string[] stroka z = File.ReadAllLines("номер
принятого задания.txt");
                                                                        string[] stroka5 =
                                                                File.ReadAllLines("карта.txt");
       stroka z[0] = 9.\text{ToString}();
                                                                       if (stroka5[0] == "1")
       StreamWriter fz = new StreamWriter("номер
принятого задания.txt", false);
       for (int k = 0; k < stroka_z.Length; k++)
                                                                          kart_por.gameObject.SetActive(true);
          fz.WriteLine(stroka_z[k]);
       fz.Close();
                                                                        par = true;
                                                                        StartCoroutine(stop());
     }
     else
                                                                     string[] stroka z = File.ReadAllLines("номер
       Destroy(pauc.gameObject);
                                                                принятого задания.txt");
       serdca.gameObject.SetActive(false);
                                                                     string[] stroka_v = File.ReadAllLines("z8-9.txt");
                                                                     if ((int.Parse(stroka_v[1]) > 0) \&\& (stroka_z[0])
     string[] stroka = File.ReadAllLines("номер
                                                                == 9.ToString())
принятого задания.txt");
     if (stroka[0] == 9.ToString())
                                                                        pobeda.gameObject.SetActive(true);
       gild.gameObject.SetActive(true);
                                                                        string[] stroka5 =
                                                                File.ReadAllLines("карта.txt");
                                                                        if (stroka5[0] == "1")
  void Update()
                                                                          kart pob.gameObject.SetActive(true);
                                                                        string[] stroka = File.ReadAllLines("HOMEP
     if (FindObjectOfType<Enemy>() != null)
                                                                принятого задания.txt");
                                                                        stroka[0] = 0.ToString();
                                                                        StreamWriter f = new StreamWriter("номер
       text_jz.text =
                                                                принятого задания.txt", false);
(FindObjectOfType<Enemy>().health).ToString();
                                                                        for (int k = 0; k < \text{stroka.Length}; k++)
                                                                          f.WriteLine(stroka[k]);
     string[] vremia = File.ReadAllLines("z8-
9 время.txt");
                                                                        f.Close();
     DateTime d1 = DateTime.Now:
     TimeSpan raznosti = d1 -
                                                                        string[] stroka2 =
DateTime.Parse(vremia[1]);
                                                                File.ReadAllLines("манеты.txt");
     if (raznosti.TotalMinutes > 45)
                                                                        stroka2[0] = (int.Parse(stroka2[0]) +
                                                                600).ToString();
     {
       licor.gameObject.SetActive(true);
       text_licor.gameObject.SetActive(false);
                                                                        StreamWriter f2 = new
                                                                StreamWriter("манеты.txt", false);
     }
     else
                                                                        for (int k = 0; k < \text{stroka2.Length}; k++)
       licor.gameObject.SetActive(false);
                                                                          f2.WriteLine(stroka2[k]);
       text_licor.gameObject.SetActive(true);
       DateTime d2 = DateTime.Now;;
                                                                        f2.Close();
```

```
string[] stroka3 =
File.ReadAllLines("выполнение зд.txt");
                                                                 Файл sbor_predmeta.cs:
       stroka3[8] = 1.ToString();
                                                                 using System;
                                                                 using System.Collections;
       StreamWriter f3 = new
                                                                 using System.Collections.Generic;
StreamWriter("выполнение зд.txt", false);
                                                                 using System.IO;
       for (int k = 0; k < \text{stroka3.Length}; k++)
                                                                 using UnityEditor;
                                                                 using UnityEngine;
          f3.WriteLine(stroka3[k]);
                                                                 public class sbor_predmeta : MonoBehaviour
       f3.Close();
                                                                   private void OnTriggerEnter2D(Collider2D other)
       string[] stroka4 =
File.ReadAllLines("ранг.txt");
                                                                      if (other.CompareTag("Player"))
       stroka4[0] = (int.Parse(stroka4[0]) +
4).ToString();//
                                                                        string[] stroka = File.ReadAllLines("z8-9.txt");
       StreamWriter f4 = new
StreamWriter("ранг.txt", false);
                                                                        string objectName = gameObject.name;
       for (int k = 0; k < \text{stroka4.Length}; k++)
                                                                        if (objectName == "Ликорис") { stroka[0] =
                                                                 (int.Parse(stroka[0])+1).ToString(); }
          f4.WriteLine(stroka4[k]);
                                                                        else if (objectName == "Огненный кристалл")
                                                                 { stroka[1] = (int.Parse(stroka[1]) + 1).ToString(); }
       f4.Close();
       string[] stroka_lik = File.ReadAllLines("z8-
                                                                        StreamWriter f = new StreamWriter("z8-9.txt",
9.txt");
                                                                 false);
       stroka_lik[1] = 0.ToString();
                                                                        for (int k = 0; k < \text{stroka.Length}; k++)
       StreamWriter f lik = new StreamWriter("z8-
9.txt", false);
                                                                           f.WriteLine(stroka[k]);
       for (int k = 0; k < \text{stroka lik.Length}; k++)
                                                                        f.Close();
          f_lik.WriteLine(stroka_lik[k]);
                                                                        string[] stroka1 = File.ReadAllLines("z8-
                                                                9 время.txt");
                                                                        DateTime d = DateTime.Now;
       f_lik.Close();
       StartCoroutine(stop());
                                                                        if (objectName == "Ликорис") { stroka1[0]
                                                                 =d.ToString(); }
                                                                        else if (objectName == "Огненный кристалл")
  private IEnumerator stop()
                                                                 { stroka1[1] = d.ToString(); }
     yield return new WaitForSeconds(3f);
                                                                        StreamWriter f1 = new StreamWriter("z8-
     Time.timeScale = 0f;
                                                                 9 время.txt", false);
                                                                        for (int k = 0; k < \text{stroka1.Length}; k++)
  public void zanovo()
                                                                           f1.WriteLine(stroka1[k]);
     par = false;
                                                                        f1.Close();
     porajenie.SetActive(false);
     FindObjectOfType<Hp>().hp = 2;
                                                                        gameObject.SetActive(false);
     FindObjectOfType<Enemy>().health = 15;
     { a.gameObject.SetActive(true); }
                                                                   }
     { b.gameObject.SetActive(true); }
     Time.timeScale = 1f;
     SceneManager.LoadScene(20);
```

}

Приложение Б

Описание тестов

Таблица Б.1 – Описание тестов	
Заглавие и шаги выполнения	Ожидаемый результат
1	2
Прохождение игрового приложения	1. Игра запускается. Отображается экран
1. Запустить игровое приложение «Путь	загрузки. Загружается главное меню игры с
авантюриста».	кнопками: «Справка», «Играть», «Музыка» и
2. Нажать на кнопку «Справка».	«Выход из игры».
3. Нажать на раздел «Общие сведения».	2. Загружается окно справки С начальной
4. Нажать на подраздел «О программе».	страницей и разделами: «Общие сведения» с
5. Нажать на подраздел «Об авторе».	подразделами «О программе», «Об авторе»,
6. Нажать на раздел «Руководство к	раздел «Руководство к применению» с
применению».	подразделами «Играть» и «Музыка».
7. Нажать на подраздел «Играть».	3. Осуществляется переход на страницу «Общие
8. Нажать на подраздел «Музыка».	сведения» с базовой информацией.
9. Нажать на кнопку «Закрыть».	4. Осуществляется переход на страницу «О
10. Нажать на кнопку «Музыка».	программе» с основной информацией о
11. Нажать на кнопку «Мелодия 2».	приложении.
12. Нажать на кнопку «Стоп».	5. Осуществляется переход на страницу «Об
13. Нажать на кнопку «Стоп».	авторе» с основной информацией о создателях
14. На сцене «Музыка» нажать на кнопку «Выход».	приложения.
«выход». 15. Нажать на кнопку «Играть».	6. Осуществляется переход на страницу «Руководство к применению» с информацией по
13. Пажать на кнопку «играть». 16. Нажать на кнопку «Далее».	кт уководство к применению» с информацией по главному экрану.
10. Пажать на кнопку «Далес». 17. Нажать кнопку «В путь».	7. Осуществляется переход на страницу «Играть»
17. Пажать кнопку «Меню».	с основной информацией о механике игры.
19. Нажать кнопку «Закрыть».	8. Осуществляется переход на страницу
20. Нажать на дверь гильдии	«Музыка» с основной информацией о настройке
«Бесконечный путь»	музыки.
21. Используя для передвижения	9. Закрывается справка.
клавиши «А», «D», «UpArrow»,	10. Загружается сцена «Музыка» с кнопками
«LeftArrow», «RightArrow», «W»,	«Мелодия 1», «Мелодия 2», «Мелодия 3»,
«Space», «S», «DownArrow» подойти к	«Мелодия 4», «Стоп», «Выход».
администратору гильдии.	11. Запускается Lindsey Stirling - Roundtable
22. Нажать кнопку «Диалог»	Rival.mp3.
23. Нажать кнопку «Далее»	12. Прекращается воспроизведение Lindsey
24. Нажать кнопку «Тест»	Stirling - Roundtable Rival.mp3.
25. Используя для передвижения	13. Продолжается воспроизведение Lindsey
клавиши «А», «D», «UpArrow»,	Stirling - Roundtable Rival.mp3.
«LeftArrow», «RightArrow», «W»,	14. Загружается главное меню игры с кнопками:
«Space», «S», «DownArrow» подойти к	«Справка», «Играть», «Музыка» и «Выход из
врагу.	игры».
26. Повторить шаг 25	15. Загружается сцена «Вход в город» с
	кнопками: «Меню» и изображением предисловия
	с кнопкой «Далее».
	16. Изменяется текст предисловия на
	изображении, появляются кнопки «Начать с
	нуля», «В путь».

1

- 27. Нажать кнопку «Гильдия»
- 28. Используя для передвижения клавиши «А», «D», «UpArrow», «LeftArrow», «RightArrow», «W», «Space», «S», «DownArrow» подойти к администратору гильдии.
