Razviti softversko rešenje - web aplikaciju koja omogućava online pretragu i iznajmljivanje automobila (Rent-a-car).

Učesnici/Korisnici sistema

- Krajnji korisnik - Pretražuje i iznajmljuje automobile, pri čemu može da razmenjuje poruke sa auto kućama/vlasnicima automobila. Naknadno može da oceni uslugu i ostavi komentare. Neregistrovani korisnik može samo da pretražuje automobile.

- Agent/Firma - Postavljaju nove automobile u ponudu, definišu cenovnik, popuste, primaju i upravljaju primljenim zahtevima za iznajmljivanje vozila, pri čemu mogu i sami unositi podatke o zauzeću pojedinih vozila.

- Administrator - Ima pristup kompletnoj bazi, a ključni zadatak mu je da upravlja poslovnim korisnicima, kao drugim entitetima. Može da registruje Agente/Firmu, odnosno odobrava postojeću registraciju za pristup sistemu i edituje permisije i role postojećim korisnicima.

- Vozilo - Uređaj u vozilu periodično emituje podatke o lokaciji koji se koriste kako bi se ono prikazalo na mapi (za ocenu 10).

Moduli projekta

- Klijentski, agentski i administratorski Front-End - Obezbeđuje interfejs i funkcionalnosti neophodne klijentu, agentu, kao i kontrolni panel za administratora sistema.

- Back-End - Sadrži kompletnu poslovnu logiku aplikacije pri čemu ovaj deo mora biti realizovan putem mikroservisa (Microservice architecture).

- Agentska aplikacija (Front-end i Back-end) - Aplikacija za iznajmljivanje automobila treba da podrži i već postojeće firme, odnosno da im omogući jednostavno oglašavanje svojih ponuda i van svojih aplikacija i da na taj način povećaju svoj promet.

- Android aplikacija - Iz vozila periodično šalje podatke o svojoj lokaciji koja se može prikazati na mapi. (Za potrebe projekta nije neophodno realizovati android aplikaciju, već je dovoljno definisati skriptu koja će periodično slati koordinate)

- Servis za slanje mejlova - Servis koji treba biti online (deploy-ovan). Istražiti GitHub student pack i videti šta se sve nudi kao rešenje za deploy (za ocenu 10, u suprotnom, slanje mejlova realizovati iz pojedinačnih mikroservisa).

Komunikacija između modula

- Komunikacija između Back-end-a i agentske aplikacije ostvaruje se putem SOAP protokola.
- Front-end aplikacije konzumiraju REST servise.
- Android aplikacija i Back-end komuniciraju tako što razmenjuju poruke koristeći message queue.
- Komunikacija između back-end-a i servisa za slanje mejlova se odvija tako što back-end publish-uje poruke u message queue koje mejl servis konzumira (jednosmerno)..

Funkcionalnosti

Back-end

Obezbeđuje navedene funkcionalnosti običnim korisnicima:

- ❏ Registoravanje i logovanje na sistem.
- ❏ Pretraživanje automobila - Minimalno korisnik treba da unese mesto odakle želi da preuzme automobil, datum kada vrši preuzimanje (hh:mm dd-MM-yyyy) (najmanje 48h od trenutka pretrage) i datum kada će izvršiti povratak automobila (hh:mm dd-MM-yyyy).
- ❏ Napredna pretraga podrazumeva da korisnik unese dodatne parametre:
  - ❏ Marku automobila (BMW, Audi, Mercedes, Tesla i ostali),
  - ❏ Model automobila (M5, R8...),
  - ❏ Vrstu goriva (benzin, plin, dizel…),
  - ❏ Tip menjača (manuelni, automatski, poluautomatski...),
  - ❏ Klasu automobila (SUV, old tajmer, gradski auto...),
  - ❏ Cenu (od - do),
  - ❏ Pređenu kilometražu automobila,
  - ❏ Kilometražu koju planira da pređe (neka vozila imaju ograničen broj kilometara koji se mogu preći tokom rentiranja, dok je kod nekih ta opcija UNLIMITED). Ako postoji ograničenje i ako se pređe više kilometara, cenovnikom je određeno koliko se onda dodatno mora platiti pri svakom pređenom kilometru,
  - ❏ Da li postoji opcija kupovine Collision Damage Waiver protekcije (kupovinom ove opcije se smanjuju troškovi u slučaju neke nezgode ili krađe automobila),
  - ❏ Broj sedišta za decu.
- ❏ Po izvršenom pretraživanju, korisniku se prikazuju sva dostupna vozila (dostupna od strane drugih korisnika ili firmi) za odgovarajući datum sa svim neophodnim podacima (slika, ocena kao i podaci opisani u naprednoj pretrazi). Navedene rezultate je moguće sortirati po:

        - ceni,
        - oceni,
        - Kilometraži.
    - Izborom na pojedinačno vozilo, prelazi se na detaljan prikaz, gde je moguće videti sve fotografije kao i podatke o automobilu (model, klasa, tip menjača itd.)
    - Sa stranice sa prikazanim rezultatima, moguće je dodavati automobile u korpu iz koje je moguće kreirati zahtev za iznajmljivanje automobila (dakle moguće je iznajmiti jedan ili više automobila). Za automobile koji potiču od različitih vlasnika, biće kreirani različiti zahtevi. Za automobile koji potiču od istog vlasnika, moguće je opcijom definisati da li će za svaki automobil biti kreiran zahtev posebno (moguće da vlasnik neka iznajmljivanja odobri, a neka odbije - što je korisniku ok), ili da sve kreira kao jedan zahtev (bundle) (korisnik želi npr. sva tri vozila, odnosno ne odgovara mu da budu odabrana dva, pa da treći automobil traži ponovo).
    - Zahtev/i se kreira/ju sa statusom PENDING, pri čemu vlasnik automobila (drugi korisnik ili firma) može da potvrdi ili odbije zahtev. Moguće je da više korisnika napravi zahtev za dobijanje jednog automobila, ali se može prihvatiti samo jedan (od strane korisnika ili agenta/firme). Onaj koji je prihvaćen, prelazi u stanje RESERVED pri čemu korisnik ima 12h da izvrši online plaćanje (Samo za studente koji slušaju poslovnu informatiku. U suprotnom, zahtev prelazi u stanje PAID). Ako se to ne ispuni u zadatom roku, zahtev prelazi u stanje CANCELED i vlasnik može da prihvati neki drugi zahtev. Ako se izvrši plaćanje, zahtev prelazi u stanje PAID pri čemu se automatski svi ostali zahtevi odbijaju (i pojedinačni zahtevi za automobil i zahtevi koji uključuju automobil u bundle-u). Zahtev koji nije obrađen 24h automatski prelazi u stanje CANCELED.
    - Nakon što je zahtev prihvaćen (Stanje RESERVED), korisnik može da razmenjuje poruke sa vlasnikom, kako bi se eventualno dodatno informisao ili izneo neki svoj zahtev za dodatnu opremu vozila).
    - Korisnik može uvek da pregleda svoju istoriju zahteva i da vidi razmenjene poruke sa vlasnikom.
    - Nakon isteka vremena korišćenja vozila, korisnik može da ostavi ocenu i komentar (koji mora biti odobren od strane administratora).
    - Korisnik pre plaćanja može u bilo kom trenutku da otkaže zahtev.
    - Svaki običan registrovani korisnik može da stavi oglas (ukupno 3) i realizuje funkcionalnosti opisane u produžetku (bez mogućnosti definisanja popusta i dobijanja statistike).

Obezbeđuje navedene funkcionalnosti agentima/firmama koji direktno koriste aplikaciju:

- Mogu da kreiraju oglas (neograničen broj) za vozilo koje se može iznajmiti tako što će uneti slike i sve neophodne informacije (navedene u naprednoj pretrazi) kao i period u kom je vozilo slobodno. Ako vozilo poseduje android uređaj, postoji mogućnost registracije vozila na licu mesta pri čemu se dobija token koji se podešava u andorid aplikaciji (skripti).
- Samostalno mogu da unesu zauzeće pojedinog automobila za određen period (u slučaju kada lice fizički rentira automobil) - što automatski odbija postojeće zahteve.
- Mogu da komuniciraju sa korisnicima čije su zahteve prihvatili kako bi sa njima dogovorili dodatne detalje.

