



SPŠ ÚSTÍ NAD LABEM

# Patient care

Dokumentace k ročníkové práci

**Autor:** Tomáš Toman

**Třída:** 3ITB

**Vedoucí práce:** Bc. Jiří Brož

2023/2024



## Prohlášení

---

Prohlašuji, že jsem ročníkovou práci na téma „Patient care“ vypracoval samostatně a s použitím uvedené literatury a pramenů.

V Litoměřicích dne 7.6.2024

Toman Tomáš

## Poděkování

---

Chtěl bych poděkovat Bc. Jiřímu Brožovi za vedení mé ročníkové práce, cenné rady a odborný dohled. Děkuji také ChatGPT za pomoc při gramatické kontrole práce a Miroslavu Doudovi za pomoc při návrhu řešení algoritmu.

# Anotace

---

Tato dokumentace je průzkum implementace softwarového projektu zaměřeného na technologie jako je JavaScript, PHP, Visual Studio Code, Coohom, Adobe Photoshop, Figma, Font Awesome, GitHub, HTML, CSS a další. Struktura dokumentu zahrnuje úvod, ve kterém je představena myšlenka projektu, řešerši, ve které jsou konkurenční produkty a následně přechází do praktické části obsahující návrhy, produktizaci a popis pro uživatele. Dokument je zakončen závěrem, ve kterém se nacházejí informace ohledně budoucnosti projektu a jeho splnění cíle, použitou literaturou a seznamem obrázků ve které najdete všechny citované texty a obrázky v tomto dokumentu.

## Klíčová slova

---

Patient care, Aplikace, Pacient, PHP, Navigace, Oddělení, Mapy, Dokumentace, Log, Nástroj

# Obsah

---

Úvod .....	8
1 Rešerše .....	9
1.1 Nedostatek podobných projektů .....	9
1.2 Epic Systems.....	9
1.3 Meditech .....	10
2 Technologie .....	11
2.1 Visual Studio Code .....	11
2.2 Coohom.....	11
2.3 Adobe Photoshop .....	11
2.4 Figma.....	11
2.5 Font Awesome .....	11
2.6 GitHub .....	12
2.7 HTML.....	12
2.8 CSS.....	12
2.9 JavaScript .....	12
2.10 AJAX.....	13
2.11 jQuery.....	13
3 Praktická část.....	14
3.1 Návrhy.....	14
3.1.1 Myšlenková mapa.....	14
3.1.2 Návrh databáze.....	14
3.1.3 První návrh loga .....	15
3.1.4 Počáteční návrhy v nástroji Figma .....	15
3.2 Produktizace.....	18
3.2.1 Přihlašovací algoritmus.....	18
3.2.2 Funkce pro přidání zaměstnance do databáze .....	19
3.2.3 Funkce pro přidání pacienta do databáze .....	20
3.2.4 jQuery funkce pro checkboxu a přiřazování jejich hodnot.....	22
3.2.5 Jednoduchý algoritmus pro vyhledávání dat v tabulkách .....	24
3.3 Popis pro uživatele .....	25
3.3.1 Přihlašovací stránka .....	25
3.3.2 Navigační menu .....	25
3.3.3 Jednotlivá oddělení.....	26
3.3.4 Jednotlivé pokoje.....	26

3.3.5	Zobrazovací tabulka zaměstnanců .....	27
3.3.6	Stránka pro přidání zaměstnance do databáze .....	27
3.3.7	Zobrazovací tabulka pacientů .....	28
3.3.8	Přidání pacienta do databáze .....	28
3.3.9	Dokumentace pacientů .....	29
3.3.10	Úprava dokumentace .....	30
3.3.11	Log akcí .....	30
Závěr .....		31
Použitá literatura .....		32
Seznam obrázků .....		33
Obsah média .....		34

# Úvod

---

Plánuji vytvořit komplexní webovou aplikaci, která bude sloužit k efektivní správě pacientů v rámci nemocničního prostředí. Tato aplikace vzniká z mé osobní motivace a zkušeností, které jsem získal díky své přítelkyni, která pracuje v oblasti zdravotnictví. Pozoroval jsem nedostatky současných systémů, které jsou často zastaralé, pomalé a málo estetické. Můj cíl je tedy vytvořit moderní a uživatelsky přívětivou platformu, která nejenže usnadní práci zaměstnancům nemocnic, ale také přispěje k celkovému zlepšení péče o pacienty.

Aplikace bude postavena na technologiích HTML, CSS, PHP a JS, což zajišťuje její univerzální přístupnost a kompatibilitu s různými zařízeními. Základem bude zabezpečené přihlašování, které bude dostupné pouze pro autorizované uživatele, přičemž registrace bude omezena na správce aplikace.

Jedním z hlavních prvků aplikace bude navigační panel, který umožní uživatelům snadný přístup k různým částem systému, jako jsou oddělení, mapy budov a další užitečné funkce. Uživatelé budou mít možnost přidávat a odebírat pacienty z lůžek, zobrazovat jejich informace a kompletní dokumentaci. Pro správce aplikace bude k dispozici také možnost správy zaměstnanců.

Důležitou součástí aplikace bude rozsáhlý log akcí, který umožní sledovat veškeré události a změny v systému, což přispěje k transparentnosti a správnému řízení celého procesu.

Celkový výpis pacientů a zaměstnanců bude přístupný na samostatné stránce s možností vyhledávání, což usnadní orientaci a vyhledávání konkrétních informací. Tímto způsobem se zajistí efektivní a plynulý chod nemocnice, zvýší se kvalita poskytované péče a optimalizuje se proces správy pacientů.



# 1 Rešerše

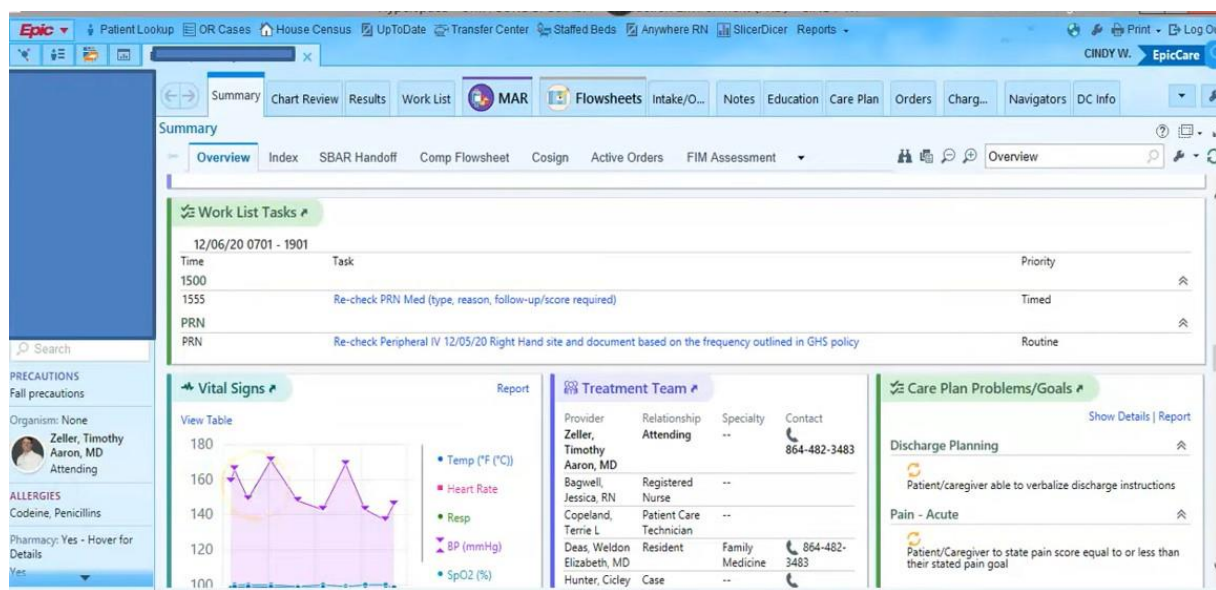
## 1.1 Nedostatek podobných projektů

Při hledání podobných projektů na téma mé práce jsem nenarazil na žádný na tolik podobný abych ho uvedl do tohoto dokumentu. Při hledání jsem použil různé vyhledávače, avšak každý projekt, který jsem našel byl naprosto o něčem jiném. I přesto bych chtěl uvést nějaké příklady tematicky podobných projektů.

## 1.2 Epic Systems

Epic Systems je americká soukromá společnost specializující se na software pro zdravotnický průmysl. Její software je používán v mnoha nemocnicích a podle společnosti měly tyto nemocnice v roce 2022 uchovávat zdravotní záznamy 78% pacientů ve Spojených státech. (1)

Tuto aplikaci jsem našel při prohledávání podobných témat a přišla mi zajímavá svým obrovským rozsahem ve Spojených státech. U detailnějšího průzkumu jsem se ovšem zarazil nad jejím poněkud zastaralým zpracováním.



Obrázek 1 - Epic EMR Software (2)

### 1.3 Meditech

MEDITECH se snaží pomoci vybudovat svět, kde každý pacient má přístup ke svým zdravotním informacím a může se plně angažovat ve své péči. Pro dosažení tohoto cíle vytvořili cloudovou platformu MEDITECH as a Service (MaaS), která umožňuje vytvářet špičkové technologie pro bezpečnou a nákladově efektivní zdravotní péči. (3)

I když je tato aplikace svojí funkcí a konceptem naprosto odlišná, stále si myslím že by svým způsobem mohla konkurovat jejím rozsahem a zpracování aplikaci Patient care.

The screenshot displays the Meditech patient care interface for a patient named Anderson, Charlotte Inn. The interface is divided into several sections:

- Top Navigation Bar:** Includes icons for Return To, Home, Chart, Document, Orders, Oncology, Discharge, Sign, Workload, Menu, and Close.
- Diagnostic Categories:** A horizontal bar with tabs for Diagnostics (History & Problems, Summary), Provider Notes (Administrative, Activity), Nurse/Allied Health (Other Clinical, Flowsheets), and Medications (Workload Items, Health Mgmt).
- Left Sidebar:** A list of diagnostic categories including All, Hematology, Chemistry, Urines, and Genetics (which is currently selected).
- Main Content Area:**
  - Genetic Orders:** A table with columns for ORDER, RESULT, REPORT, DATE, and LINKED SPECIMEN. It lists three orders: Ambry Clinical Summary (Nccn guidelines met), Solid Tumor Gene Panel (Positive), and Pharmacogenomics Panel (See gene panel).
  - Patient's Genetic History:** A section with two sub-sections: Germline and Somatic. The Germline section lists four entries for CYP2C19, CYP2C9, CYP2D6, and CYP3A4, each with a genetic finding, phenotype, report, and date. The Somatic section lists one entry for PIK3CA with a genetic finding and report date.
- Right Sidebar:** Contains patient information (Anderson, Charlotte Inn, Acc# MGO000064950), a search bar, and several expandable sections: Allergies (Milk Containing Products, Diarrhea), Bulletin Board (No Data to Display), and Oncology Treatment Summary (ONC DIAGNOSIS: Anemia of chronic illness, STAGING; ONC DIAGNOSIS: HER2-negative carcinoma of breast, STAGING).