- 29. Нажать на кнопку «F» на доске заданий
- 30. Нажать на кнопку «Задание 1»
- 31. Нажать кнопку «Принять»
- 32. Используя для передвижения клавиши «А», «D», «UpArrow», «LeftArrow», «RightArrow», «W», «Space», «S», «DownArrow» подойти к кнопкам «Радужный колокольчик» и «Снежный зверобой» и нажать на них
- 33. Нажать на кнопку «Гильдия»
- 34. Нажать на кнопку «Инвентарь»
- 35. Нажать на кнопку предмета
- 36. Нажать на кнопку «выход»
- 37. Повторить шаг 29
- 38. Нажать на кнопку «Задание 3»
- 39. Нажать кнопку «Принять»
- 40. Используя для передвижения клавиши «А», «D», «UpArrow», «LeftArrow», «RightArrow», «W», «Space», «S», «DownArrow» подойти к кнопкам «Золотой вороний глаз» и «Ледяной вороний глаз» и нажать на них
- 41. Повторить шаг 33
- 42. Нажать на кнопку «Выход»
- 43. Используя для передвижения клавиши «А», «D», «UpArrow», «LeftArrow», «RightArrow», «W», «Space», «S», «DownArrow» подойти к торговой лавке.
- 44. Нажать на дверь торговой лавки.
- 45. Используя для передвижения клавиши «А», «D», «UpArrow», «LeftArrow», «RightArrow», «W», «Space», «S», «DownArrow» подойти к владельцу лавки.
- 46. Нажать кнопку «Диалог».
- 47. Нажать кнопку «К покупке».
- 48. Нажать на предмет «Пособие по теории шифрования».
- 49. Нажать на кнопку «Купить».
- 50. Нажать на предмет «Стол».

- 17. Загружается сцена «Город» с кнопками «Инвентарь» и «Меню».
- 18. Игра останавливается, отображается меню с кнопками «Справка», «Музыка», «Выход из игры» и «Закрыть».
- 19. Меню закрывается, игровое время запускается.
- 20. Загружается сцена «Зал гильдии» с кнопами «Меню», «Инвентарь», «Выход» и администратором гильдии.
- 21. Появляется кнопка «Диалог»
- 22. Отображается изображение окна диалога с репликой, и именем персонажа и кнопкой «Далее».
- 23. Изменяется текст на изображении окна диалога, появляется кнопка «Тест».
- 24. Загружается сцена «Арена», с изображением единиц здоровья и врагом, а также кнопкой «Меню».
- 25. Теряется одна единица здоровья, игрок отталкивается от врага.
- 26. Повторяется шаг 25, игра останавливается, отображается изображение с сообщением о принятии в гильдию с рангом F и кнопка «Гильдия».
- 27. Загружается сцена «Зал гильдии» с кнопами «Меню», «Инвентарь», «Выход», администратором гильдии. Отображается текущий ранг, на доске заданий доступен ранг F.
- 28. Кнопки «Диалог» и «Тест» не отображаются.
- 29. Отображаются кнопки «Задание 1», «Задание 2», «Задание 3».
- 30. Отображается изображение с описанием задания и кнопка «Принять».
- 31. Загружается сцена «Лес», с кнопками «Меню», «Инвентарь», «Гильдия», и карточками растений «Радужный колокольчик» и «Снежный зверобой» с указанием добытого количества.
- 32. Значения на карточках задания увеличиваются. Игра останавливается. Отображается изображение сообщения о выполнении награды и получении награды и кнопкой «Гильдия».

1

- 51. Нажать на кнопку «Купить».
- 52. Нажать на предмет «Стол».
- 53. Нажать на кнопку «Руководство начинающего травника».
- 54. Нажать на кнопку «Выход».
- 55. Повторить шаг 34
- 56. Нажать на кнопку «Шифр»
- 57. Нажать на кнопку «Использовать».
- 58. Нажать на кнопку «Выход».
- 59. Нажать на кнопку «Руководство начинающего травника»
- 60. Нажать на кнопку «Использовать».
- 61. Нажать на кнопку растения
- 62. Нажать на кнопку «Выход».
- 63. Нажать на кнопку «Выход».
- 64. Повторить шаги 45-46.
- 65. Нажать кнопку «К продаже».
- 66. Нажать на кнопку добытого ресурса.
- 67. Ввести в поле «Шт.» добытое количество ресурса.
- 68. Нажать кнопку «Продать»
- 69. Ввести в поле «Шт.» значение больше добытого количества ресурса.
- 70. Нажать на кнопку «Выход».
- 71. Нажать на кнопку «Выход».
- 72. Используя для передвижения клавиши «А», «D», «UpArrow», «LeftArrow», «RightArrow», »W», «Space», «S», «DownArrow» подойти к дому.
- 73. Нажать на дверь дома.
- 74. Нажать на кнопку «Плюс»
- 75. Нажать на кнопку «Стол»
- 76. Нажать на кнопку «Плюс»
- 77. Нажать на клавишу «Плюс», наведя курсор на украшение.
- 78. Нажать на клавишу «Минус», наведя курсор на украшение.
- 79. Нажать на клавишу «Таb», наведя курсор на украшение.
- 80. Нажать на клавишу «Q», наведя курсор на украшение.
- 81. Зажав левую клавишу мыши, наведя курсор на украшение потянуть мышь.
- 82. Нажать кнопку «Выход»
- 83. Повторить шаги 72-73.

2

33. Загружается сцена «Зал гильдии» с кнопами «Меню», «Инвентарь», «Выход», администратором гильдии. Отображается текущий ранг, на доске заданий доступен ранг F. 34. Загружается сцена «Инвентарь», отображается количество добытых ресурсов.

Количество монет равно 100, количество остальных предметов 0.

- 35. Отображается описание предмета
- 36. Загружается сцена «Зал гильдии» с кнопами «Меню», «Инвентарь», «Выход», администратором гильдии. Отображается текущий ранг, на доске заданий доступен ранг F.
- 37. Повторяется шаг 29
- 38. Отображается изображение с описанием задания и кнопка «Принять».
- 39. Отображается изображение с описанием задания и кнопка «Принять».
- 40. Значения на карточках задания увеличиваются. Игра останавливается. Отображается изображение сообщения о выполнении награды и получении награды и кнопкой «Гильдия».
- 41. Повторяется шаг 33
- 42. Загружается сцена «Город» с кнопками «Инвентарь» и «Меню». Отображается текущий ранг.
- 43. Игровой персонаж перемещается к торговой лавке.
- 44. Загружается сцена «Торговая лавка» с владельцем лавки, кнопками «Меню» и «Инвентарь», отображается текущий ранг.
- 45. Появляется кнопка «Диалог»
- 46. Отображается изображение окна диалога и кнопки «К покупке», «К продаже».
- 47. Загружается сцена «Торговля», отображается интерфейс покупки из торговой лавки.
- Отображается количество добытых монет и кнопка «Выход».
- 48. Отображается информация о предмете и иго цена
- 49. Количество добытых монет уменьшается на цену предмета. В инвентаре значение данного предмета увеличивается на 1.

84. Нажать на клавишу «Backspace».

- 85. Повторить шаг 82.
- 86. Используя для передвижения клавиши «А», «D», «UpArrow», «LeftArrow», «RightArrow», «W», «Space», «S»,
- «DownArrow» подойти к магазину магии.
- 87. Используя для передвижения клавиши «А», «D», «UpArrow», «LeftArrow»,
- «RightArrow», «W», «Space», «S»,
- «DownArrow» подойти к владельцу магазина.
- 88. Нажать кнопку «Диалог».
- 89. Нажать кнопку «к покупке»
- 90. Нажать на кнопку «Карта»
- 91. Нажать на кнопку «Купить»
- 92. Нажать на кнопку «Котелок»
- 93. Нажать на кнопку «Купить»
- 94. Нажать на кнопку «Руководство по зелье варению»
- 95. Нажать на кнопку «Купить»
- 96. Нажать на кнопку «Выход»
- 97. Нажать на кнопку «Выход»
- 98. Нажать на дверь гильдии
- 99. Нажать на кнопку «Инвентарь»
- 100. Нажать на кнопку «Руководство по зелье варению»
- 101. Нажать на кнопку «Использовать»
- 102. Нажать на кнопку зелья
- 103. Нажать на кнопку «Выход»
- 104. Нажать на кнопку «Котелок»
- 105. Нажать на кнопку «Использовать»
- 106. Нажать на кнопку ингредиента.
- 107. Расположить ингредиенты с помощью мыши слоты котла в соответствии с рецептом.
- 108. Нажать на кнопку «Создать».
- 109. Нажать на кнопку зелья.
- 110. Нажать на кнопку «Выход».
- 111. Повторить шаг 29.
- 112. Нажать кнопку «Задание 2».
- 113. Нажать кнопку «Принять».
- 114. Ввести в поле ввода ответа значение «аааа».
- 115. Нажать на кнопку «Готово»
- 116. Нажать на кнопку «Заново»

2

- 50. Отображается информация о предмете и иго цена
- 51. Количество добытых монет уменьшается на цену предмета. В инвентаре значение данного предмета увеличивается на 1.
- 52. Отображается информация о предмете и иго цена
- 53. Количество добытых монет уменьшается на цену предмета. В инвентаре значение данного предмета увеличивается на 1.
- 54. Загружается сцена «Торговая лавка» с владельцем лавки, кнопками «Меню» и «Инвентарь», отображается текущий ранг.
- 55. Загружается сцена «Инвентарь», отображается количество добытых ресурсов.
- 56. Отображается информация о предмете и кнопка «Использовать».
- 57. Загружается сцена книга. Отображается изображение с информацией о методе шифрования и кнопка «Выход».
- 58. Загружается сцена «Инвентарь», отображается количество добытых ресурсов.
- 59. Отображается информация о предмете и кнопка «Использовать».
- 60. Загружается сцена книга. Отображается книга с кнопками растений и кнопка «Выход».
- 61. Отображается информация о растении.
- 62. Загружается сцена «Инвентарь», отображается количество добытых ресурсов.
- 63. Загружается сцена «Торговля», отображается интерфейс продажи из торговой лавки.
- Отображается количество добытых монет и количество добытых ресурсов и кнопка «Выход».
- 64. Повторяются шаги 45-46.
- 65. Загружается сцена «Торговля», отображается интерфейс продажи из торговой лавки.
- Отображается количество добытых монет и количество добытых ресурсов и кнопка «Выход».
- 66. Отображается информация о ресурсе и его цена за единицу.
- 67. Введённое значение отображается в поле «Шт.»

117. Ввести в поле ввода ответа 1 значение «Великие начинания даже не надо обдумывать, надо взяться да дело, иначе, заметив трудность отступишь.».

- 118. Ввести в поле ввода ответа 1 значение «-Гай Юлий цезарь. «.
- 119. Нажать на кнопку «Готово».
- 120. Нажать на кнопку «Гильдия».
- 121. Нажать на кнопку «Меню».
