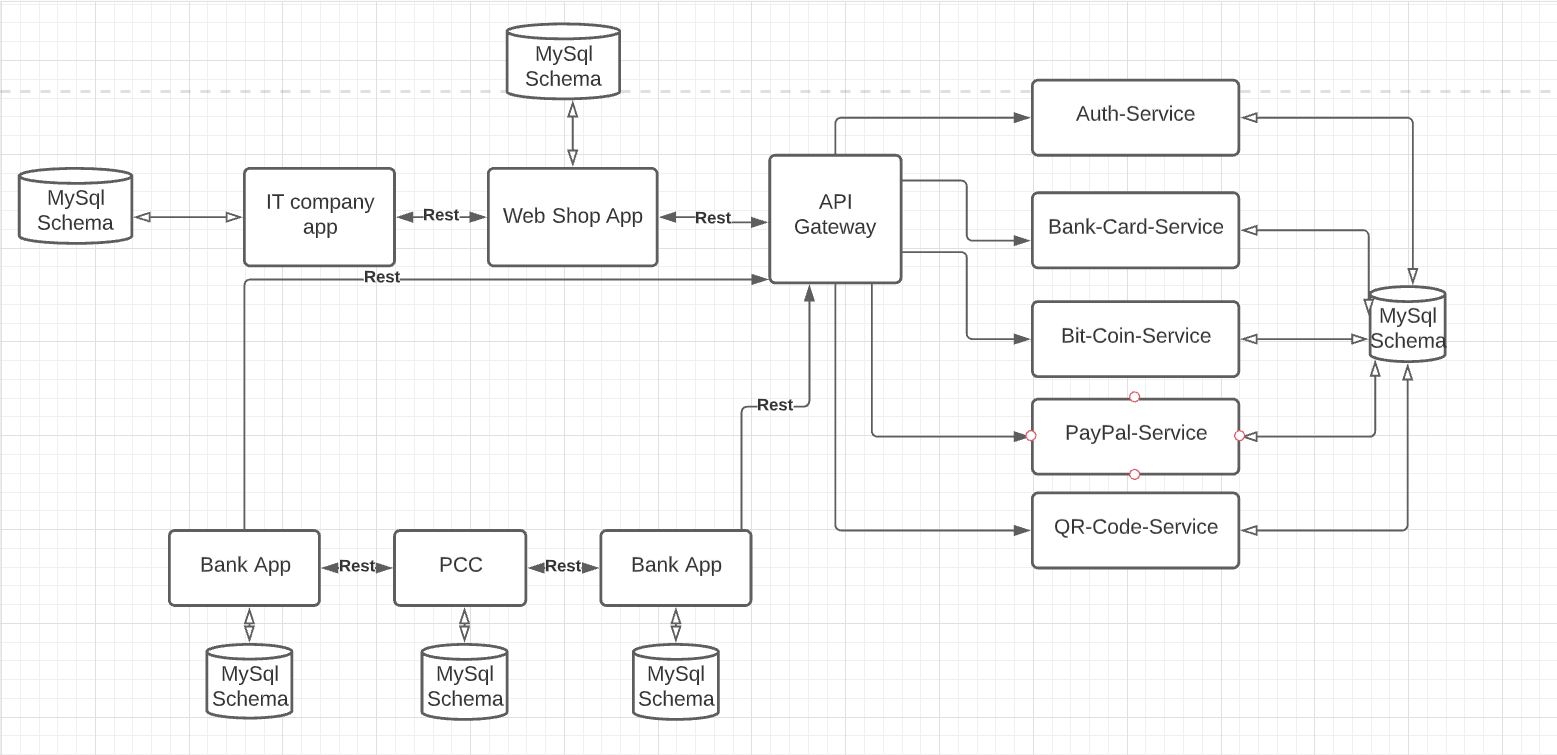
# Sistemi elektronskog placanja(IT FIRMA)

Na Slici 1. Predstavljena je arhitektura sistema koji ce biti detaljno opisan u nastavku teksta.



Slika 1

IT firma predstavlja monolitnu aplikaciju koja komunicira pomocu REST tehnologije sa aplikacijom WEB prodavnice koji dalje preko API gatewaya komunicira sa Payment Service Provider aplikacijom koja ima mikroservisnu arhitekturu.

