



Академија струковних
студија Шумадија
Одсек Крагујевац

Studijski program: Informatika
Predmet: Programski jezici

ИНФОРМАЦИОНИ СИСТЕМ ЗА УПРАВЉАЊЕ ПРОИЗВОДЊОМ

-Трансфера -

Predmetni nastavnik:
Проф. Саша Стаменовић

Студент:
Богдан Видојевић, 008/2023

Kragujevac 2025.

Sadržaj

ИНФОРМАЦИОНИ СИСТЕМ ЗА УПРАВЉАЊЕ ПРОИЗВОДЊОМ	0
Sadržaj	1
1. Uvod	2
2. ТЕХНОЛОГИЈЕ И АЛАТИ	3
Delphi и FireMonkey (FMX)	3
SQLite база података	3
FireDAC	3
3. АРХИТЕКТУРА АПЛИКАЦИЈЕ	3
4. ФУНКЦИОНАЛНОСТИ АПЛИКАЦИЈЕ	4
Додавање новог рачуна	4
Историја плаћања	5
5. ТЕХНИЧКЕ СПЕЦИФИЧНОСТИ	5
ЗАКЉУЧАК	6

1. Uvod

Развој мобилних апликација данас представља кључну област информационих технологија јер омогућава корисницима да једноставно и интуитивно обављају различите свакодневне задатке. Посебан значај имају апликације које олакшавају управљање финансијама и плаћање рачуна. Традиционални начин плаћања подразумева одлазак у банку или коришћење веб портала, док мобилне апликације нуде брже, сигурније и практичније решење доступно на сваком уређају.

Овај семинарски рад описује израду мобилне апликације развијене у окружењу **Delphi FireMonkey (FMX)**, са интегрисаном базом података **SQLite** и коришћењем **FireDAC** библиотеке за комуникацију са базом. Апликација омогућава додавање рачуна за плаћање, преглед свих сачуваних рачуна, евидентирање извршених уплата и преглед историје плаћених рачуна. Посебна пажња посвећена је једноставном и прегледном корисничком интерфејсу, уз коришћење модерних компоненти FMX-а.

2. ТЕХНОЛОГИЈЕ И АЛАТИ

Delphi и FireMonkey (FMX)

Delphi је савремено развојно окружење за израду десктоп, мобилних и веб апликација. Користи **Object Pascal** програмски језик, познат по стабилности и једноставној синтакси.

FireMonkey (FMX) је визуелни оквир за израду апликација које могу радити на различитим платформама (Windows, macOS, Android, iOS). Омогућава дизајн модерних корисничких интерфејса са прилагођеним контролама и ефектима.

SQLite база података

SQLite је једноставна, брза и самостална релациона база података, идеална за мобилне апликације јер не захтева инсталацију сервера. Подаци се чувају у једном фајлу, што олакшава имплементацију и дистрибуцију апликације.

FireDAC

FireDAC је моћна библиотека која у Delphi окружењу обезбеђује повезивање са различитим базама података. У овој апликацији користи се FireDAC драјвер за SQLite како би се обезбедила брза комуникација и лако извршавање SQL упита.

3. АРХИТЕКТУРА АПЛИКАЦИЈЕ

Апликација је организована у више екрана (табова), који омогућавају логично раздвајање функционалности и лакше сналажење корисника.

Почетни екран (Home Screen) – приказује основне информације о кориснику и баланс рачуна.

Додавање рачуна (Додај рачун) – формулар за унос података о новом рачуну који корисник жели да сачува.

Сви рачуни – листа свих рачуна сачуваних у бази, приказаних у облику модерних картица.

Детаљи рачуна и плаћање – приказ детаља одабраног рачуна са опцијама да се изврши плаћање или да се корисник врати назад.

Подаци се чувају у **SQLite бази** која има две основне табеле:

`racuni` – чува основне податке о рачунима:

`id` (јединствени идентификатор),

`svrha_uplate` (сврха уплате),

`primalac` (име примаоца),

broj_racuna (број рачуна примаоца),
poziv_na_broj (позив на број уплате),
ime_racuna_u_app (назив који корисник види у апликацији),
iznos (износ који се плаћа),
kreirano (датум и време креирања).

placanja – евидентира сваку извршену уплату:

- o id (јединствени идентификатор уплате),
- o racuni_id (страни кључ ка табели racuni),
- o iznos (износ уплате),
- o datum_uplate (датум када је уплата извршена),
- o kreirano (време када је унос евидентиран).

