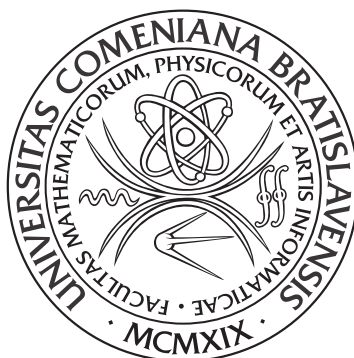


UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY



SERVEROVÁ APLIKÁCIA PRE MOBILNÉHO SPRIEVODCU FMFI
BAKALÁRSKA PRÁCA

2018

Marianna Ráchelová

UNIVERZITA KOMENSKÉHO V BRATISLAVE
FAKULTA MATEMATIKY, FYZIKY A INFORMATIKY

SERVEROVÁ APLIKÁCIA PRE MOBILNÉHO SPRIEVODCU FMFI

BAKALÁRSKA PRÁCA

MARIANNA RÁCHELOVÁ

Študijný program:	Aplikovaná informatika
Študijný odbor:	2511 Aplikovaná informatika
Školiace pracovisko:	Katedra aplikovanej informatiky
Školiteľ:	Ing. Alexander Šimko. PhD.

Bratislava 2018

Marianna Ráchelová



ZADANIE ZÁVEREČNEJ PRÁCE

Meno a priezvisko študenta: Marianna Ráchelová
Študijný program: aplikovaná informatika (Jednoodborové štúdium, bakalársky I. st., denná forma)
Študijný odbor: aplikovaná informatika
Typ záverečnej práce: bakalárska
Jazyk záverečnej práce: slovenský
Sekundárny jazyk: anglický

Názov: Servérová aplikácia pre mobilného sprievodcu FMFI
Server application for mobile guide to FMFI

Anotácia: Cieľom práce je vytvoriť serverovú aplikáciu pre mobilného sprievodcu FMFI:
- Serverová aplikácia bude centrálnym zdrojom dát pre mobilnú aplikáciu — mapu fakulty, zoznam miestností, zamestnancov, jedálne lístky a pod. Serverová aplikácia bude vedieť získavať dáta z externých zdrojov, ako napr. z candle, ais, freefood.sk a pod. Pre dáta, pre ktoré neexistuje externý zdroj, bude obsahovať administrátorský panel, cez ktorý sa budú dáta upravovať. Návrh musí byť modulárny, aby sa dalo ľahko zameniť ručné zadávanie údajov za programový modul získavajúci dáta z externého zdroja.
- Serverová aplikácia bude podporovať používateľské kontá, ktoré budú mať nastavené prístupové práva na editovanie jednotlivých dát.
- Mobilná aplikácia si bude zo serverovej aplikácie aktualizovať svoju databázu. Aktualizovanie databázy musí byť efektívne z pohľadu množstva prenesených dát, t.j. prenášať sa majú len zmeny dát a nie celá databáza. Serverová aplikácia sa taktiež musí vedieť vysporiadať s tým, že nie každý bude mať mobilnú aplikáciu aktualizovanú na najnovšiu verziu, a tým pádom môžu mobilné aplikácie potrebovať dáta v starších modeloch dát.
Súčasťou práce je integrovanie existujúceho mobilného sprievodcu FMFI so serverovou aplikáciou.

Vedúci: Ing. Alexander Šimko, PhD.
Katedra: FMFI.KAI - Katedra aplikovanej informatiky
Vedúci katedry: prof. Ing. Igor Farkaš, Dr.
Dátum zadania: 05.10.2017

Dátum schválenia: 09.10.2017

doc. RNDr. Damas Gruska, PhD.
garant študijného programu

študent

vedúci práce

Čestné prehlásenie

Čestne prehlasujem, že bakalársku prácu s názvom: *Serverová aplikácia pre mobilného sprievodcu FMFI* som vypracovala samostatne, na základe konzultácií, teoretických poznatkov, praktických poznatkov a štúdia odbornej literatúry. Neporušila som autorský zákon a zoznam použitej literatúry som uviedla na príslušnom mieste.

Pod'akovanie

Abstrakt

Cieľom bakalárskej práce je vytvoriť serverovú aplikáciu pre mobilného sprievodcu FMFI , implementovať prepojenie mobilnej a serverovej aplikácie a vytvorenie administrátorského rozhrania pre lepšiu manipuláciu s dátami. Serverová aplikácia spravuje dáta, komunikuje s mobilnou aplikáciou a zefektívňuje prácu mobilnej aplikácie.

Abstract

The aim of the bachelor thesis is to create a server application for the FMFI mobile wizard, to implement a mobile and server application interface and to create an admin interface for better data manipulation. The server application manages the data, communicates with the mobile application and makes the mobile application work more efficient.