## Payment Service Provider

Payment Service Provider predstavlja mikroservisni sistem sačinjen od servisa za plaćanje. To je

aplikacija koja raspolaže sa raznim načinima plaćanja i omogućava korisnicima ove aplikacije da

obezbede plaćanje u okviru njihovih sistema, kako ne bi morali da implementiraju plaćanja u svom sistemu, takođe i odgovornost u vezi sa plaćanjem se prebacuje na PSP.

Servisi PSP-a su opisani u nastavku teksta.

### Auth-Service

Predstavlja servis za pretplacivanje Web prodavnica na ponudjene nacine placanja.Registraciju na PSP obavaljaju admini Web prodavnica a zatim nakon prijave na PSP, mogu da odaberu nacine placanja koji ce biti podrzani u njihovim Web prodavnicama. Servisi koje nudi PSP su servis za plaćanje karticom, servis za plaćanje PayPal-om, servis za plaćanje BitCoin-om ili placanje pomocu QR koda. Pored ovih servisa biće omogućeno je uvodjenje novih nacina placanja.

### Bank Card Service

Predstavlja servis za plaćanje karticom. U sistemu postoje dvije vrste banke. Jedna banka predstavlja banku kupca, a druga predstavlja banku prodavca. Kada korisnik izabere plaćanje karticom od njega će se zahtijevati da popuni formu sa podacima koji su neophodni za obavljanje transakcije. To su recimo podaci o tome ko je prodavac, koja je količina novca koji se prenosi transakcijom, identifikator transakcije itd. Dio podataka sadrži sam PSP.

Kada se popune podaci banka prodavca provjerava da li su oni uneseni korektno. Npr. Ukoliko

prodavac nema racun u banci to bi trebalo da izbaci izuzetak kojim bi trebalo rukovati itd.

Ukoliko su svi podaci popunjeni korektno kupac se preusmjerava na sajt banke gdje moze dati svoje podatke. Na osnovu tih podataka opet se vrši provjera.

Ukoliko su banke kupca i banke prodavca iste može se provjeriti da li postoje raspoloživa sredstva i izvršiti se transakcija.

U suprotnom, tj. Ukoliko banke kupca i prodavca nisu iste, nastupa **PCC servis** koji ima za ulogu da na osnovu unijetih podataka kreira zahtjev i posalje banci kupca. Ako je zahtjev ispravan banka kupca prihvata i ako ima dovoljno sredstava izvrsi se transakcija. Zatim se preko PCC servisa vraca odgovor, rezultat transakcije, banci prodavca, kako bi ona obradila ishod transakcije. Kupac se obavijesti o uspjesnosti transakcije i u slucaju da je sve proslo kako treba.

### Pay Pal Service

Predstavlja servis za plaćanje preko PayPal mehanizma.

Kada kupac odabere placanje PayPal-om bice preusmjeren na njihov sajt da bi se prijavio na svoj

nalog. Da bi se testiralo da sve radi kako treba PayPal developer nudi opciju korišćenja sendbox-a, dakle moguće je kreirati naloge za testiranje koji ce imati podatke identične stvarnom PayPal-u, samo će biti virtuelni, a simulacija plaćanja biće identična.

Kada se korisnik prijavi bice potrebno da unese podatke o tome ko šalje, kome šalje, koja količina novca je u pitanju... Svaka transakcija obavljaće se između virtuelnih naloga.

### BitCoin Service

Predstavlja servis za plaćanje preko BitCoin kriptovalute.

Da bi se izvrsila BitCoin transakcija za kupca je potrebno da ima digitalni wallet i odredjenu

količinu BitCoin-a ili nek druge kriptovalute. U walletu se čuvaju digitalne valute. Ponaša se kao

račun banke za BitCoin pomoću kog se čuva, plaća i dobija novac. Nakon postavljanja walleta

potrebno je kupiti BitCoin-e pomoću servisa za transakciju kriptovaluta. Kada se uključi mogućnost plaćanja BitCoin-om potrebna je samo BitCoin adresa da se unese u određeno polje u walletu. Unese se količina BitCoin-a koja se želi poslati, takođe potrebno je korektno uneti podatke.

**Tehnologije koje će biti korišćenje:**

- Java Spring Boot (implementacija mikroservisa)

- React JS (implementacija frontend servisa)

- MySQL baza podataka