

# INFORMAČNÍ SYSTÉM PRO MĚSTSKÝ ÚŘAD

Hana Šumšalová

## 1. Artefakt vize

CO?

Jedná se o informační systém pro městský úřad (dále pouze MU), který bude sloužit pro evidenci potřebných informací, jako jsou údaje o jednotlivých odděleních (včetně výčtu všech oddělení), aktuálně probíhajících projektech, nebo základní a potřebné informace o zaměstnancích (telefonní čísla, číslo kanceláře, ...).

KDE?

Systém slouží převážně pro zaměstnance konkrétního městského úřadu, z tohoto důvodu se předpokládá, že přístup k systému bude umožněn přes stolní počítače umístěné na daném MU.

JAK?

Systém umožní procházet a vyhledávat různé seznamy (projekty, kanceláře, zaměstnanci, ...). Dále bude možné zadávat nové projekty a s tím bude spojena funkce přidávání zaměstnance k danému projektu. Zaměstnanci budou mít možnost procházet uložené data a případně je i upravovat (záleží na administrátorských právech konkrétního zaměstnance).

KDO?

Zaměstnanci konkrétního MU a s nimi také zahrnuto vedení města (zastupitelé, starosta). Do systému mohou dostat přístup případně i zaměstnanci externích firem, které budou vykonávat služby na zakázce MU.

KDY?

Vstup do tohoto systému nebude nikterak časově omezen, pouze v případech údržby systému, může dojít k jeho dočasnému vyřazení. O této skutečnosti však budou uživatelé dopředu upozorněni.

PROČ?

Cílem tohoto systému je zvýšit efektivitu a zlepšit správu města. Poskytuje centrální úložiště informací o jednotlivých odděleních, aktuálně probíhajících projektech a základních údajích o zaměstnancích. Tímto způsobem usnadňuje práci jak v rámci jednotlivých úředníků, tak i vedení města, včetně zastupitelů a starosty. Dále pak veřejný přístup k informacím o projektech a kontaktním údajům zaměstnanců přispívá k větší otevřenosti veřejné správy a tím také lepší orientaci činitelů o probíhajících projektech, dále pak pomáhá usnadňovat spolupráci na projektech skrze oddělení. Celkově je cílem informačního systému pro městský úřad zlepšit efektivitu správy města, poskytovat transparentnost, umožnit lepší komunikaci a dodržovat všechny relevantní právní předpisy. Tímto způsobem přispívá k lepší kvalitě veřejných služeb a celkovému rozvoji města.

## 2. Sběr požadavků na systém

### Správa oddělení:

- Uživatelé: Zaměstnanci městského úřadu.
- Akce: Zobrazení seznamu oddělení, jejich výčtu a základních informací.
- Důvod: Umožnit zaměstnancům rychlý přístup k informacím o jednotlivých odděleních.

### Správa projektů:

- Uživatelé: Zaměstnanci městského úřadu.
- Akce: Prohlížení aktuálně probíhajících projektů a jejich detailní zobrazení.
- Důvod: Umožnit zaměstnancům monitorovat a spravovat probíhající projekty.

### Správa zaměstnanců:

- Uživatelé: Zaměstnanci městského úřadu.
- Akce: Zobrazení základních informací o zaměstnancích (telefonní čísla, čísla kanceláří) a jejich úprava (s administrátorskými právy).
- Důvod: Umožnit efektivní komunikaci a aktualizaci kontaktních údajů zaměstnanců.

### Přidávání projektů:

- Uživatelé: Zaměstnanci městského úřadu s odpovídajícími právy.
- Akce: Možnost vytvářet nové projekty a přidávat zaměstnance k daným projektům.
- Důvod: Usnadnění správy projektů a jejich přidělování zaměstnancům.

### Veřejný přístup:

- Uživatelé: Všichni uživatelé systému
- Akce: Prohlížení informací o projektech a kontaktních údajů zaměstnanců (s omezeným přístupem).
- Důvod: Zajištění transparentnosti veřejné správy a informování občanů.

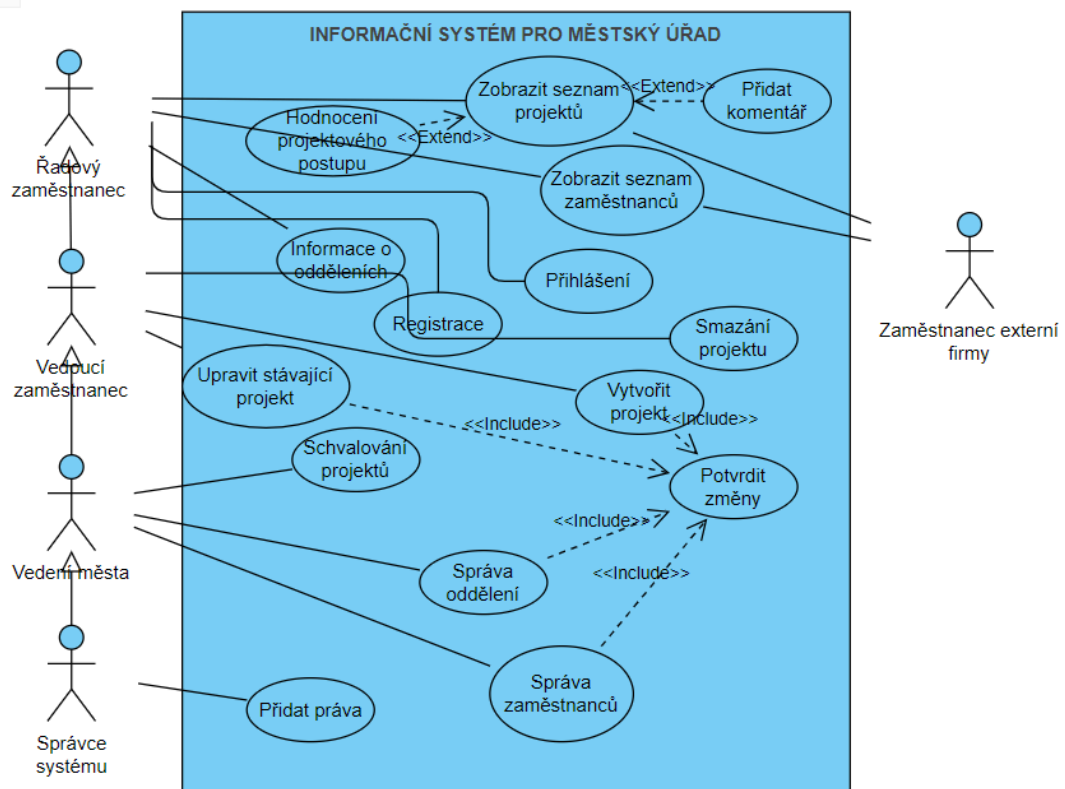
### Časově neomezený přístup:

- Uživatelé: Všichni uživatelé systému.
- Akce: Nepřetržitý přístup k systému s výjimkou plánované údržby.
- Důvod: Umožnění uživatelům pracovat s systémem bez časových omezení.

### Efektivita a transparentnost:

- Uživatelé: Všichni uživatelé systému.
- Akce: Zvýšení efektivity práce městského úřadu a zlepšení transparentnosti veřejné správy.
- Důvod: Dosáhnout lepšího řízení města a zlepšení veřejných služeb.

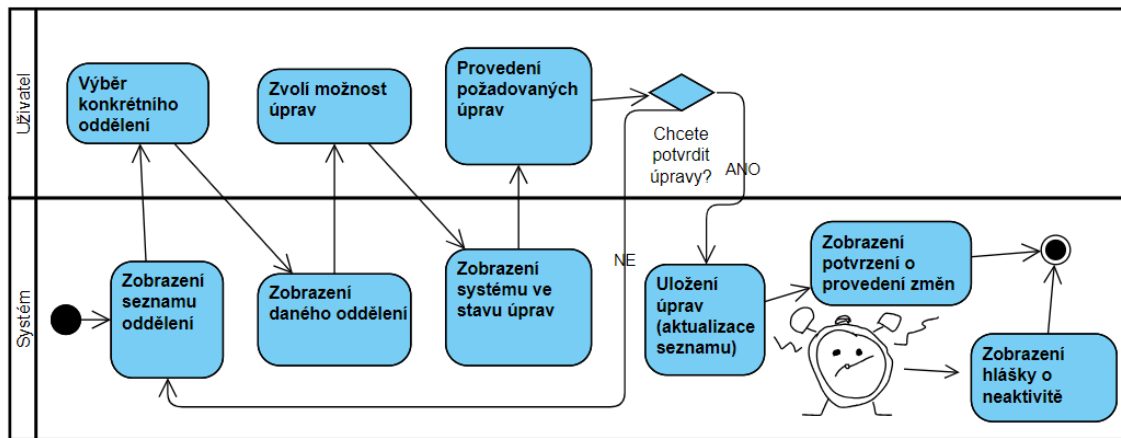
### 3. Artefakt - Use case diagram



#### Use case - Správa oddělení - úprava

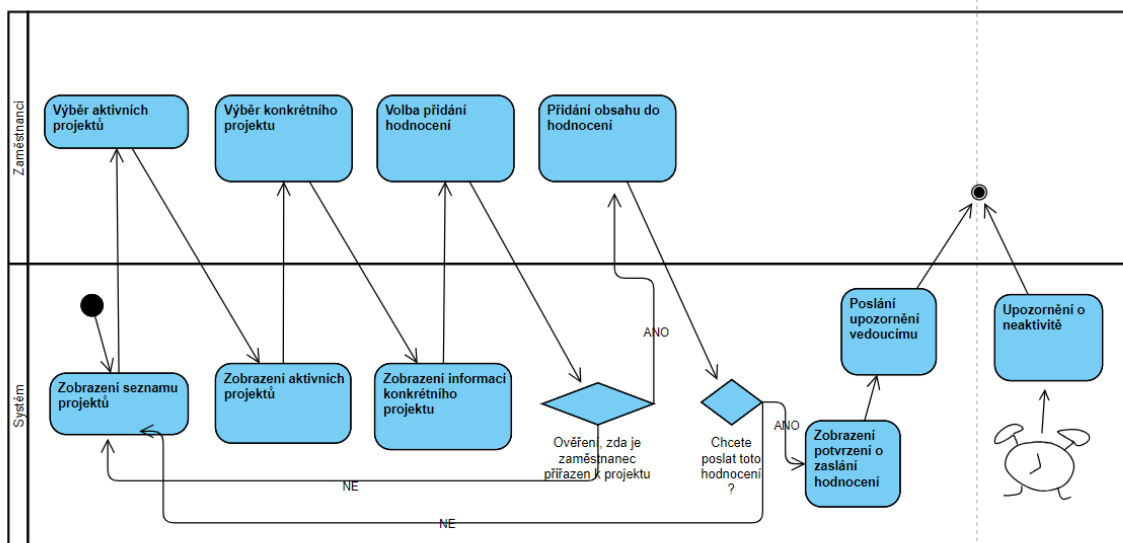
- **Aktéři:** Vedení města, správce systému
- **Předpoklady:** Uživatel je přihlášen do systému, uživatel má oprávnění k úpravě informací o odděleních
- **Základní tok:**
  1. Zaměstnanec zobrazí seznam všech oddělení
  2. Zaměstnanec vybere konkrétní oddělení k úpravě
  3. Zvolí možnost úprav
  4. Zaměstnanec provede úpravy informací o oddělení
  5. Zaměstnanec potvrdí provedené změny
- **Alternativní tok:**
  - Zaměstnanec zruší úpravy
    - Pokud zaměstnanec zruší úpravy, systém se vrátí na předchozí stav a neuloží žádné změny
    - Systém zobrazí potvrzení o zrušení zpráv
- **Výjimky:**
  - Dlouhá doba neprovedení změn
    - po 15 minutách neprovedení změn se zobrazí hláška o vypršení času a systém se navrátí k posledním uloženým datům
  - Nedostatečné oprávnění
    - Pokud zaměstnanec nemá dostatečné oprávnění, systém zobrazí chybovou hlášku "nemáte dostatečné oprávnění, v případě nutnosti se obraťte na správce systému"
    - Systém proces ukončí