Slovník pojmů

TODO

Obsah

KAPITOLA 1.....	11
ÚVOD	11
KAPITOLA 2.....	12
TEORETICKÉ VÝCHODISKÁ.....	12
2.1 Serverová aplikácia.....	12
2.2 Databáza.....	13
2.3 Aplikačný server	14
2.4 Webový server	14
2.5 Webové servisy typu REST	14
2.6 Technológie.....	14
2.6.1 Programovací jazyk Java.....	14
2.6.2 Java Enterprise Edition	16
2.6.3 PostgreSQL databáza.....	17
2.6.4 Aplikačný server Payara	18
2.6.5 Apache Maven.....	18
2.6.6 Vaadin.....	19
2.6.7 Zhrnutie technológií	19
2.8 Mobilný sprievodca FMFI	19
2.9 Návrh serverovej aplikácie a jej databázy.....	20
2.9.1 Model serverovej aplikácie.....	20
2.9.2 Databázový model	21
ZDROJE.....	22

Zoznam obrázkov

Obrázok 1: Softvérová aplikácia	13
Obrázok 2: Java EE server and containers	16
Obrázok 3: Postgres logo	17
Obrázok 4: PostgreSQL logo	17
Obrázok 5: PgAdmin	17
Obrázok 6: Administrátorské rozhranie aplikačného serveru Payara	18
Obrázok 7: Návrh serverovej aplikácie	21

Kapitola 1

Úvod

Fakulta matematiky, fyziky a informatiky disponuje mobilným sprievodcom, ktorý má slúžiť študentom pre lepšiu orientáciu v budove fakulty. Mobilná aplikácia ponúka tiež vyhľadávanie optimálnej cesty z jednej časti fakulty do druhej časti fakulty. Mobilná aplikácia nespravuje dáta konzistentne, a preto v rámci mojej bakalárskej práce sa budem venovať návrhu a implementácií serverovej aplikácie k mobilnej aplikácii a tiež integrovaniu týchto dvoch aplikácií.

Dáta, s ktorými pracuje serverová aplikácia je zoznam zamestnancov fakulty a informácie o zamestnancoch, informácie o rozvrhu jednotlivých miestností a učiteľov a informácie o rozpise denného menu v jednotlivých jedálňach fakulty.

Serverová aplikácia získava dáta prostredníctvom webových servisov z iných zdrojov a ukladá ich do svojej databázy. Tieto dáta bude poskytovať mobilnej aplikácii. Komunikácia bude medzi aplikáciami prebiehať v štýle požiadavka-odpoveď, to znamená, že príde požiadavka na dáta z mobilnej aplikácie a serverová aplikácia poskytne odpoveď s dátami mobilnej aplikácii.

Súčasťou bakalárskej práce je aj administrátorské rozhranie serverovej aplikácie. Administrátorské rozhranie je webová aplikácia, ktorá obsahuje prihlásenie pre používateľa a umožňuje správu dát. Pod správou dát rozumieme pridávanie nových dát, mazanie dát a úpravu už existujúcich dát v databáze serverovej aplikácie. Do administrátorského rozhrania je možné sa prihlásiť pod viacerými rolami. Týmito rolami sú napríklad admin a bežný používateľ. Rozhranie bude tiež umožňovať pridávanie novej mapy fakulty.

Nasledujúce kapitoly vysvetľujú, čo je serverová aplikácia, webové servisy, databáza a ako to spolu súvisí. Ďalej popisuje jednotlivé časti serverovej aplikácie, fázy implementácie serverovej aplikácie a integrácie s mobilnou aplikáciou. Bakalárska práca je zakončená záverom.

Kapitola 2

Teoretické východiská

V rámci kapitoly teoretické východiská vysvetlím pojmy čo je serverová aplikácia, databáza, aplikačný server, webový server a REST-ové webové servisy. Ďalej bližšie spomeniem jednotlivé technológie používané pri vývoji serverovej aplikácie, a prečo som zvolila práve tieto technológie. Stručne popíšem už existujúcu mobilnú aplikáciu, ku ktorej budem v závere mojej bakalárskej práce integrovať serverovú aplikáciu a stručne predstavím návrh mojej serverovej aplikácie.

