

Západočeská univerzita v Plzni
Fakulta aplikovaných věd
Katedra informatiky a výpočetní techniky

KIV/PIA-E

Semestrální práce

Obsah

1	Zadání	2
2	Implementace	8
2.1	Core	8
2.1.1	Domain	9
2.1.2	Repository	9
2.1.3	Service	9
2.2	Jdbc	9
2.2.1	Mapper	9
2.2.2	Repository	10
2.3	Web	10
2.3.1	Webapp	10
2.3.2	Controller	10
2.3.3	VO	10
2.3.4	Resources	11
3	Uživatelská příručka	12
3.1	Přihlášení	12
3.2	Úvodní stránka	13
3.3	Registrace uživatele	14
3.4	Vytvoření nového projektu	16
3.5	Podrobné zobrazení projektu	17
3.6	Úprava projektu a přiřazených uživatelů	18
3.7	Správa uživatelů	20
3.8	Vyhledání projektu a odhlášení	23
3.9	Zobrazení úvazku v poměru FTE	23
4	Instalace a spuštění	24
5	Závěr	25

1 Zadání

Target

The main objective of the project is to develop a web application allowing tracking of **allocations/jobs/project assignments (this is SINGLE thing, pick whichever name you want)** of the department members.

Department employees participate in many different projects, with different size of allocation (x-hours per week) throughout time. The application is supposed to provide project managers with tooling to oversee their team members allocations (current and planned), ensure the person's total allocation does not exceed 1.0 at any given moment, etc.

The most successful project implementation will be taken into use by the Department of Computer Science.

Base Requirements [30p]

Roles

The application users take on several distinct roles, each with its own set of permissions and access to different functionalities. The roles may be combined. The following is the mapping between the roles and permissions. The role names are placeholders and may be changed by the developer to a degree.

- Unauthorized user
 - Sign in
- Regular authorized user
 - Review their own allocations/jobs/project assignments
 - Log out
- Superior
 - Review the allocations/jobs/project assignments of all their subordinates

- Project manager
 - Review the allocations/jobs/project assignments of all people (only) on their projects
 - Input/edit allocations/jobs/project assignments of people on their projects
 - See the aggregate workload of people they want to add to their project across all projects
 - * i.e., they see the sum of people's allocations across projects, but do not see on which projects they are, nor the division of their allocations among projects
 - * The aggregate is for 1) active assignments, 2) all assignments
 - Edit their projects' information
- Department manager
 - Same as project manager, but for all projects
- Secretariat
 - Register users
 - Create projects
 - Assign people to projects
 - Edit projects and people information
 - Adds subordinates to superiors

Data hierarchy between roles goes in the following fashion:

Superior -> Worker (M:N) - relationship between workers in departmental hierarchy (independent on projects)

Project manager -> Project (M:1) - single person can have project manager role for multiple projects, but it has to be explicitly configured per-case

User registration

Done primarily by the Secretariat role by inputting their Orion username into a proper form field. The application will hold the following user information:

- Full name
- Orion login/username
- Primary workplace
- Email address

Other user information is above the minimal project criteria.

Login

A user logs in through orion username and generated password. Password is shown to registrator (Secretariat) upon registration and must be changed by user upon first login. **This requirement can be replaced by Kerberos integration (see Optional Functional Requirements below)**

Project

The project contains (at least) the following data:

- Project name
- Project manager (orion login)
- Timespan (from-to; end date is optional, if not entered viewed as indefinite)
- Description/Note

Allocation/job/project assignment

The creation of the allocation/job/project assignment contains the following data:

- Worker
- Project
- Allocation scope (as a ratio of FTE)
 - FTE=full time equivalent=40h/week
 - Entered and displayed in hours/week or x/1FTE
- Timespan (from-to)

- Description/Note

The allocation/job/project assignment has a lifecycle with the following distinct states which need to be recognizable in the application (the names are not compulsory):

- Inactive
 - Draft/intended
 - Cancelled/unrealized
 - Past
- Active

Some transitions between these states can be calculated or automatically triggered (unrealized, past), while others need to be triggered by the users with proper permissions (active, cancelled).