- 122. Нажать на кнопку «Карта».
- 123. Нажать на кнопку «Город».
- 124. Нажать на дверь гильдии.
- 125. Нажать на кнопку ранга Е на доске заданий.
- 126. Нажать кнопку «Задание 4»
- 127. Нажать кнопку «Принять».
- 128. Повторить шаги 106-109 для требуемого зелья.
- 129. Нажать на кнопку «гильдия».
- 130. Повторить шаг 125
- 131. Нажать кнопку «Задание 5»
- 132. Повторить шаги 127-129
- 133. Повторить шаг 125
- 134. Нажать кнопку «Задание 6»
- 135. Используя для передвижения клавиши «А», «D», «UpArrow», «LeftArrow», «RightArrow», «W», «Space», «S», «DownArrow» избегая врагов коснуться кристалла.
- 136. Собрать нужное число кристаллов.
- 137. Нажать кнопку «Гильдия»
- 138. Нажать на кнопку ранга D на доске заланий.
- 139. Нажать кнопку «Задание 8»
- 140. Нажать кнопку «Принять».
- 141. Коснуться летящего снаряда
- 142. Коснуться летящего снаряда
- 143. Нажать на копку «Заново»
- 144. Используя для передвижения клавиши
- «A», «D», «UpArrow», «LeftArrow»,
- «RightArrow», «W», «Space», «S»,
- «DownArrow» избегая летящих снаряда приблизиться к врагу.
- 145. Нажать на левую клавишу мыши.
- 146. Повторить шаги 144-145 девять

2

- 68. Количество ресурса уменьшается на введённое число. Количество добытых монет увеличивается на произведение цены ресурса и введённого количества.
- 69. Ничего не изменяется
- 70. Загружается сцена «Торговая лавка» с владельцем лавки, кнопками «Меню» и «Инвентарь», отображается текущий ранг.
- 71. Загружается сцена «Город» с кнопками «Инвентарь» и «Меню». Отображается текущий ранг.
- 72. Игровой персонаж перемещается к дому.
- 73. Загружается сцена «Дом» с кнопками «Инвентарь», «Плюс», «Выход» и «Меню».

Отображается текущий ранг.

- 74. Отображается список украшений с указанием имеющегося количества.
- 75. Украшение «Стол» появляется на сцене.
- 76. Список украшений с указанием имеющегося количества скрывается.
- 77. Украшение увеличивается.
- 78. Украшение уменьшается.
- 79. Украшение поворачивается в противоположное направление.
- 80. Украшение поворачивается по окружности.
- 81. Украшение перемещается.
- 82. Загружается сцена «Город» с кнопками «Инвентарь» и «Меню». Отображается текущий ранг.
- 83. Шаги 72-73 повторяются. Украшение отображается в трансформированном виде.
- 84. Украшение удаляется со сцены. В инвентаре значение данного украшения увеличивается на 1
- 85. Шаг 82 повторяется.
- 86. Загружается сцена «Магазин магии» с владельцем магазина, кнопками «Меню» и «Инвентарь», отображается текущий ранг.
- 87. Появляется кнопка «Диалог».
- 88. Отображается изображение окна диалога и кнопки «К покупке».

147. Используя для передвижения клавиши «А», «D», «UpArrow», «LeftArrow», «RightArrow», «W», «Space», «S», «DownArrow» коснуться игровым персонажем изображения растения «Паучья лилия»

- 148. Нажать кнопку «Гильдия»
- 149. Нажать на кнопку ранга D на доске заданий.
- 150. Нажать кнопку «Задание 7»
- 151. Нажать кнопку «Принять».
- 152. Используя для передвижения клавиши «А», «D», «UpArrow», «LeftArrow», «RightArrow», «W», «Space», «S», «DownArrow» переместиться к платформе с персонажем «Белка».
- 153. Используя для передвижения клавиши «А», «D», «UpArrow», «LeftArrow», «RightArrow», «W», «Space», «S», «DownArrow» коснуться игровым персонажем персонажа «Белка» 3 раза.
- 154. Повторить шаги 148-149
- 155. Нажать кнопку «Задание 9»
- 156. Нажать кнопку «Принять».
- 157. Повторить шаги 141-145
- 158. Повторить шаги 144-145 четырнадцать раз.
- 159. Используя для передвижения клавиши «А», «D», «UpArrow», «LeftArrow», «RightArrow», «W», «Space», «S», «DownArrow» коснуться игровым персонажем изображения предмета «Огненный кристалл»
- 160. Нажать кнопку «Гильдия»
- 161. Нажать кнопку «Продолжить»
- 162. Нажать кнопку «Выйти»
- 163. Нажать кнопку «Гильдия»
- 164. Нажать кнопку «Начать с нуля»
- 165. Повторить шаги 20-25
- 166. Нажать левую клавишу мыши.

2

- 89. Загружается сцена «Торговля», отображается интерфейс покупки из магазина магии. Отображается количество добытых монет и количество добытых ресурсов и кнопка «Выход».
- 90. Отображается информация о предмете и иго цена
- 91. Количество добытых монет уменьшается на цену предмета. В инвентаре значение данного предмета увеличивается на 1.
- 92. Отображается информация о предмете и иго пена
- 93. Количество добытых монет уменьшается на цену предмета. В инвентаре значение данного предмета увеличивается на 1.
- 94. Отображается информация о предмете, цена 95. Количество добытых монет уменьшается на цену предмета. В инвентаре значение данного предмета увеличивается на 1.
- 96. Загружается сцена «Магазин магии» с владельцем магазина, кнопками «Меню» и «Инвентарь», отображается текущий ранг.
- 97. Загружается сцена «Город» с кнопками «Инвентарь» и «Меню». Отображается текущий ранг.
- 98. Загружается сцена «Зал гильдии» с кнопами «Меню», «Инвентарь», «Выход», администратором гильдии. Отображается
- текущий ранг, на доске заданий доступен ранг F.
- 99. Загружается сцена «Инвентарь»,
- отображается количество добытых ресурсов.
- 100. Отображается информация о предмете
- 101. Загружается сцена книга. Отображается книга с кнопками зелий и кнопка «Выход».
- 102. Отображается информация о зелье.
- 103. Загружается сцена «Инвентарь», отображается количество добытых ресурсов.
- 104. Отображается информация о предмете
- 105. Загружается сцена «Котелок», отображается количество добытых ингредиентов.
- 106. Ингредиент появляется на сцене, отображается количества добытого ингредиента отнимается 1.
- 107. Слоты принимают ингредиенты.

Продолжение таблицы Б.1	
	2
	108. Отображается кнопка зелья, ингредиенты
	исчезают.
	109. В инвентаре количество зелья
	увеличивается на 1, кнопка зелья исчезает.
	110. Загружается сцена «Зал гильдии» с
	кнопами «Меню», «Инвентарь», «Выход»,
	администратором гильдии. Отображается
	текущий ранг, на доске заданий доступен ранг F.
	111. Повторяется шаг 29.
	112. Отображается изображение с описанием
	задания и кнопкой «Принять».
	113. Загружается сцена «Шифр» с кнопами
	«Меню», «Гильдия», полями ввода ответа и
	кнопкой «Готово»
	114. В поле ввода ответа отображается значение
	«aaaa».
	115. Игра останавливается. Отображается
	изображение сообщения «Поражение» с
	кнопками «Заново» и «Гильдия».
	116. Игра запускается. Поле для ввода ответа
	отображается пустым.
	117. В поле ввода ответа1 отображается
	значение «Великие начинания даже не надо
	обдумывать, надо взяться да дело, иначе,
	заметив трудность отступишь.».
	118. В поле ввода ответа 2 отображается
	значение «- Гай Юлий цезарь. «.
	119. Игра останавливается. Отображается
	изображение сообщения о выполненном
	задании с указанием зачисленной награды и
	кнопкой «Гильдия».
	120. Загружается сцена «Зал гильдии» с
	кнопами «Меню», «Инвентарь», «Выход»,
	администратором гильдии. Отображается
	текущий ранг, на доске заданий доступен ранг
	E.
	121. Игра останавливается. Отображается
	изображение меню с кнопками «Справка»,
	«Музыка», «Выход из игры», «Закрыть» и
	«Карта».
	122. Загружается сцена «Карта» с кнопками
	«Город» и «Лес».
	123. Загружается сцена «Город» с кнопками
	«Инвентарь» и «Меню». Отображается текущий
	ранг.

Продолжение таолицы Б.1	2
· ·	124. Загружается сцена «Зал гильдии» с
	кнопами «Меню», «Инвентарь», «Выход»,
	администратором гильдии. Отображается
	текущий ранг, на доске заданий доступен ранг
	E.
	125. Отображаются кнопки заданий ранга «Е»
	126. Отображается изображение с описанием
	задания и кнопкой «Принять».
	127. Загружается сцена «Котелок», с кнопками
	ингредиентов и указанием их добытого
	количества, и карточка задания с указание
	количества созданных зелий требуемых в
	задании и кнопкой «Гильдия».
	128. Повторяются шаги 106-109 для требуемого
	зелья. Игра останавливается. Отображается
	сообщение о выполнении задания и начислении
	наград с Кнопками «Карта» и «Гильдия».
	129. Загружается сцена «Зал гильдии» с
	кнопами «Меню», «Инвентарь», «Выход»,
	администратором гильдии. Отображается
	текущий ранг, на доске заданий доступен ранг
	E.
	130. Повторяется шаг 125.
	131. Отображается изображение с описанием
	задания и кнопкой «Принять».
	132. Повторяются шаги 127-129.
	133. Повторяется шаг 125.
	134. Загружается сцена «Руины», Отображается
	полоса жизни, текущий ранг, карточка с
	количеством исцеляющего зелья, кнопка
	«Инвентарь», кнопка «Меню», и карточка
	задания с указанием количества собранных
	кристаллов.
	135. Количество кристаллов увеличивается на
	карточке задания, кристалл исчезает и
	появляется в новом месте.
	136. Отображается изображение с сообщением о
	выполнении задания и начислении наград и
	кнопками «Гильдия» и «Карта».
	137. Загружается сцена «Зал гильдии» с
	кнопами «Меню», «Инвентарь», «Выход»,
	администратором гильдии. Отображается
	текущий ранг, на доске заданий доступен ранг
	D.
	138. Отображаются кнопки заданий ранга «D»

Продолжение таблицы Б.1	
1	2
	139. Отображается изображение с описанием
	задания и кнопкой «Принять».