2.1 Serverová aplikácia

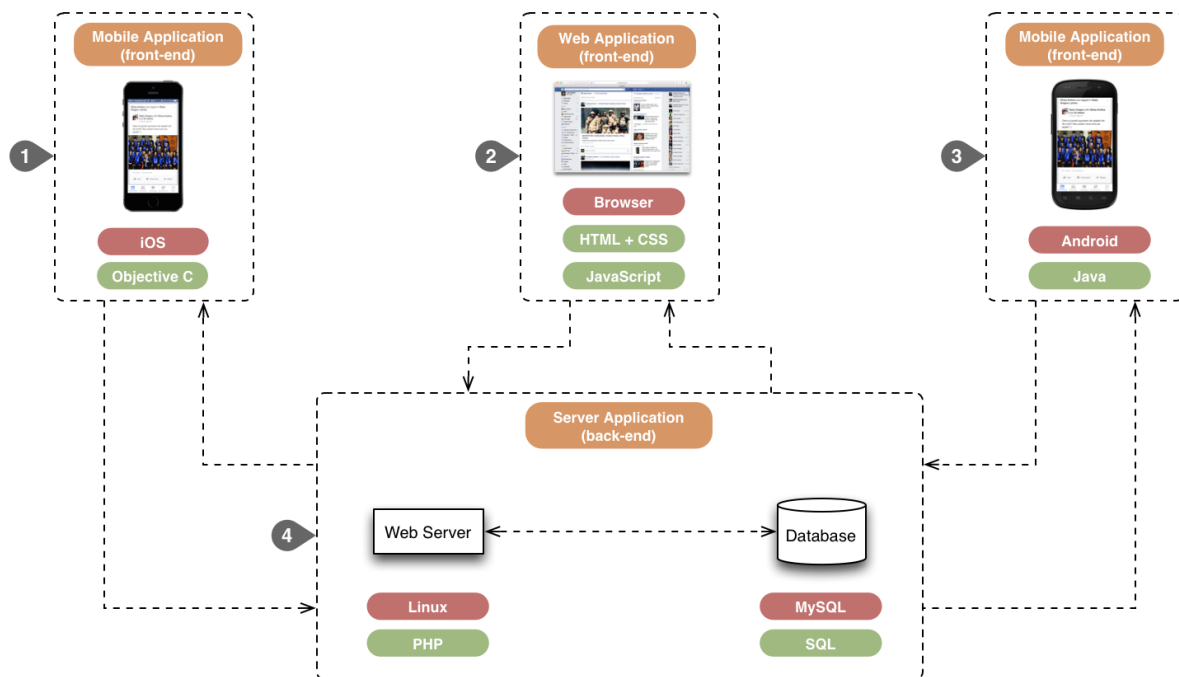
Internet, ktorý používame je plný webových stránok, aplikácii a rôznych informácií. Na internete nájdeme veľké množstvo služieb ako napríklad vyhľadávacie služby, ktoré nám pomáhajú nájsť informácie na internete, ďalej nájdeme na internete aj hry, kalkulačku, fotografie, videá a iné. Nie je však potrebné, aby sa všetky informácie nachádzali v pamäti počítača, aby sme k nim mali prístup, jednoducho si nainštalujeme jeden program a to webový prehliadač, prostredníctvom, ktorého sa ku žiadaným informáciám ľahko a rýchlo dostaneme[1].

V súčasnosti je nevyhnutné, aby prístup k informáciám bol rýchly a lacný. Webové aplikácie nám každodenne uľahčujú našu prácu a tiež život. Využívame napríklad rôzne bankové služby ako napríklad internet banking, pomocou ktorého máme neustále prehľad o našich financiách, používame ho na prevod peňazí, prichádza nám na náš účet výplata. Na to, aby táto aplikácia pracovala správne musí byť bezpečnosť a prenos dát na najvyššej úrovni. Samotná webová aplikácia nemôže spravovať všetky informácie, potrebuje spoľahlivé úložisko dát a na to, aby tieto dáta putovali z úložiska a boli poskytnuté klientskej strane potrebujeme serverovú aplikáciu. Serverová aplikácia poskytuje rôzne servisy, ktoré sa starajú o prenos dát z úložiska na klientsku stranu a z klientskej strany do úložiska. Serverová aplikácia sa stará o bezpečnosť pri prenose a uchovávaní dát.

Serverová aplikácia môže byť naprogramovaná v jazykoch PHP, Ruby, Java, .NET (ASP, C#), JavaScript (node.js), Python a mnoho ďalších. Serverová aplikácia nemá žiadne používateľské rozhranie, nie je pre používateľa viditeľná[3].

Princíp ako funguje serverová aplikácia bližšie prezentuje obrázok 1. Jednotlivé časti obrázka ukazujú, kde v štruktúre softvérovej aplikácie sa nachádza mobilná, webová a serverová aplikácia, ktorej súčasťou je webový

server a databáza. Obrázok názorne popisuje komunikáciu medzi jednotlivými časťami softvérovej aplikácie.



Obrázok 1. Softvérová aplikácia

2.2 Databáza

Dáta, s ktorými pracuje webová aplikácia musia byť niekde uložené. Do operačnej pamäte počítača sa jednoducho nezmestia. Tieto dáta by sa museli ukladať na disky, pásky a podobne, čo je veľmi pomalé, a preto uvažujeme algoritmy, ktoré efektívne spracovávajú veľké množstvo údajov na pomalom disku. Databáza je kolekcia dát uložená vo veľkých súboroch na disku. Kolekciu dát spravujú algoritmy[2]. Dáta sú systematicky uložené v tabuľkách, aby k nim aplikácia vedela rýchlo pristupovať. V databáze je možné upravovať dáta, pridávať nové dáta a mazať staré dáta. Dobre navrhnutá databáza zlepšuje prácu aplikácie, tým, že efektívne pracuje s dátami. Najpopulárnejším databázovým jazykom je SQL, na ktorého princípe sú založené databázy ako MySQL, PostgreSQL, Oracle DB, Microsoft SQL a NoSQL a tiež sú známe databázy MongoDB, Cassandra, Redis, CouchDB a ďalšie[3].

