

**VIZUALIZACIJA CSV FAJLOVA BAZIRANA NA AWS SERVERLESS PLATFORMI
VISUALIZATION OF CSV FILES BASED ON THE AWS SERVERLESS PLATFORM**Sanja Drinić, *Fakultet tehničkih nauka, Novi Sad***Oblast – PRIMENJENE RAČUNARSKE NAUKE I INFORMATIKA**

Kratak sadržaj – U ovom master radu je opisana implementacija sistema za vizuelizaciju CSV fajlova koji prikazuju bankovne troškove kroz aplikaciju MoneyDiary. Korišćenjem AWS platforme, aplikacija omogućava analizu troškova, identifikaciju obrazaca potrošnje i praćenje finansijskih promena tokom vremena. Rad istražuje AWS arhitekturu aplikacije i tehničke aspekte implementacije, te predstavlja rezultate u pogledu efikasnosti i korisnosti vizuelizacije za finansijski menadžment.

Ključne reči: vizuelizacija podataka, AWS serverless arhitektura, CSV fajlovi, bankarske transakcije.

Abstract – This paper describes the development of a system for visualizing CSV files that represent bank expenses through the MoneyDiary application. Utilizing the AWS platform, the application enables users to analyze expenses, identify spending patterns, and track financial changes over time. The paper examines the AWS architecture of the application and its technical implementation details, presenting results on the efficiency and usefulness of this visualization for financial management.

Keywords: data visualization, AWS serverless architecture, CSV files, bank transactions.

1. UVOD

U savremenom digitalnom dobu, uz sve češću upotrebu različitih vrsta platnih kartica, kao što su kreditne, debitne i virtuelne kartice, postaje jasno da je efikasno praćenje i analiza finansijskih transakcija od velikog značaja. Kompleksni obrasci potrošnje zahtevaju efikasnu analizu troškova kako bi se uspešno upravljalo ličnim finansijama, što je motivisalo razvoj informacionog sistema "MoneyDiary", namenjenog optimizaciji upravljanja finansijama i poboljšanju finansijske stabilnosti.

Za potrebe rada razvijen je informacioni sistem „MoneyDiary“ koji korisnicima omogućava da vizualizuju finansijske podatke na mesečnom i godišnjem nivou. Proces podrazumeva da korisnici mogu jednostavno učitati svoje finansijske podatke, otpremanjem fajla u CSV formatu koji sadrži informacije o transakcijama korisnika.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Miroslav Zarić, vanredni profesor.

Sistem prihvata i obrađuje podatke, a jednostavnost ovog rešenja ogleda se u mogućnosti brze i jednostavne vizualizacije finansijskih podataka, što korisnicima omogućava prepoznavanje obrazaca potrošnje, analizu troškova po kategorijama i praćenje promena kroz vreme.

Aplikacija "MoneyDiary" bazirana je na AWS serverless [1] arhitekturi. Oslanjajući se na AWS Lambda [2] funkcije za automatsku obradu CSV fajlova sa transakcijama i DynamoDB [3] za brzo i skalabilno skladištenje podataka.

1.2. Amazon Web Services

Amazon Web Services (AWS) [4] je vodeća cloud platforma koja pruža širok spektar usluga za skladištenje, obradu i analizu podataka, kao i za razvoj i pokretanje aplikacija. AWS omogućava organizacijama i pojedincima da koriste računarstvo u oblaku bez potrebe za sopstvenom infrastrukturom na globalnom nivou. AWS obezbeđuje visok nivo pouzdanosti, fleksibilnosti i sigurnosti, eliminisajući potrebu za upravljanjem fizičkom infrastrukturom. Jedna od ključnih prednosti AWS-a je njegova serverless arhitektura, koja omogućava korisnicima da se fokusiraju isključivo na razvoj aplikacija, dok AWS automatski upravlja infrastrukturom, skaliranjem i resursima.

Serverless arhitektura predstavlja model razvoja aplikacija u kojem se serveri ne upravljaju direktno od strane korisnika, već ih pružalac usluge automatski održava i skalira. Ovaj pristup predstavlja značajan napredak u odnosu na tradicionalne modele, gde je bilo potrebno ručno upravljati serverima, skalirati resurse i održavati sisteme. Pored toga, ovaj koncept smanjuje složenost upravljanja infrastrukturom i operativne troškove jer AWS automatski prilagođava resurse na osnovu zahteva aplikacije. Serverless omogućava brže iteracije, veću agilnost i efikasniju upotrebu resursa, čime se unapređuje poslovna produktivnost i ubrzava proces inovacije.

