

# Sitpass

Potrebno je implementirati aplikaciju po uzoru na [Fitpass](#) za podršku sporta i rekreacije.

Korisnik aplikacije ima na raspolaganju sledeće funkcionalnosti:

[K1] Zahtev za registraciju korisnika. Pristigle zahteve za registraciju obrađuje administrator sistema (zahtev A1). Administrator sistema je predefinisani korisnik u sistemu.

[K2] Prijava i odjava sa sistema. Kada se korisnik uspešno prijavi na aplikaciju omogućiti korisniku da se odjavi. Bez prijave na sistem nije moguće pristupiti ostatku aplikacije.

[K3] Rukovanje<sup>1</sup> objektima. Obavezna polja objekta su grad, adresa, opis objekta, discipline i radno vreme. Jedino administrator sistema može da rukuje objektom, dok menadžer objekta može da ažurira atribut objekta, poput disciplina, radnog vremena i utisaka. Objekat može, ali ne mora da sadrži slike. Ukoliko ih sadrži, mora da sadrži više njih. Pored naziva objekta prikazati i ukupnu, srednju vrednost ocene objekta.

[K4] Rezervisanje odlaska (engl. *exercise*) u objekat. Prilikom rezervacije, korisnik navodi datum odlaska i vremenski period termina. Nije moguće rezervisati termin van radnog vremena objekta.

[K5] Ostavljanje utiska (engl. *review*) na objekat. Prilikom pisanja utiska formira se ocena objekta i opcionalno ostavlja komentar. Na utisku je moguće oceniti objekat prema: 1) opremljenosti; 2) osoblju; 3) higijeni; 4) prostori; na skali od 1-10, gde svaka od navedenih stavki ima svoju ocenu. Korisnici mogu da ostave utisak samo ukoliko imaju barem jedan odlazak u datom objektu u trenutku pisanja utiska. Na recenziji prikazati informaciju na koliko odlazaka je korisnik bio u trenutku pisanja recenzije.

[K6] Pretraga i filtriranje objekata. Moguće je pretražiti sve objekte po gradu, disciplini, opsegu ocena ili radnom vremenu. Moguće je kombinovati pretragu i filtriranje prema više gradova, disciplina, opsega ocena i radnom vremenu.

[K7] Sortiranje utisaka. Na stranici objekta utiske je moguće sortirati po ocenama i datumu objavljivanja rastuće i opadajuće.

[K8] Pregled početne stranice. Korisnici na početnoj stranici vide objekte iz istog grada kao i registrovani korisnik. Korisnici takođe vide: 1) "najpopularnije objekte", tj objekte sa najviše pozitivnih ocena; 2) objekte u koje je korisnik već odlazio; 3) "isprobaj novo" objekte, tj 5 objekata na kojima korisnik sistema nema ranije rezervisan odlazak i čija disciplina nije disciplina na kojoj je korisnik ranije bio. Korisniku dati opciju da istraži još novih objekata klikom na dugme "explore more".

[K9] Promena lozinke. Pri promeni lozinke prvo se unosi trenutna lozinka i dva puta se unosi nova lozinka. Nakon promene lozinke korisniku se šalje mejl.

[K10] Promena dodatnih podataka na profilu kao i sliku. Na profilu se vidi spisak svih utisaka, objekata na kojima je korisnik menadžer i istoriju odlazaka u objekte za datog korisnika.

---

<sup>1</sup> Pod rukovanjem se podrazumevaju aktivnosti vezane za dodavanje, promenu i uklanjanje odgovarajućih pojava entiteta. Većinu informacionih sistema karakteriše neograničen period čuvanja podataka te se aktivnost uklanjanja odgovarajućih pojava entiteta retko koristi.

Menadžer objekta ima na raspolaganju sve funkcionalnosti koje ima i registrovan korisnik, kao i:

[M1] Ažuriranje disciplina, radnog vremena i objavljenih slika na objektu.

