## Технічне завдання на створення застосунку “**Personal Assistant**” (Web application)

**Основний функціонал Web application** **виконаний на Django**

## **Завдання:**

Студентам пропонується виконати апдейт **“персонального помічника”**, реалізованого по завершенню курсу Python Core, для чого необхідно створити web-інтерфейс та розширити можливості основного додатка.

## **Основні вимоги до проєкту** “**Personal Assistant**”**:**

1. Зберігати контакти з іменами, адресами, номерами телефонів, email та днями народження до книги контактів;
2. Виводити список контактів, у яких день народження через задану кількість днів від поточної дати;
3. Перевіряти правильність введеного номера телефону та email під час створення або редагування запису та повідомляти користувача у разі некоректного введення;
4. Здійснювати пошук контактів серед контактів книги;
5. Редагувати та видаляти записи з книги контактів;
6. Зберігати нотатки з текстовою інформацією;
7. Проводити пошук за нотатками;
8. Редагувати та видаляти нотатки;
9. Додавати в нотатки "теги", ключові слова, що описують тему та предмет запису;
10. Здійснювати пошук та сортування нотаток за ключовими словами (тегами).
11. Виконувати завантаження файлів користувача на хмарний сервіс та мати доступ до них. Користувач повинен мати можливість через web-інтерфейс завантажити на сервер будь-який файл та завантажити його.
12. Сортувати файли користувача за категоріями (зображення, документи, відео та ін.) і відображати тільки обрану категорію (фільтр файлів за категорією).
13. Надавати коротке зведення новин за день. Для цього ви повинні вибрати будь-яку цікаву вам область (фінанси, спорт, політика, погода) та кілька інформаційних ресурсів на задану тематику. З вибраних ресурсів збирати на запит користувача інформацію (заголовки новин, курси валют, результати спортивних подій тощо) і відображати на сторінці результатів. Що саме збирати та як можете визначити самостійно.

**Вимоги до Аутентифікації**

1. Реалізуйте механізм авторизації користувача для “**Personal Assistant**”. Web-інтерфейс повинен давати доступ до своїх функцій лише зареєстрованим користувачам.
2. Кожен зареєстрований користувач повинен мати доступ лише до своїх даних та файлів.
3. Реалізуйте механізми відновлення пароля для користувача за email

**Критерії прийому:**

1. Web-інтерфейс може бути реалізований на фреймворку Django.
2. Проєкт має бути збережений в окремому репозиторії та бути загальнодоступним (GitHub, GitLab або BitBucket).
3. Проєкт містить докладну інструкцію щодо встановлення та використання.
4. “**Personal Assistant**”зберігає інформацію в базі даних і може бути перезапущений без втрати даних.
5. Для надійності та підвищення продуктивності всю інформацію зберігати у базі даних PostgreSQL.
6. Всі критичні дані до доступу до бази даних та налаштування програми зберігаються в змінних середовищах і не завантажуються в репозитарій.
7. Проєкт повністю реалізує всі пункти вимог, описані в завданні.

**P.S.**: Ви можете розширити функціонал проєкту на свій розсуд, обов'язково проконсультувавшись з ментором перед цим. Розглядайте цей проєкт, як частину вашого портфоліо і корисний вам інструмент. З цієї причини ініціатива у розширенні та доповненні вимог до проєкту вітається. Наприклад, ви можете додати файл Dockerfile, щоб програма могла бути розміщена в контейнері Docker та образ завантажений на [dockerhub](https://hub.docker.com/).