**МІЖНАРОДНИЙ ГУМАНІТАРНИЙ УНІВЕРСИТЕТ**

**Факультет кібербезпеки програмної інженерії та комп’ютерних наук**

**Кафедра комп’ютерних наук**

**ЗВІТ**

про проходження навчальної практики

здобувача освітнього ступеню «бакалавр»

на базі практики освітнього фонду «KeepSolid”

Виконав: студент IV курсу

Спеціальності

123 «Комп’ютерна інженерія»

Андрушко Ярослав Васильович

Керівник Вакарчук А. О.

Одеса – 2024

ЗМІСТ

ВСТУП

Цей документ містить звіт про проходження навчальної практики студента Андрушка Ярослава Васильовича, яка проводилась на базі освітнього фонду «KeepSolid”.

Згідно з навчальним планом, виробнича практика є важливою складовою підготовки студентів за спеціальністю «Комп’ютерна інженерія». Вона проводиться на третьому курсі й покликана узагальнити та поглибити знання з фахових дисциплін, встановити їхні взаємозв’язки та продемонструвати практичне застосування в різних аспектах програмування. Практика сприяє розумінню студентами особливостей використання теоретичних знань у реальній роботі, адаптації до умов сучасного ринку праці та вимог роботодавців.

Основним завдання практик було отримання практичних навичок по теоретичній базі, яку ми отримали протягом навчання. В рамках цього завдання я розробляв, тестував та влагоджував особистий програмний продукт у вигляді веб-додатку бібліотеки.

# **1 ОПИС БАЗИ ПРАКТИКИ**

* 1. **Відомості про компанію**

Організація ТОВ "КІПСОЛІД УКРАЇНА" зареєстрована 04.07.2011 за юридичною адресою Україна, \*\*4, Одеська обл., місто Одеса, ПРОСПЕКТ ШЕВЧЕНКА, будинок \*\*А. Керівником організації є БОЧАРНІКОВ ОЛЕГ ПАВЛОВИЧ. Розмір статутного капіталу складає 80 000,00 грн. На момент останнього оновлення даних 22.12.2022 стан організації – Не перебуває в процесі припинення.

Види діяльності:

* 1. Комп'ютерне програмування.

77.33 Надання в оренду офісних машин і устаткування, у тому числі комп'ютерів.

58.21 Видання комп'ютерних ігор.

58.29 Видання іншого програмного забезпечення

* 1. Консультування з питань інформатизації.

62.09 Інша діяльність у сфері інформаційних технологій і комп'ютерних систем.

63.11 Оброблення даних, розміщення інформації на веб-вузлах і пов'язана з ними діяльність.

* 1. Веб-портали.
  2. Видання інших інформаційних послуг.

68.20 Надання в оренду й експлуатацію власного чи орендованого нерухомого майна.

Місцем проходження практики був освітній фонд «Кіпсолід” де я вже мав змогу проходити практику.

У цьому році через труднощі практика проходила у дистанційній формі, але це не як не заважало набувати нові знання та навички. Освітній фонд «кіпсолід” був офіційно зареєстрований у 2021 році. Але до цієї події було 15 років співпраці з навчальними закладами міста Одеса, досвід 8 років програми «Літня інтернатура”, 4 роки програми «Вступ до спеціальності” для першокурсників в ІТ спеціальностей ВНЗ Одеси, створення студентських лабораторії у ВНЗ Одеси, спроби співпрацювати з ІТ кластерами та Асоціаціями. Окрім досвіду створення програм для студентів, за ці роки компанія KeepSolid набула й розуміння обмеженості програм, спрямованих лише на відбір найкращих студентів.

Місія фонду – це поліпшити рівень життя в Україні шляхом підвищення рівня ІТ освіти та підготовки спеціалістів, здатних створювати передові ІТ-продуктів.

