**1 слайд**

Добрый день уважаемая комиссия. Меня зовут Пинчук Владислав Андреевич. Тема моей дипломной работы “Программный комплекс по выбору лекарственных препаратов на основе экспертных знаний”.

**2 слайд**

Для начала рассмотрим цели данной дипломной работы.

* предоставить возможность создания базы заболеваний и методов их лечения;
* предоставить доступ к базе любому пользователю;
* экономить время пользователей, упрощая процесс назначения лечения;
* частично разгрузить различные органы здравоохранения.

**3 слайд**

Основные функции приложения представлены на данной диаграмме. Стоит отметить, что приложение имеет систему ролей. Таким образом действуют две роли, а именно: эксперт и пользователь.

Пользователю доступны следующие функции:

* 1. Поиск заболевания по симптомам
  2. Просмотр подробной информации о заболевании
  3. Создание отзыва о заболевании

Эксперту доступны следующие функции:

1. Создание, редактирование, удаление симптомов;
2. Создание, редактирование, удаление рекомендаций;
3. Создание, редактирование, удаление заболеваний;

Поскольку эксперты постоянно должны поддерживать базу в актуальном состояний, то для этого были добавлены следующие функции.

1. Анализ отзывов о заболевании;
2. Анализ рейтинга заболевания.

**4 слайд**

На данном слайде представлена архитектура приложения. Все приложение можно разделить на 4 основных блока.

* Серверное приложение;
* Браузерное приложение;
* База данных;
* Сервис для интеграции с другими сервисами для получения данных о лекарственных средствах

**5 слайд**

Для реализации приложения использовалась среда исполнения Node.JS. Данная среда использовалась для разработки как серверного приложения, так и браузерного. Сама среда использует язык программирования JavaScript. Данный язык имеет динамическую типизацию. Данный подход имеет как свои плюсы, так и кучу минусов. Чтобы избавиться от динамической типизации и перейти к более строгой, статической типизации была применена соответствующая надстройка на языком JavaScript.

Таким образом для разработки приложения был использован строго типизированный язык программирования TypeScript. Поскольку среда исполнения Node.js не поддерживает данный язык, то сначала язык TypeScript нужно преобразовать в JavaScript путем компиляции. Для этого используется TypeScript компилятор. Таким образом большинства багов, свойственные JavaScript приложениям, можно избежать на стадии компиляции, а не на стадии исполнения.

**6 слайд**

Для разработки серверного приложения использовались следующие технологии. Для быстрого написания Rest Api приложения использовалась библиотека express. Для передачи данных использовался протокол http и формат, для отправки данных, JSON. Для хранения данных использовалась реляционная база данных MySql. Для удобной работы с базой данных использовалась ОРМ Sequelize. Данный пакет предоставляет удобный интерфейс для создания миграций и сидов для базы данных.

**7 слайд**

На данном слайде представлена схема базы данных. База данных состоит из 14 таблиц. Можем увидеть, что у таблицы Illnesses есть связи с Medicines, Recommendations и Symptoms.

**8 слайд**

Браузерное приложение было реализовано в виде SPA. Для реализации одностраничного приложения использовалась библиотека React. Хранение данных в приложении происходит в едином хранилище. Для реализации данного хранилища использовалась библиотека Redux. Интерфейс в браузерном приложении является самой важной частью, поскольку потенциальный пользователь взаимодействует только с интерфейсом. Для реализации интерфейса использовалась библиотека Material UI. Чтобы использовать последние возможности языка, а также поддерживать все актуальные браузеры использовался пакет Babel. Данный пакет преобразует последние стандарты языка в предыдущие.

**9 слайд**

Чтобы собрать браузерное приложение используется сборщик Webpack. После успешной сборки всего проекта, где находится множество различных файлов, мы получим всего несколько файлов, основные из них: html, js, css.

**10 слайд**

При разработке приложения также использовались дополнительные инструменты. Использовалась одна из самых популярных систем контроля версий GIT. Также использовались пакеты ESLint и CommitLint, которые проверяют перед каждым коммитом соответствено стиль написания кода и формат коммита.

**11 слайд**

Далее приведены скриншоты интерфейса приложения. На данном слайде предоставлена страница для поиска заболеваний. Для более точного поиска заболеваний можно ввести любое количество симптомов. После успешного поиска, в таблице появятся заболевания. В данной таблице пользователь также видит и рейтинг заболевания.

**12 слайд**

На данном слайде представлена страница с подробным описанием заболевания и методах его лечения. Чтобы оставить отзыв о полезности предоставленной информации предоставлена соответствующая кнопка.

**13 слайд**

На данной странице пользователь может оценить каждый раздел, а также оценить каждое лекарство. Для подробного описания предоставлено соответствующее поле.

**14 слайд**

Данная страница предназначена для управления всеми существующими заболеваниями. Каждое заболевание можно отредактировать, удалить или просмотреть его рейтинг, для чего предоставлены соответствующие иконки.

**15 слайд**

На данной странице представлены два графика. На первом эксперт может просмотреть рейтинг заболевания на протяжении всего времени с момента его создания. Второй график предоставляет рейтинг отдельно выбранного лекарственного средства.

**16 слайд**

Также на данной странице доступен список всех отзывов о данном заболевании. Список представлен в виде выпадающей панели.

**17 слайд**

Последнее что было сделано, это тестирование приложения. Были написаны тест-кейсы, которые представлены в пояснительной записке, для основных функций. Все исключительные ситуации в приложении обрабатываются и выводятся соответствующие сообщения. Все поля ввода имеют соответствующие проверки и при возникновении ошибки, появляется сообщение.

**18 слайд**

В ходе дипломной работы был разработан программный комплекс по подбору лекарственных средств на основе экспертных знаний. Все поставленные задачи были реализованы.