- Mogu da pregledaju ocene i komentare i ostave svoje komentare.
- Po završetku rentiranja, treba da unesu izveštaj o broju pređenih kilometara (ako je u pitanju bundle, onda za svaki automobil pojedinačno) i po potrebi unesu neke dodatne informacije (kao slobodan tekst).
- Ako je postojalo ograničenje za kilometražu koje je pređeno, automatski se obračunava cena koja se korisniku šalje na naplatu. Dokle god korisnik ne plati, neće moći da rentira nova vozila.
- Agenti mogu da zatraže statistku upotrebe njihovih vozila (koja vozila su prešla najviše kilometara, koja imaju najviše komentara, koja imaju najbolju ocenu).
- Neophodno je definisati cenovnik pri čemu treba definisati cenu za svaki dan, gde jedan cenovnik može važiti za više oglasa (dakle može postojati više cenovnika). Dodatno se u njemu treba definisati cena po kilometru za automobile za koje postoji ograničenje u kilometraži i cena za Collision Damage Waiver za oglase u kojima se ta opcija nudi.
- Moguće je definisati popuste gde bi pri dužem rentiranju vozila (više od 30 dana, cena po danu bila umanjena za neki procenat - npr. 20%).
- Postoji mogućnost praćenja vozila putem mape ukoliko vozilo može da emituje svoju lokaciju (ocena 10).

Aplikacija treba da omogući i već postojećim firmama/agentima da svoje ponude prezentuju krajnjim korisnicma. Umesto da se firma odvojeno registruje i odvojeno održava podatke na više mesta (na sopstvenoj aplikaciji i našoj), treba omogućiti sinhronizaciju podataka, odnosno pružiti API koji svaka firma može da konzumira (prethodno neophodna registracija firme). Taj API treba da odgovara funkcionalnostima opisanim iznad. Dodatno treba omogućiti sinhronizaciju lokalne baze firme i baze/a back-end-a. Usled nedostupnosti jedne od navedenih baza, one se mogu desinhronizovati, pa sa ponovnom uspostavom komunikacije obe baze ponovo treba da se dovedu u konzistentno stanje.

Napomena: Agentsku aplikaciju realizovati kao odvojen monolit sa drugačijim izgledom fronta koja treba da podrži prethodno opisane funkcionalnosti ili najbolje da ima i proširenији/drugačiji model pa da treba vršiti dodatne konverzije pre razmene podataka sa drugom aplikacijom.

Obezbeđuje navedene funkcionalnosti administratorima:

- Održavanje šifrarnika (modeli automobila, klase, tip goriva….)
- Objavljivanje ili odbijanje komentara korisnika.
- Blokirati, aktivirati i uklanjati iz sistema obične korisnike.
- Registrovati agente/firme pri čemu se registruju sledeći podaci:
  - Ime i prezime ili naziv firme
  - Adresa
  - Poslovni matični broj
- Defisati permisije za svakog korisnika pojedinačno (npr. administrator može onemogućiti kreiranje rezervacija od strane nekog korisnika koji je mnogo puta otkazivao).

Obezbeđuje navedene funkcionalnosti vozilu:

- ❏ Vozilo periodično šalje svoje koordinate koje se mogu prikazivati na mapi.