2.3 Aplikačný server

Aplikačný server slúži na vystavovanie, biznis-logiky klientskym aplikáciám prostredníctvom rôznych protokolov ako napríklad HTTP protokol[4]. To znamená, že aplikačný server zariaďuje prístup klientskej strany ku funkcionalite serverovej aplikácie . Hypertextový prenosový protokol slúži prevažne na prenos html dokumentov, ale aj iných dokumentov zo serveru ku klientovi. Každý programovací jazyk využíva iné aplikačné servery. Medzi známe aplikačné servery patria napríklad GlassFish (Java), Node.js (Javascript), Gunicorn (Python) a mnoho ďalších[5].

2.4 Webový server

Webový server má na starosti HTTP protokol, pripojenie do databázy (driver), transakcie, riadi CDI injections a iné . Keď webový server prijme HTTP požiadavku (request) , ako HTTP odpoveď (response) posiela html dokument , ktorý následne webový prehliadač zobrazí používateľom ako webovú stránku[4].

2.5 Webové servisy typu REST

Webové servisy typu REST sú v súčasnosti najpoužívanejšie webové servisy. Sú to také typy webových servisov, ktoré používajú na prácu s dátami HTTP požiadavky (requests) a to najmä typu GET,PUT,POST a DELETE požiadavky.

2.6 Technológie

V nasledujúcich častiach stručne opíšem technológie používané pri vývoji serverovej aplikácie. Opíšem programovací jazyk Java 7 Enterprise Edition, databázu PostgreSQL, Aplikačný server Payara a Apache Maven.

2.6.1 Programovací jazyk Java

Programovací jazyk Java je objektovo orientovaný programovací jazyk a patrí k obľúbeným a veľmi rozšíreným programovacím jazykom. Pre implementáciu serverovej aplikácie som zvolila programovací jazyk Java, pretože je veľmi používaný, ponúka niekoľko rôznych platform a využiteľných knižníc. Programovací jazyk Java rozlišuje niekoľko platform a to nasledovne [8]:

- *Java Standart Edition*

Je to základné jadro programovacieho jazyka Java. Definuje všetko od základných typov a objektov až po vysoko-úrovňové triedy používané na Networking, databázové pripojenie, bezpečnosť, vývoj GUI (Graphical user interface) a spracovávanie XML súborov[8].

- *Java Enterprise Edition*

Platforma Java Enterprise Edition je postavená na platforme Java Standart Edition. Platforma Java Enterprise Edition je súbor špecifikácií vhodný na vývoj Enterprise aplikácie. Možno ju považovať za rozšírenie platformy Java Standart Edition na uľahčenie vývoja distribuovaných, robustných, výkonných a vysoko dostupných aplikácií[6]. Viac priblížim platformu Java Enterprise Editon v samostatnej podkapitole 2.6.2 s názvom *Java Enterprise Edition*.

- *Java Micro Edition*

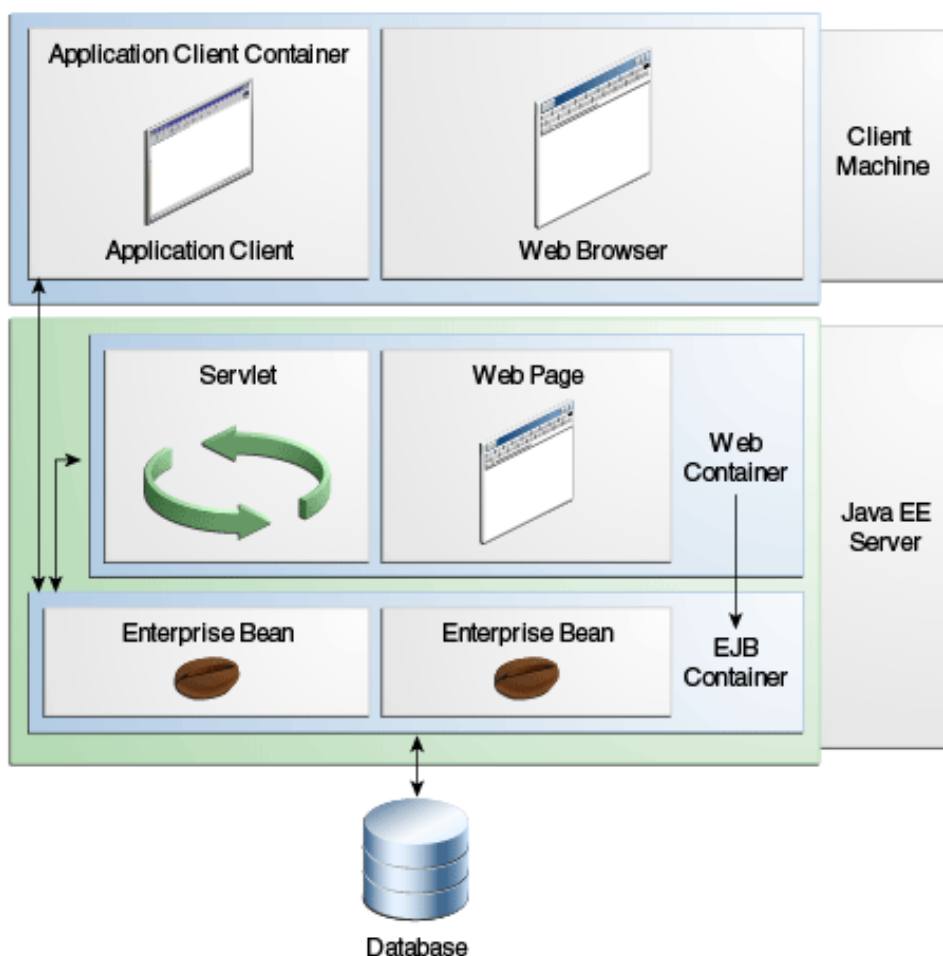
Platforma Java Micro Edition poskytuje API rozhranie pre small-footprint virtuálny stroj na spustenie aplikácií programovacieho jazyka Java na malých zariadeniach napríklad mobilných telefónoch. Rozhranie API vychádza z API platformy Java Standart Edition, spolu so špeciálnymi knižnicami užitočnými pre vývoj aplikácií pre malé zariadenia. Aplikácie platformy Java Micro Edition sú často klientmi služieb platformy Java Enterprise Edition[8].

