**1 слайд:**

Приветствие:

Здравствуйте уважаемая комиссия. Я, Пронин Арсений, представляю вам свой проект - **Разработка платформы для многопользовательских настольных игр.**

**2 слайд:**

Связи с тем, что все мы находимся на больших расстояниях и часто не хватает времени на любимое хобби, подобная платформа заинтересует тех, кто увлекается настольными играми. Её основное преимущество перед обычными настольными играми заключается в том, что на каком бы расстоянии игроки не находились, они могут сыграть друг с другом. Также плюсом, является мгновенная подготовка – не нужно расставлять фишки и фигурки перед каждой игрой, компьютер всё сделает за вас.

**3 слайд:**

**Цель проекта:**  Разработка клиент-серверного приложения для многопользовательских игр, с возможностью развития. Подключившись к серверу пользователь может выбрать игру из списка доступных, инициировать новый сеанс игры, или же присоединиться к существующему сеансу.

**4 слайд:**

Из уже существующих приложений для настольных игр можно выделить **Tabletop Simulator -** симулятор популярных классических настольных занятий. А из отечественных платформ данного жанра, можно отметить игры из **Вконтакте**, например монополия или мафия.

**5 слайд:**

В отличии от приведённых примеров, моё приложение будет составлено из разных логических настольных игр. В подобных играх отсутствует элемент случайности, из-за чего исход игры зависит только от умений игроков.

Первыми настольными играми были выбраны Коридор и Кварто, так как еще не реализована их онлайн-версия, следовательно необходимо было изучить их правила.

**6 слайд (используемые инструменты и почему):**

**Основные инструменты используемые для реализации проекта:**

* Язык программирования C++
* Кроссплатформенный фреймворк Qt
* Среда разработки Qt creator

Qt Creator — кроссплатформенная свободная (среда разработки) IDE на языках С, С++ и QML. Поддерживает много компиляторов (GCC, Clang, MinGW, MSVC, Linux ICC, GCCE, RVCT, WINSCW). Например, проект сейчас компилируется с помощью двух компиляторов: clang и gcc(mingw)

Основная задача Qt Creator — упростить разработку приложения с помощью фреймворка Qt на разных платформах.

**Плюсы Qt:**

- дизайнер окон

- кроссплатформенность

- много разных, полезных и удобных классов (сеть, графика и т.д.)

- механизм сигналов и слотов

- подробная документация с примерами

**Единственным весомым минусом является**

- использование мета-объектного компилятора, из-за чего код написанный с помощью Qt нельзя скомпилировать на другом компьютере без установки фреймворка.

**Дополнительно:**

* Для хранения исходников используется система Git (на портале github)
* Для построения документации системы используется Doxygen (с помощью неё в автоматическом режиме получены диаграммы представленные в презентации)

Для того чтобы начать работу мне пришлось изучить систему управления версиями (version control system) Git и библиотеку Qt.

**7 - 8 слайд (реализация проекта):**

Реализация приложения началась с разработки консольной версии игры Коридор. Когда же начались работы с графикой, я, используя уже существующий алгоритм логики игры, реализовал офлайновый вариант игры на одном компьютере, уже с визуализацией.

Разработка носила циклический характер: написание нового кода, его отладка и последующий рефакторинг, в ходе которого код обычно сильно упрощался благодаря использованию ООП и известных паттернов проектирования.

**9 слайд (диаграмма класса сервера):**

Параллельно с реализацией игры Коридор создавалась серверная часть приложения. Для реализации клиент-серверного общения был выбран стандартный протокол передачи данных TCP и передача сообщений через сокеты. Сначала клиент отправлял серверу строковые сообщения, но в дальнейшем были использованы типизированные команды.

По мере развития клиент вобрал в себя большую часть приложения игры и начал создавать команды, несущие данные о действиях игрока и событиях игры, которые отправлял на сервер.

В итоге игра Коридор полностью заработала в онлайн режиме и после этого была добавлена игра Кварто.

**10 слайд:**

Демонстрация: на примере двух клиентов подключающихся к серверу…

1. Подключение
2. Создание лобби
3. Просмотр список лобби
4. Подключение/Отключение и другие взаимодействия с лобби
5. Запуск игры, игра (выиграть/отключиться до окончания игры)
6. Много лобби одновременно

**11 слайд:**

Перейдём к выводам

* И так разработана платформа для многопользовательских настольных игр
* Платформа имеет возможность дальнейшего развития
* Платформа находится в свободном доступе на портале GitHub
* Цель проекта была достигнута