1.1 Мета та завдання практики

Метою практики є здобуття та закріплення практичних навичок, поглиблення теоретичних знань і їх практичне застосування. Це також передбачає вивчення ринку, аналіз доцільності використання різних технологій у конкретних випадках. У процесі студенти узагальнюють свої дослідження, знаходять рішення для поставлених завдань і сприяють їх впровадженню на підприємстві чи в організації.

Завдання практики включають:

– Опанування методів пошуку, аналізу та обробки інформації для вибору оптимальних систем, технологій і програм з урахуванням економічних та технічних характеристик.

– Робота з технічною та науковою літературою, а також ознайомлення з правилами оформлення технічної документації для програмного забезпечення.

– Засвоєння навичок розробки концепцій комп’ютерних систем, проєктування баз знань і даних, створення прикладного програмного забезпечення для інформаційних систем і технологій, а також оформлення документації до програмного забезпечення.

**1.2 Загальні відомості**

Веб-застосунок для онлайн-бібліотеки – це багатофункціональний інструмент, що дозволяє користувачам швидко і зручно отримувати доступ до книжкових колекцій, переглядати наявність літератури, а також завантажувати, або читати ці книги в онлайні.

Веб-застосунок дозволяє здійснювати пошук книг, проглядання, завантаження, рекомендацій та історії запитів у будь-який час. Електронні бібліотеки слід відрізняти від суміжних структурних типів сайтів, зокрема літературних. На відміну від літературного журналу, який виник як тип друкованого видання, але успішно й без принципових змін структури перейшов в Інтернет, електронна бібліотека не поділяється на випуски й оновлюється постійно в міру появи нових матеріалів.

На відміну від сайтів із вільною публікацією, електронна бібліотека зазвичай наповнюється за вибором координатора проєкту і, що особливо важливо, не передбачає створення навколо опублікованих текстів комунікативного середовища.

Переваги використання веб-застосунків для онлайн-бібліотек:

* Обсяг використання веб-застосунків для бібліотек значно зріс за останні роки, адже вони забезпечують найзручніший доступ до ресурсів.
* До 85% користувачів зазначають позитивний або нейтральний досвід роботи з подібними платформами.
* Більшість сучасних бібліотек інтегрують веб-застосунки для автоматизації обслуговування, що дозволяє значно знизити навантаження на персонал.
* Близько 70% користувачів відзначають зручність таких платформ завдяки швидкому доступу до потрібної інформації.
* Веб-застосунок онлайн-бібліотеки сприяє покращенню досвіду користувачів і створює сучасний підхід до популяризації читання та навчання.

## **Мета проекту та цільова аудиторія**

Метою проекту у мене було створення додатку онлайн бібліотеки згідно з технічним завданням. Використання сучасних стеків технологій і методів автоматизації.

Цільовою аудиторією такого проекту стає максимально широкий спектр людей, якім потрібно зручно і швидко знайти та завантажити книгу будь-якого жанру.

3 РЕАЛІЗАЦІЯ ПРОЕКТНОЇ РОБОТИ

3.1 Вибір стек технологій та теоретична база

Перед початком розробки потрібно вирішити стек технологій, який я збираюсь використовувати в своєму додатку. За основу я вирішив взяти мову Python, на якій досить зручно створювати веб-застосунки. Python – це високорівнева мова програмування з акцентом на простоті та читабельності коду. Вона підтримує кілька парадигм програмування, зокрема об'єктно-орієнтоване, процедурне та функціональне. Завдяки великій кількості бібліотек і фреймворків, таких як Django, Flask або FastAPI, Python ідеально підходить для створення веб-застосунків.

Крім веб-розробки, Python активно використовується в області штучного інтелекту, аналізу даних, автоматизації процесів та розробки ігор. Його популярність серед компаній-гігантів, таких як Google, Facebook, Instagram, Spotify і Netflix, зумовлена широкими можливостями, швидкістю розробки та активною спільнотою, яка постійно вдосконалює екосистему мови.

