



DIZAJN I IMPLEMENTACIJA MODERNOG SISTEMA ZA ELEKTRONSKO
PLAĆANJE
DESIGN AND IMPLEMENTATION OF A MODERN ELECTRONIC PAYMENT SYSTEM

Mladen Gajić, Fakultet tehničkih nauka, Novi Sad

Oblast – ELEKTROTEHNIKA I RAČUNARSTVO

Kratak sadržaj – *Elektronsko plaćanje predstavlja ključnu komponentu savremenog poslovanja, omogućavajući brze, sigurne i efikasne transakcije između korisnika i finansijskih institucija. Ovaj rad istražuje dizajn i implementaciju modernog sistema za elektronsko plaćanje zasnovano na mikroservisnoj arhitekturi. Korišćene tehnologije uključuju Spring Boot, React, PostgreSQL, i Docker Compose što omogućava visoku skalabilnost, fleksibilnost i sigurnost sistema. Rad takođe analizira tehničke izazove tokom implementacije, prednosti mikroservisne arhitekture, kao i potencijalna unapređenja, uključujući prelazak sa Docker Compose na Kubernetes, proširenje funkcionalnosti i usklađivanje sa zakonskim regulativama.*

Ključne reči: *elektronsko plaćanje, mikro-servisna arhitektura, integracija sa ostalim sistemima za plaćanje, bankarski sistem, transakcije*

Abstract – *Electronic payment is a key component of modern business, enabling fast, secure and efficient transactions between users and financial institutions. This paper investigates the design and implementation of a modern electronic payment system based on a microservice architecture. Technologies used include Spring Boot, React, PostgreSQL, and Docker Compose, which enables high scalability, flexibility, and system security. The paper also analyzes the technical challenges during implementation, the advantages of microservices architecture, as well as potential improvements, including the transition from Docker Compose to Kubernetes, expanding functionality and compliance with legal regulations.*

Keywords: *electronic payment, micro-service architecture, integration with other payment systems, banking system, transactions*

1. UVOD

Digitalna transformacija donela je revolucionarne promene u svim aspektima društva, uključujući način na koji obavljamo finansijske transakcije.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bila dr Dunja Vrbaški, docent.

Elektronsko plaćanje postalo je neizostavan deo globalne ekonomije, omogućavajući trenutne, sigurne i praktične transakcije između korisnika, trgovaca i finansijskih institucija. Pojava interneta, mobilnih uređaja i digitalnih platformi stvorila je uslove za razvoj sistema koji omogućavaju obavljanje transakcija bilo gde i bilo kada, eliminišući potrebu za fizičkim prisustvom ili gotovinom. U ovom radu je analiziran dizajn i implementacija modernog sistema za elektronsko plaćanje, koji koristi mikroservisnu arhitekturu za postizanje visoke skalabilnosti, modularnosti i sigurnosti.

2. MIKROSERVISNA ARHITEKTURA

Mikroservisna arhitektura je moderan pristup dizajnu softverskih sistema, gde se aplikacija sastoji od niza manjih, nezavisnih servisa, od kojih svaki obavlja specifičan zadatak. Ovi servisi komuniciraju međusobno putem dobro definisanih API interfejsa, što omogućava decentralizovan razvoj, testiranje i distribuciju. Prednosti mikroservisne arhitekture u odnosu na tradicionalnu monolitnu arhitekturu su brojne, uključujući skalabilnost, fleksibilnost, otpornost na greške i mogućnost brze isporuke novih funkcionalnosti [1].

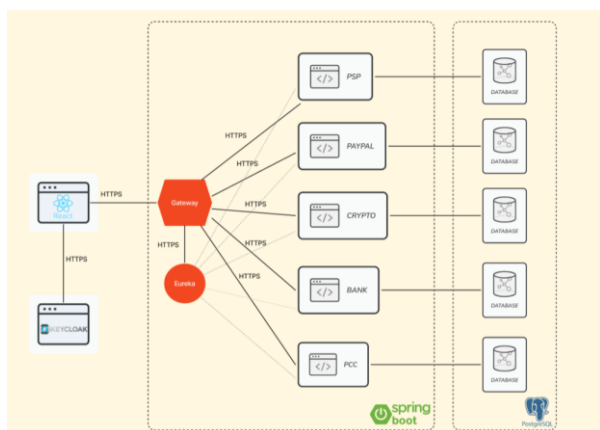
2.1. PREDNOSTI MIKROSERVISNE ARHITEKTURE

