



SISTEM ZA AUTOMATIZACIJU PROCESA EVIDENCIJE LIČNIH TROŠKOVA
SYSTEM FOR AUTOMATION OF PERSONAL EXPENSE MANAGEMENT

Tamara Luković, *Fakultet tehničkih nauka, Novi Sad*

Oblast – INFORMACIONO – KOMUNIKACIONI SISTEMI

Kratak sadržaj – Vođenje evidencije ličnih troškova veliki broj ljudi smatra gubitkom vremena. Međutim, ukoliko bi taj proces bio automatizovan, vreme za unos i analizu troškova bilo bi znatno smanjeno. Evidentiranje troškova ima velikog značaja prilikom kreiranja budžeta kao i prilikom analize za redukciju potrošnje. Takođe vođenje evidencije pruža transparentnost i uvid u stanje uštedevine po valuti. Na ovaj način korisnici mogu da lakše analiziraju mesečnu potrošnju sa ciljem smanjenja nepredviđenih troškova i planiranjem budžeta u narednom periodu za takve situacije. U ovom radu su opisane tehnologije koje su korišćene za izradu web aplikacije kao i struktura same aplikacije. Predstavljene su ključne stranice i opisane funkcionalnosti. Takođe u zaključku su definisani i naredni koraci unapredjenja.

Ključne reči: ASP.NET, MVC, PostgreSQL, NHibernate, Web aplikacija

Abstract – Keeping a record of personal expenses is considered by many people a waste of time. However, if the process was automated, the time to enter and analyze costs would be significantly reduced. Cost monitoring is of great importance when creating a budget as well as when analyzing to reduce spending. Also, keeping a record of costs provides transparency and insight into savings by currency. In this way, users can more easily analyze their monthly spending with the aim of reducing unforeseen costs and improve planning for the next period for such unpleasant situations. This paper describes the technologies used to create the web application as well as the structure of the application itself. Key views and functionalities are described. Also, the next steps for improvement are defined in the conclusion.

Keywords: ASP.NET, MVC, PostgreSQL, NHibernate, Web application

1. UVOD

Sistemi poput organizacija, kompanija pa čak i država upravljaju velikom količinom različitih i u značajnoj meri neizvesnih informacija od kojih su najzastupljenije informacije o finansijskom sektoru i stanju. Bilansi uspeha i bilansi stanja pomažu menadžmentu da upravlja takvim sistemima, međutim, ukoliko informacije nisu validne i ažurne, gube na vrednosti i otežavaju upravljanje.

NAPOMENA:

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Srđan Sladojević.

Lične finansije, korporativne finansije, javne finansije, kapital, sve su to izrazi koje čujemo na dnevnom nivou jer one predstavljaju veoma interesantnu temu u poslovnom i privatnom svetu.

Menadžment finansija je oduvek bila zanimljiva tema i kroz istoriju se provlačila u različitim oblicima. Najčešće se odnosila na finansiranje vojske i država dok danas ona predstavlja jednu od pokretačkih osnova savremenog društva. Kako je osnovni cilj korporacija ostvarivanje dobiti, upravljanje finansijama predstavlja ozbiljan i težak zadatak za menadžment. Finansijsko upravljanje treba da obezbedi realizaciju finansijskih ciljeva preduzeća [1].

Organizacija ličnih finansija se prvi put pojavljuje u Americi pre oko 40 godina, nakon što su potrošači nakon dužeg perioda prosperiteta od drugog svetskog rata, počeli da raspoložu većim količinama sredstava [2]. Od tada se broj zainteresovanih pojedinaca povećava a usponom digitalnog doba većina svoje lične troškove beleži korišćenjem tehnologije i dostupnih alata.

Vodjenje evidencije o potrošnji pojedincima može da unapredi proces planiranja osnovnih i neplaniranih mesečnih troškova, bas kao što velikim preduzećima olakšava planiranje i kontrolu ulaganja. Planiranje osnovnih troškova obezbeđuje pojedincima sigurnost, zbog čega velik broj ljudi pribegava toj metodi dok kontrola i ulaganje u fond nepredviđene troškove smanjuje neizvesnost nepovoljnih okolnosti.

Najčešći vid praćenja troškova do skoro je bio korišćenjem nekog programa za tabelarne kalkulacije kao što je "Microsoft Excel", međutim, poslednjih godina na značaju dobijaju web aplikacije i slične moderne metode kontrole troškova.

2. PRIMENJENE SOFTVERSKJE TEHNOLOGIJE

Ovo poglavlje opisuje tehnologije koje su se koristile za izradu opisane web aplikacije:

PostgreSQL predstavlja objektno-relacioni sistem za upravljanje bazom podataka koji je nastao kao projekat Kalifornijskog univerziteta "Berkley". Profesor "Michael Stonebraker" je pokrenuo projekat davne 1986-te godine [4].