	140. Загружается сцена «Паучья гора»,
	Отображается полоса жизни персонажа,
	текущий ранг, карточка с количеством
	исцеляющего зелья, кнопка «Инвентарь»,
	кнопка «Меню», количество жизней врага, враг
	и растение «Паучья лилия».
	141. Скрывается 1 полоса здоровья игрока.
	142. Скрывается 1 полоса здоровья игрока.
	Отображается изображение с сообщения о
	поражении, отображаются кнопки «Заново»,
	«Гильдия», «Карта».
	143. Скрывается изображение с сообщением о
	поражении и кнопки «Заново», «Гильдия»,
	«Карта». Восстанавливаются единицы здоровья
	игрока и врага.
	144. Игровой персонаж приближается к врагу.
	145. Происходит выстрел, появляется огненный
	снаряд и летит в направлении курсора, попадает
	во врага, количество единиц здоровья врага
	уменьшается на единицу
	146. Шаги 144-145 повторяются 9 раз,
	количество единиц здоровья врага становится
	равным 0, враг исчезает, отображение единиц
	здоровья врага исчезает.
	147. Изображение растения «Паучья лилия»
	скрывается, отображается текстовая строка
	таймера отображающего оставшееся время до
	следующего появления растения. Отображается
	изображение с сообщением о выполнении
	задания и начислении наград и кнопками
	«Гильдия» и «Карта».
	148. Загружается сцена «Зал гильдии» с
	кнопами «Меню», «Инвентарь», «Выход»,
	администратором гильдии. Отображается
	текущий ранг, на доске заданий доступен ранг
	D.
	149. Отображаются кнопки заданий «задание 7»
	и «задание 9» ранга «D»
	150. Отображается изображение с описанием
	задания и кнопкой «Принять».

Продолжение таблицы Б.1	
1	2
	151. Загружается сцена «Беличья поляна»,
	Отображается текущий ранг, кнопка
	«Инвентарь», кнопка «Меню», персонаж
	«Белка» движется в правую сторону локации,
	добегает до платформы, перемещается вверх и
	останавливается.
	152. Персонаж «Белка» движется в левую
	сторону локации, добегает до платформы,
	перемещается вверх и останавливается.
	153. Отображается изображение с сообщением о
	выполнении задания и начислении наград и
	кнопками «Гильдия» и «Карта».
	154. Повторяются шаги 148-149
	155. Отображается изображение с описанием
	задания и кнопкой «Принять».
	156. Загружается сцена «Долина вулканов»,
	Отображается полоса жизни персонажа,
	текущий ранг, карточка с количеством
	исцеляющего зелья, кнопка «Инвентарь»,
	кнопка «Меню», количество жизней врага, враг
	и предмет «Огненный кристалл».
	157. Повторяются шаги 141-145.
	158. Повторяются шаги 144-145 четырнадцать
	раз.
	159. Изображение предмета «Огненный
	кристалл» скрывается, отображается текстовая
	строка таймера отображающего оставшееся
	время до следующего появления предмета.
	Отображается изображение с сообщением о
	выполнении задания и начислении наград и
	кнопками «Гильдия» и «Карта».
	160. Загружается сцена «Зал гильдии» с
	кнопами «Меню», «Инвентарь», «Выход»,
	администратором гильдии. Отображается
	текущий ранг, на доске заданий доступен ранг
	С. Игра останавливается. Отображается
	изображение сообщением о том, что игра
	пройдена и кнопками «Начать с нуля» и
	«Продолжить».
	161. Игра запускается.
	162. Загружается сцена «Город» с кнопками
	«Инвентарь» и «Меню». Отображается текущий
	ранг.

1	2
	163. Загружается сцена «Зал гильдии» с
	кнопами «Меню», «Инвентарь», «Выход»,
	администратором гильдии. Отображается
	текущий ранг, на доске заданий доступен ранг
	Е. Игра останавливается. Отображается
	изображение сообщением о том, что игра
	пройдена и кнопками «Начать с нуля» и
	«Продолжить».
	164. Игровой прогресс сброшен.
	165. Повторяются шаги 20-25.
	166. Происходит выстрел, враг теряет единицу
	жизни.

Приложение В

Тестовый сценарий

Таблица В.1 – Тестирование

Заглавие и шаги выполнения	Ожидаемый результат	Фактический результат	Статус
1	2	3	4
Прохождение игрового приложения 1. Запустить игровое приложение «Путь авантюриста». 2. Нажать на кнопку «Справка». 3. Нажать на раздел «Общие сведения». 4. Нажать на подраздел «О программе». 5. Нажать на подраздел «Об авторе». 6. Нажать на раздел «Руководство к применению». 7. Нажать на подраздел «Играть». 8. Нажать на подраздел «Музыка». 9. Нажать на кнопку «Закрыть». 10. Нажать на кнопку «Музыка». 11. Нажать на кнопку «Мелодия 2». 12. Нажать на кнопку «Стоп». 13. Нажать на кнопку «Стоп». 14. На сцене «Музыка» нажать на кнопку «Выход». 15. Нажать на кнопку «Играть». 16. Нажать на кнопку «Далее». 17. Нажать кнопку «В путь». 18. Нажать кнопку «Меню». 19. Нажать кнопку «Закрыть». 20. Нажать на дверь гильдии «Бесконечный путь»	1. Игра запускается. Отображается экран загрузки. Загружается главное меню игры с кнопками: «Справка», «Играть», «Музыка» и «Выход из игры». 2. Загружается окно справки С начальной страницей и разделами: «Общие сведения» с подразделами «О программе», «Об авторе», раздел «Руководство к применению» с подразделами «Играть» и «Музыка». 3. Осуществляется переход на страницу «Общие сведения» с базовой информацией. 4. Осуществляется переход на страницу «О программе» с основной информацией о приложении. 5. Осуществляется переход на страницу «Об авторе» с основной информацией о создателях приложения. 6. Осуществляется переход на страницу «Руководство к применению» с информацией по главному экрану. 7. Осуществляется переход на страницу «Играть» с основной информацией о механике игры. 8. Осуществляется переход на страницу «Музыка» с основной информацией о настройке музыки. 9. Закрывается справка.	1. Игра запустилась. Отобразился экран загрузки. Загрузилось главное меню игры с кнопками: «Справка», «Играть», «Музыка» и «Выход из игры». 2. Загрузилось окно справки с начальной страницей и разделами: «Общие сведения» с подразделами «О программе», «Об авторе», раздел «Руководство к применению» с подразделами «Играть» и «Музыка». 3. Осуществился переход на страницу «Общие сведения» с базовой информацией. 4. Осуществился переход на страницу «О программе» с основной информацией о приложении. 5. Осуществился переход на страницу «Об авторе» с основной информацией о создателях приложения. 6. Осуществился переход на страницу «Руководство к применению» с информацией по главному экрану. 7. Осуществился переход на страницу «Играть» с основной информацией о механике игры. 8. Осуществился переход на страницу «Музыка» с основной информацией о настройке музыки. 9. Закрылась справка.	Пройденс

1	2	3	4
21. Используя для передвижения	10. Загружается сцена «Музыка» с	10. Загрузилась сцена «Музыка» с	
клавиши «А», «D», «UpArrow»,	кнопками «Мелодия 1», «Мелодия 2»,	кнопками «Мелодия 1», «Мелодия 2»,	
«LeftArrow», «RightArrow», «W»,	«Мелодия 3», «Мелодия 4», «Стоп»,	«Мелодия 3», «Мелодия 4», «Стоп»,	
«Space», «S», «DownArrow»	«Выход».	«Выход».	
подойти к администратору	11. Запускается Lindsey Stirling -	11. Запустился Lindsey Stirling -	
гильдии.	Roundtable Rival.mp3.	Roundtable Rival.mp3.	
22. Нажать кнопку «Диалог»	12. Прекращается воспроизведение	12. Прекратилось воспроизведение	
23. Нажать кнопку «Далее»	Lindsey Stirling - Roundtable Rival.mp3.	Lindsey Stirling - Roundtable Rival.mp3.	
24. Нажать кнопку «Тест»	13. Продолжается воспроизведение	13. Продолжилось воспроизведение	
25. Используя для передвижения	Lindsey Stirling - Roundtable Rival.mp3.	Lindsey Stirling - Roundtable Rival.mp3.	
клавиши «А», «D», «UpArrow»,	14. Загружается главное меню игры с	14. Загрузилось главное меню игры с	
«LeftArrow», «RightArrow», «W»,	кнопками: «Справка», «Играть»,	кнопками: «Справка», «Играть»,	
«Space», «S», «DownArrow»	«Музыка» и «Выход из игры».	«Музыка» и «Выход из игры».	
подойти к врагу.	15. Загружается сцена «Вход в город» с	15. Загрузилась сцена «Вход в город» с	
26. Повторить шаг 25	кнопками: «Меню» и изображением	кнопками: «Меню» и изображением	
27. Нажать кнопку «Гильдия»	предисловия с кнопкой «Далее».	предисловия с кнопкой «Далее».	
28. Используя для передвижения	16. Изменяется текст предисловия на	16. Изменился текст предисловия на	
клавиши «А», «D», «UpArrow»,	изображении, появляются кнопки	изображении, появились кнопки «Начать	
«LeftArrow», «RightArrow», «W»,	«Начать с нуля», «В путь».	с нуля», «В путь».	
«Space», «S», «DownArrow»	17. Загружается сцена «Город» с	17. Загрузилась сцена «Город» с	
подойти к администратору	кнопками «Инвентарь» и «Меню».	кнопками «Инвентарь» и «Меню».	
гильдии.	18. Игра останавливается, отображается	18. Игра остановилась, отобразилось	
29. Нажать на кнопку «F» на доске	меню с кнопками «Справка», «Музыка»,	меню с кнопками «Справка», «Музыка»,	
заданий	«Выход из игры» и «Закрыть».	«Выход из игры» и «Закрыть».	
30. Нажать на кнопку «Задание 1»	19. Меню закрывается, игровое время	19. Меню закрылось, игровое время	
31. Нажать кнопку «Принять»	запускается.	запустилось.	
	20. Загружается сцена «Зал гильдии» с	20. Загрузилась сцена «Зал гильдии» с	
	кнопами «Меню», «Инвентарь», «Выход»	кнопами «Меню», «Инвентарь», «Выход»	
	и администратором гильдии	и администратором гильдии	
	21. Появляется кнопка «Диалог».	21. Появилась кнопка «Диалог».	

