VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

MOBILNÍ APLIKACE PRO USNADNĚNÍ ONLINE ŠKOLNÍ VÝUKY A KOMUNIKACE SE STUDENTY

MOBILE APPLICATION FOR EASY ONLINE SCHOOL EDUCATION AND COMMUNICATION WITH STUDENTS

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE **LUKÁŠ OTÁHAL**
AUTHOR

VEDOUCÍ PRÁCE **Ing. TOMÁŠ CHLUBNA**
SUPERVISOR

BRNO 2023

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

Zadání bakalářské práce

145562

Ústav: Ústav počítačové grafiky a multimédií (UPGM)
Student: **Otáhal Lukáš**
Program: Informační technologie
Specializace: Informační technologie
Název: **Mobilní aplikace pro usnadnění online školní výuky a komunikace se studenty**
Kategorie: Mobilní aplikace
Akademický rok: 2022/23

Zadání:

1. Seznamte se s principy vývoje mobilních aplikací a vyberte vhodné vývojové prostředí (Qt, React Native, Flutter...).
2. Naučte se základy práce ve vybraném frameworku.
3. Nastudujte potřebné materiály k tématu výukových aplikací pro školní prostředí.
4. Nastudujte existující a možná řešení dané problematiky.
5. Navrhněte uživatelské rozhraní a výslednou funkcionalitu.
6. Aplikaci implementujte a demonstrujte její použití na různých typech dat.
7. Proveďte uživatelské či výkonnostní měření výsledku.
8. Zdokumentujte a zveřejněte výslednou aplikaci.
9. Vytvořte video reprezentující výsledky vaší práce.

Literatura:

- Hoober, Steven, and Eric Berkman. *Designing mobile interfaces: Patterns for interaction design*. " O'Reilly Media, Inc.", 2011.
- Suartama, I. Kadek, Punaji Setyosari, and Saida Ulfa. "Development of an instructional design model for mobile blended learning in higher education." *International Journal of Emerging Technologies in Learning* 14.16 (2019).
- Oyelere, Solomon Sunday, et al. "Design, development, and evaluation of a mobile learning application for computing education." *Education and Information Technologies* 23.1 (2018): 467-495.
- Botzer, Galit, and Michal Yerushalmy. "Mobile application for mobile learning." *Proceedings of IADIS International Conference on Cognition and Exploratory Learning in Digital Age (CELDA 2007)*. 2007.
- Karabatzaki, Zoe, et al. "Mobile Application Tools for Students in Secondary Education. An Evaluation Study." *International Journal of Interactive Mobile Technologies* 12.2 (2018).
- Truong, Dothang. "How to design a mobile application to enhance teaching and learning?." *International Journal of Emerging Technologies in Learning* 9.3 (2014).

Při obhajobě semestrální části projektu je požadováno:
Body 1 až 5, experimenty vedoucí k bodu 6.

Podrobné závazné pokyny pro vypracování práce viz https://www.fit.vut.cz/study/theses/

Vedoucí práce: **Chlubna Tomáš, Ing.**
Vedoucí ústavu: Černocký Jan, prof. Dr. Ing.
Datum zadání: 1.11.2022
Termín pro odevzdání: 10.5.2023
Datum schválení: 31.10.2022

Abstrakt

Cílem práce je vytvořit podpůrnou mobilní aplikaci pro studenty i učitele, rozšířenou o možnosti, které nejsou u dosud existujích školních aplikací na trhu zatím dostupné. V aplikaci podle přihlášení budou 2 pohledy. Pohled pro žáka a pohled pro učitele. Aplikace je vytvořena s důrazem na jednoduchost a přívětivost použití. Aplikace nabízí možnosti, jako je psaní a hodnocení testů, zobrazení studijních materiálů, zadávání či prohlížení úkolů, zobrazení novinek ze školy nebo školních akcí, přehled rozvrhu, přehled tříd a detailní informace žáka s klasifikací. Nově vzniklá školní aplikace je vytvořena přes platformu Jigx a data jsou uložena v databázi AzureSQL.

Abstract

The aim of the work is to create a supporting mobile application for students and teachers, enhanced with capabilities that are not yet available in existing school applications on the market. There will be 2 views in the app according to the login. A view for the student and a view for the teacher. The app is designed with an emphasis on simplicity and user friendliness. The app offers options such as writing and grading tests, viewing study materials, assigning or viewing assignments, viewing school news or school events, schedule overview, class overview and detailed student information with classification. The new school app is built using the Jigx platform and the data is stored in an AzureSQL database.

Klíčová slova

Mobilní aplikace, jig, widget, Android, iOS, Mobilní systémy, výuková aplikace

Keywords

Mobile application, jig, widget, Android, iOS, Mobile systems, educational application

Citace

OTÁHAL, Lukáš. *Mobilní aplikace pro usnadnění online školní výuky a komunikace se studenty.* Brno, 2023. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Tomáš Chlubna

Mobilní aplikace pro usnadnění online školní výuky a komunikace se studenty

Prohlášení

Prohlašuji, že jsem bakalářskou práci vypracoval samostatně pod vedením pana Ing. Chlubny. Veškerá literatura a zdroje, které jsem použil při psaní této práce, jsou citovány v textu a uvedeny v seznamu literatury.

. . . . . .
Lukáš Otáhal
2. května 2023

Poděkování

Zde bych rád nejprve poděkoval svému vedoucímu práce panu Ing. Chlubnovi za cenné rady a čas, který věnoval mně i mé práci. Dále bych chtěl poděkovat své rodině, přítelkyni a všem přátelům za podporu, kterou mi poskytovali po celou dobu psaní práce i studia. V neposlední řadě děkuji také firmě Jigx, která mi umožnila skrze jejich platformu tuto aplikaci vytvořit.

Obsah

1 Úvod **2**

2 Teoretická část **3**
2.1 Mobilní systémy 3
2.2 Existující řešení 12
2.3 Využití mobilních zařízení ve škole 17
2.4 Platforma Jigx 18

3 Analýza a návrh **21**
3.1 Analýza 21
3.2 Návrh 23

4 Implementace **27**
4.1 Home Hub 27
4.2 Konfigurace jednotlivých widgetů 30
4.3 Konfigurace testů a listů aplikace 32

5 Testování **36**
5.1 Uživatelské testování 36

6 Závěr **39**
6.1 Pokračování ve vývoji aplikace 39

Literatura **40**

A Obsah CD **43**
A.1 Ukázka finální aplikace 44
A.2 Schéma entit 45
A.3 Ukázka dotazníku 46

Kapitola 1

Úvod

Bakalářská práce se zabývá vytvořením nové mobilní aplikace, která má sloužit především k usnadnění a obohacení výuky na základních, případně středních školách. Hlavním cílem práce je vytvořit podpůrnou mobilní aplikaci pro studenty i učitele, rozšířenou o možnosti, které nejsou u dosud existujích školních aplikací na trhu zatím dostupné. Inovativními aspekty nebo možnostmi, které vzniklá mobilní aplikace nabízí, jsou pro studenty i učitele především procházení veškerých zveřejněných materiálů, psaní i tvorba cvičení nebo testů, souhrnný přehled nejnovějších aktualit ze života školy, snadnější vzájemná komunikace a také dostupnost aktuálních čísel školního časopisu.

Dílčím, ovšem neméně důležitým cílem práce je vytvořit aplikaci, která bude rychlá, snadná na ovládání a uživatelsky přívětivá, jak pro studenty, tak učitele. Zároveň je aplikace navržena tak, aby pomáhala k dosažení lepších studijních výsledků a ušetřila čas při tvorbě nebo konání jednotlivých úkonů.

Téma mobilních aplikací je v dnešní době velmi rozšířené a tím pádem i aktuální, a to především z důvodu rychlého a neustálého vývoje digitálních technologiích, které zasahují téměř do všech každodenních situací. Mobilní aplikace svým uživatelům pomáhají a usnadňují každodenní pracovní i osobní život. Jak už bylo zmíněno, mobilní aplikace nebo digitální technologie obecně, zasahují do mnoha oblastí života a tím pádem ani školství není výjimkou. Vzniklá školní mobilní aplikace je navržena a nastavena tak, aby obohacovala, usnadňovala a zároveň i vzdělávala. Svůj prostor v aplikaci mají jak studenti, tak učitelé. Její inovativnost spočívá v nových rozšířených možnostech užívání a také v její uživatelské přívětivosti, která byla otestována žáky 2. stupně základní školy.

Práce je rozdělena na teoretickou a praktickou část. V sekci 2 se práce nejprve zabývá problematikou mobilních systémů obecně, dále jsou popsány a porovnány dosud existující školní aplikace s důrazem na jejich výhody a nevýhody. Práce dále pokračuje popisem využití mobilních aplikací v dnešní době a definicí teoretických poznatků potřebných pro úspěšný návrh a realizaci aplikace. Teoretická část je zakončena představením platformy, přes kterou byla mobilní aplikace vytvořena. V sekci 3 jsou popsány prvotní myšlenky, nápady a vize, které vedly k návrhu řešení a ke vzniku aplikace samotné. V sekci 4 jsou popsány jednotlivé kroky a celkový postup navržení a vytvoření aplikace. Dále jsou zmíněny použité knihovny i struktury aplikace. Na závěr je popsáno fungování celé aplikace, implementační detaily a její využitelnost. Ve finální sekci 6 je shrnuto a zhodnoceno dosažení plánovaných cílů ve spojitosti s obecnou komparací nově vzniklé aplikace a dosud existujících školních aplikací. Za přidanou hodnotu lze v závěru považovat přehled a interpretaci výsledků testování aplikace, které bylo provedeno pomocí dotazníků u žáků a třídního učitele na 2. stupni základní školy.

Kapitola 2

Teoretická část

V teoretické části jsou popsány jednotlivé sekce vedoucí k celkovému pochopení tématu vývoje mobilních aplikací.

2.1 Mobilní systémy

Mobilní systémy jsou operační systémy, které jsou optimalizované pro chytré telefony, tablety a další mobilní zařízení. Mezi hlavní rozdíly operačního systému stolního počítače a operačního systému mobilního zařízení patří podpora mobilních funkcí - ovládání pomocí dotyku, GPS lokalizace a navigace, volání i posílání SMS, spravování souborů, uživatelské rozhraní a složitější funkce, které jsou řízeny mobilním operačním systémem. Mezi nejznámější operační systémy lze zařadit: iOS a Android. Podrobný popis konkrétních jmenovaných systémů je popsán v sekci 2.1.3 [5].

2.1.1 Mobilní telefony a smartphony

Mobilní zařízení známé také jako tzv. „kapesní zařízení“, je počítačové zařízení kapesní velikosti, obvykle používající displej s dotykovým vstupem. Mobilní zařízení lze definovat jako přenosné, komunikační a informační zařízení, které umožňuje připojení k internetové síti. Lze jej také chápat jako výpočetní zařízení, které můžeme ovládat rukou. Mezi mobilní zařízení řadíme například: notebook, mobilní telefon, smartphone, tablet, smartwatch, herní konzole, PDA a jiné zařízení (chytré brýle, náramky, apod.) [14]. Trh s mobilními zařízeními v dnešní době výrazně roste a na trhu je velké množství chytrých telefonů i tabletů. Telefony a tablety jsou oblíbené především u lidí, kteří vyžadují funkce, pohodlí a pomoc běžného počítače v prostředí, kde jeho využití není praktické nebo dostupné. Zařízení tohoto typu poskytují uživatelům funkce a vlastnosti potřebné pro studium, osobní využití, práci i podnikání [13]. Moderní a chytré telefony se ovšem v dnešní době již mohou pyšnit mnoha funkcemi, které uživateli plnohodnotně nahradí i počítač. Mezi takové funkce patří například: vysoký výkon CPU a GPU, velký úložný prostor, multitasking obrazovky, fotoaparáty s vysokým rozlišením, víceúčelové funkce a komunikační hardware. Výše popsané telefony nazýváme „smartphony“. Užité moderní mobilní operační systémy kombinují vlastnosti osobního počítačového operačního systému s dalšími funkcemi, včetně dotykového displeje, mobilních sítí, Bluetooth, Wi-Fi, globálního určování polohy, systému (GPS), mobilní navigace, videokamery, rozpoznávání řeči, hlasového záznamníku, hudebního přehrávače nebo funkcí pro rozpoznávání blízkých polí. Rozšiřující trh s aplikacemi navíc umožňuje uživatelům používat tato zařízení nejen k vyhledávání bodů zájmu, ale

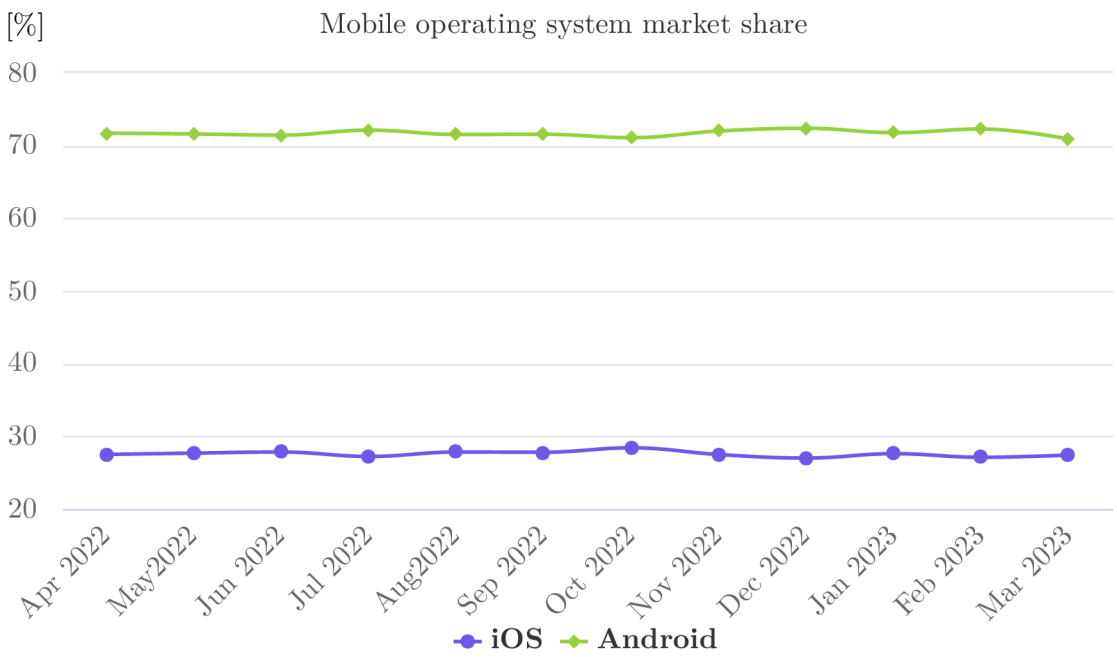
i jako GPS navigaci nebo k organizaci obchodních či studijních prácí, sdílení informací s ostatními, aktualizaci nejnovějších zpráv a vyhledávání Wifi připojení. Zároveň se mohou uživatelé bavit při různých mobilních hrách nebo videích. Mezi dodavateli mobilních zařízení má momentálně nejvyšší podíl na trhu společnost Apple [13].

