GymNest-IS

Dokumentace projektu

Autoři: Bc. Matěj Boura a Bc. Jan Sakač

# Manažerské shrnutí

Tento dokument slouží jako manažerské shrnutí pro zavedení systému GymNest, který je zaměřen na digitalizaci správy a rezervace fitness služeb. Projekt GymNest přináší komplexní řešení pro správu členství, rezervace tříd a správu uživatelů ve fitness centrech nebo tělocvičnách.

**Cíle projektu**

Automatizace rezervačních a správních procesů sníží čas strávený manuálními úkoly a umožní personálu věnovat se přímé interakci s klienty. Zároveň si klienti mohou sami spravovat lekce a zajišťovat si předplatné. Zavedení uživatelsky přívětivé platformy pro klienty, která umožní snadné rezervace tříd, sledování členství a udržení motivace prostřednictvím pravidelných aktualizací a interakcí. Efektivnější, transparentní a jednoduchá správa předplatného může přilákat nové klienty a udržet stávající klienty, což vede k vyššímu využití služeb a zvýšení příjmů.

**Klíčové vlastnosti systému**

* Integrace rezervačního systému: Centrální systém pro správu rezervací a členství pro uživatele, který je přístupný online.
* Responzivní aplikace: Aplikace pro snadný přístup klientů k jejich účtům a dalším funkcím pro různá zařízení.
* Notifikační systém: Automatické upozornění pro klienty ohledně blížících se tříd a členských obnov předplatných.

**Očekávané výsledky**

* Optimalizace procesů: Značné zlepšení v rychlosti a přesnosti správních procesů.
* Zvýšení spokojenosti klientů: Větší spokojenost klientů díky snadnému přístupu k službám a lepší podpoře.
* ROI: Očekává se rychlý návrat investice díky zvýšení efektivity a příjmů.

Projekt je navržen tak, zaplnil místo na trhu se systémy pro správu fitness center. Tento druh softwaru je sice běžný mezi fitness centry ve Spojených státech, ale na Evropském kontinentě není nijak rozšířená i přes fakt, že se provozují centra ve stylu USA. Projekt je realizován ve spolupráci s existujícími a ověřenými softwary a technologickými řešeními, což zajistí dodržení nejnovějších standardů a bezpečnostních praxí.

# Věcná část projektu

Projekt GymNest-IS představuje řešení pro digitalizaci a automatizaci procesů spojených s provozem fitness center a tělocvičen. Tento systém je navržen tak, aby odpovídal rostoucím požadavkům moderních fitness zařízení na efektivitu, přístupnost a uživatelskou přívětivost. Cílem této věcné části je poskytnout ucelený přehled o projektu a jeho funkčnostech, aby bylo možné plně ocenit hloubku a šíři plánovaných řešení a jejich potenciální dopad na dennodenní procesy a zážitek uživatelů.

## Funkční analýza

Tento segment dokumentace umožňuje hlubší pochopení toho, jak systém interaguje s koncovými uživateli a jaké operace může provádět. Cílem funkční analýzy je jasně definovat a popsat všechny operace, které systém umožňuje, včetně interakce mezi uživateli a systémem a mezi samotnými systémovými komponentami.

### Big Picture, Úvod

Projekt GymNest se zaměřuje na modernizaci a automatizaci procesů spojených s provozem fitness centra. Systém integruje klíčové funkce potřebné pro efektivní správu zákazníků, rezervací tříd, správu členství a komunikaci. GymNest je rozdělena na čtyři samostatné služby (architektura mikroslužeb), a to:

* user-service (Uživatelská služba)
* membership-service (Členská služba)
* booking-service (Rezervační služba)
* Frontend

**Uživatelská služba**

Uživatelská služba má na starosti vše spojené s uživatelem, jeho přihlášení, registraci, definování jména, příjmení, emailu, datumu narození a jeho preferencí. V této službě se také definují role (uživatel, člen, trenér a administrátor). Kromě zmíněných se také stará o autentizaci, ať už pro běžné přihlášení nebo přihlášení přes externí službu Google. Dohromady služba nabízí komplexní správu autentizace, rolí a uživatelů přes širokou paletu dostupných operací.