- *Java Fx*

Platforma Java Fx je platforma, ktorá slúži na vytváranie bohatých internetových aplikácií pomocou ľahkého aplikačného rozhrania API[8]. Platforma Java Fx je vhodná aj na lepšie pochopenie princípov objektovo-orientovaného programovania v jazyku Java. Pomocou tejto platformy je možné programovať jednoduchšie ako aj zložitejšie hry. Ako predchádzajúce platformy , tak aj platforma Java Fx vychádza zo štandardnej platformy Java Standart Edition.

2.6.2 Java Enterprise Edition

Platforma Java Enterprise Edition sa používa nielen na vývoj webových a serverových aplikácií, ale tiež na vývoj transakčných, bezpečných, interoperabilných a distribuovaných aplikácií. Java Enterprise Edition poskytuje širokú škálu špecifikácií ako napríklad štandardný spôsob spracovania transakcií prostredníctvom Java Transaction API (JTA), posielanie správ prostredníctvom Java Message Service (JMS) ,spôsob perzistencie prostredníctvom Java Persistence API (JPA) a mnoho ďalších. Java Enterprise Edition (skr. Java EE) je súbor špecifikácií určených pre podnikové aplikácie. Súbor špecifikácií je implementovaný pomocou rôznych kontajnerov[6]. V súčasnosti je známych niekoľko verzií platformy Java EE. Pri vývoji serverovej aplikácie pre mobilného sprievodcu FMFI, budem používať verziu 7, to znamená Java Enterprise Edition 7. Princíp architektúry tejto platformy bližšie popisuje obrázok 2. Java EE Server and Containers.



Obrázok 2. Java EE Server and Containers

2.6.3 PostgreSQL databáza

PostgreSQL je objektovo-relačný databázový management systém (ORDBMS), založený na POSTGRES. PostgreSQL využíva veľkú časť štandardov jazyka SQL[9]. SQL je jazyk na prácu s databázovým systémom[2]. PostgreSQL ponúka množstvo moderných vlastností[9]:

- komplexné queries (complex queries)
- cudzie kľúče (Foreign keys)
- spúšťače (triggers)
- aktualizovateľné pohľady (updatable views)
- transakčná integrita (transactional integrity)
- multiversion concurrency control

Na prácu s databázovým systémom PostgreSQL sa väčšinou používa prostredie PgAdmin a Postgres service(Obrázok 3.). Na obrázku 5 je zobrazené prostredie PgAdmin4, s ktoré používam pri práci s PostgreSQL databázovým systémom.

