

Digitaliziran študijski urnik

Specifikacija zahtev za programsko opremo

POVZETEK

Ta dokument specifikacije programske opreme opisuje razvoj mobilne aplikacije "Digitaliziran študijski urnik", namenjene študentom. Aplikacija bo avtomatsko uvozila študijske obveznosti iz sistema Wise Timetable, jih združila z osebnimi aktivnostmi, prepoznala časovne konflikte in o njih obvestila uporabnika, s čimer bo poenostavila upravljanje časa.

Filip Rap, Miha Rot
Verzija 1.0

Specifikacija zahtev za programsko opremo

Dokument pripravila:

Filip Rap,

Miha Rot

Zaščita dokumenta:

© FERI, 2025

Vse pravice pridržane. Reprodukcija po delih ali v celoti na kakršenkoli način in v kateremkoli mediju ni dovoljena brez pisnega dovoljenja avtorja. Kršitve se sankcioniраjo v skladu z avtorsko pravno in kazensko zakonodajo

Kazalo vsebine

Digitaliziran študijski urnik	0
Specifikacija zahtev za programsko opremo.....	0
1. Uvod.....	2
1.1. Namen dokumenta.....	2
1.2. Obseg storitve.....	2
1.3. Definicije, akronimi in kartice	2
2. Splošni opis zahtev.....	4
2.1. Perspektiva produkta	4
2.2. Funkcije produkta	4
2.2.1. Nefunkcionalne zahteve.....	4
2.3. Značilnost uporabnikov	5
2.4. Omejitve	5
2.5. Predpostavke in odvisnosti.....	5
2.6. Prioritete zahtev	5
3. Specifične zahteve	7
3.1. Zahteve za zunanje vmesnike	7
3.1.1. Uporabniški vmesniki	7
3.1.2. Vmesniki programske opreme	7
3.2. Zahteve delovanja programske opreme.....	7
3.3. Logične zahteve za podatkovni model	7
3.4. Omejitve pri oblikovanju	8
3.5. Značilnosti okolja	8
Priloge	9

1. Uvod

1.1. Namen dokumenta

Namen tega dokumenta je podrobno opisati funkcionalne in nefunkcionalne zahteve za razvoj mobilne aplikacije "Digitaliziran študijski urnik". Dokument služi kot podlaga za razvojno ekipo, testersko ekipo in naročnika ter zagotavlja skupno razumevanje obsega, ciljev in omejitev projekta.

1.2. Obseg storitve

Naročnik želi razviti mobilno aplikacijo za pametne telefone in tablice, namenjeno sprva študentom FERI smer IPT, kasneje pa nadgrajevanju študentom Univerze v Mariboru in kasneje vsem udeležencem izobraževanj, ker se uporablja urnik preko sistema Wise Timetable. Aplikacija bo avtomatizirala upravljanje osebnega urnika z integracijo študijskih obveznosti.

Aplikacija – produkt zagotavlja:

- Samodejen uvoz in sinhronizacija študijskega urnika iz sistema Wise Timetable.
- Ustvarjanje in upravljanje osebnega urnika z možnostjo dodajanja lastnih aktivnosti.
- Samodejno prepoznavanje in opozarjanje na časovne konflikte med aktivnostmi.
- Prioritizacija študijskih obveznosti pred osebnimi.
- Pošiljanje push-obvestil o prihajajočih obveznostih in konfliktih.
- Možnost ročnega urejanja in prilagajanja urnika.
- Dostopnost in osebne nastavitev (npr. temni način).

Aplikacija – produkt ne zagotavlja:

- Neposrednega urejanja urnika v sistemu Wise Timetable.
- Funkcije klepetanja ali družabnega omrežja med študenti.
- Vpisa v izpite ali upravljanja z ocenami.
- Integracije z drugimi koledarji (npr. Google Calendar, Apple Calendar) v začetni fazi.

Namenski cilj produkta je študentom ponuditi centralizirano, pregledno in inteligentno platformo za vodenje njihovega časa, kar zmanjša administrativno breme in preprečuje časovne konflikte, ter s tem posredno izboljša učno izkušnjo.

1.3. Definicije, akronimi in kartice

Specifikacija zahtev za programsko opremo

- **FERI:** Fakulteta za elektrotehniko, računalništvo in informatiko
- **UM:** Univerza v Mariboru
- **Wise Timetable:** obstoječi sistem za prikaz urnikov na UM
- **Push-obvestilo:** obvestilo, ki se prikaže uporabniku neposredno na mobilni napravi
- **Scraping:** avtomatizirana metoda pridobivanja podatkov iz spletnih strani
- **UI/UX:** Uporabniški vmesnik / Uporabniška izkušnja (User Interface / User Experience)

2. Splošni opis zahtev

2.1. Perspektiva produkta

Aplikacija "Digitaliziran študijski urnik" je samostojna mobilna aplikacija, ki tvori enosmerno povezavo z obstoječim sistemom Wise Timetable za potrebe uvoza podatkov. Povezava bo vzpostavljena na način, ki spoštuje pravice dostopa in infrastrukturo univerze. Aplikacija ne vpliva neposredno na druge sisteme izobraževalne ustanove.

2.2. Funkcije produkta

Produkt

omogoča:

Študentom (končnim uporabnikom):

1. Prijava v aplikacijo.
2. Uvoz in prikaz osebnega študijskega urnika iz sistema Wise Timetable.
3. Ročno dodajanje, urejanje in brisanje osebnih aktivnosti (delo, šport, prosti čas) v osebni koledar.
4. Vizualni pregled združenega urnika (študijsko + osebno) v tedenskem in dnevнем pogledu.
5. Prejemanje push ali e-mail obvestil o:
 - Prihajajočih predavanjih, vajah in izpitih.
 - Zaznanih časovnih konfliktih.
6. Pregled in razreševanje konfliktov, ki jih sistem predstavi skupaj s predlogi za premik osebnih aktivnosti.
7. Prilagajanje nastavitev aplikacije (npr. temni način, nastavitev obveščanja).

