



**FAKULTA
INFORMAČNÍCH
TECHNOLÓGIÍ
ČVUT V PRAZE**

ZADÁNÍ DIPLOMOVÉ PRÁCE

Název:	Databáza pacientov pre Centrum epilepsie nemocnice Motol
Student:	Bc. Emil Seman
Vedoucí:	Ing. Petr Ježdík, Ph.D.
Studijní program:	Informatika
Studijní obor:	Webové a softwarové inženýrství
Katedra:	Katedra softwarového inženýrství
Platnost zadání:	Do konce letního semestru 2018/19

Pokyny pro vypracování

Cílem práce je dokončení rozpracovaných částí, implementácia nových a revízia informačného systému EPICHIR pre Nemocnicu Motol.

Funkcionálne požiadavky:

- dokončenie a vylepšenie reportovacieho modulu pre operačné návrhy,
- vytvorenie nového vyhľadávania, jednoduché na používanie,
- opraviť známe chyby v systéme.

V rámci zadania je taktiež potrebné urobiť revíziu dát v databáze, zaviesť nové obmedzenia a prípadne zrefaktorovať niektoré tabuľky, aby sa predišlo nekonzistencii dát. Pri revízii databáze urobiť návrh na opravu prípadných dátových nezrovnalostí.

Nefunkcionálne požiadavky:

- aktualizovať dokumentáciu,
- naimplementovať používanie certifikátov a ich automatické obnovovanie.

Seznam odborné literatury

Dodá vedoucí práce.

Ing. Michal Valenta, Ph.D.
vedoucí katedry

doc. RNDr. Ing. Marcel Jiřina, Ph.D.
děkan

V Praze dne 14. února 2018



**FAKULTA
INFORMAČNÍCH
TECHNOLÓGIÍ
ČVUT V PRAZE**

Diplomová práce

Databáza pacientov pre Centrum epilepsie nemocnice Motol

Bc. Emil Seman

Katedra softwarového inženýrství
Vedúci práce: Ing. Petr Ježdík, Ph.D.

15. februára 2019

Pod'akovanie

Týmto by som chcel pod'akovať svojmu vedúcemu práce Petrovi Ježdíkovi za ľudský prístup a neformálne pracovné prostredie, ktoré vytvoril.

Prehlásenie

Prehlasujem, že som predloženú prácu vypracoval(a) samostatne a že som uviedol(uviedla) všetky informačné zdroje v súlade s Metodickým pokynom o etickej príprave vysokoškolských záverečných prác.

Beriem na vedomie, že sa na moju prácu vzťahujú práva a povinnosti vyplývajúce zo zákona č. 121/2000 Sb., autorského zákona, v znení neskorších predpisov, a skutočnosť, že České vysoké učení technické v Praze má právo na uzavrenie licenčnej zmluvy o použití tejto práce ako školského diela podľa § 60 odst. 1 autorského zákona.

V Prahe 15. februára 2019

.....

České vysoké učení technické v Praze

Fakulta informačních technologií

© 2019 Emil Seman. Všetky práva vyhrazené.

Táto práca vznikla ako školské dielo na FIT ČVUT v Prahe. Práca je chránená medzinárodnými predpismi a zmluvami o autorskom práve a právach súvisiacich s autorským právom. Na jej využitie, s výnimkou bezplatných zákonných licencií, je nutný súhlas autora.

Odkaz na túto prácu

Seman, Emil. *Databáza pacientov pre Centrum epilepsie nemocnice Motol*. Diplomová práca. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2019.

Abstrakt

Cieľom diplomovej práce bola úprava informačného systému EPICHIR na základe požiadaviek od Kliniky detskej neurológie FN Motol, ktorá bola zadávateľom práce. Informačný systém EPICHIR je na báze webovej aplikácie a daným nemocničným oddelením je využívaný ako kartotéka pacientov trpiacích epilepsiou.

Časť tohto dokumentu opisuje implementáciu zmien vykonaných v informačnom systéme, menovite v používateľskom rozhraní a v databáze podľa chronologicky zadaných požiadaviek klienta.

V ďalšej časti je priblížený proces importovania dát z dokumentu tabuľkového editora do informačného systému pomocou konfigurovateľnej Java aplikácie vytvorenej k tomuto účelu.

Kľúčová slova databáza pacientov, Epichir, epilepsia, informačný systém, Java, Motol, Spring, webová aplikácia

Abstract

The goal of the master thesis was modification of the information system EPICHIR based on requirements from the Department of Paediatric Neurology, Motol University Hospital, which acted as a contracting authority. Information

system EPICHIR is built on a web application, and is used by the department as a register of patients suffering from epilepsy.

A part of this document describes the implementation of changes made in the information system, specifically in the user interface and in the database according to the requirements chronologically assigned by the client.

Next section illustrates the process of importing the data from the table editor document to the information system through a configurable Java-based application created for this purpose.

Keywords Epichir, epilepsy, information system, Java, Motol, patients database, Spring, web application

Obsah

Úvod	1
1 Cieľ práce	3
2 Analýza súčasného systému	5
2.1 História a dôvod vzniku systému	5
2.2 Analýza kariet pacientov	5
2.3 Ostatné časti aplikácie	5
2.4 Technická analýza systému	6
2.5 Nedostatky systému EPICHR	10
3 Požiadavky klienta a ich realizácia	11
3.1 Metodika vykonávania jednotlivých zmien v systéme	11
3.2 Zmeny v súčasnom systéme EPICHR	11
3.3 Import aktuálnych dát	25
Záver	35
Literatúra	37
A Zoznam použitých skratiek a pojmov	39
A.1 Technické skratky	39
A.2 Medicínske skratky a pojmy	40
B Obsah priloženej SD karty	41
C Technická dokumentácia ku aplikácii na import dát z Excel súboru do databáze	43
C.1 Štruktúra kódu	43
C.2 Funkcionalita	43
C.3 Závislosti aplikácie (dependencies)	45

Zoznam obrázkov

21	Zakládne vizuálne rozhranie aplikácie Epichir	6
22	Diagram nasadenia aplikácie Epichir a príslušnej DB	7
23	Diagram obsluhy klienta pomocou MVC aplikácie postavenej na Springu	8
31	Klasifikácia epileptických záchvatov ILAE 2017, zdroj: [1]	19
32	Kategória Outcome - AED	24
33	Zobrazenie viacerých záznamov týkajúcich sa konkrétnych operácií u pacienta	26
34	Zobrazenie záznamov, kde bolo potrebné vytvoriť 2 nové záznamy histopatológií u daného pacienta	30
35	Zobrazenie záznamov na import do kategórie Komplikácie	31

Zoznam tabuliek

21	Zoznam základných tabuliek v databáze a k nim prislúchajúcich entít	9
22	Zoznam základných spoločných stĺpcov pre tabuľky kategórie pacienta	9
31	Mapovacia tabuľka zmeny číselníkových hodnôt Typ komplikácie .	16
32	Mapovacia tabuľka zmeny číselníkových hodnôt Histopatológie . .	17
33	Mapovacia tabuľka zmeny číselníkových hodnôt FCD klasifikácie .	18
34	Mapovacia tabuľka zmeny číselníkových hodnôt Intrakraniálne elektródy	18
35	Príklad importu záznamu komplikácie jedného pacienta do reálnych 3 záznamov v databáze	31
C1	Konfigurovatelné Java properties pre aplikáciu Importer	44
C2	Závislosti aplikácie importer na vybraných maven balíkoch	45

Úvod

V súčasnej dobe existuje mnoho rôznych informačných systémov zameraných na medicínsku tematiku. Informačný systém s databázou pacientov pre Centrum epilepsie nemocnice Motol je jedným z nich. Vznikol za účelom uchovávanía špecifických informácií o pacientoch trpiacich epilepsiou.

Systém je používaný každodenne lekármi vo FN Motol, ktorí s týmito pacientmi pracujú a očakávajú od systému, že im bude spoľahlivo poskytovať informácie a na základe nich im pomáhať liečiť pacientov. Výskum v oblasti epilepsie neustále napreduje, preto doktori potrebujú systém prispôbovať a neustále aktualizovať. Napríklad pridávaním možnosti uchovávať ďalšie, podrobnejšie dáta o zdravotnom stave ich pacientov, či už pomocou nových klasifikácií zdravotných príznakov, alebo možnosťami pre pokročilejšie vyhľadávanie medzi pacientmi.

Práca teda opisuje zmeny vykonané v informačnom systéme, ktoré vychádzajú priamo z požiadaviek lekárov pracujúcich so systémom. Taktiež práca opisuje aj načítanie aktuálnych dát do systému. Motivácia k tomuto načítaniu dát vznikla z dôvodu neaktuálnosti a inkonzistencie informácií uložených o pacientoch v systéme.

Cieľ práce

Cieľom práce podľa zadania bolo dokončiť rozpracované časti systému, implementácia nových častí a všeobecná revízia informačného systému EPICHR pre Nemocnicu Motol. Zadanie taktiež obsahuje viaceré funkcionálne a nefunkcionálne požiadavky. V tomto prípade Nemocnica Motol, konkrétne Klinika detskej neurológie, vystupuje ako **klient** a zadávateľ práce. Cieľom práce bolo primárne vyjsť klientovi v ústrety s požiadavkami, hoci sa primárne nezhodovali a väčšinou ani neprekrevývali s funkcionálnymi a nefunkcionálnymi požiadavkami vyplývajúcimi zo zadania. Požiadavky klienta vyžadovali úpravy informačného systému, ako používateľského rozhrania, aplikačnej logiky, tak aj samotnej databáze systému. Mimo iné klient vyžadoval vykonať aj import dát z tabuľkového dokumentového formátu priamo do informačného systému. Jedným z cieľov bolo teda vymyslieť spôsob ako tieto dáta naimportovať bez straty a poškodenia terajších dát v databáze a ako vyriešiť konflikty v dátach pri danom importe.

Sumárnym cieľom teda bolo funkcionálne vylepšiť informačný systém EPICHR a obohatiť ho o nové prvky, ktoré klient vyžadoval pri práci so systémom. Taktiež bola poskytovaná podpora k informačnému systému, jeho aktuálne nasadenej verzii a databáze prislúchajúcej k nemu.

Analýza súčasného systému

Aplikácia EPICHIR je webovou aplikáciou, ktorá umožňuje doktorom a výskumným pracovníkom evidovať dáta o pacientoch, ktorí trpia na epilepsiu a sú liečení výskumnými metódami vo FN Motol. Aplikácia pozostáva z kartotéky pacientov, administrácie používateľov, pokročilým vyhľadáváním a administráciou prekladov. Kartotéka pacientov sa oproti iným kartotékam líši najmä tým, že je špecifická pre pacientov trpiacich epilepsiou a zohľadňuje v sebe rôzne klasifikácie epilepsií a experimentálne liečby epilepsie.