Mikroservisna arhitektura omogućava skaliranje pojedinačnih servisa po potrebi, što znači da se resursi mogu efikasno koristiti i distribuirati u zavisnosti od trenutnih opterećenja. Na primer, ako određeni servis doživljava povećan promet, moguće ga je skalirati horizontalno, dodavanjem novih instanci, bez potrebe za skaliranjem cele aplikacije. Ovaj pristup omogućava bolju performansu sistema, smanjujući rizik od zagušenja i omogućavajući optimalno korišćenje resursa. Jedna od ključnih prednosti mikroservisne arhitekture je otpornost na greške. Kvar jednog servisa ne mora nužno da utiče na funkcionisanje celog sistema, što značajno povećava dostupnost i pouzdanost usluga. Zbog izolacije servisa, sistem može ostati funkcionalan čak i ako jedan ili više servisa prestanu sa radom. Ovo je posebno važno za sisteme elektronskog plaćanja, gde je neprekidnost usluge ključna za korisničko poverenje i zadovoljstvo. Razvojni timovi mogu raditi na različitim servisima nezavisno jedni od drugih, koristeći različite tehnologije i programske jezike koji najbolje odgovaraju potrebama svakog servisa. Ovaj nivo fleksibilnosti omogućava brži razvoj i lakše

održavanje sistema, kao i brzu integraciju novih funkcionalnosti [1].

2.2. MIKROSERVISI U SISTEMU ELEKTRONSKOG PLAĆANJA

Sistem za elektronsko plaćanje razvijen u ovom radu sastoji se od nekoliko ključnih mikroservisa, od kojih svaki ima specifičnu ulogu u okviru sistema. Ilustracija sistema prikazana je na slici 1.



Slika 1. Uprošćeni prikaz mikroservisne arhitekture korišćene za izradu sistema za elektronsko plaćanje

Gateway servis: Funkcioniše kao ulazna tačka u sistem, upravlja rutiranjem zahteva, autentifikacijom i autorizacijom korisnika, kao i kontrolom pristupa. Korišćenje Spring Cloud Gateway-a omogućava centralizovano upravljanje protokom podataka i primenu sigurnosnih politika.

Payment Service Provider (PSP) servis: Implementiran kao centralna tačka za integraciju sa različitim platnim platformama. Ovaj servis komunicira sa korisničkom aplikacijom i obavlja procesiranje plaćanja kroz integraciju sa eksternim platnim provajderima.

Bank servis: Simulira operacije banke unutar sistema, uključujući evidenciju klijenata, vođenje računa i obavljanje transakcija. Svaka banka u sistemu poseduje svoju instancu Bank servisa, što omogućava izolaciju podataka i specifičnu konfiguraciju za svaku instituciju.

PCC (Payment Card Clearing) servis: Obavlja funkcionalnosti klirinških kuća za platne kartice, omogućavajući autorizaciju i poravnanje transakcija između različitih banaka. Ovaj servis je ključan za obezbeđivanje sigurnih i efikasnih međubankovnih transakcija.

Crypto servis: Dizajniran za podršku plaćanja putem kriptovaluta, omogućavajući transakcije u različitim digitalnim valutama. Korišćenje Spring Boot-a za integraciju sa blockchain mrežama pruža funkcionalnosti poput generisanja adresa za uplatu, validacije transakcija i praćenja stanja na digitalnim novčanicima.

3. KORIŠĆENE TEHONOLOGIJE

Za implementaciju sistema za elektronsko plaćanje korišćene su savremene tehnologije koje omogućavaju visoku skalabilnost, fleksibilnost i sigurnost sistema. U ovom delu rada detaljno ćemo analizirati ključne tehnologije koje su korišćene u implementaciji.

3.1. Spring Boot

Spring Boot je moderan tehnološki okvir razvijen kao nadogradnja na već dobro poznati Spring Framework, namenjen za kreiranje samostalnih i proizvodnih Spring aplikacija uz značajno smanjenje složenosti konfiguracije i ubrzavanje celokupnog procesa razvoja. *Spring Boot* pruža unapred definisane konfiguracije za različite vrste aplikacija, što omogućava brži razvoj i lakše održavanje sistema. Jedna od ključnih prednosti *Spring Boot*-a je mogućnost kreiranja samostalnih aplikacija koje uključuju ugrađeni server, poput *Tomcat* servera, što eliminiše potrebu za instaliranjem i održavanjem eksternog aplikacionog servera. Ovo značajno pojednostavljuje razvoj, testiranje i distribuciju aplikacija, omogućavajući razvojnim timovima da se fokusiraju na poslovne funkcionalnosti umesto na tehničku infrastrukturu [2]. Ovaj pristup omogućio je modularan i fleksibilan razvoj sistema, koji je lako skalabilan i prilagodljiv budućim potrebama.