Obrázek 2 – Meditech (4)

## 2 Technologie

---

### 2.1 Visual Studio Code

---

Visual Studio Code je lehký, avšak výkonný editor zdrojového kódu, který je dostupný pro systémy Windows, MacOS a Linux. Obsahuje vestavěnou podporu pro JavaScript, TypeScript a Node.js a nabízí bohatý ekosystém rozšíření pro další jazyky a runtimes, jako jsou C++, C#, Java, Python, PHP, Go a .NET. (5)

Textový editor, který používám na veškerý kód.

### 2.2 Coohom

---

Coohom je aplikace pro interiérový design, která je snadno použitelná a uživatelsky přívětivá. Nabízí 3D fotorealistickou vizualizaci a technologie virtuální a rozšířené reality. Aplikace je aktuální a vhodná pro různé potřeby v oblasti interiérového designu. (6)

Coohom jsem použil na zpracování map nemocnice, protože je open-source a velice spolehlivá.

### 2.3 Adobe Photoshop

---

Adobe Photoshop je bitmapový grafický editor určený pro tvorbu a úpravy bitmapové grafiky, jako jsou například fotografie, a byl vytvořen firmou Adobe Systems. Je to jeden z nejznámějších a nejpoužívanějších nástrojů v oblasti digitálního umění a grafického designu. (7)

Veškerá grafika jako jsou loga a další maličkosti v aplikaci je tvořena v Adobe Photoshop.

### 2.4 Figma

---

Figma je grafický editor určený pro společný návrh webových stránek, aplikací a dalších designových produktů. Vznikla v roce 2016 jako alternativa ke Sketchi a Adobe XD, a během několika let se stala jedním z nejoblíbenějších nástrojů mezi designéry. (8)

Prvotní návrh aplikace jsem tvořil v nástroji Figma.

### 2.5 Font Awesome

---

Font Awesome je sada fontů a ikon založená na CSS a LESS, kterou vytvořil Dave Gandy. Původně byla navržena pro použití s Twitter Bootstrap, ale později byla začleněna do veřejné sítě pro doručování obsahu BootstrapCDN. (9)

V projektu jsem využil jejich knihovnu ikon.

## 2.6 GitHub

---

GitHub je největší a nepokročilejší vývojová platforma na světě, kterou využívá více než 83 milionů vývojářů a 4 miliony organizací. Od svého vzniku v roce 2008 slouží k vytváření, dodávání a udržování softwaru. GitHub je úložiště kódů založené na verzovacím systému Git a je převážně zaměřené na open-source projekty. Zatímco Git je nástroj příkazového řádku, GitHub poskytuje webové grafické rozhraní pro snadnější práci s kódem. (10)

GitHub jsem v projektu použil na správu a zálohování dat.

## 2.7 HTML

---

Hypertext Markup Language, známý jako HTML, je značkovací jazyk používaný pro tvorbu webových stránek, které jsou propojeny hypertextovými odkazy. HTML je klíčovým jazykem pro vytváření stránek na World Wide Webu, což umožňuje publikaci dokumentů na internetu. (11)

Základní stavební kámen celé aplikace je psán v HTML.

## 2.8 CSS

---

Cascading Style Sheets (CSS) jsou jazykem stylů používaným k určení vzhledu a stylování dokumentu napsaného ve značkovacím jazyce, jako je HTML nebo XML (včetně dialektů XML, jako jsou SVG, MathML nebo XHTML). CSS, spolu s HTML a JavaScriptem, tvoří základní technologie World Wide Webu.

CSS je navrženo tak, aby umožňovalo oddělení obsahu a prezentace, což zahrnuje rozvržení, barvy a písma. Toto oddělení může zlepšit přístupnost obsahu a usnadnit údržbu a správu webových stránek. (12)

Stylizování HTML elementů je pouze pomocí CSS.

## 2.9 JavaScript

---

JavaScript (JS) je skriptovací jazyk určený pro tvorbu moderních dynamických webů. Byl představen v devadesátých letech 20. století jako reakce na klasické statické internetové stránky, což umožnilo vznik interaktivních webových aplikací s příjemnějším uživatelským rozhraním, animacemi a 2D i 3D grafikou. JavaScript je primárně určen pro tvorbu klientských částí aplikací, ale s příchodem Node.js a podobných technologií se stal plnohodnotným jazykem i pro serverovou stranu. (13)

Velká část funkcionality a responzivity webové aplikace je pomocí JavaScriptu.

## 2.10 AJAX

---

AJAX (Asynchronous JavaScript And XML) je technologie, která umožňuje provádět HTTP požadavky přímo z JavaScriptu. Funguje podobně, jako když do adresního řádku prohlížeče zadáte webovou adresu a prohlížeč odešle HTTP požadavek a stáhne data z dané adresy (například HTML kód webové stránky). AJAX umožňuje provádět tyto požadavky přímo v JavaScriptovém kódu a hlavně na pozadí, aniž by se stránka, na které se nacházíme, musela znovu načítat. (14)

V aplikaci jsem ho použil, abych zamezil zbytečnému přesměrovávání při volání PHP funkcí.

## 2.11 jQuery

---

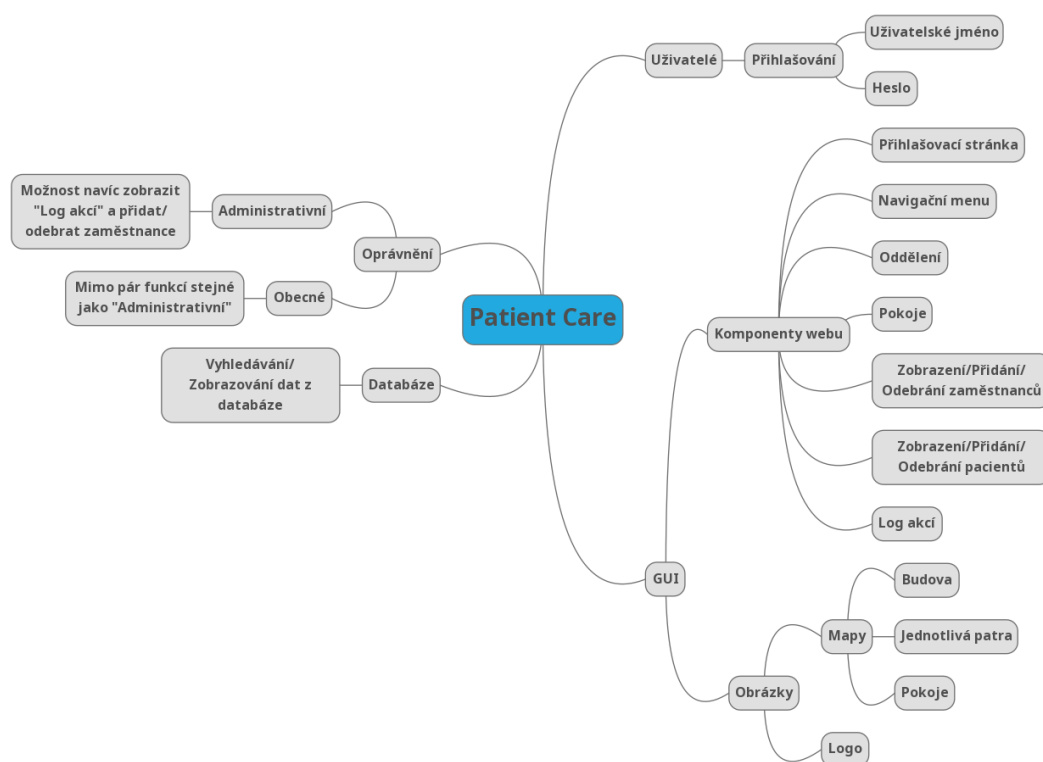
jQuery je knihovna napsaná v jazyce JavaScript, která zjednodušuje manipulaci se strukturou HTML dokumentů, ovládání událostí v JavaScriptu, práci s technologií AJAX, různé efekty a animace a nabízí mnoho dalších funkcí. Byla vytvořena v roce 2006 a je dostupná zdarma jako open source. (15)

V aplikaci jsem jí použil na mnoho funkcí jako jsou např. checkboxy a různé algoritmy.