2.1.2 Mobilní aplikace

Používání a vývoj mobilních aplikací je rychle se rozvíjejícím odvětvím. Mobilní aplikace jsou novým a zároveň rychle se rozvíjejícím trendem globálních informačních a komunikačních technologií, který má široké využití pro svou rozsáhlou oblast fungování, jako je volání, zasílání zpráv, prohlížení souborů, komunikace na sociálních sítích, hraní her nebo audio a video přehrávání. Ve velkém množství mobilních aplikací jsou některé základní aplikace předinstalovány v telefonu. Další si uživatel může stáhnout z internetu a nainstalovat je do mobilního telefonu podle svého uvažení a preferencí. Zmíněný velký počet mobilních aplikací na trhu je udržován vývojáři, vydavateli a poskytovateli mobilních aplikací. Mobilní aplikace lze rozdělit do různých kategorií, a to například dle jejich použití: komunikace (prohlížení internetu, emaily), hry a multimédia (přehrávače, videa, foto), cestování (překladače, mapy, počasí) nebo nástroje (správa profilu, adresář, správce úloh). Dále je lze rozdělit podle vývoje na: nativní aplikace, hybridní aplikace a multiplatformní aplikace [8].

2.1.3 Přehled mobilních operačních systému

Za zmínku zde také stojí operační systémy, které byly spíše populární dříve, a to v době od roku 2009 - do konce roku 2011. Mezi takové systémy patří: Windows Phone, BlackBerry OS, Symbian OS. Viz graf na obrázku 2.1 [12].



Obrázek 2.1: Mezi nejrozšířenější mobilní operační systémy patří k roku 2022 Android, který na trhu zastupuje 72.2% využívaných systémů a iOS s 27.1%[2]

Android OS - Operační systém Android pro mobilní zařízení vyvinula společnost Open Handset Alliance, kterou vede společnost Google. Google představil distribuci Android v listopadu 2007.

iOS OS - iOS (dříve iPhone OS) je mobilní operační systém vyvinutý společností Apple Inc. a distribuovaný výhradně pro hardware společnosti Apple. Jedná se o operační systém, který pohání iPhone, iPad, iPod Touch a Apple iPhoneTV. Je uzavřený, proprietární a postavený na otevřeném zdrojovém kódu Darwin core OS. Systém iOS propagoval nový styl uživatelského ovládání pro zařízení s malou obrazovkou a omezeným vstupem, konkrétně přímou manipulaci. Gesta založená na dotyku, jako je přejetí prstem, klepnutí a podržení se používají k ovládání prvků rozhraní na obrazovce a k provádění operací. Akcelerometry podporují další fyzikální gesta, jako je třesení a otáčení orientace displeje zařízení. Operační systém iOS je systém UNIXového typu.

Windows Phone - Windows Phone je proprietární operační systém pro chytré telefony vyvinutý společností Microsoft. Na trh byl uveden v roce 2010 pod názvem Windows Phone 7. Různí výrobci hardwaru, včetně společností HTC, Samsung, LG a Nokia, vyvíjejí nové zařízení se systémem Windows Phone. Jádro systému Windows Phone zpracovává nízkoúrovňový přístup k ovladačům zařízení i základní funkce jakými jsou: základní zabezpečení, správa sítě a úložiště. Nad jádrem se nacházejí tři knihovny: App pro správu aplikací, model uživatelského rozhraní pro správu uživatelského rozhraní a modul Cloud Integration pro správu aplikací, webové vyhledávání prostřednictvím služby Bing, služby určování polohy a notifikace. Aplikace směřující na API zahrnují Sliverlight, XNA, HTML/JavaScript a Common Language Runtime (CLR), který podporuje jazyky C# nebo VB .Net. Samotné jádro je proprietární. Windows OS byl navržen pro vestavěná zařízení, který kombinuje Windows Embedded CE 6.0 R3 a Windows Embedded Compact 77.

Blackberry OS - Operační systém BlackBerry OS vyvíjí společnost Research in Motion (RIM) pro své chytré telefony a tablety BlackBerry. Operační systém BlackBerry OS 1.0 debutoval v lednu 1999 jako součást pagerových/poštovních zařízení BlackBerry. Jednou z hlavních předností zařízení BlackBerry je jejich schopnost zpracovávat podnikovou elektronickou poštu. Systém BlackBerry OS podporuje jazyk Java Mobile Information Device Profile (MIDP) a Wireless Application Profile (WAP). Operační systém BlackBerry OS původně podporoval aplikace napsané v jazyce C++. Jeden typ aplikací je běhové prostředí Mobile Data Services (MDS), což je kontejner pro zpracování a zobrazení dat obvykle odesílaných z podnikového systému uživatele. Java ME byla rovněž podporována a používala se k vytváření aplikací s přístupem k rozhraním API operačního systému, která poskytují přístup ke standardnímu uživatelskému rozhraní, widgetům a různým službám operačního systému.

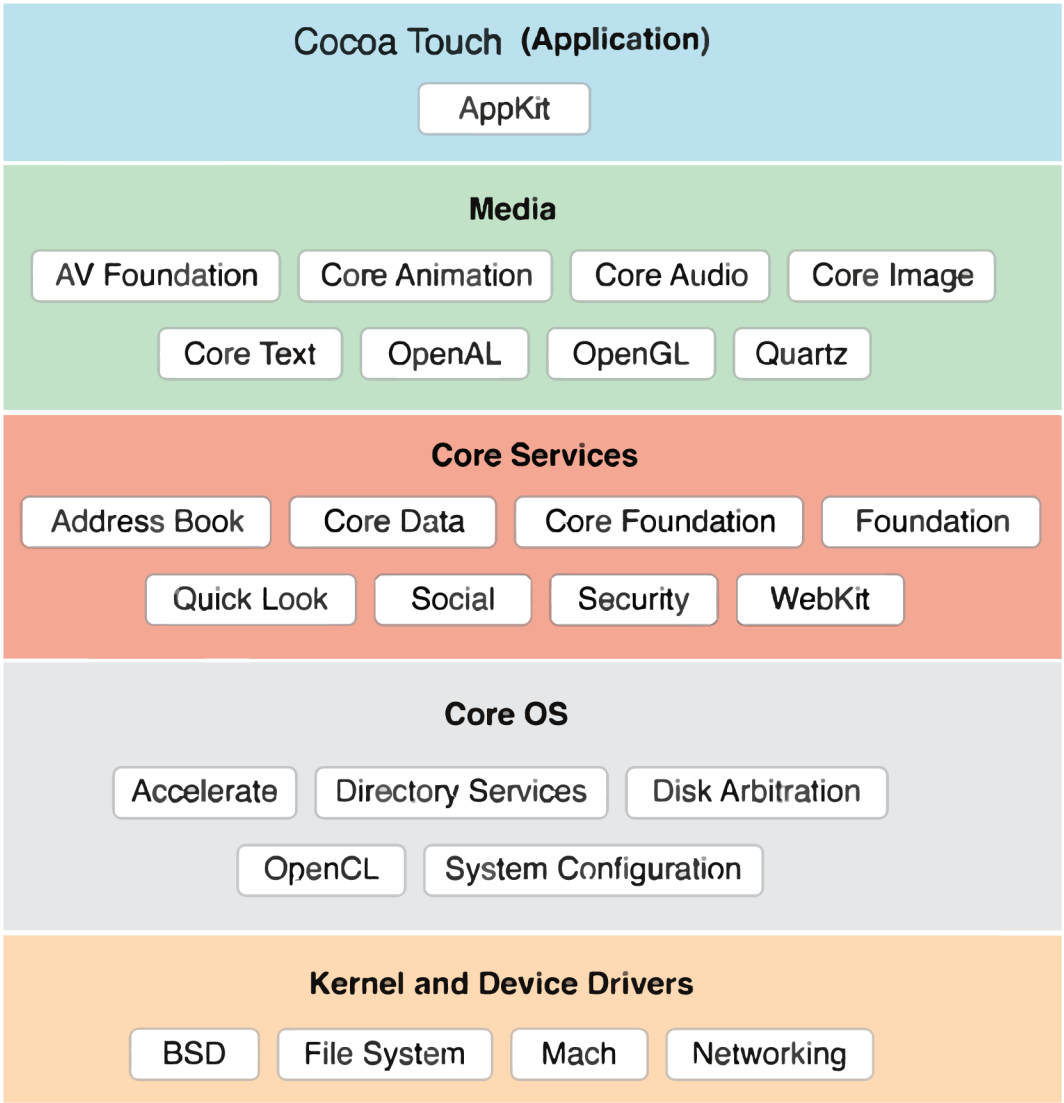
Symbian OS - Symbian je operační systém pro mobilní zařízení, vyvinutý společností Nokia. Původně se jednalo o grafický operační systém EPOC pro mobilní telefony a přenosná zařízení. V roce 1998 PSION, Nokia, Ericsson a Motorola vytvořily operační systém Symbian OS. V současné době Symbian Foundation je však provozována a udržována společností Nokia, která poskytuje přístup k systému Symbian prostřednictvím standardních licenčních smluv. Je postaven na jádru nanokernel/mikrokernel se základními funkcemi a ovladači obrazovky. Základní služby jsou umístěny nad jádrem a zahrnují nízkoúrovňové knihovny, mediální rámce, XML, správu souborového systému a hardwarovou abstrakci. Služby operačního systému zajišťují komunikaci, telefonní sítě, multimédia a grafiku. Ty podporují vrstvu aplikačních služeb s rozhraními API zaměřenými na aplikace pro vývoj

[2]Dostupné z: https://gs.statcounter.com/os-market-share/mobile/worldwide

a vrstvu pro správu uživatelského rozhraní. Nad službami operačního systému je také zahrnuta JVM (Java ME) vrstva. Nokia poskytuje SDK pro vývoj Symbianu, které podporují různé jazyky, včetně C++ a Java [1].

2.1.4 Architektura iOS

Architektura iOS se skládá ze čtyř různých vrstev, které jsou zobrazeny na obrázku 2.2.



Obrázek 2.2: Architektura iOS[3] s ukázkou jednotlivých vrstev

Vrstva Cocoa Touch - Vrchní vrstva architektury iOS, která obsahuje klíčovou sadu uživatelského rozhraní frameworku a definuje základní aplikační infrastrukturu. Dále poskytuje základní funkce, jako např. multitasking a dotykové vstupy. Obsahuje rámce jako jsou EventKit framework, GameKit framework, MapKit framework, MessageKit framework [1].

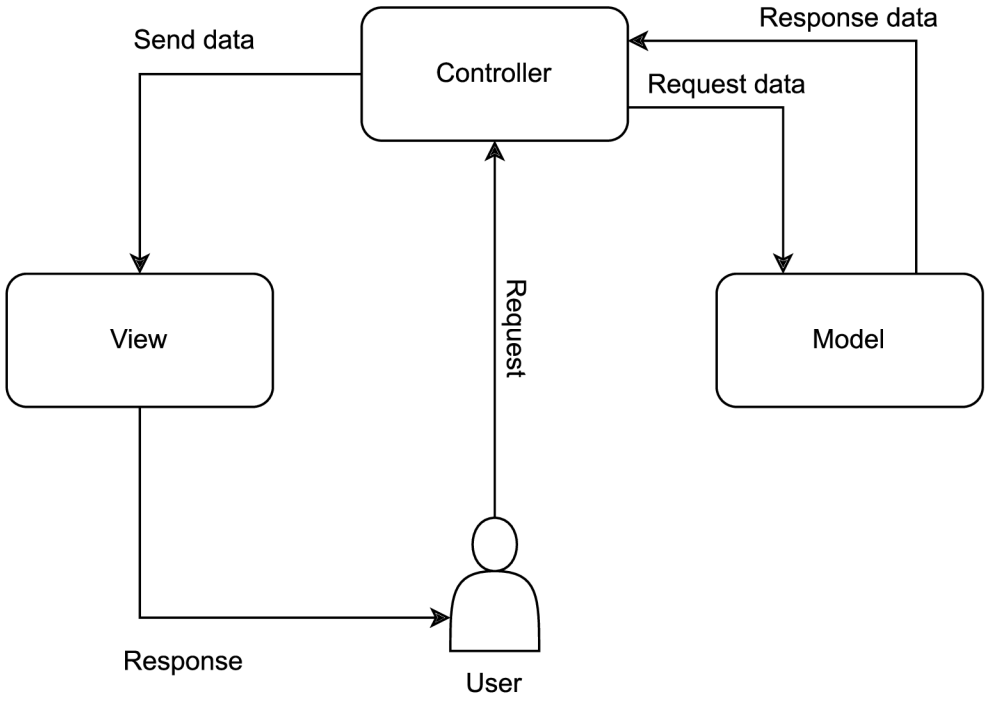
[3]Dostupné z: https://www.dotnettricks.com/learn/xamarin/understanding-xamarin-ios-build-native-ios-app

Vrstva mediálních služeb - Vrstva médií poskytuje systému iOS všechny funkce pro podporu zvuku, videa, animace a grafiky. Dále obsahuje rámce jako Core Audio framework, Core Text framework, Core Image framework [1].