- **Následná akce:** Po dokončení procesu systém aktualizuje seznam oddělení s nově upravenými informacemi.
- **Aktivitní diagram:**



### Use case - Hodnocení projektového postupu (Zaměstnanecký feedback)

- **Aktéři:** Zaměstnanci městského úřadu, externí zaměstnanci
- **Předpoklady:** Projekt je v aktivní fázi (není již ukončený, nebo ještě nezačal). Zaměstnanec má přístup k danému projektu a jeho hodnocení (je přiřazen k danému projektu).
- **Základní tok:**
  1. Zaměstnanec si zobrazí seznam projektů
  2. Zobrazí si aktivní projekty
  3. Vybere projekt, který chce hodnotit
  4. Zaměstnanec provede hodnocení projektu
  5. Zaměstnanec potvrdí své hodnocení
- **Alternativní tok:**
  - Zaměstnanec zruší úpravy
    - Pokud zaměstnanec zruší hodnocení, systém se vrátí na předchozí stav a neuloží žádné změny
    - Systém zobrazí potvrzení o zrušení zpráv
- **Výjimky:**
  - Nedostatečné oprávnění
    - Pokud zaměstnanec nemá dostatečné oprávnění, systém zobrazí chybovou hlášku "nemáte dostatečné oprávnění, v případě nutnosti se obraťte na vedoucího projektu"
    - Systém proces ukončí
- **Následná akce:** Po dokončení procesu systém uloží a zašle hodnocení vedoucímu projektu, vedoucí projektu dostane upozornění na obdrženou zpětnou vazbu.

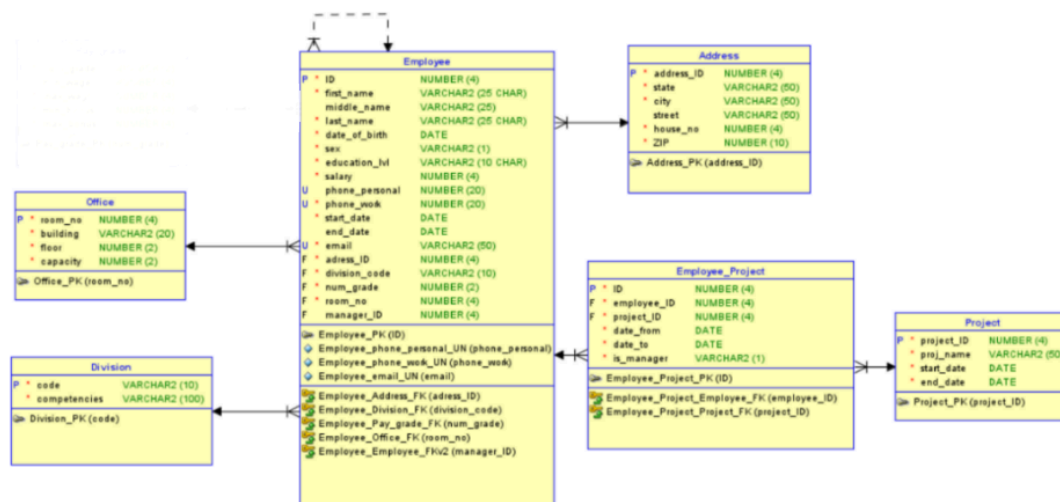


## Use case - Schvalování projektových návrhů

- **Aktéři:** Vedoucí pracovníci (vytváří návrh projektu), vedení (schvaluje navrhované projekty)
- **Předpoklady:** Zaměstnanec je přihlášený do systému a vytvořil návrh na nový projekt. Vedení má přístup k danému projektu (byl mu zaslán na schválení).
- **Základní tok:**
  1. Vedení dostává upozornění na žádost o schválení projektu
  2. Vedení si zobrazí daný návrh
  3. Vedení posoudí návrh
    - a. Schválení
    - b. Zamítnutí
    - c. Vrácení k úpravě, vložení komentáře (návrhu) pro úpravu, potvrzení odeslání úpravy
- **Alternativní tok:**
  - Vrácení k úpravě: Systém pošle upozornění zaměstnanci, která podal žádost, zaměstnanec rozklikne danou žádost, upraví návrh na základě komentáře vedení města, potvrdí úpravy a zašle žádost vedení města
- **Výjimky:**
  - Nedostatečné oprávnění
    - Pokud zaměstnanec nemá dostatečné oprávnění, systém zobrazí chybovou hlášku "nemáte dostatečné oprávnění, v případě nutnosti se obraťte na správce systému"
    - Systém proces ukončí
- **Následná akce:**
  - Žádost byla schválena: Systém přidá daný projekt do seznamu projektů a proces se ukončí
  - Zamítnutí: Systém pošle upozornění zaměstnanci, který podal žádost o zamítnutí jeho projektu



## Relační datový model



## Velikosti entit a jejich přibližný odhad

### Oddělení:

- Velikost záznamu: 5 kB
- Počet oddělení: 10
- Celkový potřebný prostor: 50 kB

### Projekty:

- Velikost záznamu: 10 kB
- Počet probíhajících projektů: 100
- Celkový potřebný prostor: 1000 kB