## 3 Praktická část

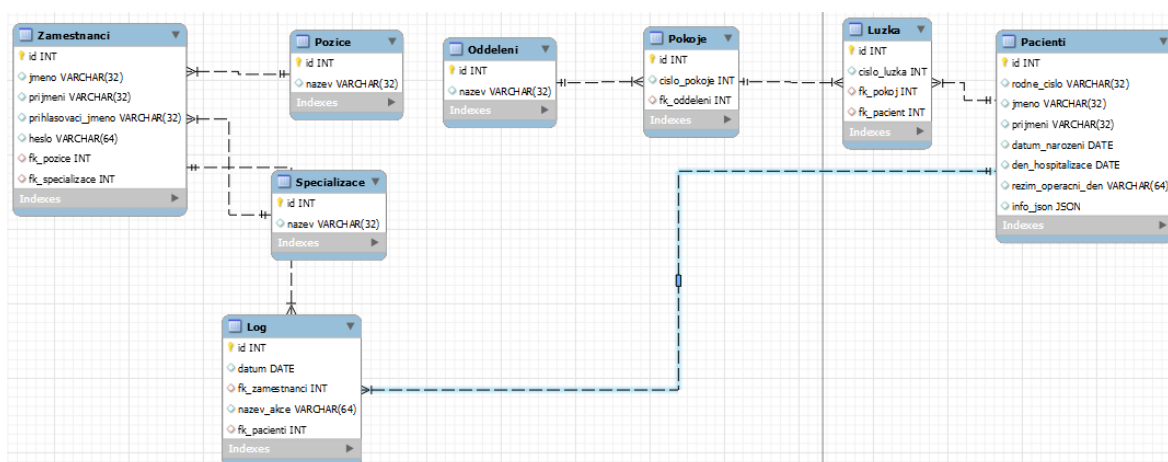
### 3.1 Návrhy

#### 3.1.1 Myšlenková mapa



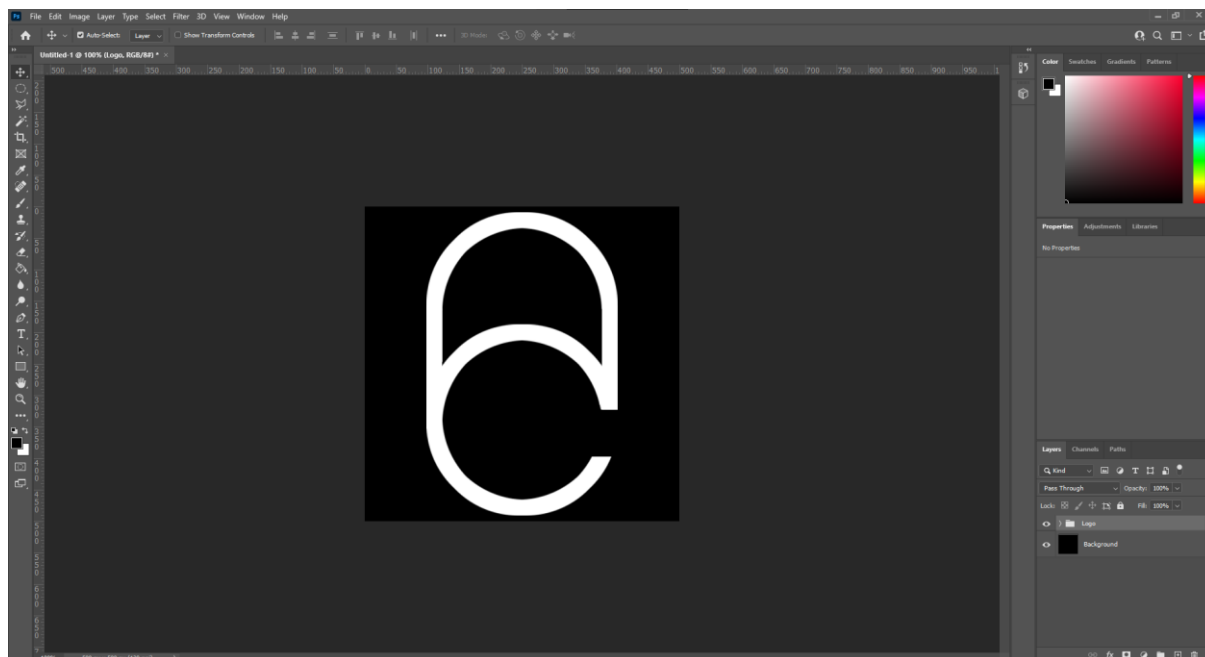
Obrázek 3 - Myšlenková mapa

#### 3.1.2 Návrh databáze



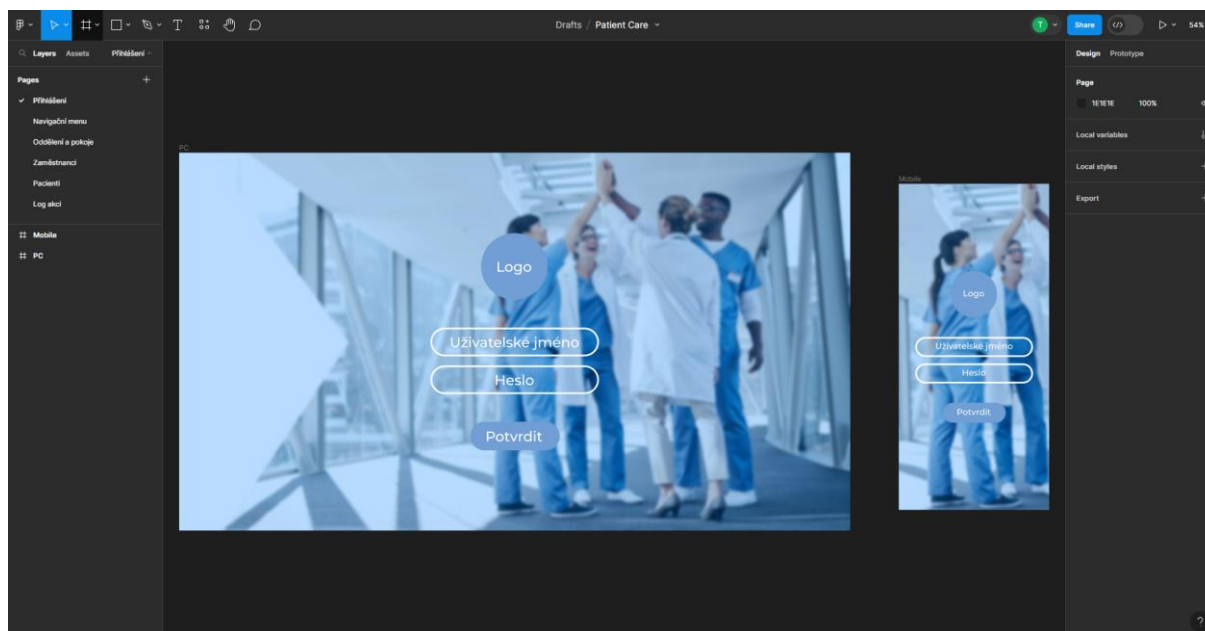
Obrázek 4 - Návrh databáze

### 3.1.3 První návrh loga

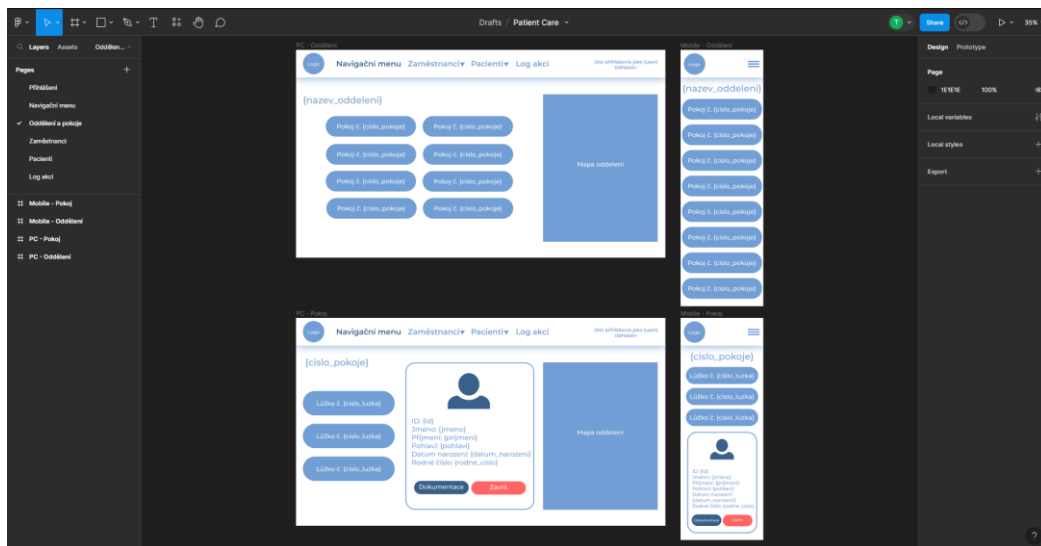


Obrázek 5 - Návrh loga

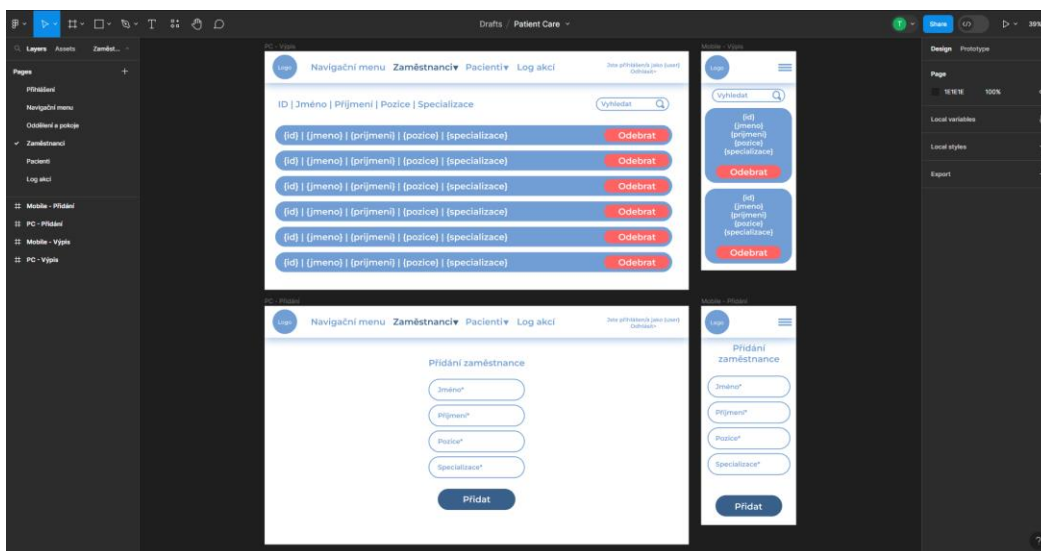
### 3.1.4 Počáteční návrhy v nástroji Figma



