



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA PODNIKATELSKÁ

FACULTY OF BUSINESS AND MANAGEMENT

## ÚSTAV INFORMATIKY

INSTITUTE OF INFORMATICS

# NÁVRH MOBILNÍ APLIKACE PRO MEDICÍNSKOU SFÉRU

DESIGN OF A MOBILE APPLICATION FOR MEDICAL SPHERE

## BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

## AUTOR PRÁCE

AUTHOR

Matej Puček

## VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Petr Dydowicz, Ph.D.

BRNO 2021

# Zadání bakalářské práce

Ústav:	Ústav informatiky
Student:	<b>Matej Puček</b>
Studijní program:	Systémové inženýrství a informatika
Studijní obor:	Manažerská informatika
Vedoucí práce:	<b>Ing. Petr Dydowicz, Ph.D.</b>
Akademický rok:	2020/21

Ředitel ústavu Vám v souladu se zákonem č. 111/1998 Sb., o vysokých školách ve znění pozdějších předpisů a se Studijním a zkušebním řádem VUT v Brně zadává bakalářskou práci s názvem:

## Návrh mobilní aplikace pro medicínskou sféru

### Charakteristika problematiky úkolu:

Úvod  
Vymezení problému a cíle práce  
Teoretická východiska práce  
Analýza problému a současné situace  
Vlastní návrh řešení, přínos práce  
Závěr  
Seznam použité literatury

### Cíle, kterých má být dosaženo:

Cílem práce je navrhnout aplikaci pro mobilní zařízení, která usnadní a urychlí komunikaci mezi lékaři a jejich pacienty. Její hlavní součástí je kalendář rezervací termínů umožňující vlastnoruční zadávání termínů do kalendářů lékařů. Samozřejmostí aplikace je taky schopnost pracovat s výměnnými lístky, posílat notifikace, selfmonitoringu pacientů a zasílat zprávy lékařům pro dotazy a zpětnou vazbu. Práce neřeší kód a zprovoznění aplikace.

### Základní literární prameny:

GARGENTA, M. Learning Android. Sebastopol, Calif.: O'Reilly, 2011. 245 p. ISBN 14-493-9050-1.  
LEE, W. M. Beginning Android application development. Indianapolis, IN: Wiley Pub., 2011. 428 s. ISBN 978-111-8087-800.  
MARTIŠEK, D. Algoritmizace a programování v Delphi. Brno: Littera, 2007. 230 s. ISBN 978-8-85763-37-9.

UJBÁNYAI, M. Programujeme pro Android. Praha: Grada, 2012. 187 s. ISBN 978-80-247-3995-3.

VELTE, A., T. VELTE a R. ELSENPETER. Cloud Computing: praktický průvodce. Brno: Computer Press, 2011. 344 s. ISBN 978-80-251-3333-0.

Termín odevzdání bakalářské práce je stanoven časovým plánem akademického roku 2020/21

V Brně dne 28.2.2021

L. S.

---

Mgr. Veronika Novotná, Ph.D.  
ředitel

---

doc. Ing. Vojtěch Bartoš, Ph.D.  
děkan

## **Abstrakt**

Bakalárska práca je venovaná návrhu mobilnej aplikácie pre medicínsku sféru, ktorá by uľahčila komunikáciu medzi lekármi a ich pacientami. Práca obsahuje analýzu trhu mobilných zariadení, trhu mobilných aplikácií na prítomnosť podobných aplikácií a návrh databáze a funkcií aplikácie. Záver práce sa zameriava na zhodnotenie výsledného návrhu riešenia pomocou vybraných metód.

## **Kľúčové slová**

mobilné zariadenie, operačný systém, mobilná aplikácia, Android, iOS, dátové modelovanie, funkčné modelovanie

## **Abstract**

The bachelor thesis is devoted to the design of a mobile application for the medical sphere, which would facilitate communication between doctors and their patients. The thesis contains an analysis of the mobile device market, the mobile application market for the presence of similar applications and the design of a database and application functions. The conclusion of the thesis focuses on the evaluation of the final design of the solution using selected methods.

## **Key words**

mobile device, operating system, mobile application, Android, iOS, data modelling, functional modelling

### **Bibliografická citácia**

PUČEK, Matej. *Návrh mobilní aplikace pro medicínskou sféru* [online]. Brno, 2021 [cit. 2021-05-06]. Dostupné z: <https://www.vutbr.cz/studenti/zav-prace/detail/135319>.  
Bakalářská práce. Vysoké učení technické v Brně, Fakulta podnikatelská, Ústav informatiky. Vedoucí práce Petr Dydowicz.

## **Čestné prehlásenie**

Prehlasujem, že predložená bakalárska práca je pôvodná a spracoval som ju samostatne. Prehlasujem, že citácia použitých prameňov je úplná, že som v práci neporušil autorské práva (v zmysle Zákona č. 121/2000 Sb., o práve autorskom a o právach súvisiacich s právom autorským).

V Brne dňa 15. mája 2021

.....

Matej Puček

## **Pod'akovanie**

Týmto by som chcel poďakovať svojmu vedúcemu práce Ing. Petrovi Dydowiczovi, Ph.D. za čas, ochotu a spoluprácu pri vypracovávaní tejto bakalárskej práce a oponentke Ing. Dite Kindlovej za jej čas pri hodnotení práce. Ďalej by som rád poďakoval mojej sestre Katke, ktorá mi ako študentka ošetrovateľstva vedela podať cenné rady a potrebné materiály v oblasti medicíny. V neposlednom rade by som chcel predovšetkým poďakovať mojej najbližšej rodine a kamarátom, ktorí pri mne vždy stáli a podporovali ma.

# OBSAH

ÚVOD.....	11
VYMEDZENIE PROBLÉMU A CIELE PRÁCE.....	12
<b>1 TEORETICKÉ VÝCHODISKÁ.....</b>	<b>13</b>
1.1 Operačný systém .....	13
1.1.1 Funkcie OS .....	13
1.1.2 Výhody a nevýhody OS .....	14
1.1.3 Mobilný OS.....	14
1.2 Aplikácie .....	16
1.2.1 Typy aplikácií .....	16
1.2.2 Mobilné vs. webové aplikácie .....	17
1.2.3 Licencovanie aplikácií .....	17
1.3 Digitálne distribučné platformy .....	18
1.3.1 Obchod Google Play .....	18
1.3.2 iOS App Store.....	19
1.4 Dátové modelovanie.....	20
1.4.1 Atribút.....	20
1.4.2 Relačný dátový model .....	21
1.4.3 Entito-relačný model.....	24
1.4.4 SQL Server .....	25
1.5 Funkčné modelovanie .....	27
1.5.1 Diagram toku dát (DFD).....	27
1.5.2 Vývojový diagram .....	28
1.6 Metódy zhodnotenia výsledku práce.....	30
1.6.1 SWOT analýza .....	30
1.6.2 Cenová politika.....	30
1.7 Medicínske pojmy v selfmonitoringu .....	31
1.7.1 Meranie krvného tlaku a pulzu .....	31
1.7.2 Meranie telesnej teploty.....	32
1.7.3 Meranie a váženie .....	33
1.7.4 Glukóza v krvi .....	33
<b>2 ANALÝZA SÚČASNÉHO STAVU.....</b>	<b>34</b>



2.1	Analýza trhu mobilných zariadení .....	34
2.2	Analýza trhu operačných systémov .....	37
2.3	Digitálne distribučné platformy .....	38
2.4	Podobné aplikácie .....	41
2.4.1	Organizéry s kalendárom pre lekárov .....	41
2.4.2	Aplikácie typu Find-A-Doctor .....	42
2.4.3	Aplikácie zostavené na mieru alebo osobný virtuálny lekár .....	42
2.4.4	Rezervačné systémy cez web .....	43
2.4.5	Univerzálne rezervačné systémy .....	44
2.5	Zhrnutie analýz.....	45
<b>3</b>	<b>VLASTNÉ NÁVRHY RIEŠENIA .....</b>	<b>47</b>
3.1	Dátový model aplikácie.....	47
3.1.1	Entito-relačný model.....	47
3.1.2	Tabuľky dátového modelu .....	49
3.2	Funkcie aplikácie .....	60
3.2.1	Registrácia a prihlásenie používateľov .....	60
3.2.2	Profily používateľov .....	62
3.2.3	Členovia rodiny.....	63
3.2.4	Notifikácie .....	64
3.2.5	Kalendár rezervácií termínov.....	67
3.2.6	Evidencia neprítomnosti .....	70
3.2.7	Výmenné lístky .....	70
3.2.8	Predpis liekov .....	71
3.2.9	Selfmonitoring .....	71
3.2.10	Šablóny rezervácií.....	73
3.2.11	Chatovací modul .....	73
3.2.12	Prepojenie s konkurenčnými aplikáciami .....	74
3.2.13	Fungovanie aplikácie v offline režime.....	75
3.2.14	Diagramové zobrazenie vybraných procesov .....	75
3.3	Zhodnotenie výsledného návrhu riešenia.....	81
3.3.1	SWOT analýza .....	81
3.3.2	Ekonomické zhodnotenie nákladov .....	83

3.3.3	Cenová politika .....	84
3.4	Prínosy práce .....	85
<b>ZÁVER</b>	.....	<b>87</b>
<b>ZOZNAM POUŽITÝCH ZDROJOV</b>	.....	<b>88</b>
<b>ZOZNAM POUŽITÝCH OBRÁZKOV</b>	.....	<b>92</b>
<b>ZOZNAM POUŽITÝCH TABULIEK</b>	.....	<b>94</b>
<b>ZOZNAM POUŽITÝCH GRAFOV</b>	.....	<b>95</b>
<b>ZOZNAM POUŽITÝCH SKRATIEK A SYMBOLOV</b>	.....	<b>96</b>

## ÚVOD

Návšteva lekára nie je v dnešnej dobe cudzia žiadnemu človekovi. Lekárov navštevujeme kvôli preventívnym prehliadkam, odberu krvi, nutnosti vyšetrenia na podozrenie z choroby, atď. Aj keď je možné lekára navštíviť zvyčajne v ľubovoľný čas jeho pracovnej doby, takéto návštevy sa však nezaobídu bez nutnosti čakania, dokým na nás príde rad. V horšom prípade môžeme na rad čakať aj dlhšie ako hodinu. V hektickom svete ako je tento si ale mnoho ľudí niekedy nemôže dovoliť stráviť veľa času čakaním u lekára, takže je pre nich vybavenie termínu esenciálnou súčasťou prípravy na návštevu lekára. Takýto termín si je možné rezervovať nie len osobne, ale aj telefonicky, čo sa však môže stretnúť s radou ďalších problémov, ako napr. nemožnosť sa dovolať, či už z dôvodu nedostupnosti lekára alebo v prípade, že pacient nemal aktuálne číslo svojho lekára a nebolo ho možné ani dohľadať na internete. S takouto skutočnosťou som mal skúsenosť v minulosti aj ja, čo ma inšpirovalo pri návrhu témy na túto bakalársku prácu, ktorá sa bude zaoberať nie len problematikou rezervácie termínov u lekára ale aj inými užitočnými funkciami, ktoré v určitom ohľade nadväzujú na túto hlavnú problematiku práce.

V súčasnosti taktiež väčšina ľudí každodenne využíva smartphony na rôzne účely, či už ide o každodenné potreby, socializovanie alebo aj zábavu. Na všetky tieto účely sa zvyčajne nájde aplikácia, ktorá svojim používateľom umožňuje vykonávať procesy spojené s daným účelom. Mobilné aplikácie sa preto stali neodlúčiteľnou súčasťou používania mobilných zariadení. Aj na základe tejto skutočnosti som sa rozhodol svoje riešenie navrhnúť práve ako mobilnú aplikáciu, ktorá by uľahčila a zefektívnila komunikáciu medzi lekármi a ich pacientami.

Táto práca sa bude skladať z troch hlavných kapitol. V prvej kapitole sa budem zaoberať teóriou spojenou s daným návrhom tejto práce. V druhej kapitole budem analyzovať trh mobilných zariadení a trh mobilných aplikácií na prítomnosť podobných aplikácií. Nakoniec, v poslednej tretej kapitole sa budem venovať samotnému návrhu riešenia pomocou dátového a funkčného modelovania, pričom výsledky návrhu riešenia za pomoci vybraných metód zhodnotím.

## VYMEDZENIE PROBLÉMU A CIELE PRÁCE

Cieľom tejto práce je navrhnúť aplikáciu pre mobilné zariadenia, ktorá urýchli a uľahčí komunikáciu medzi lekármi a ich pacientami. Navrhnutá bude nie len pre využitie pacientami, ale taktiež aj lekármi, pre obe role však bude ponúkať rozdielne funkcie. Výhodou aplikácie bude jej schopnosť nahradiť osobné návštevy alebo telefonáty pre vybavenie rezervácie, upozorniť používateľa na dôležité udalosti, ako blížiaci sa termín návštevy u lekára alebo čas vziať si tabletky a taktiež bude možné aplikáciu využiť kdekoľvek a hocikedy, ak bude mať jej používateľ prístup k internetu, niektoré funkcie však nebudú prístup k internetu bezvýhradne vyžadovať.

Hlavnou súčasťou aplikácie bude rezervačný systém obsahujúci kalendár pre zadávanie rezervácie termínov. Pacient si takýmto spôsobom bude môcť o termín iba požiadať, lekár bude môcť do kalendára zadávať a schvaľovať prichádzajúce žiadosti o termín bez obmedzení. Obe role budú pred nadchádzajúcimi termínmi včas upozornené zvolenými spôsobmi. Zrušiť rezerváciu budú môcť obe role. Notifikácie sa budú týkať aj času pre užitie liekov, ktorý pacientovi do systému zadá lekár. Pacienti s vybranými chorobami budú mať možnosť využívať funkciu selfmonitoringu, kde si sebou namerané hodnoty zadajú do formulára a odošlú svojmu lekárovi na posúdenie, pričom na základe zadaných hodnôt a porovnania s odporúčanými hodnotami budú pacienti vopred uvedomení o svojom stave. Komunikácia medzi lekárom a pacientom bude prebiehať aj prostredníctvom chatovacieho modulu, ktorý okrem písomnej komunikácie bude ponúkať aj možnosť videohovoru. Užitočnou funkciou aplikácie bude aj schopnosť pracovať s výmennými lístkami, pričom lekár bude môcť v rámci aplikácie vypísať pacientovi výmenný lístok, vďaka ktorému bude pacient následne schopný požiadať o prístup k rezervačnému kalendáru kompetentného lekára. Pacient bude mať možnosť pomocou aplikácie interagovať iba so svojim všeobecným lekárom, zubárom a odbornými lekármi po uznaní výmenného lístka, ak ho k vyšetreniu vyžadujú. Pacienti taktiež budú môcť aplikáciu začať využívať iba po schválení žiadosti o registráciu od svojho všeobecného lekára, zubára alebo odborného lekára, ktorí aplikáciu budú využívať.

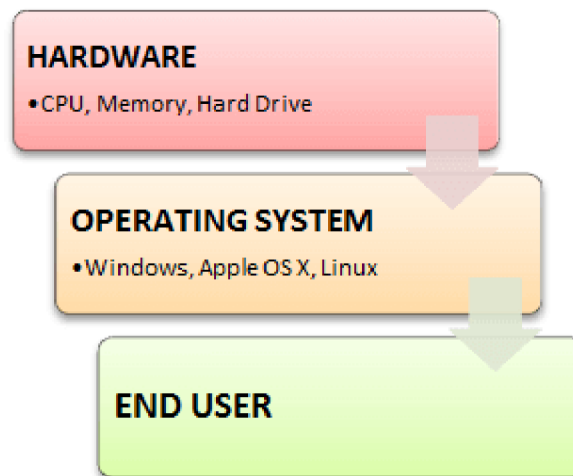
Práca rieši dátovú a funkčnú štruktúru vyššie opísanej mobilnej aplikácie. Súčasťou práce nie sú ukážky kódu aplikácie, spustenie prevádzky ani uvedenie na trh v praxi.

# 1 TEORETICKÉ VÝCHODISKÁ

## 1.1 Operačný systém

Operačný systém (OS) je softvér, ktorý sa chová ako rozhranie medzi hardvérovými komponentmi počítača a používateľom. Každý počítač musí mať aspoň jeden operačný systém, aby sa na ňom mohli spustiť programy, ktoré, aby mohli bežať a plniť svoje úlohy, potrebujú určité prostredie (ním je práve OS).

Operačný systém nám pomáha komunikovať s počítačom bez potreby ovládania jazyka, ktorému počítač rozumie. Nie je možné, aby používateľ používal počítač alebo mobilné zariadenie bez akéhokoľvek OS [1].



Obrázok č. 1: OS ako sprostredkovateľ  
(Zdroj: [1])

### 1.1.1 Funkcie OS

Medzi hlavné funkcie OS patria:

- Správa procesov – pomáha OS vytvárať a mazať procesy a poskytuje taktiež mechanizmy na synchronizáciu a komunikáciu medzi procesmi
- Správa pamäte – plní úlohu alokácie a uvoľnenia pamäťového priestoru pre programy, ktoré tieto zdroje potrebujú
- Správa súborov – spravuje všetky činnosti spojené so súbormi, ako sú napr. organizácia ukladania, vyhľadávanie, pomenovanie, zdieľanie a ochrana súborov
- Správa zariadení – sleduje všetky zariadenia pripojené k počítaču a vykonáva ich alokáciu a delokáciu z počítača. Tento modul sa taktiež nazýva I/O radič.

- Správa I/O systému – jedným z hlavných zámerov OS je skryť zvláštnosti vyššie uvedených hardvérových zariadení od používateľa
- Správa sekundárneho úložiska – systémy majú niekoľko úrovní úložiska, a to primárne, sekundárne úložisko a vyrovnávaciu pamäť
- Zabezpečenie – chráni údaje a informácie počítačového systému pred škodlivým softvérom a neautorizovaným prístupom
- Interpretácia príkazov – interpretuje príkazy dané systémovými prostriedkami
- Sieť – distribuovaný systém je skupina procesorov, ktoré nezdieľajú pamäť ani hardvérové zariadenia. Tieto procesory komunikujú prostredníctvom siete
- Sledovanie úloh – sledovanie času a zdrojov používaných rôznymi úlohami a používateľmi
- Správa komunikácie – koordinácia a pridelovanie prekladacích programov, tlmočníkov a iného softwarových zdrojov [1].

### 1.1.2 Výhody a nevýhody OS

Medzi výhody OS bezprostredne patria:

- Skrýva podrobnosti o hardvéri vytvorením abstrakcie
- Jednoduché použitie s GUI
- Ponúka prostredie, v ktorom môže používateľ spúšťať programy/aplikácie
- Zabezpečuje pohodlné používanie pre používateľa
- Je sprostredkovateľom medzi aplikáciami a hardvérovými komponentmi

Tak ako každý technický prostriedok, aj OS má svoje nedostatky. Medzi tie hlavné patria:

- Problém v OS môže mať za následok aj stratu všetkého obsahu, ktorý bol v systéme uložený
- OS softvér je pre malé organizácie pomerne drahý, čím sa zvyšuje ich finančná záťaž
- OS nikdy nie je úplne zabezpečený, k hrozbe môže dôjsť hocikedy [1]

### 1.1.3 Mobilný OS

Mobilné operačné systémy sú také OS, ktoré sú špeciálne určené na využitie na mobilných zariadeniach, tabletoch a nositeľných zariadeniach.

Medzi najrozšírenejšie mobilné OS patria Android a iOS [1].

### 1.1.3.1 Android

Android je mobilný OS založený na jadre Linuxu, ktorý bol doplnený o mnoho zmien a nových funkcií. Najnovšia stabilná verzia systému je Android 11 [2].

OS Android vyvinula spoločnosť Android, Inc. predtým, ako ho spoločnosť Google získala v roku 2005. Investori a analytici elektronického priemyslu spochybnili zámery spoločnosti Google vstúpiť na trh mobilných zariadení od tejto akvizície Androidu. Aj napriek tomu prvé komerčne dostupné zariadenie vyšlo na trh v roku 2008. V súčasnej dobe je Android najrozšírenejší OS na mobilných zariadeniach [3].

Android bol vytvorený za pomoci programovacích jazykov, ako sú Java, C a C++ [4]. Preto je na strane vývojárov kódovanie zložitých rozhraní často náročnou úlohou, ktorá si vyžaduje väčšie spoliehanie sa na Javu ako na Objective-C (u iOS). Aplikácie vytvorené pre OS Android tým pádom môžu mať pocit nižšieho štandardu pre používateľa ako aplikácie napísané pre iOS.



Obrázok č. 2: Android logo  
(Zdroj: [5])

Android je open-source projekt, a teda je pre výrobcov smartphonov ponúkaný zadarmo a tí ho môžu následne aj modifikovať podľa vlastného uváženia [6]. Výrobcovia zariadení s Androidom, ako napr. Samsung, Huawei, Xiaomi, atď, si vytvárajú vlastné grafické nadstavby, ktoré inštalujú navrch OS. Takéto prekrytie užívateľského prostredia je v podstate vlastný dizajn, ktorý pridáva extra funkcie do zariadenia a ďalšie vylepšenia, ktoré poskytujú jedinečný zážitok pri využívaní zariadenia od vybraného výrobcu zariadenia. Na druhej strane, zariadenia bez akýchkoľvek väčších úprav na užívateľskom rozhraní sa všeobecne označuje ako *stock Android*. Tieto úpravy užívateľského rozhrania však vo všeobecnosti, až na niektoré výnimky, neovplyvňujú funkcionálnosť OS Android a rovnakú aplikáciu je možné spustiť na všetkých zariadeniach od hociktorého výrobcu. Je to však vec, ktorú by vývojári mali brať v úvahu [2].

### 1.1.3.2 iOS

iOS je mobilný OS, ktorý vytvorila a využíva spoločnosť Apple na svojich mobilných zariadeniach iPhone. Prvýkrát bol uvedený na trh v roku 2007. iOS popularizoval mnoho funkcií ako je zväčšovanie, zmenšovanie, aj gesto potiahnutím prsta. iOS sa taktiež vyznačuje aj častými aktualizáciami a spoločnosť Apple je známa taktiež tým, že svoje zariadenia podporuje mnoho rokov, či už ide o riešenie chýb a bezpečnostných problémov alebo až veľké vydania prinášajúce nové a zaujímavé funkcie [7].



**Obrázok č. 3: iOS 14 logo**  
(Zdroj: [8])

Najnovšia verzia je iOS 14 a narozdiel od Androidu je to uzatvorený systém a nemôžu ho využívať iní výrobcovia mobilných zariadení. Vytvorený bol za pomoci programovacích jazykov C, C++, Objective-C a Swift.

## 1.2 Aplikácie

Aplikácia, taktiež nazývaná aj ako aplikačný program, je software, ktorý umožňuje vykonávať špecifické funkcie priamo na povel používateľa alebo iného programu. Aplikácia môže byť samostatná alebo vytvorená ako skupina programov. Program je množina operácií, ktoré spúšťajú aplikáciu pre používateľa.

Aplikácie využívajú operačný systém a iné podporné programy (zvyčajne systémové programy) k tomu, aby fungovali. Aplikácia žiada služby od iných technológií a taktiež s nimi komunikuje prostredníctvom programovacieho prostredia aplikácií (API) [9].

### 1.2.1 Typy aplikácií

Aplikácie sa odlišujú na základe rôznych kritérií, vrátane toho, na akej platforme bežia alebo ako sú licencované.



Aplikácie bežiacie na mobilných zariadeniach sa nazývajú mobilné aplikácie. Sú napísané tak, aby fungovali ako natívne aplikácie pre špecifický hardware (v tomto prípade mobilné zariadenia so špecifickým operačným systémom) [9]. Natívne iOS aplikácie sa píše v programovacích jazykoch Swift alebo Objective-C a natívne Android aplikácie v Java alebo C++ [10].

Naopak webové aplikácie sa spúšťajú vo webových prehliadačoch, ako je napr. Mozilla Firefox či Google Chrome. Tieto aplikácie môžu byť napísané v rôznych jazykoch, ako je napr. Javascript, CSS a HTML, nemajú ale prístup k hardwaru, na ktorom sú tieto aplikácie spúšťané [9].

### **1.2.2 Mobilné vs. webové aplikácie**

Mobilné aplikácie narozdiel od webových aplikácií, ktoré sa nemusia na zariadenia inštalovať a fungujú cez webové prehliadače, ale majú obmedzenú funkcionality a fungujú iba, ak je zariadenie online, musia byť nainštalované na zariadení a dokážu využívať funkcie mobilného zariadenia ako GPS a kameru a vedia fungovať aj v offline režime. Medzi ďalšie výhody mobilných aplikácií bezprostredne patrí, že sú rýchlejšie ako webové aplikácie, viac zabezpečenejšie (aplikácie najprv musia byť schválené distribučnou platformou, v ktorej budú publikované) a ich vytváranie je ľahšie vďaka dostupným vývojárskym nástrojom. Na druhú stranu aj mobilné aplikácie majú svoje nevýhody ako potreba vytvoriť vlastné verzie pre rôzne operačné systémy a tvorba, údržba a aktualizácia mobilných aplikácií je drahšia ako tvorba webových aplikácií [10].

### **1.2.3 Licencovanie aplikácií**

Aplikácie sa delia taktiež aj podľa typu licencovania. Patentované (uzavreté) softwarové aplikácie sú predmetom vlastníctva autorských práv. Koncoví používatelia si môžu zakúpiť licenciu k patentovanej aplikácii prostredníctvom predajcu poskytujúceho daný software. Používateľ nemôže modifikovať takéto aplikácie, ale môže doplniť ich funkcionality prostredníctvom doplnkov od tretích strán.

Na druhej strane Open source aplikácie poskytujú koncovým používateľom zdrojový kód na využitie alebo jeho manipuláciu. Takéto aplikácie vývojári zvyčajne ponúkajú zadarmo na základe verejnej licencie [9].

### **1.3 Digitálne distribučné platformy**

Digitálna distribúcia je metóda distribúcie, pri ktorej sa obsah dodáva používateľom bez dodania fyzického média, zvyčajne prostredníctvom sťahovania z internetu priamo na zariadenie používateľa. V dnešnej dobe má digitálna distribúcia prednosť pred konvenčnými metódami fyzickej distribúcie.

Digitálne distribučné platformy zvyčajne vyžadujú pred využívaním ich služieb prihlásenie používateľa. Používateľovi sa na týchto platformách ponúkajú ukážky v podobe obrázkov, videí, častí skladieb, vybraných strán knihy, atď. Používateľom vyžiadany obsah sa po získaní licencie a vyžiadaní stiahnutia transferuje z webového servera na pevný disk počítača. Tieto platformy sa neustále vylepšujú, aby svojim zákazníkom poskytli bezpečné online transakcie, spotrebiteľské licencie, ale aj vytvorili účinné opatrenia proti pirátstvu.

Digitálnu distribúciu si môžu zvoliť vývojári, vydavatelia, umelci a iní, ak chcú svoju prácu ľahkým spôsobom dostať do verejnej sféry záujmu s minimálnymi nákladmi. Výsledkom napokon je, že sa spotrebiteľia k tovaru dostanú lacno a ľahko, čo eskaluje vo vyššie zisky autorov poskytovaných diel na týchto platformách [11].

Medzi digitálne distribučné platformy patria známe služby ako Steam, Epic Games Store, Spotify, atď. Obchod Google Play a iOS App Store, prostredníctvom ktorých môžeme ľahko a bezpečne sťahovať aplikácie do našich mobilných zariadení, sa taktiež zaraďujú medzi digitálne distribučné platformy.

#### **1.3.1 Obchod Google Play**

Obchod Google Play ponúka pre vývojárov široké publikum, ktoré zaisťuje potencionálnu dobrú viditeľnosť aplikácie. Ďalej vývojárom ponúka menej zdĺhavý proces schvaľovania publikácie aplikácie v obchode a predloženie aplikácie k schváleniu je cenovo veľmi dostupné, je totižto zadarmo. Pri vytváraní aplikácií pre tento obchod je však nutné mať na mysli fakt, že používatelia systému Android majú v tendencii sťahovať bezplatné aplikácie [12].