### Zaměstnanci:

- Velikost záznamu: 2 kB
- Počet zaměstnanců: 100
- Celkový potřebný prostor: 200 kB

### Veřejný přístup:

- Velikost záznamu: 1 kB (pro každý veřejný přístup)
- Celkový potřebný prostor: 100 kB (předpoklad 100 veřejných přístupů měsíčně)

### Address:

- Velikost záznamu: 1 kB
- Předpokládaný počet záznamů: 200 (Adresy pro zaměstnance a projekty)
- Celkový potřebný prostor: 200 kB

## Odhad počtu uživatelů současně pracujících se systémem.

Vzhledem k charakteru úřední práce a administrativních činností předpokládáme, že současně systém využívá přibližně 100 uživatelů. Tento odhad zahrnuje zaměstnance pracující na různých odděleních městského úřadu, včetně administrativy a vedení města. Tato čísla mohou být upravena v závislosti na skutečné potřebě a dynamice práce v konkrétním městském prostředí.

## Typy operací vyvolaných uživatelem a odhad náročnosti

Procházení seznamů:

- Typy operací: Prohlížení informací o odděleních, projektech, zaměstnancích, kancelářích
- Odhadovaná datová náročnost: Nízká - operace na čtení, které vyžadují minimální datový objem.

Vyhledávání:

- Typy operací: Hledání specifických informací o projektech, zaměstnancích nebo adresách.
- Odhadovaná datová náročnost: Nízká - operace na čtení, s možným využitím indexů pro optimalizaci vyhledávání.

Zadávání nových projektů:

- Typy operací: Přidání nových projektů do systému.
- Odhadovaná datová náročnost: Střední - operace zápisu nových záznamů do tabulky projektů.

Přidávání zaměstnanců k projektům:

- Typy operací: Přiřazení zaměstnanců k existujícím projektům.
- Odhadovaná datová náročnost: Střední - operace aktualizace dat v tabulce Employee\_Project.

Prohlížení a úprava dat zaměstnanců:

- Typy operací: Procházení a případná úprava informací o zaměstnancích.
- Odhadovaná datová náročnost: Nízká až střední - operace čtení a zápisu dat v tabulce zaměstnanců.

Celkově lze očekávat, že systém bude častěji využíván pro čtení informací než pro zápis nových dat. Datová náročnost závisí na konkrétních potřebách uživatelů a dynamice změn v systému. Uvedené odhady jsou pouze orientační a mohou být upraveny v průběhu provozu systému.

## Softwarová a hardwarová specifikace

Systém bude vyvíjen a implementován v programovacím prostředí IntelliJ Idea, použitý jazyk Java. Dále bude potřeba použití databáze.

**Systémové požadavky:** Server: Umístění a dostupnost v síti, datový prostor min. 50 MB, zabezpečení proti vniknutí zvenku. Klient: 1 GB RAM, operační systém Windows 7 a dál, připojení k síti.

### Rozložení systému:

Systém bude využívat tří vrstvou architekturu, tedy:

- Prezentační vrstvu
- Logickou vrstvu
- Datovou vrstvu



## 5. Artefakt - Skica uživatelského rozhraní

Náš systém městského úřadu byl navržen s důrazem na jednoduchý, avšak efektivní design, s hlavním zaměřením na maximální funkčnost a uživatelskou přívětivost. Tímto prostředím se snažíme vytvořit prostředí se snadným a intuitivním přístupem.

Jednoduchý design systému přispívá k rychlosti a efektivitě práce uživatelů. Minimalistický přístup eliminuje zbytečné prvky a zjednodušuje uživatelské rozhraní, což v konečném důsledku zvyšuje produktivitu a snižuje možnost chyb.

Zvolili jsme modrou barvu jako dominantní prvek designu, nejen kvůli její estetické přitažlivosti, ale také kvůli pozitivnímu vlivu na psychiku uživatelů. Modrá barva je známá svým uklidňujícím a profesionálním dojmem, což vytváří příjemné prostředí pro uživatele při práci s naším systémem.

Přihlášení do systému bude pro registrované uživatele a pro návštěvníky systému, kteří budou mít omezenou funkcionalitu (především pouze zobrazení seznamů), registrace do systému bude umožněna pouze se svolením administrátora systému a to s využitím unikátního klíče.

**SYSTEM**  
**MĚSTSKÉHO ÚŘADU**  
KOPRIVNICE

Registrace

Přihlásit se

Jsem návštěvník

Pro dokončení přihlášení do systému je vyžadováno vyplnění uživatelského jména a hesla. Pro vizuální diferenciaci jsou textová pole zvolena světlejší barvou, v nichž je lehce viditelný "text", který po prvním zadání postupně mizí. V případě hesla jsou zobrazovány pouze symboly, přičemž skutečný text je uživateli dostupný až po kliknutí na ikonu oka. Tlačítka i textová pole si zachovávají konzistentní barvu napříč celým systémem, což přispívá k jednotnému a profesionálnímu vizuálnímu dojmu uživatelů.