1	2	3	4
32. Используя для передвижения	22. Отображается изображение окна	22. Отобразилось изображение окна	
клавиши «А», «D», «UpArrow»,	диалога с репликой, и именем персонажа	диалога с репликой, и именем персонажа	
«LeftArrow», «RightArrow», «W»,	и кнопкой «Далее».	и кнопкой «Далее».	
«Space», «S», «DownArrow»	23. Изменяется текст на изображении	23. Изменился текст на изображении окна	
подойти к кнопкам «Радужный	окна диалога, появляется кнопка «Тест».	диалога, появилась кнопка «Тест».	
колокольчик» и «Снежный	24. Загружается сцена «Арена», с	24. Загрузилась сцена «Арена», с	
зверобой» и нажать на них	изображением единиц здоровья и врагом,	изображением единиц здоровья и врагом,	
33. Нажать на кнопку «Гильдия»	а также кнопкой «Меню».	а также кнопкой «Меню».	
34. Нажать на кнопку «Инвентарь»	25. Теряется одна единица здоровья,	25. Потерялась одна единица здоровья,	
35. Нажать на кнопку предмета	игрок отталкивается от врага.	игрок оттолкнулся от врага.	
36. Нажать на кнопку «выход»	26. Повторяется шаг 25, игра	26. Повторился шаг 25, игра	
37. Повторить шаг 29	останавливается, отображается	остановилась, отобразилось изображение	
38. Нажать на кнопку «Задание 3»	изображение с сообщением о принятии в	с сообщением о принятии в гильдию с	
39. Нажать кнопку «Принять»	гильдию с рангом F и кнопка «Гильдия».	рангом F и кнопка «Гильдия».	
40. Используя для передвижения	27. Загружается сцена «Зал гильдии» с	27. Загрузилась сцена «Зал гильдии» с	
клавиши «А», «D», «UpArrow»,	кнопами «Меню», «Инвентарь»,	кнопами «Меню», «Инвентарь»,	
«LeftArrow», «RightArrow», «W»,	«Выход», администратором гильдии.	«Выход», администратором гильдии.	
«Space», «S», «DownArrow»	Отображается текущий ранг, на доске	Отобразился текущий ранг, на доске	
подойти к кнопкам «Золотой	заданий доступен ранг F.	заданий доступен ранг F.	
вороний глаз» и «Ледяной вороний	28. Кнопки «Диалог» и «Тест» не	28. Кнопки «Диалог» и «Тест» не	
глаз» и нажать на них	отображаются.	отобразились.	
41. Повторить шаг 33	29. Отображаются кнопки «Задание 1»,	29. Отобразились кнопки «Задание 1»,	
42. Нажать на кнопку «Выход»	«Задание 2», «Задание 3».	«Задание 2», «Задание 3».	
43. Используя для передвижения	30. Отображается изображение с	30. Отобразилось изображение с	
клавиши «А», «D», «UpArrow»,	описанием задания и кнопка «Принять».	описанием задания и кнопка «Принять».	
«LeftArrow», «RightArrow», «W»,	31. Загружается сцена «Лес», с кнопками	31. Загрузилась сцена «Лес», с кнопками	
«Space», «S», «DownArrow»	«Меню», «Инвентарь», «Гильдия», и	«Меню», «Инвентарь», «Гильдия», и	
подойти к торговой лавке.	карточками растений «Радужный	карточками растений «Радужный	
44. Нажать на дверь торговой	колокольчик» и «Снежный зверобой» с	колокольчик» и «Снежный зверобой» с	
лавки.	указанием добытого количества.	указанием добытого количества.	

1	2	3	4
45. Используя для передвижения	32. Значения на карточках задания	32. Значения на карточках задания	
клавиши «А», «D», «UpArrow»,	увеличиваются. Игра останавливается.	увеличились. Игра остановилась.	
«LeftArrow», «RightArrow», «W»,	Отображается изображение сообщения о	Отобразилось изображение сообщения о	
«Space», «S», «DownArrow»	выполнении награды и получении	выполнении награды и получении	
подойти к владельцу лавки.	награды и кнопкой «Гильдия».	награды и кнопкой «Гильдия».	
46. Нажать кнопку «Диалог».	33. Загружается сцена «Зал гильдии» с	33. Загрузилась сцена «Зал гильдии» с	
47. Нажать кнопку «К покупке».	кнопами «Меню», «Инвентарь»,	кнопами «Меню», «Инвентарь»,	
48. Нажать на предмет «Пособие	«Выход», администратором гильдии.	«Выход», администратором гильдии.	
по теории шифрования».	Отображается текущий ранг, на доске	Отобразился текущий ранг, на доске	
49. Нажать на кнопку «Купить».	заданий доступен ранг F.	заданий доступен ранг F.	
50. Нажать на предмет «Стол».	34. Загружается сцена «Инвентарь»,	34. Загрузилась сцена «Инвентарь»,	
51. Нажать на кнопку «Купить».	отображается количество добытых	отобразилось количество добытых	
52. Нажать на предмет «Стол».	ресурсов. Количество сонет равно 100,	ресурсов. Количество монет равно 100,	
53. Нажать на кнопку	количество остальных предметов 0.	количество остальных предметов 0.	
«Руководство начинающего	35. Отображается описание предмета	35. Отобразилось описание предмета	
травника».	36. Загружается сцена «Зал гильдии» с	36. Загрузилась сцена «Зал гильдии» с	
54. Нажать на кнопку «Выход».	кнопами «Меню», «Инвентарь»,	кнопами «Меню», «Инвентарь»,	
55. Повторить шаг 34	«Выход», администратором гильдии.	«Выход», администратором гильдии.	
56. Нажать на кнопку «Шифр»	Отображается текущий ранг, на доске	Отобразился текущий ранг, на доске	
57. Нажать на кнопку	заданий доступен ранг F.	заданий доступен ранг F.	
«Использовать».	37. Повторяется шаг 29	37. Повторился шаг 29	
58. Нажать на кнопку «Выход».	38. Отображается изображение с	38. Отобразилось изображение с	
59. Нажать на кнопку	описанием задания и кнопка «Принять».	описанием задания и кнопка «Принять».	
«Руководство начинающего	39. Отображается изображение с	39. Отобразилось изображение с	
травника»	описанием задания и кнопка «Принять».	описанием задания и кнопка «Принять».	
60. Нажать на кнопку	40. Значения на карточках задания	40. Значения на карточках задания	
«Использовать».	увеличиваются. Игра останавливается.	увеличились. Игра остановилась.	
61. Нажать на кнопку растения	Отображается изображение сообщения о	Отобразилось изображение сообщения о	
62. Нажать на кнопку «Выход».	выполнении награды и получении	выполнении награды и получении	
63. Нажать на кнопку «Выход».	награды и кнопкой «Гильдия».	награды и кнопкой «Гильдия».	
64. Повторить шаги 45-46.	41. Повторяется шаг 33	41. Повторился шаг 33	
65. Нажать кнопку «К продаже».			

1	2	3	4
66. Нажать на кнопку добытого	42. Загружается сцена «Город» с	42. Загрузилась сцена «Город» с	
pecypca.	кнопками «Инвентарь» и «Меню».	кнопками «Инвентарь» и «Меню».	
67. Ввести в поле «Шт.» добытое	Отображается текущий ранг.	Отобразился текущий ранг.	
количество ресурса.	43. Игровой персонаж перемещается к	43. Игровой персонаж переместился к	
68. Нажать кнопку «Продать»	торговой лавке.	торговой лавке.	
69. Ввести в поле «Шт.» значение	44. Загружается сцена «Торговая лавка» с	44. Загрузилась сцена «Торговая лавка» с	
больше добытого количества	владельцем лавки, кнопками «Меню» и	владельцем лавки, кнопками «Меню» и	
pecypca.	«Инвентарь», отображается текущий	«Инвентарь», отобразился текущий ранг.	
70. Нажать на кнопку «Выход».	ранг.	45. Появилась кнопка «Диалог»	
71. Нажать на кнопку «Выход».	45. Появляется кнопка «Диалог»	46. Отобразилось изображение окна	
72. Используя для передвижения	46. Отображается изображение окна	диалога и кнопки «К покупке», «К	
клавиши «А», «D», «UpArrow»,	диалога и кнопки «К покупке», «К	продаже».	
«LeftArrow», «RightArrow», »W»,	продаже».	47. Загрузилась сцена «Торговля»,	
«Space», «S», «DownArrow»	47. Загружается сцена «Торговля»,	отобразился интерфейс покупки из	
подойти к дому.	отображается интерфейс покупки из	торговой лавки. Отобразился количество	
73. Нажать на дверь дома.	торговой лавки. Отображается	добытых монет и кнопка «Выход».	
74. Нажать на кнопку «Плюс»	количество добытых монет и кнопка	48. Отобразилась информация о предмете	
75. Нажать на кнопку «Стол»	«Выход».	и иго цена	
76. Нажать на кнопку «Плюс»	48. Отображается информация о	49. Количество добытых монет	
77. Нажать на клавишу «Плюс»,	предмете и иго цена	уменьшилось на цену предмета. В	
наведя курсор на украшение.	49. Количество добытых монет	инвентаре значение данного предмета	
78. Нажать на клавишу «Минус»,	уменьшается на цену предмета. В	увеличилось на 1.	
наведя курсор на украшение.	инвентаре значение данного предмета	50. Отобразилась информация о предмете	
79. Нажать на клавишу «Tab»,	увеличивается на 1.	и иго цена	
наведя курсор на украшение.	50. Отображается информация о	51. Количество добытых монет	
80. Нажать на клавишу «Q»,	предмете и иго цена	уменьшилось на цену предмета. В	
наведя курсор на украшение.	51. Количество добытых монет	инвентаре значение данного предмета	
81. Зажав левую клавишу мыши,	уменьшается на цену предмета. В	увеличилось на 1.	
наведя курсор на украшение	инвентаре значение данного предмета	52. Отобразилась информация о предмете	
потянуть мышь.	увеличивается на 1.	и иго цена	
82. Нажать кнопку «Выход»	52. Отображается информация о		
83. Повторить шаги 72-73.	предмете и иго цена		

1	2	3	4
84. Нажать на клавишу	53. Количество добытых монет	53. Количество добытых монет	
«Backspace».	уменьшается на цену предмета. В	уменьшилось на цену предмета. В	
85. Повторить шаг 82.	инвентаре значение данного предмета	инвентаре значение данного предмета	
86. Используя для передвижения	увеличивается на 1.	увеличилось на 1.	
