Колледж ТОО "Astana IT University"

**Отчет**

по производственному обучению

Медицинская система управления пациентами и их медицинской историей.

Группа: ПО2209

ФИО студента:

Болатбекулы Елдар

Амралин Арлан

ФИО преподавателя:

Тулабоев К. С.

г. Астана

**Содержание**

|  |  |  |
| --- | --- | --- |
|  | Введение | 33 |
| 1 | Основная часть | 44 |
| 2 | Архитектура и Технологии | 95 |
|  | Основные функции сайта | 88 |
|  | Заключение | 110 |

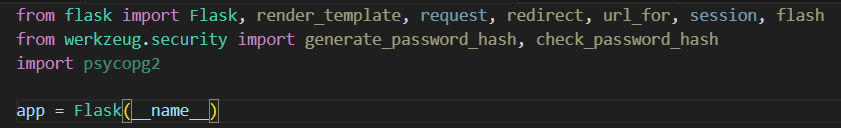
**Введение:** проект (Больница XYZ) была создана студентами группы ПО2209к Болатбекулы Елдар и Амралин Арлан. Кратко о проекте, это Веб-приложение в котором могут работать работники больницы или других медицинских учреждений, веб приложение было создано для удобства и быстроты поиска и редактирования данных в базе данных. Так же база данных в себе хранит 16 таблиц.

**Основная часть:** Разработкой веб приложения и базы данных, так же частично сайтом занимался студент группы ПО2209 Болатбекулы Елдар, основную часть веб приложения разрабатывал студент группы ПО2209к Амралин Арлан.как писалосьвыше, база данных имеет 16 таблиц, сами таблицы приведены в «Рисунок 1»

**Архитектура и Технологии**

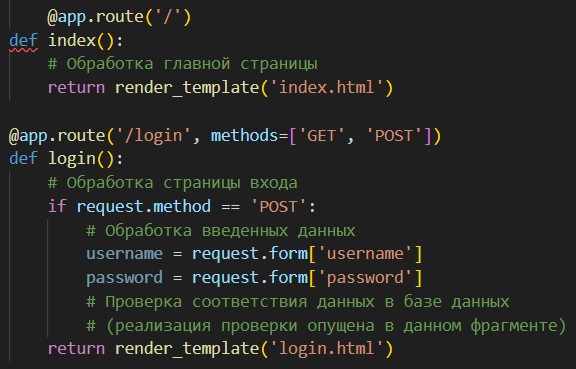
Так же при созданий Веб-приложения на языке программирования python, использовались следующие библиотеки (Flask, psycopg2).

**1. Flask-приложение**



«Рисунок 1»

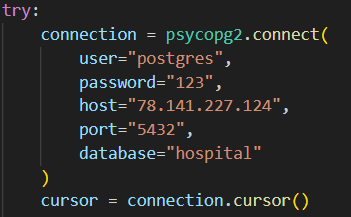
Этот код создает Flask-приложение и импортирует необходимые модули.

**2. Маршруты и обработчики запросов**

«Рисунок 2»

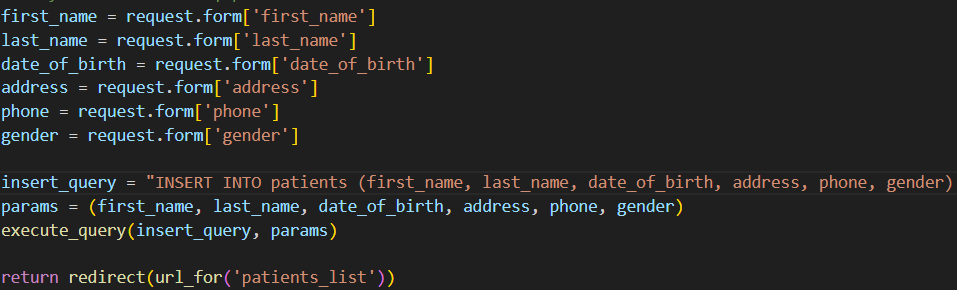
Здесь определены некоторые базовые маршруты, такие как главная страница и страница входа. Реальные обработчики запросов, в которых происходит взаимодействие с базой данных, опущены для краткости.

**3. База данных и psycopg2**



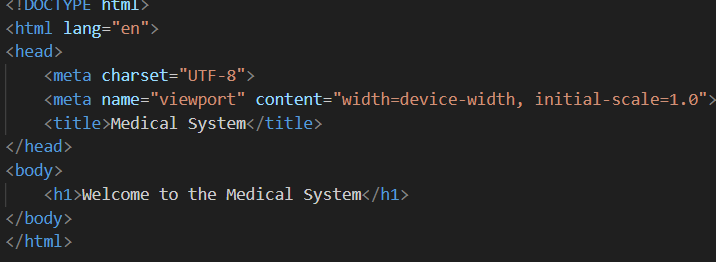
«Рисунок 3»

Здесь устанавливается соединение с базой данных PostgreSQL с использованием библиотеки psycopg2.