**Obrázok č. 4: Logo Obchodu Google Play™**  
(Zdroj: [13])

Oproti iOS App Store je hodnotiaci tím v procese schvaľovania pre Obchod Google Play viac benevolentnejší. Vývojári teda môžu viacej experimentovať. Tým pádom sa však v obchode nachádzajú aj menej kvalitné aplikácie a s toľkým počtom povolených aplikácií môže byť pre niektorých vývojárov ťažké s ich aplikáciou vyniknúť. Systém Android je taktiež veľmi fragmentovaný a využíva ho veľa výrobcov na rôznych zariadeniach, čo by mali vývojári aplikácií na Android brať do úvahy.

Jediným nákladom vývojárov, ktorí chcú publikovať svoju aplikáciu v Obchode Google Play, je jednorazový poplatok pre vývojárov vo výške 25 USD. Vývojári získavajú 70% výnosov z predaja aplikácií, pričom zvyšných 30% ide distribučnej platforme.

Zo štatistického hľadiska aj keď Obchod Google Play generuje viac ako dvojnásobný počet sťahnutí než iOS App Store, iOS App Store na druhej strane zarába zhruba dvakrát toľko peňazí z výnosov z predaja aplikácií ako Obchod Google Play [12].

### **1.3.2 iOS App Store**

iOS App Store je pre vývojárov skvelou investíciou s primeraným registračným poplatkom a vysokým percentom z výnosu z predaja určených pre vývojárov. Aj napriek nižšiemu počtu iOS používateľov oproti Androidu majú aplikácie v iOS zvyčajne vysokú viditeľnosť. Vývojárom sa ďalej ponúkajú primerané náklady na predloženie aplikácie do obchodu, získanie súhlasu na publikáciu aplikácie v obchode však môže niekedy trvať dlho. Vývojári však majú dobrú spätnú väzbu od tímu pre kontrolu aplikácií. Nevýhodou môže byť veľa konkurencie v rámci obchodu. Pri publikácii aplikácie netreba zabúdať aj na fakt, že používatelia iOS majú väčšiu tendenciu platiť za aplikácie [12].



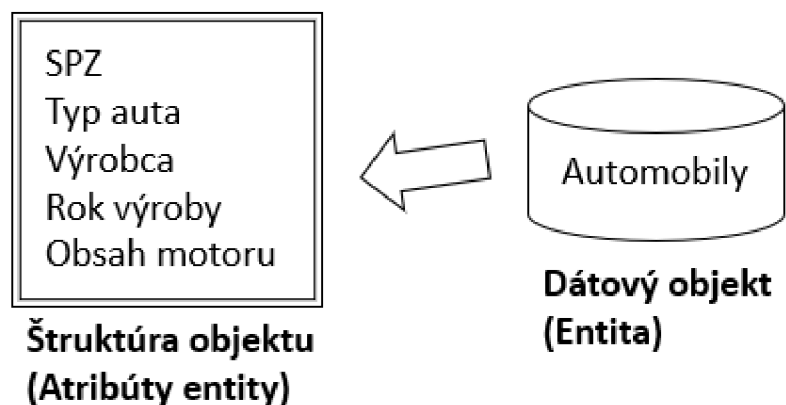
**Obrázok č. 5: iOS® App Store® logo**  
(Zdroj: [14])

Ako už bolo spomenuté, schvaľovací proces môže byť zdĺhavý a vývojári musia byť trpezliví. Keďže hodnotenia hodnotiaceho tímu obsahujú aj tvrdú spätnú väzbu, vývojári musia dbať na dodržiavanie pravidiel a zabezpečiť, aby aplikácie boli bezchybné. Chýbať nesmie ani kreativita.

Náklady pre vývojárov, ktorí chcú publikovať svoju aplikáciu v iOS App Store, sa rovnajú ročnému poplatku pre vývojárov vo výške 99 USD. Vývojári však aj v prípade tejto distribučnej platformy získavajú 70% výnosov z predaja aplikácií, pričom zvyšných 30% ide distribučnej platforme [12].

## 1.4 Dátové modelovanie

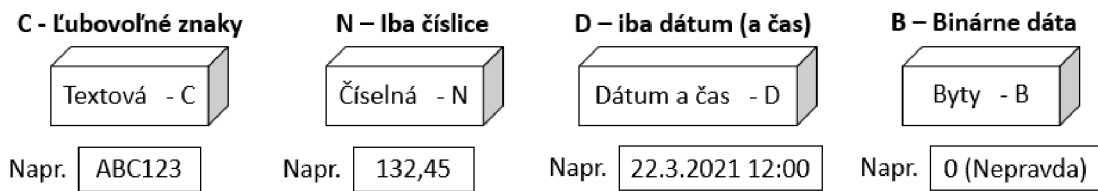
Dátový model sa skladá z entít. Entitou môžeme rozumieť hocikaký reálny objekt, či už človek, zviera alebo vec. Každá takáto entita obsahuje detaily, ktoré chceme uchovávať. Tie sa volajú atribúty. Množina atribútov entity sa nazýva štruktúra objektu [15].



Obrázok č. 6: Entita a jej atribúty  
(Zdroj: Vlastné spracovanie podľa [15])

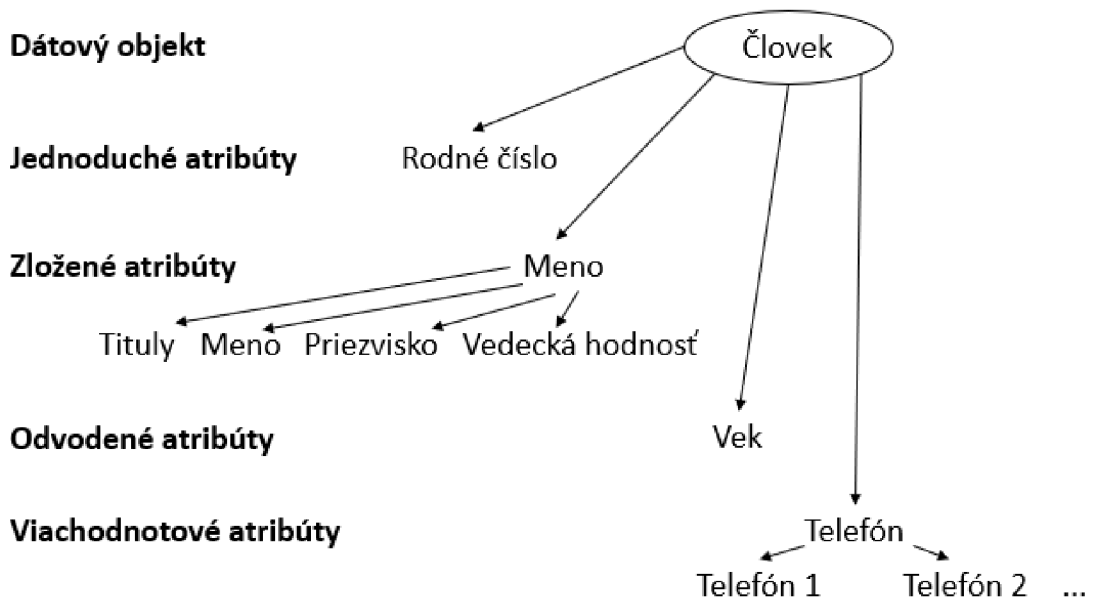
### 1.4.1 Atribút

Atribút alebo dátová položka je údaj, ktorý je atomický a ďalej nedeliteľný, ak sa teda nerozhodneme brať skupinu údajov ako jednu položku (napr. adresa). Tieto atribúty majú svoj typ a dĺžku (pevnú alebo premenlivú) [15].



**Obrázok č. 7: Typy atribútov**  
(Zdroj: Vlastné spracovanie podľa [15])

Atribúty sa ďalej delia na jednoduché (např. rodné číslo), zložené (např. meno), odvodené (např. vek) a viachodnotové (např. telefón) [15].



**Obrázok č. 8: Zložitost' atribútov**  
(Zdroj: Vlastné spracovanie podľa [15])

## 1.4.2 Relačný dátový model

Relačný dátový model nám umožňuje zachytiť nielen dáta o skúmaných objektoch, ale taktiež aj vzájomné vzťahy medzi nimi.

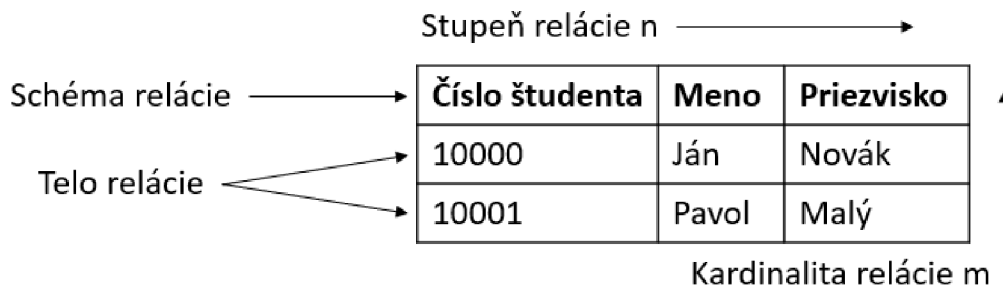
Ak máme množiny (domény) např. čísel študentov  $D_1$ , mien študentov  $D_2$  a priezvisk študentov  $D_3$ , tak potom relácia na doménach  $D_1, D_2, \dots, D_n$  je dvojica  $R = (R, R^*)$ ,

kde  $R = R(A_1:D_1, A_2:D_2, \dots, A_n:D_n)$  je schéma relácie

a  $R^* \subseteq D_1 \times D_2 \times \dots \times D_n$  je telo relácie.

$A_i$  je názov atribútu a  $D_i$  je doména, relácia R Študent je nasledovná:

Študent = {(číslo:číslo), (meno:text), (priezvisko:text)} [15].



**Obrázok č. 9: Relačný dátový model**  
(Zdroj: Vlastné spracovanie podľa [15])

Počet atribútov  $n$  relácie sa označuje ako stupeň (rád) relácie a kardinalita tela relácie  $m = |R^*|$  ako kardinalita relácie. Stupeň relácie je nemenný, zatiaľ čo kardinalita premenlivá [15].

#### 1.4.2.1 Entitná integrita

Každá relácia musí mať tzv. primárny kľúč (PK). Je to jeden alebo viacej atribútov, ktoré jednoznačne identifikujú každý riadok relácie. Má tieto 2 vlastnosti:

1. Jednoznačnosť – v relácii neexistuje druhá  $n$ -ticia, ktorá by pre danú množinu atribútov mala rovnaké hodnoty
2. Minimálnosť – žiadny atribút nie je možné vypustiť, aby sa neporušilo pravidlo 1

Každý atribút primárneho kľúča musí byť vyplnený.

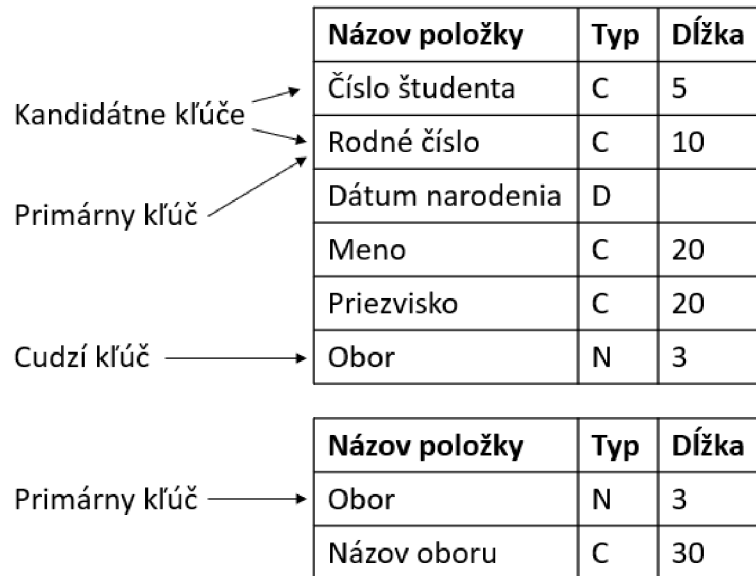
Skupina atribútov, ktorá predošlé podmienky spĺňa taktiež, ale nebola vybraná ako primárny kľúč sa nazýva kandidátny kľúč (CK). V relácii môže byť viacero kandidátnych kľúčov, ale iba jeden je vybraný ako primárny a ostatné sa stávajú alternatívnymi kľúčmi [15].

#### 1.4.2.2 Referenčná integrita

V relačnom modeli túto funkciu spĺňa cudzí kľúč (FK). Má tieto 2 vlastnosti:

1. Každá hodnota musí byť plne zadaná alebo plne nezadaná
2. Existuje iná relácia s takým primárnym kľúčom, že každá zadaná hodnota cudzieho kľúča je identická s hodnotou primárneho kľúča nejakej  $n$ -tice tejto inej relácie

Cudzí a primárny kľúč musia byť definované na rovnakej doméne a nesmie existovať žiadna hodnota cudzieho kľúča, ktorá sa nezhoduje s primárnym kľúčom [15].



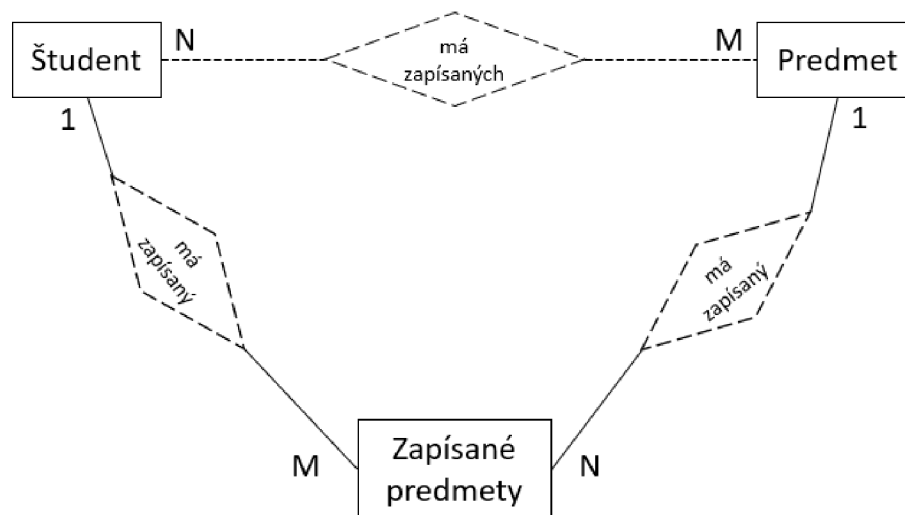
**Obrázok č. 10: Integrita relačného dátového modelu**  
(Zdroj: Vlastné spracovanie podľa [15])

### 1.4.2.3 Integritné obmedzenie vzťahov

V relačnom dátovom modeli poznáme nasledujúce druhy vzťahov:

- Vzťah 1:1 – jednej n-tici odpovedá jedna alebo žiadna n-tica inej relácie. Napr.: Jednému človekovi prislúcha jeden alebo žiadny vodičský preukaz a jeden vodičský preukaz prislúcha jednému človekovi
- Vzťah 1:N – jednej n-tici odpovedá jedna alebo viacej n-tíc inej relácie. Napr.: Študent skladá jednu alebo viacej skúšok, ale 1 konkrétna skúška prislúcha iba jednému študentovi
- Vzťah N:1 – obdoba predošlého
- Vzťah N:M – niekoľko n-ticám jednej relácie prislúcha jedna alebo viac n-tíc inej relácie. Napr.: Študent má zapísaný jeden alebo viacej predmetov a 1 predmet má zapísaný jeden alebo viacej študentov

V poslednom prípade vzťahu N:M je potreba vykonať dekompozíciu. Tá sa vykoná vytvorením pomocnej relácie obsahujúcej cudzie kľúče do oboch predošlých relácií. Tento úkon demonštruje nasledovný obrázok [15].

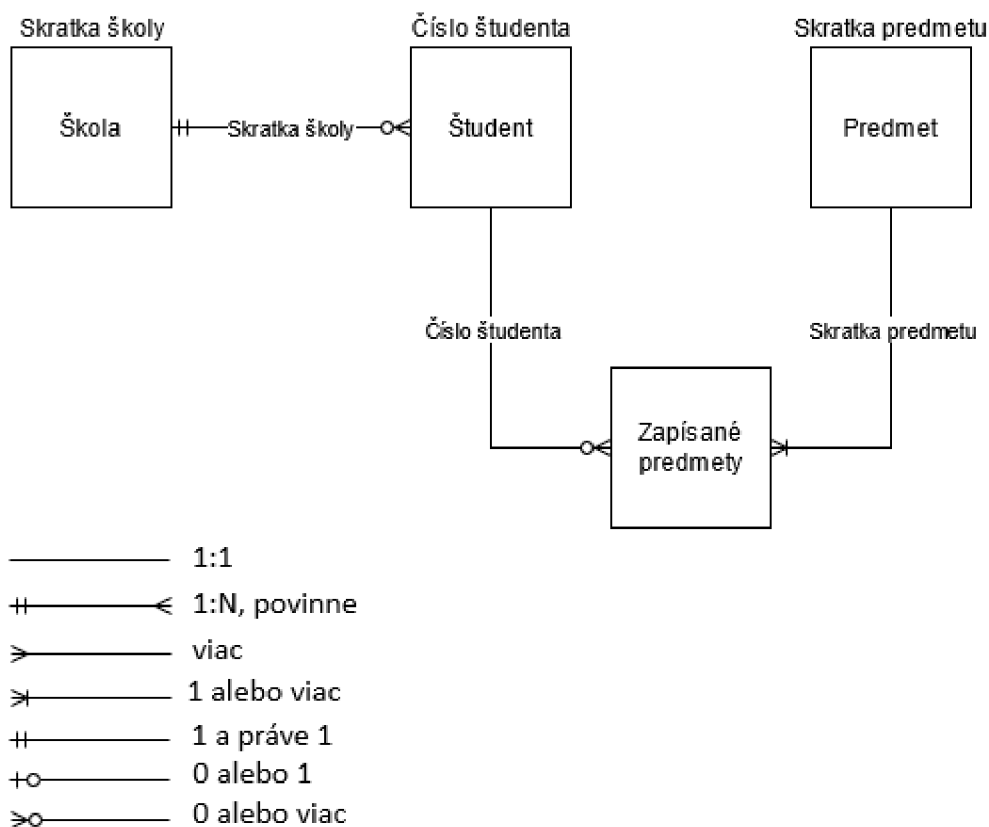


**Obrázok č. 11: Dekompozícia vzťahu N:M**  
(Zdroj: Vlastné spracovanie podľa [15])

### 1.4.3 Entito-relačný model

Ako je už možné vidieť z názvu, tento model graficky vyobrazuje relácie nad entitami. Existujú rôzne štýly, akými je možné tento model zobraziť. Pre účely tejto práce si vyberiem tzv. „inžiniersky štýl“. Pomocou tohto štýlu zobrazím dekompozíciu z predošlej podkapitoly, model ale doplním o entitu školy, pričom medzi školou a študentom bude fungovať väzba 1:N.





**Obrázok č. 12:** „Inžiniersky štýl“ entito-relačného modelu  
 (Zdroj: Vlastné spracovanie podľa [15])

#### 1.4.4 SQL Server

Návrh mobilnej aplikácie závisí aj od návrhu relačného modelu, ktorý bude opisovať, akým spôsobom sa budú dáta, s ktorými bude aplikácia pracovať, ukladať a aké budú medzi nimi vzťahy. Tento relačný model sa musí vo forme tabuliek relačnej databáze vytvoriť na SQL serveri, ktorý bude fungovať napr. na databázovej platforme Microsoft SQL Server 2019.

Microsoft SQL Server nepredstavuje iba databázu, ale ponúka aj komplexnú, výkonnú, spoľahlivú a bezpečnú serverovú platformu pre ukladanie a správu údajov v databázach a dátových skladoch a sadu nástrojov pre Business Intelligence vrátane pokročilého reportovania. Táto platforma si poradí nielen s relačnou databázou a multidimenzionálnymi údajmi (dátové sklady), ale poskytuje aj úložisko pre multimediálne súbory a dokumenty.

Na tomto SQL serveri sa vytvorí relačná databáza, čo je vlastne súbor nástrojov pre efektívne a spoľahlivé ukladanie údajov a pre manipuláciu s týmito údajmi. Za pomoci

jazyka T-SQL je následne možné spravovať tabuľky v databáze a vytvárať nad nimi transakcie a triggery [16].

#### 1.4.4.1 Dátové typy

V SQL serveri má každý stĺpec alebo premenná svoj dátový typ. Nasledovná tabuľka obsahuje hlavné dátové typy, ktoré je možné využiť prostredníctvom T-SQL jazyka [17].

**Tabuľka č. 1: Dátové typy jazyka T-SQL**

(Zdroj: Vlastné spracovanie podľa [17])

Dátový typ	Popis	Rozmedzie	Veľkosť na disku
int	celé číslo	od -2 147 483 648 do 2,147,483,647 ( $2^{31}$ )	4 bajty
bigint	celé číslo	od -9 223 372 036 854 775 808 do 9 223 372 036 854 775 807 ( $2^{63}$ )	8 bajtov
smallint	celé číslo	od -32,768 do 32,767 ( $2^{15}$ )	2 bajty
tinyint	celé číslo	od 0 do 255	1 bajt
bit	boolean	null,0,1	1 bit
money	mena	obdoba bigint	8 bajtov
smallmoney	mena	obdoba int	4 bajtov
decimal	precízne číslo	precíznosť od 1 do 38	od 5 bajtov do 17 bajtov
numeric	precízne číslo	precíznosť od 1 do 38	od 5 bajtov do 17 bajtov
float	približné (desatinné) číslo	počet číslic od 1 do 53	od 4 bajtov do 8 bajtov
real	približné (desatinné) číslo		4 bajty
char	pevný reťazec znakov	od 1 do 8000 znakov	od 1 bajtov do 8000 bajtov
varchar	premenlivý reťazec znakov	od 1 do 8000 znakov, max	od 1 bajtov do 8000 bajtov, až do 2 gigabajtov
text	text	až do dĺžky reťazca 2 147 483 647 znakov	2*dĺžka reťazca
datetime	dátum a čas	roky od 1753 do 9999	8 bajtov
binary	binary	až do veľkosti int	
image	obrázok	až do veľkosti int	

## 1.5 Funkčné modelovanie

Ak sa dátové modelovanie zaoberá problematikou dát, ktoré bude využívať informačný systém, je potrebné využiť funkčné modelovanie. Funkčné modelovanie sa zaoberá skúmaním a algoritmizáciou činností a procesov, ktoré v informačnom systéme prebiehajú.

### 1.5.1 Diagram toku dát (DFD)

Diagram toku dát je jedna z najpoužívanejších metód funkčného modelovania. Poskytuje prehľad o nadväznosti jednotlivých činností v rámci úlohy, aké dátové vstupy a výstupy sa v úlohe objavujú a kto jednotlivé činnosti vykonáva. Pre potreby tejto práce využijem štýl *Yourdon and Coad*.

Diagram toku dát sa skladá z týchto prvkov:

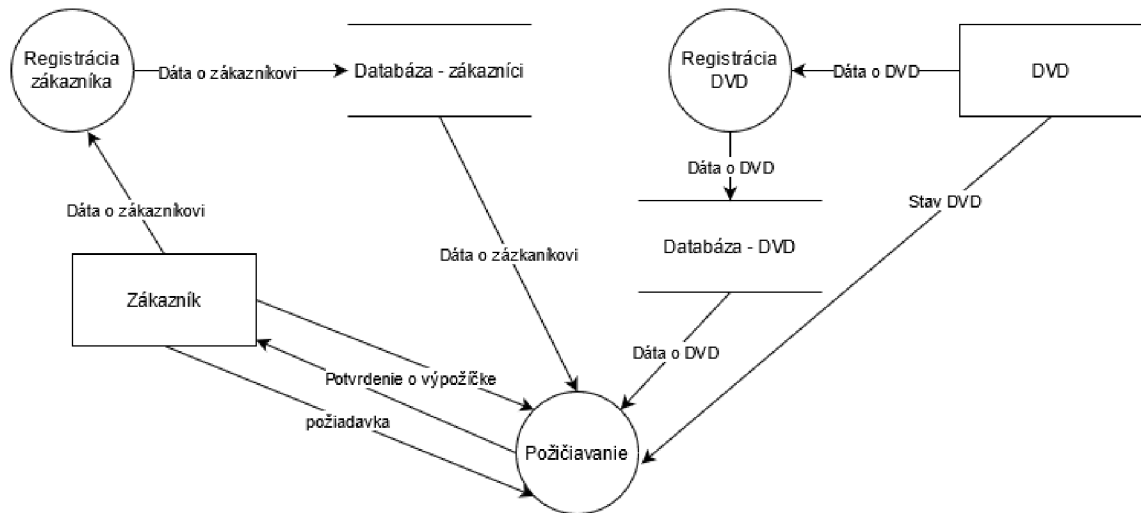
- Procesy – činnosti, pri ktorých sa transformujú vstupné dáta na výstupné
- Externá entita – objekt mimo systém, s ktorým proces komunikuje
- Uloženie dát – dátový súbor alebo zostava. Je to pasívny objekt, ktorý ukladá dáta pre neskoršie spracovanie
- Pomenovaný dátový tok – presun dát z jednej časti systému do druhej

Diagram toku dát sa riadi nasledujúcimi pravidlami:

1. Nemá obsahovať viac ako 10 procesov
2. Nesmie existovať proces, ktorý nemá žiadne vstupy ale má iba výstupy a naopak
3. Dátový tok z externej entity/uloženia musí ísť cez proces
4. Dátový tok nemá byť priamo medzi dvoma procesmi [15]



Obrázok č. 13: Prvky diagramu toku dát podľa štýlu *Yourdon and Coad*  
(Zdroj: Vlastné spracovanie podľa [15])



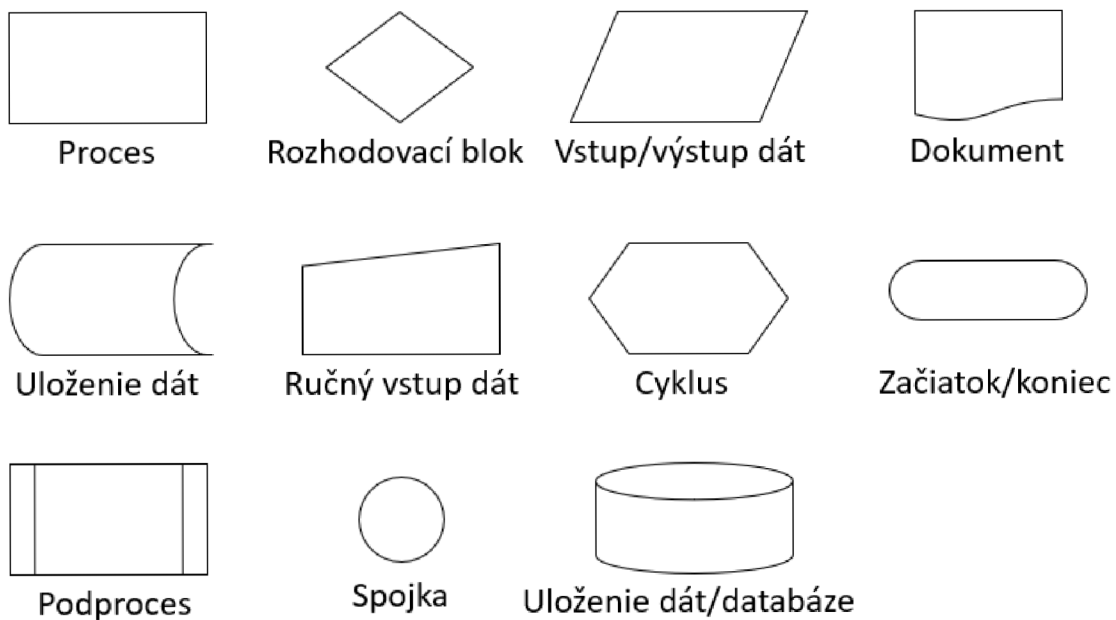
**Obrázok č. 14: Diagram toku dát**  
(Zdroj: Vlastné spracovanie podľa [15])

### 1.5.2 Vývojový diagram

Jedným z ďalších najpoužívanejších metód funkčného modelovania je aj vývojový diagram. Výhodou vývojového diagramu je možnosť zachytiť vetvenie podľa plnenia požadovaných podmienok [15].