2.1 História a dôvod vzniku systému

Informačný systém vznikol v rámci predchádzajúcich bakalárskych prác [2] za účelom uľahčenia evidencie pacientov Klinike detskej neurológie FN Motol, z dôvodu nutnosti evidovať špecifické dáta o jednotlivých pacientov, čo iné kartotéky neumožňovali. Aplikácia vznikala podobne ako bolo pristupované k zmenám opisovaným v tomto dokumente, teda po pravidelných konzultáciách s klientom, iteračne. [2]

2.2 Analýza kariet pacientov

Každý pacient v systéme má svoju vlastnú kartu (príklad na obrázku 21). Karta pozostáva z viacerých kategórií viditeľných v menu na ľavej strane. V týchto kategóriách sú uložené záznamy o vyšetreniach pacienta a jeho zdravotnom stave. Kategórie môžu pozostávať z viacerých záznamov, pričom záznamy v rámci kategórie prislúchajú jednej entite.

2.3 Ostatné časti aplikácie

Aplikácia umožňuje vyhľadávať pacientov podľa rozličných kritérií, na základe možnosti nadefinovania, ktoré hodnoty z ktorých kategórií musí záznam pa-

2. ANALÝZA SUČASNÉHO SYSTÉMU

The screenshot displays the EPICHIR application interface. At the top, there is a header with the EPICHIR logo, user information (sue), and language options (Czech and English). The main content area is divided into several sections:

- Pacient:** A sidebar on the left lists various test categories such as 'Přehled', 'Anamnéza', 'Záchvaty', 'Farmakoterapie', 'Neurologické nálezy', 'Neuropsychologie', 'Neuropsychologie - old', 'Diagnostické testy', 'Invasivní testy', 'Operace', 'Histologie', 'Komplikace', 'Outcome - záchvaty', and 'Outcome - AED'.
- Pacient:** A central section with tabs for 'Export pacienta', 'Ověření', 'Upravit', and 'Smazat'. It displays patient details: 'Pacient: [avatar]', 'Rodné číslo: [ID]', 'Datum narození: 2015-11-03 (3 let 3 měsíců)', 'Ošetřující lékař: [avatar]', 'Pohlaví: Muž', 'Věk při začátku epilepsie: - let', 'Adresa:', 'Telefon:', and 'Email:'. A note states 'Záznamy pacienta nejsou ověřeny'.
- Anamnéza:** A section titled 'Anamnéza' with a 'Zobrazit všechny' link. It lists medical history items with 'Ne' responses: 'Epilepsie v rodině', 'Pre- a perinatální rizika', 'Febrilní křeče', 'Zánět CNS', 'Úraz CNS', 'Operace CNS', 'Časná psychomotorická retardace', and 'První záchvat s horečkou'.
- Pacienti:** A sidebar at the bottom left lists 'Kartotéka' and 'Pokročilé vyhledávání'.

Obr. 21: Zakládne vizuálne rozhranie aplikácie Epichir

cienta spĺňať, aby bol nájdený. Následne je možné dáta z kariet nájdených pacientov exportovať do tabuľkového formátu Excel.

Aplikácia umožňuje aj prezeráť jednotlivé lokalizácie a administrovať select-boxy (v tomto dokumente nazývané číselníky), avšak zmeny v nich sa neprejavujú v záznamoch jednotlivých pacientov.

2.4 Technická analýza systému

2.4.1 Architektúra systému

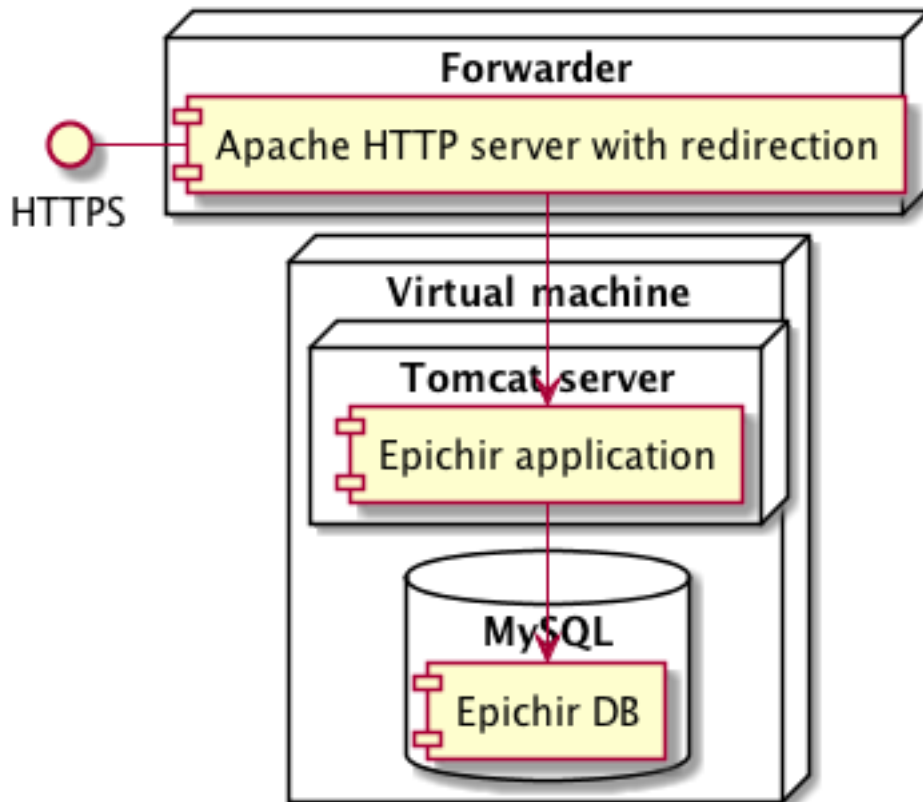
Celá aplikácia pozostáva z viacerých častí.

Ako databáza pre aplikáciu slúži MySQL riešenie, ktoré beží na produkčnom serveri ako unix-ová služba. K dátam v tejto databáze pristupuje aplikácia priamo, cez datasource poskytnutý aplikačným serverom Tomcat.

Aplikácia beží na aplikačnom serveri Tomcat. Server Tomcat má nakonfigurovaný prístup do databáze cez MySQL JDBC adaptér, ktorý je dostupný pre aplikácie bežiacie na serveri pod JNDI menom `jdbc/EpichirDB` ako Resource (typ objektu v Jave).

Kód aplikácia je napísaný v Jave, pričom je využitý aj framework Spring. Spring poskytuje, okrem iného, tieto prostriedky, ktoré sú primárne využívané v rámci aplikácie EPICHIR:

- nástroje pre ľahšiu manipuláciu s HTTP Request-ami a Response-ami



Obr. 22: Diagram nasadenia aplikácie Epichir a príslušnej DB

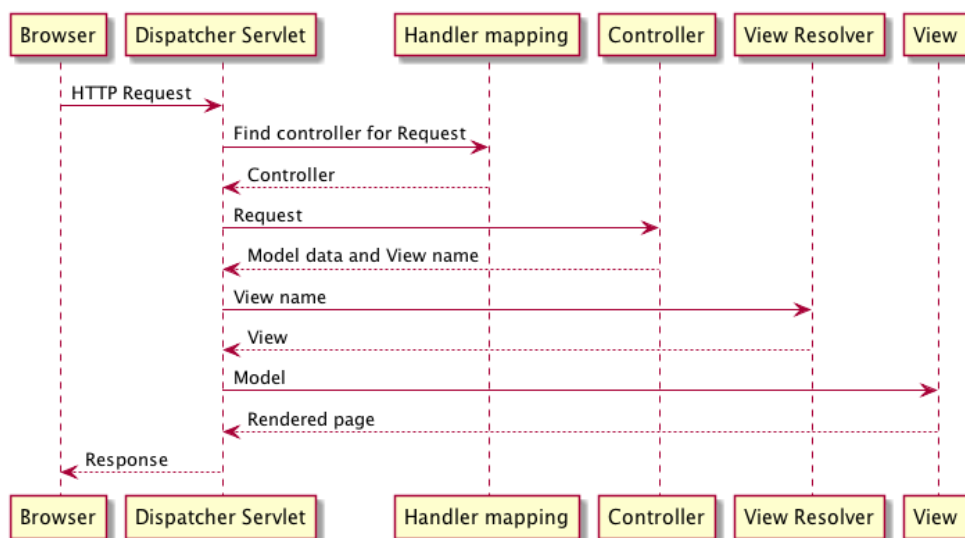
- nástroje pre jednoduchú implementáciu návrhového vzoru MVC - Model View Controller - viac v sekcii 2.4.2
- pre manipuláciu s entitami a ich napojením na databázu Spring využíva Hibernate ako nástroj implementujúci ORM (Object-relation mapping), teda mapovanie entít priamo na databázu, bez nutnosti písania SQL dopytov
- nástroje pre autorizáciu a autentikáciu používateľov

2.4.2 Popis fungovania MVC v Springu

Na obrázku 23 je možné vidieť princíp, ako funguje MVC v Springu.

Všetky názvy, ktoré sú tu spomenuté sú Java Bean-y (Java objekty, ktoré implementujú **Serializable**, majú public konštruktor a premenné sú privátne, dostupné cez getters a setters). Existuje hlavný Dispatcher Servlet, teda Servlet cez ktorý prejdú všetky Request-y v aplikácii, ktorý na základe URL cesty

2. ANALÝZA SUČASNÉHO SYSTÉMU



Obr. 23: Diagram obsluženia klienta pomocou MVC aplikácie postavenej na Springu

a HTTP metódy si vyžiada z Handler Mapping-u (služba, ktorá mapuje jednotlivé Java Beans ku URL cestám) daný Controller. Potom Servlet prepošle Request do Controlleru, ktorý ho spracuje, prípadne aplikuje nejakú logiku, načíta si potrebné entity z databáze a vytvorí Model (objekt, ktorý slúži na prenos dát), ktorý pošle naspäť do Servletu s menom View, ktorý by sa mal zavolať. Dispatcher Servlet si zistí cez View Resolver, ktoré konkrétne View, v tomto prípade vo forme JSP (JavaServer Pages - technológia, ktorá umožňuje vytvárať servlety pomocou šablón postavených na kombinácii XML, HTML a Java kódu) sa má zavolať, následne ho zavolá, JSP vytvorí odpoveď - Response a vráti ju cez Dispatcher Server priamo klientovi.