клавиши «А», «D», «UpArrow»,	54. Загружается сцена «Торговая лавка» с	54. Загрузилась сцена «Торговая лавка» с	
«LeftArrow», «RightArrow», «W»,	владельцем лавки, кнопками «Меню» и	владельцем лавки, кнопками «Меню» и	
«Space», «S», «DownArrow»	«Инвентарь», отображается текущий	«Инвентарь», отобразился текущий ранг.	
подойти к магазину магии.	ранг.	55. Загрузилась сцена «Инвентарь»,	
87. Используя для передвижения	55. Загружается сцена «Инвентарь»,	отобразилось количество добытых	
клавиши «А», «D», «UpArrow»,	отображается количество добытых	ресурсов.	
«LeftArrow», «RightArrow», «W»,	ресурсов.	56. Отобразилась информация о предмете	
«Space», «S», «DownArrow»	56. Отображается информация о	и кнопка «Использовать».	
подойти к владельцу магазина.	предмете и кнопка «Использовать».	57. Загрузилась сцена книга.	
88. Нажать кнопку «Диалог».	57. Загружается сцена книга.	Отобразилось изображение с	
89. Нажать кнопку «к покупке»	Отображается изображение с	информацией о методе шифрования и	
90. Нажать на кнопку «Карта»	информацией о методе шифрования и	кнопка «Выход».	
91. Нажать на кнопку «Купить»	кнопка «Выход».	58. Загрузилась сцена «Инвентарь»,	
92. Нажать на кнопку «Котелок»	58. Загружается сцена «Инвентарь»,	отобразилось количество добытых	
93. Нажать на кнопку «Купить»	отображается количество добытых	ресурсов.	
94. Нажать на кнопку	ресурсов.	59. Отобразилась информация о предмете	
«Руководство по зелье варению»	59. Отображается информация о	и кнопка «Использовать».	
95. Нажать на кнопку «Купить»	предмете и кнопка «Использовать».	60. Загрузилась сцена книга.	
96. Нажать на кнопку «Выход»	60. Загружается сцена книга.	Отобразилась книга с кнопками растений	
97. Нажать на кнопку «Выход»	Отображается книга с кнопками растений	и кнопка «Выход».	
98. Нажать на дверь гильдии	и кнопка «Выход».	61. Отобразилась информация о	
99. Нажать на кнопку «Инвентарь»	61. Отображается информация о	растении.	
100. Нажать на кнопку	растении.	62. Загрузилась сцена «Инвентарь»,	
«Руководство по зелье варению»	62. Загружается сцена «Инвентарь»,	отобразилось количество добытых	
101. Нажать на кнопку	отображается количество добытых	ресурсов.	
«Использовать»	ресурсов.		
102. Нажать на кнопку зелья			
103. Нажать на кнопку «Выход»			

1	2	3	4
104. Нажать на кнопку «Котелок»	63. Загружается сцена «Торговля»,	63. Загрузилась сцена «Торговля»,	
105. Нажать на кнопку	отображается интерфейс продажи из	отобразился интерфейс продажи из	
«Использовать»	торговой лавки. Отображается	торговой лавки. Отобразилось	
106. Нажать на кнопку	количество добытых монет и количество	количество добытых монет и количество	
ингредиента.	добытых ресурсов и кнопка «Выход».	добытых ресурсов и кнопка «Выход».	
107. Расположить ингредиенты с	64. Повторяются шаги 45-46.	64. Повторились шаги 45-46.	
помощью мыши слоты котла в	65. Загружается сцена «Торговля»,	65. Загрузилась сцена «Торговля»,	
соответствии с рецептом.	отображается интерфейс продажи из	отобразился интерфейс продажи из	
108. Нажать на кнопку «Создать».	торговой лавки. Отображается	торговой лавки. Отобразилось количество	
109. Нажать на кнопку зелья.	количество добытых монет и количество	добытых монет и количество добытых	
110. Нажать на кнопку «Выход».	добытых ресурсов и кнопка «Выход».	ресурсов и кнопка «Выход».	
111. Повторить шаг 29.	66. Отображается информация о ресурсе	66. Отобразилась информация о ресурсе	
112. Нажать кнопку «Задание 2».	и его цена за единицу.	и его цена за единицу.	
113. Нажать кнопку «Принять».	67. Введённое значение отображается в	67. Введённое значение отобразилось в	
114. Ввести в поле ввода ответа	поле «Шт.»	поле «Шт.»	
значение «аааа».	68. Количество ресурса уменьшается на	68. Количество ресурса уменьшилось на	
115. Нажать на кнопку «Готово»	введённое число. Количество добытых	введённое число. Количество добытых	
116. Нажать на кнопку «Заново»	монет увеличивается на произведение	монет увеличилось на произведение цены	
117. Ввести в поле ввода ответа 1	цены ресурса и введённого количества.	ресурса и введённого количества.	
значение «Великие начинания	69. Ничего не изменяется	69. Ничего не изменилось	
даже не надо обдумывать, надо	70. Загружается сцена «Торговая лавка» с	70. Загрузилась сцена «Торговая лавка» с	
взяться да дело, иначе, заметив	владельцем лавки, кнопками «Меню» и	владельцем лавки, кнопками «Меню» и	
трудность отступишь.».	«Инвентарь», отображается текущий	«Инвентарь», отобразился текущий ранг.	
118. Ввести в поле ввода ответа 1	ранг.	71. Загрузилась сцена «Город» с	
значение «- Гай Юлий цезарь. «.	71. Загружается сцена «Город» с	кнопками «Инвентарь» и «Меню».	
119. Нажать на кнопку «Готово».	кнопками «Инвентарь» и «Меню».	Отобразился текущий ранг.	
120. Нажать на кнопку «Гильдия».	Отображается текущий ранг.	72. Игровой персонаж переместился к	
121. Нажать на кнопку «Меню».	72. Игровой персонаж перемещается к	дому.	
122. Нажать на кнопку «Карта».	дому.	73. Загрузилась сцена «Дом» с кнопками	
123. Нажать на кнопку «Город».	73. Загружается сцена «Дом» с кнопками	«Инвентарь», «Плюс», «Выход» и	
124. Нажать на дверь гильдии	«Инвентарь», «Плюс», «Выход» и	«Меню». Отобразился текущий ранг.	
	«Меню». Отображается текущий ранг.		

1	2	3	4
125. Нажать на кнопку ранга Е на	74. Отображается список украшений с	74. Отобразился список украшений с	
доске заданий.	указанием имеющегося количества.	указанием имеющегося количества.	
126. Нажать кнопку «Задание 4»	75. Украшение «Стол» появляется на	75. Украшение «Стол» появилось на	
127. Нажать кнопку «Принять».	сцене.	сцене.	
128. Повторить шаги 106-109 для	76. Список украшений с указанием	76. Список украшений с указанием	
требуемого зелья.	имеющегося количества скрывается.	имеющегося количества скрылся.	
129. Нажать на кнопку «гильдия».	77. Украшение увеличивается.	77. Украшение увеличилось.	
130. Повторить шаг 125	78. Украшение уменьшается.	78. Украшение уменьшилось.	
131. Нажать кнопку «Задание 5»	79. Украшение поворачивается в	79. Украшение повернулось в	
132. Повторить шаги 127-129	противоположное направление.	противоположное направление.	
133. Повторить шаг 125	80. Украшение поворачивается по	80. Украшение повернулось по	
134. Нажать кнопку «Задание 6»	окружности.	окружности.	
135. Используя для передвижения	81. Украшение перемещается.	81. Украшение переместилось.	
клавиши «А», «D», «UpArrow»,	82. Загружается сцена «Город» с	82. Загрузилась сцена «Город» с	
«LeftArrow», «RightArrow», «W»,	кнопками «Инвентарь» и «Меню».	кнопками «Инвентарь» и «Меню».	
«Space», «S», «DownArrow»	Отображается текущий ранг.	Отобразился текущий ранг.	
избегая врагов коснуться	83. Шаги 72-73 повторяются. Украшение	83. Шаги 72-73 повторились. Украшение	
кристалла.	отображается в трансформированном	отобразилось в трансформированном	
136. Собрать нужное число	виде.	виде.	
кристаллов.	84. Украшение удаляется со сцены. В	84. Украшение удалилось со сцены. В	
137. Нажать кнопку «Гильдия»	инвентаре значение данного украшения	инвентаре значение данного украшения	
138. Нажать на кнопку ранга D на	увеличивается на 1.	увеличилось на 1.	
доске заданий.	85. Шаг 82 повторяется.	85. Шаг 82 повторились.	
139. Нажать кнопку «Задание 8»	86. Загружается сцена «Магазин магии» с	86. Загрузилась сцена «Магазин магии» с	
140. Нажать кнопку «Принять».	владельцем магазина, кнопками «Меню»	владельцем магазина, кнопками «Меню»	
141. Коснуться летящего снаряда	и «Инвентарь», отображается текущий	и «Инвентарь», отобразился текущий	
142. Коснуться летящего снаряда	ранг.	ранг.	
143. Нажать на копку «Заново»	87. Появляется кнопка «Диалог».	87. Появилась кнопка «Диалог».	
	88. Отображается изображение окна	Отобразилось изображение окна диалога	
	диалога и кнопки «К покупке».	и кнопки «К покупке».	

1	2	3	4
144. Используя для передвижения	89. Загружается сцена «Торговля»,	89. Загрузилась сцена «Торговля»,	
клавиши «А», «D», «UpArrow»,	отображается интерфейс покупки из	отобразился интерфейс покупки из	
«LeftArrow», «RightArrow», «W»,	магазина магии. Отображается	магазина магии. Отобразилось	
«Space», «S», «DownArrow»	количество добытых монет и количество	количество добытых монет и количество	
избегая летящих снаряда	добытых ресурсов и кнопка «Выход».	добытых ресурсов и кнопка «Выход».	
приблизиться к врагу.	90. Отображается информация о	90. Отобразилась информация о	
145. Нажать на левую клавишу	предмете и иго цена	предмете и иго цена	
мыши.	91. Количество добытых монет	91. Количество добытых монет	
146. Повторить шаги 144-145	уменьшается на цену предмета. В	уменьшилось на цену предмета. В	
девять раз	инвентаре значение данного предмета	инвентаре значение данного предмета	
147. Используя для передвижения	увеличивается на 1.	увеличилось на 1.	