Postgres je originalno koristio jezik koji se zvao PostQUEL za pristup informacijama iz baze, međutim, 1994-te godine su Andrew Yu i Jolly Chen dodali

Postgres SQL interpreter koji se originalno zvao Postgres95. Postgres95 je kasnije relicenciran pod Berkley softverskom licencom i neposredno nakon toga preimenovan u PostgreSQL [4].

PostgreSQL baza podataka je najnaprednija *open source* baza podataka sa mnogo prednosti od kojih je najznačajnija ta da je ona otvorenog tipa. Ova baza podataka nije u vlasništvu ni jedne kompanije, besplatna je i svako ima pristup *source code-u*. Ova baza podataka omogućava korisnicima da kreiraju svoje posebne tipove podataka, a i u osnovi nudi veći broj tipova podataka od kojih su značajniji geometrijski tipovi podataka poput tačke, linije, segmenta [3]. Takođe jedna od prednosti ove baze podataka jeste ta da sa serverske strane nudi moćan proceduralni jezik *PL/pgSQL* [3].

NHibernate predstavlja *open-source* rešenje objektno-relacionog mapiranja za Microsoft .NET platformu. NHibernate je deo Hibernate Core-a za Javu [7]. On upravlja postojećim običnim .NET objektima do i iz osnovne relacijske baze podataka. Ima ugrađenu mogućnost za automatskim generisanjem objekata na osnovu XML opisa veza između entiteta [7]. Objektno-relaciono mapiranje poput NHibernate-a predstavlja poveznicu između relacionih modela baze podataka i objektnih modela klasa u aplikaciji [5]. Takođe se može reći da on predstavlja "virtuelnu reprezentaciju" objekata baze u kodu [7].

Cilj objektno-relacionog modela jeste da eliminiše duplirane slojeve između korisnika i baze zajedno sa povećanom cenom održavanja uz dodatno eliminisanje potencijalnih grešaka koje mogu da proisteknu iz dizajna [6].

Za potrebe ovog rada korišćen je *.hbm.xml* koji uz pomoć ugrađenih xml atributa mapira podatke iz baze na klase modela. Takođe, kreiran je samo jedan xml fajl koji služi kao mapping osnova za sve klase.

.NET Framework predstavlja razvojni okvir koji je razvijen od strane *Microsoft-a* koji se pokreće na Microsoft Windows operativnom sistemu. On uključuje opširnu biblioteku koda koja se zove *Framework class library* (FCL) i omogućava interoperabilnost između programskih jezika koje podržava. Biblioteke koda predstavljaju kolekcije resursa kao što su klase, metode, procedure, skripte koje omogućavaju programerima da skrate vreme razvoja time što nudi već kreirane i temeljno testirane module.

ASP.NET predstavlja *open-source* servisno orijentisan *framework* dizajniran i razvijen od strane Microsoft-a koji omogućava programerima da razvijaju dinamičke *web* sajtove, aplikacije i servise. Ime ASP.NET je Microsoft dao kombinaciji dve svoje tehnologije: *web forms* i *web services* [8]. Korišćenjem ove tehnologije, danas je jednostavnije kreiranje dinamičnih *web* aplikacija koje se lako pokreću na svim *browser-ima* [8].

Ključne karakteristike *asp.net* tehnologije:

- ❖ Asp.net razdvaja logiku aplikacije od HTML-a
- ❖ Asp.net kod se kompajlira, a ne interpretira

Konfiguracija i *deployment* su pojednostavljeni.

MVC predstavlja arhitekturni obrazac koji razdvaja aplikaciju na tri logičke komponente a to su *model*, *view* i *controller*. Svaka komponenta se izgrađuje sa ciljem da održava specifičan razvojni aspekt aplikacije i u potpunosti razdvaja logiku od UI segmenta. Ovaj obrazac je jedan od najčešće korišćenih obrazaca prilikom razvoja *web* aplikacija.

Model predstavlja set klasa koji opisuje biznis logiku i pravila po kojima se podaci mogu menjati.

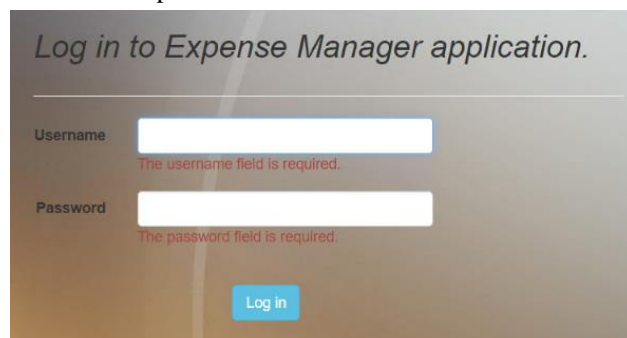
View prikazuje podatke iz modela u formatu koji je pogodan za interakciju između aplikacije i korisnika.