JSP v Springu sa môže skladať z menších stavebných celkov a to JSTL (Java Standard Tag Library) formulárov. JSTL umožňuje vytvárať šablóny jednotlivých formulárov pomocou XML súborov. [3]

2.4.3 Analýza dátových entít a databázových tabuliek

Hlavnou entitou v systéme je **PatientEntity**, reprezentujúca fyzickú kartu pacienta v kartotéke, ktorá má kardinalitu ku záznamom v jednotlivých kategóriách OneToMany (jeden záznam ku viacerým záznamom). Kategóriou nazývame podoblasť karty pacienta, ktorá uchováva napríklad operácie, záchvaty pacienta, neurologické nálezy, záchvaty, atď. Ku jednému pacientovi teda môže existovať viacero záznamov v rámci jednotlivých kategórií (aj keď niektoré sú limitované aplikačnou logikou na jeden záznam, napríklad kategória Anamnéza). Prehľad tabuliek a k nim pridružených entít využívaných v tejto

Tabuľka 21: Zoznam základných tabuliek v databáze a k nim prislúchajúcich entít

Dátová entita v aplikácii	DB pohľad ku ktorej je entita napojená cez ORM	DB tabuľka ku danému DB pohľadu
SelectBoxEntity		select_box
SelectBoxItemEntity		select_box_item
LocalizationEntity		localization
ComplicationEntity	complication_valid	complication
DiagnosticTestScalpEegEntity	diagnostic_test_scalp_eeg_valid	diagnostic_test_scalp_eeg
SeizureDetail	seizure_detail_valid	seizure_detail
OutcomeEntity	outcome_valid	outcome
InvasiveTestEegEntity	invasive_test_eeg_valid	invasive_test_eeg
HistologyEntity	histology_valid	histology

Tabuľka 22: Zoznam základných spoločných stĺpcov pre tabuľky kategórie pacienta

Názov stĺpca	Nenulový	Dátový typ	Typ uložených dát
id	áno	INT	Primárny kľúč
add_user_id	áno	INT	Cudzí kľúč do tabuľky user , označuje používateľa ktorý vytvoril záznam
added	áno	TIMESTAMP	Čas úpravy záznamu
patient_id	áno	INT	Cudzí kľúč do tabuľky patient , označuje pacienta ku ktorému sa viaže záznam
history	áno	TINYINT(1)	Označuje archívny záznam
hidden	áno	TINYINT(1)	Označuje skrytý záznam
date	áno	DATETIME	Dátum vyšetrenia, ku ktorému sa záznam viaže

práci je v tabuľke 21. Celý diagram databázových tabuliek je prístupný na priloženom médiu.

Väčšina tabuliek určených na ukladanie záznamov v jednotlivých kategóriách karty pacienta obsahujú základné údaje viditeľné v tabuľke 22.

2.5 Nedostatky systému EPICHIR

2.5.1 Export dát do formátu Excel

Export dát je momentálne možný len z pokročilého vyhľadávania a to zadaním vyhľadávacích kritérií a následným exportom buď jednotlivých výsledkov vyhľadávania alebo všetkých výsledkov súčasne. Chýba však možnosť exportovať celú kartotéku priamo, bez nutnosti vyhľadávania, čo klient v súčasnosti obchádza vyhľadávaním so žiadnymi parametrami. V prípade, že používateľ požaduje export všetkých pacientov, tak je to časovo a výkonovo náročné pre server, a stálo by za zváženie do budúca to vykonávať v rámci samostatných procesorových vláken a ukladať výsledný dokument. Neskôr by si používateľ mohol vyzdvihnúť vyexportovaný dokument napr. v schránke používateľa.

2.5.2 Používanie zastaralých JSP View

V aplikácii sa používajú pre vytváranie používateľského rozhrania JSP s JSTL formuláre. V dnešnej dobe sú tieto technológie na ústupe. V súčasných aplikáciách je trendom oddeliť back-end s logikou od front-endu. Riešením tejto problematiky by mohlo byť napríklad postavenie Rest API (Representational State Transfer Application Programming Interace) na strane back-enu, ktoré by mohlo byť využívané oddelenou front-endovou aplikáciou, postavenou napríklad na niektorej z javascriptových knižníc.

2.5.3 Administrácia select-boxov

Aplikácia síce má možnosť meniť položky v číselníkoch, avšak po ich zmene sa nevykoná žiadna transformácia dát v záznamoch jednotlivých pacientov. Preto bolo nutné v rámci tejto práce všetky takéto zmeny vykonávať ručne.

Požiadavky klienta a ich realizácia

3.1 Metodika vykonávania jednotlivých zmien v systéme

Pri vývoji softvéru bola použitá metodika Agile. Zmeny v systéme boli vykonávané postupne vždy po dohode s klientom, pričom, ak klient vyžadoval niektorú zmenu naimplementovať promptne a uprednostniť danú zmenu pred už predom dohodnutými zmenami, tak bolo klientovi vyhovené. Komunikácia s klientom prebiehala najčastejšie pomocou emailov a zároveň občasnými schôdzami priamo u klienta, kde bol dohodnutý nasledovný postup a boli odprezentované implementačné návrhy zmien, ktoré vychádzali z klientových požiadaviek. Tento postup vychádza z Manifesta agilného vývoja, ktorý napríklad hovorí o uprednostnení reagovať na zmeny pred dodržovaním predom dohodnutého plánu, alebo o funkčnom softvéri pred vyčerpávajúcou dokumentáciou. [4].

3.2 Zmeny v súčasnom systéme EPICHIR

Zmeny sú spísané chronologicky, tak ako boli vykonané po predchádzajúcej dohode s klientom. Zmeny vychádzajú priamo z požiadaviek klienta.

Ku každej zmene sú spísané nutné úpravy kódu a databáze ktoré boli vykonávané. Úpravy databáze prebiehali pomocou SQL skriptu na lokálnom prostredí na aktuálnom obraze produkčnej databáze. Po dokončení a verifikácii funkčnosti skriptu pomocou aplikácie EPICHIR a aj manuálneho nahliadania do dát, sa skript nahral a vykonal na produkčnej databáze. Simultáne bola nahraná aj nová verzia aplikácie na aplikačný server Tomcat, aby boli zmeny vždy konzistentné. Nahrávanie nových verzií a spúšťanie skriptov prebiehalo v nočných hodinách, keď pracovníci nemocnice Motol nepracovali s informačným systémom EPICHIR.

3. POŽIADAVKY KLIENTA A ICH REALIZÁCIA

V prípade zmien v číselníkových položkách bolo nutné vykonávať úpravy v troch rôznych databázových tabuľkách. V tabuľke `select_box` sa nachádzala samotná identifikácia číselníka vrátane názvu číselníka (`select_name`), ku každému číselníku prislúchali viaceré číselníkové hodnoty z tabuľky `select_box_item`. Hodnoty číselníka obsahovali cudzí kľúčom do tabuľky `select_box` a svoj vlastný primárny kľúč `select_id`. Zobrazované číselníkové hodnoty pre jednotlivé jazyky v aplikácii EPICHR sa nachádzali v tabuľke určenej pre lokalizácie s názvom `localization` a ich kódom `code`. K jednotlivým lokalizáciám pre číselníkové hodnoty aplikácia pristupovala pomocou vyskladaného cez už spomínaný stĺpec `code` vo formáte `label.select_name.select_id`, ktorý je vyskladaný samotnou aplikáciou.

Pri vykonávaní zmien v kóde alebo databáze bolo v maximálnej možnej miere prihliadané na zachovávanie už predošlej konvencie kódu a dodržiavanie už nasadených štandardov, aby bolo možné predísť nezrovnalostiam v budúcnosti, aby bola zachovaná čitateľnosť kódu či orientácia v databázových tabuľkách a aby sa predišlo zbytočnému refaktoringu kódu v budúcnosti.

3.2.1 Pridanie položky do číselníka AED a zoradenie položiek

Do kategórie pacienta Farmakoterapia klient požadoval pridať záznam PP ako číselníkovú hodnotu do rozbaľovacieho menu prislúchajúcemu k číselníkovej položke **AED**.

3.2.1.1 Zmeny v databáze

Do tabuľky `select_box_item` bol pridaný nový záznam so `select_id = 28` a zároveň do tabuľky `localization` bola pridaná lokalizácia k danému prvku, teda **PP** pre oba jazyky.

3.2.1.2 Zmeny v kóde

Pre zoradenie istých číselníkových položiek podľa abecedy bola implementovaná funkcionálna služba `SelectBoxServiceImpl`, ktorá načítava jednotlivé položky číselníkov z databázy pre použitie v rozbaľovacích zoznamoch. Položky číselníkov sú načítavané vo forme mapy, kde kľúč aj hodnota je typu `String`. Vo viacerých číselníkoch sa však používa aj položka **Nevyplnené**, ktorá má byť na prvom mieste v zozname a má kľúč **0**. Názvy číselníkových položiek boli zoradené podľa abecedy s použitím streamov z Javy 8 a metódy `sorted`, s výnimkou kľúča s hodnotu **0**, ktorý bol zoradený vždy na prvé miesto. Takto upravená mapa je vrátená ako atribút pre daný `View` pri navštívení príslušnej stránky, cez `Controller` ktorý ju získa už zo spomenutej služby `SelectBoxServiceImpl`.

Princíp fungovania View a Controller v sekcii 2.4.2.

3.2.2 Pridanie položky High-Density EEG

Do kategórie Skialpové EEG spadajúcej pod Diagnostické testy, klient požadoval pridať jednoduchú zaškrťavaciu položku s pravdivostnou hodnotou Áno / Nie s názvom **High Density EEG**.

3.2.2.1 Zmeny v databáze

Do tabuľky **diagnostic_test_scalp_eeg** bol pridaný stĺpec **high_density_eeg** s typom **TINYINT(1)** (čo je ekvivalent pre **BOOLEAN** [5]) a do tabuľky **localization** bol pridaný preklad pre High-Density EEG. K danej tabuľke **diagnostic_test_scalp_eeg** bol vymazaný starý a vytvorený nový databázový pohľad.

3.2.2.2 Zmeny v kóde

Položka High-Density EEG bola pridaná do JSTL formuláru pre kategóriu Skialpové EEG, taktiež do dátovej entity **DiagnosticTestScalpEegEntity** pod názvom **highDensityEeg** a do služby slúžiacej na export Xslx súborov.

3.2.3 Pridanie položky Včasnú pooperačné záchvaty

Do kategórie Komplikácie, klient požadoval pridať jednoduchú zaškrťavaciu položku s pravdivostnou hodnotou Áno / Nie s názvom **Včasnú pooperačné záchvaty** pre každý záznam, ktorý je možné vytvoriť v rámci kategórie Komplikácie.

3.2.3.1 Zmeny v databáze

Do tabuľky **complication** bol pridaný stĺpec **early_postoperation_seizures** s typom **TINYINT(1)** a do tabuľky **localization** bol pridaný preklad pre Včasnú pooperačné záchvaty. K danej tabuľke **complication** bol vymazaný starý a vytvorený nový databázový pohľad (**complication_valid**).

3.2.3.2 Zmeny v kóde

Do entity **ComplicationEntity** bola pridaná boolean hodnota **earlyPostoperationSeizures**, do exportovacej služby pre Xslx súbory bola táto hodnota pridaná ku exportom komplikácií a taktiež bol do JSTL formulárov pre komplikácie (**complicationForm** a **complicationTable**) pridaný zaškrťovací box ku tejto hodnote.

3.2.4 Pridanie číselníkovej položky Očakávaný neurologický deficit

Klient požadoval pridať číselníkovú položku **Očakávaný neurologický deficit** do kategórie Komplikácie. Hodnoty číselníkovej položky sú nasledovné:

0. Bez deficitu
1. Afázie
2. Hemiparéza
3. Paréza hlavových nervů
4. Porucha zorného pole
5. Mnestická porucha
6. Jiné

3.2.4.1 Zmeny v databáze

Do tabuľky `select_box` bol pridaný nový záznam so `select_name = expectedNeurologicalDeficit`. Do tabuľky `select_box_item` bolo pridaných 7 záznamov reprezentujúcich jednotlivé číselníkové hodnoty, k nim boli pridané preklady do tabuľky `localization` s kódmi `label.expectedNeurologicalDeficit.ID`, kde ID je poradové číslo hodnoty číselníka, a zároveň bol pregenerovaný databázový pohľad `complication_valid`.