Administratorjem:

1. Nadzor nad delovanjem aplikacije in povezavo z viri podatkov.
2. Oglaševanje splošnih obvestil vsem uporabnikom aplikacije.

2.2.1. Nefunkcionalne zahteve

Produkt naj omogoča:

- **Zmogljivost:** Aplikacija mora zagotavljati odzivni čas pri vseh operacijah (npr. nalaganje urnika, iskanje konfliktov) pod 3 sekunde.
- **Zanesljivost:** Aplikacija mora biti na voljo 99% časa. Uvoz urnika iz zunanjega vira mora biti uspešen v vsaj 95% poskusov.
- **Uporabnost:** Uporabniški vmesnik mora biti intuitiven in se mora držati smernic za Material Design (Android) / Human Interface Guidelines (iOS). Nov uporabnik mora biti sposoben osnovno uporabo aplikacije obvladati v manj kot 10 minutah.
- **Varnost:** Dostop do aplikacije je zaščiten s prijavo prek prijavnega obrazca. Vse osebne podatke uporabnika je treba varno shranjevati v bazi podatkov.

- **Kompatibilnost:** Aplikacija mora delovati na operacijskih sistemih Android (različica 10 in novejše) in iOS (različica 14 in novejše). Razvita bo s frameworkom React Native v Expo.

2.3. Značilnost uporabnikov

Uporabniki aplikacije imajo naslednje značilnosti:

1. Študent: Glavni uporabnik. Prisoten na vseh funkcijah aplikacije.
2. Administrator: Skrbi za tehnično stran aplikacije in globalna obveščanja. Je predstavnik razvijalca/naročnika.

2.4. Omejitve

Aplikacija ima naslednje omejitve:

- Aplikacija je odvisna od razpoložljivosti in strukture podatkov v sistemu Wise Timetable UM. Kakršne koli spremembe na strani Wise Timetable lahko zahtevajo prilagoditev v aplikaciji.
- Aplikacija deluje v slovenskem jeziku.
- Funkcionalnost aplikacije je omejena na podatke, ki jih lahko pridobi iz Wise Timetable in jih vnese sam uporabnik.

2.5. Predpostavke in odvisnosti

Za delovanje aplikacije se predpostavlja naslednji pogoji:

- Predpostavlja se nemoteno delovanje in dostopnost strežnikov Wise Timetable.
- Predpostavlja se, da ima uporabnik stabilno internetno povezavo za uvoz urnika in prejemanje obvestil.
- Razvoj je odvisen od uporabe React Native in Expo okolja.

2.6. Prioritete zahtev

Pri implementaciji zahtev se upoštevajo naslednje prioritete:

- **Visoka prioriteta (MVP - Minimal Viable Product):** Uvoz urnika, osebni koledar, prepoznavanje konfliktov, osnovna push-obvestila.
- **Srednja prioriteta (nadgradnje):** Naprednejši predlogi za razreševanje konfliktov, tedenski povzetki, podrobnejše nastavitev obveščanja.

Specifikacija zahtev za programsko opremo

- **Nizka prioriteta (prihodnje verzije):** Integracija z zunanjimi koledarji (Google/Apple), večjezična podpora (poleg SL/EN), podrobna statistika o porabi časa.

3. Specifične zahteve

3.1. Zahteve za zunanje vmesnike

3.1.1. Uporabniški vmesniki

Glavni vmesnik je mobilna aplikacija za naprave Android in iOS. Vmesnik mora biti čist, modernejšega videza in enostaven za uporabo z enim prstom. Uporabljati mora standardne UI komponente platforme (v okviru React Native/Expo).

3.1.2. Vmesniki programske opreme

Aplikacija bo komunicirala z naslednjima vmesnikoma:

- Wise Timetable: Aplikacija bo komunicirala s tem sistemom za pridobivanje podatkov o urniku. Komunicirala bo tako, da bo prenašala iCal izpise. Aplikacija bo na določen interval preverila stanje zadnje spremembe urnika na Wise Timetable in ko se bo ta sprememba zgodila bo izvedla prenos in obdelavo iCal izpisa.
- Strežnik za push-obvestila: Aplikacija bo uporabljala Expo Notification Servis ali podobno storitev za pošiljanje push-obvestil uporabnikom.
-

3.2. Zahteve delovanja programske opreme

Pri delovanju programske opreme imamo naslednje zahteve:

- Aplikacija mora brez težav podpreti do 5000 sočasno aktivnih uporabnikov.
- Operacije (nalaganje pogleda, shranjevanje aktivnosti) se morajo izvesti v manj kot 3 sekunde.
- Proses uvoza in analize urnika za konflikte ne sme biti viden uporabniku (oz. mora biti opravljen v ozadju).

3.3. Logične zahteve za podatkovni model

Glavni entiteti v aplikaciji bodo:

- **Users** { id, username, email , password }
- **Events** { id, user_id, title, type, start_time, end_time, location, description, source }

3.4. Omejitve pri oblikovanju

Koda aplikacije mora biti pisana tako, da omogoča enostavno vzdrževanje in nadgradnjo.

3.5. Značilnosti okolja

Aplikacije bo nastajala sli delovala v naslednjem okolju.

- Razvojno okolje: Visual Studio Code, Expo CLI, React Native.
- Nadzorna shema verzij: Git (GitHub).
- Backend strežnik (če bo potreben): Node.js s podatkovno bazo (npr. PostgreSQL, Firebase Firestore).

Priloge