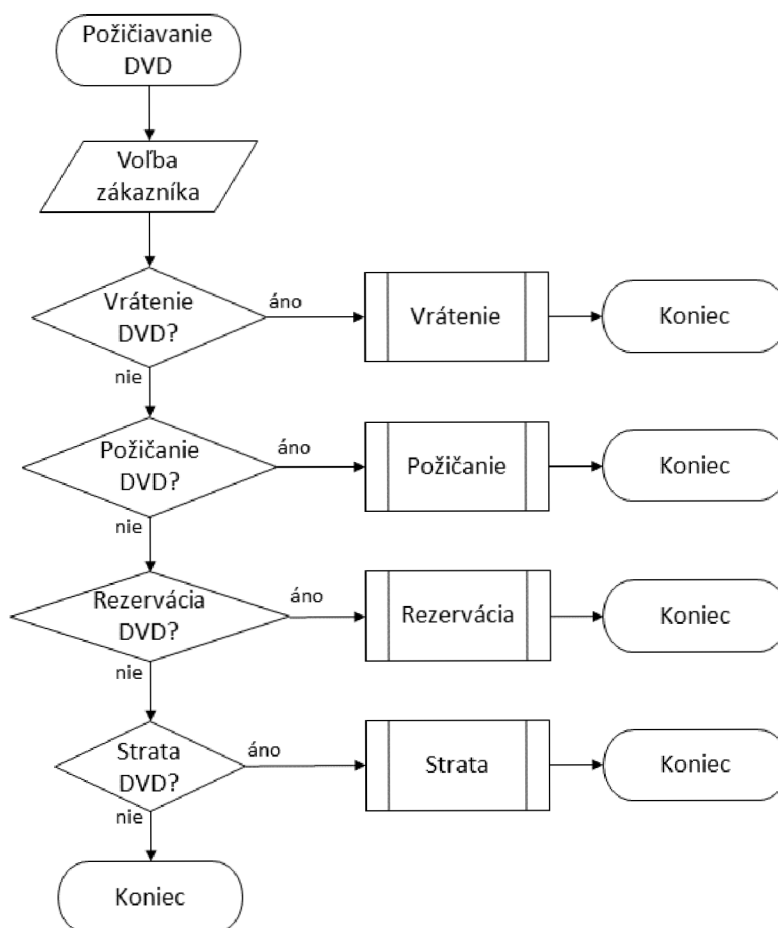
Medzi hlavné prvky vývojového diagramu patria:

- Začiatok/koniec – predstavuje začiatkový alebo konečný bod systému
- Proces – označuje konkrétnu operáciu/úkon
- Dokument – predstavuje nejaký výtlačok, napr. dokument alebo zostavu
- Rozhodovací blok – predstavuje bod rozhodnutia alebo rozvetvenia. Čiary vychádzajúce zo symbolu označujú rôzne možné situácie, ktoré vedú k rôznym čiastkovým procesom
- Vstup/výstup dát – predstavuje informácie vstupujúce a vystupujúce zo systému
- Spojka – vnútri obsahuje znak a znamená, že tok systému pokračuje na druhej spojke so zhodným znakom [18]



**Obrázok č. 15: Prvky vývojového diagramu**

(Zdroj: Vlastné spracovanie podľa [15])



**Obrázok č. 16: Vývojový diagram**

(Zdroj: Vlastné spracovanie podľa [15])

## 1.6 Metódy zhodnotenia výsledku práce

### 1.6.1 SWOT analýza

SWOT analýza je jedna z najčastejšie využívaných analytických metód. Výsledkom tejto analýzy je SWOT matica.

SWOT je skratka z anglického originálu a znamená:

- Strengths – silné stránky
- Weaknesses – slabé stránky
- Opportunities – príležitosti
- Threats – hrozby

SWOT v podstate vyjadruje vnútorné silné a slabé stránky organizácie/produktu a príležitosti a hrozby vychádzajúce z vonkajšieho prostredia. Z komplexnejšieho hľadiska sa tieto vnútorné faktory stavajú proti vonkajším faktorom a na ich základe sa formulujú stratégie ďalšieho rozvoja [19].

	<b>Slabé stránky (W)</b>	<b>Silné stránky (S)</b>
<b>Príležitosti (O)</b>	<b>WO stratégia</b> <b>„hľadania“</b> Prekonanie slabých stránok za využitia príležitostí	<b>SO stratégia</b> <b>„vyžitia“</b> Využitie silných stránok v prospech príležitostí
<b>Hrozby (T)</b>	<b>WT stratégia</b> <b>„vyhýbania sa“</b> Minimalizácia slabých stránok a vyhnutie sa hrozbám	<b>ST stratégia</b> <b>„konfrontácie“</b> Využitie silných stránok k odvráteniu hrozieb

**Obrázok č. 17: SWOT matica**  
(Zdroj: Vlastné spracovanie podľa [19])

### 1.6.2 Cenová politika

Cena nových produktov sa zvyčajne stanovuje podľa zrovnateľných produktov na trhu a reakcií zákazníka na tieto ceny. Pri zavádzaní nových produktov na trh sa využívajú 2 typy cien:

- Vysoko výnosné ceny – stanovené pomerne vysoko a sú niekoľkonásobne vyššie než celkové výrobné náklady. Využíva sa najčastejšie pri zavedení úplnej novinky (napr. nový produkt s jedinečnými vlastnosťami, ktoré v porovnaní s vlastnosťami už existujúcich produktov na trhu dokážu doplniť kvalitatívne odlišné používateľské funkcie), ktorá je schopná pre určitú dobu odolávať napodobeniam alebo ak je produkt vyhľadávaný zákazníkmi, ktorých nezaujíma jeho cena
- Prienikové ceny – jedná sa o nižšie ceny, ktoré podniku umožňujú rýchlo preniknúť na trh, dosiahnuť na ňom požadovaného podielu a zaistiť rentabilitu prostredníctvom veľkého obratu. Tieto ceny dokonca zvyknú zoslabiť atraktivitu trhu pre prípadnú konkurenciu. Nízkou cenou sa podnik snaží orientovať na široký okruh spotrebiteľov. Tento typ cenovej politiky sa ale zvyčajne uplatňuje iba v prípade silnej finančnej pozície podniku [20]

## **1.7 Medicínske pojmy v selfmonitoringu**

Súčasťou selfmonitoringu pacientov je vlastnoručné nameranie hodnôt krvného tlaku a pulzu, glukózy, telesnej teploty a aktualizácia osobných údajov (výška, váha).

### **1.7.1 Meranie krvného tlaku a pulzu**

*„Krvný tlak je miera tlaku vyvolaného krvou pulzujúcou cez tepny. Je odrazom srdcového výdaja a periférneho odporu ciev“ (21, s. 80).*

Pri meraní krvného tlaku tlakomerom rozoznávame:

- Systolický tlak – najvyššia hodnota pri kontrakcii komôr srdca
- Diastolický tlak – najnižšia hodnota, keď sú komory srdca v pokoji
- Tlaková amplitúda – rozdiel predošlých dvoch tlakov [21]

**Tabuľka č. 2: Klasifikácia hypertenzie (vysokého krvného tlaku)**

(Zdroj: Vlastné spracovanie podľa [21])

Stav	Hodnoty systolického tlaku	Hodnoty diastolického tlaku
Optimálny TK	< 120	< 80
Normálny TK	120 – 129	80 – 84
Vyšší normálny TK	130 – 139	85 – 89
Hypertenzia 1. stupňa	140 – 159	90 – 99
Hypertenzia 2. stupňa	160 – 179	100 – 109
Hypertenzia 3. stupňa	≥ 180	≥ 110
Izolovaná systolická hypertenzia	≥ 140	< 90

„Pulz (tep) je náraz krvného prúdu na stenu tepny pri kontrakcii ľavej srdcovej komory.

Účelom merania pulzu je zistiť adekvátnosť krvného obehu“ (21, s. 76).

Pulz môže byť nameraný buď tlakomerom, ak túto funkciu ponúka, alebo si ho pacient nameria sám pohmatom priložením brušiek troch stredných prstov na žilu pri zápästí [21].

**Tabuľka č. 3: Fyziologické hodnoty pulzu vo vzťahu k veku**

(Zdroj: Vlastné spracovanie podľa [21])

Vek	Počet pulzov/min
Novorodenec	120 – 140
Dojča	100 – 120
10-ročné dieťa	80 – 90
Dospelý človek	70 – 90

### 1.7.2 Meranie telesnej teploty

Pri určitých chorobách je vhodné v lekárskom daných časových intervaloch zaznamenávať telesnú teplotu pacienta. Telesnú teplotu môžeme merať na rôznych miestach tela:

- V pazuchovej jame
- V uchu
- V ústach
- V konečníku
- V slabine
- Na koži (napr. čelo) [21]



**Tabuľka č. 4: Hodnoty telesnej teploty**

(Zdroj: Vlastné spracovanie podľa [21])

Typ teploty	Hodnota
Subnormálna teplota	pod 36 °C
Fyziologická teplota	36 – 36,9 °C
Zvýšená teplota	37 – 37,9 °C
Horúčka	38 – 40 °C
Hyperpyrexia	Nad 40 °C

**1.7.3 Meranie a váženie**

Na základe výšky a váhy je možné vypočítať telový hmotnostný index (BMI) podľa vzorca

$$\text{BMI (kg/m}^2\text{)} = \text{hmotnosť tela (kg)} / \text{výška (m}^2\text{)} [21].$$

**Tabuľka č. 5: BMI**

(Zdroj: Vlastné spracovanie podľa [21])

Stav výživy	BMI
Podvýživa	< 18,5
Norma	18,5 – 24,9
Nadváha	25 – 29,9
Obezita 1. stupňa	30 – 34,9
Obezita 2. stupňa	35 – 39,9
Obezita 3. stupňa	> 40

**1.7.4 Glukóza v krvi**

Hladinu glukózy v krvi si pacient doma môže zistiť z kapilárnej krvi (odberom z prsta) pomocou glukomera. Intervaly odberu sa budú vykonávať podľa stanoviska lekára [22].

**Tabuľka č. 6: Referenčné hodnoty biochemického vyšetrenia krvi**

(Zdroj: Vlastné spracovanie podľa [22])

Parameter	Referenčné hodnoty		Laboratórna jednotka
	Muži	Ženy	
S - glukóza	3,50 – 6,10	3,50 – 6,10	mmol/l

## 2 ANALÝZA SÚČASNÉHO STAVU

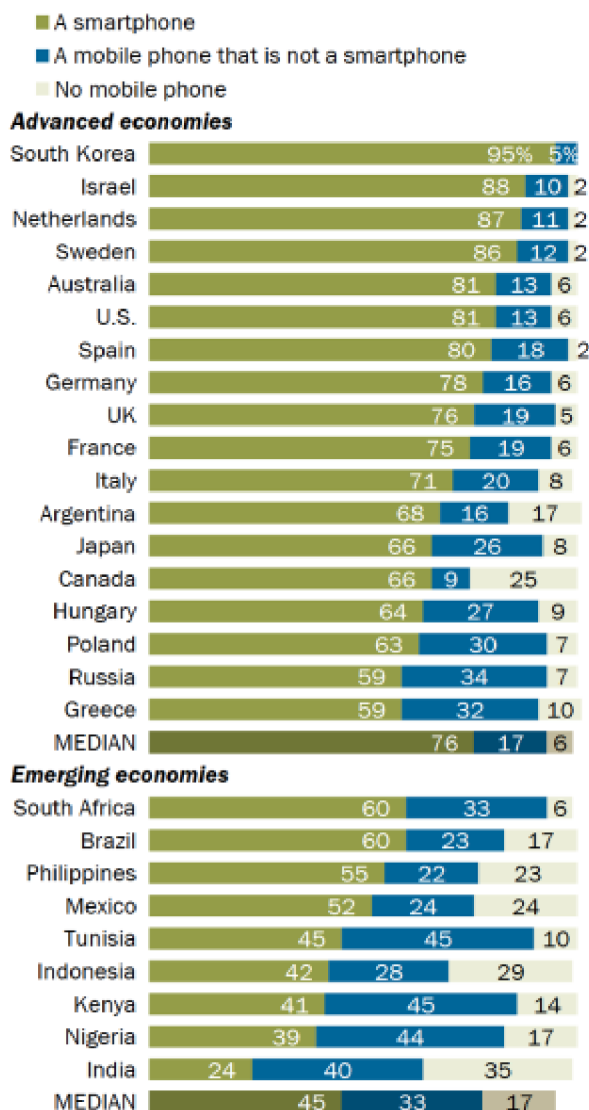
V tejto kapitole sa budem venovať analýze trhu mobilných zariadení a operačných systémov na tomto trhu. Následne pomocou internetu a Obchodu Google Play zistím prítomnosť podobných aplikácií na rezerváciu termínov u lekárov. Do úvahy budem brať najmä aplikácie určené pre český a slovenský trh. Na záver zhrniem výsledky analýz a posúdim, na ktoré operačné systémy je navrhovaná aplikáciu najvhodnejšie zamerať.

### 2.1 Analýza trhu mobilných zariadení

Využívanie mobilných zariadení sa vo svete líši najmä v závislosti, či sa vlastníci mobilných zariadení nachádzajú v krajine s rozvinutou alebo rozvíjajúcou sa ekonomikou. Z grafu nižšie je možné vidieť, že prevaha vlastníctva mobilných zariadení je práve v krajinách s rozvinutou ekonomikou, kde približne 76% opýtaných dospelých vlastníkov zo vzorky 18 krajín s rozvinutou ekonomikou vlastní smartphoney, 17% vlastní tzv. „hlúpy telefón“ a 6% nevlastní žiadne mobilné zariadenie. Táto prevaha pramení najmä z faktu, že ľudia žijúci v krajinách s rozvinutou ekonomikou majú viac finančných prostriedkov na kúpu mobilných zariadení, najmä drahších smartphonov. Na druhej strane v krajinách s rozvíjajúcou sa ekonomikou sú čísla vlastníkov mobilných zariadení o niečo nižšie, približne 45% všetkých opýtaných dospelých vlastníkov mobilných zariadení zo vzorky 9 krajín s rozvíjajúcou sa ekonomikou vlastní smartphoney, 33% vlastní „hlúpy telefón“ a až 17% nevlastní žiadne mobilné zariadenie. Ako je možné vidieť, aj napriek finančným rozdielom v sledovaných krajinách, aj v krajinách s rozvíjajúcou sa ekonomikou prevládajú smartphony. Je to najmä pre skutočnosť, že obecné prinášajú užitočnejšie funkcie ako „hlúpe telefóny“, napr. možnosť sťahovania a užívania mobilných aplikácií, jednoduchšia navigácia na internete, lepší konzum multimedialného obsahu alebo pokročilejšie funkcie fotoaparátu.

## Smartphone ownership in advanced economies higher than in emerging

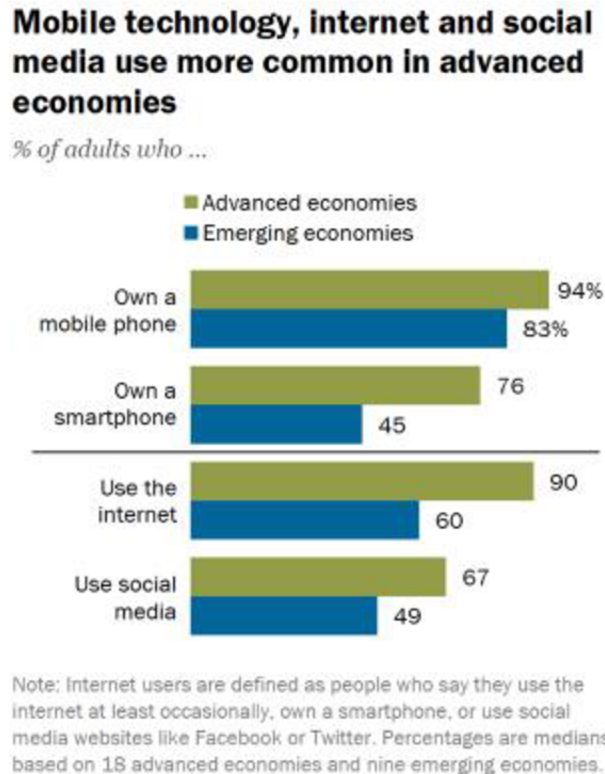
% of adults who report owning ...



**Graf č. 1: Vlastníctvo smartphonov v rozvinutých ekonomikách vyššie ako v rozvíjajúcich sa (v %)**  
(Zdroj: [23])

Na základe ďalšieho grafu je vidieť, že teda približne 94% opýtaných dospelých zo vzorky 18 krajín s rozvinutou ekonomikou vlastní nejaké mobilné zariadenie, z toho 76% vlastní smartphone. 90% týchto respondentov využíva internet na ich mobilných zariadeniach a z toho 67% aj sociálne siete. Naopak, z respondentov z krajín s rozvíjajúcou sa ekonomikou vlastní nejaké mobilné zariadenie síce nižší, ale stále vysoký počet 83% všetkých opýtaných a z toho o mnoho nižších 45% vlastní smartphone. Z hľadiska využitia svojich mobilných zariadení respondenti z krajín s rozvíjajúcou sa

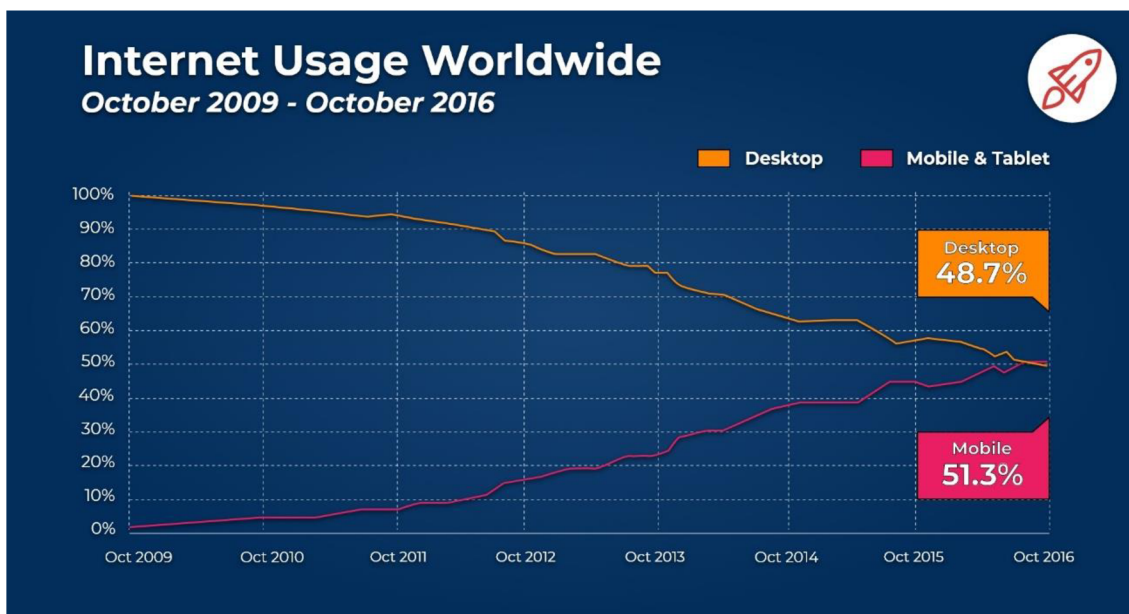
ekonomikou 60% respondentov odpovedalo, že využíva internet a 49% využíva aj sociálne siete.



**Graf č. 2: Mobilné technológie, internet a sociálne médiá sú využívané viac v rozvinutých ekonomikách (v %)**  
(Zdroj: [23])

Využívanie mobilných zariadení každým rokom stúpa a tak aj využívanie internetu prostredníctvom týchto zariadení. Ako je možné vidieť na grafu nižšie, za 7 rokov stúplo využívanie internetu z takmer 0% až na 51,3% prostredníctvom mobilných zariadení, zatiaľ čo trend pri desktopových riešeniach počas sledovaného obdobia výrazne klesal.

Rok 2016 bol jediným rokom, kedy mobilné zariadenia prebehli tie desktopové vo využívaní internetu. Pre rok 2019 to bolo 53% celkového webového prenosu prostredníctvom mobilných zariadení a 56,7% prostredníctvom desktopových zariadení [24].



**Graf č. 3: Využitie internetu vo svete (v %)**

(Zdroj: [24])

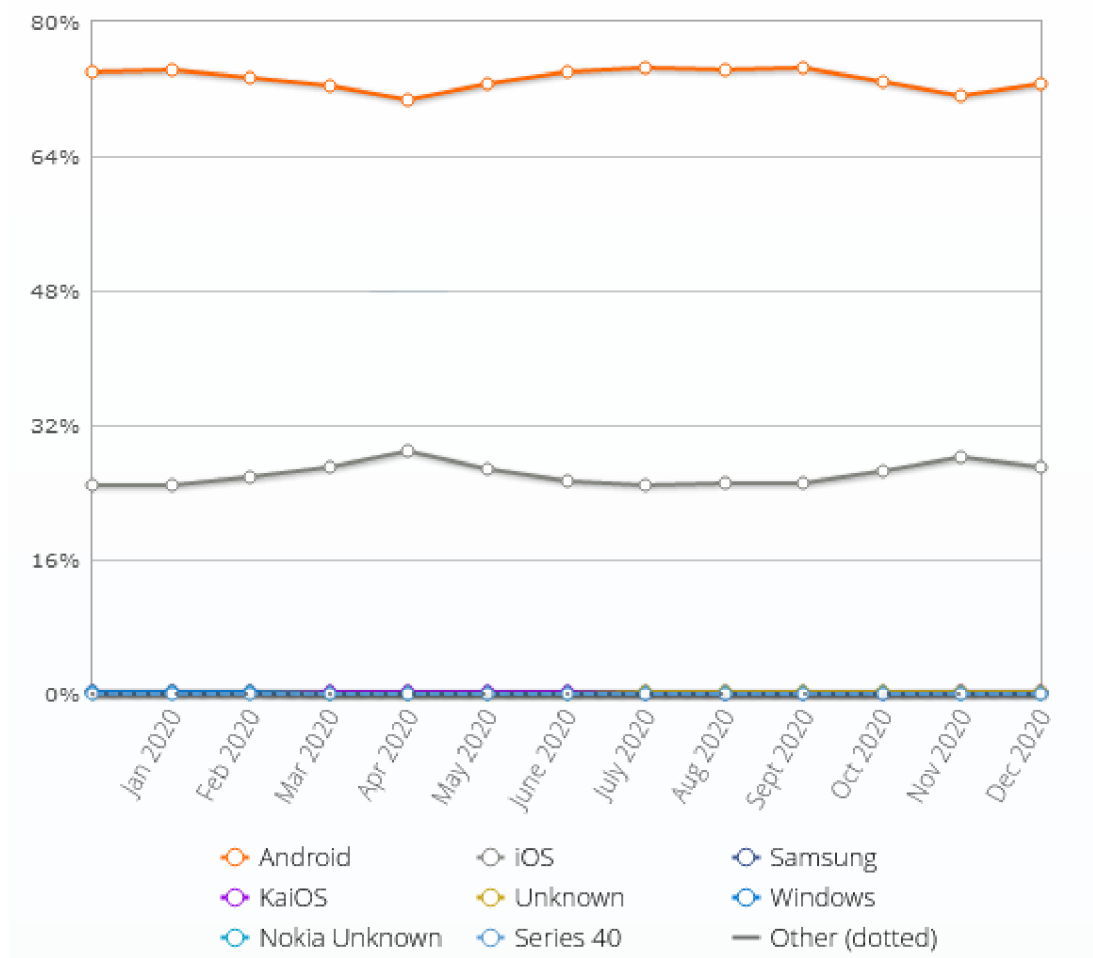
Do trhu mobilných zariadení neodmysliteľne patria aj mobilné aplikácie a webové aplikácie. Nedávne výskumy ukazujú, že používatelia mobilných zariadení 90% času strávených na týchto zariadeniach strávia v mobilných aplikáciách a iba zvyšných 10% času surfujú na internete. Tento rozdiel je vidno aj v oblasti e-commerce, kde používatelia mobilných zariadení nakupujú až 201,8 minút v mesiaci v porovnaní s 10,9 minút za mesiac na webových stránkach. Predpokladá sa, že tento rozdiel sa bude zväčšovať ešte viac, keďže stále viac firiem prechádza na riešenia webových a mobilných aplikácií [25].

## 2.2 Analýza trhu operačných systémov

Mobilné aplikácie na rozdiel od webových aplikácií je potrebné prispôbiť pre jednotlivé mobilné operačné systémy. Pre vysoký počet stiahnutí a užívaní aplikácie je preto najvýhodnejšie vytvoriť rôzne verzie aplikácie len pre najrozšírenejšie a najpoužívanejšie operačné systémy (vyššia istota potenciálneho záujmu o aplikáciu a zníženie nákladov na tvorbu aplikácie). Podľa štatistík na StatCounter prevládali iba 2 mobilné operačné systémy – Android a iOS, kde Android prevláda vo všetkých smeroch s rozmedzím tržového podielu 70% - 75% všetkých operačných systémov nainštalovaných v mobilných zariadeniach v roku 2020 a iOS mierne zaostával s rozmedzím 24% - 29%. Ako je možné vidieť aj na grafe nižšie, ostatné mobilné operačné systémy majú zanedbania hodný tržný podiel blížiaci sa k 0%. Aktuálny stav tržných podielov

jednotlivých mobilných operačných systémov pre december 2020 je Android – 72,48%, iOS – 26,91%, Samsung – 0,23%, KaiOS – 0,13% a Windows – 0,02% [26].

## Mobile Operating System Market Share Worldwide Dec 2019 - Dec 2020

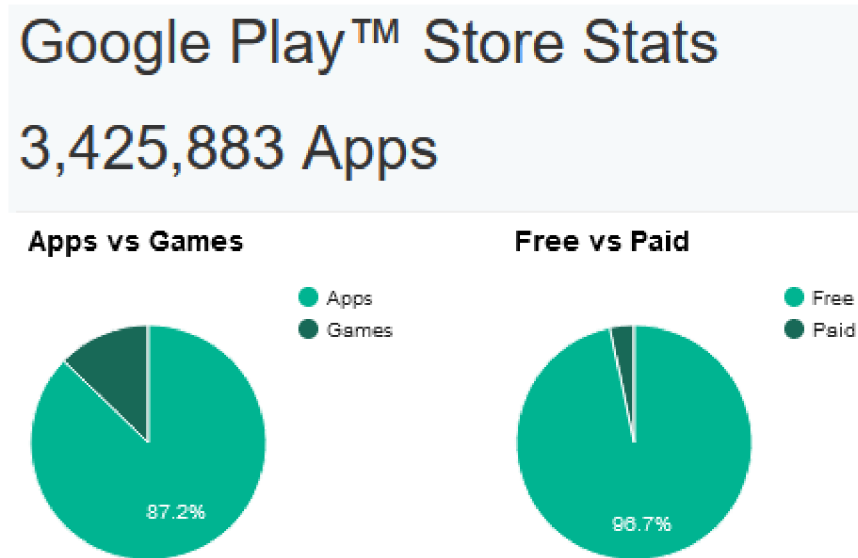


**Graf č. 4: Tržný podiel mobilných operačných systémov na svete (v %)**  
(Zdroj: [26])

### 2.3 Digitálne distribučné platformy

Výrobcovia operačných systémov v dnešnej dobe zvyknú do svojich systémov zakomponovať aj digitálne distribučné platformy (obchody pre aplikácie). Pre Android je to Obchod Google Play a pre iOS App Store. Na grafoch nižšie je možné vidieť, že Obchod Google Play oproti App Storu obsahuje omnoho vyšší počet aplikácií - rozdiel je väčší ako 1 500 000 aplikácií. V oboch platformách prevládajú aplikácie, avšak oproti

platforme od Googlu, v App Store sa nachádza viac platených aplikácií, čo môže svedčiť o tom, že jeho používatelia sú viac ochotní si za aplikácie zaplatiť.