3.2. React

React je jedna od najpopularnijih *JavaScript* biblioteka za izradu korisničkih interfejsa, razvijena od strane *Facebook*-a (sada *Meta*). *React* omogućava kreiranje interaktivnih i dinamičnih korisničkih interfejsa putem komponentne arhitekture, koja omogućava razvoj modularnih i ponovnih komponenti. U našem sistemu, *React* je korišćen za izradu korisničke stranice, omogućavajući korisnicima jednostavan i intuitivan interfejs za obavljanje transakcija. Komponentna arhitektura *React*-a omogućila je brz razvoj svih neophodnih funkcionalnosti, dok je virtuelni *DOM* osigurao visoke performanse i responzivnost aplikacije [3].

3.3. PostgreSQL

PostgreSQL je moćan sistem za upravljanje relacionim bazama podataka, poznat po svojoj stabilnosti, pouzdanosti i bogatom skupu funkcionalnosti. *PostgreSQL* podržava *ACID*(*Atomicity, Consistency, Isolation, Durability*) svojstva, koja osiguravaju doslednost i integritet podataka, što je od ključne važnosti za sisteme elektronskog plaćanja. Svaki mikroservis ima svoju bazu podataka unutar iste *PostgreSQL* instance. Ova arhitektura omogućava izolaciju podataka između mikroservisa, čime se povećava sigurnost i smanjuje mogućnost konflikata između različitih delova sistema [4].

3.4. Docker

Doker je platforma za razvoj, isporuku i pokretanje aplikacija u izolovanim okruženjima poznatim kao kontejneri. Doker omogućava brzu i efikasnu kontejnerizaciju aplikacija, što olakšava upravljanje i skaliranje sistema. Korišćenje Dokera u okviru sistema za

elektronsko plaćanje omogućilo je izolaciju svakog mikroservisa, čime su minimizovani konflikti između servisa i povećana sigurnost sistema. Docker je takođe omogućio brzo skaliranje servisa, kao i jednostavno upravljanje zavisnostima i konfiguracijama aplikacija [5].

3.5. Docker Compose

Docker Compose je alat za definisanje i pokretanje višestrukih Docker kontejnera kao jedne aplikacije. U okviru ovog sistema, *Docker Compose* je korišćen za pokretanje svih mikroservisa i njihovih zavisnosti, omogućavajući jednostavno testiranje i razvoj celokupnog sistema u lokalnom okruženju [5].

4. INTEGRACIJA SISTEMA ZA ELEKTRONSKO PLAĆANJE SA OSTALIM APLIKACIJAMA

Integracija sistema za elektronsko plaćanje sa postojećom ostalim aplikacijama predstavlja ključni korak ka unapređenju korisničkog iskustva i povećanju efikasnosti poslovanja. Elektronsko plaćanje je postalo standard u modernom poslovanju, omogućavajući brže i sigurnije transakcije. Proces integracije obuhvata nekoliko važnih elemenata, uključujući strategiju implementacije plaćanja i tehničke aspekte registracije korisnika i trgovaca u bankarskom sistemu. Takođe u sistemu elektronskog plaćanja razlikujemo nekoliko procesa plaćanja, koji su opisani u narednim delovima ovog rada.

5. PROCES NAPLATE KARTICOM

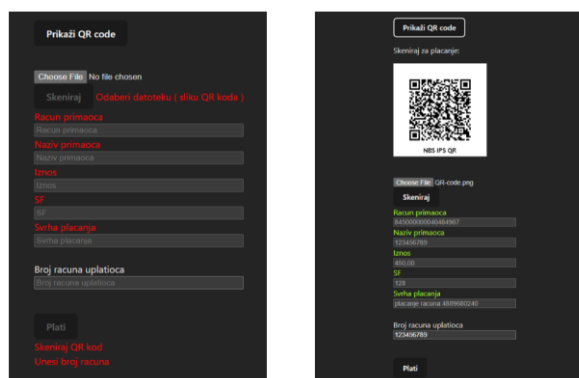
Elektronsko plaćanje karticom u sistemu elektronskog plaćanja obuhvata nekoliko faza:

- *Inicijalizacija transakcije:* Kupac potvrđuje kupovinu u aplikaciji i bira način plaćanja karticom. Nakon toga, sistem ga preusmerava na stranicu *PSP (Payment Service Provider)* gde unosi podatke o kartici.
- *Autentifikacija i validacija:* Podaci o kartici se prosleđuju *PSP* servisu, koji obavlja validaciju kod banke izdavaoca kartice. Banka proverava tačnost podataka i raspoloživost sredstava, te autorizuje transakciju.
- *Realizacija transakcije:* Ako je transakcija odobrena, sredstva se rezervišu na računu kupca i prenose na račun prodavca. Kupac dobija povratnu informaciju o uspehu transakcije.

Postoji jednostavniji i kompleksniji scenario koji može da se dogodi u zavisnosti u kojim bankama kupac i prodavac imaju račune. Jednostavni scenario jeste kada kupac i prodavac imaju račune u istoj banci, što omogućava bržu realizaciju transakcije jer se sve operacije odvijaju unutar istog sistema. Složeniji scenario jeste taj kada kupac i prodavac imaju račune u različitim bankama. U ovom slučaju, proces uključuje *PCC (Payment Clearing and Settlement)* servis, koji koordinira komunikaciju između banaka kako bi se transakcija uspešno realizovala.

6. PLAĆANJE IPS QR KODOM

Plaćanje putem IPS QR koda predstavlja moderan i brz način digitalne transakcije. Korisnik skenira QR kod putem mobilne aplikacije svoje banke, a aplikacija automatski popunjava sve potrebne podatke za transakciju, što ubrzava proces i smanjuje mogućnost greške. Ovaj metod je posebno koristan u maloprodajnim objektima i restoranima gde je brzina ključna za obavljanje što većeg broja transakcija u kratkom vremenu [6]. Implementacija *IPS* plaćanja u sistemu elektronskog plaćanja pruža korisnicima dodatnu mogućnost, omogućavajući im izbor između tradicionalnog plaćanja karticom i bržeg, jednostavnijeg plaćanja QR kodom, koji je simulirani otpremanjem slike QR koda, kao što je prikazano na slici 2.



Slika 2. Primer izgleda stranice pre i posle otpremanja QR koda

7. INTEGRACIJA SA DRUGIM PLATNIM PROVAJDERIMA

Jedna od važnih funkcionalnosti sistema za elektronsko plaćanje jeste integracija sa različitim platnim provajderima, što omogućava korisnicima fleksibilnost u izboru načina plaćanja. Sistem podržava integraciju sa *PayPal* servisom za plaćanje, kao i plaćanje kriptovalutama čime se omogućava globalna pokrivenost i jednostavno obavljanje transakcija.

7.1. INTEGRACIJA SA PAYPAL SERVISOM ZA PLAĆANJE

PayPal je jedan od najpoznatijih globalnih provajdera elektronskih plaćanja, koji omogućava korisnicima da obavljaju transakcije koristeći njihove *PayPal* naloge. U okviru ovog sistema, *PayPal* servis je razvijen kao poseban mikroservis, koji omogućava integraciju sa *PayPal* API interfejsom za obavljanje transakcija, povrat sredstava i verifikaciju stanja na računima [7]. Integracija sa *PayPal* servisom omogućava korisnicima jednostavan i siguran način plaćanja, dok sistemu omogućava dodatno proširenje funkcionalnosti u skladu sa potrebama tržišta.

7.2. INTEGRACIJA SA KRIPTOVALUTAMA

Kriptovalute, kao što su *Bitcoin* i *Ethereum*, postale su popularan način plaćanja u poslednjih nekoliko godina, posebno u industrijama gde se vrednuju decentralizacija i anonimnost transakcija. U okviru ovog sistema, *Crypto*

servis omogućava korisnicima da obavljaju transakcije koristeći kriptovalute, što uključuje generisanje adresa za uplatu, validaciju transakcija i praćenje stanja na digitalnim novčanicima. *Bitcoin* je prva kriptovaluta, lansirana 2009. godine, dok je *Hedera Hashgraph*, inovacija iz 2018. godine, koja koristi efikasniji algoritam za obezbeđivanje visokog nivoa sigurnosti i brzine. U našem sistemu, *Hedera Hashgraph* omogućava brzo generisanje i validaciju transakcija uz minimalne troškove i energetske efikasnost. Integracija sa alatima poput *HashPack* i *HashScan* dodatno osigurava transparentnost i pouzdanost u obradi kripto transakcija [8,9].

