***Specifikace požadavků na Software (SRS) pro webovou aplikaci MyProperty***

1. **Úvod**
   1. **Účel Dokumentu**

Tento dokument, Specifikace softwarových požadavků (SRS), popisuje funkční a nefunkční požadavky, omezení a další důležité aspekty webové aplikace MyProperty, která je v současné době ve vývojové fázi. Dokument slouží jako podrobný návod a referenční materiál pro vývojáře, projektové manažery a všechny další zúčastněné strany. Současně definuje rozsah projektu a působí jako smluvní závazek mezi zadavatelem a vývojovým týmem.

Je důležité zdůraznit, že vzhledem k tomu, že se aplikace nachází ve fázi vývoje, mohou být určité funkce a procesy ještě nedostatečně definovány nebo neoptimalizované. Očekává se, že během vývojového procesu dojde k identifikaci a adresaci potenciálních nedostatků a rizik. Zadavatel si je vědom možných omezení stávajících řešení a uvítá jakékoli doporučení a zlepšení, která budou směřovat k efektivnější, bezpečnější a uživatelsky přívětivější aplikaci.

* 1. **Definice, Akronymy a Zkratky**

SRS (Specifikace Požadavků na Software): Tento dokument je klíčovou částí ve vývojovém procesu. Obsahuje podrobný popis funkčních a nefunkčních požadavků na software, jenž by měl být vyvinut. Jeho cílem je zajistit, aby vývojáři měli jasné pokyny, co mají vytvořit, a aby si zadavatel mohl být jistý, že konečný produkt splní jeho očekávání.

UI (Uživatelské Rozhraní): UI se zaměřuje na vizuální aspekty aplikace, jak má vypadá a jak uživatel s aplikací interaguje na základě viditelných prvků, jako jsou tlačítka, ikony, texty a grafické prvky.

UX (Uživatelská Zkušenost): UX pokrývá celkovou zkušenost uživatele při používání aplikace, což zahrnuje široké spektrum zážitků a interakcí, které uživatelé prožívají při používání produktů nebo služeb. Nejedná se pouze o interaktivní nebo vizuální aspekty, ale také o celkové vnímání, pocity a reakce, které produkt nebo služba vyvolává.

CRUD (Create, Read, Update, Delete – Vytvořit, Číst, Upravit, Smazat): Tento akronym označuje čtyři základní operace, které lze provádět na datech. V kontextu webových aplikací jde o běžné operace, které uživatelé mohou provádět na informacích uložených v databázi prostřednictvím UI aplikace.

API (Aplikační Programovací Rozhraní): API umožňuje dvěma softwarovým aplikacím komunikovat mezi sebou pomocí sady definovaných metod a protokolů. Ve webové aplikaci se často používají API pro komunikaci mezi front-endem (klientem) a back-endem (serverem) nebo pro integraci s externími službami.

* 1. **Odkazy**

*Spring Framework Documentation:*

Oficiální dokumentace pro Spring Framework, která poskytuje kompletní informace o všech aspektech frameworku, včetně jeho modulů pro vytváření web aplikací, přístupu k datům, transakcí a mnoho dalšího. Spring je základní technologie použitá v projektu pro vývoj serverové části aplikace.

*Hibernate ORM Documentation:*

Detailní dokumentace pro Hibernate ORM, framework pro mapování objektů do relační databáze, což umožňuje efektivní práci s daty v aplikaci. V projektu se využívá pro mapování a manipulaci s datovými entitami.

*MySQL Documentation:*

Rozsáhlá dokumentace k databázovému systému MySQL, který je používán jako primární databázové řešení pro správu dat aplikace. Dokumentace nabízí návody, referenční materiály a pokyny pro optimální využití MySQL v rámci projektů.

* 1. **Přehled**

MyProperty je webová aplikace navržená pro efektivní správu majetku, která je vhodná jak pro jednotlivé živnostníky, tak pro korporátní prostředí. V současné fázi se aplikace zaměřuje primárně na potřeby živnostníků, kteří potřebují efektivně spravovat svůj majetek bez nutnosti řízení zaměstnanců. Pro tuto cílovou skupinu aplikace poskytuje robustní nástroje pro evidenci, správu a sledování majetku.