# PŘIHLÁŠENÍ

## MU KOPŘIVNICE

Uživatelské jméno

Text

Heslo

Text

Přihlásit se

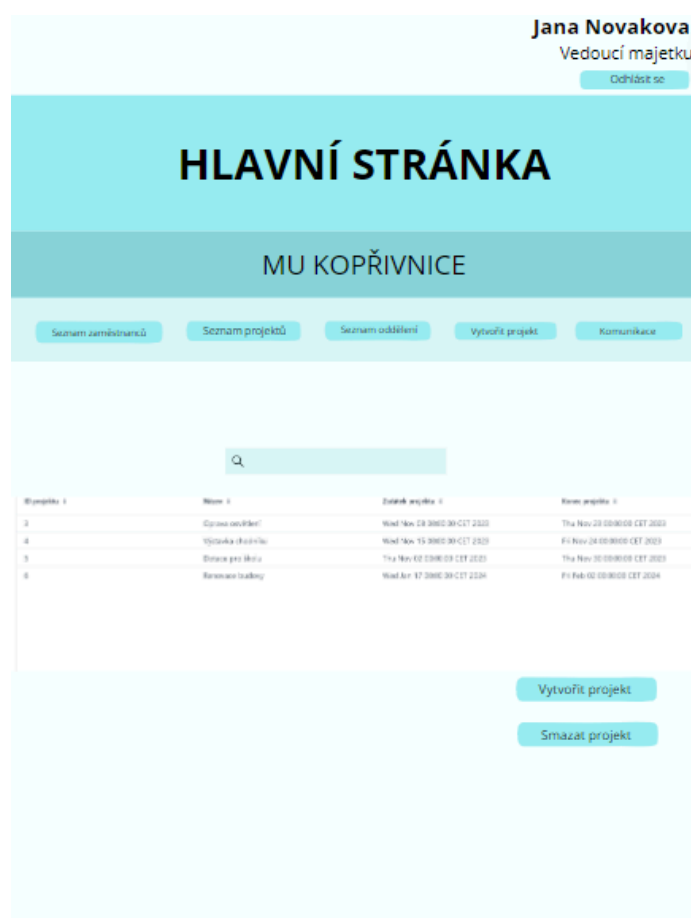
Po úspěšné registraci do systému bude uživatel přesměrován na hlavní stránku. Vpravo nahoře lze vidět uživatelské jméno a příjmení a také jeho funkce na daném MU. Hlavní strana nabídne uživateli přehledné menu klíčových funkcí systému, které jsou organizovány podle jejich relevance a uživatelských potřeb. Tato struktura umožňuje uživatelům snadný přístup k nezbytným úkolům a funkcím, přičemž samotné menu je flexibilní a přizpůsobitelné, umožňující personalizaci podle individuálních preferencí a pracovních požadavků. V pravém dolním rohu stránky je umístěna ikona bubliny s tečkami, symbolizující komunikační kanál. V případě, že uživatel obdrží nové zprávy, se nad touto ikonou automaticky zobrazí červené kolečko s číslem, které indikuje počet nepřečtených zpráv. Tato inovativní funkce nenahrazuje klasické pracovní e-maily, ale spíše slouží jako efektivní nástroj pro rychlou a plynulou komunikaci mezi zaměstnanci, čímž přispívá k optimalizaci vnitřního pracovního procesu.



Po výběru konkrétní funkcionality systému bude uživatel okamžitě přesměrován na příslušnou stránku. Můžeme si všimnout, že vždy zůstává nedotčená hlavička systému, obsahující relevantní informace o uživateli, název městského úřadu a nadpis aktuální stránky, spolu s dostupným menu pro snadnou navigaci.

Všechny seznamy budou koncipovány tak, aby vizuálně odpovídaly jednotnému formátu, což podporuje uživatelskou konzistenci a snazší orientaci. Uživatelé budou mít možnost v seznamu vyhledávat, či třídit informace abecedně, případně podle velikosti čísel.

V pravém horním rohu pod tabulkou budou umístěna tlačítka umožňující úpravu tabulky, jako vytvoření nové položky nebo odstranění existujících. Tato rozšířená tlačítka budou viditelná pouze uživatelům s odpovídajícími oprávněními, což zajišťuje bezpečnost a kontrolu nad editačními funkcemi.