клавиши «А», «D», «UpArrow»,	92. Отображается информация о	92. Отобразилась информация о	
«LeftArrow», «RightArrow», «W»,	предмете и иго цена	предмете и иго цена	
«Space», «S», «DownArrow»	93. Количество добытых монет	93. Количество добытых монет	
коснуться игровым персонажем	уменьшается на цену предмета. В	уменьшилось на цену предмета. В	
изображения растения «Паучья	инвентаре значение данного предмета	инвентаре значение данного предмета	
лилия»	увеличивается на 1.	увеличилось на 1.	
148. Нажать кнопку «Гильдия»	94. Отображается информация о	94. Отобразилась информация о	
149. Нажать кнопку «Задание 7»	предмете и иго цена	предмете и иго цена	
150. Нажать кнопку «Принять».	95. Количество добытых монет	95. Количество добытых монет	
151. Используя для передвижения	уменьшается на цену предмета. В	уменьшилось на цену предмета. В	
клавиши «А», «D», «UpArrow»,	инвентаре значение данного предмета	инвентаре значение данного предмета	
«LeftArrow», «RightArrow», «W»,	увеличивается на 1.	увеличилось на 1.	
«Space», «S», «DownArrow»	96. Загружается сцена «Магазин магии» с	96. Загрузилась сцена «Магазин магии» с	
переместиться к платформе с	владельцем магазина, кнопками «Меню»	владельцем магазина, кнопками «Меню»	
персонажем «Белка».	и «Инвентарь», отображается текущий	и «Инвентарь», отобразился текущий	
152. Используя для передвижения	ранг.	ранг.	
клавиши «А», «D», «UpArrow»,	97. Загружается сцена «Город» с	97. Загрузилась сцена «Город» с	
«LeftArrow», «RightArrow», «W»,	кнопками «Инвентарь» и «Меню».	кнопками «Инвентарь» и «Меню».	
«Space», «S», «DownArrow»	Отображается текущий ранг.	Отобразился текущий ранг.	
коснуться игровым персонажем			
персонажа «Белка» 3 раза.			

1	2	3	4
153. Используя для передвижения	98. Загружается сцена «Зал гильдии» с	98. Загрузилась сцена «Зал гильдии» с	
клавиши «А», «D», «UpArrow»,	кнопами «Меню», «Инвентарь»,	кнопами «Меню», «Инвентарь»,	
«LeftArrow», «RightArrow», «W»,	«Выход», администратором гильдии.	«Выход», администратором гильдии.	
«Space», «S», «DownArrow»	Отображается текущий ранг, на доске	Отобразился текущий ранг, на доске	
коснуться игровым персонажем	заданий доступен ранг F.	заданий доступен ранг F.	
персонажа «Белка» 3 раза.	99. Загружается сцена «Инвентарь»,	99. Загрузилась сцена «Инвентарь»,	
154. Повторить шаги 148-149	отображается количество добытых	отобразилось количество добытых	
155. Нажать кнопку «Задание 9»	ресурсов.	ресурсов.	
156. Нажать кнопку «Принять».	100. Отображается информация о	100. Отобразилась информация о	
157. Повторить шаги 141-145	предмете	предмете	
158. Повторить шаги 144-145	101. Загружается сцена книга.	101. Загрузилась сцена книга.	
четырнадцать раз.	Отображается книга с кнопками зелий и	Отобразилась книга с кнопками зелий и	
159. Используя для передвижения	кнопка «Выход».	кнопка «Выход».	
клавиши «А», «D», «UpArrow»,	102. Отображается информация о зелье.	102. Отобразилась информация о зелье.	
«LeftArrow», «RightArrow», «W»,	103. Загружается сцена «Инвентарь»,	103. Загрузилась сцена «Инвентарь»,	
«Space», «S», «DownArrow»	отображается количество добытых	отобразилось количество добытых	
коснуться игровым персонажем	ресурсов.	ресурсов.	
изображения предмета «Огненный	104. Отображается информация о	104. Отобразилась информация о	
кристалл»	предмете	предмете	
160. Нажать кнопку «Гильдия»	105. Загружается сцена «Котелок»,	105. Загрузилась сцена «Котелок»,	
161. Нажать кнопку «Продолжить»	отображается количество добытых	отобразилось количество добытых	
162. Нажать кнопку «Выйти»	ингредиентов.	ингредиентов.	
163. Нажать кнопку «Гильдия»	106. Ингредиент появляется на сцене,	106. Ингредиент появился на сцене,	
164. Нажать кнопку «Начать с	отображается количества добытого	отобразилось количества добытого	
нуля»	ингредиента отнимается 1.	ингредиента отнимается 1.	
165. Повторить шаги 20-25	107. Слоты принимают ингредиенты.	107. Слоты приняли ингредиенты.	
166. Нажать левую клавишу	108. Отображается кнопка зелья,	108. Отобразилась кнопка зелья,	
мыши.	ингредиенты исчезают.	ингредиенты исчезли.	
	109. В инвентаре количество зелья	109. В инвентаре количество зелья	
	увеличивается на 1, кнопка зелья	увеличилось на 1, кнопка зелья исчезла.	
	исчезает.		

1	2	3	4
	110. Загружается сцена «Зал гильдии» с	110. Загрузилась сцена «Зал гильдии» с	
	кнопами «Меню», «Инвентарь»,	кнопами «Меню», «Инвентарь»,	
	«Выход», администратором гильдии.	«Выход», администратором гильдии.	
	Отображается текущий ранг, на доске	Отобразился текущий ранг, на доске	
	заданий доступен ранг F.	заданий доступен ранг F.	
	111. Повторяется шаг 29.	111. Повторился шаг 29.	
	112. Отображается изображение с	112. Отобразилось изображение с	
	описанием задания и кнопкой	описанием задания и кнопкой	
	«Принять».	«Принять».	
	113. Загружается сцена «Шифр» с	113. Загрузилась сцена «Шифр» с	
	кнопами «Меню», «Гильдия», полями	кнопами «Меню», «Гильдия», полями	
	ввода ответа и кнопкой «Готово»	ввода ответа и кнопкой «Готово»	
	114. В поле ввода ответа отображается	114. В поле ввода ответа отобразилось	
	значение «аааа».	значение «аааа».	
	115. Игра останавливается.	115. Игра остановилась. Отобразилось	
	Отображается изображение сообщения	изображение сообщения «Поражение» с	
	«Поражение» с кнопками «Заново» и	кнопками «Заново» и «Гильдия».	
	«Гильдия».	116. Игра запустилась. Поле для ввода	
	116. Игра запускается. Поле для ввода	ответа отобразилось пустым.	
	ответа отображается пустым.	117. В поле ввода ответа1 отобразилось	
	117. В поле ввода ответа1 отображается	значение «Великие начинания даже не	
	значение «Великие начинания даже не	надо обдумывать, надо взяться да дело,	
	надо обдумывать, надо взяться да дело,	иначе, заметив трудность отступишь.».	
	иначе, заметив трудность отступишь.».	118. В поле ввода ответа 2 отобразилось	
	118. В поле ввода ответа 2 отображается	значение «- Гай Юлий цезарь. «.	
	значение «- Гай Юлий цезарь. «.	Игра остановилась. Отобразилось	
	119. Игра останавливается.	изображение сообщения о выполненном	
	Отображается изображение сообщения о	задании с указанием зачисленной	
	выполненном задании с указанием	награды и кнопкой «Гильдия».	
	зачисленной награды и кнопкой		
	«Гильдия».		

Продолжение таблицы В.1	<u> </u>		4
1	2	3	4
	120. Загружается сцена «Зал гильдии» с	120. Загрузилась сцена «Зал гильдии» с	
	кнопами «Меню», «Инвентарь»,	кнопами «Меню», «Инвентарь»,	
	«Выход», администратором гильдии.	«Выход», администратором гильдии.	
	Отображается текущий ранг, на доске	Отобразился текущий ранг, на доске	
	заданий доступен ранг Е.	заданий доступен ранг Е.	
	121. Игра останавливается.	121. Игра остановилась. Отобразилось	
	Отображается изображение меню с	изображение меню с кнопками	
	кнопками «Справка», «Музыка», «Выход	«Справка», «Музыка», «Выход из игры»,	
	из игры», «Закрыть» и «Карта».	«Закрыть» и «Карта».	
	122. Загружается сцена «Карта» с	122. Загрузилась сцена «Карта» с	
	кнопками «Город» и «Лес».	кнопками «Город» и «Лес».	
	123. Загружается сцена «Город» с	123. Загрузилась сцена «Город» с	
	кнопками «Инвентарь» и «Меню».	кнопками «Инвентарь» и «Меню».	
	Отображается текущий ранг.	Отобразился текущий ранг.	
	124. Загружается сцена «Зал гильдии» с	124. Загрузилась сцена «Зал гильдии» с	
	кнопами «Меню», «Инвентарь»,	кнопами «Меню», «Инвентарь»,	
	«Выход», администратором гильдии.	«Выход», администратором гильдии.	
	Отображается текущий ранг, на доске	Отобразился текущий ранг, на доске	
	заданий доступен ранг Е.	заданий доступен ранг Е.	
	125. Отображаются кнопки заданий	125. Отобразились кнопки заданий ранга	
	ранга «E»	«E»	
	126. Отображается изображение с	126. Отобразилось изображение с	
	заданием и кнопкой «Принять».	описанием задания и кнопкой	
	127. Загружается сцена «Котелок», с	«Принять».	
	кнопками ингредиентов и указанием их	127. Загрузилась сцена «Котелок», с	
	добытого количества, и карточка задания	кнопками ингредиентов и указанием их	
	с указание количества созданных зелий	добытого количества, и карточка задания	
	требуемых в задании и кнопкой	с указание количества созданных зелий	
	«Гильдия».		

1	2	3	4
	128. Повторяются шаги 106-109 для	128. Повторились шаги 106-109 для	
	требуемого зелья. Игра останавливается.	требуемого зелья. Игра остановилась.	
	Отображается сообщение о выполнении	Отобразилось сообщение о выполнении	
	задания и начислении наград с Кнопками	задания и начислении наград с Кнопками	
	«Карта» и «Гильдия».	«Карта» и «Гильдия».	
	129. Загружается сцена «Зал гильдии» с	129. Загрузилась сцена «Зал гильдии» с	
	кнопами «Меню», «Инвентарь»,	кнопами «Меню», «Инвентарь»,	
	«Выход», администратором гильдии.	«Выход», администратором гильдии.	