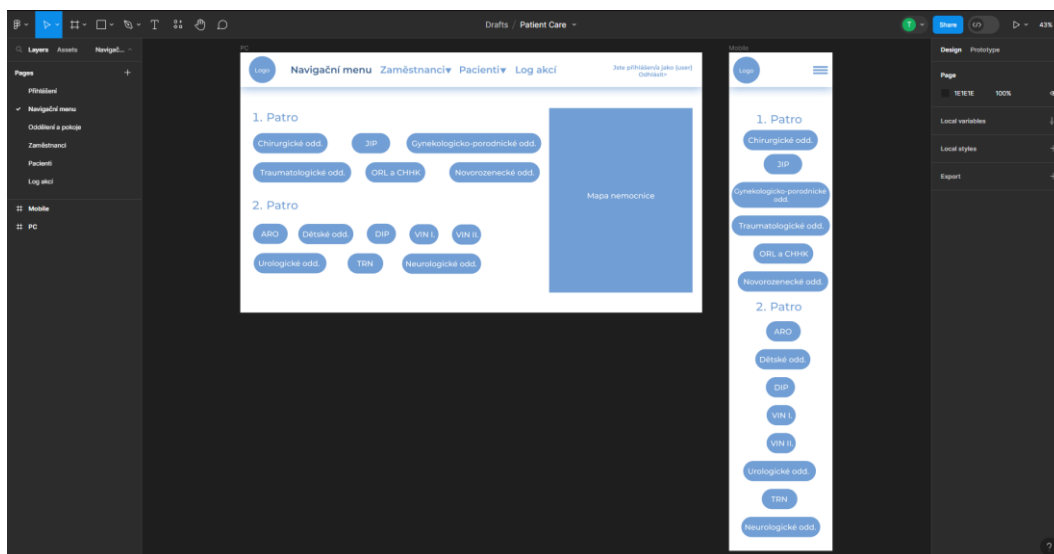
Obrázek 6 - Návrh Figma 1



Obrázek 7 - Návrh Figma 2

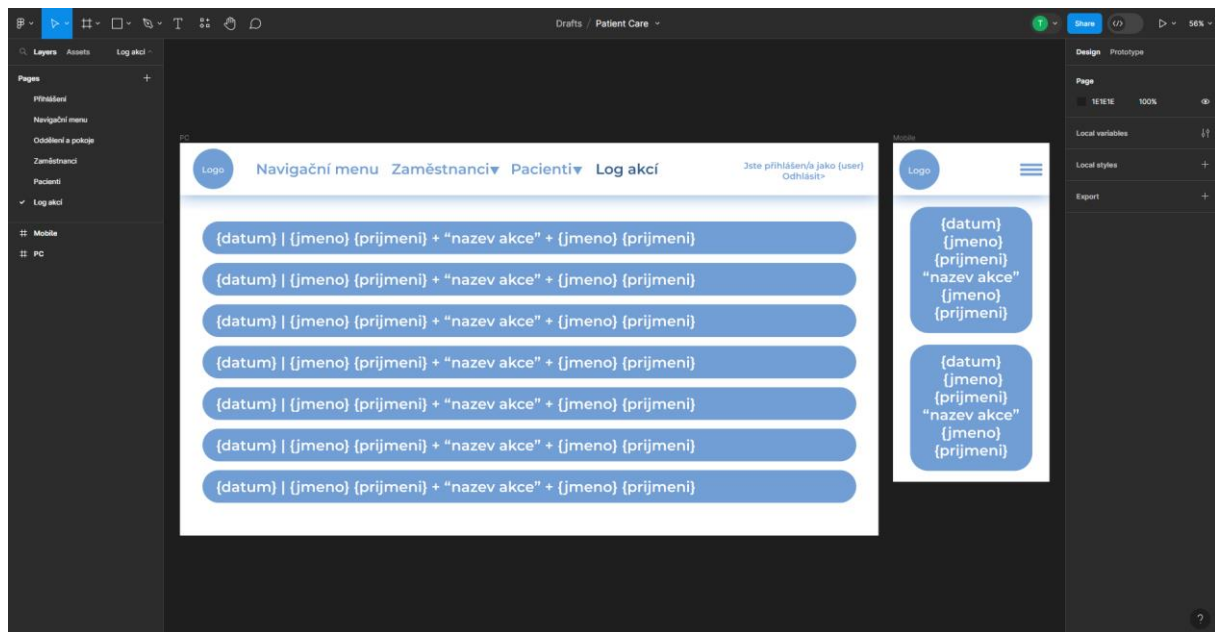


Obrázek 8 - Návrh Figma 3

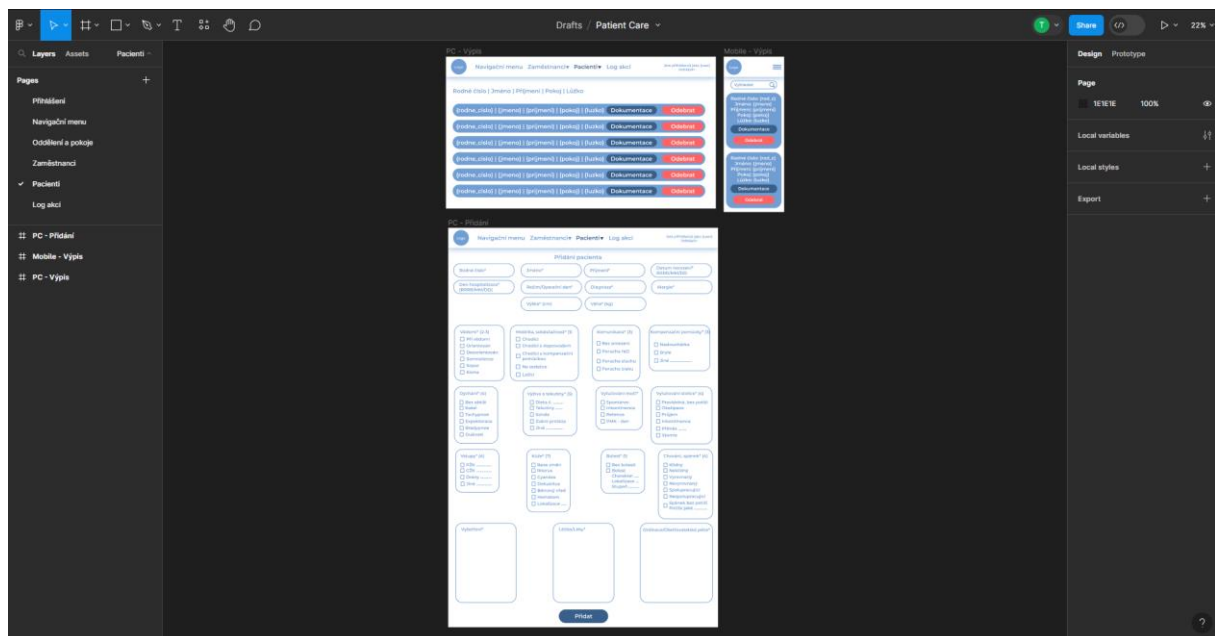


Obrázek 9 - Návrh Figma 4





Obrázek 10 - Návrh Figma 5



Obrázek 11 - Návrh Figma 6

## 3.2 Produktizace

---

### 3.2.1 Přihlašovací algoritmus

---

```
1. <?php
2. // Start nové session nebo pokračování stávající
3. session_start();
4.
5. // Zahrnutí souboru s připojením k databázi
6. include "../conn.php";
7.
8. // Ověření, zda jsou vyplněny všechny údaje
9. if (isset($_POST['uzivatelske_jmeno'], $_POST['heslo'])) {
10. // Přiřazení proměnných s ošetřením proti SQL injection
11. $uzivatelske_jmeno = pg_escape_string($conn, $_POST['uzivatelske_jmeno']);
12. $heslo = pg_escape_string($conn, $_POST['heslo']);
13.
14. // Pokud jsou nějaké údaje prázdné, přesměrování zpět na přihlášení s chybovou zprávou
15. if (empty($uzivatelske_jmeno) || empty($heslo)) {
16. header("Location: ../html/prihlaseni.php?error=Vypřte všechny údaje!");
17. exit(); // Zastavení provádění kódu
18. }
19.
20. // Dotaz na databázi pro ověření uživatele
21. $query = "SELECT * FROM zamestnanci WHERE prihlasovací_jmeno = $1";
22. $result = pg_query_params($conn, $query, array($uzivatelske_jmeno));
23.
24. // Kontrola, zda byl vrácen právě jeden záznam
25. if (pg_num_rows($result) === 1) {
26. $user = pg_fetch_assoc($result);
27.
28. // Ověření hesla
29. if (password_verify($heslo, $user['heslo'])) {
30. // Nastavení uživatelských dat do session
31. $_SESSION['user_id'] = $user['id'];
32. $_SESSION['user_jmeno'] = $user['jmeno'];
33. $_SESSION['user_prijmeni'] = $user['prijmeni'];
34. $_SESSION['user_opravneni'] = $user['opravneni'];
35.
36. // Záznam do logu
37. $logData = $_SESSION['user_jmeno'] . " " . $_SESSION['user_prijmeni'] . " se přihlásil/a.";
38. $queryLog = "INSERT INTO log (data, datum) VALUES ($1, DATE_TRUNC('second', NOW()))";
39. $resultLog = pg_query_params($conn, $queryLog, array($logData));
40.
41. // Přesměrování na menu
42. header("Location: ../html/menu.php");
43. exit(); // Zastavení provádění kódu
44. }
45. }
46.
47. // Pokud nebyl nalezen shodující se uživatel nebo špatné heslo, přesměrování s chybovou zprávou
48. header("Location: ../html/prihlaseni.php?error=Špatně zadané údaje!");
49. exit(); // Zastavení provádění kódu
50. } else {
51. // Pokud nebyla vyplněna všechna pole, přesměrování zpět na přihlášení
52. header("Location: ../html/prihlaseni.php");
53. exit(); // Zastavení provádění kódu
54. }
```

### 3.2.2 Funkce pro přidání zaměstnance do databáze

---

```
1. <?php
2. // Zahájení PHP session pro uchování dat mezi různými stránkami
3. session_start();
4.
5. // Připojení k databázi
6. include './conn.php';
7.
8. // Přiřazení hodnot z formuláře do proměnných
9. $jmeno = $_POST['jmeno'];
10. $prijmeni = $_POST['prijmeni'];
11. $prih_jmeno = $_POST['prih_jmeno'];
12. $heslo = password_hash($_POST['heslo'], PASSWORD_DEFAULT); // heslo je hashováno pro bezpečnost
13. $pozice = $_POST['pozice'];
14. $specializace = $_POST['specializace'];
15. $opraveni = $_POST['opraveni'];
16.
17. // Pokud pozice_id není "-", vyhledá se v databázi a přiřadí se do proměnné pozice_id
18. if($pozice_id != "-"){
19.     $result = pg_prepare($conn, "pozice_query", 'SELECT fk_pozice FROM pozice WHERE nazev_pozice =
20.     $1');
21.     $result = pg_execute($conn, "pozice_query", array($pozice));
22.     $pozice_id = pg_fetch_result($result, 0, 0);
23. } else {
24.     $pozice_id = NULL; // Pokud je pozice_id "-", nastaví se na NULL
25. }
26.
27. // Stejný proces se opakuje pro specializace
28. if($specializace != "-") {
29.     $result = pg_prepare($conn, "specializace_query", 'SELECT fk_specializace FROM specializace WHERE
30.     nazev_specializace = $1');
31.     $result = pg_execute($conn, "specializace_query", array($specializace));
32.     $specializace_id = pg_fetch_result($result, 0, 0);
33. } else {
34.     $specializace_id = NULL;
35. }
36.
37. // Vložení nového zaměstnance do databáze
38. $result = pg_prepare($conn, "my_query", 'INSERT INTO zamestnanci (jmeno, prijmeni, prihlasovací_jmeno,
39.     heslo, fk_pozice, fk_specializace, opraveni) VALUES ($1, $2, $3, $4, $5, $6, $7)');
40. $result = pg_execute($conn, "my_query", array($jmeno, $prijmeni, $prih_jmeno, $heslo, $pozice_id,
41.     $specializace_id, $opraveni));
42.
43. // Vytvoření záznamu o přidání zaměstnance do logu
44. $logData = $_SESSION['user_jmeno'] . " " . $_SESSION['user_prijmeni'] . " přidal/a zaměstnance " . $jmeno .
45.     " " . $prijmeni . " do databáze.";
46. $result = pg_prepare($conn, "log", "INSERT INTO log (data, datum) VALUES ($1, DATE_TRUNC('second',
47.     NOW()))");
48. $result = pg_execute($conn, "log", array($logData));
49.
50. // Přesměrování uživatele na stránku s výpisem zaměstnanců
51. header('Location: ./vypis_zamestnancu.php');
```