Extra-functional requirements

GUI will be primarily in CZ (does not apply to Erasmus students).

Optional functional requirements

User Registration and authentication

- User data import from UWB information systems (STAG integration) based on orion login [+8p]
https://is-stag.zcu.cz/napoveda/web-services/ws_changelists.html
- Integration with UWB single-sign-on system (SAML, Kerberos) [+5p]
 (replaces default username and password authentication)
<https://support.zcu.cz/index.php/Kerberos>

Notifications

- Application will send notifications via email triggered by appropriate events to the proper users. [+4b]
 Minimum list of events:
 - activation of a project/assignment,
 - expiration of assignment in near future
 - project creation and assignment

UX

- Whenever multiple allocations/jobs/project assignments are displayed, the list can be filtered, sorted, grouped, and appropriate aggregation functions (e.g., sums) will be displayed. [+3p]
- When planning project/person's assignments a user-friendly display of their existing allocations over time with sliding time window or similar feature (like, for example, an online calendar). [+5p]
- charts/graphs where appropriate [+3p]
- The ability to enter and view assignments scopes in both x/1FTE and hours/week. [+1p]
- Bilingual, CZ/EN GUI, or at least an infrastructure prepared for it. [+3p for infrastructure, +1p for translations]

Deployment and maintenance

- Detailed maintenance guide [+2p] for installation and standard/regular maintenance task
- Detailed user guide [+2p] with screenshots for the main use-cases

Other

- SPA + WS API architecture [+10p]
- customizable notification settings (which notifications user wants to receive) [+4p]
- exports (ask ppicha@kiv.zcu.cz for details on what to export)
 - to CSV [+2p]
 - PDF (pretty, not just plain text!) [+4p]

General Requirements

Allowed technology

Java

C#

Ruby

Python

Node.js (Typescript is mandatory)

PHP (Laravel or Symphony or Nette is mandatory)

Implementation requirements

- Correct implementation of selected architecture
- Reasonably long methods (typically 50-100 lines, exceptions allowed)
- Well-commented methods/functions, data models, attributes
- User management module covered with automated tests (unit, integration)
- Configurations must be read from property files, env variables, etc.

2 Implementace

Základ aplikace vychází z projektu který byl vytvořen na cvičení.

Využité technologie:

- Spring Framework
- Thymeleaf
- Tomcat
- Docker (MySQL)
- Mockito

Struktura projektu:

- Core
- Jdbc
- Web

Implementovaná rozšíření:

- Infrastruktura pro více jazyků
- Uživatelská příručka s obrázky
- Zobrazení úvazku v hodinách a poměru FTE

2.1 Core

Tento modul pracuje s daty. Dělí se na 3 hlavní sekce:

- Domain
- Repository
- Service

2.1.1 Domain

Zde jsou definovány všechny základní třídy:

- Assignment
- Project
- Subordinate
- User

Třída `User` představuje uživatele pracující s aplikací. Tito uživatelé mohou obsahovat podřízené, což je realizováno přes třídu `Subordinate`. Třída `Project` definuje projekty do kterých jsou uživatelé přidáni. Jejich přidání do projektu, a tedy i následná definice přiřazení, je popáno pomocí třídy `Assignment`.

2.1.2 Repository

Zde jsou definovány rozhraní, přes která se ukládají a načítají data typu `Domain`. Tyto rozhraní jsou zde navíc implementována do tříd, které jsou schopné data ukládat do paměti.

2.1.3 Service

Tato sekce obsahuje `Service` třídy, které definují jaké akce lze provádět nad `Domain` daty. Tyto třídy zpravidla využívají `Repository` rozhraní k provedení požadované akce.

2.2 Jdbc

Tento modul slouží jako prostředek pro komunikaci s databází. Dělí se na 2 sekce:

- Mapper
- Repository

2.2.1 Mapper

Tato sekce obsahuje mapper pro každou třídu z `Domain`. Tyto třídy ukládají nahraná data z databáze, která se poté zobrazí uživateli.

2.2.2 Repository

Stejně jako v předešlém modulu **Core**, zde třídy slouží k ukládání a načítání dat. Tentokrát se ale pracuje s daty přímo z databáze. Tedy tyto třídy obsahují seškeré SQL příkazy potřebné pro komunikaci aplikace s databází.