**Graf č. 5: Obchod Google Play™ štatistiky (v %)**  
(Zdroj: [27])



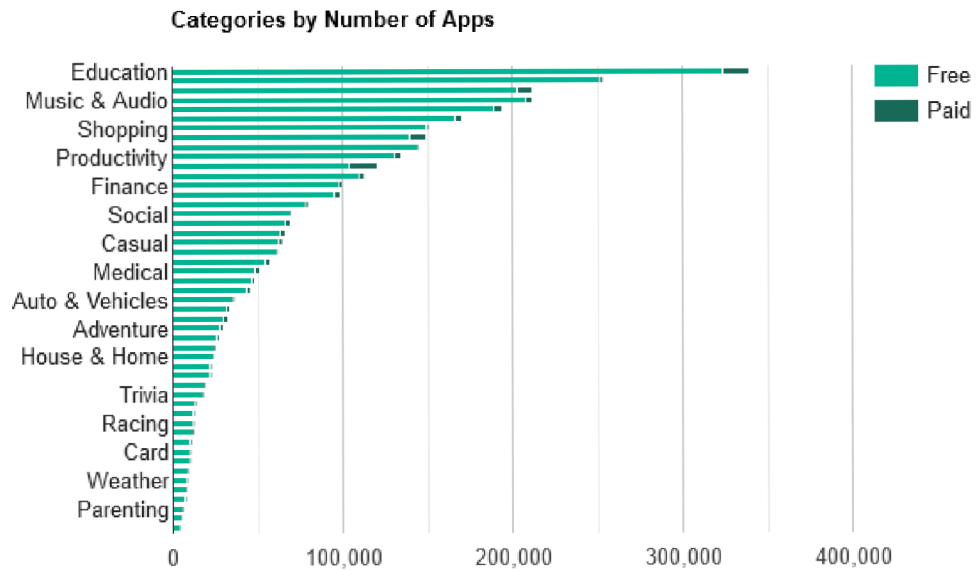
**Graf č. 6: iOS® App Store® štatistiky (v %)**  
(Zdroj: [27])

Na základe nasledujúcich grafov možno odhadnúť zacielenie na používateľov jednotlivých platforiem. Na platforme od Googlu sa kladie väčší dôraz na dospeljší segment (vzdelanie, nakupovanie, hudba), kde to na na App Store skôr na segment využívajúci svoje zariadenia pre konzum multimédií a hier. Obe však disponujú aj veľkým počtom aplikácií určených pre segment dospelých používateľov, ktorí svoje

zariadenia využívajú nie len na prácu, ale aj každodenné potreby (produktivita, zdravie, životný štýl, sociálne siete, financie,...).

## Category Distribution Google Play™ Store

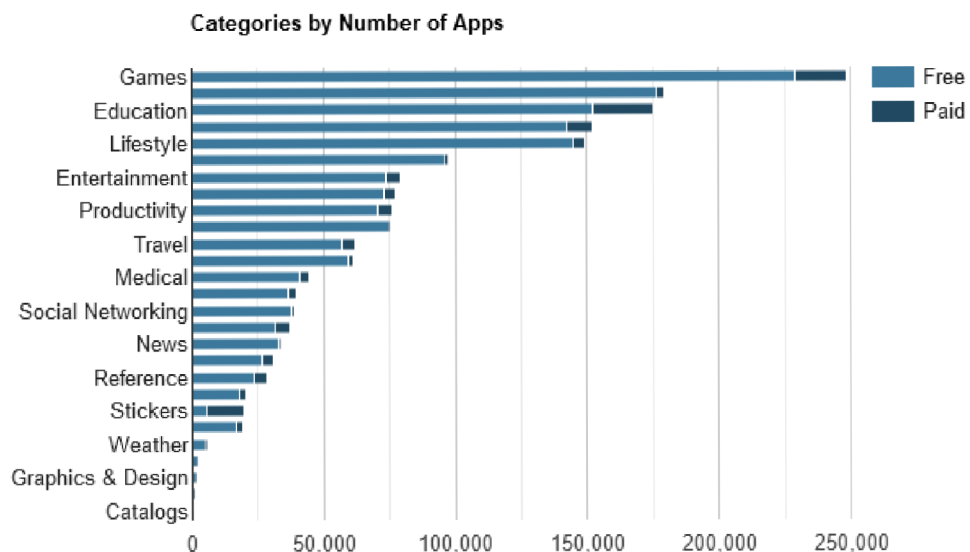
Total categories: 49



**Graf č. 7: Rozdelenie kategórií v Obchode Google Play™ (počet aplikácií)**  
(Zdroj: [27])

## Category Distribution iOS® App Store®

Total categories: 27



**Graf č. 8: Rozdelenie kategórií v iOS® App Store® (počet aplikácií)**  
(Zdroj: [27])

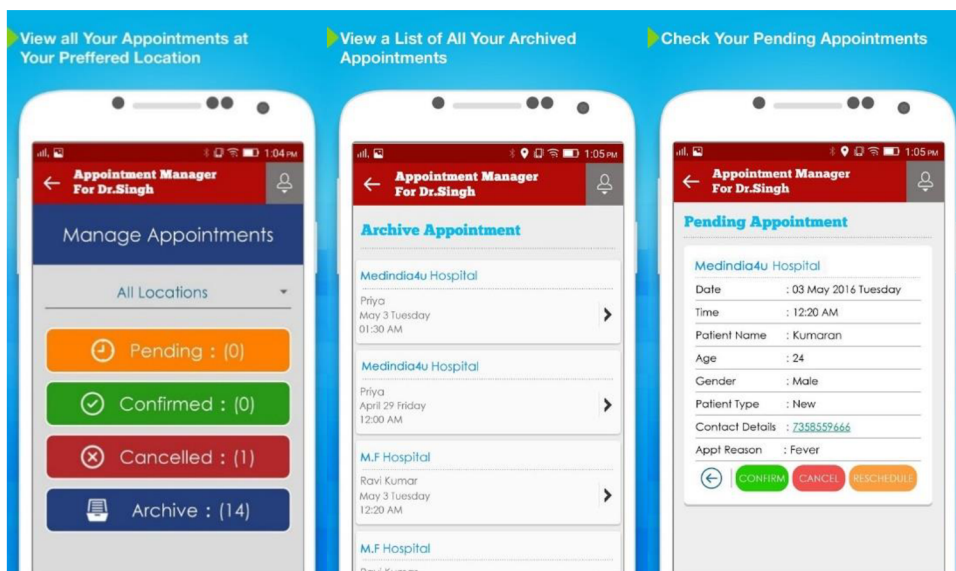


## 2.4 Podobné aplikácie

V tejto podkapitole uvediem a opíšem už existujúce aplikácie, ktoré slúžia na vytváranie termínov u lekárov (osobne, telefonicky, či online). Pri hľadaní podobných aplikácií som sa stretol s niekoľkými typmi aplikácií – tie rozdelím do jednotlivých podkapitol. Ku každému typu uvediem aspoň jednu aplikáciu. Do analýzy zahrniem nie len mobilné aplikácie, ale aj webové stránky určené na vytváranie termínov, pretože pre český a slovenský trh je niekoľko systémov pre vytváranie termínov vytvorených práve v tejto podobe. Pri mobilných aplikáciách budem sledovať najmä aplikácie dostupné z Obchodu Google Play.

### 2.4.1 Organizéry s kalendárom pre lekárov

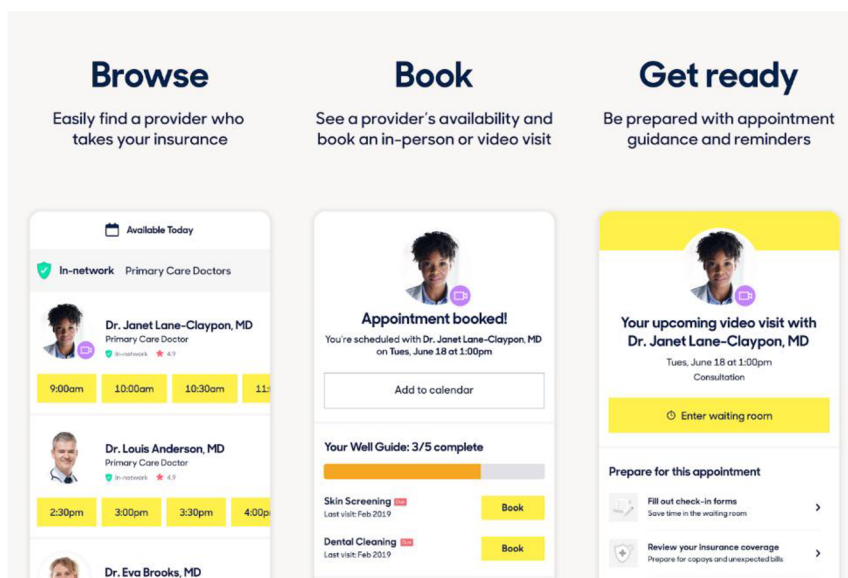
Tento typ aplikácií je vytvorený len pre lekárov. Umožňuje im zapisovať si termíny do kalendára a vytvárať si tak prehľad o termínoch jeho pacientov. Príkladom je mobilná aplikácia *Appointment Manager: Doctors*, ktorá je prispôbena pre lekárov zaregistrovaných v aplikácii *Medindia's Online Appointment*, ktorá spadá pod Medindiu, najväčšiu Indickú zdravotnícku stránku. Lekár si cez túto aplikáciu môže spravovať prichádzajúce žiadosti o termíny od pacientov, ktorí si zarezervovali termín cez *Doctor's Home Page* (priradenej od Medindie) alebo aplikácie typu Find-A-Doctor. Každá zmena odošle aj SMS a e-mail upozornenie lekárovi aj pacientovi [28].



Obrázok č. 18: Obrázky k aplikácii *Appointment Manager: Doctors*  
(Zdroj: [28])

## 2.4.2 Aplikácie typu Find-A-Doctor

Ďalším typom sú aplikácie určené najmä pre pacientov. Tieto aplikácie umožňujú pacientom vyhľadať lekárov a zarezervovať si u nich termín. Príkladom je mobilná aplikácia *Zocdoc*, ktorá pacientom umožňuje vyhľadať najbližších lekárov a špecialistov, ktorých môžu vyhľadať aj na základe hodnotení a či príjmu pacientove poistenie, a zarezervovať si u nich termín buď osobne alebo cez videochat [29].

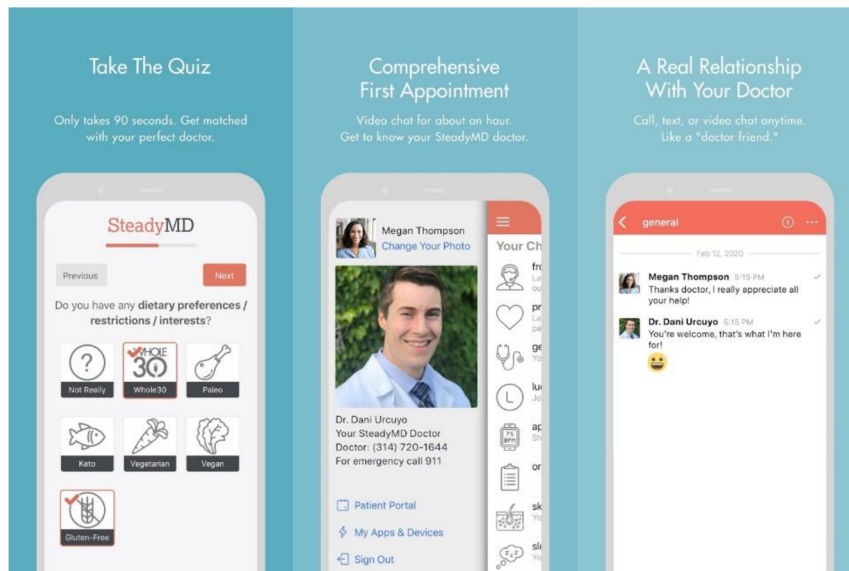


Obrázok č. 19: Obrázky k aplikácii *Zocdoc*  
(Zdroj: [29])

## 2.4.3 Aplikácie zostavené na mieru alebo osobný virtuálny lekár

Existujú aj tzv. *App Buildery*, ktoré umožňujú lekárom aj klinikám vytvoriť univerzálnu mobilnú aplikáciu (zvyčajne existuje aj ekvivalent vo forme webovej stránky) pre ich potreby a bez znalosti programovania. Jedným príkladom je aj nástroj *appypie appmakr*, ktorý umožňuje vytvoriť aj aplikácie pre lekárov. Tvorca vlastnej aplikácie si môže zvoliť vlastný branding, pridať do zoznamu lekárov, ktorí majú záujem byť súčasťou aplikácie, a vybrať z funkcií, ktoré nástroj ponúka, ako sú upozornenia, vytváranie reportov a štatistík, rezervačný systém s in-app platbami, možnosť správ aj video hovorov, posielanie reportov pacientom a mapa s GPS funkciou pre navigáciu ku klinike [30]. Takto vytvorené aplikácie môžu časom ale vytvoriť zhľuk podobných aplikácií s prípadnou nevýhodou, že nie každý lekár bude zaevidovaný v každej z nich.

Jedným príkladom takto vytvorenej aplikácie je *SteadyMD*. Táto aplikácia umožňuje na základe krátkeho kvízu vybrať pacientovi lekára na mieru bez nutnosti opúšťať dom. Vyšetrenia prebiehajú online ako správy alebo video hovor a lekár môže aj odoslať predpísaný recept na lieky do lokálnej lekárne pacienta [31].



**Obrázok č. 20: Obrázky k aplikácii *SteadyMD***  
(Zdroj: [31])

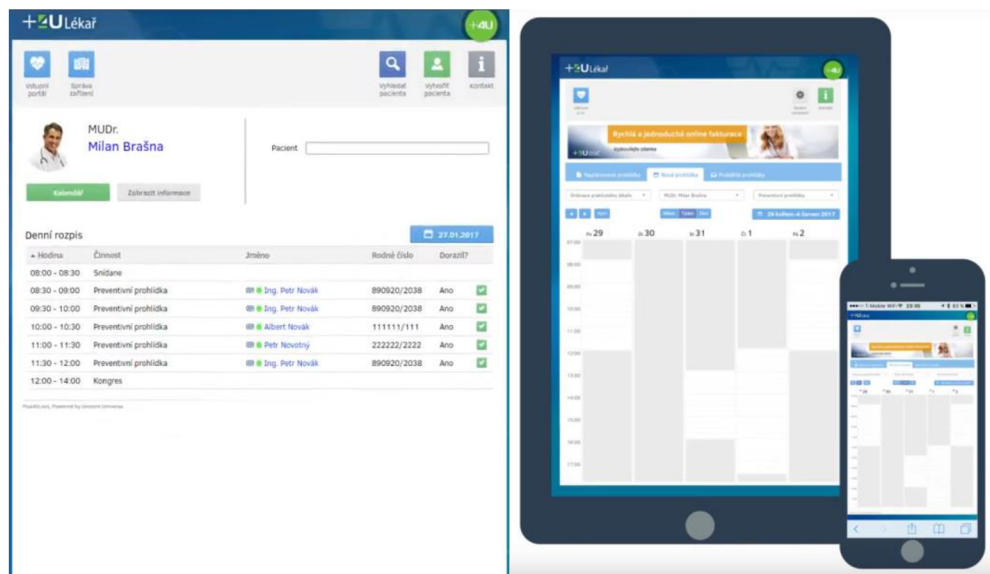
Po preskúmaní funkcií aj ostatných aplikácií uvedených na stránkach nástroja *appypie appmakr* som zistil, že takto vytvorené aplikácie na mieru a mnohé ďalšie aplikácie typu digitálnych osobných virtuálnych lekárov sú určené práve na neosobný kontakt s lekármi pomocou správ a video hovorov, avšak niektoré disponujú aj inými funkciami ako aj vyššie spomínaná aplikácia *SteadyMD*. Príkladom takej aplikácie je napríklad *MDLIVE*, ktorá disponuje aj možnosťou rezervácie okamžitej návštevy lekára na jeho klinike (zvyčajne do 15 minút) a mapou s označenými klinikami dostupných lekárov [32].

#### **2.4.4 Rezervačné systémy cez web**

Pre český a slovenský trh sa mi nepodarilo nájsť priamo mobilnú aplikáciu s rezervačným systémom pre lekárov, našiel som však webové aplikácie alebo stránky, ktoré touto funkcionalitou disponujú.

Pre český trh to je *+4U Lékař*, ktorý ponúka platený rezervačný systém pre lekárov od spoločnosti Plus4U. Sadzba je 499 CZK na mesiac (pričom je potrebné vopred zakúpiť licenciu pre lekára vo výške 5 988 CZK) a tento systém obsahuje denný rozpis

a prehľadný diár, históriu návštev pacientov, upozornenia, správu viacerých ordinácií lekára, karty pacienta, definíciu činností, na ktoré sa môžu pacienti objednať, hromadné odosielanie správ v prípade neprítomnosti a možnosť prihlásenia sa z akéhokoľvek zariadenia disponujúcim internetovým pripojením a webovým prehliadačom. Do tohto systému sa môžu registrovať aj pacienti, ktorí služby rezervačného systému naopak využívajú zadarmo. Po rezervovaní si termínu sa termín zapíše do diára lekára aj pacienta a rôzne úkony majú rôzne farby, takže diár ostáva prehľadný [33]. Takto môžu pacienti pristupovať k rezervačným systémom svojich lekárov alebo špecialistov oveľa efektívnejšie a pohodlnejšie, ak lekári daného pacienta túto službu využívajú.



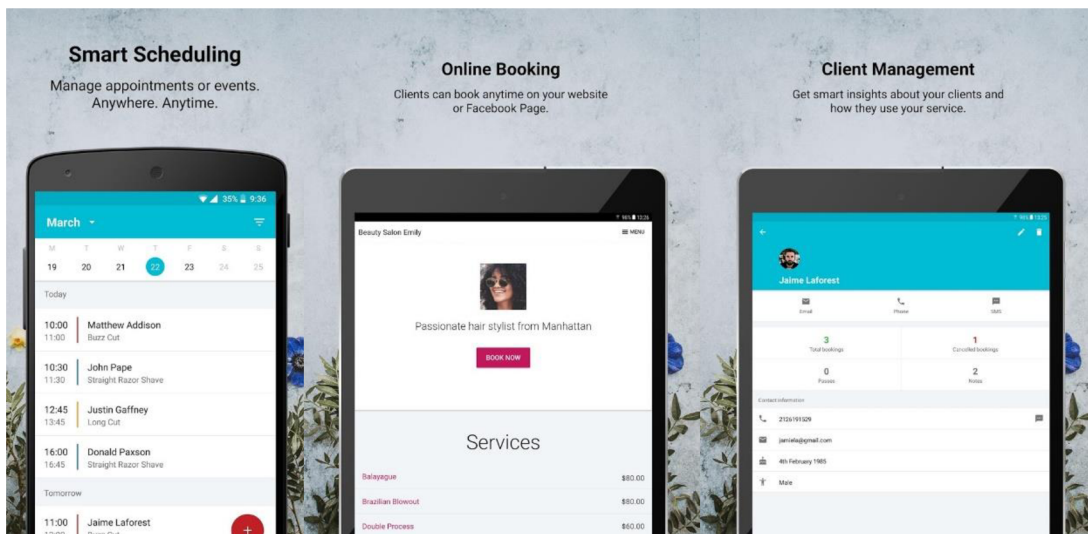
**Obrázok č. 21: Obrázky k rezervačnému systému +4U Lékař**  
(Zdroj: [34])

Pre slovenský trh zase existuje rezervačný systém na webovej stránke *NávštevaLekára.sk*. Tento portál je taktiež ponúkaný pre pacientov zadarmo. Umožňuje objednať sa na najbližší voľný termín alebo na presný deň a čas, kedy to pacienti najviac vyhovuje. Samozrejmosťou sú aj SMS upozornenia [35]. Tak isto ako aj pri vyššie spomenutom rezervačnom systéme pre český trh, aj tento systém ponúka efektívne a pohodlné riešenie pre pacientov, ak ich lekári túto službu využívajú.

## 2.4.5 Univerzálne rezervačné systémy

Tieto systémy môže využívať akýkoľvek podnikateľ bez ohľadu na činnosť podnikania. Jedným takým príkladom je napríklad rezervačný systém *Reservio*, ktorý je možný získať

aj vo forme mobilnej aplikácie. Výhodou tohto rezervačného systému je, že sa dá prispôbiť na mieru pre rôzne typy podnikania (aj pre lekárov). Ďalej tento systém umožňuje pridať rezervačný systém na webové stránky podnikateľa, pomocou ktorého sa môžu klienti rezervovať na voľné termíny, posielat' upozornenia a taktiež aj spravovať informácie o klientoch [36]. Tento typ rezervačných systémov umožňuje klientom rezervovanie na termíny vo viacerých podnikoch (klinika, kaderníctvo, telocvičňa,...) na jednom mieste, avšak pri nedostatočnom diferencovaní služieb pre podniky môže byť tento systém veľmi jednoduchý a nedostačujúci pre potreby daného podniku.



Obrázok č. 22: Obrázky k aplikácii *Reservio Business*  
(Zdroj: [37])

## 2.5 Zhrnutie analýz

Po analýzách jednotlivých trhov a podobných aplikácií som dospel k záveru, že v dnešnej dobe už väčšina ľudí používa smartphony, dokonca aj v rozvojových krajinách (aj vďaka širokému sortimentu smartphonov od lacných zariadení až po vlajkové lode). Využívanie internetu na týchto mobilných zariadeniach sa stalo neodlučiteľnou súčasťou života mnohých ľudí, dokonca sa vyrovnalo aj využívaniu internetu na desktopových riešeniach. Z operačných systémov jednoznačne prevládajú Android a iOS a je výhodné aplikáciu prispôbiť práve pre tieto systémy a distribuovať ju cez ich vlastné digitálne distribučné platformy (Obchod Google Play a iOS App Store).

Návrh pre rezervačný systém som sa rozhodol riešiť ako mobilnú aplikáciu, pretože mobilné aplikácie ponúkajú niekoľko výhod oproti webovým aplikáciám, ako napríklad

schopnosť fungovať aj v offline režime (naposledy aktualizovaný kalendár bude možné vidieť aj bez pripojenia na internet a taktiež používateľ bude dostávať notifikácie o nadchádzajúcich termínoch aj v tomto režime).

Mobilná aplikácia navrhovaná v tejto práci bude využívať niektoré funkcie zahraničných aj domácich medicínskych aplikácií. Momentálne však na trhu (najmä českom a slovenskom) neexistuje riešenie, ktoré by obsahovalo napríklad aj možnosť chatovania v aplikácií, ale taktiež aj slúžilo na rezerváciu termínov do budúcnosti a zároveň bolo dostupné aj ako mobilná aplikácia. V ďalšej kapitole budem navrhovať riešenie mobilnej aplikácie, ktoré by obsahovalo nie len tieto dve funkcie, ale taktiež aj niekoľko ďalších, vďaka ktorým by sa stalo potencionálnym substitútom pre už existujúce riešenia, ale taktiež sa od konkurencie aj odlíšilo (schopnosť aplikácie pracovať s výmennými lístkami).

### **3 VLASTNÉ NÁVRHY RIEŠENIA**

V tejto kapitole sa budem zaoberať návrhom dátovej a funkčnej štruktúry vybranej aplikácie pre medicínsku sféru. Na základe teoretických poznatkov a analýze súčasného stavu som sa rozhodol, ako som už v analýze spomenul, pre návrh mobilnej aplikácie. Dôvodom je fakt, že väčšina ľudí v dnešnej dobe už smartphone vlastní a využíva na ňom natívne mobilné aplikácie, ktoré oproti tým webovým ponúkajú mnoho výhod. V prípade tohto navrhovaného riešenia by mobilná aplikácia benefitovala možnosťou obmedzenej funkcionality v režime offline – notifikácie o nadchádzajúcich termínoch či nutnosti užiť lieky by chodili aj bez pripojenia k internetu vďaka cache dátam uložených na lokálnom úložisku mobilného zariadenia používateľa, a teda by bolo možné vidieť viac informácií aj po otvorení aplikácie, ktorá by si načítala dáta z cache pamäte. Ďalej by aplikácia ťažila z väčšej integrácie s OS a mohla využívať aj iné funkcionality smartphonu, napr. v chatovacom module by bolo možné využiť priamo kameru mobilného zariadenia.

V nasledujúcich podkapitolách uvediem entito-relačný model, dátovú štruktúru vo forme tabuliek, zobrazenie vybraných procesov pomocou DFD a vývojových diagramov a taktiež aj ukážky niektorých funkcionalít aplikácie vo forme obrázkov. Nakoniec výsledný návrh zhodnotím pomocou SWOT analýzy a ekonomického zhodnotenia nákladov na vývoj a distribúciu navrhovanej aplikácie, pričom budem riešiť aj možnú cenovú politiku pre predaj licencií pre lekárov.

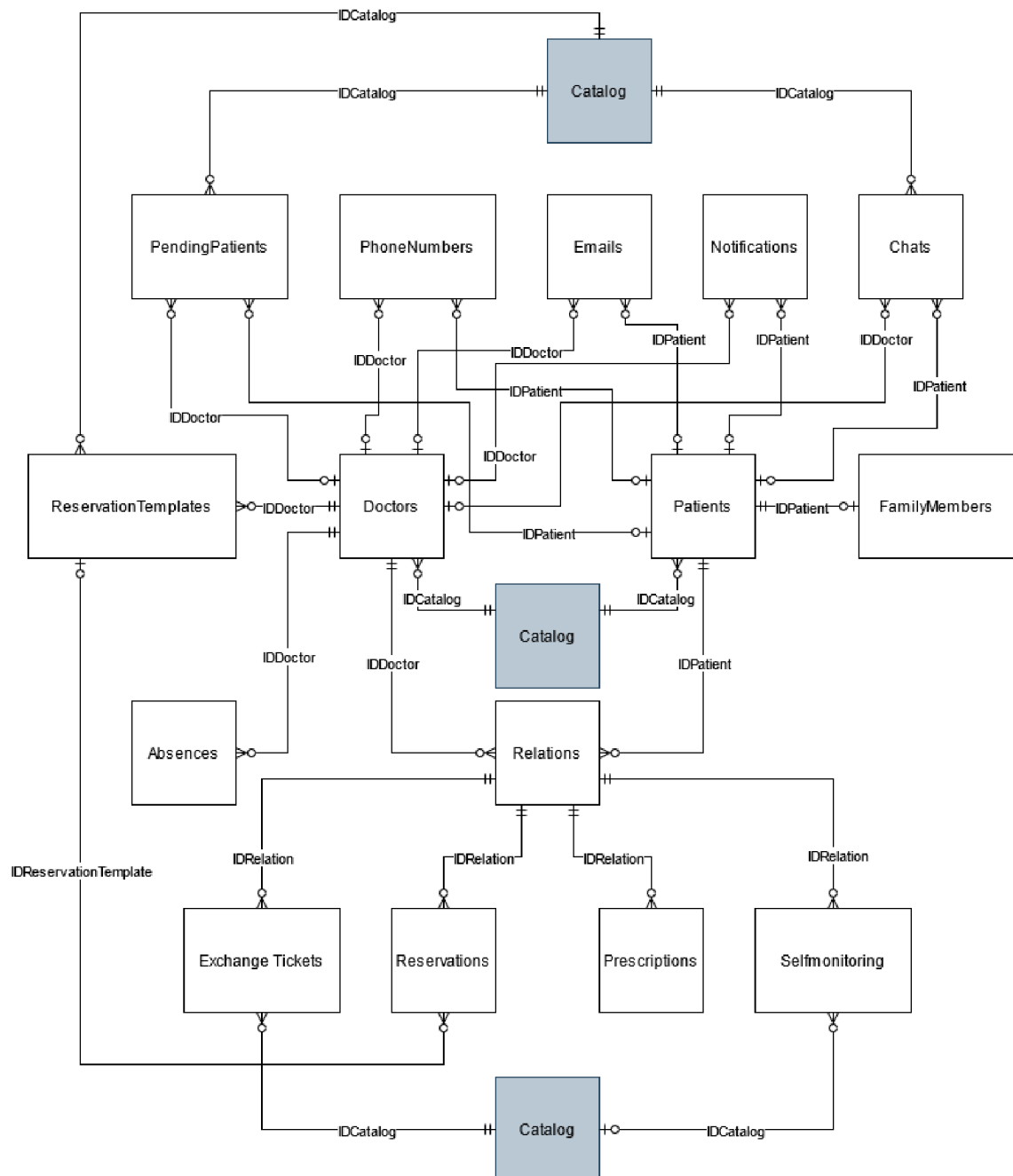
#### **3.1 Dátový model aplikácie**

Dátový model aplikácie znázorním pomocou ER modelu a detailným popisom jednotlivých entít prostredníctvom tabuliek vo formáte pre MS SQL server.

