

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования

**«САРАТОВСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ
ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ Н. Г. ЧЕРНЫШЕВСКОГО»**

Кафедра математической кибернетики и компьютерных наук

**РАЗРАБОТКА ГРАДОСТРОИТЕЛЬНОГО СИМУЛЯТОРА С
АКЦЕНТОМ НА РЕАЛИСТИЧНЫЙ ГОРОДСКОЙ ТРАФИК**

АВТОРЕФЕРАТ БАКАЛАВРСКОЙ РАБОТЫ

студента 4 курса 411 группы

направления 02.03.02 — Фундаментальная информатика и информационные
технологии

факультета КНиИТ

Фирсова Максима Сергеевича

Научный руководитель

доцент, к. ф.-м. н.

А. С. Иванова

Заведующий кафедрой

к.ф.-м.н., доцент

С. В. Миронов

Саратов 2024

ВВЕДЕНИЕ

В конце XX века технологии развивались стремительными темпами, и компьютеры стали неотъемлемой частью жизни многих людей. Из инструментов для служебного и лабораторного использования они превратились в устройства, доступные широкому кругу пользователей. С появлением персональных компьютеров люди не только начали работать с офисными программами и мультимедийными приложениями, но и увлеклись компьютерными играми, которые стали неотъемлемой частью нашей культуры и развлечений.

Существует большое количество жанров компьютерных игр, некоторые игры становятся культовыми. Особое место среди игровых жанров занимают стратегии, в частности, градостроительные симуляторы. Эти игры позволяют игрокам строить и управлять городами, воплощая свои архитектурные и управленческие идеи.

Цель бакалаврской работы — создание градостроительного симулятора с реалистичной системой трафика. Проект направлен на создание интересной игры, предлагающей пользователям интересный опыт управления виртуальным городом, включая сложную систему трафика.

Для выполнения основной цели определены следующие задачи:

1. исследовать возможности игрового движка Unity3D при создании игровых приложений;
2. реализовать интерфейсы для игровых объектов;
3. создать класс, отвечающий за позиционирование объектов на игровом поле;
4. реализовать игровую логику, включая симуляцию экономики;
5. реализовать механизм прокладки путей возможного движения игровых объектов на игровом поле в ходе игры;
6. интегрировать в приложение алгоритм поиска пути в графе A^* для оптимизации движения игровых объектов.

Эти шаги обеспечат создание качественного градостроительного симулятора, способного привлечь и удержать интерес пользователей, предоставляя им увлекательный игровой опыт с реалистичной симуляцией городского трафика.

Структура и объём работы.

Для решения поставленных задач выполнена выпускная квалификационная работа, включающая в себя введение, 2 основные главы, заключение, список использованных источников из 24 наименований и 2 приложений. Работа изложена на 77 страницах, содержит 34 рисунка.

Первая глава имеет название «Анализ предметной области» и содержит основную теоретическую информацию о технологиях и подходах, которые использовались при разработке игрового приложения.

Вторая глава имеет название «Реализация игрового приложения», данная глава содержит подробное описание процесса выполнения работы.

Выпускная квалификационная работа заканчивается заключением, списком использованных источников, а также приложениями с кодом А-Б.

КРАТКОЕ СОДЕРЖАНИЕ РАБОТЫ

Анализ Предметной области

Данная глава состоит из нескольких разделов: «Актуальность компьютерных игр», «Градостроительные симуляторы», «Применяемые технологии».

Актуальность компьютерных игр. Компьютерные игры стали важной частью современной культуры и быстрорастущим сектором индустрии развлечений. Их влияние на общество и экономику растет с увеличением числа игроков.

С 1950-х годов игры прошли значительный путь развития: от Tennis for Two (1958) и Spacewar! (1962) до сложных игр 1990-х годов, таких как «Doom». В 2000-х игры стали разнообразнее и совершеннее. Современные консоли, онлайн-игры, инди-игры и новые технологии, такие как виртуальная и дополненная реальность, открывают новые возможности.

Жанр стратегий, начиная с Civilization (1991), стал особенно важным и ключевым элементом киберспорта. Технологическое развитие улучшило графику и механику игр, таких как StarCraft и Age of Empires.

Число игроков в мире достигло более 3 миллиардов, а рынок видеоигр в 2023 году оценивался в более чем 200 миллиардов долларов. Киберспорт стал массовым явлением с более чем 500 миллионами зрителей.