2.3 Web

Tento modul slouží jako vstupní bod aplikace a je zodpovědný za komunikaci s uživatelem. Tedy zpracování požadavků a vykreslování samotného webu. Dělí se na 4 hlavní sekce:

- Webapp
- Controller
- VO
- Resources

2.3.1 Webapp

Slouží ke spuštění samotné aplikace. Je zde definován celkový kontext a nastavení aplikace. Také obsahuje potřebné třídy pro realizaci více lokalizací aplikace.

2.3.2 Controller

Zde třídy definují url adresy aplikace a jak se aplikace zachová na požadavky od uživatele. Tedy pokud uživatel zažádá o html stránku přes validní url adresu, tak načte potřebná data a stránku vykreslí. Aplikace využívá server-side rendering. Jestliže přijde požadavek na zpracování akce od uživatele, např.: stisknutí tlačítka, tak také definuje jak se aplikace zachová. Po zpracování požadavku předá/vyvolá příslušnou metodu ze **Service** tříd z modulu **Core**, která požadavek zpracuje už interně.

2.3.3 VO

Zkratka pro **Value Object**. Slouží k uchování dat než dojde k jejich zpracování. Využito především ve formulářích, které uživatelé vyplňují.

2.3.4 Resources

Obsahuje potřebné soubory pro iniciální spuštění s databází, veškeré html soubory a `properties` soubory pro angličtinu a češtinu.

3 Uživatelská příručka

3.1 Přihlášení

Výchozí bod aplikace je `/pia-sp/spring/`, tedy v našem případě:

`http://localhost:8080/pia-sp/spring/`

Jelikož nejsme přihlášení, tak budeme automaticky přesměrováni na přihlašovací stránku `/pia-sp/spring/login` 3.1.



Obrázek 3.1: Přihlašovací stránka

Po úspěšném přihlášení můžeme vstoupit do aplikace, neboli jsme přesměrováni zpět na původní zadanou adresu, tedy v našem případě `/pia-sp/spring/` 3.2.

Výchozí uživatelé jsou:

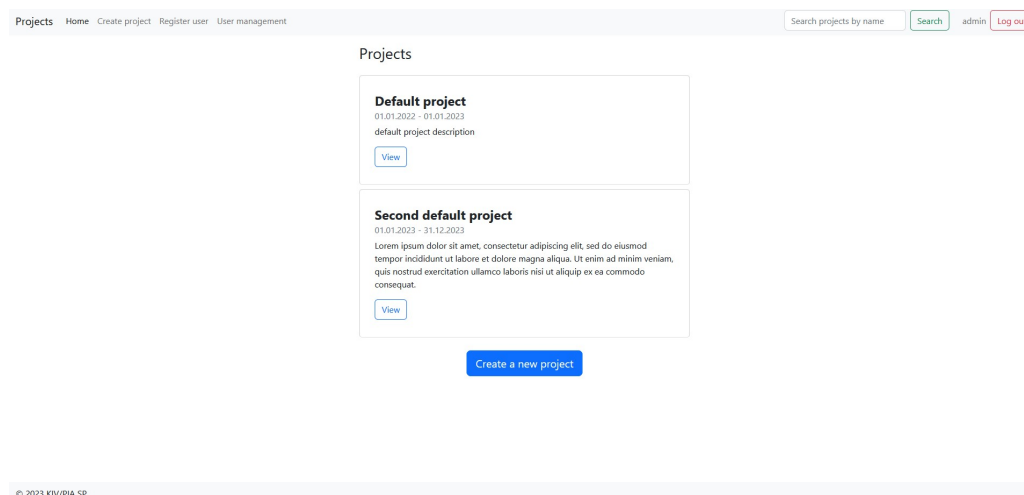
- admin
- dmanager
- pmanager
- superior
- default
- default2

Heslo uživatele admin je *admin*. Pro ostatní uživatele je heslo *password*. Uživatel admin má práva sekretariátu, ostatní uživatelé odpovídají dalším

právům následovně: department manager, project manager, superior a běžný uživatel.

