

**Dokumentacija izvedenih lastnosti**

(ni imena?)

Študent:

Domen Hribernik

Denis Železnik

Merisa Mustajbašić

Datum in mesto:

23.08.2023, Maribor

Kazalo

[1 Dokumentacija izvedenih lastnosti 1](#_Toc143680577)

[1.1 Implement language for displaying runs (PrePJ) 1](#_Toc143680578)

[1.2 Get data from Strava API and scrape weather (PriPJ) 1](#_Toc143680579)

[1.3 Display data gotten from Strava and Weather (PriPJ) 1](#_Toc143680580)

[1.4 Spletna Storitev (SP) 1](#_Toc143680581)

[1.5 Spletni Vmesnik (SP) 2](#_Toc143680582)

[1.6 Project Initiation and Tool Integration (PP) 3](#_Toc143680583)

[1.7 Virtual Machine Setu pand Configuration (SA) 3](#_Toc143680584)

[1.8 Docerization and Containerization (SA) 3](#_Toc143680585)

[1.9 Documentation (PP) 4](#_Toc143680586)

# Dokumentacija izvedenih lastnosti

Večje funkcionalnosti smo razdelili po Epicih.

## Implement language for displaying runs (PrePJ)

* **Defina a construct for describing a run**

Namen: Določi konstrukt za opis teka.

Izvedba in uporaba: Definirali strukturo, ki bo omogočila opis tekov.

* **Write 10 test examples for language**

Namen: Preveri delovanje jezikove konstrukcije

Izvedba in uporaba: Pripravili 10 testnih primerov, ki bodo preizkusili različne vidike jezika.

* **Write a formal definition fort he language in BNF**

Namen: Opredi jezik v BNF obliko.

Izvedba in uporaba: Sestavili formano BNF definicijo, ki bo natančno opisala sintakso jezika.

* **Implement lexical analysis**

Namen: Analizirati vnos jezika in ga razdeliti na leksične enote.

Izvedba in uporaba: Ustvarili funkcijo za leksikalno analizo, ki bo preoblikovala jezik v leksične enote.

* **Implement syntactical analysis**

Namen: Preveriti gramatično strukturo vnesenega jezika.

Izvedba in uporaba: Ustvarili funkcijo za sintaktično analizo, ki bo preverila ustreznost gramatične strukture jezika.

* **Implement semantical analysis and build a geoJSON output**

Namen: Obdelati vneseni jezik in ustvariti izhodne podatke v geoJSON obliki.

Izvedba in uporaba: Izvedli semantično analizo in oblikovali izhodne podatke v obliki geoJSON.

## Get data from Strava API and scrape weather (PriPJ)

* **Look into Strava API**

Namen: Razumevanje StravaAPI-ja.

Izvedba in uporaba: Vpogled v StravaAPI za razumevanje dokumentacije.

* **Make the Strava API**

Namen: Pridobitev podatkov iz StravaAPI-ja.

Izvedba in uporaba: Pošiljanje Strava identitete za pridobitev podatkov o tekih.

* **Weather API with scrape{it}**

Namen: Pridobitev podatkov za vreme.

Izvedba in uporaba: Scrapanje podatkov iz spletne strani za vreme v Sloveniji.

## Display data gotten from Strava and Weather (PriPJ)

* **Store Activity data from strava**

Namen: Shranba tekov uporabnika.

Izvedba in uporaba: Oblikovanje podatkov priboljenih iz StravaAPI-ja in shranjevanje teh v bazi.

## Spletna Storitev (SP)

* **Set up** **database**

Namen: Vzpostavitev podatkovne baze na Mongodb Atlas.

Izvedba in uporaba: Registracija na Mongodb, vzpostavitev povezave s MongodbCompass.

* **Make the API**

Namen: Zasnova in Osnovna implementacija API.

Izvedba in uporaba: Narejena registracija, prijava in vzpostavljena povezavo med bazo in API.

* **Implement JWTokens**

Namen: Dodatna zaščita z authentikacijo .

Izvedba in uporaba: Preverjanje identitete pri zahtevkih z pomembnimi informacijami.

* **Implement Web Sockets**

Namen: Real-time komunikacija med uporabnikom in strežnikom.

Izvedba in uporaba: Pridobitev lokacije uporabnika, in izračun razdalje med ostalimi teki.

* **Implement Geospatial Queries**

Namen: Iskanje tekov v določeni okolici.

Izvedba in uporaba: Z uporabo web socketov, povpraševanje po tekih v določeni okolici.

## Spletni Vmesnik (SP)

* **User Login/Register**

Namen: Omogočiti uporabnikom registracijo in prijavo v sistem

Izvedba in uporaba: Razvili funkcionalnost za registracijo in prijavo uporabnikov, ki bo omogočila dostop do sistema.

* **Create figma design for website**

Namen: Oblikovati vizualni načrt spletnega mesta.

Izvedba in uporaba: Uporabili orodje Figma za izdelavo grafičnega načrta celotne spletne strani.

* **Create a view for the run page**

Namen: Izdelati uporabniški vmesnik za prikaz podatkov o teku.

Izvedba in uporaba: Razvili grafični vmesnik, ki bo prikazoval podatke o posameznem teku na zemljevidu za analizo teka.