1.3. AWS Amplify

AWS Amplify [5] je sveobuhvatan set alata i usluga dizajniran za razvoj i implementaciju cloud-podržanih mobilnih i web aplikacija. Pruža različite biblioteke, komponente korisničkog interfejsa, kao i podršku za serversku stranu na AWS platformi. Pored toga, Amplify Hosting omogućava kontinuiranu isporuku i hosting, olakšavajući jednostavan i efikasan proces razvoja.

Ključna prednost AWS Amplify-a je njegova jednostavnost korišćenja, posebno putem CLI alata, koji omogućava timovima da brzo konfigurišu i upravljaju

cloud resursima. Integracija sa drugim AWS servisima omogućava izgradnju složenih aplikacija sa bogatim funkcionalnostima. Takođe, Amplify podržava više platformi, što ga čini fleksibilnim i pogodnim za raznovrsne razvojne timove, obezbeđujući skalabilnost i sigurnost.

1.4. Amazon Cognito

Amazon Cognito [6] je servis koji omogućava upravljanje korisničkim identitetima, autentifikaciju i autorizaciju u okviru web i mobilnih aplikacija. Ovaj servis omogućava bezbedno upravljanje korisničkim podacima, omogućavajući korisnicima da se registruju, prijavljuju i pristupaju različitim resursima unutar aplikacije.

Ključna prednost Amazon Cognita je podrška za različite metode autentifikacije, uključujući prijavu putem društvenih mreža kao što su Facebook i Google, kao i prilagođene opcije verifikacije putem SMS-a i e-pošte, uz podršku za multifaktorsku autentifikaciju (MFA). Ovaj servis automatski upravlja tokenima i korisničkim sesijama, značajno olakšavajući upravljanje identitetima u aplikacijama.

Amazon Cognito se lako integriše sa ostalim AWS servisima, omogućavajući razvoj skalabilnih i sigurnih aplikacija. Takođe, ovaj servis obezbeđuje visok nivo sigurnosti kroz enkripciju podataka tokom prenosa i skladištenja, koristeći TLS protokol i AWS Key Management Service, što osigurava pristup podacima samo ovlašćenim korisnicima i servisima.

1.5 Amazon API Gateway

Amazon API Gateway [7] je potpuno upravljani servis koji omogućava efikasno povezivanje klijentskih aplikacija sa serverskom stranom sistema, podržavajući izgradnju skalabilnih i robusnih aplikacija. API Gateway podržava kreiranje RESTful i WebSocket API-ja, omogućavajući aplikacijama da upravljaju zahtevima u realnom vremenu i osiguraju dvosmernu komunikaciju.

Ključna prednost Amazon API Gateway-a je njegova sposobnost automatskog skaliranja, što omogućava obradu velikog broja simultanih API poziva bez smanjenja performansi aplikacije. Pored toga, servis omogućava keširanje odgovora na API zahteve, poboljšavajući performanse i smanjujući opterećenje na serverskoj stani sistema. Takođe, podržava razvijanje verzionisanje API-ja omogućava razvoj i održavanje različitih verzija istog API-ja bez ugrožavanja postojećih aplikacija.

1.6. AWS Lambda

AWS Lambda [2] je serverless servis koji omogućava pokretanje koda kao odgovor na određene događaje bez potrebe za upravljanjem fizičkom infrastrukturom. Podržava različite programske jezike, kao što su Python, Java, Node.js, C# i Golang, pružajući fleksibilnost u izboru tehnologija. Lambda funkcije mogu biti aktivirane različitim događajima, poput promena u Amazon S3 skladištu, unosa u Amazon DynamoDB bazi podataka ili API zahteva poslatih putem Amazon API Gateway-a.

Jedna od ključnih prednosti AWS Lambda servisa je automatsko skaliranje, koje se prilagođava broju pokrenutih događaja, osiguravajući optimalnu upotrebu

resursa i smanjenje troškova. Naplaćuje se samo vreme stvarnog izvršenja koda, što ga čini ekonomičnim rešenjem za aplikacije sa promenljivim opterećenjem.

AWS Lambda integriše bezbednosne funkcije putem AWS Identity and Access Management (IAM) servisa, omogućavajući precizno upravljanje dozvolama za pokretanje funkcija i pristup drugim AWS resursima. Takođe se lako integriše sa ostalim AWS servisima, čineći ga idealnim za razvoj skalabilnih i fleksibilnih aplikacija bez potrebe za kompleksnom infrastrukturom.