### 3.2.3 Funkce pro přidání pacienta do databáze

---

```
1. <?php
2.     // Zahájení nebo pokračování PHP session pro uchování dat mezi různými stránkami
3.     session_start();
4.
5.     // Připojení k databázi
6.     include './conn.php';
7.
8.     // Přiřazení hodnot z formuláře do proměnných
9.     $rodne_cislo = $_POST['rodne-cislo'];
10.    $jmeno = $_POST['jmeno'];
11.    $prijmeni = $_POST['prijmeni'];
12.    $datum_narozeni = $_POST['datum-narozeni'];
13.    $den_hospitalizace = $_POST['den-hospitalizace'];
14.    $rezim_operacni_den = $_POST['rezim-operacni-den'];
15.
16.    // Přiřazení dalších hodnot z formuláře do proměnných
17.    // Pokud hodnota neexistuje, je nastavena na prázdné pole
18.    $diagnoza = $_POST['diagnoza'];
19.    $alergie = $_POST['alergie'];
20.    $vyska = $_POST['vyska'];
21.    $vaha = $_POST['vaha'];
22.    $vedomi = isset($_POST['vedomi']) ? $_POST['vedomi'] : array();
23.    $mobilita = isset($_POST['mobilita']) ? $_POST['mobilita'] : array();
24.    $komunikace = isset($_POST['komunikace']) ? $_POST['komunikace'] : array();
25.    $pomucky = isset($_POST['pomucky']) ? $_POST['pomucky'] : array();
26.    $dychani = isset($_POST['dychani']) ? $_POST['dychani'] : array();
27.    $vyziva = isset($_POST['vyziva']) ? $_POST['vyziva'] : array();
28.    $moc = isset($_POST['moc']) ? $_POST['moc'] : array();
29.    $stolice = isset($_POST['stolice']) ? $_POST['stolice'] : array();
30.    $vstupy = isset($_POST['vstupy']) ? $_POST['vstupy'] : array();
31.    $kuze = isset($_POST['kuze']) ? $_POST['kuze'] : array();
32.    $bolest = isset($_POST['bolest']) ? $_POST['bolest'] : array();
33.    $chovani = isset($_POST['chovani']) ? $_POST['chovani'] : array();
34.    $vysetreni = $_POST['vysetreni'];
35.    $lecba = $_POST['lecba'];
36.    $ordinace = $_POST['ordinace'];
37.
38.    // Vytvoření asociativního pole s informacemi o pacientovi
39.    // Hodnoty jsou spojeny do řetězců pomocí funkce implode
40.    $info_json = json_encode(array(
41.        "diagnoza" => $diagnoza,
42.        "alergie" => $alergie,
43.        "vyska" => $vyska,
44.        "vaha" => $vaha,
45.        "vedomi" => implode(';', $vedomi),
46.        "mobilita,sobestacnost" => implode(';', $mobilita),
47.        "komunikace" => implode(';', $komunikace),
48.        "kompenzacni_pomucky" => implode(';', $pomucky),
49.        "dychani" => implode(';', $dychani),
50.        "vyziva,tekutiny" => implode(';', $vyziva),
```

```

51.     "vylucovani_moci" => implode('; ', $moc),
52.     "vylucovani_stolice" => implode('; ', $stolice),
53.     "vstupy" => implode('; ', $vstupy),
54.     "kuze" => implode('; ', $kuze),
55.     "bolest" => implode('; ', $bolest),
56.     "chovani,spanek" => implode('; ', $chovani),
57.     "vysetreni" => $vysetreni,
58.     "lecba,leky" => $lecba,
59.     "ordinace,osetrovatelska_pece" => $ordinace
60. );
61.
62. // Příprava SQL dotazu pro vložení nového pacienta do databáze
63. $result = pg_prepare($conn, "patient_query", 'INSERT INTO pacienti (rodne_cislo, jmeno,
    prijmeni, datum_narozeni, den_hospitalizace, rezim_operacni_den, info_json) VALUES ($1, $2, $3,
    $4, $5, $6, $7)');
64. // Spuštění dotazu s hodnotami z formuláře
65. $result = pg_execute($conn, "patient_query", array($rodne_cislo, $jmeno, $prijmeni,
    $datum_narozeni, $den_hospitalizace, $rezim_operacni_den, $info_json));
66.
67. // Vytvoření záznamu o akci pro log
68. $logData = $_SESSION['user_jmeno'] . " " . $_SESSION['user_prijmeni'] . " přidal/a pacienta " .
    $jmeno . " " . $prijmeni . " do databáze.";
69.
70. // Příprava SQL dotazu pro vložení záznamu do logu
71. $result = pg_prepare($conn, "log", "INSERT INTO log (data, datum) VALUES ($1,
    DATE_TRUNC('second', NOW()))");
72. // Spuštění dotazu s hodnotami pro log
73. $result = pg_execute($conn, "log", array($logData));
74. ?>

```

### 3.2.4 jQuery funkce pro checkboxu a přiřazování jejich hodnot

---

```
1. <script>
2.   jQuery(function(){
3.     // Vytvoření prázdného objektu pro skupiny checkboxů
4.     var checkboxGroups = {};
5.
6.     // Pro každý checkbox na stránce
7.     jQuery('input[type="checkbox"]').each(function(){
8.       var $this = $(this);
9.       var name = this.name;
10.
11.      // Pokud skupina pro daný checkbox ještě neexistuje, vytvoříme ji
12.      if (!checkboxGroups[name]) {
13.        checkboxGroups[name] = [];
14.      }
15.
16.      // Pokud je checkbox zaškrtnutý, přidáme ho do příslušné skupiny
17.      if (this.checked) {
18.        checkboxGroups[name].push($this);
19.      }
20.    });
21.
22.    // Když uživatel klikne na checkbox
23.    jQuery('input[type="checkbox"]').click(function(){
24.      var $this = $(this);
25.      var max = $this.data('max');
26.      var name = this.name;
27.
28.      // Pokud je checkbox zaškrtnutý
29.      if (this.checked) {
30.        // Přidáme ho do příslušné skupiny
31.        checkboxGroups[name].push($this);
32.        // Pokud je počet zaškrtnutých checkboxů v dané skupině větší než max, odškrtneme první
        zaškrtnutý checkbox
33.        if (checkboxGroups[name].length > max) {
34.          checkboxGroups[name].shift().prop('checked', false);
35.        }
36.      } else {
37.        // Pokud je checkbox odškrtnutý, odstraníme ho ze skupiny
38.        var index = checkboxGroups[name].indexOf($this);
39.        if (index > -1) {
40.          checkboxGroups[name].splice(index, 1);
41.        }
42.      }
43.    });
44.
45.    // Při odeslání formuláře
46.    jQuery('form').on('submit', function(e){
47.      var checkboxNames = ['vedomi[]', 'mobilita[]', 'komunikace[]', 'pomucky[]', 'dychani[]',
        'vyziva[]', 'moc[]', 'stolice[]', 'vstupy[]', 'kuze[]', 'bolest[]', 'chovani[]'];
48.      for (var i = 0; i < checkboxNames.length; i++) {
49.        // Zkontrolujeme, zda je v každé skupině checkboxů alespoň jeden zaškrtnutý
50.        var checked = jQuery('input[name=" + checkboxNames[i] + "]:checked').length > 0;
51.        if (!checked) {
52.          // Pokud ne, zobrazíme upozornění a odeslání formuláře zastavíme
53.          alert('V každém poli prosím zaškrtněte aspon jednu možnost.');
```

```

54.         e.preventDefault();
55.         return;
56.     }
57. }
58. });
59.
60. // Funkce pro manipulaci s inputy a checkboxy
61. var handleInputChange = function(inputId, checkboxId, prefix) {
62.     // Když se změní stav checkboxu
63.     jQuery('#' + checkboxId).change(function(){
64.         if (this.checked) {
65.             // Pokud je zaškrtnutý, nastavíme focus na příslušný input
66.             jQuery('#' + inputId).focus();
67.         } else {
68.             // Pokud je odškrtnutý, vymažeme hodnotu příslušného inputu
69.             jQuery('#' + inputId).val("");
70.         }
71.     });
72.
73.     // Když se změní hodnota inputu
74.     jQuery('#' + inputId).on('input', function(){
75.         var $this = jQuery(this);
76.         var text = $this.val();
77.         // Nastavíme hodnotu checkboxu na prefix + text z inputu
78.         jQuery('#' + checkboxId).val(prefix + text);
79.     });
80. };
81.
82. // Konfigurace inputů a checkboxů
83. var inputConfigs = [
84.     { inputId: 'pomuckyInput', checkboxId: 'pomuckyJine', prefix: 'Jiné: ' },
85.     { inputId: 'vyzivaInput1', checkboxId: 'vyzivaJine1', prefix: 'Dieta č.: ' },
86.     { inputId: 'vyzivaInput2', checkboxId: 'vyzivaJine2', prefix: 'Tekutiny: ' },
87.     { inputId: 'vyzivaInput3', checkboxId: 'vyzivaJine3', prefix: 'Jiné: ' },
88.     { inputId: 'stoliceInput', checkboxId: 'stoliceJine', prefix: 'Příměs: ' },
89.     { inputId: 'vstupyInput1', checkboxId: 'vstupyJine1', prefix: 'PŽK: ' },
90.     { inputId: 'vstupyInput2', checkboxId: 'vstupyJine2', prefix: 'CŽK: ' },
91.     { inputId: 'vstupyInput3', checkboxId: 'vstupyJine3', prefix: 'Drény: ' },
92.     { inputId: 'vstupyInput4', checkboxId: 'vstupyJine4', prefix: 'Jiné: ' },
93.     { inputId: 'kuzelInput', checkboxId: 'kuzelJine', prefix: 'Lokalizace: ' },
94. ];
95.
96. // Pro každou konfiguraci zavoláme funkci handleInputChange
97. inputConfigs.forEach(function(config) {
98.     handleInputChange(config.inputId, config.checkboxId, config.prefix);
99. });
100.
101. // Speciální případ pro checkbox 'bolestJine'
102. jQuery('#bolestJine').change(function(){
103.     if (this.checked) {
104.         // Pokud je zaškrtnutý, nastavíme focus na první input
105.         jQuery('#bolestInput1').focus();
106.     } else {
107.         // Pokud je odškrtnutý, vymažeme hodnoty všech tří inputů
108.         jQuery('#bolestInput1').val("");
109.         jQuery('#bolestInput2').val("");
110.         jQuery('#bolestInput3').val("");
111.     });
112.
113. // Když se změní hodnota kteréhokoli z tří inputů
114. jQuery('#bolestInput1, #bolestInput2, #bolestInput3').on('input', function(){
115.     var text1 = jQuery('#bolestInput1').val();
116.     var text2 = jQuery('#bolestInput2').val();
117.     var text3 = jQuery('#bolestInput3').val();
118.     // Nastavíme hodnotu checkboxu na 'Charakter: ' + text1 + ' ; Lokalizace: ' + text2 + ' ; Stupeň: ' + text3
119.     jQuery('#bolestJine').val('Charakter: ' + text1 + ' ; Lokalizace: ' + text2 + ' ; Stupeň: ' + text3);
120. });
121. });
122. </script>

```

### 3.2.5 Jednoduchý algoritmus pro vyhledávání dat v tabulkách

---

```
1. <script>
2.   $(document).ready(function() {
3.     // Když uživatel napíše do prvku s ID "search"
4.     $("#search").on("keyup", function() {
5.       // Převědeme hodnotu vstupu na malá písmena
6.       var value = $(this).val().toLowerCase();
7.
8.       // Pro každý řádek v těle tabulky s ID "table-tbody"
9.       $("#table-tbody tr").filter(function() {
10.        // Zobrazíme nebo skryjeme řádek na základě toho, zda text řádku obsahuje hodnotu vstupu
11.        $(this).toggle($(this).text().toLowerCase().indexOf(value) > -1);
12.      });
13.    });
14.  });
15. </script>
```



### 3.3 Popis pro uživatele

#### 3.3.1 Přihlašovací stránka

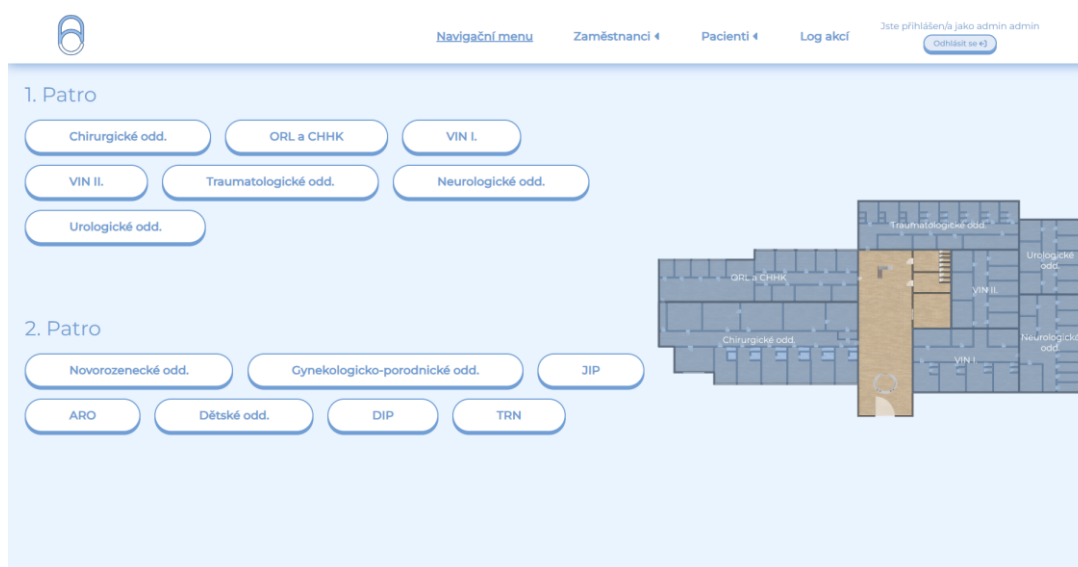
Při rozkliknutí webové stránky uživatel zadá přihlašovací údaje, které mu dal jeho zaměstnavatel. Pokud se vypíše chybová hláška „Vypĺňte všechny údaje!“, tak některý z údajů chybí. Ovšem, pokud se objeví chybová hláška „Špatně zadané údaje!“, tak buď údaje nebyly správně zadány, anebo se uživatel v databázi nenachází.