3.2.4.2 Zmeny v kóde

Do entity `ComplicationEntity` bola pridaná nová int hodnota `expectedNeurologicalDeficit`, do Controllera `ComplicationController` bolo pridané načítavanie hodnôt číselníka pre rozbaľovací zoznam, ktoré sú predané do View ako Model. Do JSTL formulárov pre komplikácie (`complicationForm` a `complicationTable`) bolo pridané rozbaľovacie menu s položkami číselníka získanými z Modelu.

3.2.5 Pridanie číselníkovej položky Funkčný následok komplikácie

Do kategórie Komplikácie klient požadoval pridať číselníkovú položku **Funkčný následok komplikácie** s nasledovnými hodnotami:

0. Nezvoleno
1. Afázie

2. Hemiparéza
3. Paréza hlavových nervů
4. Porucha zorného pole
5. Porucha hlavových nervů
6. Mnesticá porucha
7. Smrt
8. Jiné

3.2.5.1 Zmeny v databáze

Do tabuľky s číselníkmi `select_box` bol pridaný nový záznam so `select_name = functional_result` a k nemu boli pridané prislúchajúce záznamy do tabuľky `select_box_item` podľa číselníkových hodnôt vyššie. Preklady boli pridané do tabuľky `localization` s kódmi v tvare `label.functionalResult.ID`. Nakoniec bol vymazaný stávajúci a vygenerovaný nový databázový pohľad `complication_valid`.

3.2.5.2 Zmeny v kóde

Do entity `ComplicationEntity` bola pridaná nová int hodnota `functionalResult`, v Controlleri `ComplicationController` bol Model posiadaný do View obohatený o položky daného číselníka a v JSTL formulároch pre komplikácie (`complicationForm` a `complicationTable`) boli pridané hodnoty položky daného číselníka v rámci nového rozbaľovacieho menu.

3.2.6 Prepracovanie kategórie Komplikácie

V rámci kategórie Komplikácie bolo potrebné premenovať číselníkovú položku **Komplikácie** na **Etiológiu komplikácie**. Potom číselníkovú položku **Typ komplikácie** premenovať na **Tiaž komplikácie**. Nasledovne premenovať číselníkovú položku **Priebeh komplikácie** na **Typ komplikácie**, tým pádom pôvodná položka **Typ Komplikácie** je nahradená inou. Zároveň **Typ komplikácie** má nadobudnúť nové číselníkové hodnoty. Mapovanie týchto hodnôt je v tabuľke 31.

Do kategórie Komplikácie bolo taktiež potrebné pridať dátumovú položku, označujúcu ku operácii z ktorého dňa sa daný záznam komplikácie viaže.

3. POŽIADAVKY KLIENTA A ICH REALIZÁCIA

Tabuľka 31: Mapovacia tabuľka zmeny číselníkových hodnôt Typ komplikácie

Pôvodná číselníková hodnota v Priebeh komplikácie	Pôvodné ID v zázname	Nové ID v zázname	Nová číselníková hodnota v Typ komplikácie
Nezadané	0	0	Bez komplikácií
Bez komplikácií	1	0	Bez komplikácií
S komplikáciami	2	1	Komplikácia resekcie
-	-	2	Komplikácia invazívneho moni- torovania

3.2.6.1 Zmeny v databáze

V tabuľke **complication** bol premenovaný stĺpec **complication_type** na stĺpec **complication_weight**, k tomu príslušný záznam v tabuľke **select_box** bol upravený na **select_name = complicationWeight** a následne aj kódy všetkých prekladov v tabuľke **localization** z hodnoty stĺpca **code = label.complicationType.ID** na **code = label.complicationWeight.ID**.

Potom bol v tabuľke **complication** premenovaný stĺpec **with_complication** na stĺpec **complication_type**, taktiež príslušný záznam v tabuľke **select_box** bol upravený na **select_name = complicationType** (z pôvodného nekonzistentného názvu **process**) a následne aj kódy všetkých prekladov v tabuľke **localization** z hodnoty stĺpca **code = label.process.ID** na **code = label.complicationType.ID**.

Hodnoty v terajšom stĺpci **complication_type** tabuľky **complication** boli pretransformované podľa tabuľky 31.

Pre pridanie dátumu operácie bol pridaný stĺpec **to_operation_date** do tabuľky **complication**.

Následne bol pregenerovaný databázový pohľad **complication_valid**.

3.2.6.2 Zmeny v kóde

Entita **ComplicationEntity** bola upravená tak aby odpovedala tabuľke **complication**, teda premenné boli premenované podobne ako stĺpce v tabuľkách. Podobne bola upravená služba slúžiaca na export dát a jednotlivé JSTL formuláre a JSP View súbory.

3.2.7 Prepracovanie kategórie Histológia

V rámci kategórie Histológie bolo potrebné kompletne prepacovať číselníkové položky **Histopatológia**, **Klasifikácia FCD** a zároveň pretransformovať aktuálne záznamy pacientov pod kategóriou Histológia podľa mapovacej tabuľky 32. Pre záznamy ktoré mali nastavenú **Histopatológiu** na hodnotu FCD bolo potrebné nastaviť aj novú číselníkovú hodnotu pre **FCD klasifikáciu** a to

3.2. Zmeny v súčasnom systéme EPICHR

Tabuľka 32: Mapovacia tabuľka zmeny číselníkových hodnôt Histopatológie

Pôvodná číselníková hodnota	Pôvodné ID v zázname	Nové ID v zázname	Nová číselníková hodnota
Zvoľte klasifikáciu	0	10	Nedostupná
FCD	1	3	HS (a FCD klasifikáciu nastaviť na FCD IIIa (+HS))
Gliálna lézie	2	8	Jazva
HS	3	3	HS
Hamarton	4	4	Hypothalamický hamarton
MCD iná	5	1	MCD
Nedostupná	6	10	Nedostupná
Normálna	7	0	Normálna
Nádor	8	2	Nádor
Posttraumatické zmeny	9	8	Jazva
SWS	10	7	Vaskulárne lézie
TSC	11	5	TSC
Vaskulárne lézie	12	7	Vaskulárne lézie
Zápalové lézie	13	6	Encefalitída
-	-	9	Kombinovaná patológia

FCD IIIa (+HS). Taktiež bolo potrebné pretransformovať doterajšie hodnoty číselníka **Klasifikácia FCD** v záznamoch pacientov podľa mapovacej tabuľky 33.

3.2.7.1 Zmeny v databáze

Boli nahrané nové preklady do tabuľky **localization** pre číselníkové hodnoty FCD Klasifikácia a Histopatológia, nadbytočné boli vymazané. Analogicky k tomu boli upravené záznamy v tabuľke **select_box_item**.

Pre transformáciu doterajších záznamov v rámci jednej SQL transakcie boli záznamy v tabuľke **histology** upravené nasledovne: hodnota stĺpca **histopathology** bola zvýšená o 100, aby pri transformovaní na nové ID podľa tabuľky 32 sa nepomiešali už transformované záznamy s doterajšími, následne boli aktualizované hodnoty už na konkrétne ID podľa mapovania.

Podobne sa postupovalo aj pri mapovaní doterajších hodnôt FCD Klasifikácie na nové.

3.2.7.2 Zmeny v kóde

Neboli nutné, nové číselníky neboli pridávané, len upravované už existujúce pomocou SQL skriptu.

3. POŽIADAVKY KLIENTA A ICH REALIZÁCIA

Tabuľka 33: Mapovacia tabuľka zmeny číselníkových hodnôt FCD klasifikácie

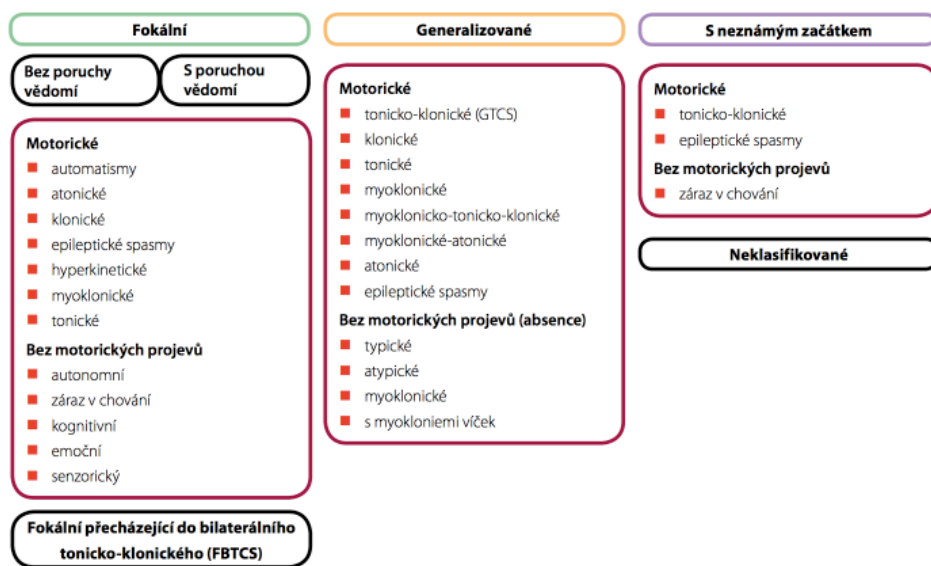
Pôvodná číselníková hodnota	Pôvodné ID v zázname	Nové ID v zázname	Nová číselníková hodnota
Nezvolené	0	0	Bez FCD
0	1	8	Nešpecifikovaný typ FCD
FCD typ IIa	2	2	FCD IIa
FCD typ IIb	3	2	FCD IIa
FCD typ Ia	4	1	FCD I
FCD typ Ib	5	1	FCD I
mMCD typ I	6	8	Nešpecifikovaný typ FCD
mMCD typ I	7	8	Nešpecifikovaný typ FCD
-	-	3	FCD IIb
-	-	4	FCD IIIa
-	-	5	FCD IIIb
-	-	6	FCD IIIc
-	-	7	FCD IIId

Tabuľka 34: Mapovacia tabuľka zmeny číselníkových hodnôt Intrakraniálne elektródy

Pôvodná číselníková hodnota	Pôvodné ID v zázname	Nové ID v zázname	Nová číselníková hodnota
Nezvolené	0	0	Nezvolené
Intracereb. + subdur. stripy + gridy	1	1	Intracerebreálne + subdurálne
Intracerebreálne	2	2	Intracerebreálne
Intracerebreálne + subdurálne gripy	3	1	Intracerebreálne + subdurálne
Intracerebreálne + subdurálne stripy	4	1	Intracerebreálne + subdurálne
Subdurálne stripy	5	3	Subdurálne
Subdurálne gridy	6	3	Subdurálne
Subdurálne stripy + gridy	7	3	Subdurálne

3.2.8 Zmena položiek v číselníku Intrakraniálne elektródy

Požiadavka klienta bola zlúčiť niektoré hodnoty v číselníku **Intrakraniálne elektródy**, ktorý spadá do kategórie Invazívni testy - iEEG. Zo súčasných 8 hodnôt mali vzniknúť len 4, pričom záznamy s pôvodnými hodnotami mali byť transformované na nové podľa mapovacej tabuľky 34.