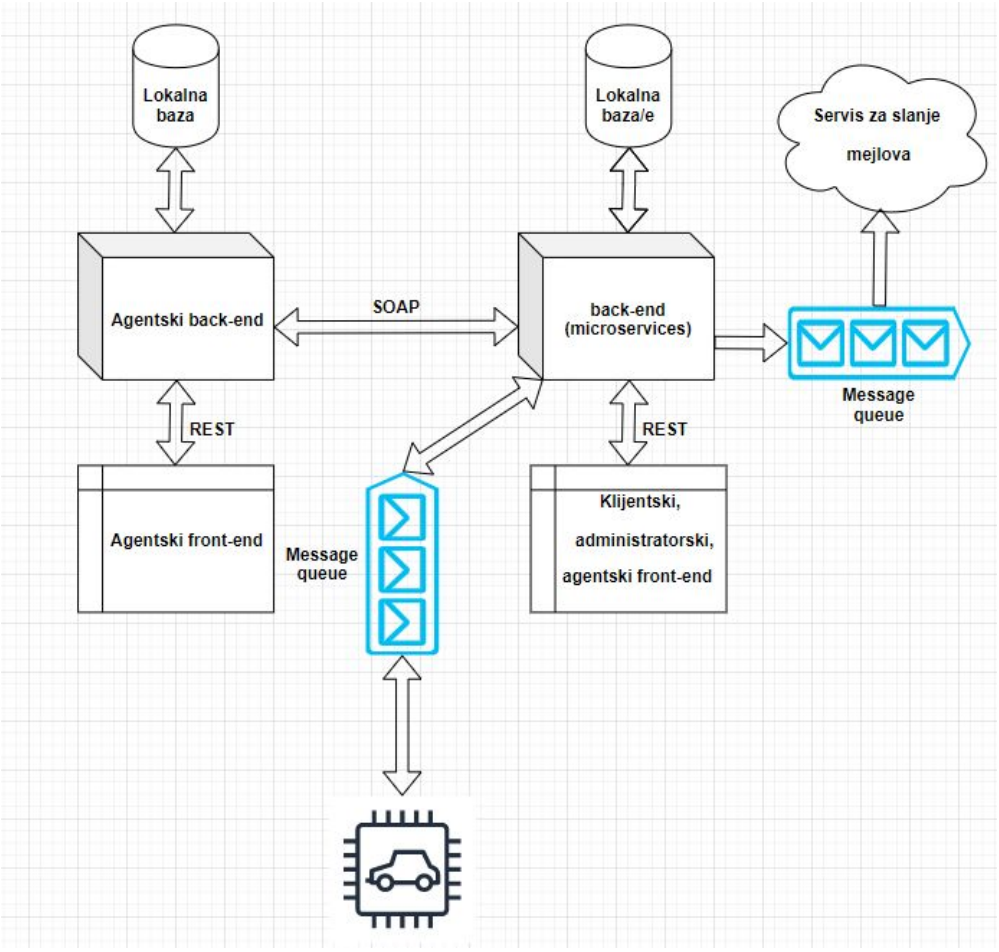
Za sve navedene entitete koje je moguće identifikovati odavde, neophodno je podržati kreiranje, čitanje, izmenu i brisanje (CRUD) pri čemu treba identifikovati kada je dozvoljeno menjanje i brisanje.

Način realizacije projekta

Projekat se realizuje timski, pri čemu timovi broje do 4 člana. Timovi sa 3 člana su takođe prihvatljivi. Tim od 2 člana iako je moguć, pretpostavljamo da bi bilo previše posla po članu da bi se projekat uspešno realizovao i imao korektne funkcionalnosti.

Studenti treba da:

- Razviju model podataka neophodan za realizaciju kompletnih funkcionalnosti (Analizirati koji podaci se koriste u sistemu kao i koje međuzavisnosti postoje).
- Definišu neophodne komunikacije kako bio ceo sistem funkcionisao na adekvatan način (Definisati adekvatne servisne endpointe pri čemu voditi računa da je back-end neophodno razviti kao skup mikroservisa).
- Realizuju sve navedene funkcionalnosti vodeći računa o svim graničnim slučajevima, odnosno omogućiti pravilno funkcionisanje aplikacije.



ANEKS 1 - Poslovna informatika

U slučaju da slušate predmet Poslovna informatika, i da imate želju da ukombinujete ovaj projekat sa tim predmetom, Rent-a-car projekat bi trebalo proširiti tako da se oslanja na jedan od narednih podsistema. Detaljnije o tome šta se očekuje od ovih podsistema će biti rečeno kroz predavanja i odvojenu specifikaciju zadataka, dok se ovde govori samo o tome šta je ključno za saradnju sa Rent-a-Car aplikacijom, na šta se oslanja Rent-a-Car aplikacija.