Компьютерные игры влияют на социальные и культурные аспекты жизни, способствуя общению и сотрудничеству, и используются как инструменты

обучения, развивая навыки решения проблем, критического мышления и командной работы.

Градостроительные симуляторы. Градостроительные симуляторы — это поджанр стратегических видеоигр, в которых игроки управляют городами, контролируя аспекты городской жизни, такие как экономика, инфраструктура, социальные услуги и окружающая среда. История жанра начинается с игры SimCity, выпущенной в 1989 году, которая стала первой вехой в жанре и привела к выпуску множества продолжений и аналогичных игр.

В 2015 году выпуск Cities: Skylines установил новый стандарт в жанре, внедрив сложные модели управления трафиком, экологические симуляции и социальные механики. Игры, такие как Tropico и серия Anno, предлагают уникальные подходы к городскому управлению.

Современные градостроительные симуляторы предлагают разнообразный геймплей с различными режимами: экономический симулятор, песочницу, симуляторы трафика и проекты дизайна собственного города. Передовые игровые движки, такие как Unity и Unreal Engine, упростили процесс разработки игр, позволяя даже небольшим командам создавать высококачественные симуляторы.

История градостроительных симуляторов показывает эволюцию жанра от простых текстовых игр до сложных симуляций, предоставляя игрокам разнообразный опыт управления виртуальными городами.

Применяемые технологии. Данный подраздел состоит из трех частей: «Игровые движки», «Игровой движок Unity» и «Язык программирования C#».

Игровые движки. Игровые движки — это программное обеспечение, облегчающее разработку компьютерных игр, особенно коммерческих проектов. Они предоставляют инструменты для создания игрового мира, графики, анимации, физики, звука и других аспектов. Игровой движок также поддерживает кроссплатформенность, позволяя играм работать на различных устройствах.

Современные игровые движки включают визуальный редактор, системы управления памятью, графику, ввод, аудио, физику, сети, искусственный интеллект, многопоточность и скриптовый язык. Существует три типа движков:

1. С открытым исходным кодом: Полностью доступные для использования, как Godot.
2. Бесплатные или условно-бесплатные: Предоставляют инструменты бес-

платно или за процент от дохода, как Unreal Engine, Unity и CryEngine.

3. Платные или частные: Специализированные движки, разработанные для конкретных игр, как Frostbite и Rage.

Популярные платформы, такие как Steam, Origin и Epic Games Store, собирают статистику по использованию движков. Unity является одним из самых популярных движков. Игровые движки играют ключевую роль в развитии индустрии, предоставляя разработчикам мощные инструменты для создания качественных игр и способствуя инновациям.

Игровой движок Unity. Для работы выбран игровой движок Unity, популярный среди независимых разработчиков. Созданный в 2005 году, Unity поддерживает кроссплатформенность и предлагает мощные инструменты для графики, анимации и физики. Платформа включает библиотеку ассетов и Unity Asset Store для покупки и продажи контента.

Unity имеет низкий порог вхождения, удобный интерфейс и доступную ценовую политику. Бесплатная версия ограничена доходом до 100 тысяч долларов в год, а подписка стоит недорого. Недостатки включают требования к оптимизации и возможные проблемы с весом мобильных приложений.

Unity используется для создания разнообразных игр, от простых до сложных проектов. Среди успешных игр на Unity: Rimworld, Subnautica, Pokemon Go, Cities: Skylines, Rust, Hearthstone, Cuphead и Untitled Goose Game. В дипломном проекте будет использована 3D версия Unity.

Язык программирования C#. Основным языком программирования в Unity является C#. Этот объектно-ориентированный язык был разработан Microsoft и используется для различных приложений, включая настольные, веб-приложения, мобильные и облачные сервисы.

C# интегрируется с .NET, поддерживает асинхронное программирование, автоматическое управление памятью и типовую безопасность. Он используется в Unity для создания скриптов, управляющих поведением игровых объектов, анимацией, физикой и взаимодействием с пользователем.

Скрипты в Unity пишутся как классы, наследуемые от MonoBehaviour, что позволяет управлять игровым процессом через методы, такие как Start и Update. Unity API и механизм корутин упрощают создание сложных игровых процессов, обеспечивая высокую гибкость и продуктивность разработки.

Реализация игрового приложения

Данная глава состоит из следующих частей:

1. Создание базовых объектов.
2. Пользовательский интерфейс.
3. Игровая экономика и симуляция игрового времени.
4. Работа с игровыми объектами.
5. Система строительства экономических объектов.
6. Строительство дорожных объектов.
7. Создание дорожного транспорта.
8. Система управления транспортом.
9. Симуляция дорожного движения.