Controller komponenta se ponaša kao interfejs između modela i *view-a* koja ima za cilj da procesuiru svu biznis logiku i pristigle zahteve, manipuliše podacima korišćenjem *model* komponente i njenih karakteristika.

3. FUNKCIONALNOSTI

Svrha izade ove aplikacije jeste mogućnost beleženja i analize individualne potrošnje. Korisnik aplikacije ima mogućnost da evidentira sve svoje prilike i odlive, generiše statistiku potrošnje za analizu, vrši exportovanje podataka u json format kao i da vodi evidenciju o dugovanima.

Kako bi korisnik mogao da pristupi podacima iz baze, prvo mora da se uloguje na aplikaciju sa admin kredencijalima. Na slici 1 je prikazana login forma sa validacionim porukama.



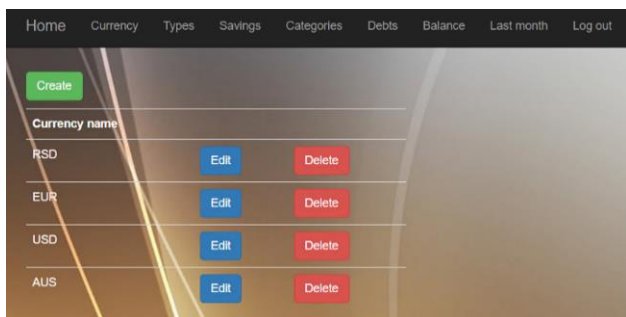
Slika 1. Login forma sa validacionim porukama

Nakon uspešnog logovanja, korisnik ima mogućnost da vidi početnu stranicu sa navigacionim bar-om gde su prikazane forme kojima može da pristupi (slika 2).



Slika 2. Početna stranica

Currency tab omogućuje pregled, dodavanje i uklanjanje valuta koje korisnik upiše u bazu (slika 3).



Slika 3 Prikaz svih valuta

Types tab kao i valuta, prikazuje samo naziv tipa koji se koristi da upiše kategorije koje korisnik može da unese u bazu.

Categories tab opisuje kategorije koje korisnik može da doda kao što su kategorija „obrazovanje“, „šoping“, „plata“. Pored naziva kategorije korisnik vrši odabir tip kategorije a to mogu biti „trošak“ ili „prihod“.

Savings tab (slika 4) nema opcije za dodavanje novog *item*-a u bazu kao ni editovanje ni brisanje već postojeg. CRUD operacije se nad ovom tabelom izvršavaju CRUD operacijama sledećih tabela:

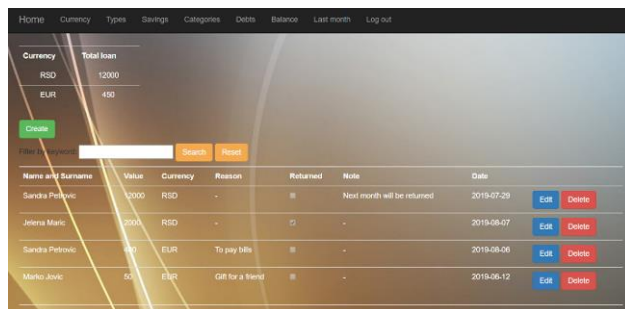
- ❖ Dodavanje nove valute kreira item u ovoj tabeli i postavlja vrednost uštedevine na nula
- ❖ Kreiranje novog item-a u balans tabeli vrši update ove tabele tako da ukoliko je tip kategorije “Expense” vrši se oduzimanje vrednosti dok se za “Income” tip vrši povećavanje vrednosti
- ❖ Brisanje poslednjeg item-a balans tabele za određenu valutu uklanja informaciju o uštedevini za tu valutu

Value	Currency name
0	USD
0	AUS
70	EUR
209765	RSD

Slika 4. Prikaz Savings forme

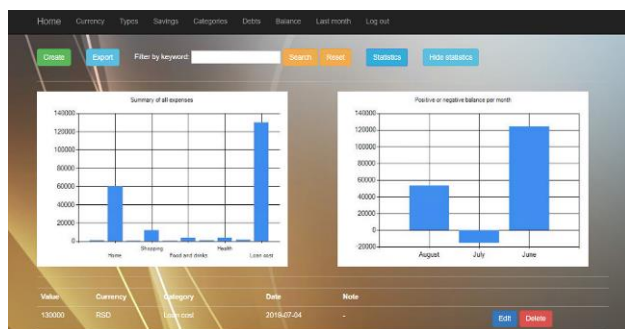
Debts tab prikazuje sva izmirena i neizmirena dugovanja prema korisniku aplikacije (slika 5).