**Členská služba**

V této službě se definuje členství v systému. Je celá řada předdefinovaných členství, ale systém umožňuje vytvářet také vlastní typ členství. Samotné členství představuje možnost k připojení se k aktivitám, které nabízí dané sportovní nebo fitness centrum. Následně tato služba obsahuje logiku pro platby a vše s tím spojené. Umožňuje výpis historie plateb, status platby (v případě problémů), popis a případné spojení s konkrétním typem členství. To vše usnadňuje práci s financemi.

**Rezervační služba**

Rezervační služba definuje jednotlivé aktivity, rezervace, notifikační logiku a také správu kalendáře. Aktivity představují sporty nebo cvičení (jóga, box, pilates, …), ke kterým se členové mohou přihlásit prostřednictvím další komponenty rezervace. Členové s příslušný typem členství se mohou přihlásit k vypsané aktivity a tím si jí rezervovat. Po rezervaci se spustí notifikační logika, která má za úkol upozornit uživatele na vytvoření rezervace a na její zahájení s určitým předstihem.

**Frontend**

++

### UC model

**Uživatelské role**

* Uživatel: Registrovaný člověk v systému, potencionální člen
* Člen: Může procházet třídy, rezervovat a zrušit lekce, sledovat historii svých aktivit a obnovovat členství.
* Trenér: Má možnost spravovat své třídy, zaznamenávat účast na třídách a komunikovat s členy.
* Administrátor: Zodpovídá za správu uživatelských účtů, nastavení tříd, ceníků a monitorování celkového provozu centra.