V další fázi vývoje, plánované do druhého ročníku, bude aplikace rozšířena o moduly pro správu zaměstnanců. Tyto moduly umožní interní komunikaci mezi zaměstnanci, podávání žádostí o majetek a nahlášení poškození u majetku. Zaměstnanci také budou mít přehled o majetku, za který jsou zodpovědní, přímo ve svém uživatelském profilu.

Navíc, v rámci tohoto rozšíření bude implementována funkce pro plánování inventury majetku, včetně možnosti zaznamenání plánů do integrovaného kalendáře. Tento modul bude také zaveden v druhém ročníku vývoje.

Cílem aplikace MyProperty je poskytnout uživatelům komplexní, intuitivní a přizpůsobitelné řešení pro správu majetku, které usnadní administrativní zátěž a zvýší produktivitu jak jednotlivců, tak týmů ve větších organizacích.

1. **Celkový popis**
   1. **Perspektiva aplikace**

MyProperty je nezávislá webová aplikace, určená do firemní infrastruktury. Tento systém je navržen tak, aby poskytoval řešení pro správu majetku, které lze jednoduše přizpůsobit firemním potřebám. V základní konfiguraci aplikace nabízí komplexní nástroje pro evidenci, sledování změn u majetku a správu fyzického majetku, což zahrnuje kategorizaci majetku, zaznamenávání změn a uchovávání údajů o fyzickém majetku plus možnost uložit faktury či obrázky, což přispívá k zvýšení efektivity správy majetku.

V budoucnu bude aplikace rozšířena o pokročilé moduly pro správu uživatelů, které umožní firmám efektivně spravovat své zaměstnance. Toto rozšíření bude zahrnovat funkci pro interní komunikaci mezi zaměstnanci. Dalším rozšířením této aplikace bude přidání funkce pro nahlašování poškození u majetku nebo podávání žádostí oddaný majetek, všechny tyto požadavky bude vyřizovat IT správce nebo majitel firmy. Posledním významným plánovaným rozšířením bude modul pro plánování inventur v integrovaném kalendáři společně s modulem pro správu notifikací.

V konečné fázi vývoje by měla aplikace poskytnout robustní systém pro správu majetku, zaměstnanců, notifikací, interní komunikace, vyřizování žádostí o majetek nebo plánování inventury.

* 1. **Funkce Aplikace**
* Evidence majetku: Zaznamenávání všech aktiv s logováním změn v kartě majetku
* Správa zaměstnanců: Záznam a správa údajů o zaměstnancích a jejich rolí
* Interní komunikace: Posílání zpráv mezi zaměstnanci
* Kalendář: Plánování událostí a úkolů s vazbou na majetek (plánování inventur)
* Žádosti o majetek: Možnost zaslání žádosti o majetek či nahlášení závady pro každého zaměstnance v rámci firmy
  1. **Charakteristika uživatelů**
* Vlastník (Owner): Správce systému s nejvyššími oprávněními.
* Správce (Administrator): Má rozsáhlá práva pro správu majetku a uživatelů.
* Zaměstnanec (Employee): Standardní uživatel systému s omezenými právy.
  1. **Omezení**

Aplikace MyProperty bude vyvinuta v jazyce Java. Základem backendu bude framework Spring Boot, který umožňuje efektivní a rychlý vývoj mikroslužeb s minimální konfigurací.

Pro mapování objektů na databázové tabulky bude použito Hibernate ORM, což je mocný nástroj pro manipulaci s daty v databázi, který poskytuje transparentní abstrakci datové vrstvy.

Frontend aplikace bude realizován s použitím Thymeleaf, moderního šablonovacího engine pro Java, který umožňuje vytvářet dynamicky generované webové stránky. Díky Thymeleaf bude uživatelské rozhraní responzivní a přizpůsobí se různým zařízením a velikostem obrazovek, což zaručí pohodlné uživatelské zkušenosti na jakémkoli zařízení.

Využití MySQL databáze přináší omezení spojená s relačními databázemi, včetně náročnosti na údržbu schémat a obtížnější škálovatelnosti v porovnání s NoSQL databázemi, zvláště pokud aplikace vyžaduje extrémně vysokou dostupnost a rychlost zápisu.