Obr. 31: Klasifikácia epileptických záchvatov ILAE 2017, zdroj: [1]

3.2.8.1 Zmeny v databáze

V tabuľke **localization** boli upravené prvé 4 záznamy pre číselník Intrakraniálne elektródy, zvyšné s kódmi `label.intracranialElectrodes.ID` s ID 4 a vyššie boli vymazané. Podobne boli vymazané položky so `select_id` 4 a vyššie z tabuľky `select_box_item`. Následne boli upravené záznamy v tabuľke `invasive_test_eeg`, konkrétne stĺpec `intracranial_electrodes` podľa tabuľky 34.

3.2.8.2 Zmeny v kóde

Jediná nutná zmena sa udiala v JSP View `invasiveTestEegFieldset` kde boli vymazané nadbytočné hodnoty pre rozbaľovací zoznam s hodnotami číselníka Intrakraniálne elektródy.

3.2.9 Pridanie klasifikácie ILAE 2017

V kategórii záchvaty sú uchovávané záznamy o klasifikáciách a zaradení záchvatov u pacienta. Ku už existujúcej klasifikácii ILAE (International League Against Epilepsy - Medzinárodná liga proti epilepsii) bolo potrebné pridať novú klasifikáciu **ILAE 2017**. Ako podklad pre vytvorenie číselníkových hodnôt a rozbaľovacích položiek vo formulároch slúžil obrázok 31. To znamenalo pridať do kategórie Záchvaty pod detail konkrétneho záchvatu viaceré položky. Hlavná riadiaca číselníková položka je ILAE 2017 s hodnotami:

- Nezvoleno

3. POŽIADAVKY KLIENTA A ICH REALIZÁCIA

- Fokálne
- Generalizované
- S neznámym začiatkom

V prípade vybrania položky Fokálne, používateľovi sa zobrazia 3 nasledovné položky: rozbaľovací zoznam s číselníkovoými hodnotami Stav vedomia pozostávajúci z nasledovných hodnôt:

- Nevieme
- Bez poruchy vedomia
- S poruchou vedomia

, rozbaľovací zoznam s číselníkovoými hodnotami Fokálne - klasifikácia s týmito hodnotami:

- M: nezaradené
- M: automatizmy
- M: atonické
- M: klonické
- M: epileptické spasmy
- M: hyperkinetické
- M: myoklonické
- M: tonické
- BMP: nezaradené
- BMP: autonómne
- BMP: záraz v správaní
- BMP: kognitívne
- BMP: emočné
- BMP: senzorický

a zaškrtnutá položka FBTCS (Fokálne prechádzajúce do bilaterálneho tónicko-klonického).

V prípade vybrania položky Generalizované sa zobrazí len jedna ďalšia položka a to rozbaľovací zoznam Generalizované - klasifikácia s číselníkovoými hodnotami:

- M: nezaradené
- M: tonicko-klonické
- M: klonické
- M: tonické
- M: myoklonické
- M: myoklonicko-tonicko-klonické
- M: myoklonické-atonické
- M: atonické
- M: epileptické spasmy
- BMP: nezaradené
- BMP: typické
- BMP: atypické
- BMP: myoklonické
- BMP: s myokloniámi viečok

V prípade zvolenia číselníkovej hodnoty S neznámym začiatkom sa používateľovi zobrazí taktiež len jedna ďalšia položka: S neznámym začiatkom - klasifikácia s nasledovnými číselníkovými hodnotami:

- M: nezaradené
- M: tonicko-klonické
- M: epileptické spasmy
- BMP: nezaradené
- BMP: záraz v chovaní
- neklasifikované

3.2.9.1 Zmeny v databáze

V tabuľke **seizure_detail** bolo vytvorené stĺpce **ilae2017_classification**, **ilae2017_focal_state_of_consciousness**, **ilae2017_focal_type**, **ilae2017_generalized_type** a **ilae2017_unknown_start_type** typu **SMALLINT(6)**, ktoré slúžia pre ukladanie číselníkových hodnôt spomentých vyššie a stĺpec **ilae2017_focal_ftcs** typu **TINYINT(1)** pre uloženie pravdivostnej hodnoty.

Následne boli vytvorené záznamy v tabuľke **select_box**, jeden pre každý číselník spomenutý v sekcii 3.2.9. Ku každému z nich boli vytvorené záznamy podľa počtu prvkov v číselníkoch v tabuľke **select_box_item** a k nim aj preklady v tabuľke **localization**.

3.2.9.2 Zmeny v kóde

Do entity **SeizureDetailEntity** boli pridané premenné pre ukladanie číselníkových hodnôt a Do zložky s klientskými javascriptovými súbormi, ktoré ovládajú zmeny v používateľskom rozhraní bol pridaný nový súbor **seizureDetail.js** pre Detail záchvatov, ktorý zabezpečoval skrytie a zobrazenie skupín prvkov podľa aktuálne zvolenej klasifikácie ILAE 2017 z rozbaľovacieho menu. Do JSTL formulárov boli pridané jednotlivé rozbaľovacie položky a zaškrťavacie polia.

Do exportovacej služby Excel dokumentov boli ku záložke detail záchvatov pridané všetky novo spomínané hodnoty.

3.2.10 Úprava kategórie Outcome

Z dôvodu pridania samostatnej kategórie Outcome - AED (popísané v kapitole 3.2.11) bolo potrebné z kategórie Outcome presunúť doterajšiu položku **AED** a na základe klientových požiadaviek aj odstrániť položky **EEG**, **MRI** a **Neuropsychológia**.

Zároveň klient požadoval pridať zaškrťavacie políčko s názvom **Lost to follow-up** a rozbaľovacie menu s výberom číselníkových hodnôt **Follow-up** s nasledovnými hodnotami:

- Nedá sa posúdiť
- Bez záchvatov
- Redukcia záchvatov ≥ 90
- Redukcia záchvatov ≥ 50
- Redukcia záchvatov <50

3.2.10.1 Zmeny v databáze

Do tabuľky **outcome** boli pridané stĺpce **lost_to_follow_up** typu **TINY-INT(1)** a **follow_up** typu **SMALLINT(6)**. Do tabuľky **select_box** bol pridaný nový záznam označujúci nový číselník s hodnotou **select_name = followUp** a k nemu bolo pridaných 5 záznamov pre každú číselníkovú hodnotu do tabuľky **select_box_item** vrátane prekladov do tabuľky **localization**. Z dôvodu uchovania historických údajov stĺpce **aed**, **mri**, **eeg** a **neuropsychology** neboli odstránené.

3.2.10.2 Zmeny v kóde

Do entity **OutcomeEntity** boli pridané dve nové premenné **lostToFollowUp** a **followUp**. Do Controllera pre Outcome bolo pridané načítavanie Follow-up číselníkových hodnôt, ktoré boli pridané do formulárov pre použitie v rozbaľovacom menu, taktiež bola do formulárov pridaná aj zaškrtvacia položka pre Lost to follow-up. Novo pridané premenné do entity sú posielané z Controllera v rámci Model-u do jednotlivých View-s. Z entity **OutcomeEntity** boli odstránené doterajšie premenné pre MRI, EEG, AED a Neuropsychológiu, taktiež boli odstránené polia pre ich zobrazenie v príslušných JSTL formulároch.

3.2.11 Pridanie kategórie Outcome - AED

Klient si vyžiadal prídanie kategórie Outcome - AED, ktorá by ku každej operácii daného pacienta evidovala maximálne jeden záznam, ktorý hovorí o tom, či u pacienta boli vysadené lieky AED. Záznam tejto kategórie obsahuje **Dátum vyšetrenia**, **Dátum vysadenia** a číselníkovú položku **AED** s hodnotami:

- Nevysadené
- Vysadené
- Redukované

Ukážka zobrazenia maximálne jedného záznamu kategórie Outcome - AED je na obrázku 32.

3.2.11.1 Zmeny v databáze

Bola vytvorená nová databázová tabuľka reprezentujúca entitu z kódu **OutcomeAedEntity** s názvom **outcome_aed**, so základnými stĺpcami pre databázové tabuľky kategórií karty pacienta (bližšie popísané v tabuľke 22) a týmito ďalšími stĺpcami: **aed** typu **SMALLINT(6)** pre číselníkovú hodnotu AED, **operation_id** typu **INT** slúžiacu ako cudzí kľúč viažúci sa na id v tabuľke operácií, stĺpec s Dátumom vysadenia **dropped_date** a stĺpec

3. POŽIADAVKY KLIENTA A ICH REALIZÁCIA

Outcome - AED

Pacient:	██████████	Rodné číslo:	██████████	Datum narodení:	██████████
Ošetřující lékař:	██████████	Pohlaví:	Muž	Věk při začátku epilepsie:	- let
Adresa:		Telefon:		Email:	

Záznamy pacienta nejsou ověřeny

Operace ze dne: 2018-03-05

Outcome - AED	Datum vysazení	
Vysazený	2019-01-30	Upravit záznam

Komentář: Stav pacienta OK.

Operace ze dne: 2016-05-08

Outcome - AED	Datum vysazení	
		Přidat záznam

Obr. 32: Kategorie Outcome - AED

`comment` typu `TEXT`. Následně byl vytvořený databázový pohled `outcome_aed_valid` na který je cez ORM napojená daná entita.

3.2.11.2 Změny v kóde

Bola vytvorená nová entita `OutcomeAedEntity` s hodnotami `aed` (presunuté z kategórie Outcome) typu `int`, dátumom `droppedDate` označujúcim Dátum vysadenia typu `Date`, `operationId` typu `int` označujúcim id operácie a k tejto hodnote viažúca sa premenná `operation` typu `OperationEntity` slúžiaca pre ORM mapovanie s kardinalitou `ManyToOne` - čiže jedna operácia môže mať viacero záznamov (hoci bolo spomenuté, že ku každej operácii prislúcha maximálne jeden záznam, jedná sa o platný nehistorizovaný záznam, teda v databáze ich môže byť uložených viacero). Nasledovne bol vytvorený Controller `OutcomeAedController` podporujúci štandardné CRUD operácie pre danú entitu s metódami:

- `outcomeAedCreateGet` - metóda naplňujúca Model a vracajúca View pre vytvorenie záznamu,
- `outcomeAedCreatePost` - metóda ukladajúca nový záznam a vracajúca presmerovanie na list záznamov,
- `outcomeAedEditGet` - metóda naplňujúca Model a vracajúca View pre editáciu záznamu,
- `outcomeAedEditPost` - metóda ukladajúca editovaný záznam a vracajúca presmerovanie na list záznamov,

- `outcomeAedListGet` metóda naplňujúca Model jednotlivými záznamami Outcome - AED pre príslušné operácie pacienta a vracajúca View pre zobrazenie listu záznamov.