Vrstva základních služeb - Pod mediální vrstvou se nachází základní vrstva služeb, která poskytuje služby používané ve vyšších vrstvách. Obsahuje rámce jako jsou např. rámec Address Book, Core Location framework, Core Telephony framework [1].

Základní vrstva operačního systému - Nejnižší vrstva architektury iOS je Core OS. Zajišťuje zpracování, propojení a bezpečnostní podporu pro aplikace iOS. Obsahuje rámce jako jsou např. Accelerate framework, System Framework, External Accessory a bezpečnostní rámec [1].

Při vývoji mobilních aplikací na iOS je nejčastěji využívána architektura Model-view-controller. Ukázka architektury je na obrázku 2.3.
Model-view-controller Hlavním cílem MVC je oddělit logickou struktura kódu a rozdělit ji na datovou vrstvu, prezentační vrstvu a třetí vrstvu, která všechny tyto části spojuje. Části MVC jsou následující:
Model - reprezentuje veškerá data a datovou logiku, se kterými uživatel pracuje. Představuje data, která se přenášejí mezi View a Controllerem.
View - zobrazuje celkovou logiku uživatelského rozhraní aplikace. Reprezentuje data uložená v Modelu. Zobrazuje veškeré UI komponenty, se kterými uživatel pracuje.
Controller - funguje jako rozhraní mezi Modelem a View, které zpracovává všechnu obchodní logiku. Zpracovává příchozí požadavky, manipuluje s daty pomocí Modelu a komunikuje s View při vykreslování výstupu [4].



Obrázek 2.3: Ukázka architektury MVC[4] s popisem komunikace mezi komponentami

Hlavní důvody, proč je iOS postavený kolem vrstvené architektury, jsou následující:

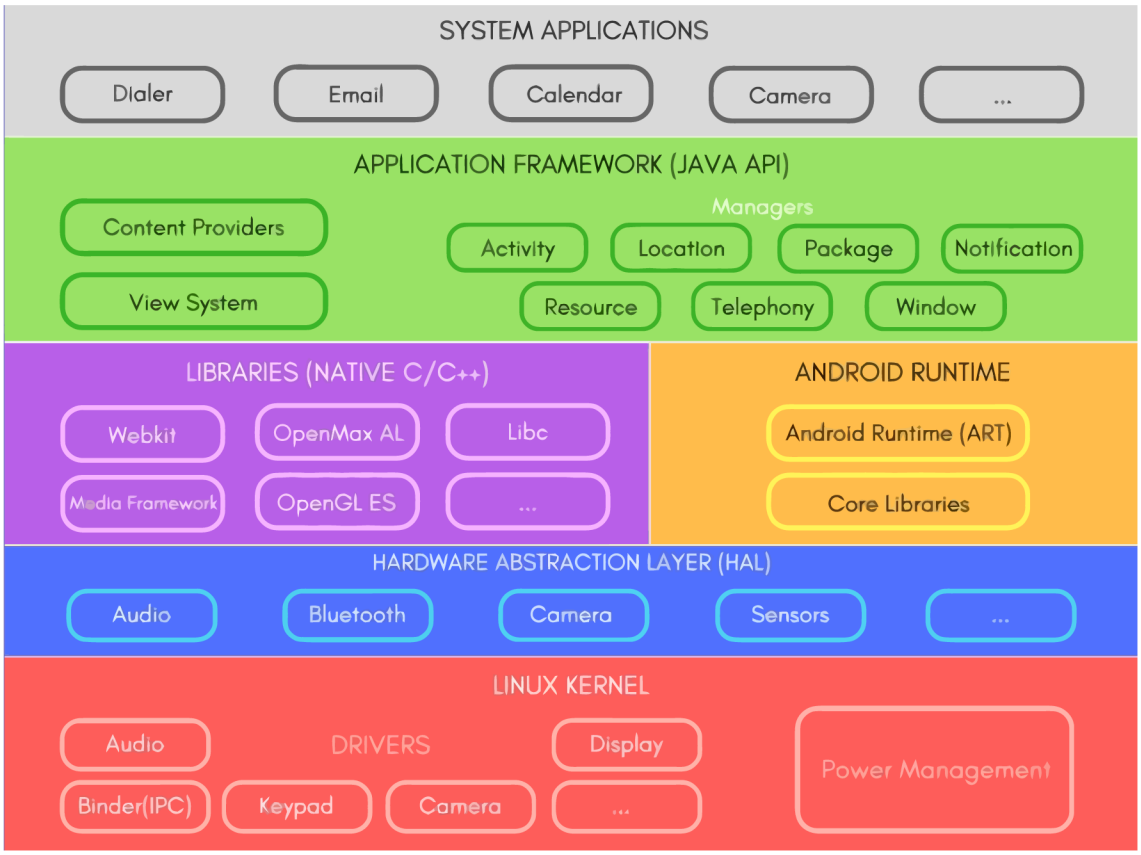
1. Tato architektura rozděluje design na malé části, kde každá vrstva přispívá ke službám poskytovanými nižšími vrstvami takovým způsobem, aby nejvyšší vrstva poskytovala úplnou sadu služeb pro správu komunikace a provozování aplikací.

2. Aplikace iOS nekomunikují s podkladovým hardwarem přímo, ale prostřednictvím souboru přesně definovaných systémových rozhraní.

3. Ve srovnání s jinými styly architektury iOS vítězí díky snadnosti vývoje, a to především z důvodu, že tento vzor je velmi dobře známý a není tolik složitý na implementaci. Funkce jsou totiž rozděleny do jednotlivých zmíněných vrstev [2].

2.1.5 Architektura Android

Operační systém Android využívá přístup založený na vrstvené architektuře, která je znázorněna na obrázku 2.4. Mezi tyto vrstvy patří:



Obrázek 2.4: Architektura Android[5] s ukázkou jednotlivých vrstev

Jádro Linuxu - tato vrstva je základem platformy Android. Obsahuje všechny nízkoúrovňové ovladače pro podporu různých hardwarových komponent. Android Runtime se

[4]Dostupné z: https://www.codecademy.com/article/mvc
[5]Dostupné z: https://www.studytonight.com/android/android-architecture

spoléhá na Linuxové jádro pro základní systémové služby, jako jsou: paměť, správa procesů, síťový zásobník, model ovladače, zabezpečení [1].

Hardwarová abstrakční vrstva (HAL) - poskytuje abstrakci mezi hardwarem a zbytkem softwarového zásobníku [1].

Android Runtime (ART) - je navržen pro provoz aplikací v omezeném prostředí, které má omezený výkon baterie, výpočetní a paměťový výkon. Od systému Android 5.0 běží každá aplikace ve vlastním procesu v rámci vlastní instance virtuálního stroje ART, což zvyšuje důležitost správy procesů. ART používá soubory DEX, což je typ bajtového kódu, speciálně navržený pro systém Android, který pomáhá ART efektivněji spravovat paměť. Obsahuje sadu základních knihoven, které vývojářům umožňují psát aplikace pro Android pomocí programování v jazyce Java. Před systémem Android 5.0 se jako běhové prostředí systému Android používal Dalvik. ART je schopen provádět kompilaci AOT (Ahead-of-time) i JIT (Just-in-time). Má také velmi efektivní garbage collection [9].

Knihovny - vývojářům jsou k dispozici prostřednictvím aplikačního frameworku Android. Obsahuje knihovny v jazyce C/C++ používané komponentami systémů Android. Mezi knihovny patří:

1. Knihovna SQLite je používaná pro ukládání dat a nenáročná na paměť mobilního telefonu a provádění úloh.

2. Knihovna WebKit poskytuje především engine pro prohlížení webu a dalších souvisejících funkcí.

3. Knihovna správce ploch je zodpovědná za vykreslování oken a kreslení ploch různých aplikací na obrazovce.

4. Knihovna media framework poskytuje mediální kodeky pro zvuk a video.

5. Knihovny OpenGl (Open Graphics Library) a SGL (Scalable Graphics Library) jsou grafické knihovny pro 3D, respektive 2D vykreslování.

6. Knihovna FreeType slouží k vykreslování písem [9].

Aplikační Framework - Jedná se o soubor rozhraní API, napsaných v jazyce Java, který vývojářům poskytuje přístup ke kompletní sadě funkcí operačního systému Android. Vývojáři mají plný přístup ke stejným rozhraním API frameworku, která používají základní aplikace a tím pádem mohou více rozšířit funkce své aplikace. Dále umožňuje a zjednodušuje opakované použití základních komponent a služeb, jako např [1]:

1. Správce aktivit: Spravuje životní cyklus aplikací.

2. Správce oken: Spravuje okna a kreslící plochy a je abstrakcí knihovny správce ploch.

3. Zprostředkovatelé obsahu: Umožňuje aplikaci přistupovat k datům z jiných aplikací nebo sdílet vlastní data, tj. poskytuje mechanismus pro výměnu dat mezi aplikacemi.

4. Systém zobrazení: Obsahuje stavební bloky uživatelského rozhraní, které se používají k vytvoření uživatelského rozhraní aplikace, včetně seznamů, mřížek, textů, rámečků, tlačítek a také provádí správu událostí prvků uživatelského rozhraní.

5. Správce balíčků: Spravuje různé druhy informací souvisejících s balíčky aplikací, které jsou v zařízení aktuálně nainstalovány.

6. Správce telefonie: Umožňuje aplikaci využívat telefonní funkce zařízení.
7. Správce prostředků: Zajišťuje přístup k nekódovým zdrojům (lokalizované řetězce, bitové mapy, grafika a rozvržení).
8. Správce umístění: Zabývá se možnostmi informovanosti o poloze.
9. Správce oznámení: Umožňuje aplikacím zobrazovat vlastní upozornění ve stavovém řádku [9].

Aplikace - Na vrcholu zásobníku aplikací pro systém Android se nacházejí systémové aplikace a další aplikace, které si uživatelé mohou stáhnout z oficiálního obchodu Play systému Android, známého také jako Obchod Google Play. Tato vrstva využívá všechny vrstvy pod ní, pro správné fungování těchto mobilních aplikací [1].

2.1.6 Porovnání Android a iOS

Android je velmi dobře přizpůsobitelný operační systém s otevřeným zdrojovým kódem, který lze snadno změnit, upravit a přizpůsobit jakékoli funkci. Systém Android běží na Linuxovém jádře. Podporuje jazyky: Java, C, C++, XML, Jazyk assembleru, Python, Shell script. **iOS** je operační systém s uzavřeným zdrojovým kódem, ale s otevřeným zdrojovým kódem pro komponenty. Apple používá vlastní jádro založené na BSD a programuje téměř všechny své knihovny od nuly, díky tomu má jedinečný výkon.

Výkon - Android je platforma založená na Javě a k provádění aplikací používá Java Virtual Machine (Dalvik). Java byla původně považována za pomalou a těžkopádnou platformu, ale později byla optimalizována a nyní je její výkon extrémně rychlý. iPhone OS je napsán pomocí Objective C, což je objektově orientovaná verze jazyka C, která používá zprávy. Existuje systém předávání zpráv nazvaný *objcmsgSend()*, který je kritickou součástí softwaru pro cokoli napsaného pro OS X na Macu nebo iPhonu. Výkon na těchto zařízeních je rychlý.

Otevřený / uzavřený zdrojový kód - Společnost Apple nabízí uzavřený, proprietární systém s bezkonkurenčním marketingem, vynikajícím uživatelským rozhraním a silnou vazbou mezi hardwarem a softwarem. Google dává přednost přístupu open-source k vývoji nových aplikací. Oproti Androidu, iOS nenabízí zpracování na pozadí pro aplikace třetích stran. Lze vytvořit libovolnou aplikaci využívající libovolné funkce.

Zabezpečení - V systému iOS existují určitá privilegia, která omezují přidávání nového softwaru pro uživatele. Každá aplikace běží na jediném jádře UNIX. V případě, že se vyskytne nějaký problém ze strany spuštěné aplikace, může to ovlivnit celý systém, zatímco v systému Android běží každá aplikace samostatně. Pro každou funkci jsou předem definována oprávnění a autorizace. Z hlediska bezpečnosti je tedy Android pro uživatele mnohem lepší než iOS [9].

2.1.7 Nativní aplikace

Nativní aplikace jsou napsané a vyvinuté speciálně pro konkrétní mobilní operační systém. K vytvoření skutečných nativních aplikací pro Android je nutné použít programovací jazyk Java, pro iOS a Windows Phone programovací jazyk Objective C. Společnými klíčovými vlastnostmi nativních aplikací je, že tyto aplikace mají neomezený přístup k hardwaru zařízení a podporují všechna uživatelská rozhraní a interakce dostupné v příslušném mobilním

operačním prostředí. Nativní aplikace jsou vhodnější pro tvorbu obsahu díky výkonu a přístupu k hardwaru. Nativní aplikace spolupracují s operačním systémem zařízení způsobem, který jim umožňuje pracovat rychleji a pružněji než alternativní typy aplikací. Jako příklady nativních aplikací lze uvést: Spotify, WhatsApp, Tesla, Waze, Pokemon GO.
Mezi **výhody** nativních aplikací patří:

1. Široké funkční možnosti díky využití možností základního zařízení.
2. Rychlý a svižný výkon softwaru.
3. Uživatelské rozhraní, které lépe odpovídá uživatelskému prostředí operačního systému.
4. Neomezený přístup k nativním funkcím zařízení.
5. Nativní aplikace jsou také méně zavislé na knihovnách a rozhraní.