#### Přihlášení a registrace s nákupem členství

***Popis:*** Nový uživatel chce vstoupit do systému GymNest a zakoupit si členství.

***Aktéři:*** Nový uživatel

***Předpoklady:*** Uživatel má přístup k internetu a webovému prohlížeči.

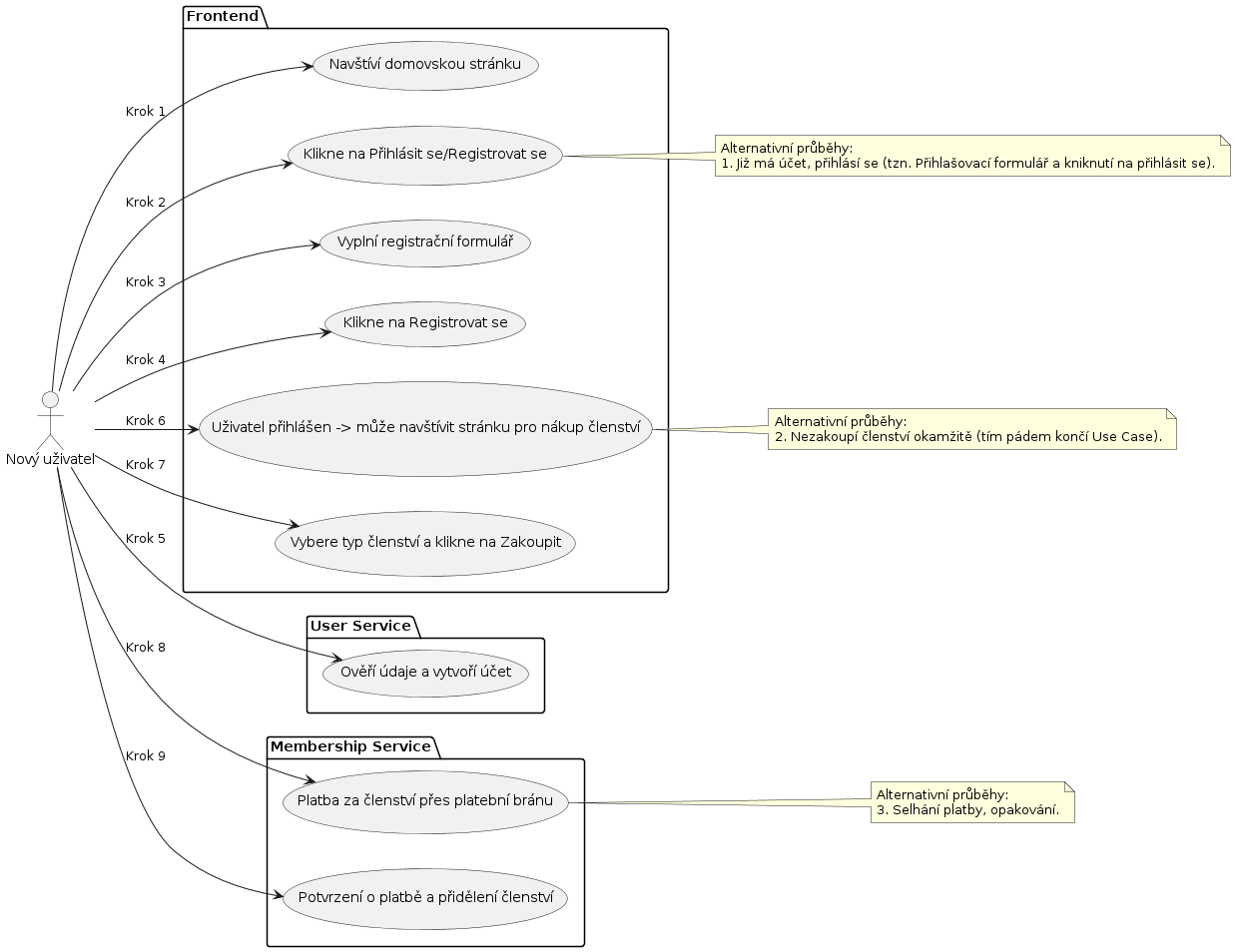
***Postup:***

1. Uživatel přistupuje k domovské stránce systému GymNest.
2. Uživatel klikne na možnost "Přihlásit se" nebo "Registrovat se".
3. Pokud uživatel ještě nemá účet, vyplní registrační formulář s potřebnými informacemi (jako je jméno, e-mail a heslo).
4. Po vyplnění formuláře uživatel klikne na tlačítko "Registrovat se".
5. Systém ověří zadané údaje a vytvoří nový účet pro uživatele.
6. Uživatel je automaticky přihlášen do systému a přesměrován na stránku pro nákup členství.
7. Uživatel vybere požadovaný typ členství a klikne na možnost "Zakoupit".
8. Systém přesměruje uživatele na platební bránu, kde provede platbu za vybrané členství.
9. Po úspěšné platbě je uživateli automaticky přiděleno zakoupené členství a potvrzení o platbě je zasláno na jeho e-mail, zároveň mu systém přiřadí oprávnění: člen.

***Alternativní průběh:***

* Pokud uživatel již má účet, přihlásí se pomocí svých přihlašovacích údajů a pokračuje od kroku 6.
* Pokud se uživatel rozhodne nezakoupit členství okamžitě, může zvolit možnost prohlížení rozvrhu lekcí nebo jiných funkcí aplikace.
* Pokud platba za členství selže, systém uživatele o tom informuje a nabízí možnost opakování platby.

***Znázornění***

******

#### Zobrazení rozvrhu a rezervace lekce

***Popis*:** Registrovaný uživatel si chce zobrazit rozvrh skupinových lekcí a rezervovat si místo na vybrané lekci.

***Aktéři*:** Registrovaný uživatel

***Předpoklady*:** Uživatel je registrován do systému.

***Postup*:**

1. Uživatel se přihlásí do systému pomocí svého uživatelského jména a hesla.
2. Po úspěšném přihlášení uživatel se nachází na domovské stránce aplikace GymNest.
3. Uživatel vybere možnost "Zobrazit rozvrh" z hlavního menu.
4. Systém zobrazí rozvrh skupinových lekcí pro daný týden, který obsahuje informace jako datum, čas, typ lekce a instruktora.
5. Uživatel prochází dostupné lekce a vybírá tu, na kterou se chce zúčastnit.
6. Po výběru konkrétní lekce uživatel klikne na možnost "Rezervovat místo".
7. Systém ověří dostupnost místa na lekci a potvrdí rezervaci.
8. Uživatel obdrží potvrzení o rezervaci a má místo na vybrané lekci zajištěno.