##### **3.1.1 Entito-relačný model**

Na znázornenie entito-relačného modelu riešenej aplikácie som si vybral „inžiniersky štýl“ zobrazenia. Model obsahuje všetky entity, ktoré budú predstavovať jednotlivé tabuľky v dátovom modeli aplikácie, aj väzby medzi nimi. Každá väzba má uvedený cudzí kľúč, ktorý prepája obe entity. Podrobnejší opis k jednotlivým entitám uvediem v nasledovných podkapitolách vzťahujúcich sa k jednotlivým tabuľkám dátového modelu.

Entita Katalóg sa v ER modeli nachádza niekoľkokrát z toho dôvodu, že predstavuje súhrnný číselník, ktorý obsahuje dáta všetkých číselníkov odlišené daným príznamkom v rámci dátového modelu, a preto musel byť vyobrazený niekoľkokrát, aby ho bolo možné priehľadne prepojiť s ostatnými entitami.



**Obrázok č. 23: ER model aplikácie**  
(Zdroj: Vlastné spracovanie)



### 3.1.2 Tabuľky dátového modelu

Jednotlivé entity vyjadrim ako tabuľky pre MS SQL server, pretože navrhovaná aplikácia potrebuje pre svoju funkčnosť ukladať údaje na centralizované miesto, ku ktorého dátam by mali prístup používatelia z hociktorého miesta na zemi. Túto úlohu bude plniť práve spomínaný databázový server (využiť by sa samozrejme mohol aj iný typ databázového servera, napr. Oracle).

Každú entitu/tabuľku databáze zobrazím pomocou tabuľky, ktorá bude obsahovať 4 stĺpce. Prvý stĺpec bude označovať integritné vymedzenie atribútu, druhý názov atribútu, tretí jeho dátový typ a štvrtý poznámku k danému atribútu. V poznámkach budem uvádzať najmä, či atribút musí obsahovať hodnotu (not null) alebo musí byť jedinečný (unique), čo majú jednotlivé hodnoty atribútu znamenať, mernú jednotku alebo, ak je atribút cudzím kľúčom, akú má referenčnú integritu (references <názov tabuľky> (<primárny kľúč v referovanej tabuľke>)). Pojem atribút v nasledujúcich podkapitolách bude označovať stĺpec/pole tabuľky (v nadväznosti na teóriu relačného dátového modelu), nie vlastnosti stĺpca databázovej tabuľky. Všetky riadky týchto tabuliek dohromady budú predstavovať schému relácie (stĺpce databázovej tabuľky).

#### 3.1.2.1 Tabuľka Doctors

Tabuľka Doctors bude obsahovať záznamy o lekároch využívajúcich túto aplikáciu. Každý lekár musí mať priradený typ z číselníka, pretože na základe tohto atribútu sa budú dostupné funkcie aplikácie pre rôzne typy lekárov odlišovať. Odborní lekári si môžu na základe atribútu RequiresExchangeTicket uviesť, či je k požiadaniu o prístup ku kalendáru rezervácií daného lekára potrebný výmenný lístok. Každý lekár si môže taktiež uviesť lekára, ktorý ho bude zastupovať v prípade neprítomnosti (väzba 1:1 v rámci tej istej tabuľky). Samozrejmosťou je aj možnosť uviesť potrebné informácie o lekárovi pre pacientov v podobe *richtextu*, teda okrem textu, do vlastných informácií môžu lekári pridať aj obrázky a odkazy.

**Tabuľka č. 7: Tabuľka Doctors**

(Zdroj: Vlastné spracovanie)

	Názov	Dátový typ	Poznámka
PK	ID	int	Not null
	Username	varchar(20)	Not null, unique
	Password	varchar(255)	Not null
	FirstName	varchar(30)	Not null
	LastName	varchar(30)	Not null
	Degree	varchar(15)	
	ScientificDegree	varchar(15)	
	BirthDate	datetime	Not null
	ReferenceNumber	varchar(10)	Not null
	Address	varchar(30)	Not null
	City	varchar(30)	Not null
FK	State	int	Not null, references Catalog (ID)
FK	DoctorType	int	Not null, references Catalog (ID)
	RequiresExchangeTicket	bit	Not null
	Info	binary	
	ProfilePicture	image	
FK	DeputyDoctor	int	References Doctors (ID)

**3.1.2.2 Tabuľka Patients**

Každý používateľ, ktorý bude aplikáciu využívať, bude reprezentovaný jedným záznamom v tabuľke Patients. O pacientoch sa budú zbierať iba tie najdôležitejšie informácie, ktoré budú potrebné pre operatívnu činnosť aplikácie.

**Tabuľka č. 8: Tabuľka Patients**

(Zdroj: Vlastné spracovanie)

	Názov	Dátový typ	Poznámka
PK	ID	int	Not null
	Username	varchar(20)	Not null, unique
	Password	varchar(255)	Not null
	IdentificationNumber	varchar(10)	Not null
	FirstName	varchar(30)	Not null
	LastName	varchar(30)	Not null
	Degree	varchar(15)	
	ScientificDegree	varchar(15)	
	BirthDate	datetime	Not null
	ReferenceNumber	varchar(10)	Not null
	Address	varchar(30)	Not null
	City	varchar(30)	Not null
FK	State	int	Not null, references Catalog (ID)
FK	HealthInsurance	int	Not null, references Catalog (ID)
	ProfilePicture	image	

### 3.1.2.3 Tabuľka FamilyMembers

Táto tabuľka bude tvoriť rámec funkcie *Členovia rodiny*. Tabuľka bude obsahovať cudzí kľúč do tabuľky Patients, pričom každý takýto pacient bude mať priradené číslo FamilyGroup, ktoré bude pre členov skupiny rovnaké. Novovytvorená skupina bude mať priradené stále najväčšie FamilyGroup číslo v tabuľke navýšené o 1. Atribút Privilege bude následne určovať, aké práva budú mať jednotliví členovia skupiny.

#### Tabuľka č. 9: Tabuľka FamilyMembers

(Zdroj: Vlastné spracovanie)

	Názov	Dátový typ	Poznámka
PK	ID	int	Not null
FK	IDPatient	int	Not null, References Patients (ID)
	FamilyGroup	int	Not null
	Privilege	tinyint	Not null, 0 – admin, 1 – manage others, 2- no privileges

### 3.1.2.4 Tabuľka PhoneNumbers

Nejedná sa o dekompozičnú tabuľku. Aj keď obsahuje cudzie kľúče do tabuliek Doctors a Patients, vyplnená je vždy iba jedna hodnota, ktorá určuje, ktorej entite dané telefónne číslo prislúcha. Dôvodom tejto tabuľky je, že niektorí používatelia si môžu želať dostávať SMS notifikácie alebo mať uvedených viacero kontaktných telefónnych čísel. Tieto čísla môžu pre ostatných rozlíšiť tak, že k nim pridajú popis, o aké tel. číslo sa jedná. Zároveň si môže zvoliť, či chce aby dané číslo bolo pre iných viditeľné (lekár musí mať aspoň jeden záznam s viditeľným telefónnym číslom).

#### Tabuľka č. 10: Tabuľka PhoneNumbers

(Zdroj: Vlastné spracovanie)

	Názov	Dátový typ	Poznámka
PK	ID	int	Not null
FK	IDDoctor	int	References Doctors (ID)
FK	IDPatient	int	References Patients (ID)
	Title	varchar(20)	
	PhoneNumber	varchar(12)	Not null
	Public	bit	Not null

### 3.1.2.5 Tabuľka Emails

Nejedná sa o dekompozičnú tabuľku. Podobne ako to bolo pri tabuľke PhoneNumbers, aj v tomto prípade si používatelia môžu želať dostávať notifikácie alebo mať uvedených viacero e-mailových adries. Tieto e-mailové adresy môžu pre ostatných používateľov

odlíšiť taktiež popisom. Nastavenie viditeľnosti e-mailov pre ostatných nechýba ani v tejto tabuľke.

**Tabuľka č. 11: Tabuľka Emails**

(Zdroj: Vlastné spracovanie)

	Názov	Dátový typ	Poznámka
PK	ID	int	Not null
FK	IDDoctor	int	References Doctors (ID)
FK	IDPatient	int	References Patients (ID)
	Title	varchar(20)	
	Email	varchar(40)	Not null
	Public	bit	Not null

**3.1.2.6 Notifications**

Nejedná sa o dekompozičnú tabuľku. Táto tabuľka vytvára rámec pre notifikácie aplikácie. Každá notifikácia je označená príznakom Type z číselníka, ktorý udáva, ktoré atribúty je potrebné vyplniť pre správnu funkčnosť notifikácie. Príkladom môže byť napr. vytvorenie rezervácie pacientom. Vytvorí sa notifikácia s Type = „Nová žiadosť o termín“, čo aplikácií hovorí, že je potrebné vyplniť atribút FromPatient s ID pacienta, ktorý žiada o termín a atribút IDReservation s ID práve vytvorenej rezervácie. Vďaka tomuto je možné naformátovať vzhľad a odkazy v notifikácii tak, aby sa lekár mohol presmerovať na profil pacienta alebo na detail rezervácie. Obdobne to platí aj pre ostatné Type:

- Type = „Nová žiadosť o registráciu“ => vyplní sa IDPendingPatient
- Type = „Nová žiadosť o prístup ku kalendáru rezervácií termínov“ => vyplní sa FromPatient (aj IDExchangeTicket, ak sa žiadosť poslala lekárovi, ktorý vyžaduje výmenný lístok)
- Type = „Nový termín“ => vyplní sa FromDoctor a IDReservation
- Type = „Zmena stavu žiadosti o prístup ku kalendáru rezervácií“ => vyplní sa FromDoctor
- Type = „Zmena stavu žiadosti o termín“ => vyplní sa FromDoctor a IDReservation
- Type = „Nadchádzajúci termín“ => vyplní sa IDReservation a FromDoctor (do notifikácie pacienta) alebo FromPatient (do notifikácie lekára)
- Type = „Možné oneskorenie termínu“ => vyplní sa FromDoctor a IDReservation

- Type = „Priradený nový výmenný lístok“ => vyplní sa FromDoctor a IDExchangeTicket
- Type = „Nepřítomnosť lekára“ => vyplní sa FromDoctor
- Type = „Pridaný nový selfmonitoring“ => vyplní sa FromPatient a IDSelfmonitoring
- Type = „Nový predpis lieku“ => vyplní sa FromDoctor a IDPrescription
- Type = „Čas užiť lieky“ => vyplní sa IDPrescription
- Type = „Pozvánka k lekárovi“ => vyplní sa FromDoctor

**Tabuľka č. 12: Tabuľka Notifications**

(Zdroj: Vlastné spracovanie)

	Názov	Dátový typ	Poznámka
PK	ID	int	Not null
FK	IDDoctor	int	References Doctors (ID)
FK	IDPatient	int	References Patients (ID)
	Notification	text	Not null
	Delivered	bit	Not null
FK	NotificationType	int	Not null, references Catalog (ID)
	FromDoctor	int	References Doctors (ID)
	FromPatient	int	References Patients (ID)
	IDReservation	int	References Reservations (ID)
	IDPrescription	int	References Prescriptions (ID)
	IDSelfmonitoring	int	References Selfmonitoring (ID)
	IDExchangeTicket	int	References ExchangeTickets (ID)
	IDPendingPatient	int	References PendingUsers (ID)

### 3.1.2.7 Tabuľka Chats

Nejedná sa o dekompozičnú tabuľku. Táto tabuľka predstavuje možné jednoduché riešenie pre chatovací modul aplikácie. Obsahoval by záznamy obsahujúce ID oboch účastníkov komunikácie, správu samotnú a jej stav z pohľadu odosielateľa. Záznam by sa v tabuľke vytvoril až po odoslaní správy s priradením hodnoty pre atribút Status = 0. Jedná sa však len o možné ukladanie správ, vhodnejšie riešenia chatovacieho modulu budem opisovať v tejto práci neskôr.

**Tabuľka č. 13: Tabuľka Chats**

(Zdroj: Vlastné spracovanie)

	Názov	Dátový typ	Poznámka
PK	ID	int	Not null
FK	IDDoctor	int	Not null, references Doctors (ID)
FK	IDPatient	int	Not null, references Patients (ID)
	Message	text	Not null
	Status	tinyint	Not null, 0 – sent, 1 – delivered, 2 - seen

**3.1.2.8 Tabuľka Relations**

Keďže sa medzi entitou lekára a entitou pacienta nachádza väzba N:M, pretože jeden lekár môže vyšetriť N pacientov a jeden pacient môže navštevovať M lekárov, bolo potrebné vytvoriť dekompozíciu. Pre túto dekompozíciu som vytvoril dekompozičnú tabuľku Relations, ktorá bude uchovávať ID lekára aj pacienta, pričom určí, či sa jedná o primárny vzťah (všeobecný lekár alebo zubár pacienta), zastupovanie (zástupný lekár všeobecného lekára alebo zubára pacienta) alebo dočasný vzťah (ak si pacient potrebuje rezervovať termín u odborného lekára). Uchovávať sa, samozrejme, bude aj údaj o aktivnosti daného vzťahu medzi lekárom a pacientom. Záznamy o primárnych vzťahoch a zastupovaní budú mať zvyčajne hodnotu Active = 1, čo sa môže zmeniť na 0 napr., keď sa pacient rozhodne zmeniť lekára. Naopak dočasné vzťahy budú mať hodnotu atribútu Active = 1 iba v prípade, keď odborný lekár povolí prístup ku kalendáru rezervácií danému pacientovi a zmení sa na 0 po ukončení termínu rezervácie, po uplynutí určitého dlhšieho časového intervalu (1 týždeň) alebo po zrušení vzťahu lekárom.

**Tabuľka č. 14: Tabuľka Relations**

(Zdroj: Vlastné spracovanie)

	Názov	Dátový typ	Poznámka
PK	ID	int	Not null
FK	IDDoctor	int	Not null, references Doctors (ID)
FK	IDPatient	int	Not null, references Patients (ID)
	Active	bit	Not null
	RelationType	tinyint	Not null, 0 – primary, 1 – deputy, 2 - temporary

**3.1.2.9 Tabuľka Reservations**

Jedná sa o prvú tabuľku, ktorá obsahuje cudzí kľúč do tabuľky Relations. Aj keď by technicky bolo možné v tejto tabuľke zahrnúť cudzie kľúče do tabuliek Doctors a Patients, rozhodol som sa práve pre túto variantu, pretože sa k informáciám o lekárovi

a pacientovi dá ľahko dostať aj cez záznam v dekompozičnej tabuľke Relations, ktorý už ID pre obe entity obsahuje, vymedzuje vzťah medzi nimi a jeho aktívnosť a taktiež zjednodušuje ER model aplikácie. Záznamy do tejto tabuľky môžu pridávať lekári aj pacienti. Potrebné je hlavne vyplniť hodnotu atribútu From (čas začiatku termínu), atribút To (čas konca termínu) sa v prípade, ak je zvolený IDReservationTemplate (výber zo šablón rezervácií daného lekára) vypočíta sám na základe dĺžky trvania podľa šablóny. Ak sa nevyberie žiadna šablóna, vytvorí sa rezervácia bez šablóny, tejto už je ale potrebné doplniť atribút To ručne. Nechýba ani atribút Status, ktorý určuje stav rezervácie.

**Tabuľka č. 15: Tabuľka Reservations**

(Zdroj: Vlastné spracovanie)

	Názov	Dátový typ	Poznámka
PK	ID	int	Not null
FK	IDRelation	int	Not null, references Relations (ID)
	From	datetime	Not null
	To	datetime	
FK	IDReservationTemplate	int	references ReservationTemplates (ID)
	Status	tinyint	Not null, 0 – new, 1 – approved, 2 - declined
	Note	text	

**3.1.2.10 Tabuľka Absences**

Svoju neprítomnosť môžu lekári zaznamenať v tejto tabuľke. Lekár zadá dĺžku neprítomnosti a popřípade doplní aj oznam pre svojich pacientov.

**Tabuľka č. 16: Tabuľka Absences**

(Zdroj: Vlastné spracovanie)

	Názov	Dátový typ	Poznámka
PK	ID	int	Not null
FK	IDDoctor	int	Not null, references Doctors (ID)
	From	datetime	Not null
	To	datetime	Not null
	Note	text	

**3.1.2.11 Tabuľka ExchangeTickets**

Tabuľka ExchangeTickets je ďalšou, ktorá obsahuje cudzí kľúč do dekompozičnej tabuľky Relations. Záznam do tejto tabuľky pridáva iba lekár, v ktorom určí k akému odbornému lekárovi má pacient zájsť a dôvod poslania pacienta k odbornému lekárovi.

Atribút ApprovedBy sa vyplní po odoslaní a schválení výmenného lístka odborným lekárom.

**Tabuľka č. 17: Tabuľka ExchangeTickets**

(Zdroj: Vlastné spracovanie)

	Názov	Dátový typ	Poznámka
PK	ID	int	Not null
FK	IDRelation	int	Not null, references Relations (ID)
FK	ForDoctorType	int	Not null, references Catalog (ID)
FK	ApprovedBy	int	References Doctors (ID)
	Note	text	

**3.1.2.12 Tabuľka Prescriptions**

Referenčná integrita aj v prípade tejto tabuľky je obdobná tabuľke Reservations. Do tejto tabuľky pridáva záznam iba lekár, pričom uvedie interval v hodinách, v ktorom má pacient dané lieky užiť. Notifikácie o nutnosti užitia liekov začne pacient dostávať, po prvom užití liekov a zadaním hodnoty atribútu StartedToTake (túknutím na tlačidlo *Užil som lieky prvýkrát*). Od daného dátumu sa bude stále v danom intervale posielat' pacientovi notifikácia a vyplní sa atribút LastTaken podľa vzorca StartedToTake + Interval. Po užití všetkých tabletiiek pacient stlačí tlačidlo *Užil som všetky tabletky* a prepíše sa atribút Finished na 1 a notifikácie sa prestanú generovať.

**Tabuľka č. 18: Tabuľka Prescriptions**

(Zdroj: Vlastné spracovanie)

	Názov	Dátový typ	Poznámka
PK	ID	int	Not null
FK	IDRelation	int	Not null, references Relations (ID)
	DrugsName	varchar(30)	Not null
	Interval	tinyint	Not null, [hod]
	StartedToTake	datetime	
	LastTaken	datetime	
	Finished	bit	Not null

**3.1.2.13 Tabuľka PendingPatients**

Táto tabuľka má obdobnú štruktúru ako tabuľka Patients. Slúži na to, že pacient nemôže aplikáciu použiť skôr, ako mu to povolí lekár, takže pri registrácii používateľa sa ním zadané údaje uložia do tejto tabuľky, dokým registráciu neschváli lekár. Do žiadosti je možné zapísať aj poznámku pre lekára. Po schválení sa všetky údaje prepíšu do tabuľky Patients, pričom sa pacientovi vygeneruje username a heslo (je možné ich neskôr



zmeniť), ktoré mu prídu na zvolený spôsob doručenia (email alebo SMS). Ak má záznam v tabuľke PendingPatients vyplnený atribút FromPatient, znamená to, že registrácia bola vykonaná už existujúcim pacientom pridaním nového člena do skupiny daného pacienta. V tomto prípade sa po schválení zapíše aj nový záznam do tabuľky FamilyMembers, kde IDPatient = ID nového pacienta a FamilyGroup = FamilyGroup číslo pacienta, ktorý vytvoril záznam do tabuľky PendingPatients.

#### Tabuľka č. 19: Tabuľka PendingPatients

(Zdroj: Vlastné spracovanie)

	Názov	Dátový typ	Poznámka
PK	ID	int	Not null
	IdentificationNumber	varchar(10)	Not null
	FirstName	varchar(30)	Not null
	LastName	varchar(30)	Not null
	Degree	varchar(15)	
	ScientificDegree	varchar(15)	
	BirthDate	datetime	Not null
	ReferenceNumber	varchar(10)	Not null
	Address	varchar(30)	Not null
	City	varchar(30)	Not null
FK	State	int	Not null , references Catalog (ID)
FK	HealthInsurance	int	Not null, references Catalog (ID)
FK	FromPatient	int	References Patients (ID)
FK	ForDoctor	int	Not null, references Doctors (ID)
	Note	text	

#### 3.1.2.14 Tabuľka Selfmonitoring

Referenčná integrita aj v prípade tejto tabuľky je obdobná tabuľke Reservations. Na rozdiel od ostatných tabuliek využívajúcich danú referenčnú integritu však do tejto tabuľky zapisuje iba pacient. Pacient si zvolí jeden z daných typov selfmonitoringu a na jeho základe sa vyberie, ktoré atribúty je potrebné vyplniť. Nastáť môžu dané prípady:

- SelfmonitoringType = „Meranie váhy a výšky“ => vyplnia sa atribúty Weight, WeightUnit, Height a HeightUnit
- SelfmonitoringType = „Meranie pulzu a tlaku v krvi“ => vyplnia sa atribúty SystolicPressure, DiastolicPressure, PulsePressure a Pulse
- SelfmonitoringType = „Meranie teploty“ => vyplnia sa atribúty Temperature, TemperatureUnit a BodyPart
- SelfmonitoringType = „Meranie glukózy v krvi“ => vyplní sa atribút BloodGlucose

**Tabuľka č. 20: Tabuľka Selfmonitoring**

(Zdroj: Vlastné spracovanie)

	Názov	Dátový typ	Poznámka
PK	ID	int	Not null
FK	IDRelation	int	Not null, references Relations (ID)
FK	SelfmonitoringType	int	Not null, references Catalog (ID)
	Weight	float	
	WeightUnit	int	References Catalog (ID)
	Height	float	
	HeightUnit	int	References Catalog (ID)
	SystolicPressure	int	
	DiastolicPressure	int	
	PulsePressure	int	= SystolicPressure - DiastolicPressure
	Pulse	int	
	Temperature	float	
	TemperatureUnit	int	References Catalog (ID)
FK	BodyPart	int	References Catalog (ID)
	BloodGlucose	float	

### 3.1.2.15 Tabuľka ReservationTemplates

Do tejto tabuľky si môžu jednotliví lekári vytvoriť záznamy o šablónach, ktoré môžu ich pacienti pri rezervácií termínov použiť. Príkladom môže byť napr. TemplateType = „Preventívna prehliadka“ a Duration = 30, teda pacient by si zarezervoval termín na preventívnu prehliadku s dĺžkou trvania 30 minút. Pacienti však môžu využiť iba šablóny, ktoré majú vyplnený atribút TemplateType. Ak v zázname nie je vyplnený atribút TemplateType, znamená to, že sa vyplňujú atribúty DayOfWeek, From a To, ktoré určujú, ktorý deň v týždni a čas od kedy do kedy v rámci daného dňa má byť v kalendári rezervácií termínov daného lekára zablokovaná možnosť online rezervácie termínov pre pacientov. Nevyplnené atribúty From a To znamenajú zablokovanie možnosti rezervácie pre pacientov na celý daný deň.

**Tabuľka č. 21: Tabuľka ReservationTemplates**

(Zdroj: Vlastné spracovanie)

	Názov	Dátový typ	Poznámka
PK	ID	int	Not null
FK	IDDoctor	int	Not null, references Doctors (ID)
FK	TemplateType	int	Not null, references Catalog (ID)
	Duration	int	[min]
	DayOfWeek	tinyint	1-5
	From	int	[min]
	To	int	[min]

**3.1.2.16 Tabuľka Catalog**

Táto tabuľka predstavuje jeden veľký číselník, ktorý obsahuje údaje všetkých ostatných číselníkov, ktoré sa do atribútov ostatných tabuliek zapisujú. Takto nie je potreba pre každý číselník vytvoriť vlastnú tabuľku, ale všetky takéto údaje je možné uložiť na jednom mieste. Tieto údaje sa od seba odlišujú pomocou odlišovacieho príznaku v atribúte Type, čo v podstate predstavuje celé číslo. Odlíšenie záznamov v tabuľke Catalog pre číselníky v ostatných tabuľkách by mohlo byť nasledovné:

- Type = 1 => číselník pre atribút State v tabuľkách Doctors, Patients a PendingPatients
- Type = 2 => číselník pre atribút DoctorType v tabuľke Doctors a atribút ForDoctorType v tabuľke ExchangeTickets
- Type = 3 => číselník pre atribút HealthInsurance v tabuľkách Patients a PendingPatients
- Type = 4 => číselník pre atribút NotificationType v tabuľke Notifications
- Type = 5 => číselník pre atribút SelfmonitoringType v tabuľke Selfmonitoring
- Type = 6 => číselník pre atribút WeightUnit v tabuľke Selfmonitoring
- Type = 7 => číselník pre atribút HeightUnit v tabuľke Selfmonitoring
- Type = 8 => číselník pre atribút TemperatureUnit v tabuľke Selfmonitoring
- Type = 9 => číselník pre atribút BodyPart v tabuľke Selfmonitoring
- Type = 10 => číselník pre atribút TemplateType v tabuľke ReservationTemplates

**Tabuľka č. 22: Tabuľka Catalog**

(Zdroj: Vlastné spracovanie)

	Názov	Dátový typ	Poznámka
PK	ID	int	Not null
	Title	varchar(40)	Not null
	Type	tinyint	Not null, odlišovací príznak pre číselníky

### 3.1.2.17 Dedenie atribútov

Každá databázová tabuľka okrem svojich vlastných atribútov zdedí aj nasledovné atribúty. Tieto atribúty sú určené najmä pre archivačný a analytický účel. Atribúty uchovávajúce ID používateľov nie sú cudzími kľúčmi, iba celočíselná hodnota. V ktorej tabuľke hľadať ID používateľa udáva atribút -ByType. Ak sa jedná o záznamy vytvorené systémom, používa sa ID = 0.

**Tabuľka č. 23: Tabuľka pre dedenie atribútov**

(Zdroj: Vlastné spracovanie)

	Názov	Dátový typ	Poznámka
	Created	datetime	Not null, dátum vytvorenia
	CreatedBy	int	Not null, ID používateľa, ktorý záznam vytvoril
	CreatedByType	tinyint	Not null, 0 – System, 1 – Doctor, 2 - Patient
	Modified	datetime	Not null, dátum úpravy
	ModifiedBy	int	Not null, ID používateľa, ktorý záznam upravil
	ModifiedByType	tinyint	Not null, 0 – System, 1 – Doctor, 2 - Patient

## 3.2 Funkcie aplikácie

V tejto kapitole opíšem funkcie, ktoré bude aplikácia ponúkať pre lekárov aj pacientov. Zameriam sa hlavne na slovný popis funkcií a podmienok, ktorým funkčnosť niektorých funkcií predchádza, pričom ukážem možný vzhlľad niektorých častí aplikácie spojených s týmito funkciami. Nakoniec pomocou diagramov funkčného modelovania zobrazím vybrané procesy.

### 3.2.1 Registrácia a prihlásenie používateľov

Môj nápad pre túto aplikáciu je, že lekár si musí zakúpiť licenciu k tomu, aby mohol danú aplikáciu z funkcie lekára používať. Vydavateľ aplikácie by už registráciu lekára zaistil a jemu by sa po získaní prihlasovacích údajov stačilo do aplikácie prihlásiť. Tieto údaje si môže lekár hocikedy zmeniť, username však musí byť v tabuľke Doctors jedinečný.

Na druhej strane aplikácia by pre pacientov bola zadarmo (registrovať by sa mohli vlastnoručne iba dospelí ľudia). Tí by sa mohli do aplikácie zaregistrovať už na prihlasovacej obrazovke aplikácie. Pacient vyplní povinné políčka na registračnom formulári, pričom musí vyplniť aspoň jeden kontakt pre spôsob doručenia prihlasovacích údajov (telefónne číslo alebo e-mail). Doručeniu prihlasovacích údajov predchádza

schválenie registrácie všeobecným lekárom alebo zubárom pacienta, ktorý túto aplikáciu musí používať. Ten dostane notifikáciu na novú žiadosť o registráciu. Po schválení sa pacientovi pošlú prihlasovacie údaje na zvolený spôsob doručenia a vytvára sa väzba medzi pacientom a lekárom. Taktiež sa vytvára väzba so zástupným lekárom, ktorého má uvedeného vo svojom profile lekár pacienta. Pacient si taktiež môže svoj username a heslo hocikedy zmeniť, avšak stále platí, že username musí byť v tabuľke Patients jedinečný. Samozrejmosťou je prípadná neskoršia zmena všeobecného lekára aj zubára.

