

ДОКЛАД

Слайд 1. Титульный

Здравствуйте! Меня зовут [REDACTED], и вместе с Д [REDACTED] мы подготовили курсовой проект по дисциплине «Программирование» на тему: **Игровая программа “Тамагочи”**. Руководителем проекта является [REDACTED]. Сегодня мы расскажем о сути проекта, этапах разработки и его возможностях.

Слайд 2. Содержание

Вот структура нашей презентации. Мы начнём с введения, далее расскажем о целях проекта, об анализе жанра, о том, какие функции реализованы в программе, как она работает, и в завершение — подведём итоги и расскажем, что можно улучшить.

Слайд 3. Введение

Компьютерные игры сегодня — это увлекательный и интересный процесс, который стал особенно популярным в последнее время с множеством жанров: стратегии, шутеры, квесты, симуляторы, файтинги. Нас заинтересовал именно жанр **симуляторов**, а конкретно — игр, в которых нужно ухаживать за виртуальными существами. Такие игры просты по механике, но при этом вызывают эмоциональную привязанность у игроков.

Слайд 4. Задачи курсового проекта

Нам предстояло провести анализ предметной области: изучить ключевых представителей жанра такие как: Dogz and Catz, Neko Atsume и Pou и выявить их основные механики. На основе полученных данных спроектировать собственный симулятор питомца с классическим набором функций.

Слайд 5. Компьютерная игра Tamagotchi

Теперь перейдём к самой классике — игре Tamagochi, которая впервые предложила именно такой симулятор виртуального питомца. В ней игроку предстояло:

Следить за состоянием существа, поддерживая его здоровье, сытость и настроение;

Предпринимать все возможные действия, чтобы питомец выжил как можно дольше;

В процессе развивать в себе привычку заботы и эмоциональную вовлечённость.

Именно эти ключевые принципы и стали отправной точкой для нашего проекта.

Слайд 6. Функции в программе

В нашей программе реализован полный набор функций для взаимодействия с питомцем. Пользователь может ухаживать за любимцем — кормить, гладить и менять ему имя или внешний вид — при этом игра завершится, если забота прекратится на длительный срок. Параметры голода, здоровья и настроения меняются автоматически со временем. Они всегда отображаются на экране вместе с уведомлениями о критических состояниях. Визуальную мягкость создают плавные анимации, а звуковое оформление и фоновая музыка усиливают погружение. Наконец, несколько слотов сохранений позволяют возвращаться к разным этапам игры без потери прогресса.

Слайд 7. Разработка программы

Проект реализован на языке Си с использованием библиотеки SDL2 — это даёт высокую производительность и кроссплатформенность. Программа построена по модульному принципу: каждая часть (ввод, логика, рендеринг, сцены) реализована отдельно и разбита по файлам. Это облегчает поддержку и масштабирование проекта. Интерфейс адаптируется под

разрешение экрана, благодаря чему игра удобно работает на разных устройствах.

Слайд 8. Обобщённый алгоритм программы

Работа программы выглядит так:

При запуске пользователь видит стартовый экран с информацией о программе. После нажатия кнопки “Старт” пользователь переходит на сцену меню, где может вернуться либо назад к информации, выключить программу или еще раз нажать кнопку “Старт”. Если пользователь желает начать играть то следующая сцена будет связана с выбором сохранения от которого он может начать играть (всего их 3), если сохранение под таким номером не существует то оно создается со стартовыми значениями. Затем запускается основная сцена с питомцем. Игрок может кормить его, гладить или настраивать. Питомец реагирует на действия: изменяются параметры. Изменения сразу отражаются на экране. При выходе игра будет автоматически сохранять прогресс.

Слайд 9. Структура питомца

Вся информация о питомце хранится в специальной структуре данных. Там содержатся: Имя, координаты на экране, внешний вид и масштаб изображения; Значения параметров: здоровье, голод, настроение; Счётчик времени жизни, позволяющий вести статистику питомца. Эта структура позволяет быстро загружать и сохранять состояние питомца.

Слайд 10. Сохранения в программе

При выходе из игры или игровой сцены все данные о питомце автоматически записываются в файл. При следующем запуске сохранения игра сразу восстанавливает состояние. Сохраняется внешность, положение на экране и состояние питомца — игрок может продолжить точно с того места, где закончил. Также на слайде представлен скриншот как это меню выглядит в игре.

Слайд 11. Игровой процесс

На данном слайде представлены скриншоты из программы: первый скриншот это основное меню игры, второй это уже игровой процесс и третий это меню кастомизации.

Слайд 12. Подведение итогов

Мы создали **полноценную копию** легендарного "Tamagotchi" — виртуального питомца, за которым нужно ухаживать. Но наша версия не просто повторяет оригинал, а **расширяет его возможности!** Нами был получен опыт работы с кроссплатформенностью, а точнее программа была портирована на такие системы как: Windows, Linux и Android. Получены навыки работы в команде, использован Git для контроля версий и совместной разработки. Также итоговая программа готова к дальнейшему расширению: ввести систему достижений, игровую валюту для покупки скинов и мини-игры для ее заработка.