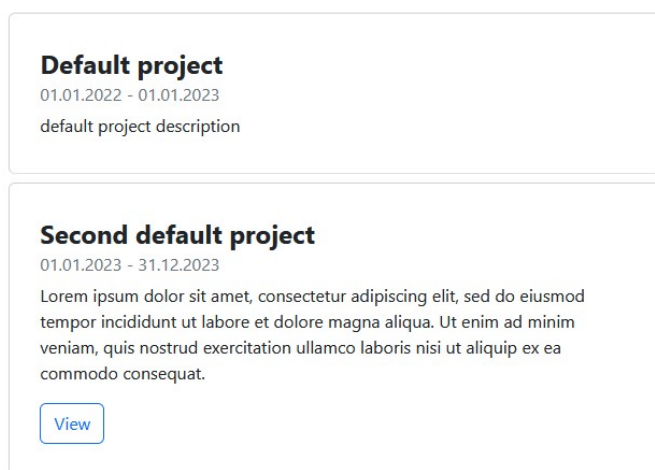
3.2 Úvodní stránka

Úvodní stránka aplikace. Její obsah se liší podle role přihlášeného uživatele.



Obrázek 3.2: Úvodní stránka pro sekretariát

Položky v menu jako **Register user** a **Create project** vidí pouze sekretariát. Sekretariát a Department manager si také mohou zobrazit všechny projekty naopak od Projekt manažera, který si může zobrazit jen svoje projekty 3.3, či od běžného uživatele.



Obrázek 3.3: Zobrazení projektů pro Projekt manažera

3.3 Registrace uživatele

Stránka obsahující formulář pro registraci uživatele 3.4.

Register a new user

Firstname

Lastname

Username

Password

Role

Workplace

Email

Obrázek 3.4: Vytvoření nového uživatele

Pokud uživatel již existuje 3.5 nebo nejsou vyplněna všechna pole formuláře, tak uživatel o tom bude informován.

Register a new user

An account already exists for this username.

Failure! Operation failed. Please try again

Firstname

Test

Lastname

Test

Username

admin

Obrázek 3.5: Uživatel již existuje

3.4 Vytvoření nového projektu

Stránka obsahující formulář pro vytvoření nového projektu 3.6. Formulář opět kontroluje zda projekt již existuje, všechna pole jsou vyplněna a že zadaný rozestup pro datum je validní.

Create a new project

Project name

Project manager username

Project from

Project to

Project description

Create

Obrázek 3.6: Vytvoření nového projektu

3.5 Podrobné zobrazení projektu

Stránka zobrazující detaily projektu. Jsou zde vidět přiřazení uživatelé a detaily o jejich přiřazení 3.7. Podle role přihlášeného uživatele lze upravit projekty či přiřazení uživatele 3.9, nebo přidat nového uživatele do projektu 3.8.

Default project

01.01.2022 - 01.01.2023

default project description

Users assigned

Assign New User

Name	Username	Role	Email	Time assigned	Assignment scope	Assignment note	Assignment status
Emma Williams	default	REGULAR-USER	emmawilliams@zcu.cz	01.01.2022 - 01.01.2023	25.0 h/week	default assignment of basic user	Active
Jane Doe	dmanager	DEPARTMENT-MANAGER	janedoe@zcu.cz	01.01.2022 - 01.01.2023	16.0 h/week	default assignment of department manager	Active

Obrázek 3.7: Zobrazení projektu pro Sekretariát

Users assigned

Assign New User

Username

Add user

Obrázek 3.8: Přidání uživatele do projektu

Default project

01.01.2022 - 01.01.2023

default project description

Users assigned

Name	Username	Role	Email	Time assigned	Assignment scope	Assignment note	Assignment status	Action
Emma Williams	default	REGULAR-USER	emmawilliams@zcu.cz	01.01.2022 - 01.01.2023	25.0 h/week	default assignment of basic user	Active	
Jane Doe	dmanager	DEPARTMENT-MANAGER	janedoe@zcu.cz	01.01.2022 - 01.01.2023	16.0 h/week	default assignment of department manager	Active	

Obrázek 3.9: Zobrazení projektu pro manažery

3.6 Úprava projektu a přiřazených uživatelů

Ze stránky projektu lze upravit projekt 3.10 nebo přiřazení uživatele 3.11. Oba formuláře jsou opět ošetřeny na validní vstupy.