Mezi **nevýhody** nativních aplikací patří:

1. Více základen kódu, protože každé zařízení má vlastní verzi aplikace.
2. Náklady na další vývojáře, kteří vytvářejí a spravují kódovou základnu pro každou platformu.
3. Čas strávený na více sestaveních pro jednotlivé platformy při každé aktualizaci funkcí.
4. Aktualizace a úpravy je nutné provádět zvlášť v každém kódu [6].

2.1.8 React Native

React Native je framework JavaScriptu pro vytváření nativně vykreslujících mobilních aplikací pro iOS a Android. Je založen na Reactu, knihovně JavaScriptu společnosti Facebook pro tvorbu uživatelských rozhraní, ale místo na prohlížeč se zaměřuje na mobilní platformy. React Native umožňuje snadný vývoj pro Android i iOS současně. Nabízí rychlejší vývoj mobilních aplikací a efektivnější sdílení kódu napříč systémy iOS, Android a webem. Zároveň to není na úkor komfortu koncového uživatele nebo kvality aplikace. React navíc pracuje odděleně od hlavního vlákna uživatelského rozhraní, takže aplikace může zachovat vysoký výkon. Obsahuje také podporu pro asynchronní provádění operací mezi jednotlivými uživateli. Kódem JavaScriptu v aplikaci a nativní platformou umožňuje využití vláken v nativních modulech a lze tak provádět mnoho různých operací na pozadí a zároveň neblokovat uživatelské rozhraní [17].

2.1.9 Životní cyklus vývoje aplikace

Proces životního cyklu vývoje softwaru je používaný při vývoji jakéhokoli softwarového produktu. Existuje mnoho různých modelů životního cyklu. Mezi tyto modely patří: Vodopádový model, Spirálový model, model Prototypování. Každý model je popsán vlastní posloupností činností. Vývojové kroky, resp. činnosti se mohou v každém modelu lišit, ale všechny modely zahrnují: identifikaci požadavků, analýzu a návrh, vývoj a prototypování, testování a nasazení, údržbu.

1. Fáze identifikace: V první fázi se provádí průzkum již existujících aplikací. Dále se shromažďují nápady a návrhy. Tyto nápady a návrhy mohou pocházet od zákazníka nebo od vývojářů. Ve výsledné aplikaci to byl nejdříve průzkum existujících řešení a poté tvorba nápadu, co všechno by měla aplikace obsahovat.

2. Fáze návrhu: V této fázi se nápady zpracovávají a tým pro tvorbu aplikací je přepracovává do podoby počátečního návrhu aplikace. Funkčnost aplikace je rozdělena do modulů a do prototypů, které společně budou tvořit výslednou aplikaci. V této fázi jsou také definovány funkční požadavky. Ve výsledné aplikaci to byl první návrh řešení a wireframe schéma systému.

3. Fáze vývoje: V této fázi se aplikace vyvíjí. Programování různých modulů téhož prototypu může probíhat paralelně. Proces vývoje může probíhat ve 2 fázích: 1. Programování funkčních požadavků, 2. Programování uživatelského rozhraní. Ve výsledné aplikaci to byl prvotní prototyp se statickými daty.

4. Fáze vytváření prototypu: V této fázi se analyzují funkční požadavky každého prototypu a prototypy se upravují podle potřeby, jsou testovány a zaslány klientovi pro zpětnou vazbu. Po získání zpětné vazby jsou požadované změny implementovány prostřednictvím vývojové fáze. Když je druhý prototyp hotov, je integrován s prvním prototypem, otestován a poté odeslán klientovi. Fáze vývoje prototypování a testování se následně opakují, dokud není hotový finální prototyp. Práce ve fázi prototypování je dokumentována a poté předána do fáze testování. Ve výsledné aplikaci to bylo vytvoření finálního prototypu aplikace s daty z databáze.

5. Fáze testování: Testování je jednou z nejdůležitější fází vývoje. Testování prototypů je prováděno na emulátoru/simulátoru, po němž následuje testování na skutečném zařízení. Na skutečném zařízení by mělo testování být provedeno na více operačních systémech a také na více verzích systému, a to pro odhalení případných chyb. Ve výsledné aplikaci to bylo otestování aplikace na základní škole.

6. Fáze nasazení: Po dokončení testování a získání konečné zpětné vazby od klienta se aplikace spustí a je připravena k nasazení. Poté se nahraje do daného obchodu s aplikacemi k použití pro uživatele. Ve výsledné aplikaci k nasazení teprve dojde.

7. Fáze údržby: Zpětná vazba je od uživatelů průběžně shromažďována a následně jsou provedeny požadované změny ve formě oprav chyb nebo vylepšení produktu. Veškeré opravy chyb, nové funkce nebo vylepšování výkonu by měly být poskytovány ve formě aktualizací aplikace [15].

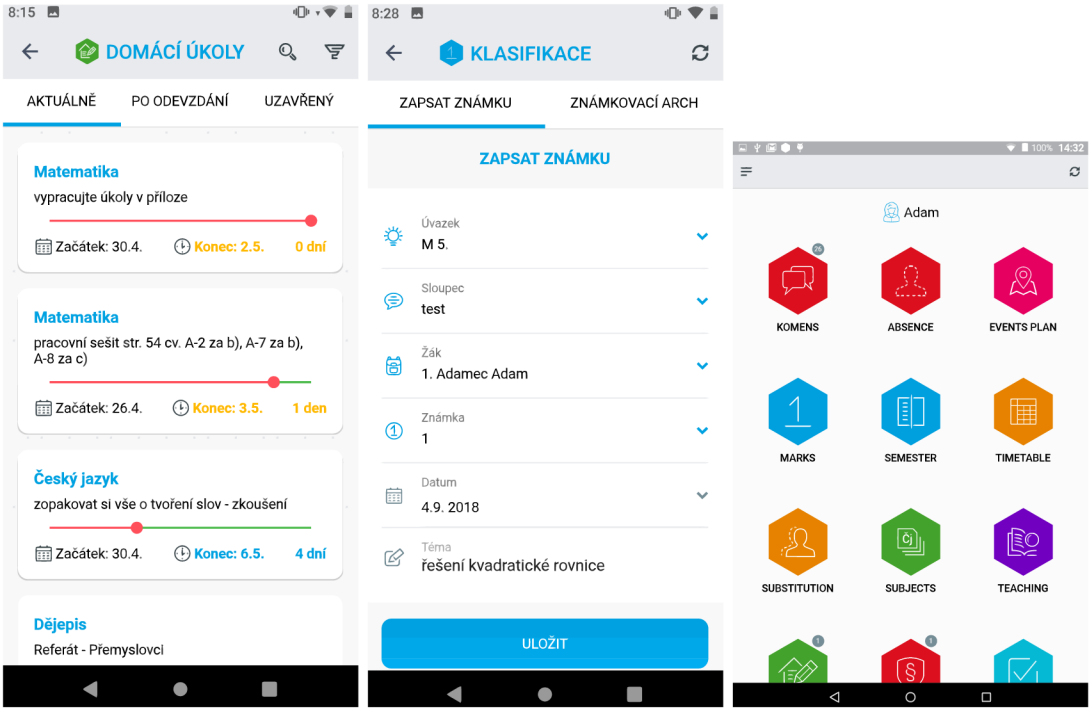
2.2 Existující řešení

Jedním z důležitých kroků této práce je průzkum již existujících aplikací na trhu. V následující kapitole jsou zpracovány popisy vybraných aplikací, které jsou vhodné pro usnadnění výuky a komunikace se studenty. S ohledem na rozsah práce zde jsou popsány 3 hlavní aplikace.

2.2.1 Bakaláři[6]

Tato aplikace se řadí mezi nejrozšířenější školní systém v České republice. Napomáhá komunikaci mezi školou, studenty a rodiči. I když se aplikace řadí mezi nejrozšířenější školní

systémy, má dle recenzí pouze průměrné hodnocení. Z recenzí dále vyplývá, že je aplikace poměrně pomalá a často problémová na použití. Aplikaci Bakaláři lze využít ve webové i mobilní formě. Na obrázku 2.5 lze vidět ukázky z mobilní aplikace.



Obrázek 2.5: Ukázka z aplikace Bakaláři[8] ,kde na prvním obrázku je vidět přehled domácích úkolů, na druhém je přehled klasifikace žáka a na třetím je zobrazena domovská obrazovka

Mobilní aplikace podporuje:

1. Osobní údaje žáka – dle výběru mohou rodiče kontrolovat nejen zda škola eviduje správné údaje (telefony, adresy...), ale zda systém umožňuje i přímé ohlašování změn přes webové rozhraní.

2. Průběžná klasifikace – přehled všech průběžně zadaných známek; lze volit, zda zobrazovat i témata, váhy známek, poznámky, průměr apod.

3. Průběžná docházka – napojení na modul Třídní kniha, evidence zameškaných hodin (zobrazení po dnech, měsících, ale přímo i po vyučovacích hodinách).

4. Rozvrh – tabulka rozvrhu, a to buď stálý rozvrh nebo aktuální rozvrh včetně změn v souvislosti se suplováním.

5. Domácí úkoly – přehled domácích úkolů (aktivních i ukončených). Vyučované předměty – seznam předmětů s uvedením učitelů, kteří daný předmět žáka vyučují.

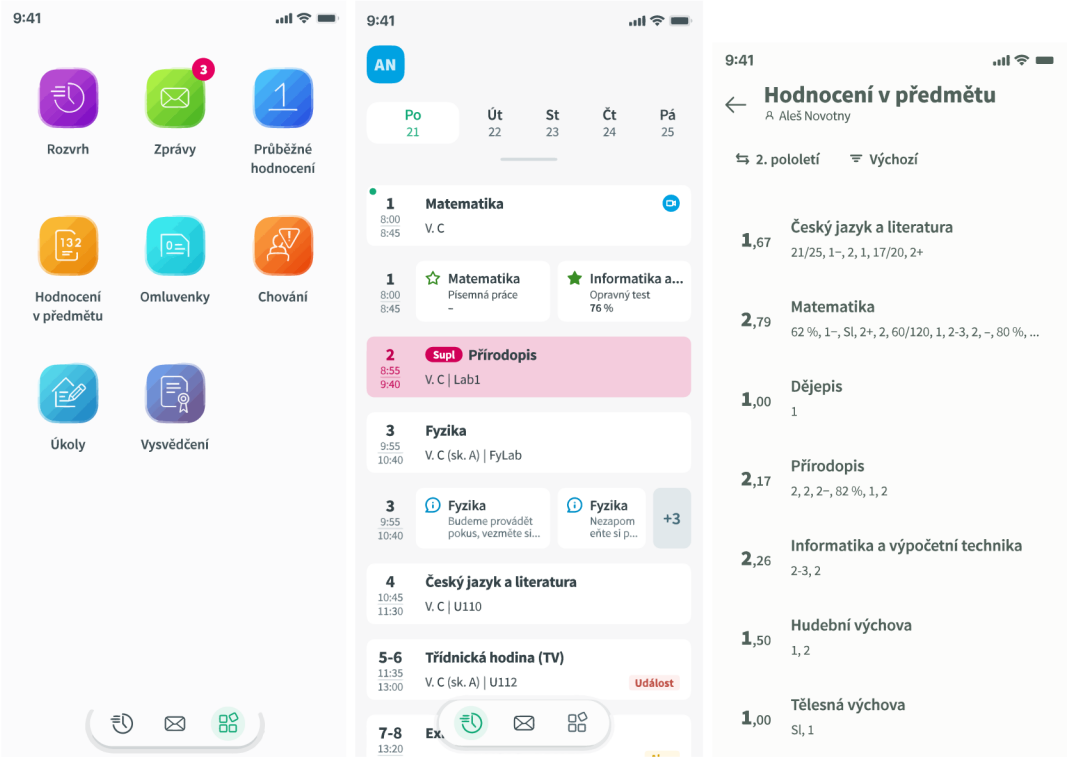
6. Přehled výuky – seznam odučených témat.

[6]Dostupné z: https://www.bakalari.cz/
[8]Dostupné z: https://play.google.com/store/apps/details?id=cz.bakalari.mobile

7. Předměty - detaily a informace o učiteli a tématech.
8. Domácí úkoly – přehled domácích úkolů (aktivních i ukončených).
9. Režim offline podpory prohlížení rozvrhu a suplování.

2.2.2 Škola online[9]

Další velice rozšířený informační systém pro školy, který umožňuje rychle a efektivně zpracovávat veškerou školní agendu při zachování vysokého uživatelského komfortu. Lze využít ve webovém i mobilním prostředí, viz obrázek 2.6. Podobně jako aplikace Bakaláři má ale dle recenzí horší, než průměrné hodnocení.