***Alternativní průběh:***

* Pokud uživatel nenajde vhodnou lekci v aktuálním týdnu, může se podívat na rozvrh následujících týdnů.
* Pokud je lekce již plně obsazena, uživatel nemůže provést rezervaci a musí vybrat jinou lekci.
* Uživatel může také procházet rozvrh lekcí podle typu (např. aerobic, spinning) nebo instruktora.

***Znázornění***

Obsah obrázku text, diagram, snímek obrazovky, Písmo

Popis byl vytvořen automaticky

#### Zobrazení historie plateb a transakcí

***Popis*:** Registrovaný uživatel si chce zobrazit historii svých plateb za členství a transakcí spojených s aplikací GymNest.

***Aktéři*:** Registrovaný uživatel

***Předpoklady*:** Uživatel je registrován do systému.

***Postup*:**

1. Uživatel se přihlásí do systému pomocí svého uživatelského jména a hesla.
2. Po úspěšném přihlášení uživatel se nachází na domovské stránce aplikace GymNest.
3. Uživatel vybere možnost "Historie plateb" nebo "Transakce" z hlavního menu.
4. Systém zobrazí seznam všech plateb za členství a transakcí, které uživatel provedl v minulosti.
5. Uživatel může procházet seznam a prohlížet detaily každé platby nebo transakce, jako je částka, datum a stav platby.
6. Po skončení prohlížení uživatel klikne na možnost "Zavřít" nebo se vrátí zpět na domovskou stránku.

***Alternativní průběh:***

* Pokud uživatel nemá žádné platby nebo transakce v historii, systém zobrazí odpovídající zprávu, že historie je prázdná.
* Pokud systém není schopen načíst historii platby nebo transakcí z databáze, uživatel obdrží chybové hlášení a může to zkusit znovu později.

***Znázornění***

Obsah obrázku text, diagram, řada/pruh, Písmo

Popis byl vytvořen automaticky

#### Správa Aktivit a Rozvrhů

***Popis*:** Vlastník posilovny může jednoduše spravovat aktivity a rozvrhy lekcí a tréninků v posilovně pomocí aplikace GymNest.