Pored svih dugovanja, ova stranica prikazuje i sumu svih nedospelih dugova po valuti. Ova tabela se automatski popunjava krairanjem novih item-a u tabeli sa dugovanjima. Takođe ova forma nudi i mogućnost filtriranja rekorda po ključnoj reči bilo koje kolone.



Slika 5. Prikaz Debts forme

Balance tab predstavlja kolekciju najznačajnijih podataka u bazi. Ova tabela predstavlja svrhu kreiranja aplikacije. Forma za prikaz podataka omogućava uvid u sve item-e iz baze podataka, operacije za manipulaciju podataka kao što su kreiranje novog rekorda, izmena postojećeg ili brisanje podataka. Takođe nudi opciju za pretragu po ključnoj reči, export podataka u .json format i klikom na „Statistics“ dugme otvara sekciju za prikaz statistike (slika 6).



Slika 6. Prikaz Balance forme a expand-ovanom statistikom

Levi grafik predstavlja sumu svih troškova i odnos potrošnje po kategorijama dok grafik sa desne strane prikazuje izlazno stanje poslednja tri meseca.

Last month tab po uzoru na Balance tab pruža podatke o svim rekordima u balance tabeli za poslednji mesec. Takođe ima opciju za export podataka u .json format, filtriranje podataka po ključnoj reči kao i prikaz statistike stanja za tekući mesec do momenta generisanja upita (slika 7).

Slika 7. Last month tab

4. ZAKLJUČAK

U ovom radu predstavljen je sistem za automatizaciju menadžmenta ličnih troškova sa osvrtom na korišćene tehnologije. Sistem predstavlja podršku upravljanja budžetom korisnika aplikacije kao i mogućnost za analizu potrošnje kroz pregled statistike i pregled troškova za tekući mesec. Statistike koje se generišu prilikom otvaranja forme za prikaz svih prihoda i rashoda kao i prihode i rashode za tekući mesec, omogućavaju uvid u stanje potrošnje po kategorijama.

Na taj način korisnik može da utvrdi gde postoji prostora za smanjenjem troškova i da li je stanje uštedevine na optimalnom nivou.

Ono što takodje predstavlja važan segmen ove aplikacije jesu podaci o dugovanjima prema korisniku budući da svi nekad u životu pozajme novac prijateljima, porodici. Na ovaj način možemo dobiti jasnu sliku visini nenaplaćenog iznosa.

Dalji pravci razvoja ovakve aplikacije se mogu jasno utvrditi a to su:

- ❖ Implementiranje multikorisničkog okruženja. Omogućavanje korisnicima da kreiraju naloge koje će povezati sa Gmail nalogom i na koje bi im stizala generisana statistika potrošnje za svaki mesec
- ❖ Mogućnost importovanja objekata balans tabele u .json format
- ❖ Kreiranje mobilne aplikacije za brži pristup podacima i statistikama
- ❖ Mogućnost pristupa podacima iz baze za odabrani period koji korisnik unosi odabirom datuma kroz dropdown listu

5. LITERATURA

- [1] dr Marko, Ivaniš. ",Upravljanje finansijama ". " *Univerzitet Singidunum, Beograd* (2008).
- [2] Stojanović, Gordana. "Standardizovani zahtevi za planere ličnih finansija." *Bankarstvo* 35.5-6 (2006): 54-61

- [3] Douglas, Korry, and Susan Douglas. *PostgreSQL: a comprehensive guide to building, programming, and administering PostgreSQL databases*. SAMS publishing, 2003.
- [4] Conrad, Tim. "Postgresql vs. mysql vs. commercial databases: It's all about what you need." (2006).
- [5] Dentler, Jason. *NHibernate 3.0 Cookbook*. Packt Publishing Ltd, 2010
- [6] G ruca, Aleksandra, and Przemysław Podsiadło. "Performance Analysis of .NET Based Object-Relational Mapping Frameworks." *International Conference: Beyond Databases, Architectures and Structures*. Springer, Cham, 2014.
- [7] Cure, Aaron. *NHibernate 2 Beginner's Guide*. Packt Publishing Ltd, 2010.
- [8] Liberty, Jesse, and Dan Hurwitz. *Programming Asp. Net*. " O'Reilly Media, Inc.", 2003.

Kratka biografija:



Tamara Luković rođena je 1993. god. u Novom Sadu. Gimnaziju "Isidora Sekulić" završava 2012. godine nakon čega upisuje osnovne studije u oblasti Inženjerskog menadžmenta na Fakultetu tehničkih nauka u Novom Sadu. Nakon završenih osnovnih akademskih studija upisuje master akademske studije takođe na Fakultetu tehničkih nauka, smer Inženjerstvo informacionih sistema.