Obrázek 2.6: Ukázka z aplikace Školy online[11] ,kde na prvním obrázku je vidět domovská obrazovka, na druhém je rozvrh hodin na aktuální den a na třetím obrázku je zobrazena klasifikace žáka

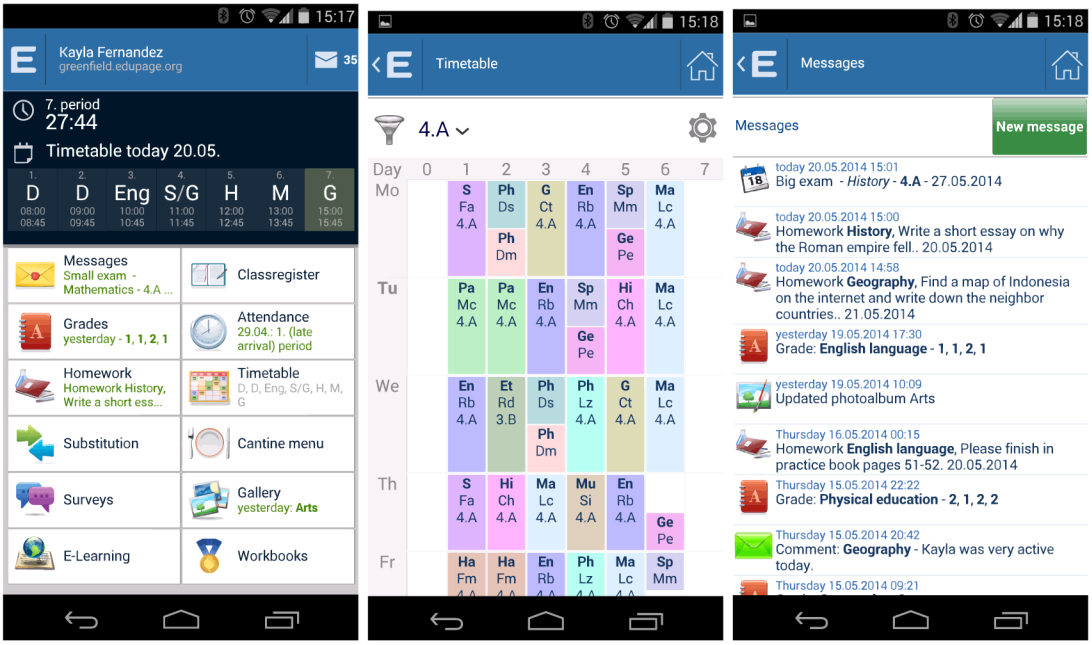
Žáci a rodiče mohou aplikaci využít především k prohlížení aktuálního rozvrhu, jeho změn a případné suplování. Dále si mohou zobrazit školní akce, poznámky k hodinám, přehled docházky, výchovná opatření, klasifikaci žáka a rodiče mohou také omlouvat absence žáka a komunikovat s učiteli pomocí zpráv. Mohou si také zobrazit veškeré úlohy, případně naplánované písemky. Dále si žáci a rodiče mohou prohlédnout průběžnou klasifikaci, pololetní klasifikaci a elektronické vysvědčení.

[9]Dostupné z: https://www.skolaonline.cz/
[11]Dostupné z: https://play.google.com/store/apps/details?id=cz.skolaonline.mobile

Z pohledu školy a učitele odpadá nutnost nosit do každé třídy notebook, případně tištěnou třídní knihu pro zapisování absence či hodnocení žáků. Pro toto zapisování stačí tablet nebo chytrý telefon. Učitelé mohou především sledovat změny rozvrhu, zadávat známky, zapisovat do třídní knihy, číst zprávy, komunikovat s rodiči a žáky, omlouvat absence žáků, zadávat hodnocení chování a plánovat písemky.

2.2.3 EduPage[12]

Jedná se o další rozšířenou aplikaci pro školní využití. Lze využít ve webovém i mobilním prostředí viz obrázek 2.7.



Obrázek 2.7: Ukázka z aplikace EduPage[14] ,kde na prvním obrázku je zobrazena domovská obrazovka, na druhém je zobrazen rozvrh hodin na týden a na třetím jsou zobrazeny zprávy ze školy

Pro žáka aplikace umožňuje zobrazení domácích úkolů, předešlých písemek, přihlašování a odhlašování obědů, zobrazení rozvrhu a suplování v rozvrhu, odesílání zpráv spolužákům, přehled známek a testů, přehled docházky i absencí, zobrazení školního fotoalba a přehled školních anket.

Z pohledu učitele lze aplikaci využít pro zapisování učiva, zapisování docházky žáků i známek, zadávání úloh a zobrazení přehledu aktuálního rozvrhu.

2.2.4 Další mobilní aplikace se zaměřením pouze na určitý předmět

Duolingo[15] je volně dostupná aplikace ke stažení, která je především určena pro výuku a zdokonalení se v cizích jazycích. V aplikaci se lze naučit či zdokonalit v mnoha cizích jazycích, jako jsou např: Angličtina, Němčina, Francouzština, Portugalština, Španělština

[12] Dostupné z: https://mobile.edupage.org/#!/home
[14] Dostupné z: https://play.google.com/store/apps/details?id=air.org.edupage

a další.

Světová geografie[16] je zeměpisná aplikace, která umožňuje uživatelům naučit se vše o zemích světa. U jednotlivých zemí se uživatelé mohou naučit mapu země, jazyk, hlavní město, náboženství, měnu a další. Aplikace je koncipována jako kvízová hra, která uživateli umožňuje učit se zabávnou formou. Jednotlivé kvízy lze konfigurovat a také měnit oblasti znalostí.

Microsoft teams[17] jsou dostupné v mobilní verzi a také umožňují škole komunikaci se studenty, případně rodiči. Dále aplikace nabízí možnosti sdílení souborů, týmovou spolupráci na projektech i úkolech, plánování školních akcí nebo registrování na cvičení a zkoušky. Aplikace také umožňuje zadávání úkolů žákům. Lze také provádět skupinové hovory, online prezentace a kurzy.

2.2.5 Uživatelské rozhraní

Uživatelské rozhraní hraje klíčovou roli v rozhodování, jestli uživatel bude danou aplikaci využívat. Z daného důvodu by každá aplikace měla být navržena se zaměřením na lidské vnímání a lidský faktor. V případě, že je rozhraní správně strukturováno, uživatelé budou aplikaci pravděpodobně uspokojivě hodnotit a tím pádem se k využívání aplikace budou rádi vracet. Návrh softwaru, zejména návrh uživatelského rozhraní, je přirozeně iterativní, protože software umožňuje změny tak dlouho, dokud vývojáři nechtějí provádět změny za účelem vylepšení nebo experimentů. Mezi základní principy dobrého návrhu uživatelského rozhraní patří [10]:

1. Jednoduchost: Zajistit co nejjednodušší rozhraní. Používat postupné odhalování, skrývání věcí, dokud nejsou opětovně potřeba. Představit nejprve běžné a nezbytné funkce, důležité funkce uvádět na viditelném místě a složitější či méně často používané funkce skrýt.

2. Reakčnost: Požadavek uživatele je třeba vyhodnotit rychle. Zpětná vazba by měla být vizuální.

3. Předvídatelnost: Předvídatelnost snižuje počet chyb a umožňuje rychlejší dokončení úkolů. Všechna očekávání uživatele by měla být splněna jednotně a úplně.

4. Přehlednost: Rozhraní musí být jasné z hlediska vizuálního vzhledu, koncepce a znění. Vizuální prvky by měly být srozumitelné, vztahující se k reálným pojmům a funkcím uživatele. Metafory nebo analogie by měly být realistické a jednoduché.

5. Estetická příjemnost: Dobrý design by měl kombinovat výkon, funkčnost a jednoduchost s příjemným vzhledem. Vizuálně příjemná kompozice je atraktivní pro oko, která přitahuje pozornost podprahově a tím jasně a rychle předává sdělení.

6. Konzistence: Systém by měl vypadat, chovat se a fungovat po celou dobu stejně. Stejná činnost by měla vždy přinést stejný výsledek, funkce prvků by se neměla měnit.

7. Ovladatelnost: Ovládání by mělo pro uživatele znamenat pocit kontroly, ve smyslu, že systém reaguje na akce uživatele. Uživatel by neměl mít pocit, že systém ovládá jeho.

[15] Dostupné z: https://www.duolingo.com/
[16] Dostupné z: https://play.google.com/store/apps/details?id=com.age.wgg.appspot
[17] Dostupné z: https://www.microsoft.com/en-us/microsoft-teams/group-chat-software

8. Efektivnost: Pohyby očí a rukou nesmí být zbytečné. Pozornost člověka musí být v případě potřeby upoutána relevantními prvky obrazovky. Postupné pohyby očí mezi prvky obrazovky by měly být předvídatelné, zřejmé a krátké.

9. Přímočarost: Úkoly by se měly provádět přímo. Dostupné alternativy by měly být viditelné, čímž se sníží mentální zátěž uživatele [3].

2.3 Využití mobilních zařízení ve škole

V kapitole je vysvětlena důležitost a aktuálnost tématu této práce, tedy mobilní studentské aplikace. Zároveň je popsáno a vysvětleno, v čem mohou být mobilní technologie ve školách prospěšné a jak mohou být využívány [7].

Digitální technologie zasahují do oblasti vzdělávání a školství. Technologické kompetence, které představují schopnost rychle se orientovat a vzdělávat v oblasti technologií, jsou určitě jednou z těchto důležitých kompetencí [11]. Mobilní technologie mohou do výuky nebo vzdělávání obecně přinést úplně nový rozměr. Z nejdůležitějších přínosů lze jmenovat větší přenesení odpovědnosti, ale i pozornosti na žáka. Mobilní technologie v žácích mohou probouzet vyšší míru snahy o samostudium a vlastní řízené učení.

Celkově může práce s technologiemi ve výuce přinést vyšší míru angažovanosti i zapojení žáků, jelikož výuka tak pro ně může být mnohem více zajímavá a záživná. To s sebou může i často přinést vyšší efektivitu výuky. Efektivita výuky, záživnost nebo například i pravidelná zpětná vazba, které může být žákům dávána díky technologiím mnohem rychleji, přispívá k celkové spokojenosti studentů. Vše zmíněné tedy mohou mobilní technologie ovlivňovat v pozitivním i negativním směru [7].

Rozměr, který práce s technologiemi přináší spočívá samozřejmě především v tom, jaké vzdělávací nebo školní aplikace můžou školy a vzdělávací zařízení využívat. Více o dosud dostupných vzdělávacích aplikacích je podrobně popsáno v části 2.2. S možnostmi, které mohou mobilní technologie ovlivnit souvisí také výhody a nevýhody. Zde bude vyjmenován pouze výčet hlavních výhod a nevýhod.

K obecným **výhodám** využití mobilních technologií ve vzdělání lze zařadit:

1. V dnešní době už většina žáků a studentů ví, jak s moderními či mobilními technologiemi zacházet, a proto odpadá nutnost je s technologiemi učit zacházet.

2. Další výhoda souvisí s možnostmi, které byly zmíněny výše a to, že práce s mobilními technologiemi často vede k aktivnějšímu přístupu a většímu zapojení studentů do výuky.

3. Mohou být využívána pro žáky se specifickými vzdělávacími potřebami pro ulehčení jejich studia (handicapovaní apod.).

4. Ulehčují celkovou komunikaci i zpětnou vazbu mezi učitelem a žákem a zároveň umožňují propojení většího množství dat.

5. Přispívají ke schopnosti společného učení a týmové práce.

6. Data se dají zpřístupnit v online podobě, což podporuje učení i na jiných místech než ve škole a zároveň je možné informace opakovaně procházet či rychle vyhledávat.

7. Žáci nebo studenti se učí samostatnosti a vlastní organizaci svých studijních povinností.

8. Prací s technologiemi podporují rozšíření svých digitálních kompetencí [16].

U každého uživatele lze samozřejmě vnímat i různé potencionální **nevýhody** a to:

1. Žádné zařízení nemá neomezenou paměť, takže často nemusí pojmout obrovské množství informací.

2. Jelikož se jedná o mobilní zařízení, může dojít k poruše a tím i ke ztrátě dat.

3. Záměr vzdělávací aplikace může být nesprávně pochopen a tím může dojít k aktivitě prováděné skrze mobilní technologie, ovšem bez jakéhokoliv didaktického záměru a poté i výsledku.

4. Mobilní technologie mohou být často využívána k šikaně, ale i podvádění a opisovaní při výuce.

5. Často se opakujícím úskalím je také vetší množství uživatelů, kteří se v jeden moment připojí ke svému zařízení.

6. Ve školách či vzdělávacích zařízení, kde nemají silné a stabilní připojení k sítí, může docházet k častým výpadkům připojení [7].

2.4 Platforma Jigx

Jedná se o platformu pro vývoj mobilních aplikací. Umožňuje multiplatformní vývoj nativních aplikací pro Android i iOS. Platforma Jigx[18] umožňuje využití různorodých datových poskytovatelů jako je: AzureSQL, Salesforce, SAP, a další cloudové poskytovatele pro uložení dat a poté propojení s aplikací pomocí Rest API. Pomocí Rest API umožňuje také napojení a integrování na již existující systémy a napojení na nástroje pro automatizaci pracovních postupů a procesů. Vývoj probíhá přes Visual Studio Code s Jigx builder Vsix extension. Toto rozšíření umožňuje přímé propojení s platformou a také konfiguraci veškeré aplikace, která je popsána v sekci 4. Zároveň toto rozšíření umožňuje nápovědu uživateli, které komponenty použít a také je možnost nahlédnout do schémat a zjistit veškeré možnosti, které v každé komponentě mohou být využity. Veškeré komponenty, které platforma využívá jsou využité z knihovny React Native. Veškeré zdrojové soubory pro konfiguraci aplikace, práci s komponentami a daty jsou psány v jazyku YAML s možností využití jazyka JSONata k dalším úpravám.