***Aktéři*:** Vlastník posilovny

***Předpoklady*:** Uživatel je registrován do systému s oprávněním admin nebo alespoň trenér.

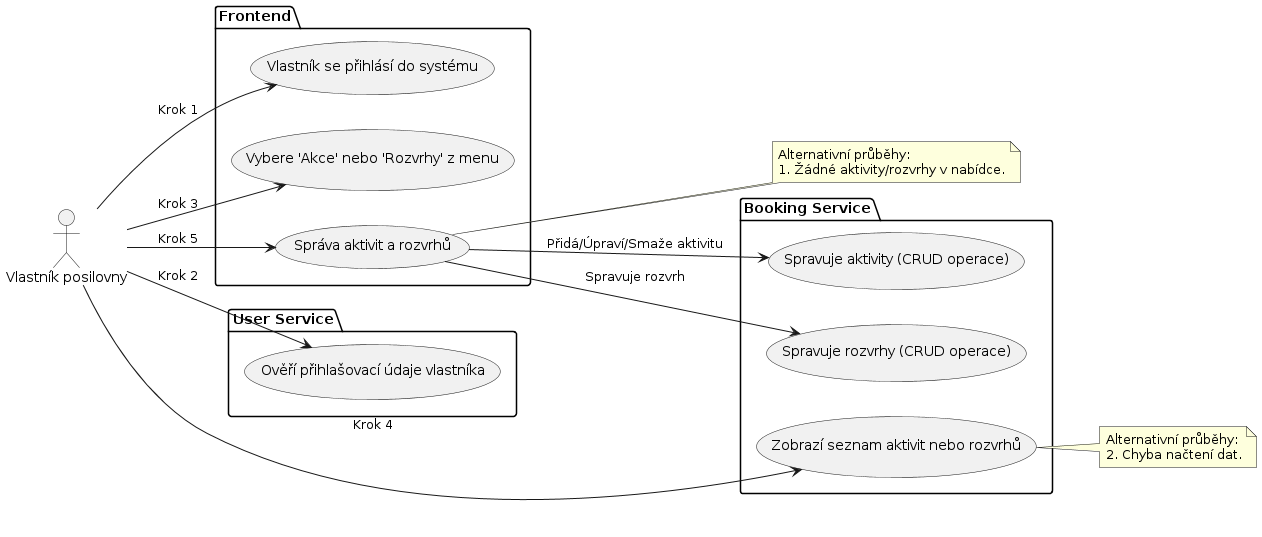
***Postup*:**

1. Vlastník posilovny se přihlásí do systému pomocí svých přihlašovacích údajů.
2. Po úspěšném přihlášení je vlastník posilovny přesměrován na domovskou stránku aplikace GymNest.
3. Vlastník posilovny klikne na možnost "Správa Aktivit" nebo "Rozvrhy" z hlavního menu.
4. Systém zobrazí seznam aktuálně nabízených aktivit nebo rozvrh lekcí a tréninků v posilovně.
5. Vlastník posilovny může provádět následující činnosti:
   * ***Přidání Nové Aktivity:***
     + Klikne na tlačítko "Přidat Aktivitu".
     + Vyplní formulář s informacemi o nové aktivitě.
     + Potvrdí přidání aktivity do systému.
   * ***Úprava Aktivity:***
     + Vybere existující aktivitu ze seznamu.
     + Klikne na tlačítko "Upravit".
     + Upraví požadované údaje aktivity.
     + Potvrdí provedené změny.
   * ***Mazání Aktivity:***
     + Vybere aktivitu, kterou chce odstranit.
     + Klikne na tlačítko "Smazat".
     + Potvrdí smazání aktivity ze systému.
   * ***Správa Rozvrhů:***
     + Prochází seznam lekcí a tréninků v rozvrhu.
     + Může provádět úpravy, jako jsou změny časů a lokalit nebo přiřazování trenérů k jednotlivým aktivitám.

***Alternativní průběh:***

* Pokud vlastník posilovny nemá žádné aktivity v nabídce, systém zobrazí odpovídající zprávu, že seznam je prázdný.
* Pokud systém není schopen načíst aktivity nebo rozvrh z databáze, vlastník posilovny obdrží chybové hlášení a může to zkusit znovu později.

***Znázornění***



## Technický design

Základním stavebním blokem aplikace GymNest je Node.js, který byl zvolen tak, aby zajišťovaly robustnost, bezpečnost a škálovatelnost aplikace.

Frontend projektu je založen na Reactu, moderním JavaScriptovém knihovním frameworku pro vytváření uživatelských rozhraní, který umožňuje efektivní a interaktivní web s jednostránkovou aplikací (SPA). Aplikace využívá širokou paletu nástrojů a knihoven, které podporují rychlý vývoj a poskytují robustní funkčnost.

### Popis architektury

Popis používaných nástrojů, knihoven a služeb používané v aplikaci GymNest.

#### Backend

Architektura backendu je navržena pro podporu modulárnosti a snadné škálování. Aplikace je rozdělena do více vrstev, které zahrnují: prezenční, logickou a datovou vrstvu.

**Express.js:** Jako základní framework pro backendovou aplikaci je používán Express.js. Tento minimalistický a flexibilní framework poskytuje soubor bohatých funkcí pro webové a mobilní aplikace.

**MySQL2:** MySQL2 je rychlý MySQL driver pro Node.js, který podporuje Promise API a umožňuje efektivně komunikovat s databází MySQL.

**Sequelize:** Sequelize je ORM (Object-Relational Mapping) pro Node.js, který podporuje databáze MySQL. Umožňuje nám snadno manipulovat s daty pomocí JavaScriptu.

**Passport.js:** Pro autentizační strategie je používán Passport. Je to middleware pro Node.js. Podporuje autentizační strategie jako jsou OAuth, JWT atd.