V metódach naplňujúcich Model pre View boli do Modelu pridané položky číselníka AED. Ku kategórii Outcome - AED boli vytvorené JSTL formuláre pre editáciu a vytváranie záznamu `outcomeAedForm`, pre zobrazenie jedného záznamu `outcomeAedTable` a pre zobrazenie všetkých záznamov v rámci jedného pacienta `outcomeAed`. Boli vytvorené 3 štandardné JSP View ktoré zahŕňujú vyššie spomenuté JSTL formuláre, a to pod cestou `views/patient/outcomeAed`.

3.2.12 Ostatné zmeny a opravy chýb

V systéme boli opravené niektoré nekritické chyby ako zmena hesla u používateľov, ukladanie zmien pri editácii používateľov, oprava exportov dát, oprava lazy načítavania entít z View či oprava automatického generovania certifikátov na produkčnom serveri.

3.3 Import aktuálnych dát

Keďže aplikácia dlhodobejšie nenaplnjala požiadavky klienta a neumožňovala ukladať niektoré špecifické klasifikácie u pacientov, klient udržoval aktuálne dáta len v proprietárnom dokumentovom formáte `.xlsx`. Takýto dokument je však náročné zdieľať a udržiavať aktuálny medzi všetkými členmi tímu. Tým pádom dáta v databáze systému EPICHR prestali byť aktuálne a nemali byť ako synchronizované s dátami udržiavanými v dokumentovom formáte. Z dôvodu neaktuálnosti dát sa teda klient rozhodol o import dát z dokumentového zošita Excel do databázy EPICHR-u. Dáta boli importované iteratívne, podľa dohody s klientom, po jednotlivých oblastiach.

3.3.1 Štruktúra importovaných dát

Každý riadok v Excel tabuľke prislúchal jednej operácii u konkrétneho pacienta. Ak mal teda pacient operácií viacero, mal pod sebou viacero riadkov, avšak už nemal vyplnené všetky dáta pre každú operáciu. Pacient, ktorý mal viacero operácií mal identifikáciu pacienta iba pri prvom zázname, ďalšie bolo už bez mena pacienta, jeho ID a dátumu narodenia, vid' obrázok 33.

Každý stĺpec v tabuľke bol označený názvom, nie však kategóriou, do ktorej hodnoty z daného stĺpca spadali. Bolo nutné komunikovať s klientom, do ktorej kategórie spadá daný stĺpec a ktorú položku označuje. Hodnoty v stĺpcoch mohli byť týchto formátov:

- Dátum

3. POŽIADAVKY KLIENTA A ICH REALIZÁCIA

PACIENT	číslo pacienta	datum narodení	pohlaví	y	časné pooperační záchvaty
x	x	x	2		0
x	x	x	1		0
					0
					0
x	x	x	1		0
x	x	x	1		1
x	x	x	2		0

Obr. 33: Zobrazenie viacerých záznamov týkajúcich sa konkrétnych operácií u pacienta

- Číselná hodnota
- Číselníková hodnota, ktorá prislúchala číselníku v prvých riadkoch daného stĺpca.

V prípade číselníkových hodnôt, bolo nutné manuálne prekontrolovať či sa poradie a položky číselníka zhodujú s tými v databáze aplikácie EPICHIR. V prípade nezahody s databázou EPICHIR boli upravené číselníkové hodnoty a hodnoty dát priamo v Excel tabuľke alebo boli hodnoty upravené samotným importom pomocou úpravy kódu importujúcej aplikácie.

3.3.2 Popis importujúcej aplikácie

Aplikácia je napísaná v Jazyku Java verzie 1.8. Je postavená na frameworku Spring Boot verzie 2.0, a teda aplikácia je samostatne spustiteľná, bez nutnosti aplikačného servera. Cieľ aplikácie je jednoduchý, a to import dát z formátu .xlsx priamo do MySQL databáze bežiacom na lokálnom prostredí. Toto riešenie umožňuje priamo skontrolovať zmenené dáta, ich správnosť voči zdrojovému dokumentu a možnosť vrátiť sa k zálohe, v prípade, že dáta neboli importované správne. V takomto prípade bolo možné okamžite opraviť kód importovacej aplikácie a iteratívne vyriešiť všetky konflikty v dátach.

Z dôvodu nutnosti riešenia konfliktov v dátach bolo teda zvolené proprietárne riešenie vo forme samostatnej aplikácie, hoci išlo o jednorázové riešenie. Po importe dát bol predpoklad, že klient a tím ľudí pracujúcich so systémom EPICHIR už nebudú ďalej udržiavať dáta v dokumentovom formáte, ale priamo v systéme EPICHIR.

3.3.3 Technický popis

Pre prístup a manipuláciu s dátami neboli použité klasické SQL dopyty, ale bol zvolený ORM (z angl. Object-relational Mapping) prístup. ORM nám dovoľuje namapovať objekty, v tomto prípade jednotlivé entity, ku relačnej databáze, v tomto prípade ku MySQL databáze. Aplikácia využíva nakopírované dátové triedy z aplikácie EPICHIR, ktoré reprezentujú entity perzistované v databáze a ich príslušné DAO objekty, ktoré umožňujú manipuláciu s danými entitami cez metódy daného programovacieho jazyka, v tomto prípade JAVA. Z dôvodu použitia už existujúcich DAO objektov a dátových tried bola zabezpečená rovnaká konzistencia dát ako pri ukladaní cez systém EPICHIR, so všetkými integritnými obmedzeniami. ORM bolo zabezpečené cez Hibernate framework, ktorý je obsiahnutý priamo v Spring Boot frameworku.

3.3.3.1 Konfigurácia Spring Boot

Pre správne fungovanie frameworku bolo potrebné nakonfigurovať DataSource (prístup k databáze) a Hibernate. Pre konfiguráciu Spring Boot-u je využívaný súbor application.properties, ktorý si framework načíta automaticky pri štarte aplikácie. V ňom bolo potrebné nakonfigurovať štyri property, a to adresu URL kde sa nachádza databáza, meno a heslo pre prihlásenie k databáze a meno triedy JDBC ovládača.[6]

```
spring.datasource.url=jdbc:mysql://localhost:3306/epichir
spring.datasource.driver-class-name=com.mysql.jdbc.Driver
spring.datasource.username=root
spring.datasource.password=heslo
```

DAO objekty z aplikácie EPICHIR využívajú pre manipuláciu s dátami SessionFactory interface. Je teda potrebné vyrobiť správnu implementáciu, v tomto prípade postačí LocalSessionFactoryBean, ktorý pochádza z balíka Spring Boot. Tomuto objektu stačí nastaviť cez príslušné metódy správny DataSource a balík kde sa nachádzajú dátové entity.

3.3.4 Postup pri importe dát

Pred každým importom dát bol vytvorený obraz aktuálnej databázy z produkčného prostredia. Pomocou tohto obrazu bola vytvorená databáza na lokálnej pracovnej stanici a prekontrolovaná jej funkčnosť pomocou aplikácie EPICHIR. Po úspešnej kontrole dát sa spustil import dát pre istú kategóriu kariet pacientov. Import dát na začiatku načítal dáta z Excel dokumentu do pamäte, vytvorila sa mapa pre každého pacienta s novými hodnotami. V prípade, že sa hodnoty nepodarilo úspešne prečítať poprípade nastala iná chyba, aplikácia to zapísala do logu, čo mohlo byť viditeľné okamžite za behu importu. Tým pádom bolo možné iteratívne opraviť prípadné chyby v ap-

3. POŽIADAVKY KLIENTA A ICH REALIZÁCIA

likácii alebo dátach (napríklad zlý oddeľovač desatinných miest, nesprávny formát dátumu), prepísať databázu z obrazu a spustiť import znovu.

Aplikácia potom pre každú mapu načítaných dát v pamäti sa pokúsila nájsť pacienta priamo v databáze. Po jeho nájdení boli načítané dátové entity týkajúce sa aktuálne importovanej kategórie dát. Keďže dokument s dátami na import neobsahoval komentáre, ale len štruktúrované dátové hodnoty, komentáre obsiahnuté v načítaných dátových entitách boli uložené do pamäte. Následne sa do vytvorili nové entity a naplnili sa hodnotami z mapy pacienta. K nim sa pridali komentáre zo starých entít. Staré entity boli označené ako historické a následne perzistované do databáze. Nové entity boli taktiež perzistované do databáze.

3.3.5 Import jednotlivých kategórií kariet pacienta

Excel tabuľka v čase importu disponovala 283 záznamami, každý záznam obsahoval 58 hodnôt / stĺpcov. Jednotlivé časti dát boli importované po menších logických celkoch, podľa aktuálnej priority klienta. Importovacia aplikácia umožňuje nakonfigurovať, ktoré celky sa majú v aktuálnom behu programu importovať a ktoré nie.

Obraz databáze bol vytvorený v neskorých nočných hodinách na produkčnom databázovom serveri, uložený na lokálnu pracovnú stanicu, kde prebehol import dát menšieho logického celku, následne ich kontrola, vytvoril sa nový obraz databáze z lokálnej pracovnej stanice a ten bol nahraný na produkčný databázový server. Pred každou zmenou v dátach boli vyrobené zálohy databázových obrazov, aby bolo možné sa k nim kedykoľvek vrátiť.

3.3.5.1 Import základných položiek karty pacienta a kategórie anamnéza

Každý pacient má vo svojej karte zapísané základné údaje ako ošetrojúceho lekára, rodné číslo, pohlavie, adresu, dátum narodenia, vek pri začiatku epilepsie. Z týchto údajov boli z tabuľky importované len údaje veku pri začiatku epilepsie. V kategórii anamnéza boli importované pravdivostné údaje (čiže hodnoty áno alebo nie) o epilepsii v rodine pacienta, prenatálnych rizikách, febrilných kŕčoch, úraze CNS (centrálny nervový systém), zápale CNS a infantilných spazmách. Taktiež bola v rámci tejto kategórie importovaná číselníková hodnota epileptického syndrómu.

Po importe dát do lokálnej databáze boli dáta skontrolované cez aplikáciu EPICHIR, následne bol vytvorený obraz databázy, ktorý bol nahraný na vzdialený produkčný databázový server a následne boli dáta opäť skontrolované na produkčnom prostredí, vrátane správneho behu aplikácie EPICHIR. Klient bol vyzvaný s prebehnutým importom a bol vyzvaný na kontrolu dát.

3.3.5.2 Import kategórie histológia

Import histológie bol uskutočnený po oprave položiek v kategórií Histológia. Kategória histológia obsahuje len 2 položky, a to Histopatológiu a Klasifikáciu FCD. Klasifikácia FCD sa zobrazuje a vyplňa len pri zvolenej histopatológii MCD.