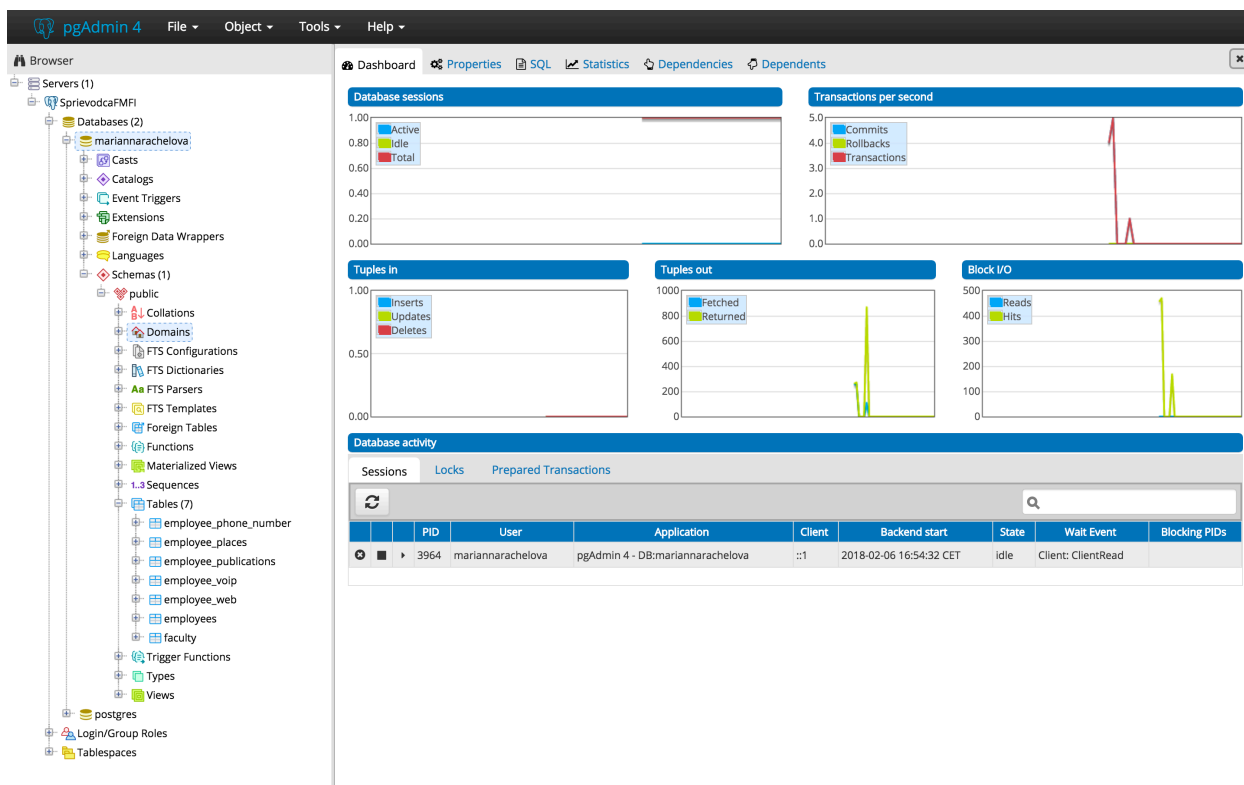


Obrázok 3. Postgres logo



PostgreSQL

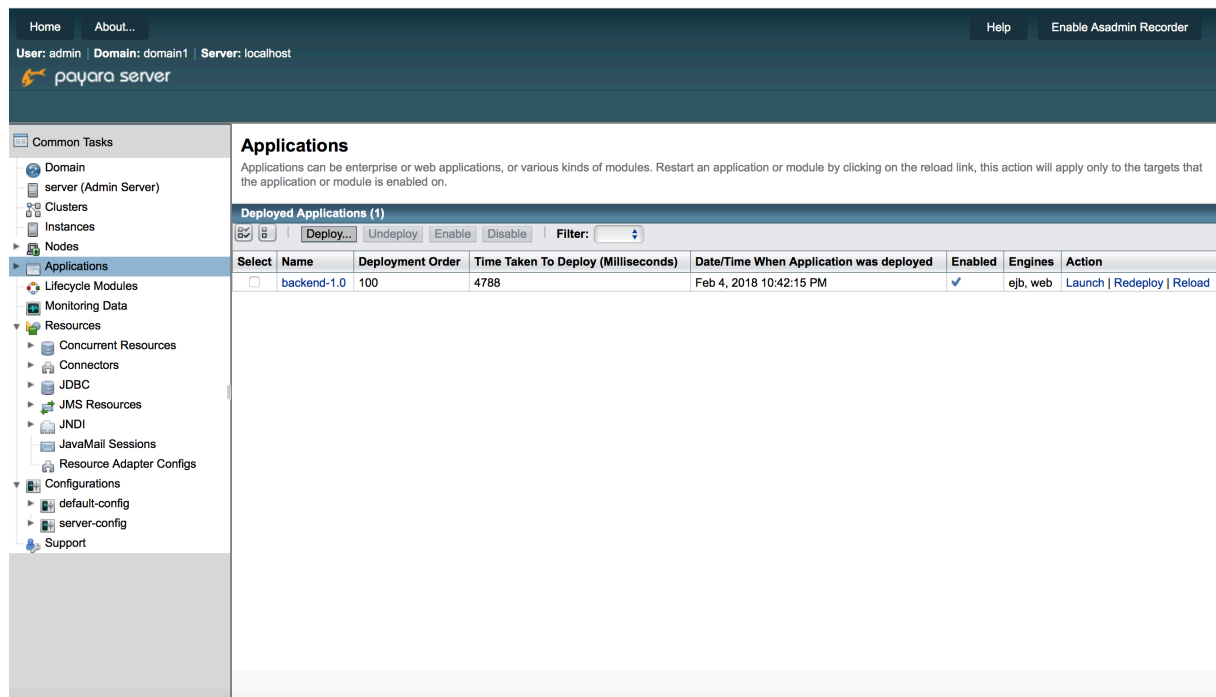
Obrázok 4. PostgreSQL logo



Obrázok 5. Prostredie PgAdmin 4

2.6.4 Aplikačný server Payara

Aplikačný server *Payara*, je open-source aplikačný server, ktorého základom je aplikačný server GlassFish. Payara server je výborná voľba pri vývoji aplikácií, ktoré sa programujú v programovacom jazyku Java, konkrétne na platforme Java Enterprise Edition[10]. Aplikačný server Payara obsahuje administrátorské rozhranie, ktoré je zobrazené na obrázku 6 nižšie.