Korišćenje kriptovaluta kao načina plaćanja pruža korisnicima dodatnu fleksibilnost, dok sistemu omogućava prilagođavanje savremenim trendovima u finansijskom sektoru.

8. ZAKLJUČAK

Ovaj rad predstavlja sveobuhvatan pristup dizajnu i implementaciji savremenog sistema za elektronsko plaćanje, koristeći mikroservisnu arhitekturu i savremene tehnologije kao što su *Spring Boot*, *React*, *PostgreSQL* i *Docker*. Sistem je dizajniran da bude fleksibilan, skalabilan i siguran, sa fokusom na integraciju sa različitim platnim provajderima i prilagođavanje savremenim trendovima u finansijskom sektoru. Sistem za elektronsko plaćanje razvijen u ovom radu ima visoku fleksibilnost i skalabilnost, ali postoje i dodatna unapređenja koja bi mogla poboljšati performanse i konkurentnost sistema.

8.1. POTENCIJALNA UNAPREĐENJA

Jedno od glavnih unapređenja je prelazak sa *Docker Compose* na *Kubernetes* platformu, što bi omogućilo bolju automatizaciju, skalabilnost i upravljanje resursima. *Kubernetes* pruža napredne mogućnosti za orkestraciju kontejnera, kao što su automatsko skaliranje, upravljanje opterećenjem i otpornost na greške, što bi dodatno poboljšalo performanse i pouzdanost sistema [10].

Kako bi sistem bio usklađen sa zakonskim regulativama u različitim državama, neophodno je proučiti i implementirati relevantne zakonske zahteve. Ovo uključuje usklađivanje sa zakonima o zaštiti podataka, kao što je *GDPR* u Evropi, kao i sa standardima za bezbednost platnih kartica (*PCI DSS*). Implementacija zakonskih regulativa osigurava da sistem ostane u skladu sa zakonskim okvirima, čime se smanjuje rizik od pravnih posledica i povećava poverenje korisnika [11,12].

Kako bi sistem ostao konkurentan na dinamičnom tržištu elektronskih plaćanja, neophodno je kontinuirano proširivati funkcionalnosti i prilagođavati se novim tehnologijama i trendovima.

9. LITERATURA

[1] Blog Instituta za Matematiku i Informatiku u Kragujevcu, <https://blog.imi.pmf.kg.ac.rs/spring-cloud-mikroservis-arhitektura/>, pristupano 09.09.2024.

[2] Spring Boot Documentation, <https://docs.spring.io/spring-boot/index.html>, pristupano 07.09.2024.

[3] React Documentation, <https://react.dev>, pristupano 08.09.2024.

[4] Worsley, J., & Drake, J. D. (2002). "Practical PostgreSQL". O'Reilly Media, Inc."

[5] Docker Documentation, <https://docs.docker.com/>, pristupano 09.09.2024.

[6] IPS sistem plaćanja Narodne Banke Srbije, <https://www.nbs.rs/sr/ciljevi-i-funkcije/platni-sistem/nbs-operator/ips-nbs/index.html>, pristupano 12.09.2024.

[7] Paypal integration for Spring-Boot backend <https://medium.com/@lsampath210/paypal-integration-for-spring-boot-backend-243e71c89a74>, pristupano 09.09.2024.

[8] Osnove Crypto tehnologije, <https://www.coinbase.com/learn/crypto-basics>, pristupano 12.09.2024

[9] Hashing it out with Hedera Hashgraph <https://www.messari.io/report/hashing-it-out-with-hedera-hashgraph> pristupano 08.09.2024.

[10] Kubernetes Documentation, <https://kubernetes.io/docs/home/>, pristupano 10.09.2024.

[11] Zakon o zaštiti podataka EU, <https://gdpr-info.eu/>, pristupano 14.09.2024

[12] Payment Card Industry Security Standards Council, <https://www.pcisecuritystandards.org/>, pristupano 12.09.2024.

Kratka biografija:



Mladen Gajić rođen 2000. godine u Loznicima, odrastao je u Malom Zvorniku. Srednju tehničku školu "Ivan Sarić" u Subotici završio je 2019. godine. Fakultet Tehničkih Nauka u Novom Sadu upisuje iste te 2019. godine, i postaje student jednog od najprestižnijih fakulteta u Srbiji.

kontakt: mladengajic00@gmail.com