1. Poslovna banka
    a) Plaćanje (proslediti nalog za plaćanje banci)
    b) Preuzimanje stanja na računu, ili izvoda
2. Prodaja (fakturisanje)
   U okviru zasebnog servisa se smeštaju cenovnici, narudžbe, fakture, šifrarnici (i ostalo što je potrebno za kompletan podsistem), dok se Rent-a-Car radi ovakvih informacija obraća ovom servisu.
3. Likvidatura
   Primanje faktura radi zatvaranja
4. Knjigovodstvo
   Automatsko knjiženje svih poslovnih događaja koji se dese u Rent-a-Car sistemu

ANEKS 2 - Bezbednost u sistemima elektronskog poslovanja

Rent-a-car rešenje je neophodno obezbediti integracijom bezbednosnih kontrola u njegove module, kao i uvođenjem bezbednosnim alatima koji su opisani u ovom poglavlju.

PKI

Implementirati alat za podršku infrastrukture javnih ključeva. Specifikacija projektnog zadatka je definisana kroz skup korisničkih zahteva. Potrebno je dizajnirati i implementirati PKI vođeni ovim zahtevima.

Bezbednost modula

Potrebno je obezbediti čitav Rent-a-car sistem. Svaku bezbednosnu kontrolu treba integrisati prateći *best practice* konfiguraciju i šablone bezbednog dizajna (višeslojna odbrana, najmanja privilegija, jednostavan dizajn, itd.).

Zaštita podataka

Osetljivi podaci sa kojim aplikacija radi treba da budu obezbeđeni u skladištu, u transportu i tokom upotrebe. Identifikovati osetljive podatke, definisati i implementirati prikladne bezbednosne kontrole. Podaci čije skladištenje se ne može izbeći treba da budu šifrovani ili heširani ukoliko je to prikladno. Poruke u internoj komunikacije treba da imaju očuvanu poverljivost, integritet i neporecivost, kao i da budu zaštićene od *reply* napada. Komunikacija između veb-čitača i servera treba da bude zaštićena sigurnom konfiguracijom HTTPS protokola. Sertifikate generisati putem PKI alata.

Kontrolna pristupa

Korisnički interfejsi modula treba da podrži prikladne mehanizme za autentifikaciju i autorizaciju.

Mehanizmi autentifikacije treba da podrže bezbednu registraciju, prijavu na sistem, odjavu, promenu lozinke i resetovanje lozinke. Autorizacija podrazumeva kontrolu pristupa po RBAC modelu.

OWASP Top 10

Kompletan sistem treba da reguliše sve rizike sa OWASP Top 10 liste, gde je neophodno sastaviti temeljan izveštaj kako su koji rizici adresirani. Objasniti koje grupe napada su relevantne, kako je sistem zaštićen od njih, ili kako bi bio zaštićen prilikom postavljanja u produkciju.

Zadatak za 10

Za najvišu ocenu je neophodno realizovati jednu od celina definisanih u ovom poglavlju.

Single sign-on

Potrebno je omogućiti single sign-on (u daljem tekstu SSO) prijavu na kompletan sistem. Mehanizam za SSO se može implementirati konfigurisanjem gotovih rešenja, poput Active Directory ili Keycloak i njihovom integracijom sa ostatkom sistema.

Penetration testing

Sprovesti penetraciono testiranje veb-aplikacija i servera upotrebom bar dva alata iz grupe: Nmap, Nikto, dirbuster, sqlmap, OWASP ZAP, Burp Suite. Formirati izveštaja penetracionog testa i regulisati ranjivosti.

Threat model

Kreirati model pretnji implementiranog sistema. Model pretnji podrazumeva sagledavanje ranjivosti, pretnji, napada, kontrola za njihovo sprečavanje i negativnih uticaja uz arhitekturne dijagrame i dijagrame tokova podataka.