Obrázek 12 - Přihlašovací stránka

#### 3.3.2 Navigační menu

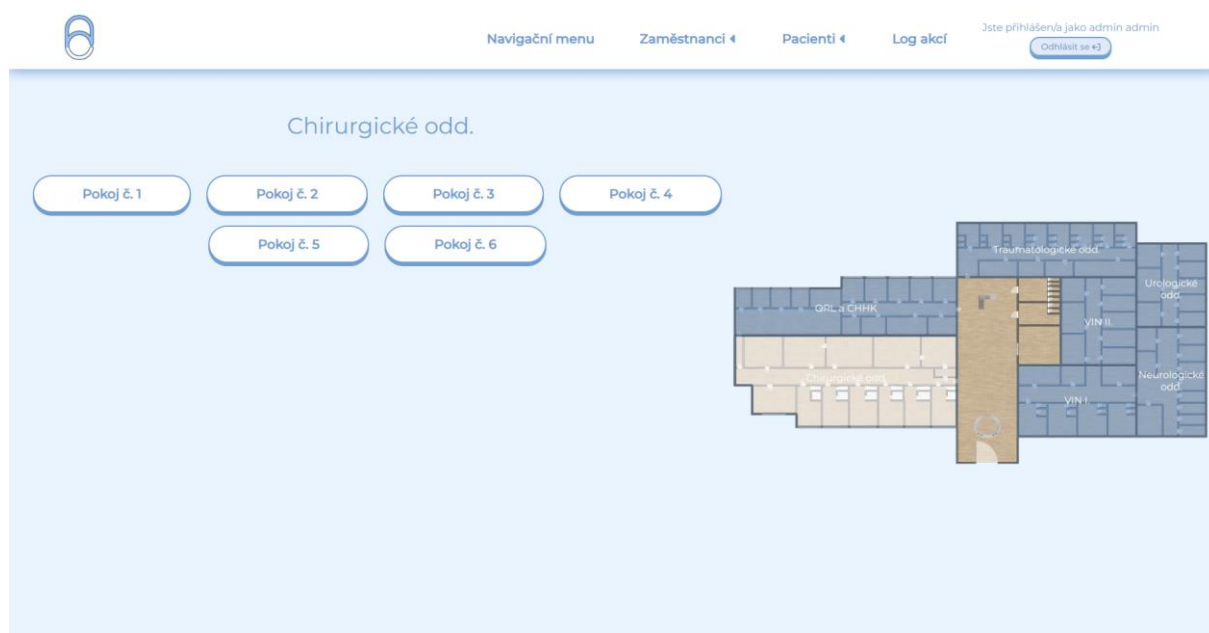
V horním navigačním panelu se uživatel může dostat téměř kamkoliv v aplikaci, anebo se pomocí tlačítka „Odhlásit se“ odhlásit. Když uživatel najede kurzorem na nějaké tlačítko u jednotlivých pater, zobrazí se na mapě vyznačené oddělení. Po kliknutí se uživatel dostane na vybrané oddělení.



Obrázek 13 - Navigační menu

### 3.3.3 Jednotlivá oddělení

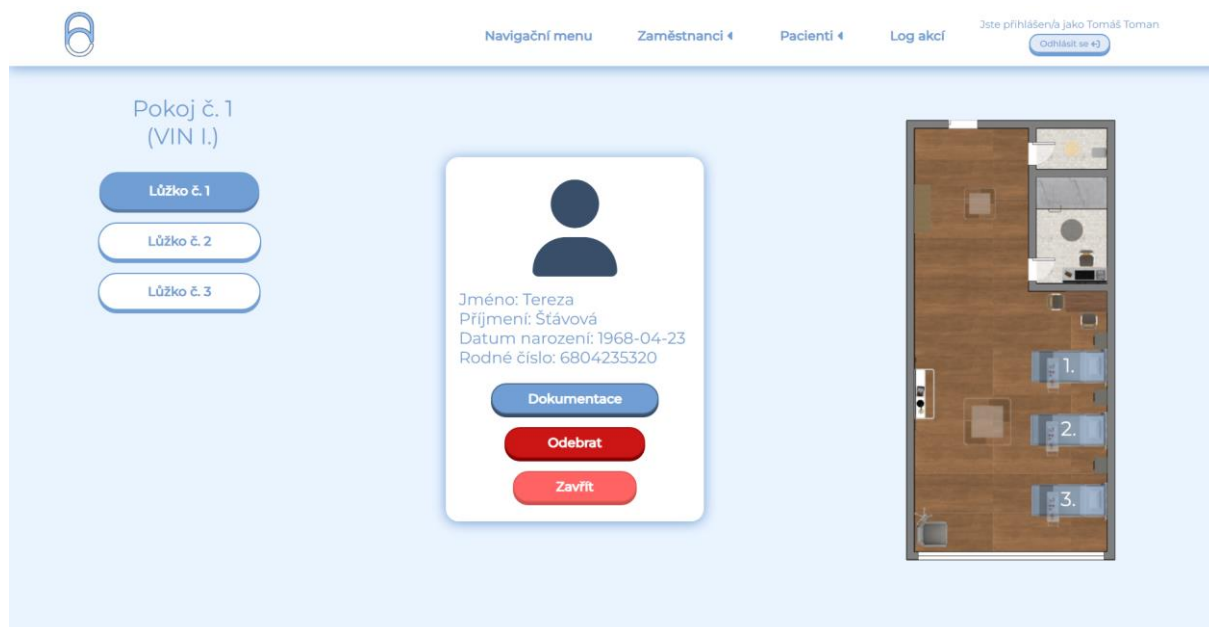
Po přesunutí na dané oddělení si uživatel může vybrat pokoj, na který se chce dále přesunout. Na mapě je kvůli přehlednosti zvýrazněno oddělení, ve kterém se právě nachází.



Obrázek 14 - Stránka s oddělením

### 3.3.4 Jednotlivé pokoje

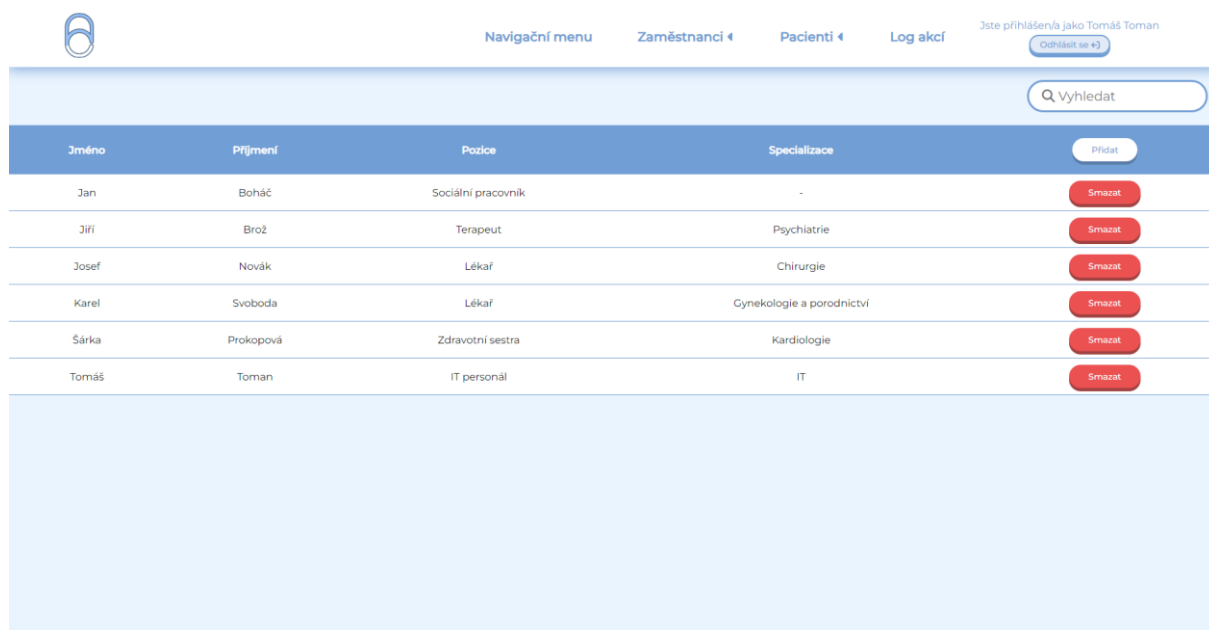
Když uživatel najede kurzorem na tlačítko lůžka, dané lůžko se zvýrazní. Po stisknutí tlačítka se zobrazí tabulka s informacemi o pacientovi, který na lůžku právě leží. Pokud na lůžku žádný pacient neleží objeví se možnost „Přidat“ a po stisknutí se zobrazí tabulka se všemi pacienty bez lůžka.



Obrázek 15 - Stránka pokoje

### 3.3.5 Zobrazovací tabulka zaměstnanců

Uživatel na této stránce uvidí všechny zaměstnance v databázi, pokud má patřičná oprávnění bude je moct i odebrat z databáze. V pravém horním rohu je možnost si zaměstnance vyhledat pro rychlejší práci se systémem.