	Отображается текущий ранг, на доске	Отобразился текущий ранг, на доске	
	заданий доступен ранг Е.	заданий доступен ранг Е.	
	130. Повторяется шаг 125.	130. Повторились шаг 125.	
	131. Отображается изображение с	131. Отобразилось изображение с	
	описанием задания и кнопкой «Принять».	описанием задания и кнопкой «Принять».	
	132. Повторяются шаги 127-129.	132. Повторились шаги 127-129.	
	133. Повторяется шаг 125.	133. Повторился шаг 125.	
	134. Загружается сцена «Руины»,	134. Загрузилась сцена «Руины»,	
	Отображается полоса жизни, текущий	Отобразилась полоса жизни, текущий	
	ранг, карточка с количеством	ранг, карточка с количеством	
	исцеляющего зелья, кнопка «Инвентарь»,	исцеляющего зелья, кнопка «Инвентарь»,	
	кнопка «Меню», и карточка задания с	кнопка «Меню», и карточка задания с	
	указанием количества собранных	указанием количества собранных	
	кристаллов.	кристаллов.	
	135. Количество кристаллов	135. Количество кристаллов увеличилось	
	увеличивается на карточке задания,	на карточке задания, кристалл исчез и	
	кристалл исчезает и появляется в новом	появился в новом месте.	
	месте.	136. Отобразилось изображение с	
	136. Отображается изображение с	сообщением о выполнении задания и	
	сообщением о выполнении задания и	начислении наград и кнопками	
	начислении наград и кнопками	«Гильдия» и «Карта».	
	«Гильдия» и «Карта».	_	

1	2	3	4
	137. Загружается сцена «Зал гильдии» с	137. Загрузилась сцена «Зал гильдии» с	
	кнопами «Меню», «Инвентарь»,	кнопами «Меню», «Инвентарь»,	
	«Выход», администратором гильдии.	«Выход», администратором гильдии.	
	Отображается текущий ранг, на доске	Отобразился текущий ранг, на доске	
	заданий доступен ранг D.	заданий доступен ранг D.	
	138. Отображаются кнопки заданий	138. Отобразились кнопки заданий ранга	
	ранга «D»	«D»	
	139. Отображается изображение с	139. Отобразилось изображение с	
	описанием задания и кнопкой	описанием задания и кнопкой	
	«Принять».	«Принять».	
	140. Загружается сцена «Паучья гора»,	140. Загрузилась сцена «Паучья гора»,	
	Отображается полоса жизни персонажа,	Отобразилась полоса жизни персонажа,	
	текущий ранг, карточка с количеством	текущий ранг, карточка с количеством	
	исцеляющего зелья, кнопка «Инвентарь»,	исцеляющего зелья, кнопка «Инвентарь»,	
	кнопка «Меню», количество жизней	кнопка «Меню», количество жизней	
	врага, враг и растение «Паучья лилия».	врага, враг и растение «Паучья лилия».	
	141. Скрывается 1 полоса здоровья	141. Скрылась 1 полоса здоровья игрока.	
	игрока.	142. Скрывается 1 полоса здоровья	
	142. Скрывается 1 полоса здоровья	игрока. Отобразилось изображение с	
	игрока. Отображается изображение с	сообщения о поражении, отобразились	
	сообщения о поражении, отображаются	кнопки «Заново», «Гильдия», «Карта».	
	кнопки «Заново», «Гильдия», «Карта».	143. Скрылось изображение с	
	143. Скрывается изображение с	сообщением о поражении и кнопки	
	сообщением о поражении и кнопки	«Заново», «Гильдия», «Карта».	
	«Заново», «Гильдия», «Карта».	Восстановились единицы здоровья	
	Восстанавливаются единицы здоровья	игрока и врага.	
	игрока и врага.	144. Игровой персонаж приблизился к	
	144. Игровой персонаж приближается к	врагу.	
	врагу.		

1	2	3	4
	145. Происходит выстрел, появляется	145. Произошёл выстрел, появился	
	огненный снаряд и летит в направлении	огненный снаряд и полетел в	
	курсора, попадает во врага, количество	направлении курсора, попал во врага,	
	единиц здоровья врага уменьшается на	количество единиц здоровья врага	
	единицу	уменьшилось на единицу	
	146. Шаги 144-145 повторяются 9 раз,	146. Шаги 144-145 повторились 9 раз,	
	количество единиц здоровья врага	количество единиц здоровья врага стало	
	становится равным 0, враг исчезает,	равным 0, враг исчез, отображение	
	отображение единиц здоровья врага	единиц здоровья врага исчезло.	
	исчезает.	147. Изображение растения «Паучья	
	147. Изображение растения «Паучья	лилия» скрылось, отобразилась текстовая	
	лилия» скрывается, отображается	строка таймера отображающего	
	текстовая строка таймера	оставшееся время до следующего	
	отображающего оставшееся время до	появления растения. Отобразилось	
	следующего появления растения.	изображение с сообщением о	
	Отображается изображение с	выполнении задания и начислении	
	сообщением о выполнении задания и	наград и кнопками «Гильдия» и «Карта».	
	начислении наград и кнопками	148. Загрузилась сцена «Зал гильдии» с	
	«Гильдия» и «Карта».	кнопами «Меню», «Инвентарь»,	
	148. Загружается сцена «Зал гильдии» с	«Выход», администратором гильдии.	
	кнопами «Меню», «Инвентарь»,	Отобразился текущий ранг, на доске	
	«Выход», администратором гильдии.	заданий доступен ранг D.	
	Отображается текущий ранг, на доске	149. Отобразились кнопки заданий	
	заданий доступен ранг D.	«задание 7» и «задание 9» ранга «D»	
	149. Отображаются кнопки заданий	150. Отобразилось изображение с	
	«задание 7» и «задание 9» ранга «D»	описанием задания и кнопкой	
	150. Отображается изображение с	«Принять».	
	описанием задания и кнопкой «Принять».		

1	2	3	4
	151. Загружается сцена «Беличья	151. Загрузилась сцена «Беличья	
	поляна», Отображается текущий ранг,	поляна», отобразился текущий ранг,	
	кнопка «Инвентарь», кнопка «Меню»,	кнопка «Инвентарь», кнопка «Меню»,	
	персонаж «Белка» движется в правую	персонаж «Белка» двигался в правую	
	сторону локации, добегает до	сторону локации, добежал до	
	платформы, перемещается вверх и	платформы, переместился вверх и	
	останавливается.	остановился.	
	152. Персонаж «Белка» движется в левую	152. Персонаж «Белка» двигался в левую	
	сторону локации, добегает до	сторону локации, добежал до	
	платформы, перемещается вверх и	платформы, переместился вверх и	
	останавливается.	остановился.	
	153. Отображается изображение с	153. Отобразилось изображение с	
	сообщением о выполнении задания и	сообщением о выполнении задания и	
	начислении наград и кнопками	начислении наград и кнопками	
	«Гильдия» и «Карта».	«Гильдия» и «Карта».	
	154. Повторяются шаги 148-149	154. Повторились шаги 148-149	
	155. Отображается изображение с	155. Отобразилось изображение с	
	описанием задания и кнопкой	описанием задания и кнопкой	
	«Принять».	«Принять».	
	156. Загружается сцена «Долина	156. Загрузилась сцена «Долина	
	вулканов», Отображается полоса жизни	вулканов», отобразилась полоса жизни	
	персонажа, текущий ранг, карточка с	персонажа, текущий ранг, карточка с	
	количеством исцеляющего зелья, кнопка	количеством исцеляющего зелья, кнопка	
	«Инвентарь», кнопка «Меню»,	«Инвентарь», кнопка «Меню»,	
	количество жизней врага, враг и предмет	количество жизней врага, враг и предмет	
	«Огненный кристалл».	«Огненный кристалл».	
	157. Повторяются шаги 141-145.	157. Повторились шаги 141-145.	
	158. Повторяются шаги 144-145	158. Повторились шаги 144-145	
	четырнадцать раз.	четырнадцать раз.	

1	2	3	4
	159. Изображение предмета «Огненный	159. Изображение предмета «Огненный	
	кристалл» скрывается, отображается	кристалл» скрылось, отобразилась	
	текстовая строка таймера до следующего	текстовая строка таймера до следующего	
	появления предмета. Отображается	появления предмета. Отобразилось	
	изображение с сообщением о	изображение с сообщением о	
	выполнении задания и начислении	выполнении задания и начислении наград	
	наград и кнопками «Гильдия» и «Карта».	и кнопками «Гильдия» и «Карта».	
	160. Загружается сцена «Зал гильдии» с	160. Загрузилась сцена «Зал гильдии» с	
	кнопами «Меню», «Инвентарь»,	кнопами «Меню», «Инвентарь»,	
	«Выход», администратором гильдии.	«Выход», администратором гильдии.	
	Отображается текущий ранг, на доске	Отобразился текущий ранг, на доске	
	заданий доступен ранг С. Игра	заданий доступен ранг С. Игра	
	останавливается. Отображается	остановилась. Отобразилось	
	изображение сообщением о том, что игра	изображение сообщением о том, что игра	
	пройдена и кнопками «Начать с нуля» и	пройдена и кнопками «Начать с нуля» и	
	«Продолжить».	«Продолжить».	
	161. Игра запускается.	161. Игра запустилась.	
	162. Загружается сцена «Город» с	162. Загрузилась сцена «Город» с	
	кнопками «Инвентарь» и «Меню».	кнопками «Инвентарь» и «Меню».	
	Отображается текущий ранг.	Отобразился текущий ранг.	
	163. Загружается сцена «Зал гильдии» с	163. Загрузилась сцена «Зал гильдии» с	
	кнопами «Меню», «Инвентарь»,	кнопами «Меню», «Инвентарь»,	
	«Выход». Отображается текущий ранг, на	«Выход». Отобразился текущий ранг, на	
	доске заданий доступен ранг Е. Игра	доске заданий доступен ранг Е. Игра	
	останавливается. Отображается	остановилась. Отобразилось	
	изображение сообщением о прохождении	изображение сообщением о прохождении	
	игры, кнопками «Начать с нуля» и	игры, кнопками «Начать с нуля» и	
	«Продолжить».	«Продолжить».	
	164. Игровой прогресс сброшен.	164. Игровой прогресс сброшен.	
	165. Повторяются шаги 20-25.	165. Повторились шаги 20-25.	
	166. Происходит выстрел, враг теряет	166. Произошёл выстрел, враг потерял	
	единицу жизни	единицу жизни	