Dôvodom nutnosti podania žiadosti o registráciu je zabránenie spamu a prípadnej registrácie kvantu nových používateľov aj bez úmyslu reálnej interakcie s lekármi, čo by mohlo najmä v začiatkoch aplikácie spomaliť jej chod. Pacient môže aplikáciu aj tak plnohodnotne využívať, iba ak sa medzi ním a lekárom nachádza väzba, ináč sa registrovať na termíny ani chatovať s lekárom nemôže. Registrácia na termíny je ale hlavnou funkciou aplikácie, bez ktorej sú ostatné funkcie aplikácie zbytočné. Pre chatovanie s lekárom však stačí, aby mal pacient s lekárom aspoň raz aktívnu väzbu (neplatí pre všeobecných lekárov a pediatrov), čo lekárom umožní komunikovať iba s pacientami, ktorých reálne vyšetřili, a ktorí potrebujú pomôcť alebo poradiť. V prípade, že pacient nemá všeobecného lekára alebo zubára, ktorý by túto aplikáciu využíval, ešte neznamená, že aplikáciu nemôže využiť pre registráciu na termíny u odborných lekárov, ktorí nepotrebujú výmenný lístok. Žiadosť o registráciu je možné poslať aj takémuto lekárovi, je vhodné však do poznámky napísať dôvod odoslania žiadosti práve jemu. V takomto prípade sa po schválení vytvára iba dočasná väzba, ktorá po skončení registrovaného termínu alebo po určitom časovom intervale vyprší (1 týždeň). Aplikácia sa stane naspäť plne funkčnou iba po vytvorení ďalšej aktívnej väzby.

V prípade, že rodič chce spravovať iba termíny svojho potomka, pošle žiadosť o registráciu s vhodnou poznámkou pediatrovi jeho potomka, pričom lekár žiadosť schválí a zvolí nevytváranie väzby (môže iba pediater). Zákonný zástupca následne vytvorí skupinu a do nej pridá nového člena – svojho potomka, ktorého registráciu potvrdí jeho pediater a až teraz sa vytvorí primárna väzba medzi lekárom a pacientom - dieťaťom.



Obrázok č. 24: Návrh prihlasovacieho a registračného formulára  
(Zdroj: Vlastné spracovanie)

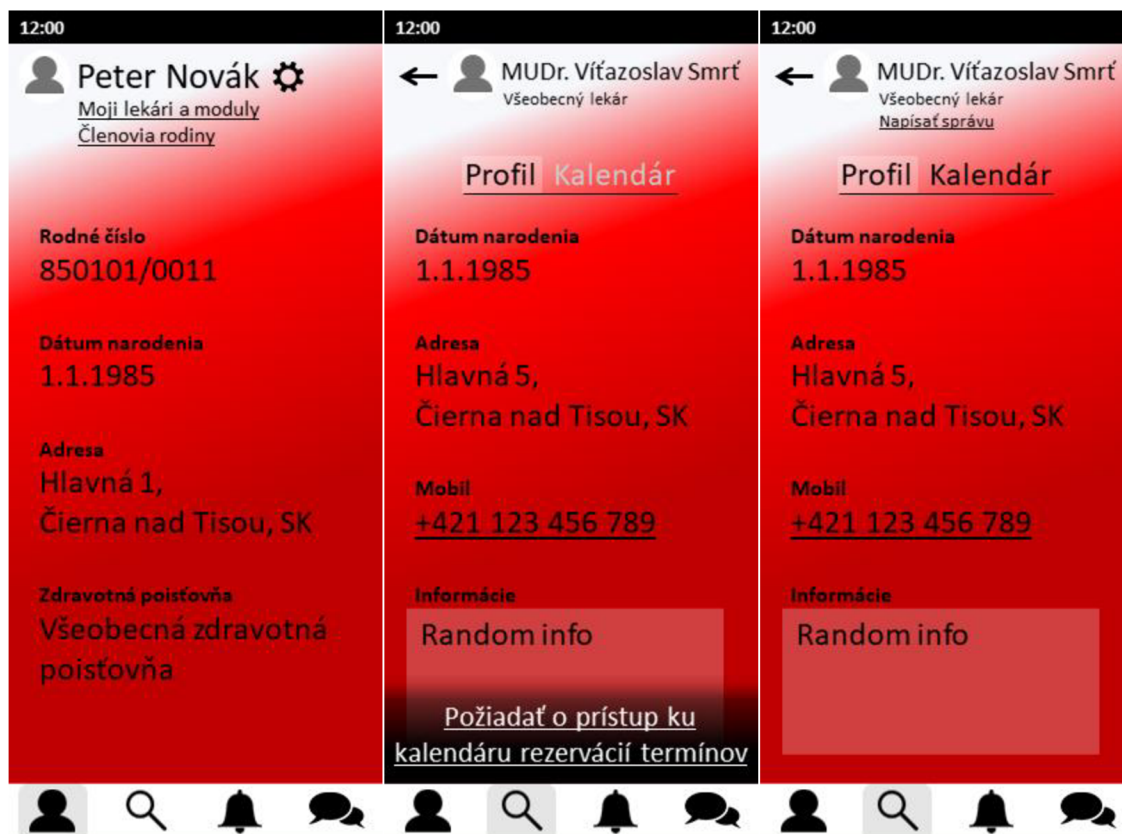
### 3.2.2 Profily používateľov

Každý používateľ aplikácie môže vidieť a spravovať implicitne iba svoj profil. Výnimku tvoria administrátori skupiny *Členovia rodiny* a tí, ktorí v tejto skupine majú nastavené právo *Spravuje ostatných*. V tom prípade môžu vidieť a spravovať aj profily ostatných členov skupiny. Pacienti, ktorí nie sú v rovnakej skupine si nemôžu navzájom vidieť ani detaily v profile, dokonca sa ani vyhľadať v kategórii *Vyhľadávanie* – pre túto aplikáciu by to nemalo žiadny osôh. Výnimkou zase sú pacienti v rovnakej skupine, ktorí môžu vidieť detaily všetkých členov skupiny, aj keď nemajú priradené žiadne práva.

Každý pacient si môže vyhľadať každého lekára využívajúceho túto aplikáciu a taktiež vidieť aj jeho detaily, nemôže však s lekárom, s ktorým nemá väzbu, nijak interagovať.

Lekár môže vidieť detaily všetkých svojich pacientov, s ktorými má väzbu alebo ktorí mu poslali žiadosť o prístup ku kalendáru rezervácií (po ťuknutí na notifikáciu).

Pacient zo svojho profilu môže vidieť svojich lekárov, výmenné lístky, predpisy liekov a záznamy selfmonitoringu na jednom mieste po ťuknutí na odkaz *Moji lekári a moduly*. Obdobne je to aj pre lekárov, oni však v tomto module vidia všetkých svojich pacientov a ich záznamy selfmonitoringu a predpisy liekov aj výmenné lístky, ktoré im daný lekár priradil.



Obrázok č. 25: Návrh vzhľadu profilov jednotlivých používateľov z pohľadu pacienta  
(Zdroj: Vlastné spracovanie)

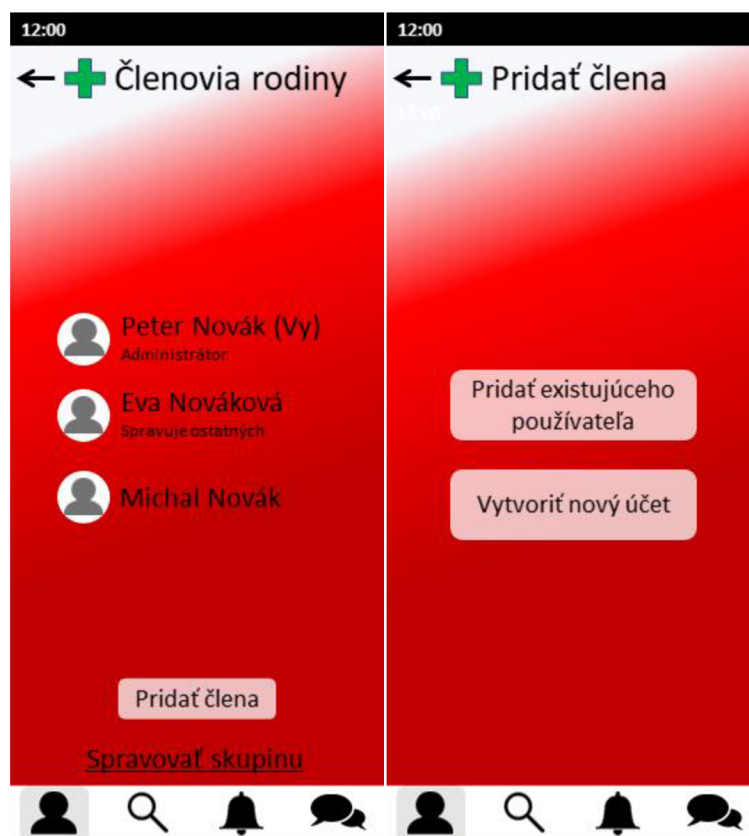
### 3.2.3 Členovia rodiny

Aplikácia bude pre rodiny ponúkať dost' užitočnú funkciu *Členovia rodiny*. Takto môžu členovia skupiny ostatným členom skupiny pridávať a spravovať ich registrácie na termíny, pridávať záznamy o selfmonitoringu, vidieť prepisy liekov a výmenné lístky a prepísať informácie v profile. V podstate môžu pracovať s účtami ostatných ako so svojim. Skupinu zakladá jeden existujúci pacient, pričom sa automaticky stáva administrátorom skupiny (môže spravovať členov skupiny, vymazať skupinu a taktiež pracovať s účtami ostatných členov). Administrátor skupiny môže menovať ďalšieho administrátora alebo iba inému členovi skupiny priradiť právo spravovať účet ostatným

členom skupiny (obe práva môžu dostať iba dospelí pacienti). Členovia bez práv môžu vidieť iba členov skupiny a ich profil, nemôžu s nimi ale ináč interagovať. Iba dospelý pacient môže opustiť skupinu.

Táto funkcia slúži aj pre registráciu potomkov zákonnými zástupcami (deti sa samotné do aplikácie zaregistrovať nemôžu). Pri pridávaní člena si môže administrátor skupiny vybrať, či chce pridať existujúceho pacienta (iba podľa username) alebo zaregistrovať nového člena do aplikácie. Takáto registrácia musí byť taktiež schválená lekárom.

Aj keď má táto funkcia najmä slúžiť pre rodiny, je možné ju využiť aj k vytváraniu skupiny ľudí, ktorí nie sú v príbuzenskom vzťahu. Platí však, že používateľ môže byť členom iba jednej skupiny.



Obrázok č. 26: Návrh funkcie *Členovia rodiny*  
(Zdroj: Vlastné spracovanie)

### 3.2.4 Notifikácie

Tak ako väčšina mobilných aplikácií, aj táto bude posielat' svojim používateľom notifikácie o dôležitých udalostiach. Všetky notifikácie sa budú nachádzať v samostatnej



kategórií *Notifikácie*, kde môže používateľ vidieť všetky notifikácie alebo ich vyfiltrovať podľa kategórie. Každú novú notifikáciu aplikácia zobrazí ako OS notifikáciu, aby používateľ nič nezmeškal, pričom si bude môcť ešte zvoliť, či bude chcieť určité notifikácie dostávať aj iným spôsobom (SMS alebo e-mail). Po ťuknutí na notifikáciu sa aplikácia presmeruje k danej udalosti (napr. ak lekárovi príde žiadosť o termín, po ťuknutí na notifikáciu mu aplikácia zobrazí detaily žiadosti). Niektoré notifikácie budú mať v kategórií *Notifikácie* odkaz aj k inej stránke aplikácie, ak to obsah notifikácie umožňuje (napr. pri notifikácii o vytvorení nového predpisu lieku dáva zmysel, aby primárne notifikácia presmerovala k detailom o predpise lieku, ale taktiež aby sa pacient vedel presmerovať aj do profilu lekára, ktorý mu liek predpísal).

Notifikácie sa budú vytvárať pri týchto udalostiach:

- Nová žiadosť o registráciu (pre lekára od čakajúceho pacienta) => po ťuknutí na notifikáciu presmeruje k detailom žiadosti o registráciu
- Nová žiadosť o prístup ku kalendáru rezervácií termínov (pre lekára od pacienta) => po ťuknutí na notifikáciu presmeruje do profilu pacienta
- Zmena stavu žiadosti o prístup ku kalendáru rezervácií (pre pacienta od lekára) => po ťuknutí na notifikáciu presmeruje do profilu lekára
- Nová žiadosť o termín (pre lekára od pacienta) => po ťuknutí na notifikáciu presmeruje k detailom žiadosti o termín, možné aj do profilu pacienta
- Nový termín (pre pacienta od lekára) => po ťuknutí na notifikáciu presmeruje k detailom o termíne, možné aj do profilu lekára
- Zmena stavu žiadosti o termín (pre pacienta od lekára) => po ťuknutí na notifikáciu presmeruje k detailom o termíne, možné aj do profilu lekára
- Nadchádzajúci termín (pre obe role) => po ťuknutí na notifikáciu presmeruje k detailom o termíne (doručí sa hodinu pred začiatkom termínu), možné aj do profilu opačnej role
- Možné oneskorenie termínu (pre pacienta od lekára) => po ťuknutí na notifikáciu presmeruje k detailom o termíne, možné aj do profilu lekára
- Nový priradený výmenný lístok (pre pacienta od lekára) => po ťuknutí na notifikáciu presmeruje k detailom žiadosti o termín, možné aj do profilu pacienta
- Neprítomnosť lekára (pre pacienta od lekára) => po ťuknutí na notifikáciu presmeruje do profilu lekára

- Pridaný nový selfmonitoring (pre lekára od pacienta) => po ťuknutí na notifikáciu presmeruje k detailom o selfmonitoringu, možné aj do profilu pacienta
- Nový predpis lieku (pre pacienta od lekára) => po ťuknutí na notifikáciu presmeruje k detailom o predpisu lieku, možné aj do profilu lekára
- Čas užiť lieky (pre pacienta) => po ťuknutí na notifikáciu presmeruje k detailom o predpisu lieku
- Pozvánka k lekárovi (pre pacienta od lekára) => po ťuknutí na notifikáciu presmeruje do profilu lekára

OS notifikácie bude používateľ dostávať aj pri prijatí nových správ, tieto notifikácie však budú spravované chatovacím modulom a preto sa nebudú nezobrazovať v kategórií *Notifikácie* a nebude ich možné dostať ani iným spôsobom. Po ťuknutí na notifikáciu by bol používateľ presmerovaný do konverzácie s danou osobou.



Obrázok č. 27: Návrh kategórie *Notifikácie*  
(Zdroj: Vlastné spracovanie)

### 3.2.5 Kalendár rezervácií termínov

Hlavnou funkciou navrhovanej aplikácie je možnosť rezervácie termínov u lekárov pacientami. Predtým ako si môže pacient požiadať o termín, musí mať aktívnu väzbu s daným lekárom. Primárne má aktívnu väzbu nepretržite so svojim všeobecným lekárom (pediatrom u dieťaťa) a zubárom a môže si u nich žiadať o termín hocikedy. Na druhej strane, aj keď má pacient neustále aktívnu väzbu so zástupným lekárom, prístup k jeho kalendáru má iba v prípade, že jeho primárny lekár má zaevidovanú neprítomnosť. Pacient môže mať aktívnu väzbu v jednom momente iba s maximálne dvoma všeobecnými lekármi/pediatrami.

Ak pacient nemá s daným lekárom aktívnu väzbu, nemôže si požiadať o termín. Môže však vidieť v profiloch odborných lekárov momentálnu obsadenosť v kalendári rezervácií termínov daného lekára. Táto možnosť dáva pacientovi voľnosť vo výbere najdostupnejšieho odborného lekára. Pacient však nemôže vidieť obsadenosť v kalendári rezervácií termínov iného všeobecného lekára/pediatra (neplatí pre zubára, pretože sa v skutočnosti môže stať, že zubár pošle pacienta k inému zubárovi napr. pri zarovnávaní zubov strojčekmi).

Pre prístup ku kalendáru rezervácií termínov lekára, je potrebné danému lekárovi poslať žiadosť o prístup (poprípade priložiť výmenný lístok, ak to daný lekár vyžaduje). Lekárovi následne príde notifikácia a môže takúto žiadosť schváliť alebo zamietnuť. O zmene stavu je informovaný notifikáciou aj pacient (v prípade schválenia žiadosti s priloženým výmenným lístkom je výmenný lístok schválený a už sa nedá využiť pre inú žiadosť). Po schválení sa vytvára dočasná aktívna väzba medzi lekárom a pacientom (v prípade, že pacient nemá žiadnu aktívnu väzbu so všeobecným lekárom alebo zubárom, vytvára sa primárna aktívna väzba) a pacient môže začať používať kalendár rezervácií termínov v profile dotyčného lekára alebo začať komunikáciu prostredníctvom chatu. Dočasná väzba zaniká koncom schváleného termínu alebo po uplynutí určitého časového intervalu (1 týždeň, ak nemá pacient zaregistrovaný žiadny schválený termín). Ak pacient nebol nikdy daným lekárom vyšetrený, väzba sa vymaže úplne. V prípade, že pacient využil výmenný lístok pre prístup ku kalendáru rezervácií termínov a nemal u daného lekára, ktorý mu výmenný lístok schválil, žiadny schválený termín sa po pozastavení väzby výmenný lístok znova stáva platným (vymaže sa hodnota atribútu ApprovedBy).

Pacient žiada o termín pomocou tlačidla *Požiadat' o termín*. Následne na formulári vyplní typ termínu a vyberie čas začiatku (dĺžka trvania termínu sa dopočíta podľa šablóny). Pacient taktiež môže k tejto žiadosti pridať poznámku (zapiše sa ako <meno pacienta>:<text poznámky>). V prípade, že pacient nevyberie typ termínu zo šablón lekára, vytvára sa rezervácia bez šablóny, ktorej stačí pridať iba čas začiatku, dĺžku trvania termínu by už doplnil alebo pozmenil lekár podľa potreby. Možné je nastaviť iba rozmedzie času termínu (ak sa doplní aj atribút To), ktorý nezasahuje do času schváleného termínu, môže však zasahovať do času termínu, ktorý čaká na schválenie, v takom prípade môže lekár však schváliť iba jeden z týchto termínov. Nie je možné rezervovať termín ani v čase neprítomnosti lekára. Následne príde lekárovi notifikácia o novej žiadosti, ktorú buď schváli alebo zamietne. V prípade zamietnutia musí lekár doplniť do poznámky dôvod (pridá sa k predošlej poznámke od pacienta v obdobnom formáte). O zmene stavu žiadosti je pacient taktiež informovaný notifikáciou. Obaja následne budú informovaný hodinu pred začiatkom termínu o tejto skutočnosti.

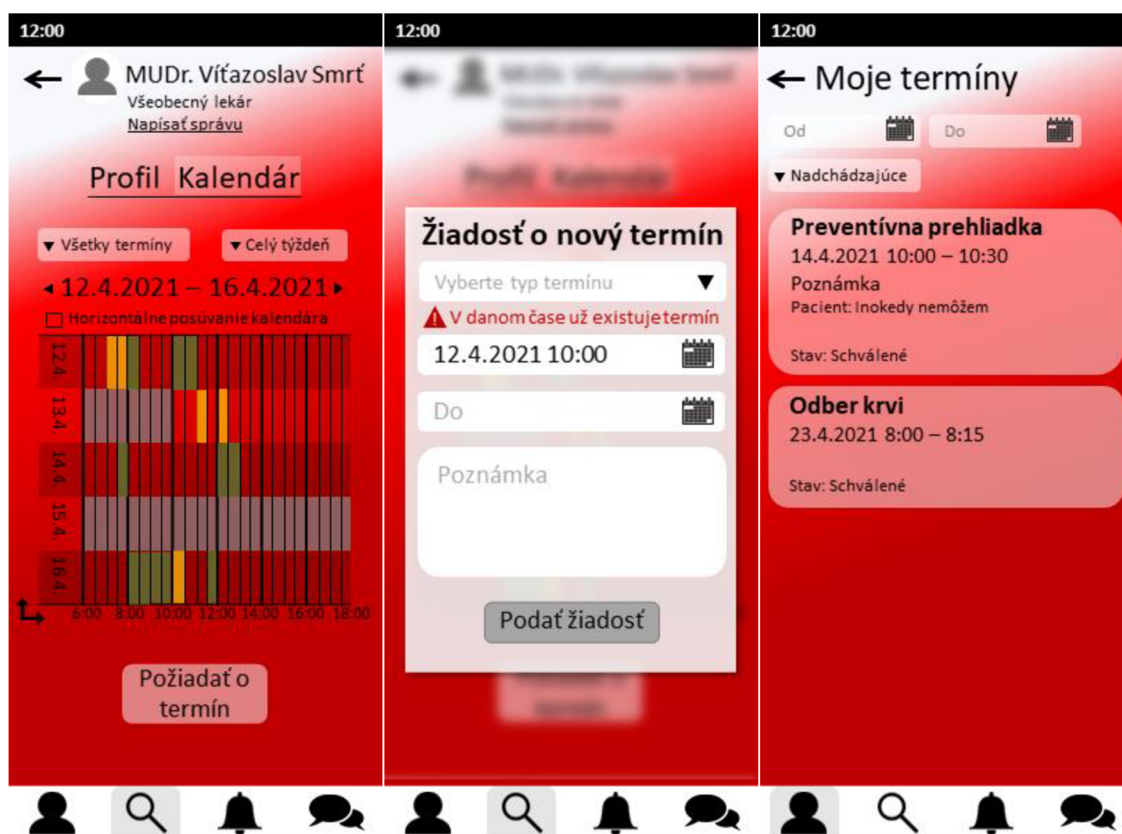
Táto funkcia má lekárovi zabezpečiť aj prehľad o nadchádzajúcich termínoch. Lekár teda môže, narozdiel od pacienta, ktorý musí o termín požiadať, zadať určitému pacientovi schválený termín rovno do kalendára rezervácií termínov v prípade, že si o tento termín pacient požiadal osobne, telefonicky alebo sa dohodli o navštevujúcom termíne (aj pre už neaktívne väzby). O termín takýmto spôsobom si môže zažiadať taktiež aj pacient, ktorý túto aplikáciu nevyužíva. Pre tento prípad bude v tabuľke Patients vopred pripravený pacient s ID = 0, čo bude naznačovať, že ide o pacienta bez registrácie v aplikácii.

Nie je zvykom, aby lekár zapisoval pacientovi termín bez vedomia pacienta. Môže sa však stať, že lekár potrebuje pacienta určitým spôsobom vyšetriť (napr. pacient neprišiel na povinnú preventívnu prehliadku). Preto táto aplikácia taktiež ponúka možnosť poslať pacientovi pozvánku na určitý typ vyšetrenia, na základe ktorého by si mal pacient sám zažiadať o termín alebo sa s lekárom prostredníctvom chatu dohodnúť na vhodnom termíne.

Rezervácie sa nemusia stále týkať iba osobných návštev lekára. Keďže táto aplikácia zahŕňa aj chatovací modul, lekári môžu ponúkať aj online porady pre svojich pacientov. Jednoducho by si pacient zažiadal o termín typu online schôdza a v daný čas by sa s lekárom stretli online prostredníctvom videohovoru.

V reálnom svete sa môže hocikedy stať, že sa musí určitý termín predĺžiť alebo sa k lekárovi dostane naliehavý neodkladný prípad a všetky ostatné termíny v daný deň budú musieť začať s oneskorením. V takomto prípade môže lekár využiť možnosti upozorniť všetkých pacientov jedným ťuknutím na tlačidlo *Upozorniť na oneskorenie*, ktorí majú v daný deň od daného času zarezervované termíny, že sa ich termíny začnú s určitým oneskorením, ktoré lekár odhaduje.

Pre lepší prehľad v termínoch bude aplikácia ponúkať možnosť filtrácie termínov podľa stavu a času buď priamo v kalendári alebo kategórii (*Moje*) *Rezervácie* po presmerovaní z odkazu *Moji lekári/pacienti a moduly* v profile používateľa. Pacienti môžu v kalendári lekárov vidieť iba schválené termíny (zelená) a termíny čakajúce na schválenie (oranžová), zatiaľ čo lekár môže vidieť aj nim zamietnuté termíny (fialová). V kalendári bude taktiež možnosť zobrazenia termínov v celom týždni (s prehodením osí času) alebo iba v jednotlivom dni pre lepšiu prehľadnosť. Pri výbere *Celý týždeň* sa bude ponúkať aj možnosť povoliť horizontálne posúvanie pre horizontálne rozťahnutie kalendára, aby sa doňho zmestili aj informácie o termínoch (pacienti uvidia iba trvanie termínu).



Obrázok č. 28: Návrh kalendára rezervácií termínov  
(Zdroj: Vlastné spracovanie)

### 3.2.6 Evidencia neprítomnosti

V prípade, že lekár bude určité dni neprítomný, môže si v aplikácii zaevidovať neprítomnosť. Vo formulári lekár vyplní trvanie neprítomnosti od a do, pričom môže zanechať svojim pacientom odkaz, ktorý sa im v čase neprítomnosti lekára bude zobrazovať v profile lekára. Po zaevidovaní neprítomnosti príde všetkým pacientom daného lekára notifikácia o tejto skutočnosti. V čase trvania neprítomnosti lekára je v jeho kalendári rezervácií termínov dané rozmedzie času označené sivou farbou (nemožno rezervovať termín v daný čas). Pre toto isté rozmedzie času je ale pre jeho pacientov umožnené rezervovanie termínov u zastupujúceho lekára, pričom pacienti môžu vidieť v kalendári zastupujúceho lekára už existujúce schválené a neschválené termíny taktiež iba pre toto rozmedzie času.

### 3.2.7 Výmenné lístky

Hlavnou výhodou navrhovanej aplikácie oproti konkurencii by bola schopnosť pracovať s výmennými lístkami. Lekár vypíše pacientovi výmenný lístok s poznámkou pre určitý typ lekára, ku ktorému má pacient zájsť na vyšetrenie. Pacientovi následne príde notifikácia o tejto skutočnosti. Následne pacient môže požiadať o prístup lekára daného typu k jeho kalendáru rezervácií termínov. O tejto skutočnosti bude lekár taktiež notifikovaný. Pri schválení žiadosti sa vyplní aj atribút `ApprovedBy` s ID schvaľujúceho lekára v zázname výmenného lístka, aby sa nemohol použiť znova pre inú žiadosť o prístup ku kalendáru rezervácií termínov a vytvára sa aj dočasná väzba medzi daným lekárom a pacientom.

Ak sa registrácia termínu v určitom časovom intervale od schválenia prístupu ku kalendáru rezervácií termínov neuskutoční (1 týždeň), tak sa zo záznamu výmenného lístka odstráni atribút `ApprovedBy`, aby sa dal výmenný lístok znova použiť a pacient tak neprišiel o možnosť vyšetrenia u daného typu lekára.

Každý pacient môže po ťuknutí na odkaz *Moji lekári a moduly* v svojom profile v kategórii *Moje výmenné lístky* vidieť všetky výmenné lístky, ktoré boli danému pacientovi priradené. Tie si následne môže vyfiltrovať podľa stavu a času. Lekári môžu taktiež vidieť a vyfiltrovať v obdobnom module pre lekárov všetky výmenné lístky, ktoré svojim pacientom vypísali.

### 3.2.8 Predpis liekov

Prvotný nápad na túto funkciu je, že lekár predpisuje lieky svojim pacientom a teda, iba on môže zadať, ktoré lieky a v akom intervale ich pacient musí užívať. Pacientovi príde notifikácia o tom, že mu lekár predpísal užívanie liekov v určitých intervaloch. Pacient následne po prvom užití liekov stlačí tlačidlo *Užil som lieky* a od tejto chvíle budú pacientovi v danom intervale chodiť notifikácie o nutnosti užitia daného lieku, dokým nestlačí tlačidlo *Užil som poslednú tabletku*. V prípade, že pacient zabudol alebo nemohol v danom momente užiť liek a chce zmeniť čas, od ktorého sa mu začne odpočítavať nutnosť užitia lieku, môže tak urobiť stlačením tlačidla *Teraz som užil liek*.