В якості фреймворку я обрав Django, з яким уже знайомий. Django – це потужний та популярний фреймворк для розробки веб-застосунків на мові Python. Він створений з акцентом на швидкість розробки, масштабованість і безпеку. Django є вибором багатьох розробників завдяки його "батарейкам, що входять у комплект" (batteries included), тобто великій кількості вбудованих інструментів і можливостей, які спрощують розробку.

Основні переваги Django:

1. Швидкість розробки. Django допомагає швидко створювати функціональні прототипи та повноцінні веб-додатки. Завдяки вбудованим інструментам, таким як ORM (Object-Relational Mapping), розробникам не потрібно вручну писати запити SQL.

2. Архітектура MVC/MVT. Django використовує архітектуру MVT (Model-View-Template), яка є дуже схожою на MVC. Це дозволяє чітко розділяти логіку застосунку, бізнес-логіку та презентацію.

3. Вбудована адмін-панель. Django має автоматично генеровану адмін-панель, яка дозволяє управляти базою даних без необхідності створення власного інтерфейсу.

4. Масштабованість. Django легко справляється з великими проектами, завдяки чому підходить як для малих стартапів, так і для великих корпорацій.

5. Безпека. Django пропонує вбудовані засоби для захисту від поширених загроз, таких як SQL-ін'єкції, XSS, CSRF та інші.

6. Розширюваність. Django має величезну кількість сторонніх пакетів (Django Packages), які дозволяють швидко додати нову функціональність, наприклад, обробку платежів, створення REST API тощо.

У Django існує шаблонізатор Jinja, який є альтернативою стандартному шаблонізатору Django Template Language (DTL). Jinja – це потужний, швидкий і гнучкий шаблонізатор, що забезпечує більшу схожість із синтаксисом Python і пропонує розширені можливості для роботи зі складними веб-інтерфейсами.

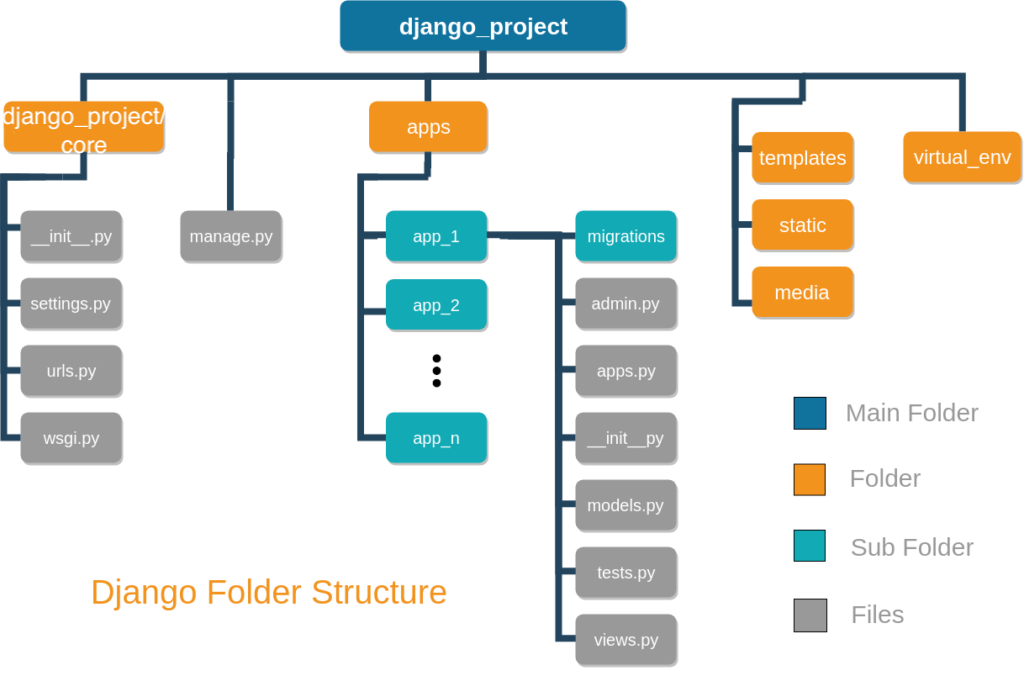


Рисунок 1 – Файлова структура проекту Django

Особливості Jinja:

1. Продуктивність. Jinja значно швидший за DTL завдяки оптимізації рендерингу шаблонів.