[M2] Rukovanje utiscima. Menadžer ima uvid u sve utiske koji su ostavljeni na objektu od strane korisnika i može da sakrije ili ukloni pojedinačne utiske. Uklonjeni utisci se logički brišu i ocena sa utiska poništava i ne računa se u ukupnoj oceni objekta. Ocene ostavljene na sakrivenom utisku se i dalje računaju u ukupnoj oceni objekta.

[M3] Ostavljanje komentara na utisku. Moguće je odgovoriti (eng. *reply*) na utisak. Ukoliko je menadžer odgovorio na utisak, običan korisnik može odgovoriti na komentar menadžera. Moguć je proizvoljan broj odgovora na komentar (*reply na reply na reply...*).

[M4] Analitika (engl. *analytics*) objekata. Za svaki objekat na kojem je korisnik menadžer, moguće je videti broj korisnika i broj recenzija objekta na podrazumevano nedeljnom, mesečnom i godišnjem nivou. Moguće je videti periode u danu u kojima je bilo najviše ili najmanje korisnika objekta (najviše ili najmanje prometne periode) na dnevnom, nedeljnom i mesečnom nivou. Pored podrazumevanih nivoa, omogućiti pregled svake od analitika za nasumično definisan vremenski opseg (početni i krajnji datum). [Za predmet KVT]: Koristiti grafičku biblioteku za vizuelizaciju, poput [Chart.js](#), [Apache ECharts](#) ili sl.

Administrator sistema ima na raspolaganju sve funkcionalnosti koje ima menadžer objekta i registrovan korisnik, ali i:

[A1] nastavak na K1. Obrada zahteva za registraciju. Administrator ima pravo da prihvati ili odbije zahtev za registraciju. Korisnik može da se prijavi na sistem tek kada je zahtev za registraciju prihvaćen. Nakon što je zahtev obrađen, korisniku se šalje mejl.

[A2] nastavak na K3. Upravljanje objektima i menadžerima za objekte. Administrator može da dodeli korisniku sistema ulogu menadžera objekta. Moguće je dodati i ukloniti menadžere sa objekta. Menadžer koji je uklonjen sa objekta postaje običan korisnik. Ukoliko objekat nema nijednog menadžera, objekat postaje neaktivan i ne prikazuje se u listi dostupnih objekata. Za neaktivne objekte nije moguće rezervisati odlazak ili ostaviti utisak. Administrator ima uvid u sve neaktivne objekte i može im dodeliti menadžera, pri čemu objekat prelazi u aktivan status.

## UES DEO:

Omogućiti indeksiranje objekata u Elasticsearch-u, omogućiti čuvanje slika i dokumenata u MinIO bazi. Dodatno, implementirati funkcionalnost da se pored slika, za objekat može upload-ovati PDF dokument koji sadrži opis objekta u slobodnoj formi (*full-text*).

[S1] Pretraga objekata:

- Pretraživanje objekata po nazivu
- Pretraživanje objekata po opisu
- Pretraživanje objekata po opisu iz zakačenog PDF fajla (sadržaj PDF-a se parsira i indeksira u ES kao *Text* polje)
- Pretraživanje objekata po opsegu broja review-a (od - do), gde može biti zadata donja i/ili gornja granica opsega
- Pretraživanje objekata po prosečnoj oceni (po kategoriji) u opsegu (od - do), gde može biti zadata donja i/ili gornja granica opsega
- Kombinacija prethodnih parametara pretrage (BooleanQuery, omogućiti AND i OR operator između polja)

- Pretprocesirati upit, tako da bude nezavisan od velikog i malog slova, kao i ćirilskog ili latiničkog pisma (morate pisati svoju konfiguraciju za analyzer, nije dovoljno samo iskoristiti ugrađeni Serbian Analyzer)
- Obezbediti podršku u poljima forme za unos PhraseQuery (kada je vrednost polja unešena pod dvostrukim navodnicima), PrefixQuery (kada je vrednost polja unešena sa karakterom „\*“ npr. „Ivan M\*“) i FuzzyQuery (empirijski odredite distancu koja vam odgovara, pokreće se kada vrednost polja počinje sa karakterom „~“ npr. „~rizika“)
- Implementirati „more like this“ pretragu koja će uzeti u obzir polja: naslov, opis i opis iz PDF-a. Min/max term frequency podesiti empirijski.