Повезаност табела је реализована преко **страног кључа** `racuni_id` у табели `placanja`, што омогућава праћење уплата за сваки конкретан рачун.

4. ФУНКЦИОНАЛНОСТИ АПЛИКАЦИЈЕ

Додавање новог рачуна

Корисник може унети све потребне податке за нови рачун:

сврха уплате,

прималац,

број рачуна,

позив на број,

износ,

назив рачуна који ће се приказивати у апликацији.

Кликом на дугме „**Сачувај рачун**“ подаци се валидирају (нпр. проверава се да ли је број рачуна правилно унета и да ли су сва обавезна поља попуњена). Након тога се извршава **SQL INSERT** упит који податке уписује у табелу `racuni`.

Приказ свих рачуна

Сви сачувани рачуни приказују се у посебном табу у облику картица. Свака картица садржи назив рачуна и износ који је везан за њега. Приказ је дизајниран тако да буде прегледан и визуелно модеран, уз употребу **RoundRect** компоненти и прилагођених боја и маргина.

Листа рачуна се динамички освежава сваки пут када корисник дода нови рачун или се врати на овај екран.

Детаљан приказ рачуна и плаћање

Кликом на постојећи рачун отвара се екран са детаљима. Ту су приказана сва поља која су унета приликом додавања (прималац, сврха, број рачуна, позив на број, износ и назив). На дну се налазе два дугмета:

Плати рачун – евидентира уплату у табели `placanja` и враћа корисника на листу свих рачуна.

Назад – враћа корисника без извршавања било какве акције.

Када корисник потврди плаћање, апликација уписује нови ред у табелу `placanja` помоћу SQL **INSERT** упита. Након тога се листа уплата и приказ историје освежавају у реалном времену.

Историја плаћања

На почетном екрану налазе се две зоне (`rHistory` и `rHistory2`) које приказују листу извршених уплата. За сваку уплату приказује се:

име рачуна (назив који је корисник унео),

износ уплате,

датум када је уплата извршена.

Позадина сваког реда дизајнирана је у модерном стилу са благо заобљеним ивицама и прилагођеном бојом, како би историја била прегледна и естетски пријатна.

Листа се освежава аутоматски приликом покретања апликације и након сваке нове уплате.

5. ТЕХНИЧКЕ СПЕЦИФИЧНОСТИ

FireDAC и SQLite: коришћен је `TFDConnection` за повезивање са базом и `TFDQuery` за извршавање SQL наредби.

Валидација уноса: имплементиране су функције за проверу броја рачуна, филтрирање недозвољених карактера и нормализацију уноса.

Динамичко креирање UI елемената: картице рачуна и историја плаћања генеришу се програматски, што омогућава флексибилно прилагођавање приказа.

Подршка за виšekратно освежавање: након сваке измене (додавање или плаћање) UI се одмах ажурира и приказује ново стање базе.

Локална база: сви подаци се чувају у једном .db фајлу, што чини апликацију лаганом и једноставном за пренос.

ЗАКЉУЧАК

Развијена мобилна апликација демонстрира како уз коришћење **Delphi FireMonkey окружења**, **FireDAC** компоненте и **SQLite** базе може да се креира функционално и прегледно решење за вођење личних финансија и евиденцију рачуна. Апликација је модуларна, проширива и погодна за даљи развој.

Потенцијална унапређења укључују:

додавање филтрирања и претраге рачуна по имену или primaocu,
имплементацију нотификација за подсетник на плаћање,
могућност синхронизације са сервером или cloud базом,
извоз података о плаћањима у CSV/PDF.

Овај пројекат представља пример практичне примене програмирања мобилних апликација са локалним складиштењем података, уз јасну архитектуру, једноставан UI и поуздану манипулацију подацима. Коришћењем FireMonkey оквира омогућено је да апликација буде преносива на различите платформе, док је SQLite обезбедио брз и једноставан начин рада са подацима без потребе за додатним сервером.