Tak ako tomu bolo aj pri rezerváciách, aj táto funkcia bude ponúkať prehľad o priradených predpisoch liekov pre pacientov a vypísaných predpisoch liekov pre lekárov cez odkaz v profile *Moji lekári/pacienti a moduly* v kategórii *Predpisy liekov*. Chýbať nebude ani možnosť filtrácie podľa času, stavu užívania, používateľa (pre lekára filtrovanie podľa pacienta a naopak) a zobrazenie detailov o predpisoch, kde sa budú vyššie spomínané tlačidlá spojené s užívaním liekov nachádzať.

Pri tejto funkcii je ale potrebné myslieť aj na to, že nie každý lekár sa rozhodne pre používanie navrhovanej aplikácie a teda predpis lieku nie je možné do aplikácie zadať. V tomto ohľade by sa napríklad dal rozšíriť dátový model o ďalšiu tabuľku, ktorá by mala obdobnú štruktúru ako tabuľka Prescriptions, rozdiel by bol však v referenčnej integrite - cudzí kľúč by bol do tabuľky Patients. Takto by si pacient mohol zadať predpisy liekov aj sám. Ďalšou možnosťou by mohlo byť aj rozšírenie existujúcej tabuľky Prescriptions o cudzí kľúč do tabuľky Patients.

### 3.2.9 Selfmonitoring

V čase choroby pacienta alebo iných životných udalostiach a príhodách môže lekár požadovať od svojich pacientov informácie o určitých telesných funkciách. Na toto bude slúžiť funkcia *Selfmonitoring*, kde môžu pacienti dané informácie v pár ťuknutiach odoslať svojmu lekárovi (pričom už pri zadávaní uvidí pacient porovnanie zadaných hodnôt s tými odporúčanými). Pacient nájde daný modul cez odkaz *Moji lekári a moduly*. Tam založí nový záznam selfmonitoringu, kde musí zadať typ selfmonitoringu, podľa čoho sa mu budú meniť políčka, ktoré je potrebné vyplniť. Jedná sa o tieto typy:

- Typ „Meranie váhy a výšky“ – pacient vyplní nameranú výšku a váhu, zvolenie merných jednotiek dáva zmysel podľa štandardu v danom regióne
- Typ „Meranie pulzu a tlaku v krvi“ – pacient vyplní nameraný systolický a diastolický tlak, podľa čoho sa automaticky dopočíta tlaková amplitúda a ďalej vyplní aj pulz
- Typ „Meranie teploty“ – pacient vyplní nameranú teplotu a časť tela, kde teplotu nameral. Zvolenie merných jednotiek dáva zmysel podľa štandardu v danom regióne
- Typ „Meranie glukózy v krvi“ – pacient vyplní nameranú glukózu v krvi

Po uložení záznamu sa lekárovi odošle notifikácia o tejto skutočnosti. Týmto spôsobom sa uľahčí a urýchli práca lekára, keďže nie je potrebné k danému úkonu využiť chat a všetky záznamy o selfmonitoringu má každý lekár uchované na jednom mieste, vďaka čomu sa vynechá organizácia nazbieraných informácií od pacientov.

Ako tomu bolo aj pri rezerváciách, aj tento modul ponúka obdobné filtrovanie záznamov pre jednotlivých používateľov (okrem stavu) a k tomuto modulu sa dá taktiež dostať cez odkaz v profile *Moji lekári/pacienti a moduly* v kategórii *Selfmonitoring*.



Obrázok č. 29: Návrh funkcie *Selfmonitoring*  
(Zdroj: Vlastné spracovanie)



### 3.2.10 Šablóny rezervácií

Táto funkcia umožňuje lekárom vytvárať šablóny pre lekárov, ktorými sa bude riadiť ich kalendár rezervácií termínov a ktoré vytvárajú prehľadnejšie a jednoduchšie podávanie žiadostí o termín pre pacientov. Lekár má na výber z dvoch typov šablón. Prvý typ je určený pre pacientov. Lekár vytvorí šablónu, v ktorej vyberie typ vyšetrenia a nastaví mu dĺžku trvania. Tieto šablóny sa následne zobrazujú pacientom pri vytváraní rezervácií u daného lekára v číselníku *Typ termínu* a podľa nich sa dopočítava atribút *To*. Druhý typ šablóny vytvára obmedzenie pri rezervácií termínov pacientami. Lekár vyplní deň v týždni a čas od a dokedy si praje, aby si pacienti nemohli rezervovať termín, čo sa následne zobrazí ako šedá časť v kalendári rezervácií termínov daného lekára. Tento typ šablóny je užitočný najmä pre prípad, keď si lekár potrebuje rezervovať čas pre pacientov, ktorí aplikáciu nevyužívajú a prichádzajú na vyšetrenia v náhodný čas (toto obmedzenie však neplatí, ak rezerváciu termínu vytvára samotný lekár).

Prehľad o vytvorených šablónach nájde lekár v odkaze v svojom profile *Moji pacienti a moduly* v kategórii *Moje šablóny rezervácií*, kde si záznamy môže filtrovať podľa typu šablóny.

### 3.2.11 Chatovací modul

Tento modul by fungoval obdobne ako väčšina existujúcich chatovacích riešení. Lekár alebo pacient by cez profil druhého používateľa bol cez odkaz *Napísať správu* presmerovaný na konverzáciu s daným používateľom, kde môže napísať správu, poslať fotografiu, video alebo hlasovú správu. Z začať konverzáciu môže používateľ aj vyhľadáním druhého používateľa prostredníctvom vyhľadávania v chatovacom module. Po prvom odoslaní správy sa oboj používateľom bude dané vlákno konverzácie naďalej už zobrazovať na hlavnej stránke chatovacieho modulu. Okrem iného bude tento modul ponúkať aj možnosť videohovoru (zачať hovor bude môcť iba lekár). Samozrejmosťou bude aj vidieť stav správy. Pri odoslaní správy bude mať nová správa stav *odosiela sa*, pričom pri uložení správy do databáze sa stav prehodí do *odoslané*. Pri notifikovaní druhého používateľa (ak je pripojený k internetu) sa stav preklopí do *doručené* a následne pri otvorení správy druhou stranou sa stav správy mení na *videné*.

Účel tohto modulu je v uľahčení konverzácie medzi lekárom a pacientom (konverzácia medzi rovnakým typom používateľov nie je možná). Pacient môže taktiež kontaktovať lekára týmto spôsobom iba v prípade, že s ním má aktívnu väzbu. Obdobne to platí aj pre lekára.

V tejto práci je navrhnuté najjednoduchšie riešenie chatovacieho modulu, ktoré je zahrnuté v dátovom modeli aplikácie. Samotný chatovací modul je ale komplexné riešenie, od ktorého sa vyžaduje dochvilnosť, spoľahlivosť a bezpečnosť. Táto aplikácia by ťažila aj s možnosťou videohovoru. Preto je možné, že by tento modul bol dodaný treťou stranou, kde by vybraná firma dodala, spravovala a udržiavala dané riešenie v podobe pluginu (rozšírenia) pre navrhovanú aplikáciu.



**Obrázok č. 30: Návrh chatovacieho modulu**  
(Zdroj: Vlastné spracovanie)

### 3.2.12 Prepojenie s konkurenčnými aplikáciami

V reálnom svete je viac než isté, že nie všetci lekári budú využívať len jednu aplikáciu na správu rezervácií termínov. Už v súčasnosti existuje niekoľko alternatív aplikácií pre daný účel, z ktorých si lekári môžu vybrať. Táto skutočnosť môže pacientov nútiť

využívať viacero aplikácií v prípade, že lekári, ktorých navštevujú používajú rôzne aplikácie pre rezerváciu termínov.

V tomto ohľade by bolo výhodné, ak by sa po vzájomnej dohode a v spolupráci s ostatnými vydavateľmi konkurenčných aplikácií vytvorili pluginy pre tieto aplikácie, ktoré by umožňovali vyhľadávanie lekárov aj v rámci konkurenčných aplikácií, požiadať o prístup a vidieť obsadenosť ich kalendárov rezervácií termínov a v neposlednom rade rezervovať si u daného lekára z konkurenčnej aplikácie termín. Spolupráci na tomto rozšírení by však predchádzala ochota ostatných vydavateľov pristúpiť k prepojeniu daných aplikácií.

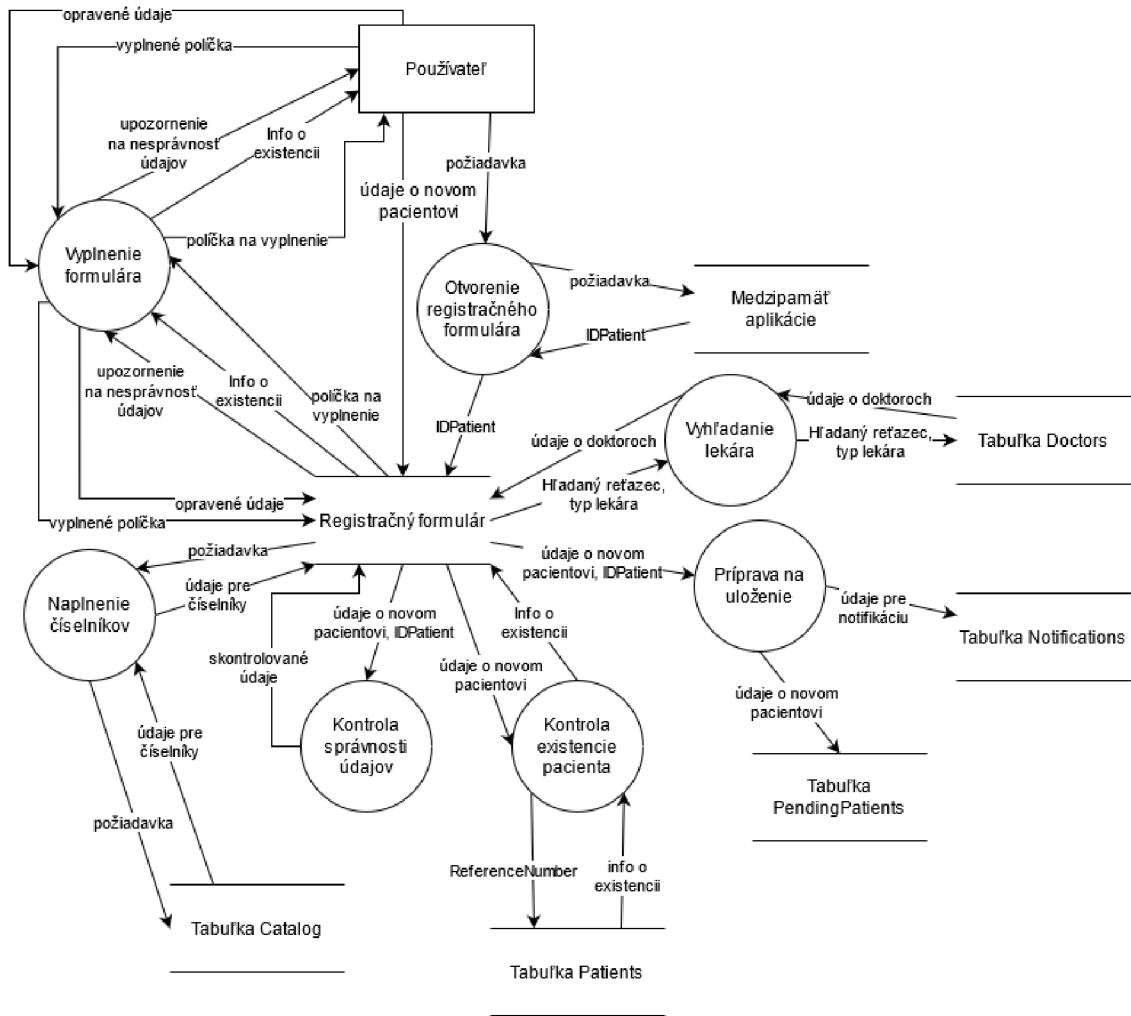
### **3.2.13 Fungovanie aplikácie v offline režime**

Funkcie spomenuté v predošlých kapitolách by fungovali podľa popisu iba v prípade, žeby bol používateľ pripojený k internetu. Avšak, keďže aplikácia je mobilná, môže si vo svojej vyrovnávacej pamäti uchovávať historické dáta (napr. až 1 mesiac staré). Takto by aplikácia umožnila aj bez prístupu k internetu zobraziť profil pacienta/lekára, jeho notifikácie, správy a dáta v odkaze *Moji lekári/pacienti a moduly* v takom stave, v akom boli načítané pri poslednom zapnutí jednotlivých modulov s pripojením na internet. Táto funkcia však umožňuje aplikácii uchovávať aj dáta o tom, kedy bude mať používateľ nadchádzajúce termíny alebo kedy má užiť lieky (v tomto prípade by však nešlo zmeniť čas užívania pri neskoršom užití liekov) a upozorniť ho na to, aj keď nie je pripojený k internetu.

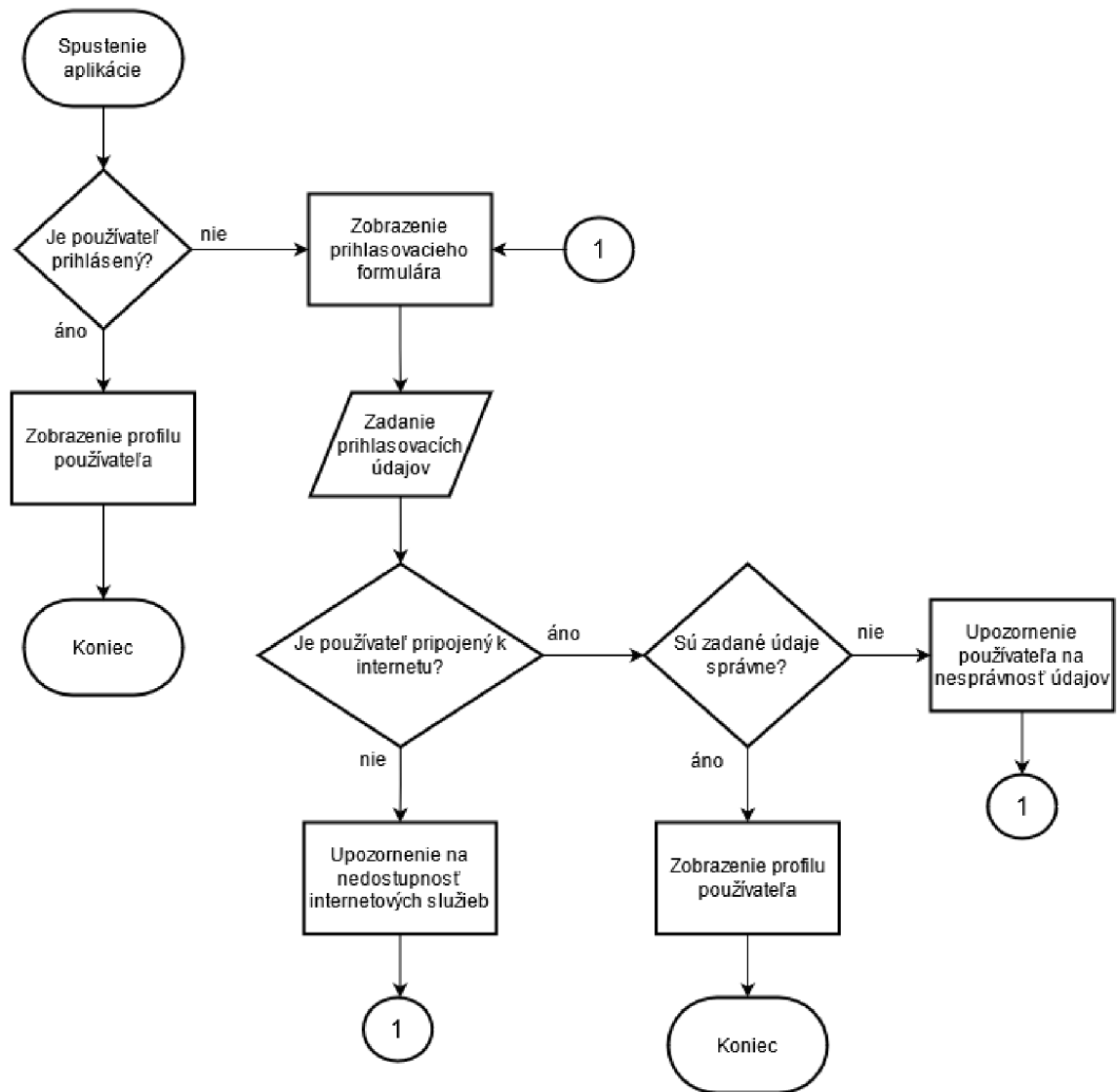
### **3.2.14 Diagramové zobrazenie vybraných procesov**

Na zobrazenie procesov som sa rozhodol pre diagramy funkčného modelovania – DFD a vývojové diagramy. Vybral som si 5 procesov, ktoré plnia najdôležitejšie úlohy aplikácie. Zobrazené budú nasledovnými diagramami:

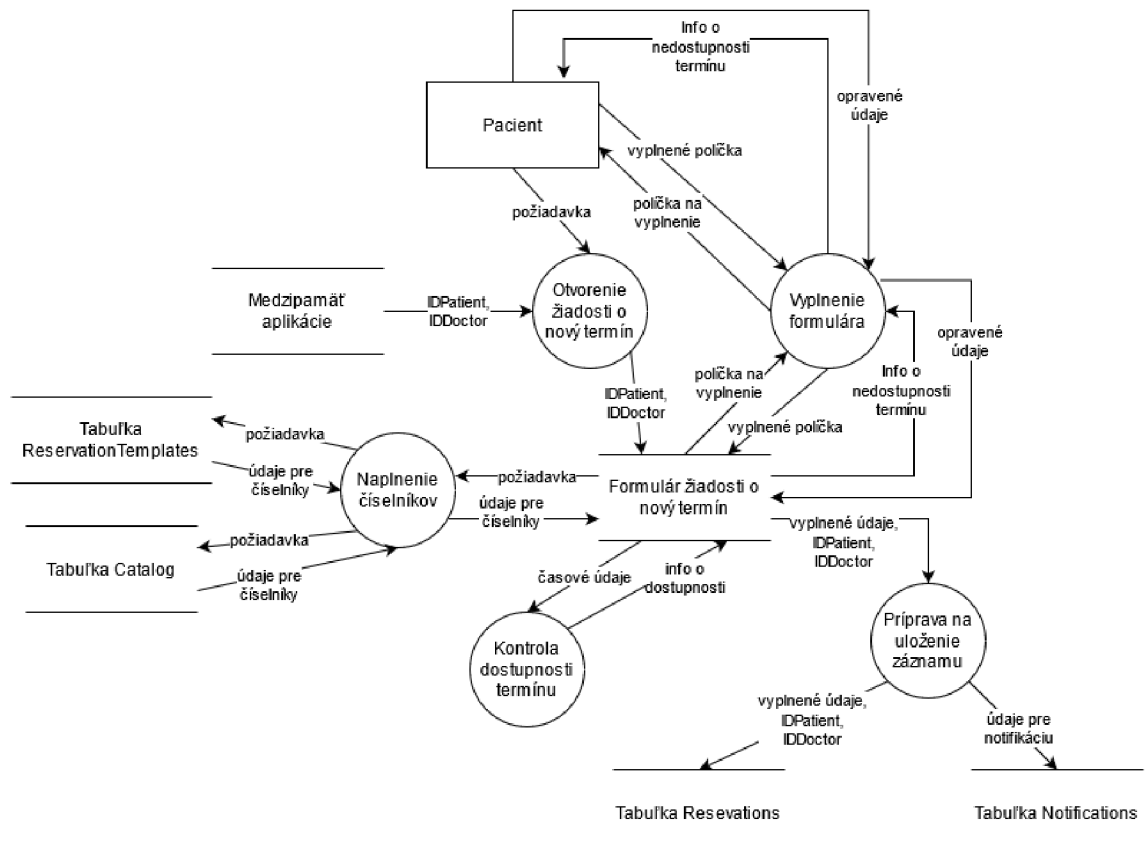
- Registrácia používateľa – DFD
- Prihlásenie používateľa – Vývojový diagram
- Žiadosť o termín – DFD
- Schvaľovanie termínov – Vývojový diagram
- Podanie žiadosti o prístup ku kalendáru rezervácií termínov – Vývojový diagram



**Obrázok č. 31: Registrácia používateľa - DFD**  
(Zdroj: Vlastné spracovanie)



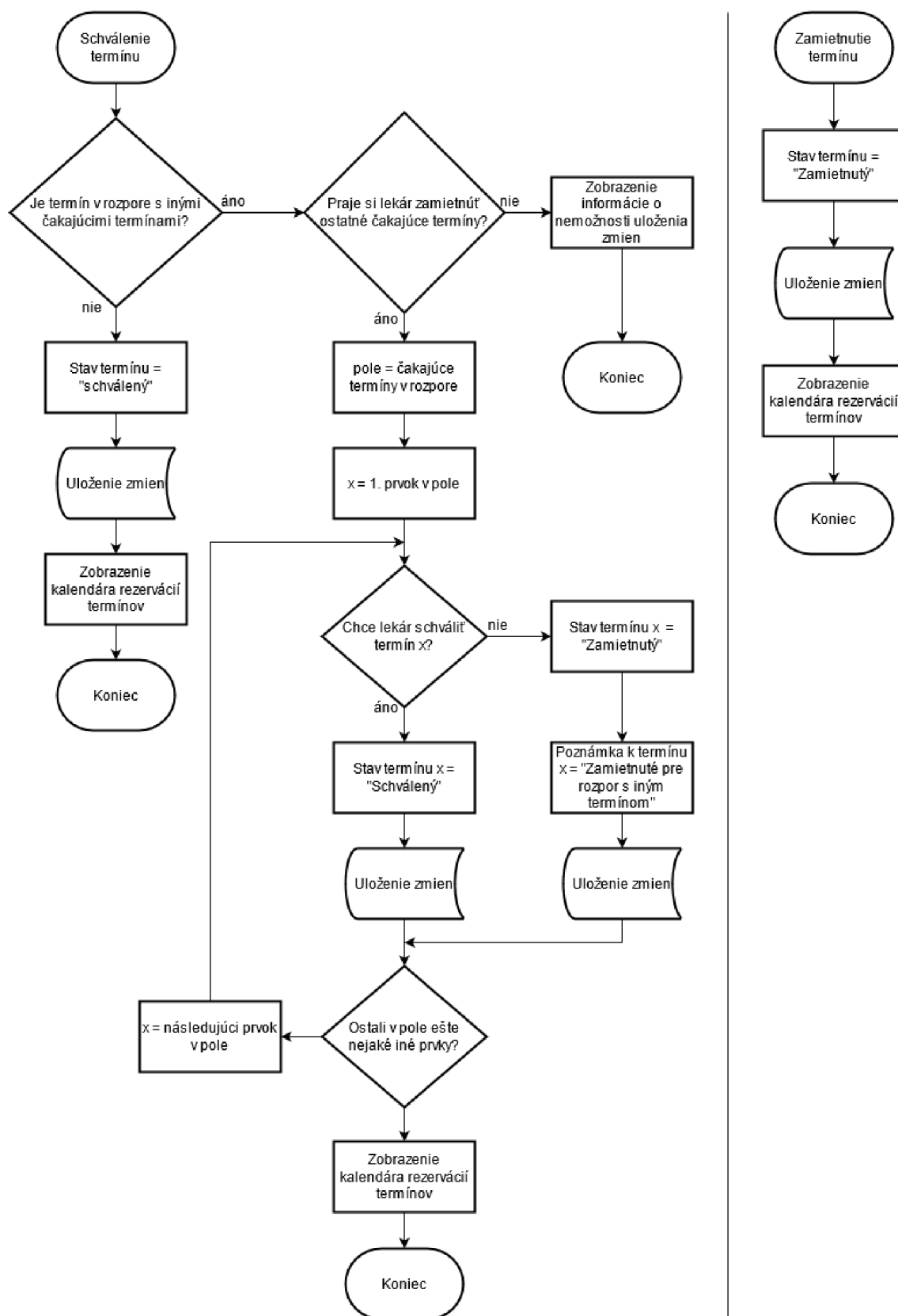
**Obrázok č. 32: Prihlásenie používateľa – Vývojový diagram**  
(Zdroj: Vlastné spracovanie)



Obrázok č. 33: Žiadosť o termín - DFD  
(Zdroj: Vlastné spracovanie)

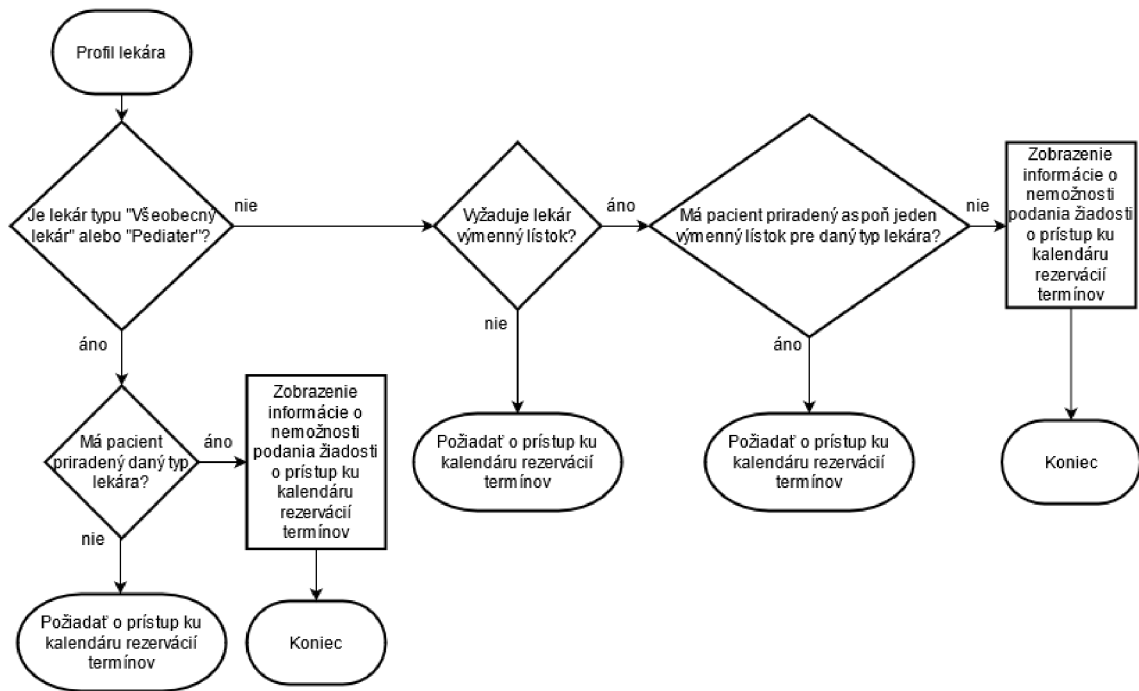


Obrázok č. 34: Schvaľovanie termínov – Vývojový diagram  
(Zdroj: Vlastné spracovanie)



Obrázok č. 35: Podprocesy schvaľovania termínov – Vývojový diagram  
(Zdroj: Vlastné spracovanie)





**Obrázok č. 36: Podanie žiadosti o prístup ku kalendáru rezervácií termínov – Vývojový diagram**  
(Zdroj: Vlastné spracovanie)

### 3.3 Zhodnotenie výsledného návrhu riešenia

Na zhodnotenie výsledného návrhu riešenia som si vybral SWOT analýzu, ekonomické zhodnotenie nákladov na vývoj a distribúciu navrhovanej aplikácie a nastavenie cenovej politiky.

#### 3.3.1 SWOT analýza

Pre túto analýzu som si vybral jednoduchú SWOT maticu, v ktorej vypíšem silné a slabé stránky navrhovanej aplikácie a príležitosti a hrozby na základe konkurencie na trhu a potencionálnych používateľov.