2. Схожість із Python. Синтаксис Jinja набагато ближчий до синтаксису. Python, що полегшує його використання для розробників, які вже знайомі з Python.

3. Гнучкість. Підтримка складних логічних конструкцій, кастомних фільтрів, макросів і включень (includes), що дозволяє створювати багаторазові та модульні шаблони.

4. Розширення. Можливість створювати власні фільтри, теги та інші розширення для додаткової функціональності.

3.1 Розробка веб-застосунку

Спочатку я створив віртуальне оточення. Це потрібно для правильної роботи статичних файлів і щоб ніякі зміни в комп’ютері не впливали на проект. Потім я створив проект. Нижче на зображенні відображено кінцеву структуру папок в проекті.

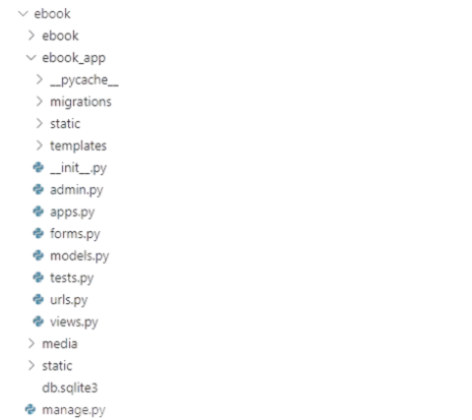


Рисунок 2 – Кінцева файлова структура проекту

Після всіх цих дій я можу розпочати саму розробку додатку. Перше, що потрібно зробити – це створити моделі. Моделі в Django – це основа для роботи з базою даних у проекті. Вони представляють структуру даних у вигляді Python-класів і дозволяють працювати з таблицями бази даних, не пишучи SQL-запити вручну. Django використовує ORM (Object-Relational Mapping), щоб автоматично перетворювати Python-об'єкти в записи бази даних і навпаки. Так як додаток у мене в першу чергу стосується книг, я створив модель книги.

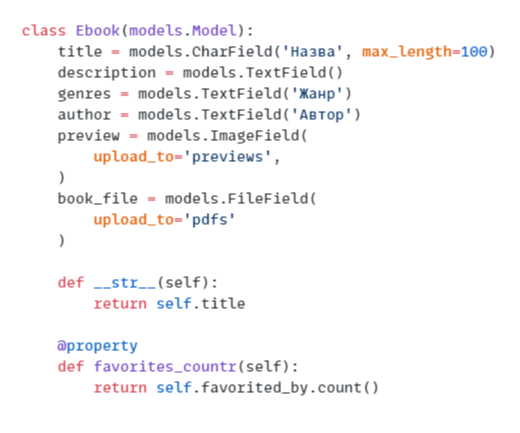


Рисунок 3 – Модель об’єкту книги

Після створення класу потрібно опреділитися з полями, які будуть характеристиками книги. Поле “title” відповідає за заголовок книги, поле «description” відповідає за опис книги, поле «genres” відповідає за жанри книг, поле «author” відповідає за авторів книги. Ці поля використовують тип “TextField”, який відповідає за довгий текст без обмежень довжини. Поле «preview” відповідає за зображення книги та використовує тип “ImageField”, який відповідає за файл зображення і завантажується в папку “previews”. Поле «book\_file” відповідає за файл книги, який можна завантажити з сайту і використовуює тип “FileField”.