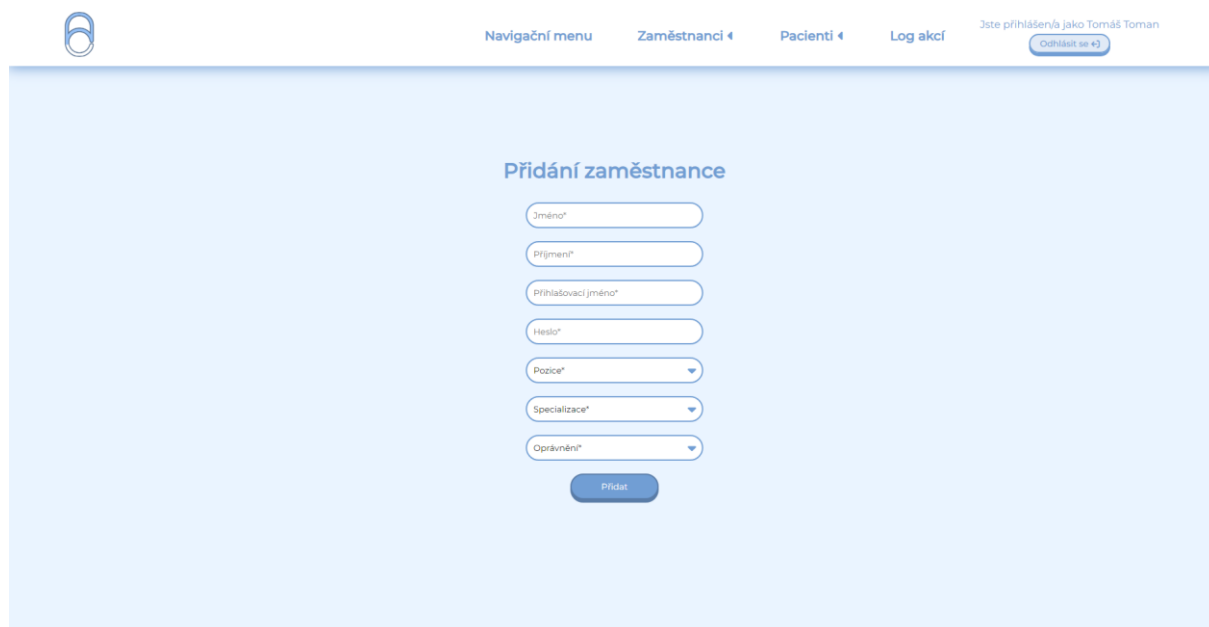


Jméno	Příjmení	Pozice	Specializace	Přidat
Jan	Boháč	Sociální pracovník	-	Smazat
Jiří	Brož	Terapeut	Psychiatrie	Smazat
Josef	Novák	Lékař	Chirurgie	Smazat
Karel	Svoboda	Lékař	Cynekologie a porodnictví	Smazat
Šárka	Prokopová	Zdravotní sestra	Kardiologie	Smazat
Tomáš	Toman	IT personál	IT	Smazat

Obrázek 16 - Zobrazovací tabulka zaměstnanců

### 3.3.6 Stránka pro přidání zaměstnance do databáze

Na této stránce uživatel s příslušnými oprávněními může přidat nového zaměstnance do databáze. Stačí pouze zadat požadované údaje.



Přidání zaměstnance

Jméno\*

Příjmení\*

Přihlašovací jméno\*

Heslo\*

Pozice\*

Specializace\*

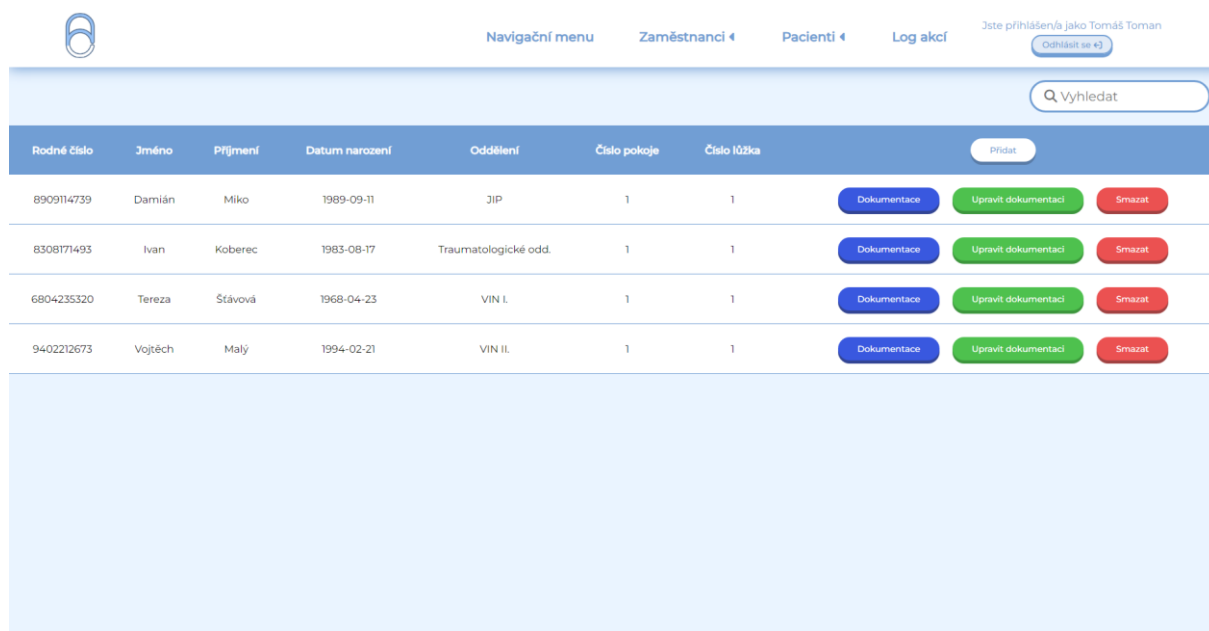
Oprávnění\*

Přidat

Obrázek 17 - Přidání zaměstnance

### 3.3.7 Zobrazovací tabulka pacientů

V této tabulce zaměstnanec zjistí všechny potřebné údaje jako jsou rodná čísla pacientů, jména, popřípadě na jakém pokoji se aktuálně nachází. Po pravé straně jsou tři tlačítka a s příslušnými oprávněními je uvidíte. Modré slouží na zobrazení dokumentace, zelené na její upravení a červené na jeho smazání z dokumentace. Nahoře už je jen tlačítko, které slouží pro přesměrování na stránku, kde se pacienti přidávají a nad tím pole pro vyhledávání v tabulce.

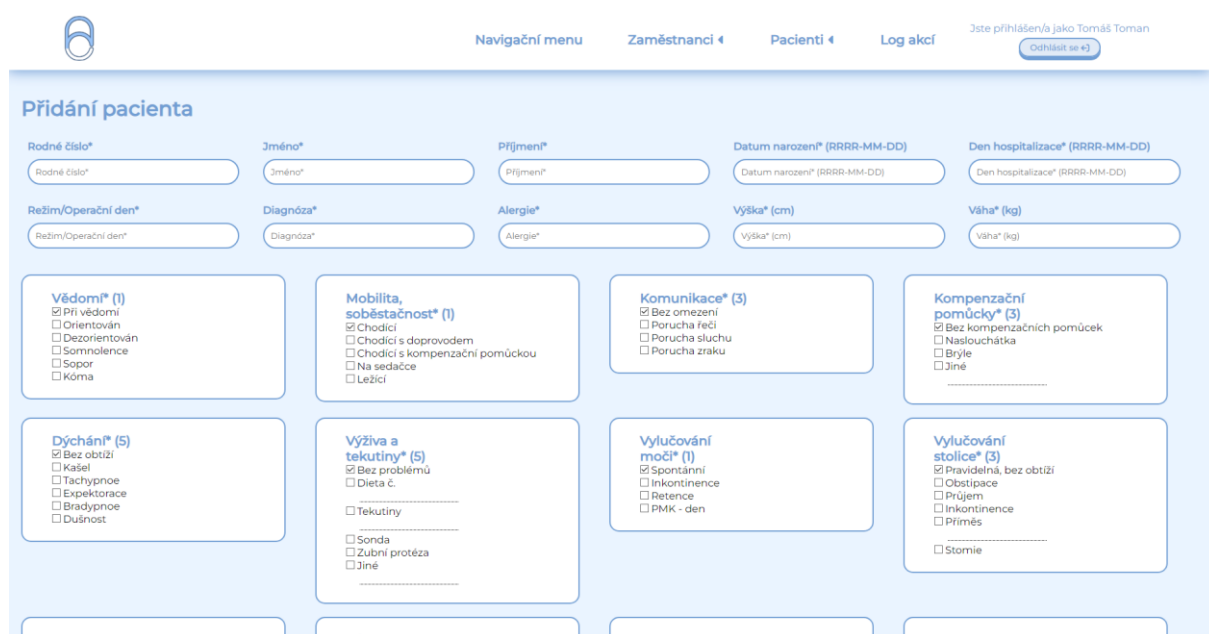


Rodné číslo	Jméno	Příjmení	Datum narození	Oddělení	Číslo pokoje	Číslo lůžka	
8909114739	Damián	Miko	1989-09-11	JIP	1	1	<button>Dokumentace</button> <button>Upravit dokumentaci</button> <button>Smazat</button>
8308171493	Ivan	Koberec	1983-08-17	Traumatologické odd.	1	1	<button>Dokumentace</button> <button>Upravit dokumentaci</button> <button>Smazat</button>
6804235320	Tereza	Štávková	1968-04-23	VIN I.	1	1	<button>Dokumentace</button> <button>Upravit dokumentaci</button> <button>Smazat</button>
9402212673	Vojtěch	Malý	1994-02-21	VIN II.	1	1	<button>Dokumentace</button> <button>Upravit dokumentaci</button> <button>Smazat</button>

Obrázek 18 - Tabulka pacientů

### 3.3.8 Přidání pacienta do databáze

Na této stránce, která slouží k přidání pacienta do databáze, je nutné vyplnit všechny požadované údaje a následně kliknout na tlačítko „Přidat“ umístěné dole na stránce. Pro přístup na tuto stránku je zapotřebí mít patřičná oprávnění.



**Přidání pacienta**

Rodné číslo\*  Jméno\*  Příjmení\*  Datum narození\* (RRRR-MM-DD)  Den hospitalizace\* (RRRR-MM-DD)

Režim/Operační den\*  Diagnóza\*  Alergie\*  Výška\* (cm)  Váha\* (kg)

**Vědomí\* (1)**  
☒ Při vědomí  
☐ Orientován  
☐ Dezorientován  
☐ Somnolence  
☐ Sopor  
☐ Kóma

**Mobilita, soběstačnost\* (1)**  
☒ Chodící  
☐ Chodící s doprovodem  
☐ Chodící s kompenzační pomůckou  
☐ Na sedačce  
☐ Ležící

**Komunikace\* (3)**  
☒ Bez omezení  
☐ Porucha řeči  
☐ Porucha sluchu  
☐ Porucha zraku

**Kompenzační pomůcky\* (3)**  
☒ Bez kompenzačních pomůcek  
☐ Naslouchátka  
☐ Brýle  
☐ Jiné

**Dýchání\* (5)**  
☒ Bez obtíží  
☐ Kašel  
☐ Tachypnoe  
☐ Expektorace  
☐ Bradypnoe  
☐ Dušnost

**Výživa a tekutiny\* (5)**  
☒ Bez problémů  
☐ Dieta č.  
☐ Tekutiny  
☐ Sonda  
☐ Zubní protéza  
☐ Jiné

**Vylučování moči\* (1)**  
☒ Spontánní  
☐ Inkontinence  
☐ Retence  
☐ PMK - den

**Vylučování stolice\* (3)**  
☒ Pravidelná, bez obtíží  
☐ Obstipace  
☐ Průjem  
☐ Inkontinence  
☐ Příměs  
☐ Stomie


Obrázek 19 - Přidání pacienta 1

<b>Vstupy* (4)</b> <input checked="" type="checkbox"/> Bez vstupů <input type="checkbox"/> PŽK <input type="checkbox"/> ČŽK <input type="checkbox"/> Drény <input type="checkbox"/> Jiné	<b>Kůže* (6)</b> <input checked="" type="checkbox"/> Bez změn <input type="checkbox"/> Ikterus <input type="checkbox"/> Cyanóza <input type="checkbox"/> Dekubit <input type="checkbox"/> Běrcový vřed <input type="checkbox"/> Hematom <input type="checkbox"/> Lokalizace	<b>Bolest* (1)</b> <input checked="" type="checkbox"/> Bez bolesti <input type="checkbox"/> Bolest: Charakter Lokalizace Stupěň	<b>Chování, spánek* (3)</b> <input checked="" type="checkbox"/> Klidný <input type="checkbox"/> Neklidný <input checked="" type="checkbox"/> Vyrovnaný <input type="checkbox"/> Nevyrovnaný <input checked="" type="checkbox"/> Spolupracující <input type="checkbox"/> Nespolupracující
<b>Vyšetření*</b>                       	<b>Léčba/Léky*</b>                       	<b>Ordinace/ Ošetřovatelská péče*</b>                       	

Obrázek 20 - Přidání pacienta 2

### 3.3.9 Dokumentace pacientů

Na této stránce se vám zobrazí kompletní zpráva daného pacienta, ke které se lze dostat přes tabulku pacientů.