**JWT:** Pro implementaci tokenů je používána knihovna jsonwebtoken. JWT umožňuje bezpečné předávání informací mezi klientem a serverem jako JSON objekty.

**Dotenv:** Knihovna dotenv se používá pro správu environmentálních proměnných v Node.js, což umožňuje snadné konfigurování různých aspektů aplikace bez potřeby změn kódu.

**Pug:** Pro generování šablonového HTML na serveru je púoužíván Pug, což je vysoce výkonný šablonovací engine.

**Swagger:** Pro dokumentaci API je tu nástroj Swagger (swagger-jsdoc a swagger-ui-express), který umožňuje snadno generovat a zobrazovat interaktivní dokumentaci API, která je kompatibilní se standardem OpenAPI.

**Morgan, Debug, HTTP-Errors:** Morgan je middleware pro logování HTTP požadavků, který pomáhá s debugováním a sledováním požadavků přijímaných serverem. Debug je knihovna umožňující jednoduché debugování aplikace, a http-errors slouží k vytváření HTTP chyb v Express aplikacích, což usnadňuje ošetření chybových stavů.

#### Frontend

Architektura je navržena s důrazem na modularitu a znovupoužitelnost komponent. Využíváme kontejnerový vzor, kde logické kontejnery spravují stavy a data, zatímco prezentující komponenty jsou čistě pro zobrazení UI.

**MUI (Material-UI):** Knihovna pro React, která poskytuje sady přednastavených komponent ve stylu Material Design. Použití této knihovny zaručuje konzistentní a atraktivní uživatelské rozhraní, snižuje potřebu vlastního CSS a zrychluje vývoj.

**Emotion:** Emotion je knihovna pro styled-components, která umožňuje styling komponent přímo s využitím JavaScriptu. Nabízí efektivní způsoby, jak manipulovat styly na základě props a stavů aplikace.

**Axios:** Axios je klient pro HTTP požadavky založený na Promises, který umožňuje snadné provádění požadavků na API.

**React Router Dom:** React Router je standardní knihovna pro routování v React aplikacích, která umožňuje navigaci mezi různými komponentami bez nutnosti opětovného načítání stránky.

**Date-fns:** Knihovna pro práci s daty v JavaScriptu, která poskytuje množství užitečných funkcí pro manipulaci a formátování datumů.

**PayPal (React-PayPal-js)**: Integrace PayPal pro jednoduché zpracování plateb v aplikaci, což poskytuje uživatelům bezpečné a spolehlivé rozhraní pro transakce.

**Google OAuth (React-OAuth-Google):** Komponenta pro zjednodušení autentizace uživatelů pomocí Google OAuth, což zvyšuje uživatelský komfort při přihlašování.

Dále pro vývojové nástroje jsou používané:

**React Scripts:** Zahrnuje soubor nástrojů a konfigurací (včetně Babel a Webpack), které usnadňují vývoj aplikací vytvořených pomocí Create React App.

**ESLint:** Linter, který pomáhá udržovat kód čistý a konzistentní dle doporučených praktik.

### Doménový / datový model

V projektu jsou definovány struktury dat, která aplikace používá, a to jak na úrovni databáze, tak na úrovni aplikace. Tento model popisuje entitní vztahy, atributy a omezení, která se aplikují na data uložená v systému. Doménový model pomáhá pochopit, jak se jednotlivé části systému navzájem souvisejí a jak jsou data organizována a spravována. Je klíčový pro návrh efektivního, logického a udržitelného softwarového systému.

**Hlavní aspekty doménového/datového modelu**

* Entity (Entitní typy)
* Atributy
* Vztahy
* Omezení integrity

Doménový/datový model v projektu je navržen s ohledem na všechny klíčové funkce a požadavky systému. Je navržen tak, aby na každé vrstvě docházelo k validaci vstupů (např.: je uživatelské jméno unikátní? Tento prvek se řeší i v logice backendu, ale důslednost na integritu je v celém projektu).

### Návrh infrastruktury

+++

### UML diagram

Obsah obrázku text, diagram, Písmo, dokument

Popis byl vytvořen automaticky