YAML

YAML[19] je jazyk pro serializaci dat, který se často používá k vytváření konfiguračních souborů v jakémkoli programovacím jazyce. Je nadmnožinou jazyka JSON. Jedním z hlavních rozdílů je, že nové řádky a odsazení v jazyce YAML skutečně něco znamenají. Základní strukturou souboru YAML je mapa a velmi obecně se zapisuje v podobě klíče a hodnoty.

[18] Dostupné z: https://www.jigx.com/
[19] Dostupné z: https://yaml.org/

JSONata

JSONata[20] je otevřený výrazový jazyk, který slouží k dotazování a transformaci dat JSON. Poskytuje mnohem bohatší a pokročilejší sadu možností mapování. Umožňuje vývojářům aplikací mapovat zdrojové soubory, do kterých mohou psát přímé dotazy a funkce JSONata bez nutnosti použití klienta. JSONata slouží k extrakci dat, která jsou ukryta v rozsáhlých strukturách JSON. Pomocí JSONata lze: manipulovat z řetězci, kombinovat a agregovat číselné výrazy a data, provádět transformace dat i času a vytvářet složitější struktury JSON pro komplexnější úlohy.

Figma[21]

Je užitečný nástroj pro návrháře, který umožňuje vytvořit návrhy pro: webové stránky, loga, prototypy mobilních aplikací. Figma umožňuje návrh uživatelského rozhraní a uživatelských zkušeností. Figma je užitečná při vytváření prototypů aplikací, protože je možné přímo udělat návrh a zjistit, jak bude vypadat na iOS či Android zařízení, a jak bude na zařízení fungovat.

AzureSQL

AzureSQL[22] databáze je relační databáze jako služba database-as-a-service (DBaaS) hostovaná v Azure, která spadá do oborové kategorie jako služba Platform-as-a-Service (PaaS). Je nejlépe využitelná pro moderní cloudové aplikace, které chtějí využívat nejnovější stabilní funkce SQL Serveru. AzureSQL umožňuje vytvořit vysoce dostupnou a výkonnout vrstvu ukládání dat pro aplikace. Umožňuje zpracovávat jak relační data, tak nerelační struktury, jako jsou grafy, JSON, prostorové a XML. AzureSQL database nabízí následující možnosti nasazení:

1. Jako jediná databáze s vlastní sadou prostředků spravovaných prostřednictvím logického serveru SQL. Jednotlivá databáze je podobná uzavřené databázi v serveru SQL Server. Tato možnost je optimalizována pro moderní vývoj nových aplikací vytvořených v cloudu.

2. Elastic pool, což je kolekce databází se sdílenou sadou prostředků spravovaných prostřednictvím logického serveru.

JSON

JSON[23] (JavaScript Object Notation) je odlehčený formát pro výměnu dat. Je založen na podmnožině standardu programovacího jazyka JavaScript ECMA-262. Je to textový formát, který je zcela nezávislý na jazyku, ale používá konvence jazyků jako jsou např: C,C++, C#, Java, JavaScript a další. Jedná se o ideální jazyk pro výměnu dat. Je postaven na dvou strukturách: 1. Kolekce - dvojic jméno/hodnota. V různých jazycích se realizuje jako objekt, záznam, slovník, hashovací tabulka a seznam s klíčem. 2. Uspořádaný seznam hodnot.

[20] Dostupné z: https://jsonata.org/
[21] Dostupné z: https://www.figma.com/about/
[22] Dostupné z: https://azure.microsoft.com/en-us/
[23] Dostupné z: https://www.json.org/json-en.html

SQLite

SQLite[24] je knihovna v jazyce C, která implementuje malý, rychlý, samostatný, vysoce spolehlivý a plně funkční databázový engine SQL. SQLite je zabudován ve všech mobilních telefonech. SQLite je knihovna v procesu, která implementuje samostatný transakční databázový engine SQL bez serveru s nulovou konfigurací. SQLite čte a zapisuje přímo do běžných diskových souborů. Kompletní databáze SQL s několika tabulkami, indexy a pohledy je obsažena v jediném diskovém souboru.

[24] Dostupné z: https://sqlite.org/about.html

Kapitola 3

Analýza a návrh

Tato kapitola se věnuje návrhu řešení vedoucímu k vytvoření nové mobilní aplikace.

3.1 Analýza

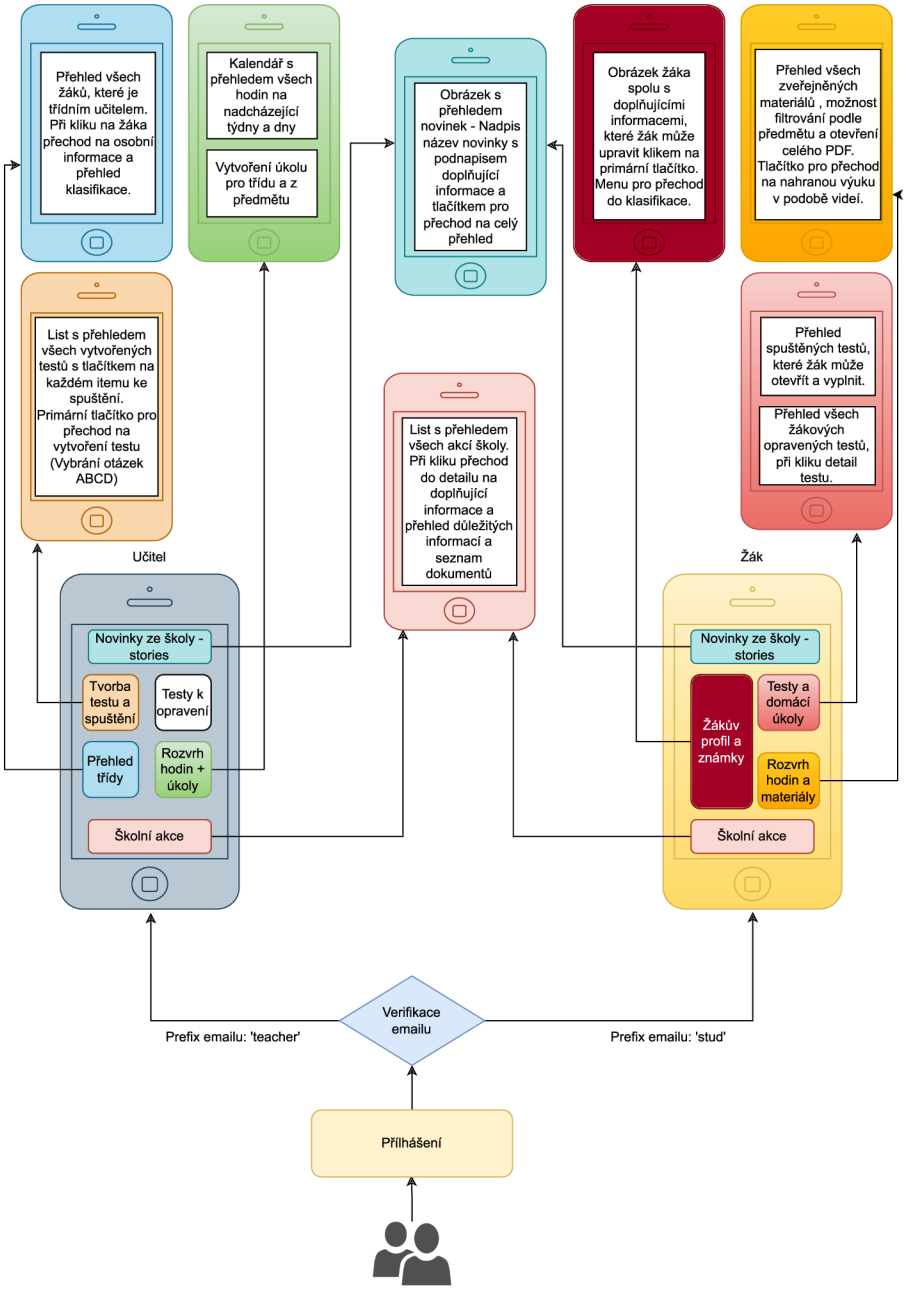
Prvním krokem při analýze a návrhu bylo prozkoumání trhu se zaměřením na již existující školní aplikace. Tyto aplikace jsou popsány v sekci 2.2. Nejprve bylo potřeba zjistit, co aplikace podporují a nabízejí, dále bylo nutné aplikace vyzkoušet a také projít dosavadní recenzce a zpětné vazby uživatelů na tyto aplikace. V recenzích se nejčastěji opakuje tvzení, že aplikace jsou uživateli často vnímany jako méně než průměrné. Na základě zmíněných recenzí je možné za problematické považovat tyto body:

1. Špatná stabilita a rychlost aplikací, které vedou k odhlašování uživatelů.
2. Problémy s aktualizacemi, které způsobují nekonzistentní chování aplikací.
3. Problémy při zobrazování základních funkcí jako je: rozvrh, známky, omluvenky a absence.
4. Aplikace podporují komunikaci s učiteli pomocí zpráv, které nefungují robustně.
5. Problémy se zobrazováním notifikací.

Na základě této analýzy a průzkumu začala tvorba prvního návrhu, která zahrnovala představu o tom, co by nově vzniklá aplikace měla umět a obsahovat oproti ostatním aplikacím a také jaké společné prvky by měla mít.

3.1.1 Rozdělení aplikace

Aplikace je určena k použití pro Android i iOS, jelikož platforma Jigx umožňuje multiplatformí vývoj. Aplikace je rozdělena do dvou základních uživatelských pohledů. V první řadě jsou to žáci, kteří aplikaci využívjí k psání testů, zobrazení studijních materiálů, rozvrhu, školních akcí, novinek ze školy, domácích úkolů a klasifikace. V druhém pohledu jsou to učitelé, kteří aplikaci využijí k vytváření testů pro žáky, zadávání úkolů a cvičení pro žáky, sledování žákovy klasifikace, zobrazení rozvrhů, novinek ze školy a školních akcí. Pro oba pohledy bude jedna sdílená databáze. Oba pohledy využívají podobnou logiku, pracují se stejnými entitami a zobrazení některých částí je stejné. Schéma systému je zobrazeno na obrázku 3.1.



Obrázek 3.1: Wireframe schéma systému, kde se zobrazení obsahu liší podle typu uživatele

3.1.2 Rozdíl oproti existujícím řešením

Oproti existujícím řešením je ve výsledné aplikaci navíc možnost psát testy, které žákům vytvoří sám učitel. Dále je zde přehled zveřejněných studijních materiálů, které si lze otevřít jako PDF. Pro učitele i žáky je zde přehled školních akcí a nejnovějších událostí ze školy.

3.1.3 Použité technologie při návrhu

Při samotném návrhu je potřeba stanovit, které technologie při vývoji aplikace využít. Technologie jsou zvoleny na základě problematiky a s ohledem na dosavadní zkušenosti

s nástrojem. Technologie, které jsou využity při návrhu a implementaci jsou tyto: Figma, Jigx platforma.

3.2 Návrh

Prvotní návrh byl vytvořen jako diagram 3.1 pro představu, jak bude vypadat průchod a práce s aplikací. Druhý návrh byl poté vytvořen v aplikaci Figma, a to již s prvky mobilní aplikace pro představu, jak by mohlo vypadat první GUI. Třetí návrh byl tvořen přímo přes platformu Jigx s reálnými daty. Všechny tyto návrhy byly inspirovány nejen dosud existujícími řešeními, ale také dalšími existujícími mobilními aplikacemi.

3.2.1 Rozdělení aplikace z pohledu dat a funkcí

Domovská stránka a všechny widgety, které jsou vidět na domovské stránce se zobrazí na základě toho, kdo je přihlášen. V případě přihlášení žáka se zkontroluje email a zobrazí se věci spojené s žákem.
Jestliže se přihlásí učitel, tak se znovu zkontroluje email a zobrazí se požadované widgety a data. Všechny entity s datovými typy jsou zobrazeny na obrázku A.2.
Přehled widgetů a entit s daty z pohledu **žáka**:

Novinky ze školy - představují přehled nejnovějších událostí, novinek ze školy nebo vydané školní časopisy. Každá hodnota je uložena v databázi s názvem, doplňujícími informacemi, odkazem pro thumbnail a odkazem na požadovanou událost.

Profil - v profilu jsou vidět všechny informace o žákovi s možností přechodu do klasifikace, kde si žák bude moci zobrazit veškeré známky s datem, kdy byl klasifikován a z jakého předmětu. Hodnoty pro žáka jsou v databázi: jméno, email, telefon, adresa, zdravotní pojišťovna.

Testy - představují přehled testů, které budou moci žáci otevřít, napsat a odevzdat. V druhé řadě představují přehled již odevzdaných a ohodnocených testů. Každý test je uložen u konkrétního žáka a obsahuje datum, známku, z jakého předmětu byl žák hodnocen, všechny otázky i odpovědi a doplňující komentář učitele.

Domácí úkoly - jedná se o přehled všech zadaných nadcházejících úloh. V tomto widgetu se zobrazí pouze nadcházející úlohy od aktuálního určeného data. Úkoly, které budou mít datum odevzdání od aktuálního data rozdíl méně než 24 hodin, budou zvýrazněny. Každá úloha bude mít popis zadání, bude určeno do jakého předmětu se má úkol vypracovat, datum odevzdání, případně doplňující fotografie a informace.

Materiály - v této sekci jsou dostupné všechny zveřejněné studijní materiály. Materiály jsou rozděleny podle předmětů. Žák má možnost jednotlivé materiály filtrovat podle předmětů. Každý materiál je možné otevřít v telefonním webovém prohlížeci jako je PDF formát. V případě, že škola bude mít nahrávanou výuku, dá se využít sekce nahraná výuka, kde si žák bude moci spustit hodinu jako video přímo v aplikaci.