[Navigační menu](#)
[Zaměstnanci](#)
[Pacienti](#)
[Log akcí](#)

Jste přihlášen/a jako Tomáš Toman  
[Odhlásit se](#)

#### Dokumentace Tereza Šťávořová

Rodné číslo:	6804235320
Jméno:	Tereza
Příjmení:	Šťávořová
Datum narození:	1968-04-23
Den hospitalizace:	2024-05-01
Režim/Operační den:	Klidový
Diagnóza:	Zánět močového měchýře, DM, Hypertenze
Výška (cm):	150
Váha (kg):	90
Vědomí:	Dezorientován
Mobilita, soběstačnost:	Ležící
Komunikace:	Porucha řeči, Porucha sluchu
Kompenzační pomůcky:	Brýle
Dýchání:	Dušnost
Výživa a tekutiny:	Dieta č. 9, Tekutiny: 2l, Zubní protéza
Vylučování moči:	PMK - den
Vylučování stolice:	Průjem, Inkontinence
Vstupy:	PŽK: 2
Kůže:	Běrcový vřed, Lokalizace: Pravá noha
Bolest:	Bez bolesti

Obrázek 21 - Dokumentace pacientů

### 3.3.10 Úprava dokumentace

Na této stránce je možnost upravit informace ohledně stavu pacienta. Pro přístup k úpravě jsou žádoucí patřičná oprávnění. Odkaz na úpravu dokumentace se nachází v tabulce s pacienty.

**Úprava dokumentace**

Rodné číslo\* 6804235320 Jméno\* Tereza Příjmení\* Šťávoval Datum narození\* (RRRR-MM-DD) 1968-04-23 Den hospitalizace\* (RRRR-MM-DD) 2024-05-01

Režim/Operační den\* Klidový Diagnóza\* Zánět močového měchýře, DM, Hypertenze Alergie\* - Výška\* (cm) 150 Váha\* (kg) 90

**Vědomí\* (1)**  
☐ Při vědomí  
☐ Orientován  
☒ Deorientován  
☐ Somnolence  
☐ Sopor  
☐ Kóma

**Mobilita, soběstačnost\* (1)**  
☐ Chodící  
☐ Chodící s doprovodem  
☐ Chodící s kompenzační pomůckou  
☐ Na sedačce  
☒ Ležící

**Komunikace\* (3)**  
☐ Bez omezení  
☒ Porucha řeči  
☒ Porucha sluchu  
☐ Porucha zraku

**Kompenzační pomůcky\* (3)**  
☐ Bez kompenzačních pomůcek  
☐ Naslouchátka  
☒ Brýle  
☐ Jiné

**Dýchání\* (5)**  
☐ Bez obtíží  
☐ Kašel  
☐ Tachypnoe  
☐ Expektorační  
☐ Bradypnoe  
☒ Dušnost

**Výživa a tekutiny\* (5)**  
☐ Bez problémů  
☐ Sonda  
☒ Zubní protéza  
☒ Dieta č.  
9  
☒ Tekutiny  
2l  
☐ Jiné

**Vylučování moči\* (1)**  
☐ Spontánní  
☐ Inkontinence  
☐ Retence  
☒ PMK - den

**Vylučování stolice\* (3)**  
☐ Pravidelná, bez obtíží  
☐ Obstipace  
☒ Průjem  
☐ Inkontinence  
☐ Stomie  
☐ Příměs

Obrázek 22 - Úprava dokumentace

### 3.3.11 Log akcí

V logu akcí je možné zobrazení všech akcí, které proběhly v databázi. K vyhledání akcí slouží funkce umístěná v pravém horním rohu.

**Log akcí**

Q vyhledat

Datum a čas	Akce
2024-05-12 15:57:52	Tomáš Toman se odhlásil/a.
2024-05-12 15:57:22	Jan Boháč se odhlásil/a.
2024-05-12 15:38:08	Šárka Prokopová se odhlásil/a.
2024-05-12 15:37:27	Jan Boháč se odhlásil/a.
2024-05-12 15:34:48	Jan Boháč se odhlásil/a.
2024-05-12 15:30:28	Jan Boháč se odhlásil/a.
2024-05-12 15:28:55	Šárka Prokopová se odhlásil/a.
2024-05-12 15:25:57	Šárka Prokopová přidal/a pacienta Tereza Šťávoval na lůžko.
2024-05-12 15:25:56	Šárka Prokopová odebral/a pacienta Tereza Šťávoval z lůžka.
2024-05-12 15:21:49	Jan Boháč se odhlásil/a.
2024-05-12 15:14:30	Tomáš Toman se odhlásil/a.
2024-05-12 14:57:37	Jan Boháč se odhlásil/a.
2024-05-12 14:20:30	Tomáš Toman se odhlásil/a.
2024-05-12 14:08:39	Tomáš Toman přidal/a zaměstnance Jan Boháč do databáze.
2024-05-12 14:04:15	Tomáš Toman přidal/a zaměstnance Jan Boháč do databáze.
2024-05-12 14:03:36	Tomáš Toman přidal/a zaměstnance Jan Boháč do databáze.
2024-05-08 16:08:35	Tomáš Toman přidal/a zaměstnance Jiří Brož do databáze.
2024-05-08 16:07:17	Tomáš Toman přidal/a zaměstnance Karel Svoboda do databáze.
2024-05-08 16:06:10	Tomáš Toman přidal/a zaměstnance Josef Novák do databáze.

Obrázek 23 - Log akcí

## Závěr

---

Mým cílem bylo vytvořit webovou aplikaci, která usnadní zaměstnancům nemocnic správu pacientů a zvýší jejich efektivitu. Díky zkušenostem mé přítelkyně, která pracuje v oblasti zdravotnictví, jsem si uvědomil potřebu modernizace současných systémů, které často trpí zastaralostí a obtížnou použitelností.

Výsledná aplikace má přehledný a jednoduchý design, který by měl být snadno použitelný pro všechny zaměstnance nemocnic. Přihlašování do aplikace je zabezpečeno pomocí hesla a databáze chráněna proti SQL injection útokům.

Navigace v aplikaci je intuitivní díky hornímu navigačnímu panelu a přehlednému navigačnímu menu, které umožňuje rychlý přístup k různým částem systému, včetně oddělení, pokojů a map budov.

Důležitou součástí aplikace je možnost přidávání a odebírání pacientů z lůžek, zobrazení jejich informací a kompletní dokumentace. Pro správce aplikace je k dispozici také správa zaměstnanců.

Rozsáhlá dokumentace pacientů je plně automatizovaná a zobrazuje veškerá relevantní data z databáze. Vyhledávání v tabulkách je implementováno pomocí jednoduchého jQuery algoritmu.

Log akcí ukládá veškeré události prováděné v databázi, což zajišťuje transparentnost a správnou správu systému. Zaměstnanci mají rozdělená oprávnění podle svého zaměstnání.

Vývoj aplikace mi obohatil dovednosti v PHP, AJAX a jQuery. Použil jsem také aplikace Coohom pro tvorbu map a Adobe Photoshop pro úpravy grafiky. Tento projekt bude součástí mého portfolia a přispěje k mému profesnímu rozvoji.

## Použitá literatura

---

1. en.wikipedia.org. [Online] [https://en.wikipedia.org/wiki/Epic\\_Systems](https://en.wikipedia.org/wiki/Epic_Systems).
2. www.emrfinder.com. [Online] <https://www.emrfinder.com/wp-content/uploads/2016/09/Epic-EMR-Software-Patient-Summary-Overview-Vital-Signs.jpg>.
3. ehr.meditech.com. [Online] <https://ehr.meditech.com>.
4. ehr.meditech.com. [Online] <https://ehr.meditech.com/themes/ehrmeditech/images/home/genomics-screen-cap--homepage.jpg>.
5. code.visualstudio.com. [Online] <https://code.visualstudio.com/docs>.
6. play.google.com. [Online] <https://play.google.com/store/apps/details?id=com.coohom.capp&hl=cs&gl=US>.
7. cs.wikipedia.org. [Online] [https://cs.wikipedia.org/wiki/Adobe\\_Photoshop](https://cs.wikipedia.org/wiki/Adobe_Photoshop).
8. www.drawplanet.cz. [Online] <https://www.drawplanet.cz/co-je-tedy-figma/>.
9. cs.wikipedia.org. [Online] [https://cs.wikipedia.org/wiki/Font\\_Awesome](https://cs.wikipedia.org/wiki/Font_Awesome).
10. digitalniarchitekti.cz. [Online] <https://digitalniarchitekti.cz/clanek/github/>.
11. cs.wikipedia.org. [Online] [https://cs.wikipedia.org/wiki/Hypertext\\_Markup\\_Language](https://cs.wikipedia.org/wiki/Hypertext_Markup_Language).
12. en.wikipedia.org. [Online] <https://en.wikipedia.org/wiki/CSS>.
13. www.rascasone.com. [Online] <https://www.rascasone.com/cs/blog/co-je-javascript-pro-zacatecniky>.
14. www.itnetwork.cz. [Online] <https://www.itnetwork.cz/javascript/oop/ajax-v-javascriptu-zakladni-dotazy>.
15. www.strafelda.cz. [Online] <https://www.strafelda.cz/jquery>.



## Seznam obrázků

---

Obrázek 1 - Epic EMR Software (2).....	9
Obrázek 2 – Meditech (4) .....	10
Obrázek 3 - Myšlenková mapa .....	14
Obrázek 4 - Návrh databáze .....	14
Obrázek 5 - Návrh loga .....	15
Obrázek 6 - Návrh Figma 1 .....	15
Obrázek 7 - Návrh Figma 2 .....	16
Obrázek 8 - Návrh Figma 3 .....	16
Obrázek 9 - Návrh Figma 4 .....	16
Obrázek 10 - Návrh Figma 5 .....	17
Obrázek 11 - Návrh Figma 6 .....	17
Obrázek 12 - Přihlašovací stránka .....	25
Obrázek 13 - Navigační menu.....	25
Obrázek 14 - Stránka s oddělením .....	26
Obrázek 15 - Stránka pokoje .....	26
Obrázek 16 - Zobrazovací tabulka zaměstnanců .....	27
Obrázek 17 - Přidání zaměstnance.....	27
Obrázek 18 - Tabulka pacientů.....	28
Obrázek 19 - Přidání pacienta 1 .....	28
Obrázek 20 - Přidání pacienta 2 .....	29
Obrázek 21 - Dokumentace pacientů.....	29
Obrázek 22 - Úprava dokumentace.....	30
Obrázek 23 - Log akcí.....	30

## Obsah média

---

1. Složka „Patient care“ -> Samostatný projekt
2. Patient care – Dokumentace.docx -> Dokumentace v editovatelné verzi
3. Patient care – Dokumentace.pdf -> Dokumentace ve formátu pdf
4. Export databáze.sql -> Export databáze