«Рисунок 4»

Выполнение SQL-запросов происходит через курсор. Здесь показан пример запроса данных из таблицы "patients". Реальные запросы зависят от конкретных потребностей вашего приложения.



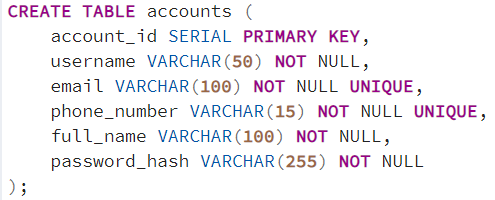
**4. Шаблоны HTML**

«Рисунок 5»

Здесь приведен простой HTML-код для главной страницы. Реальные шаблоны должны включать в себя элементы для взаимодействия с данными из Python-кода.

**Регистрация и Аутентификация**

* 1. Форма Регистрации: Имеется отдельная страница на сайте с регистрацией. Запрашиваются следующие данные:



«Рисунок 6»

1.2 Проверка Уникальности: При вводе данных происходит проверка уникальности электронной почты или другого идентификатора для предотвращения дублирования учетных записей.

1.3 Сохранение в БД: После успешной проверки учетная запись сохраняется в базе данных.

**Управление Пациентами**

2.1 Создание и Регистрация Пациентов:

На основе уникального идентификатора (например, номера страховки) создаются медицинские карты пациентов.

2.2 Регистрация включает в себя ввод основных данных: ФИО, дата рождения, контактная информация.

2.3 Медицинская История и Электронные Записи:

Ведение электронной медицинской карты, содержащей информацию о прошлых и текущих посещениях, диагнозах, предписанных лекарствах и процедурах.

правление Рецептами и Лечением

Запись и отслеживание выписанных рецептов, лекарств и назначенных процедур.

2.4 Лабораторные Исследования и Медицинские Тесты:

Отслеживание результатов лабораторных исследований, медицинских тестов и обследований.

2.5 Связь с Врачами и Медперсоналом:

Возможность взаимодействия пациента с врачами, запрос консультаций, обмен сообщениями.

**Управление Врачами и Специальностями**

Регистрация и Учет Врачей:

Создание профилей врачей с основной информацией: ФИО, специализация, контактные данные.

Управление Специальностями:

Создание списка медицинских специальностей и их учет в системе.

Назначение Врачей на Должности:

Привязка врачей к конкретным медицинским специальностям и должностям.

График Работы и Прием Пациентов:

Формирование графика работы врачей и их доступности для приема пациентов.

Учет Медицинских Записей и Рецептов:

Возможность врачей создавать и управлять медицинскими записями, рецептами и предписаниями.

**Назначения и Рецепты**

**3.1 Назначения Лечения:**

Врачи могут создавать назначения для пациентов, включая медикаменты, процедуры и рекомендации.

3.2 Электронные Рецепты:

Выписка электронных рецептов для пациентов, включая информацию о прописанных лекарствах и дозировках.

3.3 Учет Выданных Препаратов:

Отслеживание выданных препаратов в аптеках на основе выписанных рецептов.

**Лабораторные Исследования и Тесты**

4.1 Результаты Исследований:

Врачи получают электронные результаты лабораторных исследований, что обеспечивает быстрый доступ и улучшает диагностику.

4.2 Интеграция с Медицинскими Записями:

Связь результатов лабораторных тестов с медицинскими записями пациентов для создания полной карты здоровья.

**Хирургические Вмешательства Процедуры**

5.1 Ввод Данных о хирургических процедурах

5.2 Запись Истории медицинских процедур

**Вакцинации и Профилактика:**

Ведение Вакцинационных Журналов:

Создание и поддержание электронных записей о проведенных вакцинациях для каждого пациента.

**Интерфейс**

Интерфейс простой, ничем не выделяется.

**Разработка сайта**

Основную часть разработки сайта занимался студент ПО2209 Амралин Арлан совместно с Болатбекулы Елдаром. Сайт был написан на языках (html, java script, css). Сайт имеет отдельный вкладки для работы с каждой таблицей из базы данных. Внесённые данные так же сохраняются в самой базе данных, и сами данные можно просмотреть на самом сайте.

**Заключение**

Внедрение системы управления медицинской информацией значительно оптимизирует процессы в больнице, повышает эффективность взаимодействия между персоналом и обеспечивает более качественное предоставление медицинских услуг.