Создание базовых объектов. Для игрового проекта важны базовые объекты: камера, освещение, игровое поле и скрытые объекты, отвечающие за логику игры.

Основные объекты: `Directional Light`, `CameraAnchor` и `Main Camera`. Они обеспечивают освещение сцены, управление камерой и визуализацию игрового мира. `Directional Light` создаёт реалистичные тени и эффекты. `CameraAnchor` упрощает управление камерой, отделяя её движение от настроек. Скрипт `CameraController.cs` регулирует перемещение камеры, поворот и зум.

`Main Camera` визуализирует сцену. Она настроена для работы в перспективном режиме с полем зрения 30-60 градусов. Скрипт `CameraSmooth.cs` обеспечивает плавное перемещение камеры и стабильную частоту кадров.

Игровое поле (`Ground`) — основа сцены, где размещаются все объекты. Оно имеет визуальную сетку для удобства строительства и физику, предотвращающую провал объектов.

Скрытые объекты управляют логикой игры. `GameManager` координирует основные аспекты игры, взаимодействует с объектами и симулирует игровой процесс. `TrafficManager` управляет движением на дорогах, создаёт транспортные узлы и маршруты. Скрипты `CarSpawner` и `DrivingTestExecutor` отвечают за создание машин и тестирование вождения.

Пользовательский интерфейс. Пользовательский интерфейс (UI) важен для взаимодействия игрока с приложением через визуальные элементы. В Unity он создаётся с помощью `Canvas` и включает кнопки, текстовые поля и панели.

Основные элементы UI:

- Toolbar: содержит основную игровую информацию, кнопку для непрерывного строительства, кнопку паузы и индикаторы состояния;
- Traffic Toolbar: предназначен для взаимодействия с трафиком;
- PopUp Window: всплывающее информационное окно, выводящее информацию о выбранном объекте.

Особенности UI:

- Иконки в Toolbar: представляют функции для строительства различных объектов, управляемые скриптом BuildingPlacer.cs;
- SerialBuilder: функция непрерывного строительства, активируемая кнопкой или клавишей "B";
- Статистика в StatsText: выводит информацию о времени игры, бюджете города, затратах на строительство и другие экономические показатели;
- Модуль управления временем: включает кнопку паузы, индикатор состояния паузы и временную шкалу до обновления экономики;
- TextMeshPro: используется для улучшения качества отображения текста в Unity.

Эти элементы обеспечивают удобное управление игровым процессом и информацией для игрока.

Игровая экономика и симуляция игрового времени. Экономика играет ключевую роль в градостроительных симуляторах, добавляя стратегический элемент в игровой процесс. Она включает четыре основных ресурса: население, рабочие места, товары и деньги.

Основные аспекты экономики:

- Деньги: основной ресурс, используемый для строительства и улучшения инфраструктуры. Каждое здание требует определенные затраты, которые зависят от количества рабочих мест и их доходности;
- Формула для денег: $\text{Деньги} = \text{количество рабочих мест} \times \text{доходность рабочих мест}$. Эта формула используется в функции CalculateMoney скрипта City.cs для подсчета бюджета города;
- Рабочие места и товары: количество рабочих мест зависит от текущего населения и числа доступных рабочих мест. Товары подсчитываются на основе производства коммерческих зданий.

Эти элементы экономики контролируются и управляются через скрипт City.cs, который обеспечивает подсчет бюджета, управление временем и другие игровые

аспекты.

Работа с игровыми объектами. Созданная система работы с игровыми объектами предоставляет пользователю ряд функций для управления игровыми элементами: строительство объектов, получение информации о объектах, взаимодействие с экономикой, логикой и трафиком, удаление объектов. Основные скрипты и функции:

- **BuildingPlacer:** основной скрипт, отвечающий за строительство зданий. Реализует функции отображения временного варианта перед установкой, поворота здания и отключения режима строительства;
- **Пресеты (Presets):** используются для быстрого создания объектов с одинаковыми параметрами, повышая производительность и консистентность проекта. Пресеты сохраняют необходимые параметры, включая экономические, и упрощают модернизацию проекта;
- **Всплывающее информационное окно:** выводится при выделении игрового объекта левой кнопкой мыши, предоставляя информацию и возможность удаления объекта клавишей «Delete».