1. **Detailní požadavky**
   1. **Funkční požadavky**
      1. **Modul Evidence majetku**

* Crud operace pro majetek s povolením na základě role uživatele
* Kategorizace majetku podle typu kategorie a umístění
* Historie změn a přesunů každé položky
* Možnost evidovat až tři obrázky ke každému majetku ve formátu png, jpg, jpeg
* Možnost přidat profilový obrázek každého majetku ve formátu png, jpg, jpeg
* Možnost evidovat až dva dokumenty pdf (faktury)
* Možnost stáhnout pdf dokumenty nebo je zobrazit v novém okně
  + 1. **Modul autentizace a autorizace uživatelů**
* Možnost pro nové uživatele zaregistrovat se do systému, včetně ověření přes email
* Při zapomenutí hesla možnost odeslat na email odkaz s obnovením hesla
* Možnost změnit heslo
* Bezpečný přístup pomocí hesla a emailu (šifrovaného hesla)
* Možnost odhlásit se
* Omezení přístupu na základě rolí owner, admin a user
  + 1. **Modul správy zaměstnanců (rozšíření v druhém ročníku) a komunikace**
* Přidání, mazání a úpravy zaměstnanců pro uživatele s rolí owner nebo admin
* Sledování využití majetku zaměstnanci
* Možnost zasílání zpráv mezi zaměstnanci, interní chat pro zasílání zpráv
* Sledování oznámení a upozornění
  + 1. **Modul kalendář a žádosti o majetek (rozšíření v druhém ročníku)**
* Vytváření, úpravy a zobrazení událostí
* Možnost plánovat inventury
* Přidání úkolů a připomínek
* Podání a schvalování žádostí o majetek, či nahlašování závad u majetku
  1. **Nefunkční požadavky**
* Rychlá odezva systému a minimální doba načítání
* Systém musí být dostupný 24/7 s výjimkou pravidelných údržb
* Použití šifrování a bezpečného uložení hesel
* Zabezpečení proti různým hrozbám jako je CSRF nebo útokům na systém
* Podpora všech hlavních prohlížečů a responzivní design
* Intuitivní uživatelské rozhraní s přístupnými návody pro uživatele
  1. **Externí rozhraní**
* RESTful API pro integraci s externími systémy (je potřeba optimalizovat)
* Možnost importu a exportu dat ve formátech CSV a Excel (připraveno pro vývoj do druhého ročníku)
* Generování tiskových sestav reportů pro inventuru a správu majetku (v druhém ročníku)

1. **Databázový model**

* Každý uživatel má přiřazenou jednu z rolí (owner, admin nebo employee)
* Uživatel může být zařazen pouze do jedné firmy (aby mohl být v další firmě, tak musí využít jiný email)
* Firma může obsahovat více umístění majetku ve firmě například kanceláře nebo sklad
* Na každém umístění může být více firemního majetku
* Každý majetek je zařazen do své kategorie například počítače, auta, nářadí
* Každý majetek může obsahovat soubory jako jsou obrázky či faktury
* Soubor je označen určitým typem pro lepší manipulaci například images, invoices nebo profilImage
* U každého majetku jsou zaznamenávány všechny změny a uloženy do tabulky property\_history
* Každá změna je označena určitým typem pro lepší správu například inventory number, …

Obsah obrázku text, snímek obrazovky, diagram, Písmo

Popis byl vytvořen automaticky

1. **Shrnutí**

Během vývoje webové aplikace MyProperty, zaměřené na evidenci majetku, jsem se setkal s řadou odborných výzev. Tyto výzvy spočívaly primárně v samostudiu backendového programovacího jazyka Java s využitím frameworku Spring Boot. Původně jsem očekával, že se tyto znalosti získám z přednášek, cvičení z předmětu tvorba webových aplikací I., avšak realita byla jiná, což mě vedlo k samostudiu prostřednictvím dostupných online zdrojů.

Přestože jsem během procesu vývoje narazil na technické problémy a některé části kódu nebyly zpočátku optimálně řešeny, tato zkušenost mi poskytla cenné poznatky. Uvědomil jsem si, že některé segmenty aplikace by bylo vhodné vzít a přepracovat pro zvýšení efektivity a čitelnosti kódu.

Díky intenzivní práci s Java Spring Boot a dalšími technologiemi jako Hibernate, Lombok a Thymeleaf jsem si výrazně rozšířil své technické dovednosti a zlepšil svou schopnost řešit složité vývojové úkoly. Celkově mi tato zkušenost umožnila zlepšovat se v oblasti vývoje aplikací od návrhu až po samotné zkompletování celé aplikace v jazyce Java s využitím frameworku spring boot.