Rozvrh - jedná se o přehled rozvrhu na nadcházející dny. V rozvrhu je možnost vidět, kdy výuka začíná, kdy končí, v jaké učebně výuka probíhá, jaký předmět je vyučován a případně doplňující informace k dané výuce.

Školní akce - v sekci týkající se školních akcí jsou zobrazeny veškeré nadcházející akce školy. Každá akce má svůj název, obrázek, termín a čas, kdy se akce koná, organizátora, podstatné informace a adresu místa konání. Některé akce mohou obsahovat informace o ceně a případně doplňující dokumenty spojené se školní akcí.

Přehled widgetů a entit s daty z pohledu **učitele** bez společných prvků z žáka:

Testy ke spuštění - v této sekci má učitel možnost spustit vytvořené testy žákům k vyplnění. Zároveň v tomto widgetu má možnost vytvořit nový test, který se bude skládat z 6 otázek s možnostmi ABCD a jedné textové otázky. Test je možné vytvořit pro určitý předmět a třídu. Učitel má také možnost přidat nové otázky do databáze na základě již existujících, u kterých bude moci změnit znění otázky či přidat nové alternativní odpovědi. Učitel vytvoří test s názvem pro konkrétní třídu a předmět, vybere otázky a spustí test žákům.

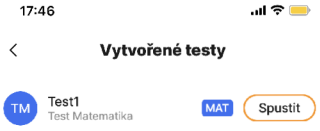
Testy k opravení - zde se zobrazí všechny odevzdané testy od žáků. Při otevření určitého testu učitel uvidí správné a špatné odpovědi ke každé otázce a také automaticky vyhodnocený celkový počet bodů. Zároveň je zde automatický návrh známky na základě bodového hodnocení. Jestliže žák udělal maximálně jednu chybu, bude hodnocen nejvyšší známkou, dále s každou další chybou se řád snižuje. Učitel zde má možnost také udělit i jinou známku, než navrhovanou a také přidat doplňující komentář k hodnocení.

Moje třída - jedná se o základní přehled žáků ve třídě, ve které je učitel třídním profesorem. V případě přechodu pohledu v aplikaci na žáka zde uvidí všechny podstatné informace o žákovi, jako je jméno, adresa, email, telefon a zdravotní pojišťovna. Dále uvidí veškerou žákovu klasifikaci, čímž bude moci sledovat, jak si žák vede po klasifikační stránce.

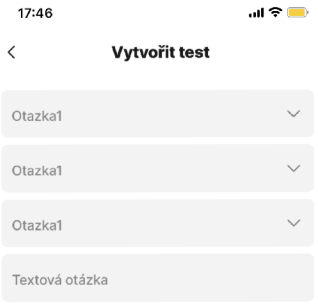
Vytvořit úkol - v této části má učitel možnost vytvořit domácí úkol pro žáky. Každý úkol, který navrhne bude vytvořen pro konkrétní třídu a předmět. Učitel má možnost přidat obrázek domácího úkolu, nastavit datum odevzdání, název domácího úkolu a případně přidat další dopňující informace.

3.2.2 Návrh gui

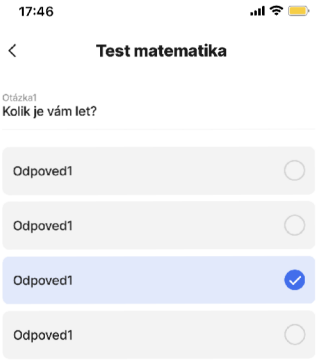
Všechny prvky, které jsou na domovské stránce a na jednotlivých sekcích, jsou poskládány tak, aby práce s nimi byla co nejefektivnější a nejintuitivnější. Každá sekce bude mít vlastní název a také obrázek jako „header image“, který se bude měnit na základě kontextu obrazovky. Na každé stránce bude tlačítko pro rychlý přechod na domovskou stránku a zpětná šipka pro přechod na předchozí obrazovku. Celkový design je inspirován školními aplikacemi a v aplikaci jsou použity obrázky se školní tématikou.
Na základě 5.1 zpětné vazby a analýzy existujících aplikací se poté tvořily jednotlivé návrhy rozhraní. Prvotní návrh jednotlivých obrazovek je na obrázku 3.2.



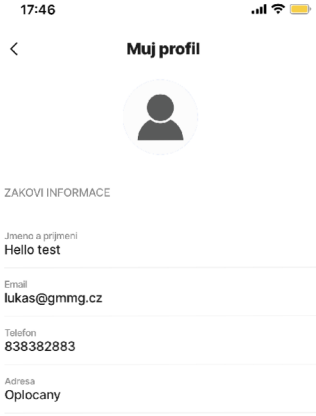
Obrázek demonstruje návrh obrazovky s vytvořenými testy ke spuštění



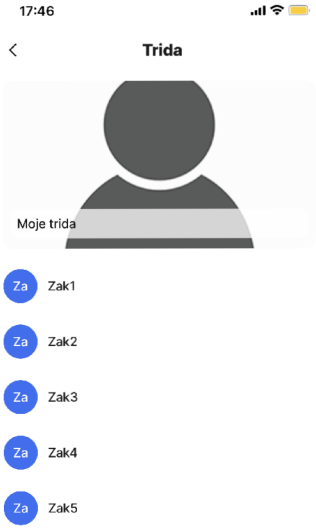
Obrázek demonstruje návrh formy na vytvoření testu



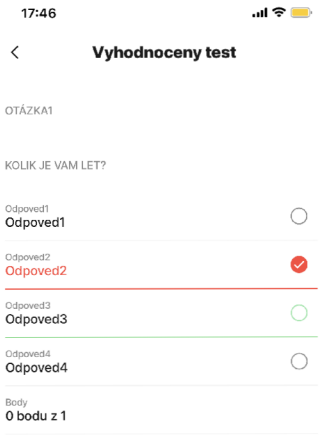
Obrázek demonstruje návrh formy na zadání odpovědi k otázce



Obrázek demonstruje návrh profilu žáka



Obrázek demonstruje návrh s přehledem učitelovy třídy



Obrázek demonstruje návrh vyhodnoceného testu

Obrázek 3.2: Ukázka návrhu jednotlivých obrazovek pro pohled učitele a žáka

3.2.3 Finální návrh

Oproti původnímu návrhu se při realizaci finálního návrhu změnila domovská obrazovka, kde se místo 1x1 widgetů přidaly widgety přímo s obrázky týkající se školní výuky. Dále byl přidán obrázek žáka, který je přihlášen a list s přehledem školních akcí byl přidán na domovskou obrazovku. Do jednotlivých obrazovek oproti prvotnímu návrhu byly přidány obrázkové hlavičky, a to do horní části obrazovky. Jednotlivé listy byly upraveny a byly k nim přidány další elementy pro lepší uživatelské vnímání a zároveň, aby byla možnost listy filtrovat a vyhledávat v nich. K informacím, jako je telefonní číslo a email byly přidány ikony pro rychlou interakci. Finální vzhled aplikace je zobrazen na obrázku A.1.

Kapitola 4

Implementace

V této kapitole je popsána implementace výsledné mobilní aplikace.

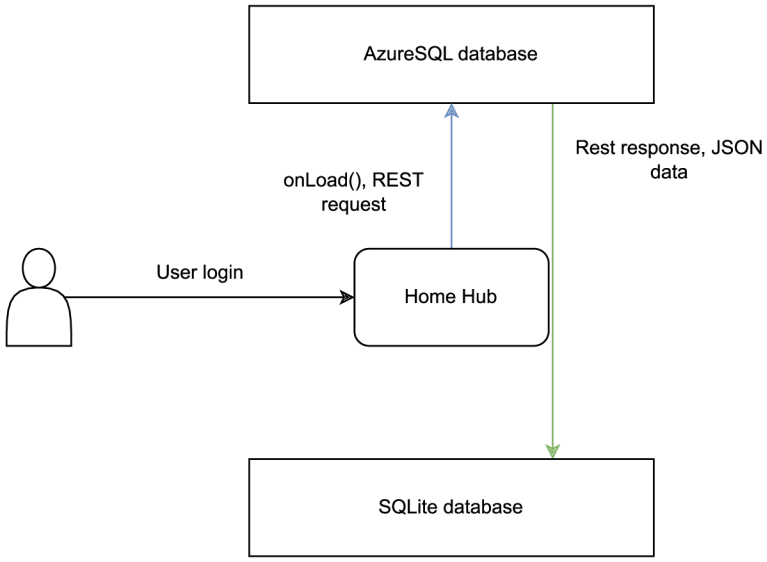
4.0.1 Správa kódu

Při implementaci aplikace byl využit distribuovaný systém správy verzí Git[1], který umožňuje uchování historie změn ve zdrojovém kódu. Hlavním důvodem využití bylo sledování a revize změn zdrojového kódu a také uchování kopie projektu na vzdáleném serveru. Nástroj byl zvolen GitHub[2] jako verzovací systém, na základě předešlých zkušeností s tímto nástrojem.

4.1 Home Hub

Pro přístup a přihlášení do aplikace byly vytvořeny 2 účty, pro pohled učitele email: bpteacherproject@gmail.com a pro žáka: bpstudproject@gmail.com. V případě správného přihlášení emailem a heslem, je uživatel přesměrován do uživatelského prostředí domovské obrazovky, kde uvidí veškerý vzhled aplikace na základě přihlášeného uživatele. Uživatel je nadále přihlášen, i když zavře nebo ukončí aplikaci. Zároveň se při příhlášení provede *onLoad* akce, která zašle GET REST požadavek na AzureSQL server, na kterém jsou uložena data. Konfigurace funkce je zobrazena ve výpise 4.1. Schéma komunikace je zobrazeno na obrázku 4.1. Z důvodu bezpečnosti jsou všechny REST požadavky z mobilní aplikace směrovány přes cloud Jigx do instance Microsoft AzureSQL. V cloudu Jigx se neukládají žádná data a používá se pouze jako směrovací proxy server, který umožňuje vytváření bílých seznamů IP v Microsoft AzureSQL. Výsledek funkce se vrátí do aplikace v zařízení jako pole JSON. Každý záznam v poli je uložen jako řádek v lokální databázi SQLite, v entitách zadefinovaných funkcí. Akce, která zašle požadavek na server je *sync-entites*, kde se definuje data provider, entita, kde budou uložena data a název funkce. Každý záznam je uložen v SQLite s jedinečným id. Všechna data uložena v SQLite databázi, jsou poté dostupná kdykoliv na zařízení a lze je mazat či editovat i pokud je zařízení offline. Ke smazání SQLite databáze dojde pouze v případě, že se uživatel z aplikace odhlásí, nebo když provede funkci na smazaní dat ze SQLite v aplikaci.

[1] Dostupné z: https://git-scm.com/
[2] Dostupné z: https://github.com/



Obrázek 4.1: Ukázka komunikace se serverem, kde v případě přihlášení uživatele dojde ke spuštění onLoad eventu, který zašle požadavek na server, server odpoví na požadavek a data jsou uložena v lokální databázi

```
onLoad:
#event ktery se provede pri nacteni aplikaci
  type: list-akce
  options:
    actions:
    #definice akce, kde je potreba definovat poskytovatele, entitu kam
        budou data ulozena a~nazev funkce ktera ulozeni provede
      type: akce_pro_sync_dat
        options:
          provider: DATA_PROVIDER_SQL
          entita: nazev
          funkce: nazev

#definice funkce, kde se definuje poskytovatel, pripojeni na server, metoda
    a~nazev procedury pripadne query
provider: DATA_PROVIDER_SQL
connection: jigx.lukas.database
method: execute
procedure: vybrat_testove_otazky
```

Výpis 4.1: Ukázka konfigurace akce a funkce pro stažení dat ze serveru do lokální databáze, dataset je zobrazen ve výpise 4.2

```
"recordset":
  {
    "id": 1,
    "otazka": "Ktera planeta nema prstenec?",
    "spravna_odpoved": "Mars",
    "alternativni_odpoved1": "Jupiter",
    "alternativni_odpoved2": "Neptun",
    "alternativni_odpoved3": "Saturn",
    "body": 1
  }
```

Výpis 4.2: Ukázka stažených dat z databáze

Platforma Jigx umožňuje používat různé možnosti při personalizaci Home Hub. Na Home Hub jsou použité 2 prvky. Jedním z nich je skupina Stories, která zobrazuje nejnovější novinky ze školy. Tyto stories mohou být v podobě skupinových obrázků či videí. Veškerá konfigurace stories se provádí v jigu a poté je zadefinována v index souboru[3] a tato konfigurace je zobrazena ve výpise 4.4. Konfigurace stories je zobrazena ve výpise 4.3. Každá stories má definovaný data set jako pole hodnot, které se má vykreslit. U stories lze definovat název každého obrázku či videa doplňující informace v podobě podtitulku a akci, která se provede při stisku tlačítka na jednotlivé story.

```
stories:
  type: component.story-group
  #nazev komponenty s definici dat a~elementu
  options:
    data: pole dat datasourcu
    groupName: Novinky ze skoly
    #nazev skupiny kde budou zobrazeny stories
    item:
      type: story_image
      options:
      #definice jednotlivych elementu ktere budou zobrazeny u obrazku
        nazev
        akce_pro_onPress
        #ktera se provedu po stisku na tlacitko
        source_image
```

Výpis 4.3: Definice stories

Druhým prvkem jsou samotné widgety, které se definují v každém jigu. Každý widget může mít velikost 1×1, 2×2 ,4×2, 2×4 či 4×4. Každý widget může mít také vlastní vzhled v podobě: obrázku, videa, listu, grafu, mapy, avatara či sekce s nadpisy. Konfiguraci, které widgety lze zobrazit na Home Hub lze provést přímo v index souboru na základě vyhodnocení výrazu. Konfigurace je zobrazena ve výpise 4.4.