Эти элементы системы обеспечивают удобство и эффективность в работе с игровыми объектами, что критически важно для комфортного геймплея и развития проекта.

Система строительства экономических объектов. В градостроительном симуляторе основная функция — строительство экономических объектов. Пользователь может предварительно видеть здание перед его строительством. Основные типы экономических объектов включают жилые, производственные и коммерческие здания. В Unity здания представляют собой объекты на игровой сцене, для создания которых используются префабы. Префабы позволяют создавать, настраивать и повторно использовать игровые объекты, упрощая управление и обновление всех экземпляров одного типа объекта сразу. Реализованные в проекте типы зданий могут быть дополнены новыми благодаря удобной системе экономических пресетов, которые влияют на подсчет экономических показателей в файле `City.cs`.

Строительство дорожных объектов. Дороги в градостроительных симуляторах играют ключевую роль, обеспечивая транспортную систему и поддерживая развитие города. Они представляют собой основные элементы инфраструктуры, необходимые для социального и экономического развития. Система дорог

включает в себя простые прямые участки и перекрестки, которые легко расширяются и дополняются новыми элементами. Дороги в проекте реализованы как префабы с узлами, которые логически соединяют различные сегменты дорожной сети, обеспечивая возможность движения по определенным направлениям. Перекрестки, в свою очередь, играют важную роль в соединении различных дорог и поддержании функциональности изменения направлений движения.

Создание дорожного транспорта. В градостроительных симуляторах реализация машин требует аккуратной работы с их физическими и визуальными моделями. Основные требования к машинам включают реалистичную физику движения, взаимодействие между машинами и интеграцию в игровой мир. Для этого используются префабы, содержащие физические и визуальные модели, а также специальные элементы для управления поведением колес и движением. Критическим компонентом является использование технологии Raycast для обеспечения "зрения" машины и избегания столкновений с объектами в окружении.

Создание дорожного транспорта. Трафик в градостроительных симуляторах играет ключевую роль в развитии города, влияя на экономику и качество жизни жителей. Симуляция трафика основана на создании дорожной сети, которая включает в себя маршрутные точки, дороги и перекрестки. Каждая маршрутная точка определяет параметры направления движения, что позволяет машинам перемещаться по городу. Для построения и оптимизации дорожной сети используется специализированный скрипт TrafficNodeManager.cs. Оптимальные маршруты вычисляются с помощью алгоритма A*, который находит кратчайший путь между точками в сети дорог. Полученные маршруты передаются машинам для визуализации и исполнения в игровом мире.

ЗАКЛЮЧЕНИЕ

В ходе выполнения работы был разработан градостроительный симулятор на игровом движке Unity. Данный движок, будучи одним из самых популярных и мощных игровых движков, предоставил все необходимые инструменты для создания современного и комплексного игрового проекта. Благодаря использованию игрового движка, удалось создать качественную игру, что подчёркивает его привлекательность для индивидуальных разработчиков и небольших команд.

Процесс разработки включал несколько важных этапов: анализ потребностей и определение целевой аудитории, проектирование и концептуализация, разработка прототипов и макетов. Анализ потребностей позволил лучше понять рынок и определить, что именно хотят видеть пользователи в игре. Этап проектирования включал разработку концепции игры, определение основной механики, сценария и архитектуры игрового мира, что обеспечило создание интересной и логически завершённой игры.

В ходе выполнения данной работы были приобретены ценные знания и навыки в программировании на языке C# и работе с платформой Unity. Были исследованы основы проектирования игровых механик, разработка сложных систем трафика и взаимодействие различных игровых элементов. Реализация проекта позволила освоить множество аспектов разработки игр, включая создание и интеграцию различных модулей, оптимизацию производительности и тестирование.

В результате выполнения данной работы была достигнута основная цель проекта — «разработка градостроительного симулятора с акцентом на реалистичный городской трафик». Проект предлагает пользователям интересный опыт управления виртуальным городом, включая систему трафика и экономику, что делает его интересным и увлекательным. При этом проект обладает большим потенциалом к развитию. Проект поддерживает возможность добавления новых зданий, экономических и социальных параметров, улучшение системы трафика и проведение его различных симуляций и исследований.

Все поставленные задачи были успешно выполнены, и проект является логически завершённым. Основываясь на полученных знаниях и навыках, можно утверждать, что Unity предоставляет широкие возможности для разработки игр любого масштаба.