**ТЕХНОЛОГИЧНО УЧИЛИЩЕ ЕЛЕКТРОННИ СИСТЕМИ**

**към ТЕХНИЧЕСКИ УНИВЕРСИТЕТ - СОФИЯ**

**ДОКУМЕНТАЦИЯ**

Тема : Блог за пътувания

Изготвил: Научен Ръководител:

Никола Петров Любомир Стоянов

**Съдържание**

[**Въведение 3**](#_i2e5s8g6ywqy)

[**Файлове 3**](#_6x9qasz1rsc9)

[main.py 3](#_5qshk6vm7nrt)

[home.html 4](#_w2vu9n96b01j)

[more\_info.html 4](#_hwp7xtdr8umc)

[about.html 5](#_e42954fh1b34)

[contact.html 5](#_918f364hcprt)

[new\_journey.html 5](#_gvj54p6kcnst)

[loading\_screen.html 6](#_wyput32ejfcf)

[app.yaml 6](#_q0dx8lkllapq)

[requirements.txt 6](#_qvllgxcqmtqg)

[**Важни линкове 6**](#_76k0pn4cayax)

## 

# Въведение

Направен е уебсайт за споделяне на пътувания из света. Frontend е написан на HTML, което означава, че потребителят ще може да взаимодейства с уебсайта чрез визуални елементи като бутони, формуляри и др. Backend на уебсайта е написан на Python, който е отговорен за обработка на заявките на потребителите и изпращането им към API-то на Flask. Flask е използван за свързване на Frontend-a с Backend-a и рендиране на HTML страниците.

За да бъде достъпен на потребителите, уебсайтът е deploy-нат в Google Cloud, който осигурява висока отзакоустойчивост и скалируемост на услугата. Това означава, че уебсайтът ще бъде наличен почти по всяко време и може да се справи с голям брой заявки от множество потребители едновременно.

Информацията за пътуванията се пази на две места - снимките в cloud storage (object storage), а останалата информация в firebase firestore.

Целта на уебсайта е да предостави на потребителите други гледни точки на света и за всички места на които сме били, както и да ни даде вдъхновение за бъдеща локация, която бихме посетили.

# Файлове

## main.py

Уеб апликация, изградена с Flask, която се свързва с услуги като Firebase Firestore и Google Cloud Storage.

При стартиране на приложението, то инициализира Flask, Firebase Firestore и Google Cloud Storage. По време на инициализацията се извличат начални данни от Firestore и съхраняват в променлива journeys.

Основната функционалност на приложението се дефинира чрез рутинг - определянето на обработваща функция (view) за всяка URL-адреса. В приложението са дефинирани следните рути:

* /: Зарежда шаблона "loading\_screen.html", който показва екран за зареждане.
* /fetch\_data: Извлича данни от Firestore и Cloud Storage.
* /home: Зарежда шаблона "home.html" и предава списъка journeys на шаблона за изобразяване на информацията за пътуванията.
* /journey/<journey\_id>: Зарежда шаблона "more\_info.html" и предава информация за конкретно пътуване и снимки свързани с него.
* /new\_journey: Обработва формата за добавяне на ново пътуване. Валидира и запазва данните в Firestore и качва снимките в Cloud Storage.
* /contact: Обработва формата за контакт, записва данните в Firestore.
* /about: Зарежда шаблона "about.html".

## home.html

HTML страница, която цели да покаже пътешествията. Страницата:

* Съдържа навигационно меню (<nav>), което съдържа връзки към различните раздели на блога - *home*, *new journey*, *about*, *contact*.
* Има фиксирано заглавие (<img>) в горната част на страницата, което съдържа лого на блога.
* Има контейнер (<div class="scrollmenu">), който показва картички с информация за различни пътешествия. Тези картички са генерирани чрез for цикъл в HTML шаблона и показват изображение на пътешествието, името на държавата и бутон за повече информация.