1.7. Amazon DynamoDB

Amazon DynamoDB [3] je serverless NoSQL baza podataka koja pruža brze i predvidljive performanse uz beskonačnu skalabilnost. Specijalizovana je za rad sa nestrukturiranim podacima, omogućavajući skladištenje i pristup velikim količinama podataka bez potrebe za kompleksnim operacijama upravljanja bazom. Struktura podataka u DynamoDB-u je fleksibilna, jer ne zahteva definisanje šeme unapred, što omogućava lako prilagođavanje promenama u aplikacijama.

Jedna od ključnih prednosti DynamoDB-a je automatsko skaliranje, koje dinamički prilagođava kapacitet baze u skladu sa opterećenjem, omogućavajući rukovanje velikim količinama podataka bez prekida u radu ili smanjenja performansi. Takođe, podržava globalne tablice, omogućavajući replikaciju podataka u više AWS regiona za veću dostupnost i otpornost na kvarove.

Sistem je visoko bezbedan, s integrisanim mehanizmima za enkripciju podataka u miru i tokom prenosa, kao i naprednim kontrolama pristupa putem AWS Identity and Access Management (IAM) [6]. DynamoDB omogućava korisnicima da se fokusiraju na razvoj.

1.8. AWS Identity and Access Management

AWS Identity and Access Management (IAM) [6] je servis koji omogućava sigurno upravljanje pristupom AWS resursima, pružajući ključnu kontrolu nad time ko može pristupiti određenim resursima i koje akcije mogu preduzimati. IAM omogućava kreiranje i upravljanje korisničkim nalogima i grupama, definisanje detaljnih pravila pristupa, kao i kontrolu nad dozvoljenim radnjama na različitim AWS resursima.

Jedna od najvažnijih prednosti AWS IAM-a je detaljna kontrola pristupnih prava, koja omogućava da samo ovlašćeni korisnici imaju pristup specifičnim resursima i operacijama. Takođe, IAM podržava multifaktorsku autentifikaciju (MFA), čime se dodatno osigurava pristup osetljivim resursima.

AWS IAM je potpuno integrisan sa svim AWS servisima, što omogućava centralizovano i dosledno upravljanje pristupnim pravima širom cele AWS infrastrukture. Servis takođe omogućava praćenje i evidentiranje svih aktivnosti korisnika i servisa unutar AWS naloga, što doprinosi visokom nivou bezbednosti i transparentnosti.

1.9. AWS CodeCommit

AWS CodeCommit [7] je servis za kontrolu verzija koji omogućava privatno skladištenje i upravljanje Git repozitorijumima unutar AWS oblaka. Dizajniran je da

pruži fleksibilno i sigurno rešenje za kontrolu verzija, omogućavajući razvojnim timovima da efikasno sarađuju na razvoju softvera. CodeCommit podržava sve standardne Git komande, što olakšava integraciju sa postojećim alatima i razvojnim tokovima.

Jedna od ključnih prednosti AWS CodeCommit-a je visoka dostupnost i sigurnost. Podaci su automatski šifrovani tokom prenosa i dok se nalaze u okviru AWS CodeCommit-a, osiguravajući zaštitu od neovlašćenog pristupa. Pored toga, CodeCommit je skalabilan, omogućavajući timovima da upravljaju repozitorijumima bilo koje veličine, što ga čini pogodnim za razvoj kritičnih aplikacija koje zahtevaju pouzdanu i sigurnu platformu za verzionisanje koda.

1.10. Angular radni okvir

Angular [8] je jedan od najpoznatijih radnih okvira koji je razvio Google i koji je postao standard za razvoj web aplikacija. Ovaj okvir omogućava razvoj složenih i interaktivnih korisničkih interfejsa, oslanjajući se na TypeScript, jezik koji proširuje JavaScript dodavanjem tipova i naprednih funkcija.

Angular se ističe svojom modularnom arhitekturom, koja omogućava razvijanje aplikacija u lako održivim i razdvojenim modulima. Njegova značajna karakteristika je dvostrano vezivanje podataka (engl. two-way data binding), koje omogućava sinhronizaciju podataka između modela i prikaza u realnom vremenu, pojednostavljujući upravljanje stanjem aplikacije, posebno u složenim interfejsima.

Pored toga, Angular pruža robustan set alata za upravljanje formama, rutama i komunikacijom sa serverom, što ga čini idealnim za aplikacije koje zahtevaju visoku interaktivnost i brze reakcije na korisničke akcije. Angular CLI dodatno olakšava razvoj, omogućavajući jednostavno kreiranje, konfiguraciju i upravljanje projektima.