<p><b>Silné stránky</b></p> <ul style="list-style-type: none"> <li>▪ Natívna mobilná aplikácia</li> <li>▪ Dôležité lekárske moduly na jednom mieste</li> <li>▪ Jednoduchá, rýchla a spoľahlivá komunikácia medzi lekárom a pacientom</li> <li>▪ Fungovanie v obmedzenom offline režime</li> <li>▪ Uchovávanie historických dát</li> <li>▪ Schopnosť pracovať s výmennými lístkami</li> <li>▪ Riešenia namierené proti spamu a neoprávnenému prístupu</li> <li>▪ Upozornenia na dôležité udalosti</li> </ul>	<p><b>Slabé stránky</b></p> <ul style="list-style-type: none"> <li>▪ Riešenie len ako mobilná aplikácia (chýba webová)</li> <li>▪ Dopyt aj po iných ako navrhovaných funkciách</li> </ul>
<p><b>Príležitosti</b></p> <ul style="list-style-type: none"> <li>▪ Vysoký podiel využívania smartphonov na trhu</li> <li>▪ Veľmi vysoké množstvo potencionálnych používateľov – pacientov</li> <li>▪ Vysoké množstvo potencionálnych používateľov – lekárov</li> <li>▪ Aplikácia s prídavnými funkciami oproti konkurencii</li> <li>▪ Možnosť prepojenia s konkurenčnými aplikáciami</li> <li>▪ Inovácie v technológiach</li> </ul>	<p><b>Hrozby</b></p> <ul style="list-style-type: none"> <li>▪ Konkurenčné aplikácie</li> <li>▪ Návyky ľudí</li> <li>▪ Neochota lekárov prejsť na nové riešenia</li> <li>▪ Neochota konkurencie prepojiť aplikácie</li> </ul>

**Obrázok č. 37: SWOT matica navrhovaného riešenia**

(Zdroj: Vlastné spracovanie)

Ako je možné vidieť z obrázku, kladné stránky z vnútorného aj vonkajšieho prostredia aplikácie prevládajú nad tými zápornými a je teda možné povedať, že by mala navrhovaná aplikácia na trhu potenciál. Je potrebné však brať do úvahy aj slabé stránky aplikácie a hrozby vyplývajúce z okolia aplikácie, na základe ktorých by sa vytváral strategický plán prieniku aplikácie na trh. Z vyššie udaných bodov je zjavné, že aplikácia musí sľúbiť stále aktualizácie a prípadné pridávanie nových užitočných funkcií. Ďalej by

prieniku aplikácie na trh mohla pomôcť aj správne zameraná marketingová kampaň pre určený segment.

### **3.3.2 Ekonomické zhodnotenie nákladov**

Ekonomické zhodnotenie nákladov bolo vyčíslené hrubým odhadom celkových nákladov na vývoj aplikácie od skúseného programátora, ktorý zahrňuje samotný vývoj aplikácie pre OS Android a súčasne aj iOS, back-end aplikácie, testovanie a vytvorenie servera určeného pre aplikáciu. Keďže sa jedná iba o hrubý odhad, je nutné brať v úvahu aj skutočnosť, že niektoré súčasti aplikácie pri vývoji môžu zabráť viac času ako by sa mohlo zdať, preto je nutné zvážiť aj možnú finančnú rezervu. Pri priemernej mzde skúseného programátora, ktorá môže vyšplhať aj na 200 000 CZK mesačne, a vzhľadom k zložitosti navrhovanej aplikácie sú celkové náklady na vývoj aplikácie pomerne vysoké.

Odhadnutá bola taktiež celková cena nákladov na vývoj aplikácie, ak by bola aplikácia naprogramovaná junior programátorom (napr. štážištom/študentom). V tomto prípade sa môže jednať až o 10-násobné zníženie celkových nákladov na vývoj navrhovanej aplikácie. Tento odhad však predpokladá iba nenarušený priebeh vývoja daným junior programátorom, v skutočnosti však v takomto prípade do vývoja zvyčajne zasahuje aj skúsenejší programátor, čo by celkové náklady na vývoj aplikácie, prirodzene, zvýšilo.

Medzi náklady je potrebné pripočítať aj náklady vyplývajúce zo zverejnenia aplikácie na digitálnych distribučných platformách Obchod Google Play, čo bude činiť 25 USD, a iOS App Store, čo bude činiť 99 USD [12]. Pre konverziu ceny z USD na CZK som použil kurz devízového trhu podľa Českej národnej banky z dňa 26.4.2021, čo činí 1 USD za 21,353 CZK [38].

Vyššie spomenuté náklady zahrňujú náklady pre vývoj aplikácie až po jeho distribúciu (vrátane). Celkový súhrn týchto nákladov je v tabuľke nižšie.

**Tabuľka č. 24: Celkové náklady na vývoj a distribúciu navrhovanej aplikácie**  
(Zdroj: Vlastné spracovanie)

<b>Programuje:</b>	<b>Senior programátor</b>		<b>Junior programátor</b>
<b>Finančná rezerva</b>	nie	áno	nie
<b>Vývoj aplikácie</b>	cca 2 000 000 CZK	cca 3 000 000 CZK	cca 300 000 CZK
<b>Distribúcia – Obchod Google Play</b>	533,83 CZK		
<b>Distribúcia – iOS App Store</b>	2113,95 CZK		
<b>Spolu</b>	cca 2 002 647,78 CZK	cca 3 002 647,78 CZK	cca 302 647,78 CZK

Náklady na aplikáciu týmto ale nekončia. Po zverejnení aplikácie je nutné aj naďalej jej poskytovať podporu, vydávať aktualizácie (oprava chýb, nové funkcie) a spravovať a udržiavať server. Je teda nutné počítať aj s následnými variabilnými mesačnými nákladmi na prevádzku aplikácie.

### 3.3.3 Cenová politika

Prvotný návrh na aplikáciu je taký, že je primárne určená pre lekárov, ktorým by uľahčila komunikáciu a interakciu so svojimi pacientmi. Preto dáva zmysel, aby mohol používateľ pod účtom lekára používať aplikáciu iba po zakúpení licencie pre lekára. Pre pacientov by už ale aplikácia bola bezplatná, z dôvodu prilákania vyššieho počtu potencionálnych používateľov. Na stiahnutie by bola v obchodoch Google Play a iOS App Store taktiež bezplatne.

V tomto prípade by však bolo potrebné vybrať správnu cenu za licenciu pre lekára. Ponúkajú sa 2 možnosti:

- Vysoko výnosná cena – v tomto prípade by sa zvolila cena, ktorá odpovedá cenám konkurenčných aplikácií (v tomto prípade je najväčší konkurent aplikácia +4U Lékař, ktorej cena za licenciu činí 5 988 CZK a následne sa platí mesačný poplatok vo výške 499 CZK [33]) alebo nastaviť cenu trošku vyššie, ale argumentovať to tým, že aplikácia ponúka viac funkcií ako jeho konkurencia
- Prieniková cena – nastaviť cenu nižšiu ako konkurencia, čo by mohlo spočiatku prilákať vyšší počet záujemcov o aplikáciu

V hľadiska, že aplikácia nemusí byť vyvinutá a vydaná spoločnosťou s vysokým kapitálom, je bezpečnejší prvý variant ceny za licenciu pre lekára. Táto cena však nemusí vytvárať jediné výnosy z aplikácie. Je taktiež možné pacientom ako používateľom, ktorí aplikáciu využívajú bezplatne, zobrazovať na vyhradených miestach v rámci aplikácie reklamy, čo by znamenalo ďalší výnos pre vydavateľa aplikácie z reklám. Vhodné by taktiež bolo zobrazovať iba reklamy, ktoré súvisia s účelom aplikácie (napr. reklamy na lieky, fitness aplikácie, atď.). V neposlednom rade, ďalším možným výnosom by mohli byť dobrovoľné príspevky od pacientov, ktorí by chceli podporiť tvorcov aplikácie, za čo by im na výmenu napr. mohlo byť zrušené zobrazovanie reklám v aplikácii.

### **3.4 Prínosy práce**

Z navrhovanej aplikácie v tejto práci budú ťažiť najmä lekári a ich pacienti. Vďaka intuitívnemu prostrediu bude aplikácia ponúkať jednoduché, rýchle a spoľahlivé riešenie pre komunikáciu a interakciu medzi lekármi a pacientmi. Používateľom aplikácie by sa ponúkala automatická organizácia dát na jednom mieste, čo im umožní ľahké dohľadanie potrebných dát, dokonca aj pomocou filtrovania výsledkov. Kalendár rezervácií termínov, chaty, selfmonitoring a výmenné lístky by úplne nahradili osobnú návštevu alebo telefonické rozhovory určené pre vybavenie záležitostí spojených s danými funkciami. Aj v prípade neprítomnosti a nedostupnosti lekára táto aplikácia zabezpečuje riadne a spoľahlivé doručenie potrebných dát priamo lekárovi, čo rieši aj problém neschopnosti dovolať sa lekárovi. Žiadny používateľ taktiež nikdy nezmešká upozornenie na dôležité udalosti aj vďaka možnosti výberu doručenia notifikácií nie len prostredníctvom OS notifikácií mobilného zariadenia, ale taktiež aj prostredníctvom SMS alebo e-mailu. Vďaka schopnosti aplikácie uchovávať historické dáta budú používatelia upozorňovaní OS notifikáciami mobilného zariadenia na dôležité udalosti aj v prípade, že nemajú prístup k internetu na základe dát uložených vo vyrovnávacej pamäti aplikácie, ktoré si aplikácia stiahla v čase, keď mobilné zariadenie používateľa bolo pripojené k internetu. Obávať sa taktiež nebude treba prípadného spamu od nežiadúcich pacientov, pretože tí môžu aplikáciu používať jedine, ak im to povolí ich lekár a interagovať s jeho kalendárom rezervácií termínov alebo mu napísať prostredníctvom chatu môžu iba v prípade, ak mu lekár schváli prístup k týmto funkciám. Prístup k týmto funkciám trvá rôzne dlho pri rôznych typoch lekárov, takže sa niektoré funkcie pre pacientov zablokujú

aj v prípade, že si termín u lekára raz rezervoval, čo pre lekárov znamená dostávanie žiadostí iba od pacientov, ktorí reálne u daného lekára vyšetrenie potrebujú.

Na druhej strane táto aplikácia by mala potenciálny prínos aj pre vydavateľa a vývojárov aplikácie, a to vo forme výnosov z predaja licencií pre lekárov, podielov z reklám a prípadných dobrovoľných príspevkov od pacientov. Aj keď sa analýza tejto práce sústreďovala skôr na český a slovenský trh, po preskúmaní ďalších trhov a zistení ich požiadaviek a kritérií by sa aplikácia mohla rozšíriť aj do zahraničia, čo by mohlo zvýšiť renomé vydavateľa aj vývojárov aplikácie.

## ZÁVER

Cieľom tejto práce bolo vytvorenie návrhu riešenia mobilnej aplikácie, ktorá by uľahčila a zefektívnila komunikáciu medzi lekármi a ich pacientami a ponúkala by intuitívne a jednoduché prostredie pre navigáciu a spravovanie hlavných modulov aplikácie. Hlavnou súčasťou aplikácie je kalendár rezervácie termínov, pomocou ktorého môžu používatelia vidieť obsadenosť daného lekára a taktiež si o termín u daného lekára zažiadať. Pri tejto súčasti však funkcie aplikácie nekončia. Samozrejmosťou tejto aplikácie je aj možnosť spravovať členov rodiny, chatovanie so svojimi lekármi a pacientami, dostávať notifikácie o dôležitých udalostiach a fungovanie v obmedzenom offline režime. Úzko späté funkcie s rezerváciou termínov ako sú selfmonitoring, predpisy liekov a výmenné lístky sú v aplikácii taktiež prítomné a vytvárajú pridanú hodnotu oproti iným konkurenčným aplikáciám. V práci sa rieši aj odolnosť voči spamu.

Riešenie návrhu aplikácie som zhotovil v troch hlavných kapitolách. V prvej kapitole som sa venoval teórii, ktorá priamo nadväzuje na návrh riešenia v tejto práci, preto som v nej zahrnul najmä teóriu OS, digitálnych distribučných platforiem, dátového a funkčného modelovania a vysvetlenie medicínskych pojmov využívaných v selfmonitoringu. V druhej kapitole som sa zaoberal analýzou trhu mobilných zariadení, mobilných OS a mobilných aplikácií na prítomnosť podobných aplikácií, pričom za priamych konkurentov som považoval najmä aplikácie pre český a slovenský trh. V poslednej tretej kapitole som vytvoril samotný návrh aplikácie, kde som databázu aplikácie zobrazil pomocou dátového modelovania a funkcie aplikácie pomocou slovného opisu a obrázkov, pričom podstatné funkcie, na ktorých aplikácia stavia som zobrazil aj pomocou diagramov funkčného modelovania. Nakoniec som výsledky práce zhodnotil SWOT analýzou, ekonomickým zhodnotením nákladov na vývoj a distribúciu navrhovanej aplikácie a nastavením cenovej politiky pre predaj licencií pre lekárov.

Táto práca mi priniesla aj osobný prínos, keďže som si pri jej vypracovaní oprášil svoje vedomosti z dátového i funkčného modelovania a taktiež aj z oblasti grafiky, ktoré som nabral počas môjho štúdia na fakulte. Okrem využitia svojich poznatkov som sa počas vypracovania tejto práce naučil aj mnoho nových vecí z danej problematiky a verím, že mi táto práca dala mnoho nových a užitočných skúseností, ktoré budem môcť v budúcnosti využiť.

## ZOZNAM POUŽITÝCH ZDROJOV

1. What is Operating System? Types of OS, Features and Examples. In: *Guru99* [online]. [cit. 2021-03-25]. Dostupné z: <https://www.guru99.com/operating-system-tutorial.html>
2. CERVANTES, Edgar. What is Android? Here's everything you need to know. In: *Android Authority* [online]. ©2020 [cit. 2021-03-25]. Dostupné z: <https://www.androidauthority.com/what-is-android-328076/>
3. CHEN, James. Android Operating System. In: *Investopedia* [online]. ©2021 [cit. 2021-03-25]. Dostupné z: <https://www.investopedia.com/terms/a/android-operating-system.asp>
4. The Android Open Source Project on Open Hub: Languages Page [online]. In: *Synopsys: Black Duck Open Hub* [online]. ©2020 [cit. 2021-03-25]. Dostupné z: [https://www.openhub.net/p/android/analyses/latest/languages\\_summary](https://www.openhub.net/p/android/analyses/latest/languages_summary)
5. *Android The platform pushing what's possible* [online]. California: Google LLC, [cit. 2021-03-25]. Dostupné z: <https://www.android.com/>
6. Android is for everyone. *Google* [online]. California: Google LLC, [cit. 2021-03-25]. Dostupné z: <https://www.android.com/everyone/>
7. SPEIGHT, Adam. What is iOS and what does iOS stand for? In: *Recombu* [online]. ©2020 [cit. 2021-03-25]. Dostupné z: <https://recombu.com/mobile/article/what-is-ios-and-what-does-ios-stand-for>
8. iOS 14 – Apple. *Apple* [online]. California: Apple Inc, ©2021 [cit. 2021-03-25]. Dostupné z: <https://www.apple.com/ios/ios-14/>
9. GILLIS, Alexander S. What is application? In: *SearchSoftwareQuality* [online]. [cit. 2021-03-24]. Dostupné z: <https://searchsoftwarequality.techtarget.com/definition/application>
10. Mobile Apps vs Web Apps: Which is the Better Option? In: *Sagara Technology Idea Lab* [online]. ©2019 [cit. 2021-03-24]. Dostupné z: <https://sagaratechnology.medium.com/mobile-apps-vs-web-apps-which-is-the-better-option-868106c88730>
11. Digital Distribution Law and Legal Definition. In: *USLegal* [online]. [cit. 2021-03-26]. Dostupné z: <https://definitions.uslegal.com/d/digital-distribution/>



12. VISWANATHAN, Priya. iOS App Store vs. Google Play Store. In: *Lifewire* [online]. ©2020 [cit. 2021-03-26]. Dostupné z: <https://www.lifewire.com/ios-app-store-vs-google-play-store-for-app-developers-2373130>
13. *Google Play* [online]. California: Google LLC, [cit. 2021-03-26]. Dostupné z: <https://play.google.com/store>
14. *App Store – Apple* [online]. California: Apple Inc, ©2021 [cit. 2021-03-25]. Dostupné z: <https://www.apple.com/app-store/>
15. KOCH, Miloš a Bernard NEUWIRTH. *Datové a funkční modelování*. 4., rozšířené vyd. Brno: Akademické nakladatelství CERM®, s.r.o., 2010. ISBN 978-80-214-4125-5.
16. LACKO, Ľuboslav. *Jak vyzrát na Microsoft SQL Server 2008* [online]. Brno: Computer Press, 2009 [cit. 2021-03-22]. ISBN 978-80-251-2101-6. Dostupné z: <https://ndk.cz/view/uuid:6ae97280-ee61-11e4-b834-005056827e51>
17. Data types (Transact-SQL). In: *Microsoft* [online]. ©2017 [cit. 2021-03-22]. Dostupné z: <https://docs.microsoft.com/en-us/sql/t-sql/data-types/data-types-transact-sql?view=sql-server-ver15>
18. Flowchart Tutorial (with Symbols, Guide and Examples). In: *Visual Paradigm* [online]. [cit. 2021-03-28]. Dostupné z: <https://www.visual-paradigm.com/tutorials/flowchart-tutorial/>
19. GRASSEOVÁ, Monika, Radek DUBEC a David ŘEHÁK. *Analýza podniku v rukou manažera* [online]. Brno: Computer Press, 2010 [cit. 2021-03-24]. ISBN 978-80-251-2621-9. Dostupné z: <https://ndk.cz/view/uuid:45ffbdba0-39a6-11e4-8e0d-005056827e51>
20. KAŇOVSKÁ, Lucie a David Schüller. *Základy marketingu*. 2., přepracované vyd. Brno: Akademické nakladatelství CERM®, s.r.o., 2015. ISBN 978-80-214-5107-0.
21. FERTAIOVÁ, Terézia, ed. *Ošetrovateľské postupy I*. 1. vyd. Prešov: Vydavateľstvo FZO PU v Prešove, 2019. ISBN 978-80-555-2232-6.
22. FERTAIOVÁ, Terézia, ed. *Ošetrovateľské postupy II*. 1. vyd. Prešov: Prešovská univerzita, Fakulta zdravotníckych odborov, 2020. ISBN 978-80-555-2406-1.
23. SILVER, Laura. Smartphone Ownership Is Growing Rapidly Around the World, but Not Always Equally. In: *Pew Research Center* [online]. ©2019 [cit. 2021-01-18].

- Dostupné z: <https://www.pewresearch.org/global/2019/02/05/smartphone-ownership-is-growing-rapidly-around-the-world-but-not-always-equally/>
24. Mobile Vs. Desktop Internet Usage (Latest 2020 Data). In: *BroadbandSearch* [online]. [cit. 2021-01-18]. Dostupné z: <https://www.broadbandsearch.net/blog/mobile-desktop-internet-usage-statistics>
  25. Mobile app vs mobile website statistics. In: *JMango360* [online]. [cit. 2021-01-18]. Dostupné z: <https://jmango360.com/mobile-app-vs-mobile-website-statistics/>
  26. Mobile Operating System Market Share Worldwide. In: *StatCounter* [online]. ©2020 [cit. 2021-01-19]. Dostupné z: <https://gs.statcounter.com/os-market-share/mobile/worldwide>
  27. Google Play vs the iOS App Store | Store Stats for Mobile Apps. *42matters* [online]. Zürich: 42matters AG, ©2020 [cit. 2021-01-19]. Dostupné z: <https://42matters.com/stats>
  28. Appointment Manager: Doctors. In: *Google Play* [online]. ©2018 [cit. 2021-01-20]. Dostupné z: <https://play.google.com/store/apps/details?id=medindia4u.net.doctorappointment&hl=en&gl=US>
  29. Zocdoc Find A Doctor & Book On Demand Appointments. In: *Google Play* [online]. ©2020 [cit. 2021-01-20]. Dostupné z: <https://play.google.com/store/apps/details?id=com.zocdoc.android&hl=en&gl=US>
  30. Doctor's Appointment App Builder. *App Builder Appy Pie* [online]. Appy Pie, ©2021 [cit. 2021-01-20]. Dostupné z: <https://www.appypie.com/doctor-appointment-app>
  31. SteadyMD – Your Personal Doctor, Online. In: *Google Play* [online]. ©2020 [cit. 2021-01-20]. Dostupné z: <https://play.google.com/store/apps/details?id=com.steadymd.chat&hl=en&gl=US>
  32. MDLIVE: Talk to a Doctor 24/7. In: *Google Play* [online]. ©2020 [cit. 2021-01-20]. Dostupné z: <https://play.google.com/store/apps/details?id=com.mdlive.mobile&hl=en&gl=US>
  33. +4U Lékař | Rezervační systém pro lékaře i pacienty. *Plus4U | Objevte svůj digitální svět* [online]. Praha: Plus4U Net a.s., [cit. 2021-04-26]. Dostupné z: <https://www.plus4u.net/lekar/rezervacni-system>

34. Plus4U Net. Rezervační systém +4U Lékař. In: *Youtube* [online]. 18. 7. 2017 [cit. 2021-01-21]. Dostupné z: [https://www.youtube.com/watch?v=mksepsDW95g&feature=emb\\_logo](https://www.youtube.com/watch?v=mksepsDW95g&feature=emb_logo)
35. Objednanie k lekárovi cez internet. In: *NávštevaLekára.sk* [online]. 20. 9. 2013 [cit. 2021-01-21]. Dostupné z: <https://www.navstevalekara.sk/clanky.html?article=14>
36. *Reservio* [online]. Brno: Reservio, ©2021 [cit. 2021-01-21]. Dostupné z: <https://www.reservio.com/cs/>
37. Reservio Business – Appointment Scheduling. In: *Google Play* [online]. ©2020 [cit. 2021-01-21]. Dostupné z: <https://play.google.com/store/apps/details?id=com.reservio&hl=en&gl=US>
38. Kurzy devízového trhu. *Česká národní banka* [online]. Praha: Česká národní banka, ©2021 [cit. 2021-04-26]. Dostupné z: <https://www.cnb.cz/cs/financni-trhy/devizovy-trh/kurzy-devizoveho-trhu/kurzy-devizoveho-trhu/>

## ZOZNAM POUŽITÝCH OBRÁZKOV

Obrázok č. 1: OS ako sprostredkovateľ .....	13
Obrázok č. 2: Android logo .....	15
Obrázok č. 3: iOS 14 logo .....	16
Obrázok č. 4: Logo Obchodu Google Play™ .....	19
Obrázok č. 5: iOS® App Store® logo .....	19
Obrázok č. 6: Entita a jej atribúty .....	20
Obrázok č. 7: Typy atribútov .....	21
Obrázok č. 8: Zložitosť atribútov .....	21
Obrázok č. 9: Relačný dátový model .....	22
Obrázok č. 10: Integrita relačného dátového modelu .....	23
Obrázok č. 11: Dekompozícia vzťahu N:M .....	24
Obrázok č. 12: „Inžiniersky štýl“ entito-relačného modelu .....	25
Obrázok č. 13: Prvky diagramu toku dát podľa štýlu <i>Yourdon and Coad</i> .....	27
Obrázok č. 14: Diagram toku dát .....	28
Obrázok č. 15: Prvky vývojového diagramu .....	29
Obrázok č. 16: Vývojový diagram .....	29
Obrázok č. 17: SWOT matica .....	30
Obrázok č. 18: Obrázky k aplikácii <i>Appointment Manager: Doctors</i> .....	41
Obrázok č. 19: Obrázky k aplikácii <i>Zocdoc</i> .....	42
Obrázok č. 20: Obrázky k aplikácii <i>SteadyMD</i> .....	43
Obrázok č. 21: Obrázky k rezervačnému systému <i>+4U Lékař</i> .....	44
Obrázok č. 22: Obrázky k aplikácii <i>Reservio Business</i> .....	45
Obrázok č. 23: ER model aplikácie .....	48
Obrázok č. 24: Návrh prihlasovacieho a registračného formulára .....	62
Obrázok č. 25: Návrh výzoru profilov jednotlivých používateľov z pohľadu pacienta .....	63
Obrázok č. 26: Návrh funkcie <i>Členovia rodiny</i> .....	64
Obrázok č. 27: Návrh kategórie <i>Notifikácie</i> .....	66
Obrázok č. 28: Návrh kalendára rezervácií termínov .....	69
Obrázok č. 29: Návrh funkcie <i>Selfmonitoring</i> .....	72
Obrázok č. 30: Návrh chatovacieho modulu .....	74
Obrázok č. 31: Registrácia používateľa - DFD .....	76

Obrázok č. 32: Prihlásenie používateľa – Vývojový diagram .....	77
Obrázok č. 33: Žiadosť o termín - DFD .....	78
Obrázok č. 34: Schvaľovanie termínov – Vývojový diagram .....	79
Obrázok č. 35: Podprocesy schvaľovania termínov – Vývojový diagram .....	80
Obrázok č. 36: Podanie žiadosti o prístup ku kalendáru rezervácií termínov – Vývojový diagram .....	81
Obrázok č. 37: SWOT matica navrhovaného riešenia.....	82

## ZOZNAM POUŽITÝCH TABULIEK

Tabuľka č. 1: Dátové typy jazyka T-SQL.....	26
Tabuľka č. 2: Klasifikácia hypertenzie (vysokého krvného tlaku).....	32
Tabuľka č. 3: Fyziologické hodnoty pulzu vo vzťahu k veku.....	32
Tabuľka č. 4: Hodnoty telesnej teploty.....	33
Tabuľka č. 5: BMI.....	33
Tabuľka č. 6: Referenčné hodnoty biochemického vyšetrenia krvi.....	33
Tabuľka č. 7: Tabuľka Doctors.....	50
Tabuľka č. 8: Tabuľka Patients.....	50
Tabuľka č. 9: Tabuľka FamilyMembers.....	51
Tabuľka č. 10: Tabuľka PhoneNumbers.....	51
Tabuľka č. 11: Tabuľka Emails.....	52
Tabuľka č. 12: Tabuľka Notifications.....	53
Tabuľka č. 13: Tabuľka Chats.....	54
Tabuľka č. 14: Tabuľka Relations.....	54
Tabuľka č. 15: Tabuľka Reservations.....	55
Tabuľka č. 16: Tabuľka Absences.....	55
Tabuľka č. 17: Tabuľka ExchangeTickets.....	56
Tabuľka č. 18: Tabuľka Prescriptions.....	56
Tabuľka č. 19: Tabuľka PendingPatients.....	57
Tabuľka č. 20: Tabuľka Selfmonitoring.....	58
Tabuľka č. 21: Tabuľka ReservationTemplates.....	59
Tabuľka č. 22: Tabuľka Catalog.....	59
Tabuľka č. 23: Tabuľka pre dedenie atribútov.....	60
Tabuľka č. 24: Celkové náklady na vývoj a distribúciu navrhovanej aplikácie.....	84

## **ZOZNAM POUŽITÝCH GRAFOV**

Graf č. 1: Vlastníctvo smartphonov v rozvinutých ekonomikách vyššie ako v rozvíjajúcich sa (v %) .....	35
Graf č. 2: Mobilné technológie, internet a sociálne média sú využívané viac v rozvinutých ekonomikách (v %) .....	36
Graf č. 3: Využitie internetu vo svete (v %) .....	37
Graf č. 4: Tržný podiel mobilných operačných systémov na svete (v %) .....	38
Graf č. 5: Obchod Google Play™ štatistiky (v %) .....	39
Graf č. 6: iOS® App Store® štatistiky (v %) .....	39
Graf č. 7: Rozdelenie kategórií v Obchode Google Play™ (počet aplikácií).....	40
Graf č. 8: Rozdelenie kategórií v iOS® App Store® (počet aplikácií) .....	40

## **ZOZNAM POUŽITÝCH SKRATIEK A SYMBOLOV**

- API – Application Programming Interface (rozhranie pre programovanie aplikácií)
- BMI – Body Mass Index (index telesnej hmotnosti)
- CK – Candidate Key (kandidátny kľúč)
- CSS – Cascading Style Sheets (kaskádové štýly)
- DFD – Data Flow Diagram (diagram toku dát)
- ER – Entity Relationship (entito-relačný)
- FK – Foreign Key (cudzí kľúč)
- GUI – Graphical User Interface (grafické používateľské rozhranie)
- HTML – HyperText Markup Language (hypertextový značkovací jazyk)
- I/O – Input/Output (vstupno-výstupný)
- MS – Microsoft
- PK – Primary Key (primárny kľúč)
- SQL – Structured Query Language (štruktúrovaný vyhľadávací jazyk)
- SWOT – Strengths, Weaknesses, Opportunities, Threats (silné a slabé stránky, príležitosti a hrozby)
- T-SQL – Transact-SQL (transakčný SQL)
- TK – tlak v krvi