Резюме на дипломния проект

1.Представяне

* Уважаема г-жо Директор, Уважаема комисия, уважаеми гости ( ако има такива)

Аз съм Десислава Панчова

Темата на моя дипломен проект е ,,Работа с ORM и Django ORM‘‘

2.Увод

В съвременния дигитален свят управлението на данните е ключов фактор за успешната работа на уеб приложенията. Технологичният напредък налага използването на мощни, ефективни и сигурни механизми за съхранение, обработка и извличане на информация. В този проект ще разгледаме една от най-популярните платформи за разработка на уеб решения – Django, и по-специално неговата интеграция с бази данни чрез обектно-релационния машинен слой (ORM).

Целта на нашия проект е да представи основните принципи на работа с бази данни в Django, да анализира ролята на моделите и връзките между тях, както и да демонстрира как ORM улеснява изпълнението на CRUD операции (създаване, четене, обновяване и изтриване на данни) без необходимост от писане на SQL заявки. Също така практическата част реализира система за анкети, чрез което се илюстрират възможностите на Django ORM.Темата е актуална и приложима във времена на големи информационни натоварвания и бързо променящи се технологии.

3. Описание: Концепция на работа с бази данни в Django

Django използва модели – Python класове, които представят таблици в база данни, а атрибутите на тези класове съответстват на колоните. Този подход, познат като обектно-релационно моделиране (ORM), позволява разработчиците да работят с данни чрез Python обекти, без да пишат директно SQL. Поддържат се различни видове връзки между модели: едно към едно, едно към много и много към много.

4.Реализация на обектно-релационни модели

В практическата реализация са дефинирани модели за анкети (Poll), въпроси (Question) и отговори (Option). Един Poll съдържа множество въпроси (ManyToMany връзка), а всеки въпрос има множество опции (ForeignKey). Всеки модел включва полета с подходящи типове и помощни методи за управление на данните и статистика за отговори.

CRUD операциите се извършват чрез лесни и интуитивни методи на Django ORM, което значително ускорява разработката и намалява риска от грешки. Системата поддържа статистика за броя на отговорите на въпросите и опциите.

5.Обща теория: Видове полета и заявки

Django предлага богата палитра от полета – CharField, TextField, IntegerField, BooleanField, DateTimeField и други, както и полета за качване на файлове и изображения. Заявките се извършват чрез ORM методи като all(), filter(), get(), update() и delete(). Освен това се поддържат сложни заявки с агрегиращи функции, филтриране и сортиране.

6. Описание: Анализ на съществуващите решения и избор на технологии

За релационни бази данни са избрани популярни сървъри като SQLite (по подразбиране), MySQL и PostgreSQL, които Django поддържа изцяло. Проекти като Flask и Ruby on Rails са разгледани като алтернативи, но Django e предпочетен заради своята интегрирана ORM система, вградена административна конзола, добра документация и голяма общност.

7.Заключение

Накратко в моят дипломен проект се доказва, че Django ORM предоставя стабилна и гъвкава рамка за работа с бази данни. Това ни осигурява ефективност, сигурност и бързо развитие на уеб приложения, отговарящи на съвременните изисквания. Усвояването на Django и неговите средства за управление на данни е ключово за всеки софтуерен разработчик, който иска да създава професионални и надеждни системи.

8.Финал

Благодаря Ви за вниманието!

Ако имате някакви въпроси, ще се радвам да ви отговоря!