Zbog ovih karakteristika, izabrala sam Angular kao radni okvir za razvoj korisničkog interfejsa ovog informacionog sistema. Ovaj izbor omogućio je dodavanje naprednih vizuelnih komponenti, kao što su grafikoni i interaktivni elementi, čime je kreirana moderna i intuitivna web aplikacija.

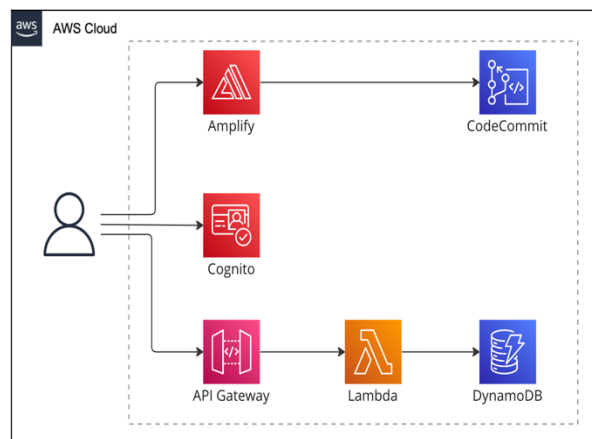
2. SPECIFIKACIJA SISTEMA

Osnovne funkcionalnosti sistema su pristup informacijama neautentifikovanim korisnicima, registracija novih korisnika, prijava i odjava, kao i vizualizaciju finansijskih podataka kroz grafičke prikaze. Sistem pruža grafički prikaz mesečnih i godišnjih transakcija, grupisanih po kategorijama i tipu, uz mogućnost poređenja troškova kroz različite vremenske periode. Takođe, omogućeno je otpremanje CSV fajlova sa bankarskim transakcijama, čime se automatizuje obrada podataka korisnika. Nefunkcionalni zahtevi odnose se na integraciju sa AWS servisima kako bi se obezbedila infrastruktura za skladištenje i obradu podataka. Pored toga, informacioni sistem vrši autentifikaciju putem JWT tokena, verifikaciju naloga putem elektronske pošte, kao i zaštita korisničkih podataka tokom prenosa i u miru. Sam korisnički sistem je

intuitivan, jednostavan za korišćenje, i prilagođen lakom snalaženju kroz aplikaciju.

3. IMPLEMENTACIJA

Informacioni sistem „MoneyDiary” je implementiran bazirajući se na naprednoj serverless arhitekturi zasnovanoj na AWS platformi, što omogućava visoku skalabilnost, pouzdanost i efikasnost u upravljanju finansijskim podacima korisnika (Slika 3.1).



Slika 3.1 Arhitektura sistema

Klijentska aplikacija je razvijena u Angular radnom okviru i hostovana pomoću AWS Amplify servisa, koji pruža kontinuiranu integraciju (CI), automatsku isporuku (CD) i jednostavnu koordinaciju s drugim AWS servisima. AWS Amplify olakšava održavanje aplikacije, omogućavajući automatsko ažuriranje i brzu isporuku novih verzija, čime se poboljšava razvojni proces i smanjuje vreme potrebno za implementaciju.

Upravljanje pristupom i sigurnošću korisničkih naloga realizovano je kroz AWS Cognito, servis koji omogućava registraciju, prijavu i verifikaciju korisnika putem email-a. AWS Cognito omogućava korisnicima sigurno upravljanje nalogima, osiguravajući da su osetljivi podaci zaštićeni na visokom nivou kroz multifaktorsku autentifikaciju (MFA) i šifrovanje podataka. Ovaj servis igra ključnu ulogu u omogućavanju sigurnog pristupa funkcionalnostima aplikacije, tako da samo ovlašćeni korisnici mogu pristupiti podacima i funkcijama unutar sistema.

Glavna poslovna logika aplikacije „MoneyDiary” implementirana je putem AWS Lambda funkcija, koje omogućavaju dinamičku obradu podataka bez potrebe za upravljanjem fizičkom infrastrukturom. AWS Lambda funkcije obrađuju različite aspekte poslovanja, kao što su učitavanje finansijskih podataka iz CSV fajlova, analiza mesečnih troškova i kategorizacija transakcija. Lambda funkcije se automatski skaliraju na osnovu opterećenja, što omogućava sistemu da efikasno odgovara na varijabilne zahteve korisnika, bez opasnosti od zagušenja ili prekida rada.