Edit Default project

Project name

Default project

Project from

01. 01. 2022

Project to

01. 01. 2023

Project description

default project description

Update

Obrázek 3.10: Upravení projektu

Edit Assignment

Assignment from

01. 01. 2022

Assignment to

01. 01. 2023

Assignment scope

20

Assignment note

default assignment of basic user

Assignment status

Active

Update







Obrázek 3.11: Upravení přiřazení uživatele

3.7 Správa uživatelů

Tato stránka se liší nejvíce podle role přihlášeného uživatele. Podle přihlášené role vidí uživatel jiná data v tabulce. Každý záznam v tabulce je interaktivní, kde po kliknutí se objeví nová řada obsahující projekty spojené s tímto uživatelem a jeho přiřazení do těchto projektů.

- **Sekretariát:** vidí všechny uživatele a jejich jednotlivá přiřazení 3.12. Uživatele tady lze může i upravit 3.14 nebo přidat podřízeného k nadřízenému 3.13.
- **Department manager:** vidí všechny přiřazené uživatele a jejich aktivní a celkové vytížení 3.15.
- **Project manager:** vidí uživatele na jejich projektech a jejich aktivní a celkové vytížení 3.16.
- **Superior:** vidí jejich podřízené a jejich jednotlivé přiřazení 3.17.
- **Regular user:** vidí jen sebe a svoje přiřazení 3.18.

All users

Assign Subordinate						
Firstname	Lastname	Username	Role	Workplace	Email	Action
John	Doe	admin	SEKRETARIAT	fav	johndoe@zcu.cz	
Jane	Doe	dmanager	DEPARTMENT-MANAGER	fav	janedoe@zcu.cz	
Robert	Smith	pmanager	PROJECT-MANAGER	fav	robertsmith@zcu.cz	
Briskit	Himchin	superior	SUPERIOR	fav	briskithimchin@zcu.cz	
Emma	Williams	default	REGULAR-USER	fav	emmawilliams@zcu.cz	
Project name		Assigned time	Assignment scope	Assignment note	Assignment status	
Default project		01.01.2022 - 01.01.2023	20.0 h/week	default assignment of basic user	Active	
Second default project		Not yet assigned	0.0 h/week	newly assigned	Draft	
Jack	Porter	default2	REGULAR-USER	fav	jackporter@zcu.cz	

Obrázek 3.12: Správa uživatelů pro Sekretariát

All users

Assign Subordinate

Superior Username

Subordinate Username

Add Subordinate

Obrázek 3.13: Přidání podřízeného k nadřízenému

Edit user information

Firstname

Briskit

Lastname

Himchin

Username

superior

Password

Role

Superior

Workplace

fav

Email

briskithimchin@zcu.cz

Update

Obrázek 3.14: Úprava uživatele

All assigned users

Firstname	Lastname	Username	Role	Workplace	Email
Jane	Doe	dmanager	DEPARTMENT-MANAGER	fav	janedoe@zcu.cz
Robert	Smith	pmanager	PROJECT-MANAGER	fav	robertsmith@zcu.cz
Emma	Williams	default	REGULAR-USER	fav	emmawilliams@zcu.cz
Active workload		Overall workload			
10.0 h/week		30.0 h/week			

Obrázek 3.15: Správa uživatelů pro Department manažera

All users on my projects

Firstname	Lastname	Username	Role	Workplace	Email
Emma	Williams	default	REGULAR-USER	fav	emmawilliams@zcu.cz
		Active workload	Overall workload		
		10.0 h/week	30.0 h/week		
Robert	Smith	pmanager	PROJECT-MANAGER	fav	robertsmith@zcu.cz