* **Make the page to display the runs**

Namen: Omogočiti uporabnikom pregled njihovih preteklih tekov.

Izvedba in uporaba: Razvili stran, ki bo prikazovala seznam in podrobnosti o tekih, ki jih je uporabnik opravil.

* **Make the home page**

Namen: Izdelati domačo stran spletnega mesta.

Izvedba in uporaba: Oblikovali domačo stran, ki uporabniku predlaga trase v njegovi okolici, ki jih lahko uporabi za tek.

* **Make admin tools**

Namen: Omogočiti administratorjem učinkovito upravljanje sistema.

Izvedba in uporaba: Razvili orodja in funkcionalnosti, ki bodo administratorjem omogočala nadzor nad uporabniki in sistemom.

* **Create graphs to show run data**

Namen: Vizualno prikazati podatke o tekih z grafi.

Izvedba in uporaba: Implementirali grafične prikaze, ki bodo uporabnikom prikazovali statistične podatke o njihovih tekih.

* **Create responsive design for mobile**

Namen: Zagotoviti dobro uporabniško izkušnjo na mobilnih napravah.

Izvedba in uporaba: Oblikovali odzivno oblikovanje, ki se bo prilagajalo različnim velikostim zaslonov in omogočalo enostavno uporabo na mobilnih napravah.

## Project Initiation and Tool Integration (PP)

* **Setup Azure for the team**

Namen: Konfiguracija okolja v storitvi Azure za potrebe ekipe.

Izvedba in uporaba: Izvajanje postopka nastavitve okolja v storitvi Azure, ki bo omogočilo enostavno delo ekipe.

* **Integrate GitHub repositories with Jira**

Namen: Povezava repozitorijev iz GitHuba s sistemom Jira.

Izvedba in uporaba: Povezovanje GitHub repozitorijev s sistemom Jira za boljše sledenje razvoju in upravljanje projektov.

* **Upload all reppositories currently not on Jira**

Namen: Dodajanje vseh repozitorijev, ki še niso vključeni v sistem Jira.

Izvedba in uporaba: Prenos in vključitev obstoječih repozitorijev v sistem Jira, da bo celoten razvoj in sledenje centralizirano upravljano.

## Virtual Machine Setup and Configuration (SA)

Namen: Ustvariti in konfigurirati virtualni stroj z operacijskim sistemom Linux.

Izvedba: Ustvarite Linux virtualni stroj v Azure okolju, izbirate operacijski sistem, vire (CPU, RAM, shranjevanje), in izvedete osnovno namestitev.

Uporaba: Uporabite virtualni stroj za razvoj, testiranje in izvajanje aplikacij v Linux okolju brez potrebe po fizični strojni opremi, kar omogoča agilnost in enostavnost pri razvoju programske opreme. Zagotovite varnost z ustrezno konfiguracijo in posodabljanjem.

## Docerization and Containerization (SA)

* **Deploy project to Docer**

Namen: Uvajanje projekta v Docker za učinkovitejše upravljanje in izvajanje.

Izvedba in uporaba: Projekt bo vzet v Docker okolje, ustvarjena bo Docker slika aplikacije za prenosljivost med okolji. To omogoča enostavnejšo namestitev, posodabljanje in izvajanje v različnih okoljih.

* **Setup Docer on VB**

Namen: Konfiguracija Docker na virtualnem strežniku (VB) za razvoj in testiranje.

Izvedba in uporaba: Docker bo nameščen in konfiguriran na virtualnem strežniku, kar omogoča razvoj in testiranje aplikacij v izoliranem okolju. To izboljšuje produktivnost in preprečuje konflikte med komponentami.

* **Setup Docker Hub registry**

Namen: Ustvariti Docker Hub registry za shranjevanje in deljenje Docker slik.

Izvedba in uporaba: Docker Hub registry bo nastavljen kot centralno mesto za shranjevanje Docker slik, kar olajša deljenje med člani ekipe in uporabo slik v različnih okoljih.

* **GitHub Actions workflow**

Namen: Ustvariti avtomatiziran potek dela z uporabo GitHub actions.

Izvedba in uporaba: Nastavitev GitHub actions za samodejno izvajanje testov, gradnjo Docker slik in implementacijo v Docker registry ob vsakem pushu v glavno vejo. To izboljša stabilnost in učinkovitost razvojnega procesa.

* **Setting up Webhook**

Namen: Povezati GitHub s Docker Hubom za samodejno posodabljanje Docker slik.

Izvedba in uporaba: Ustvarite sprožilec dogodkov (Webhook) med GitHubom in Docker Hubom, kar omogoča samodejno posodabljanje slik v Docker registru ob spremembi kode ali pushu.

* **Setup Docker for backend & frontend**

Namen: Konfiguracija Dockerja za vsebovanje backend in frontend komponent.

Izvedba in uporaba: Docker bo uporabljen za ustvarjanje ločenih vsebnikov za backend in frontend komponente aplikacije, kar omogoča boljšo modularnost, vzdrževanje in skaliranje.

## Documentation (PP)

Namen: Ustvariti dokumentacijo za projektne naloge.

Izvedba in uporaba: Ustvarili dokumente za prikaz dela in funkcionalnosti pri projektnih nalogah.