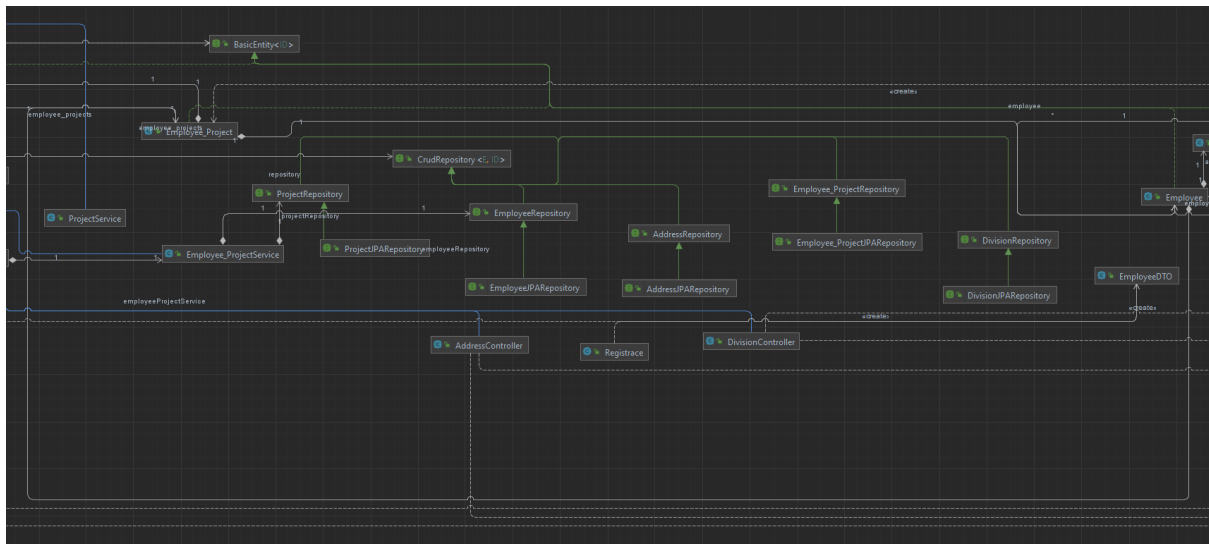
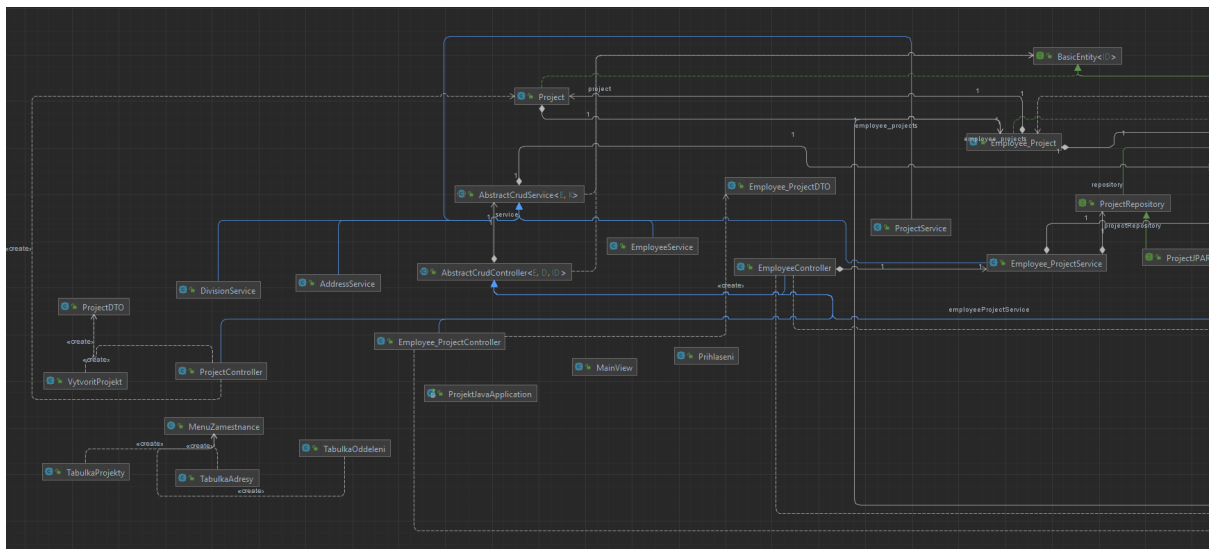
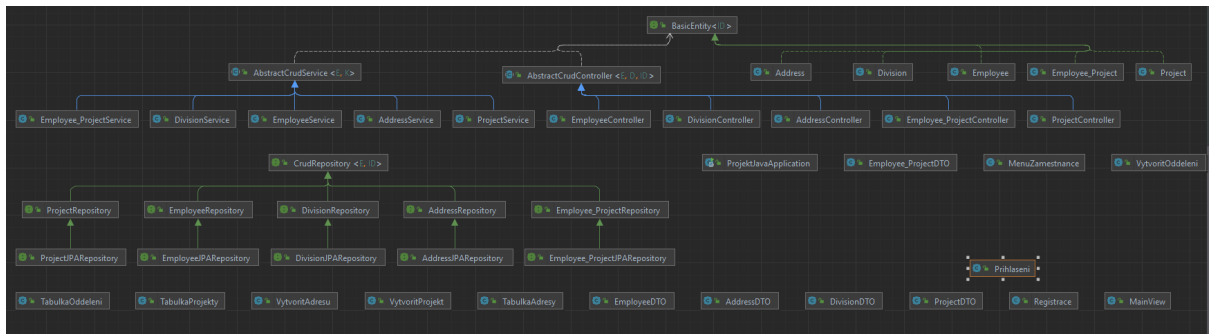


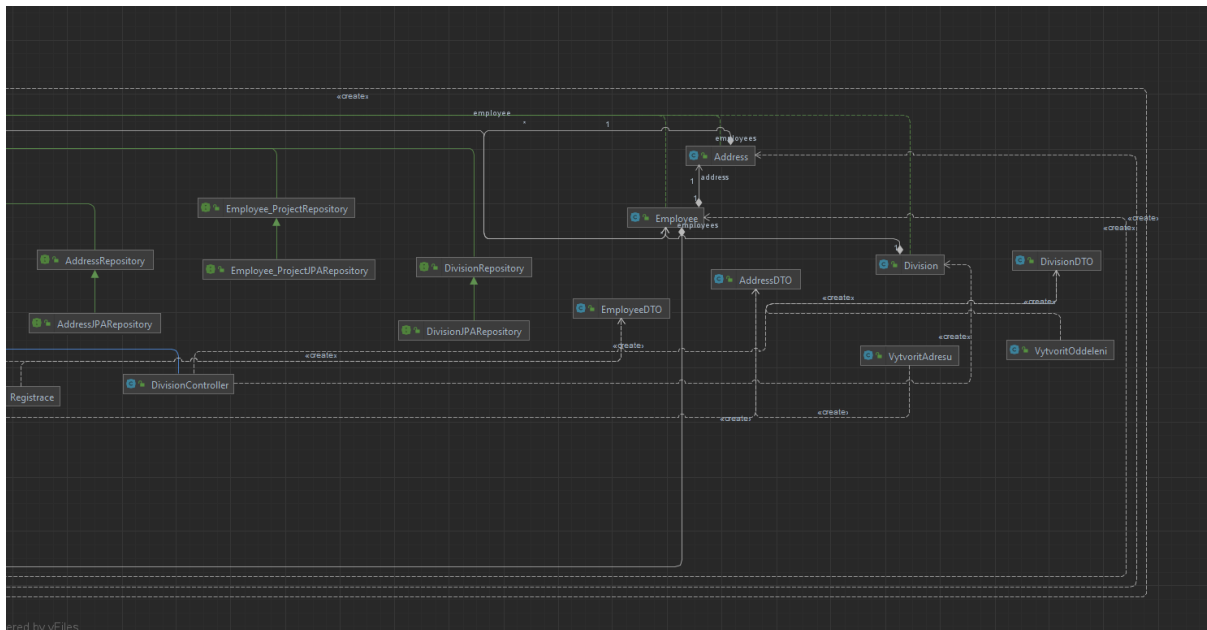
Při interakci s dalšími funkcemi, jako je přidávání projektu, zůstává estetika a uspořádání prvků shodné s designem popsáním výše při přihlašování uživatele.