Obrázok 6. Administrátorské rozhranie aplikačného serveru Payara

2.6.5 Apache Maven

Apache Maven je softvérový projekt manažment a nástroj na pochopenie[11]. Maven poskytuje vývojárovi kódu nasledovné zlepšenia pri vývoji nového kódu[11]:

- Zjednodušenie procesu build-ovania
- Zabezpečenie jednotného systému build-ovania
- Zabezpečenie kvality informácií o projekte
- Poskytovanie usmernení pre vývoj osvedčených postupov
- Umožnenie transparentnej migrácie na nové vlastnosti

Využití tohto nástroju je mnoho a je veľmi rozšíreným nástrojom pri vývoji aplikácií.

2.6.6 Vaadin

Vaadin je vývojárska platforma, ktorá ponúka všetky nástroje potrebné na vývoj webovej aplikácie. Produktmi Vaadin sú Vaadin Framework a Vaadin Elements[12].

2.6.7 Zhrnutie technológií

V tejto podkapitole zhrniem, ako všetky vyššie spomenuté technológie navzájom spolu súvisia pri vývoji serverovej aplikácie. Na začiatku vývoja akejkoľvek aplikácie stojí rozhodnutie výberu vhodných technológií. Výber technológií je veľmi dôležitý, pretože od toho závisí akým smerom sa ďalej bude vývoj aplikácie poberať. Je vhodné vybrať technológiu, ktorá je dostupná, to znamená, že je dostupná korektná dokumentácia alebo je jednoduché ju stiahnuť a nainštalovať vo svojom počítači a podobne. Zvolená technológia by mala byť tiež moderná a bezpečná.

Ja som zvolila pri implementácii serverovej aplikácie programovací jazyk Java platforma Enterprise Edition 7, pretože je to veľmi rozšírená platforma pri vývoji aplikácií a použitie tejto platformy ponúka veľa možností. Serverová aplikácia spravuje dáta, ktoré sú uložené v databáze. Zvolila som PostgreSQL databázu, pre jednoduchú a efektívnu prácu s touto databázou a tiež kvôli predchádzajúcim skúsenostiam s databázou. Každá časť kódu naprogramovaná v Jave musí byť pred spustením kódu prejsť procesom build. Na urýchlenie a zefektívnenie týchto procesov slúži nástroj Apache Maven, ktorý obsahuje v sebe súbor knižníc, ktoré zariaďujú proces build, ale aj napríklad process clean, install, package a podobne. Potom ako je kus kódu zbalený prostredníctvom procesu clean a install alebo clean a package, potrebujeme tento kus kódu dostať na aplikačný server. Proces prenosu kódu na aplikačný server sa nazýva „deployment“. Po tomto prenose Aplikačný server, v mojom prípade Payara Server vystaví webový servis frontendovej časti aplikácie, aby táto časť mala prístup k danému kusu kódu. Tento kus kódu sa nazýva aj „biznis-logika“ aplikácie.

2.8 Mobilný sprievodca FMFI

Mobilný sprievodca FMFI[7] je frontendová časť aplikácie, ku ktorej je integrovaná serverová časť aplikácie. Mobilný sprievodca FMFI je mobilná aplikácia, ktorá zobrazuje mapu fakulty, to znamená jednotlivé časti fakulty ako

učebne, v ktorých prebieha výučba a kancelárie zamestnancov fakulty. Mobilná aplikácia tiež ponúka vyhľadávanie trasy z jednej časti fakulty do druhej časti fakulty. Mobilného sprievodcu FMFI naprogramoval Martin Bohumel v rámci svojej bakalárskej práce.