Prilikom prikaza rezultata prikazati naziv objekta, opis objekta (onaj koji se zadaje iz UI-a, ne treba onaj iz PDF-a), omogućite sortiranje rezultata po nazivu (smislite kako da dopunite indeksnu strukturu kako biste podržali ovo) i dinamički sažetak (Highlighter). Iz prikaza svih objekata omogućiti preuzimanje PDF-a sa opisom objekta

Za implementaciju aplikacije iskoristiti sledeće softverske pakete:

- Za Serverske veb tehnologije:
  - Spring framework
  - Apache Tomcat (ne mora biti posebno integrisan, može Spring Boot)
  - MySQL ili PostgreSQL
- Za Klijentske veb tehnologije
  - Angular framework
- Za UES
  - ElasticSearch
  - MinIO

Podatke kojima upravlja aplikacija organizovati uz oslonac na SUBP<sup>2</sup>.

### Nefunkcionalni zahtevi

Podržati autentifikaciju korisnika upotrebom korisničkog mejla i lozinke i autorizaciju korisnika upotrebom mehanizma tokena.

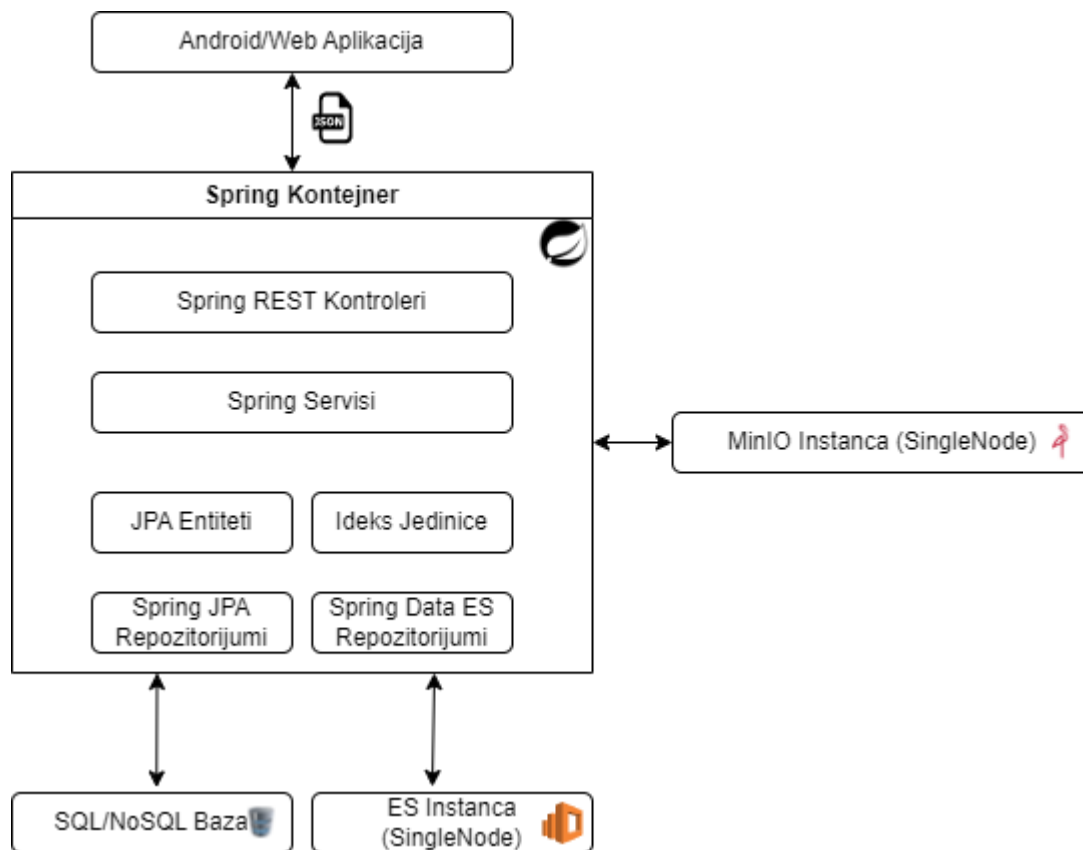
Beležiti poruke o važnim događajima koji su nastali prilikom izvršavanja aplikacije.