AWS API Gateway služi kao centralna tačka za komunikaciju između klijentske aplikacije i Lambda funkcija. Ovaj servis omogućava da se HTTP zahtevi iz klijentske aplikacije bezbedno preusmere na odgovarajuće Lambda funkcije, upravljajući autentifikacijom i

autorizacijom korisnika, kao i rutiranjem zahteva. API Gateway pruža visok nivo sigurnosti, omogućavajući filtriranje i kontrolu pristupa prema validnim JWT tokenima koji se izdaju putem AWS Cognito servisa.

Podaci o korisničkim transakcijama smešteni su u AWS DynamoDB, visoko skalabilnoj NoSQL bazi podataka, koja omogućava brz pristup i skladištenje velikih količina nestrukturiranih podataka. DynamoDB se posebno ističe po svojoj sposobnosti da automatski prilagođava kapacitet u skladu sa zahtevima aplikacije, obezbeđujući optimalnu brzinu odgovora čak i pod visokim opterećenjem. Organizacija podataka u DynamoDB omogućava jednostavnu pretragu i filtriranje pomoću primarnih i sekundarnih ključeva, što ubrzava obradu složenih upita, kao što su pretrage po vremenskim periodima i korisnicima.

AWS Identity and Access Management (IAM) koristi se za kontrolu pristupa svim AWS servisima unutar sistema „MoneyDiary”. IAM omogućava precizno definisanje prava pristupa i operacija koje korisnici i servisi mogu izvršavati, primenjujući princip najmanjih privilegija, čime se obezbeđuje visok nivo sigurnosti. Ova kontrola pristupa osigurava da samo ovlašćeni korisnici i servisi mogu pristupiti ključnim resursima, čime se smanjuje rizik od neovlašćenih promena ili zloupotreba podataka.

Praćenje performansi sistema i logova realizovano je putem AWS CloudWatch-a, koji omogućava detaljan uvid u rad infrastrukture i aplikacija u realnom vremenu. CloudWatch prikuplja i analizira podatke o performansama sistema, omogućavajući timovima brzo identifikovanje i rešavanje potencijalnih problema, kao i optimizaciju resursa kako bi sistem ostao visoko efikasan i pouzdan.

4. ZAKLJUČAK

U ovom radu prikazan je razvoj informacionog sistema "MoneyDiary" za upravljanje ličnim finansijama, zasnovanog na AWS serverless arhitekturi. Opisan je način na koji sistem koristi servise poput AWS Lambda, API Gateway i DynamoDB za dinamičku obradu podataka, kao i integracija sa AWS Cognito za upravljanje korisničkom autentifikacijom. Takođe, obuhvaćeno je hostovanje klijentske aplikacije razvijene u Angular okviru putem AWS Amplify servisa.

Sistem omogućava obradu i vizualizaciju finansijskih podataka korisnika, uz visok nivo sigurnosti i pouzdanosti. Planovi za potencijalno unapređenje sistema uključuju integraciju algoritama za mašinsko učenje radi predviđanja finansijskih trendova i proširenje sistema na višejezične interfejsse kako bi se povećala globalna dostupnost. Pored toga, moguća unapređenja uključuju i proširenje integracije sa dodatnim finansijskim servisima i uvođenje naprednih metoda analize podataka.

Iako oslanjanje na AWS nosi potencijalne rizike, poput povremenih prekida u dostupnosti, važno je napomenuti da je AWS vodeći cloud provajder sa visokim standardima za sigurnost i dostupnost servisa.

5. LITERATURA

- [1] Sbarski, P. and Kroonenburg, 2017. Serverless Architectures on AWS: With Examples Using AWS Lambda
- [2] AWS Lambda
<https://aws.amazon.com/lambda/>
- [3] Amazon DynamoDB
<https://aws.amazon.com/dynamodb/resources/>
- [4] AWS Documentation
https://docs.aws.amazon.com/?nc2=h_ql_doc_do
- [5] AWS Amplify
<https://aws.amazon.com/amplify/getting-started/?nc=sn&loc=5>
- [6] AWS Identity and Access Management
<https://aws.amazon.com/iam/getting-started/?nc=sn&loc=3>
- [7] AWS CodeCommit
<https://www.google.com/search?client=safari&rls=en&q=AWS+CodeCommit&ie=UTF-8&oe=UTF-8>
- [8] Angular
<https://angular.io>

Kratka biografija:



Sanja Drnić je rođena 10. novembra 1999. godine u Novom Sadu. Školske 2018/2019. godine upisuje osnovne studije na Fakultetu tehničkih nauka u Novom Sadu na smeru Računarstvo i automatika. Nakon završetka osnovnih studija, na istom fakultetu upisuje master studije na studijskom programu Primenjene računarske nauke i informatika - Elektronsko poslovanje. Položila je sve ispite propisane planom i programom.