Obrázek 3.16: Správa uživatelů pro Projekt manažera

All subordinates

Firstname	Lastname	Username	Role	Workplace	Email
Emma	Williams	default	REGULAR-USER	fav	emmawilliams@zcu.cz
	Project name	Assigned time	Assignment scope	Assignment note	Assignment status
	Default project	01.01.2022 - 01.01.2023	20.0 h/week	default assignment of basic user	Canceled
	Second default project	02.01.2023 - 13.01.2023	10.0 h/week	second assignment	Active

Obrázek 3.17: Správa uživatelů pro Nadřízené

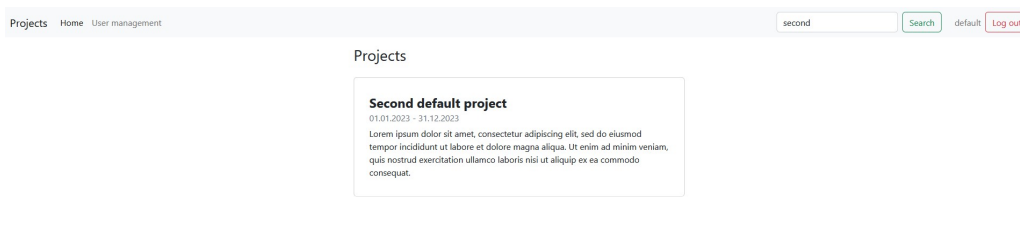
Account overview

Firstname	Lastname	Username	Role	Workplace	Email
Emma	Williams	default	REGULAR-USER	fav	emmawilliams@zcu.cz
Project name		Assigned time	Assignment scope	Assignment note	Assignment status
Default project		01.01.2022 - 01.01.2023	20.0 h/week	default assignment of basic user	Canceled
Second default project		02.01.2023 - 13.01.2023	10.0 h/week	second assignment	Active

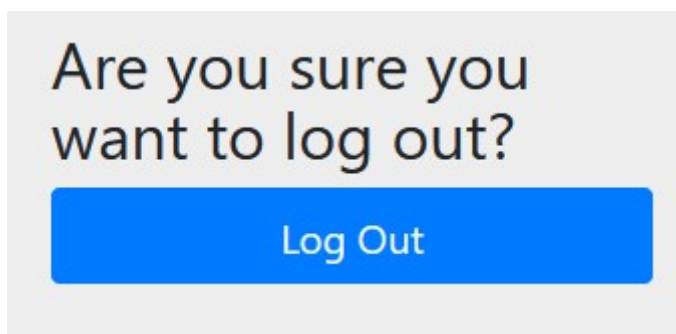
Obrázek 3.18: Správa uživatelů pro Běžného uživatele

3.8 Vyhledání projektu a odhlášení

Na úvodní stránce také lze projekty filtrovat podle jména 3.19 a nebo se odhlásit 3.20 přes položky v pravém horním rohu.



Obrázek 3.19: Vyhledání projektu



Obrázek 3.20: Odhlášení uživatele

3.9 Zobrazení úvazku v poměru FTE

Kdykoli je vidět **scope** přiřazení uživatele, tak pokud ukážeme kurzorem na danou hodnotu, tak se v tooltipu objeví poměr FTE pro danou hodnotu 3.21.



Obrázek 3.21: Zobrazení úvazku v poměru FTE

4 Instalace a spuštění

Pro úspěšné spuštění aplikace musíme mít nainstalovaný docker, kde spustíme Docker image pro MySQL databázi příkazem:

```
docker run -it -rm -name mysql -p 3306:3306 -e MYSQL_ROOT_PASSWORD=root  
mysql:8
```

V dockeru se lze k dané databázi připojit přes terminál tímto příkazem:

```
mysql -user=root -password=root
```

Poté musíme samotnou aplikaci zkompileovat pomocí Mavenu v rootu aplikace příkazem:

```
mvn clean package
```

Samotnou aplikaci spustíme přes třídu `Webapp` z modulu `pia-web`. Poté jen zbývá otevřít prohlížeč a zadat adresu:

```
http://localhost:8080/pia-sp/spring/
```

5 Závěr

Výsledkem semestrální práce je aplikace, která funguje jako správa projektů a uživatelů nad těmito projekty. Práce splňuje všechny základní body zadání a je implementováno i pár volitelných rozšíření.