2.9 Návrh serverovej aplikácie a jej databázy

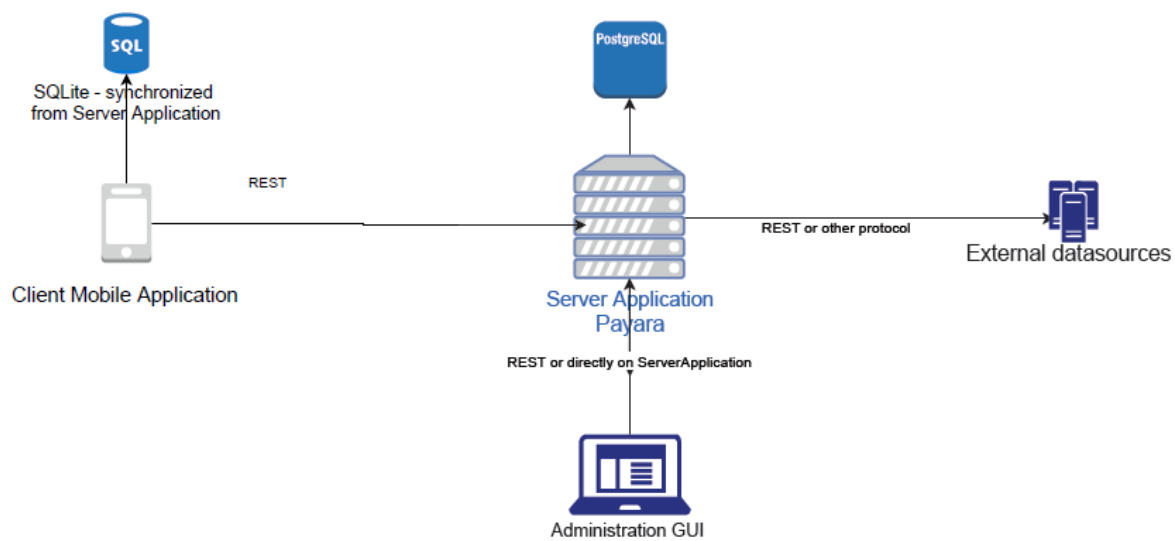
V tejto podkapitole priblížim návrh serverovej aplikácie a vysvetlím stručne ako funguje.

2.9.1 Model serverovej aplikácie

Serverová aplikácia pozostáva z niekoľko častí a to z databázy, aplikačného serveru a administrátorského rozhrania. Komunikácia medzi serverovou časťou „backendom“ a mobilnou aplikáciou „frontendom“ je vyriešená prostredníctvom REST-ových webových servisov, ktoré som spomenula a vysvetlila v podkapitole 2.5. Dáta z databázy serverovej aplikácie sa posielajú na základe požiadavky – request a sú posielané ako odpoveď-response. Dáta sa posielajú zbalené vo vhodnej štruktúre ako napríklad json súbor, ktorý je možné jednoducho a rýchlo spracovať. Mobilná aplikácia súbor json spracuje a uloží do svojej SQL Lite databázy. Dáta v databáze serverovej časti a mobilnej časti sú vždy rovnaké. SQL Lite databáza sa vždy aktualizuje podľa serverovej databázy. Serverová aplikácia získava dáta dvomi spôsobmi a to nasledovne:

1. Používa REST-ové webové servery na získanie dát z externých zdrojov, ako napríklad prostredníctvom HTTP GET požiadavky je možné získať HTML dokument z konkrétnej webovej stránky
2. Dáta sú vložené do databázy prostredníctvom Administrátorského rozhrania

Prostredníctvom administrátorského rozhrania je možné vkladať nové dáta do databázy, mazať dáta z databázy a aktualizovať už existujúce dáta. Aplikačný server slúži na vystavovanie biznis-logiky mobilnej aplikácií. Ako návrh vyzerá a kde sa nachádzajú v návrhu jednotlivé časti aplikácie popisuje obrázok 7. Návrh serverovej aplikácie.



Obrázok 7. Návrh serverovej aplikácie

2.9.2 Databázový model

TODO

Zdroje

[1] Leonard Richardson, Sam Ruby Foreword by David Heinemeier Hansson, *RESTful Web Services*, O'Reilly Media, United States of America , 2007, ISBN: 978-0-596-52926

[2] Ing. Alexander Šimko, PhD. , *I-AIN-221/15 Databázy (I)*, Fakulta matematiky, fyziky a informatiky Univerzity Komenského, 2017

[3] Nitrovery blog, *Types of Software Projects*, [18.6.2014], [online]. Dostupné na internete pod adresou:

<http://blog.nitrovery.com/itsimply/2014/06/18/Types-of-Software-Projects/>

[4] Tony Sintes, *App server, Web server: What's the difference ?*

[23.8.2002],[online]. Dostupné na internete pod adresou:

<https://www.javaworld.com/article/2077354/learn-java/app-server-web-server-what-s-the-difference.html>

[5] Wikipedia , free encyclopedia, *List of application servers* [online]. Dostupné na internete pod adresou:

https://en.wikipedia.org/wiki/List_of_application_servers#Java

[6] Antonio Goncalves, *Beginning Java EE 7*, 24.7.2013, Apress, ISBN: 143024626X

[7] Bc. Martin Bohumel, *Mobilný sprievodca FMFI*, Fakulta matematiky, fyziky a informatiky Univerzity Komenského, 2017

[8] Oracle Dokumentácia, dostupná na internete pod adresou:

<https://docs.oracle.com/javase/6/firstcup/doc/gkhoy.html>

[9] PostgreSQL dokumentácia, dostupná na internete pod adresou:

<https://www.postgresql.org/docs/10/static/index.html>

[10] Payara, dokumentácia, dostupná na internete pod adresou:

<https://www.payara.fish/about>

[11] Apache Maven, dokumentacia, dostupna na internete pod adresou:

<https://maven.apache.org/>

[12] Vaadin, dokumentácia, dostupná na internete pod adresou:

<https://vaadin.com/>