Při přidávání nového projektu se zobrazují textová pole s podobným vizuálním vzhledem jako v případě přihlašovací stránky. Tímto zůstává uživatelská zkušenost jednotná a intuitivní, což přispívá k plynulému a bezproblémovému používání systému.

Procedura smazání projektu je realizována prostřednictvím vložení identifikátoru (ID) příslušného projektu. Tento krok zajišťuje bezpečnost a přesnost, eliminuje možnost chyb a zůstává v souladu s celkovým designem, což přináší uživatelům jednotný a profesionální dojem při interakci s různými částmi systému.

## 6. Návrh doménového modelu

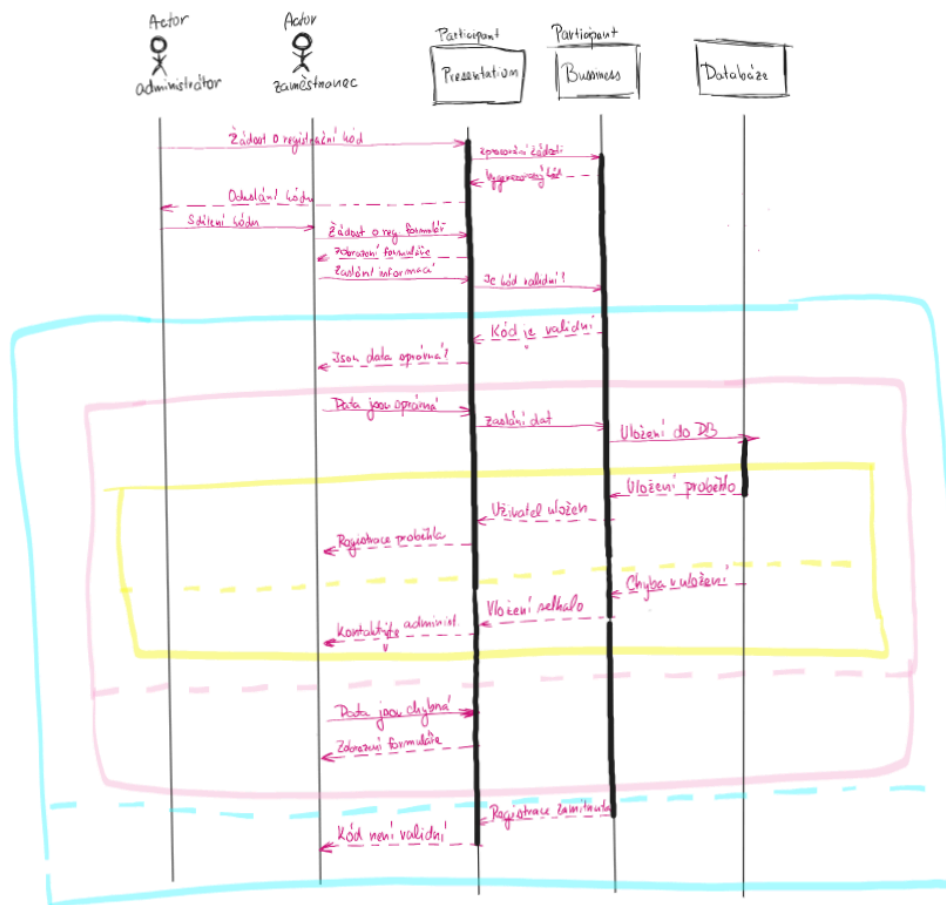




Návrhové vzory využit v projektu:

- Třívrstvá architektura
  - Tento pattern zajišťuje systematické oddělení 3 vrstev (prezentační, aplikační, datová vrstva). Tímto způsobem dosáhneme oddělení odpovědností, což zlepšuje modularitu a udržitelnost kódu.
- Data Transfer Objects (DTOs)
  - Využití můžeme vidět například na EmployeeDTO, pro přenos dat mezi různými vrstvami naší aplikace. Tento vzor izoluje prezentaci od implementačních detailů v business logice a datové vrstvě. To nám umožňuje snadnou komunikaci mezi vrstvami a minimalizuje závislosti.
- Unit of Work
  - Pro řízení transakcí a udržení konzistentního stavu v databázi implementujeme vzor Unit of Work. To je zejména užitečné při správě transakcí v kontextu s databázovým úložištěm, konkrétně EmployeeJPARespository.
- Repository Pattern
  - Implementujete Generic Repository Pattern prostřednictvím EmployeeJPARespository a EmployeeRepository rozhraní. Tento vzor umožňuje jednotný přístup k práci s databází pro různé entity.
- Decorator Pattern
  - Umožňuje dynamicky přidávat novou funkcionalitu existujícím objektům bez změny jejich kódu. Decorator Pattern je užitečný v situacích, kdy je třeba přidat nebo odstranit funkcionalitu dynamicky a bez ovlivnění existujících tříd.

Dále se využívá dependency Injection pro správu závislostí mezi různými komponentami aplikace. Tato technika zvyšuje modularitu a usnadňuje testování našeho kódu.