V tabuľke Excel boli dáta v dvoch stĺpcoch, Histopatológii a Klasifikácii FCD. V prípade, že Klasifikácia FCD bola rozličná od číselníkovej hodnoty 0 (bez FCD), bolo potrebné vyrobiť 2 záznamy v databáze. Prvý s číselníkovou hodnotou histopatológie 1 (MCD) a klasifikáciou FCD s číselníkovou hodnotou zo stĺpca v tabuľke Excel. Druhý záznam mal mať histopatológiu podľa hodnoty v stĺpci tabuľky Excel a hodnotu klasifikácie FCD nastavenú na 0 (bez FCD). V prípade, že histopatológia bola v tabuľke Excel na hodnote 1 (MCD), vyrobil sa len jeden záznam s príslušnou Klasifikáciou FCD podľa tabuľky. Na obrázku 34 je možné vidieť zvýraznené príklady záznamov, kde bolo potrebné vyrobiť 2 záznamy. Pri dokončení importu boli ku novo vytvoreným dátam v databáze pridané komentáre zo zneplatnených záznamov. Dáta boli následne perzistované v databáze. Dáta boli následne skontrolované pomocou aplikácie EPICHR, minimálne pre každý typ hodnoty aspoň po jednom zázname. Následne bol nahraný nový obraz databáze s importovanými záznamami na produkčný databázový server. Import prebehol úspešne a klient bol vyzvaný na kontrolu.

3.3.5.3 Import kategórie komplikácie

Kategória Komplikácie obsahuje 7 položiek, Operácia zo dňa, Typ komplikácie, Tiaž komplikácie, Etiológia komplikácie, Funkčný dôsledok komplikácie, Včasná pooperačná záchvaty a Očakávaný neurologický deficit. Pre hodnoty spadajúce do kategórie komplikácie bola s klientom dohodnutá možnosť do bunky vložiť viacero hodnôt oddelených čiarkou. To znamenalo, že bolo vytvorených toľko záznamov v databáze u daného pacienta, koľko bol maximálny počet číselníkových hodnôt v ktorejkoľvek bunke pri danom zázname v prislúchajúcej kategórii komplikácie. Pri vytváraní každého takého záznamu pre daného pacienta sa brali hodnoty z každej bunky na indexe, ktorý sa zhodoval s poradovým číslom aktuálne vytváraného záznamu pre jedného pacienta. V prípade, že v ktorejkoľvek bunke bolo menej záznamov ako požadovaný index aktuálne vytváraného záznamu, bola importovaná posledná číselníková hodnota z danej bunky.

Na príklade prvého importovaného pacienta na obrázku 35 (riadok označený žltým obdĺžnikom) sú v stĺpci Etiológia komplikácie pri zázname pacienta v bunke tri číselníkové hodnoty oddelené čiarkou. Teda boli vytvorené tri záznamy kategórie Komplikácia pre daného pacienta. Všetky tri mali zhodnú hodnotu ANO pre Včasné pooperačné záchvaty, prvý záznam mal Pribeh komplikácie nastavený na číselníkovú hodnotu 1 teda Komplikácie resekcie,

3. POŽIADAVKY KLIENTA A ICH REALIZÁCIA

histopatologie	klasifikace FCD
Normální	bez FCD
MCD	FCD 1
Nádor	FCD 2A
HS	FCD 2B
hypothalamický hamartom	FCD 3A
TSC	FCD 3B
Encefalitida	FCD 3C
Vaskulární léze	FCD 3D
Jizva	nespecifikovaný typ MCD
FCD+HS	
Kombinovaná patologie	

histopatologie	klasifikace FCD
1	3
3	4
3	0
3	0
2	5
1	3
3	0
2	5
2	5
3	4
1	3
3	4
2	0
3	0
3	4
3	4
3	0
3	4

Obr. 34: Zobrazenie záznamov, kde bolo potrebné vytvoriť 2 nové záznamy histopatológií u daného pacienta

druhý a tretí mali hodnotu nastavenú na 2 – Komplikácie invazívneho monitorovania. Prehľadnejšia interpretácia importu tohto pacienta je v tabuľke 35.

Pri dokončení importu boli ku novo vytvoreným dátam v databáze pridané komentáre zo zneplatnených záznamov predošlých komplikácií. Dáta boli následne perzistované v databáze. Dáta boli následne skontrolované pomocou aplikácie EPICHR, minimálne pre každý typ hodnoty aspoň po jednom zázname. Následne bol nahraný nový obraz databáze s importovanými záznamami na produkčný databázový server. Import prebehol úspešne a klient bol vyzvaný na kontrolu.

3.3.5.4 Import kategórie Outcome – záchvaty

Kategória Outcome – záchvaty bola premenovaná z predošlej kategórie Outcome. Kategória môže obsahovať vždy 4 záznamy ku každej existujúcej operácii

3.3. Import aktuálnych dát

Tabuľka 35: Príklad importu záznamu komplikácie jedného pacienta do reálnych 3 záznamov v databáze

Včasná pooperačná záchvaty	Priebeh komplikácie	Typ komplikácie	Etiológia komplikácie	Funkčný následok komplikácie	Očakávaný neurologický deficit
1	1	1	2	1	2
1	2	1	3	2	2
1	2	1	6	2	2

Kód	Číslo pacienta	datum narodení	pohlaví	časné pooperačné záchvaty	komplikácie	typ komplikácie	etiologie komplikácie	funkční následek komplikácie	očekávaný neurologický deficit
0				ne	Bez komplikácií	Nezvoleno	Nezvoleno	Nezvoleno	Bez deficitu
1			žena	ano	Komplikácie resekcie	Minor	Neplánovaný rozsah resekcie	Afázie	Afázie
2			muž	NA	Komplikácie invazívneho monitorování	Major	Edém	Hemiparéza	Hemiparéza
3							Zánět	Paréza hlavových nervů	Paréza hlavových nervů
4							Hydrocefalus	Porucha zorného pole	Porucha zorného pole
5		dd.mm.yyyy					Ischémie	Mnestická porucha	Mnestická porucha
6							Jiné	Jiné	Jiné
7							Krváčení	Smrt	
8									
9									
10									

PACIENT	Číslo pacienta	datum narodení	pohlaví	časné pooperačné záchvaty	průběh komplikácie	typ komplikácie	etiologie komplikácie	funkční následek komplikácie	očekávaný neurologický deficit
x	x	x	1	1	1, 2	1	2, 3, 6	1, 2	2
x	x	x	1	1	0	0	0	0	0
x	x	x	1	1	0	0	0	0	0
x	x	x	2	0	0	0	0	0	0
x	x	x	2	0	0	0	0	0	0
x	x	x	1	0	0	0	0	0	0
x	x	x	1	0	0	0	0	0	0
x	x	x	2	0	1	1	5	2	0
x	x	x	2	0	0	0	0	0	0
x	x	x	1	0	0	0	0	0	0
x	x	x	1	0	0	0	0	0	0
x	x	x	1	0	0	0	0	0	2, 4
x	x	x	1	1	1	1	2, 3	2, 6	0
x	x	x	2	0	0	0	0	0	0
x	x	x	2	1	0	0	0	0	4

Obr. 35: Zobrazenie záznamov na import do kategórie Komplikácie

u daného pacienta, a to po prvom, druhom, piatom a desiatom roku od operácie. Každý záznam obsahuje položku Follow-up a zaškrťavaciu položku Lost to follow-up. Excel tabuľka obsahuje stĺpce s číselníkovými hodnotami pre položky Follow-up pre každý záznam operácie daného pacienta. V tomto prípade, každá bunka v stĺpoch Follow-up pre jednotlivé roky od operácie mohla obsahovať len jednu hodnotu. Po dohode s klientom boli importované hodnoty vždy len ku poslednej operácii, z dôvodu, že klient sleduje zmenu počtu záchvatov u pacienta vzhľadom ku poslednej uskutočnenej operácii a snaží sa ich u pacienta redukovat' na minimum. V dokumente Excel boli pri každom pacientovi vyplnené záznamy Follow-up taktiež vždy len pre poslednú operáciu. Predošlé záznamy boli historizované a komentáre prislúchajúce ku záznamom v databáze EPICHR pre poslednú operáciu u pacienta boli uložené do pamäte. Následne boli tieto komentáre pridané ku novo vytvoreným záznamom a jednotlivé záznamy boli perzistované do databáze. Podobne ako v ostatných prípadoch, obraz databáze bol nahraný na produkčný server a klient bol o tom informovaný a vyzvaný na kontrolu dát.

3.3.5.5 Import kategórie Outcome – AED

Kým pri kategórii Outcome - záchvaty sú ku každej operácii maximálne 4 záznamy, v kategórii Outcome – AED je to vždy maximálne jeden záznam ku jednej konkrétnej operácii daného pacienta. Kategória Outcome - AED informuje o vysadení alebo nevysadení AED a prislúchajúceho dátumu, kedy tak nastalo. V dokumente používanom pri importe dát sa nachádza ku kategórii Outcome – AED len jedna položka na import, a to číselníková hodnota reprezentujúca, či boli AED vysadené po danej operácii. Aj v tomto prípade boli dáta importované vždy len pre danú poslednú operáciu pre každého pacienta, pretože číselníkové hodnoty pre predošlé operácie pacientov neboli k dispozícii a neboli vyplnené v dokumente. Položka dátum vysadenia AED nebola v dokumente k dispozícii a bola automaticky nastavená vždy mesiac po danej operácii u pacienta. Po úspešnom importe dát bol nahraný nový obraz databáze na produkčný server a klient bol vyzvaný na kontrolu záznamov.

3.3.5.6 Import kategórií záchvaty, neurologické nálezy a invazívny test ECoG

Dáta pre kategórie záchvaty, neurologické nálezy a invazívne testy ECoG boli importované len pre záznamy v dokumente Excel, ktoré sa pre daného pacienta týkali vždy len poslednej operácie. V kategórii záchvaty boli importované nasledujúce číselníkové položky z dokumentu: Frekvencia záchvatov, Status epilepticus (jeden záchvat trvajúci dlhšie ako 5 minút alebo viacero po sebe nasledujúcich záchvatov u pacienta [7] a SGTCS – Sekundárne generalizované záchvaty. Klasifikácia záchvatov nebola importovaná z dôvodu zavedenia novej klasifikácie ILAE z roku 2017, a teda táto položka musela byť u pacientov klasifikovaná manuálne. Všetky tri importované hodnoty boli vložené už do existujúceho záznamu v databáze EPICHR, pretože karty pacientov v systéme obsahovali vždy po jednom zázname záchvatov. Podobne je aj v kategórii neurologické nálezy u pacientov vždy jeden záznam. Z tejto kategórie bola importovaná len číselníková hodnota abnormálny neurologický nález, ostatné hodnoty neboli v dokumente prítomné. V kategórii invazívne testy – ECoG [8] boli importované dve položky. Obidve boli číselníkového typu, prvá indikovala, či bol vykonaný invazívny test ECoG a druhá hodnota indikovala či a ako bol vykonaný ECoG test po resekcii. Obidve hodnoty boli importované do posledného záznamu z kategórie Invazívny test ECoG u daného pacienta. Po importe boli dáta náhodne skontrolované u viacerých pacientov priamo v aplikácii EPICHR a následne bol nahraný obraz databázy po importe na produkčný server. Klient bol potom vyzvaný na kontrolu importovaných dát.

3.3.6 Zhodnotenie importu dát

Všetky kategórie dát pacientov, ktoré klient vyžadoval nainportovať do databáze informačného systému EPICHR sa úspešne podarili, klient vždy po-

3.3. Import aktuálních dat

tvrdzoval ich správnosť po vykonaní importu jednotlivých kategórií.