## more\_info.html

HTML страница, която показва допълнителна информация за пътуване. Тя включва следните елементи и функционалности:

* Заглавие и бутон за връщане към предишната страница.
* Данни за пътуването, като заглавие и държава, показани в центъра на страницата.
* Секция със скролиращ контейнер, в който се визуализират изображенията, свързани с пътуването.
* Описание на пътуването, показано в текстов формат.

Тази HTML страница използва шаблониране със стойности, които се подават от main.p, за да се покажат конкретни данни за пътуването и изображенията. Това означава, че при всяко зареждане на страницата, се очаква да бъдат подадени данните за пътуването и изображенията, които ще се покажат в съответните места на страницата.

## about.html

HTML страница с информация за автора и неговият проект за пътешествия. В HTML се задават стилове за фона, шрифтовете и разположението на елементите на страницата. В страницата има навигационно меню с връзки към различни раздели на уебсайта. С помощта на JavaScript се добавя ефект на промяна на навигационното меню при скролиране, като се променя и цветът на активния елемент в менюто. Описани са и някои лични информации за автора и неговата страст към пътешествията. Целта на страницата е да представи автора и неговите пътешествия на посетителите и да ги вдъхнови да изследват света и да се свържат с различни култури и хора.

## contact.html

HTML страница с форма за контакт. Страницата включва навигационно меню, което съдържа четири връзки: "Home", "New Journey", "About" и "Contact". При избор на връзката "Contact", има анимация, която подчертава тази връзка.

Основната част на страницата е контейнер с формата за контакт. Формата се състои от полета за въвеждане на име, имейл и съобщение, както и от бутон "Submit" за изпращане на формата. Когато потребителят изпрати формата, тя ще бъде изпратена към "/contact" с HTTP метод POST.

Целта на тази страница е да предостави начин на потребителите да се свържат с нас чрез изпращане на съобщение през формата за контакт.

## new\_journey.html

HTML страница с формуляр, която позволява на потребителите да въведат информация свързана с техните пътувания. На страницата се показва навигационно меню с четири връзки (Home, New Journey, About, Contact), които са представени като текстови линии. При навлизане на мишката върху всяка от връзките се визуализира анимация в зададен цвят, която подчертава избраната връзка.

Под навигационното меню се намира формулярът, който съдържа полета за въвеждане на заглавие, държава, описание и изображения свързани с пътуването. Полетата са стилизирани с CSS, за да имат определена визуална външност - широчина, височина, цвят на фона, граница и други. Има също така и бутон за изпращане на формуляра.

Целта на този код е да предостави удобен начин за въвеждане на информация за пътувания и да осигури приятен и интуитивен интерфейс за потребителите, които използват страницата.

## loading\_screen.html

HTML страница със зареждащ екран (loading screen), който се появява докато се изпълнява AJAX заявка за извличане на данни от сървъра.

Когато потребителят достъпи страницата, тя показва зареждащ екран с логото на SVG, което се върти в анимация, и съобщение "Loading". В същото време, се изпраща AJAX заявка към /fetch\_data на сървъра. Когато заявката завърши успешно (със статус 200) и получените данни са верни (response === true), потребителят се пренасочва към страницата /home. В случай на грешка или неочакван отговор, се изписва грешка в конзолата на браузъра.

Това позволява на потребителя да види зареждащ екран докато се извличат необходимите данни от сървъра и се подготвя страницата за преглед. Това може да бъде полезно за подобряване на потребителския интерфейс и усещането за отзивчивост на уеб приложението.

## app.yaml

Файлът играе ролята на конфигурационен файл, който определя какво ще се случва с приложението в средата на Google App Engine. Той предоставя гъвкавост и контрол върху скалируемостта и поведението на приложението в средата на App Engine.

## requirements.txt

Файл, които е нужен, за да може да се deploy-не приложението в Google Cloud. Той указва какви библиотеки са използвани.

# Важни линкове

GitHub - https://github.com/ItsRizee/vot\_project

Линк към сайта - https://ambient-stone-383715.lm.r.appspot.com/