## Arhitektura aplikacije

Aplikacija je raspoređena na više uređaja (što se UES dela tiče, sve može biti na jednom uređaju): Veb brauzer ili android aplikacija, Springkontejner (u Tomcat serveru ili pokrenut pomoću Spring Boot), SUBP, Elasticsearch i MinIO. Dijagram rasporeda prikazan je na slici 1.

---

<sup>2</sup> Skraćeno od Sistem za Upravljanje Bazama Podataka.



**Slika 1:** Arhitektura aplikacije – dijagram rasporeda

Back-end aplikaciju implementirati upotrebom Spring framework-a [1], dozvoljeno je koristiti i Spring Boot [2]. Front-end aplikacija mora postojati i komunicira sa back-end aplikacijom putem ekspozovanih RESTful servisa. Dozvoljeno je front-end aplikaciju implementirati kao Android aplikaciju ili kao veb aplikaciju upotrebom HTML, CSS, JavaScript, JQuery, AngularJS, ili React tehnologije. Kao SUBP koristiti MySQL Server [3] ili neki drugi relacioni SUBP. Za beleženje poruka koristiti log4j API [4]. Za izgradnju softvera koristiti Maven [5] ili neki drugi alat, a dozvoljeno je i koristiti Spring Boot i na taj način konfigurisati i pokretati aplikaciju.

Za formiranje ocene potrebno je demonstrirati funkcionalnosti kroz korisnički interfejs na oba predmeta.

Veb aplikacija je Angular aplikacija i ocenjuje se na *Klijentskim veb tehnologijama* (KVT). Ukoliko postoje studenti koji su položili *Softverske veb tehnologije* (SVT), a nisu položili KVT- *backend* deo rešenja može biti realizovan kroz *Firebase* [6] ili neki sličan servis.

## Model podataka

Na [linku](#) prikazan je model podataka aplikacije. Neregistrovan korisnik (entitet AccountRequest) ima jedino mogućnost slanja zahteva za registraciju na aplikaciji. Entitet User predstavlja registrovanog korisnika aplikacije i namenjen je skladištenju podataka koji se koriste za autentifikaciju i autorizaciju. Korisnik takođe može biti administrator sistema (entitet Administrator) ili menadžer objekta (entitet Manages). Objekat je opisan entitetom Facility, dok su povezane discipline sa objektom i radno vreme objekta opisani entitetima Discipline i WorkDay. Rezervacija odlaska u objekat je opisana entitetom Exercise. Entitet Review predstavlja utisak od strane korisnika koji ima barem jedan odlazak u objekat. Pri utisku, korisnik mora da navede ocene vezane za objekat (entitet Rate), ali može ostaviti i komentar (entitet Comment).

## Napomene

1. Studenti ne smeju svojevolsjno da menjaju model baze podataka.
2. U delu Opis funkcionalnosti zahtevi koji su obojeni **žutom** bojom označavaju zahteve za ocenu 6, zahtevi obojeni **crvenom** bojom označavaju zahteve za ocenu 7, zahtevi obojeni **zelenom** bojom označavaju zahteve za ocenu 8, zahtevi obojeni **roze** bojom označavaju zahteve za ocenu 9, a zahtevi obojeni **plavom** bojom označavaju zahteve za ocenu 10.
3. Projekat se radi samostalno.

## Reference

1. Spring Framework, <https://spring.io/>
2. Spring Boot, <https://projects.spring.io/spring-boot/>
3. MySQL, <https://www.mysql.com/>
4. Log4j, <https://logging.apache.org/log4j/2.x/>
5. Maven, <https://maven.apache.org/>
6. Firebase, <https://firebase.google.com/>