Sekvenční diagram zachycuje průběh registrace nového zaměstnance do systému. První fází je, že je potřeba vygenerovat administrátorem unikátní kód pro zaměstnance, který mu umožní registraci do systému. Použití kódu znamená, že můžeme nechat samotnou registraci, jakožto vyplnění vstupních informací na zaměstnanci, čímž jsme schopni ušetřit časovou náročnost celého procesu. Zaměstnanci je zobrazen registrační formulář, společně s políčkem pro kód, jež byl přiřazen a sdělen zaměstnanci. Při odeslání formuláře se nejdříve ověří přístupový kód. Pokud kód souhlasí, proces pokračuje, pokud ověření z nějakého důvodu selže, vznikne nám alternativní vlákno.

Po úspěšné ověření systém požádá o ověření informací, které jsou zpracovány a uloženy v databázi. Při tomto postupu může dojít k dvou odděleným tokům a to při chybně zadaných parametrech, nebo při selhání uložení do databáze.

## 6. Artefakt - Popis architektury systému

Klíčové komponenty zahrnují desktopovou aplikaci, webovou aplikaci, aplikační server, data mapper a MySQL server.

### 1. Desktopová aplikace:

- **Popis:** Desktopová aplikace poskytuje uživatelům přímý přístup k informačnímu systému na stolních počítačích v rámci městského úřadu. Uživatelé mohou prostřednictvím této aplikace procházet, vyhledávat a upravovat informace o odděleních, projektech a zaměstnancích.
- **Interakce:** Komunikuje s Aplikačním serverem - desktopová aplikace využívá aplikační server pro získávání a aktualizaci dat.

### 2. Webová aplikace:

- **Popis:** Webová aplikace umožňuje uživatelům přístup k informačnímu systému prostřednictvím webového prohlížeče. To poskytuje možnost vzdáleného přístupu k informacím, což je užitečné pro zaměstnance, kteří potřebují pracovat mimo kancelář.
- **Interakce:** Komunikuje s Aplikačním serverem- webová aplikace využívá aplikační server pro získávání a aktualizaci dat.

### 3. Aplikační server:

- **Popis:** Aplikační server je centrální částí systému, která zpracovává business logiku a interaguje s datovou vrstvou. Zajišťuje správu transakcí, zpracování požadavků od prezentačních vrstev a manipulaci s daty v databázi.
- **Interakce:**
  - Komunikuje s Desktopovou a Webovou aplikací: Přijímá požadavky od obou aplikací a poskytuje odpovídající data.
  - Spolupracuje s Data Mapperem: Aplikační server využívá Data Mapper k efektivní manipulaci s daty v databázi.

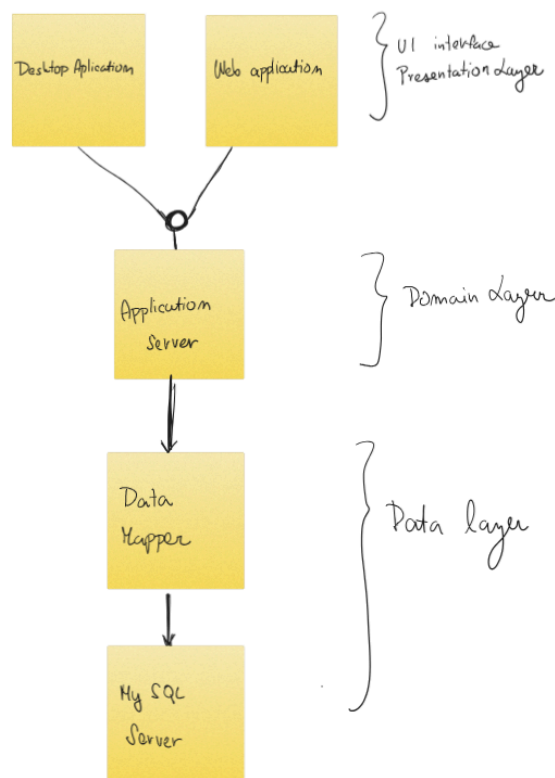
### 4. Data Mapper:

- **Popis:** Data Mapper slouží jako rozhraní mezi aplikačním serverem a databází. Zajišťuje mapování objektů na relační data a umožňuje efektivní operace čtení a zápisu do databáze.
- **Interakce:**
  - Komunikuje s Aplikačním serverem: Přenáší data mezi aplikačním serverem a MySQL serverem.

### 5. MySQL Server:

- **Popis:** MySQL server představuje databázový systém, kde jsou uložena všechna data související s informačním systémem pro městský úřad. Obsahuje tabulky pro oddělení, projekty, zaměstnance a další relevantní informace.
- **Interakce:**
  - Komunikuje s Data Mapperem: Přijímá a odesílá data prostřednictvím Data Mapperu.





Tento diagram komponent ilustruje propojení klíčových částí informačního systému, které společně umožňují efektivní správu a sdílení informací v rámci městského úřadu.

Windows a Linux PC:

- Komentář: Kombinace Windows a Linux počítačů umožňuje flexibilitu pro uživatele s různými preferencemi a potřebami. Je důležité zajistit, aby všechny aplikace a služby byly dostupné na obou platformách nebo byly poskytnuty ekvivalentní náhrady.

Desktopová aplikace a Webová aplikace:

- Komentář: Nabídka jak desktopové, tak webové aplikace poskytuje uživatelům možnost volby podle svých potřeb. Desktopová aplikace může poskytnout pokročilé funkce a rychlejší odezvu, zatímco webová aplikace umožňuje přístup z různých zařízení a míst.

Aplikační server:

- Komentář: Centrální aplikační server slouží jako místo pro zpracování business logiky a interakce s databází. Je důležité zajistit, aby byl server dostatečně výkonný a škálovatelný v případě zvýšeného provozu.

Microsoft SQL Databáze:

- Komentář: Používání Microsoft SQL Serveru pro ukládání dat je rozumná volba, zejména pokud jste ve Windows prostředí. Databáze by měla být správně navržena a optimalizována pro potřeby aplikace.