Záver

Cieľom práce bolo dokončiť rozpracované časti, implementovať nové celky a revidovať informačný systém EPICHIR.

Za rozpracovanú časť systému sa dá považovať to, že systém nebol aktívne používaný z dôvodu inkonzistencie dát. Lekári zabúdali dáta do systému nahrávať a preferovali ich uchovávanie v tabuľkovom dokumente. Lekárom v systéme chýbali nové klasifikácie a položky, ktoré si takto radšej uchovávali v tabuľkovom formáte. Taktiež hodnoty číselníkových položiek boli neaktuálne.

V prvej časti práce sa podarilo aktualizovať tieto neaktuálne položky a číselníky. Taktiež sa vyhovel klientovi a vykonali sa zmeny nad rámec zadania, najmä pridávanie nových položiek do systému a prepracovanie stávajúcich. Tento proces bol pomerne časovo náročný nielen na implementáciu, ale aj komunikáciu s klientom, aby bolo možné presne špecifikovať zmeny, ktoré klient očakáva.

Následne vznikla najviac časovo náročná požiadavka klienta, a to vykonať import dát z tabuľkového dokumentu do databáze systému. Pre tento účel bola vyvinutá samostatná aplikácia, aby sa predišlo konfliktom a mohli byť aplikované pravidlá pri nahrávaní aktuálnych dát. Import dát prebiehal postupne, po každej časti sa dáta kontrolovali a komunikoval sa nasledujúci postup s klientom. V súčasnom stave sú dáta v databáze aktuálne a zrevizované, názvy tabuliek a ich stĺpcov odpovedajú ich vizuálnym položkám a premenným v kóde. Týmto sa dosiahla konzistencia celého kódu.

Zmeny, ktoré klient vyžadoval vrátane importu dát boli úspešne vykonané, manuálne otestované a overené klientom. Z dôvodu časovej náročnosti celého procesu, nebolo možné dokončiť niektoré body zo zadania, napríklad vylepšiť reportovací modul, či vytvoriť nový vyhľadávací modul, keďže úspešne vykonaným zmenám klient priradil vyššiu prioritu.

Literatúra

- [1] prof. MUDr. Petr Marusič Ph.D; doc. MUDr. Hana Ošlejšková Ph.D; prof. MUDr. Milan Brázdil Ph.D; aj.: Nové klasifikace epileptických záchvatů a epilepsií ILAE 2017. *Neurologie pro praxi*, , č. 1, 2018: s. 32–36. Dostupné z: <https://www.neurologiepropraxi.cz/pdfs/neu/2018/01/07.pdf>
- [2] Dušek, J.: *Specializovaný informační systém pro evidenci pacientů zařazených do epilepto-chirurgického programu - redesign, reimplementace a optimalizace*. Bakalářská práce, Fakulta informačních technologií, ČVUT, 2014.
- [3] Johnson, R.; Hoeller, J.; Donald, K.; aj.: *Spring Framework Reference Documentation*. Pivotal Software, Inc., 2016. Dostupné z: <https://docs.spring.io/spring/docs/4.2.x/spring-framework-reference/html/mvc.html>
- [4] Beck, K.; Beedle, M.; Van Bennekum, A.; aj.: Manifesto for agile software development. 2001.
- [5] Oracle Corporation: *MySQL 8.0 Reference Manual*. 2019. Dostupné z: <https://dev.mysql.com/doc/refman/8.0/en/>
- [6] Webb, P.; Syer, D.; Long, J.; aj.: *Spring boot reference guide*. Pivotal Software, Inc., 2018. Dostupné z: <https://docs.spring.io/spring-boot/docs/current/reference/html/boot-features-sql.html>
- [7] Trinka, E.; Höfler, J.; Zerbs, A.: Causes of status epilepticus. *Epilepsia*, ročník 53, 2012: s. 127–138.
- [8] Ferraro, J. A.; Ruth, R. A.: Clinical electrocochleography. *Hear J*, ročník 38, 1985: s. 51–5.

Zoznam použitých skratiek a pojmov

A.1 Technické skratky

EPICHIR - informačný systém pre uchovávanie dát o pacientoch s epilepsiou

Klient - Oddelenie detskej neurológie nemocnice Motol

Spring Boot - framework postavený na Jave

DAO - Data Access Object - Objekt cez ktorý sa pristupuje k dátam

DataSource - v Jave väčšinou označuje objekt, cez ktorý je možné pristupovať k relačnej databáze

ORM - Object-relation mapping - Mapovanie dát medzi relačnou databázou a objektovým programovacím jazykom

Hibernate - ORM nástroj pre programovací jazyk Java

Property - pojem využívaný v JAVE pre konfiguráciu, každej property prináleží aj jej hodnota, väčšinou zapisovaná ako nazov.property=hodnota

JDBC ovládač/adaptér - Java knižnica určená na prístup ku konkrétnemu typu databáze

JNDI - mechanizmus v Jave pomocou ktorého sa dá pristupovať k prostriedkom pomocou ich špecifického mena

Tomcat - Aplikačný server pre Java aplikácie

View v MVC - Prezentačná vrstva v návrhovom vzore MVC

Controller v MVC - Logická vrstva spracúvajúca dáta v návrhovom vzore MVC

JSTL View - Java Standard Tag Library - súbor prvkov pre tvorbu vizuálnych formulárov

Rest API - Representational State Transfer Application Programming Interface - aplikačné rozhranie ku ktorému sa väčšinou pristupuje pomocou HTTP protokolu

A.2 Medicínske skratky a pojmy

AED - Anti epileptic drugs - lieky zabraňujúce vznikom epileptických záchvatov

ECoG - Electrocochleography - Spôsob zaznamenávania stimulačného potenciálu vnútorného ucha a sluchového nervu [8]

ILAE - International League Against Epilepsy - Medzinárodná liga proti epilepsii

SGTCS - Sekundárne generalizované záchvaty

CNS - centrálny nervový systém

Obsah priloženej SD karty

readme.txt	stručný popis obsahu
document.....	adresár s týmto dokumentom vo formáte L ^A T _E X a pdf
documentationEpichirApp	dokumentácia k nasadeniu EPICHIR aplikácie z predošlých prác [2]
databasescripts	adresár s vytvorenými SQL skriptami, ktoré boli vykonávané nad databázou
└─ epichir.sql	skript pre vytvorenie prázdnej epichir db
sourcecodes	
└─ epichir	aktuálny zdrojový kód aplikácie epichir
└─ importer	aktuálny zdrojový kód aplikácie na import dát
model.mwb	model Epichir DB - MySQL Workbench
dbdiagram.pdf	export modelu Epichir DB
notes	adresár s poznámkami vedenými k projektu

Technická dokumentácia ku aplikácii na import dát z Excel súboru do databáze

C.1 Štruktúra kódu

Aplikácia je vytvorená ako maven projekt. Pomocou maven-u je možné spustiť kompiláciu zdrojového kódu a vytvoriť finálny archív s binárnymi súbormi. Aplikácia pozostáva z dvoch hlavných balíkov:

- `cz.epichir.importer` – balík s funkcionalitou pre čítanie dokumentu excel, pre vytváranie mapy údajov pre jednotlivých pacientov a pre samotnú aktualizáciu dát
- `cz.cvut.fel.isarg.epichir` – balík nakopírovaný z aplikácie EPICHR, ktorý obsahuje entity a DAO služby

C.2 Funkcionalita

Aplikácia ma jednu hlavnú funkcionalitu, a to aktualizáciu dát v pripojenej databáze MySQL dátami vyextrahovanými z dokumentu Excel.

C.2.1 Konfigurácia aplikácie

Aplikáciu je možné nastaviť pomocou Java properties. Súbor jednotlivých nastavení je viditeľný v tabuľke C1. Java properties je možné nastaviť argumentami pri štarte programu alebo externým properties súborom (cesta je definovateľná cez property **importer.properties**), ktorý má najvyššiu prioritu.

C. TECHNICKÁ DOKUMENTÁCIA KU APLIKÁCII NA IMPORT DÁT Z EXCEL SÚBORU DO DATABÁZE

Tabuľka C1: Konfigurovateľné Java properties pre aplikáciu Importer

Java property	Prednastavená hodnota	Popis
importer.properties		Cesta k súboru s properties
logging.path	logs	Cesta, kde sa majú uložiť logy z begu programu
spring.datasource.url	jdbc:mysql://localhost:3306/epichir	Pripojovacia URL k databáze EPICHIR
spring.datasource.username	root	Meno používateľa na prístup k databáze EPICHIR
spring.datasource.password		Heslo používateľa na prístup k databáze EPICHIR
server.port	8090	Port na ktorom bude spustená aplikácia
import.file		Cesta k dokumentu Excel, kde sa nachádzajú dáta pre import
epichir.histology	false	Označuje či sa má vykonať import dát z kategórie Histológia (true/false)
epichir.complication	false	Označuje či sa má vykonať import dát z kategórie Komplikácie (true/false)
epichir.seizures	false	Označuje či sa má vykonať import dát z kategórie Záchvaty (true/false)
epichir.neurological	false	Označuje či sa má vykonať import dát z kategórie Neurologické nálezy (true/false)
epichir.diagnostic	false	Označuje či sa má vykonať import dát z kategórie Diagnostické testy (true/false)
epichir.invasiveEcog	false	Označuje či sa má vykonať import dát z kategórie Invazívne testy (true/false)
epichir.outcomeAed	false	Označuje či sa má vykonať import dát z kategórie Outcome - AED (true/false)
epichir.outcome	false	Označuje či sa má vykonať import dát z kategórie Outcome (true/false)

C.3. Závislosti aplikácie (dependencies)

Tabuľka C2: Závislosti aplikácie importer na vybraných maven balíkoch

Balík	Popis
org.springframework.boot:spring-boot-starter-parent	Rodičovská závislosť na štartovacom balíku pre aplikácie bežiacie na Spring Boot
mysql:mysql-connector-java	JDBC kontektor slúžiaci pre Spring Boot ako prostriedok na pripojenie k databáze
org.apache.poi:poi a poi-ooxml	Knižnice od Apache Software Foundation ktoré uľahčujú prácu s Excel dokumentami
joda-time:joda-time	Knižnica uľahčujúca prácu s dátumovými a časovými objektami

C.2.2 Skompilovanie a spustenie aplikácie

Aplikáciu je možné skompilovať cez príkaz

```
mvn package
```

Následne je možné spustiť skompilovaný NAZOV.jar súbor nachádzajúci sa v zložke `target` príkazom

```
java -jar NAZOV.jar --property=value
```

kde `property=value` je možné zreťaziť a poskytnúť všetky možné nastavenia programu popísané v sekcii C.2.1, napríklad

```
java -jar importer.jar
--importer.properties=/Users/user/importer/settings.properties
--import.file=/Users/user/importer/epichir.xls
--spring.datasource.url=jdbc:mysql://localhost:8889/epichir
--spring.datasource.password=password
```

.

C.3 Závislosti aplikácie (dependencies)

Závislosti aplikácie sú popísané v tabuľke C2.