[3]cesta k index souboru: /source-code.zip/studyisfun/index.jigx

```
name: studyisfun
title: StudyIsFun
#nazev ktery bude videt na domovske strance
onLoad: akce_pro_sync_dat
#definice stories
stories:
  "definice_jigu"
#definice jednotlivych jigu
widgets:
  size: 2x2
    jigId: definice_jigu
    isHidden: =@ctx.user.email = 'bpteacherproject@gmail.com'
```

Výpis 4.4: Konfigurace Home Hub apikace

4.2 Konfigurace jednotlivých widgetů

Každý jig má svůj název, který je daný názvem souboru a tím na něj lze poté odkazovat. V případě kliku na jednotlivý widget, se poté uživatel dostane do jigu. Widget je zobrazení, které je dostupné na domovské obrazovce a konfigurace se provádí v jigu. Jig lze konfigurovat a personalizovat pro uživatele. Jig má pět základních konfigurací:

1. kalendář - tento jig zobrazí kalendářní data a poskytuje přehled o plánovaných událostech.

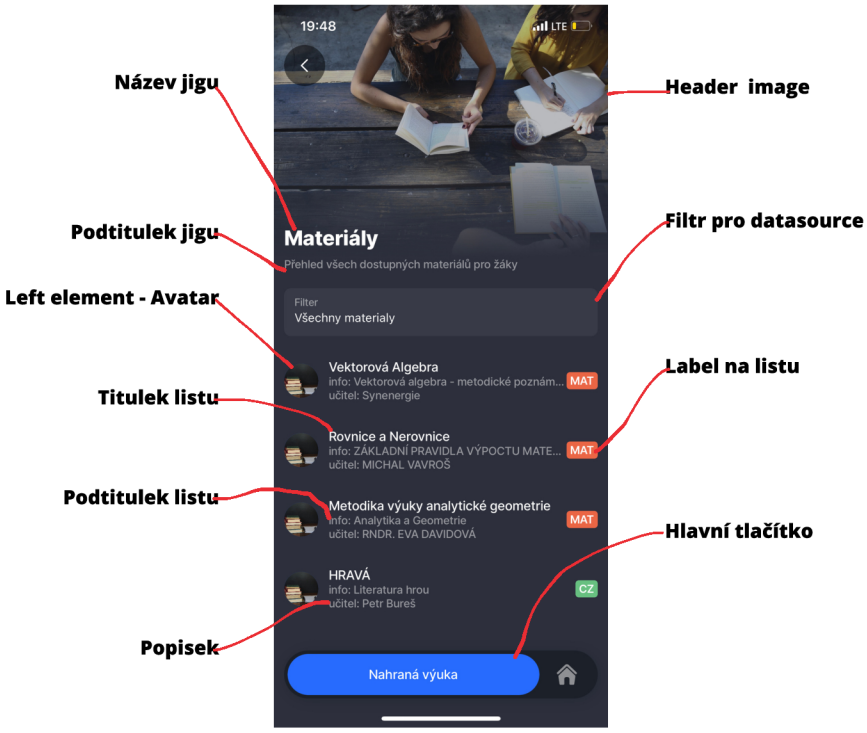
2. kompozitní - je složený z více prvků, může být složen i z více jigů najednou

3. default - tento jig umožňuje vkládání různých komponent, jako jsou: formy, listy, obrázeky, entity, mapy, grafy a videa.

4. dokument - používá k zobrazení dokumentů souvisejících s projektem, lze jej zobrazit dvěma způsoby: jako PDF nebo stránka HTML.

5. list - je proměnlivá, uspořádaná posloupnost prvků a souvisejících hodnot. Umožňují uchovávat pohromadě data, která k sobě patří, zhušťovat kód a provádět stejné metody a operace s více hodnotami najednou. Každou položku listu lze vyvolat samostatně pomocí indexování. Listy jsou složený datový typ, složený z menších částí a jsou velmi flexibilní, protože do nich lze přidávat nebo odebírat a měnit hodnoty.

V každém jigu lze také zobrazit záhlaví, které přidá jigu lepší uživatelské vnímání. V záhlaví jigu lze přidat popis jednotlivého jigu a dalé zobrazit komponenty, jako jsou: graf, mapu lokace či obrázek spojený s kontextem jigu. Ukázka jigu s popisem elementů je zobrazena na obrázku 4.2.



Obrázek 4.2: Ukázka jigu s popisem jednotlivých elementů na jigu

U každého jigu je možné definovat i rozdílné akce. Ty se zobrazí jako hlavní tlačítko na stránce, které při stisku provede nadefinované akce. Mezi tyto akce patří:

1. action-list - tato akce umožňuje spustit více akcí, přičemž uživatel musí stisknout pouze jedno tlačítko. Akce lze nastavit tak, aby se spouštěly postupně nebo aby byly provedeny jako hromadná funkce.

2. confirm - akce potvrzení vytvoří vyskakovací okno před provedením akce. Slouží jako pojistka pro uživatele, který se pokouší provést destruktivní akci nebo akci, která by mohla mít pro uživatele jiné důsledky, např: dotaz, zda si uživatel přeje pokračovat ve vymazání záznamu. Tuto akci lze také použít jako potvrzení pro poskytnutí zpětné vazby po provedení akce.

3. execute-entities - slouží k úpravě, vytváření a mazání více řádků v určité tabulce v databázi.

4. execute-entity - slouží k ukládání, aktualizaci nebo mazaní dat z databázi v závislosti na zvolené metodě.

5. go-back - slouží k návratu na předchozí stránku. Tuto akci lze použít v seznamu akcí nebo ji lze použít společně s jinou akcí.

6. go-to - tato akce slouží k přesměrování uživatele na jinou stránku.
7. open-url - akce slouží k otevření webové stránky v závislosti na zadané url.
8. submit-form - akci lze použít na formuláři k uložení nebo aktualizaci dat do databáze.
9. sync-entities - akci lze použít k synchronizaci dat lokální databáze s databází na serveru.

4.2.1 Konfigurace dat v jigu

K datům, která jsou uložena v lokální databázi na zařízení lze přistoupit přes SQL dotaz. Veškerá data jsou definována v části **datasources**. Konfigurace datasource je zobrazena ve výpise 4.5. Dotazování probíhá nad SQLite databází příkazem select. Každý datasource má svůj název a lze jej poté navázat na konkrétní komponenty, příkazem: **=@ctx.datasources.<název>.<položka-příkazu-select>**. Datasources lze použít a nakonfigurovat přímo v jigu nebo je použít jako globální datasource dostupný pro všechny další jigy. Každému datasourcu lze upřesnit dotazy pomocí queryParameter.

```
datasources:
#nazev datasourcu
  klasifikace:
    type: datasource.sqlite
    options:
      provider: DATA_PROVIDER_LOCAL
#nazev entity, kde se nachazi data
      entities:
        entity: vsechny_pouzite_tabulky
      queryParameters: zakovo_id
      query: |
        SELECT data
        FROM [testy] t
        LEFT JOIN [testy_zaci] tz ON id_test
        WHERE id_zak = query_parameter and status = 'opraveno'
            ORDER BY datum DESC
```

Výpis 4.5: Vybrání klasifikace pro specifického žáka, kde status testu je změněn na opraveno

4.2.2 Použité komponenty v aplikaci

Komponenty použité ve výstupní aplikaci jsou: různé konfigurace listů pro zobrazení dat v kontextu, upravitelné formuláře pro vytvoření a úpravu dat, záhlaví jigu, kalendář, mapa lokace, webový pohled, stories a neupravitelný formulář, který slouží pouze pro čtení dat.

4.3 Konfigurace testů a listů aplikace

Každá komponenta obsahuje různé variace možností, jak komponentu upravit, na kterém jigu ji použít, případně jestli ji lze využít na widgetu. Každá komponenta má svůj název a **instanceId**, přes které je komponenta odkazovatelná a lze přistoupit i ke stavu komponenty. Tímto stavem lze například zjistit, jestli je komponenta validní a jakou hodnotu

nabývá. Data, která se mapují na komponenty, lze také upravovat pomocí JSONata výrazů. Dále je možné provádět například: řetězcové výrazy, matematické funkce, či použít logické výrazy. Výraz je vypsán ve výpise 4.6.

```
initialValue: |=
                for all answers{
                  If(spravna_odpoved_testu = zakova_odpoved) {
                      pricti 1
                  }
                  else{
                      pricti 0
                  }}
```

Výpis 4.6: Výraz pro vyhodnocení správné odpovědi v testu. V tomto případě se kontroluje předem definovaná správná odpověď otázky s odpovědí zadanou žákem při vyplnění a odevzdání testu. Pokud je odpověď správná, k výslednému počtu bodů se přičte 1, jinak 0. Takhle se zkontrolují všechny otázky a odpovědi a učitel poté uvidí výsledný počet bodů u testu.

4.3.1 Vytvoření a spuštění testu

Vytvoření testu je uděláno jako default jig, ve kterém je formulář na výběr z rozevíracích seznamů. V jigu je datasource s otázkami, které jsou závislé na zvolení předmětu a třídy. Tento datasource je poté použit pro rozevírací seznamy na výběr jednotlivých otázek. Učitel poté vybírá otázky z těchto rozevíracích seznamů a také má možnost vytvořit jednu textovou otázku. Jakmile klikne na tlačítko uložení, test se vytvoří, data jsou poslána na server AzureSQL a do databáze je vytvořen nový záznam testu. Nové id testu je použito k vytvoření záznamu do druhé tabulky testu pro žáky, kde se vytvoří záznamy pro všechny žáky, kteří jsou ve třídě pro vybraný test. Ukázka procedury je výpise 4.7. Uživatel je přesměrován zpět na list s testy ke spuštění a každý test v tomto listu má vlastní tlačítko na spuštění. Při kliku na spuštění se změní status testu na „Spusteno“ a tento test je spuštěn všem žákům třídy, pro kterou byl test vytvořen.

```
(
     @data_z_aplikace nvarchar(255)
)
AS
BEGIN
INSERT values to table test
-- set new id
DECLARE @test_id int = (SELECT SCOPE_IDENTITY())
INSERT values to table test_zak
END
```

Výpis 4.7: Ukázka procedury na vytvoření testu

4.3.2 Vytvoření nové otázky

Do vytvoření nové otázky lze přejít sekundárním tlačítkem, které je dostupné v jigu, přes který se vytváří nový test. Vytvoření nové otázky je uděláno přes defaultní jig, kde je

formulář s rozbalovacím seznamen na výběr otázky, ze které může být vytvořena nová. Na základě výběru otázky se zobrazí celý název otázky, alternativní odpovědi a správná odpověď. Tyto zobrazené informace můžou být dále upraveny přes textové pole. Při uložení formuláře je nová otázka vytvořena do databáze na předmět i třídu a může být poté použita při tvorbě nového testu.

4.3.3 Odevzdání testu a opravení

Žák otevře test z listu zveřejněných testů a po otevření přejde na default jig s formulářem. Data z datasource pro otázky a odpovědi jsou vybrány na základě id testu. Ve formuláři je poté vždy název otázky a pod ní 4 zaškrtávací políčka, kde žák může zvolit odpověď. Ukázka vyhodnocení odpovědi je ve výpise 4.8. Poslední otázka je textová, kde žák napíše odpověď ručně. Při odevzdání testu jsou všechny odpovědi a data uložena a odeslána na server do databáze, kde se odpovědi uloží podle id testu a žáka, který tento test odevzdal.

```
odpoved1: =
    If (stav_checkboxu = true){
        odpoved_z_datasourcu
    }
    else{
        zkontrolovat dalsi checkbox
    }
```

Výpis 4.8: Ukázka odeslání odpovědi na základě vyhodnocení stavu vybraného zaškrtávacího políčka

Zároveň se status testu změní na „Odevzdano“, takže žák už test znovu neuvidí a pouze učitel uvidí žákův odevzdaný test a může ho opravit. Učitel při otevření odevzdaného testu uvidí odpovědi u každé otázky, vyhodnocený celkový počet bodů na základě porovnání odpovědí žáka a správné odpovědi. Bude moci udělit známku, která bude navržena podle celkového počtu bodů a také přidat doplňující komentář. Po uložení se změní status testu na „Opraveno“ a test se zobrazí žákovi pouze ve čtecím režimu. Žák si bude moci prohlédnout test, své odpovědi, známku a komentář učitele.

4.3.4 Studijní materiály

Studijní materiály jsou vytvořeny jako list jig, který umožňuje zobrazení pole hodnot. V tomto jigu je také použit filtr, který umožňuje filtrování materiálů podle předmětů. V kontextu jigu je dostupný stav tohoto filtru, proto jeho stav lze využít jako podmínku při vybírání dat z datasourcu a zobrazit tak jen materiály na daný předmět. Ukázka konfigurace listu je ve výpise 4.9.

```
data: pole_dat_z_datasource
item:
  type: component.list-item
  options:
    pro vsechny data z~pole{
    #elementy listu
    title: item.nazev
    subtitle: item.info
    description: item.vytvoril
    leftavatar: item.jpg
    label: item.title,item.color
    onPress event
    #akce ktera se provede po stisku na element listu
    }
```

Výpis 4.9: Konfigurace listu v jigu

Použité materiály

Obrázky v databázi, textové formáty pdf a použitá videa v aplikaci byla převzata z dostupných internetových zdrojů pro demonstraci, jak se tyto formáty budou vykreslovat a zobrazovat při použití reálných dat. Dostupnost a citace těchto dokumentů jsou uvedeny v readme souboru[4].

[4]Soubor dostuoný na cd v : readme.md

Kapitola 5

Testování

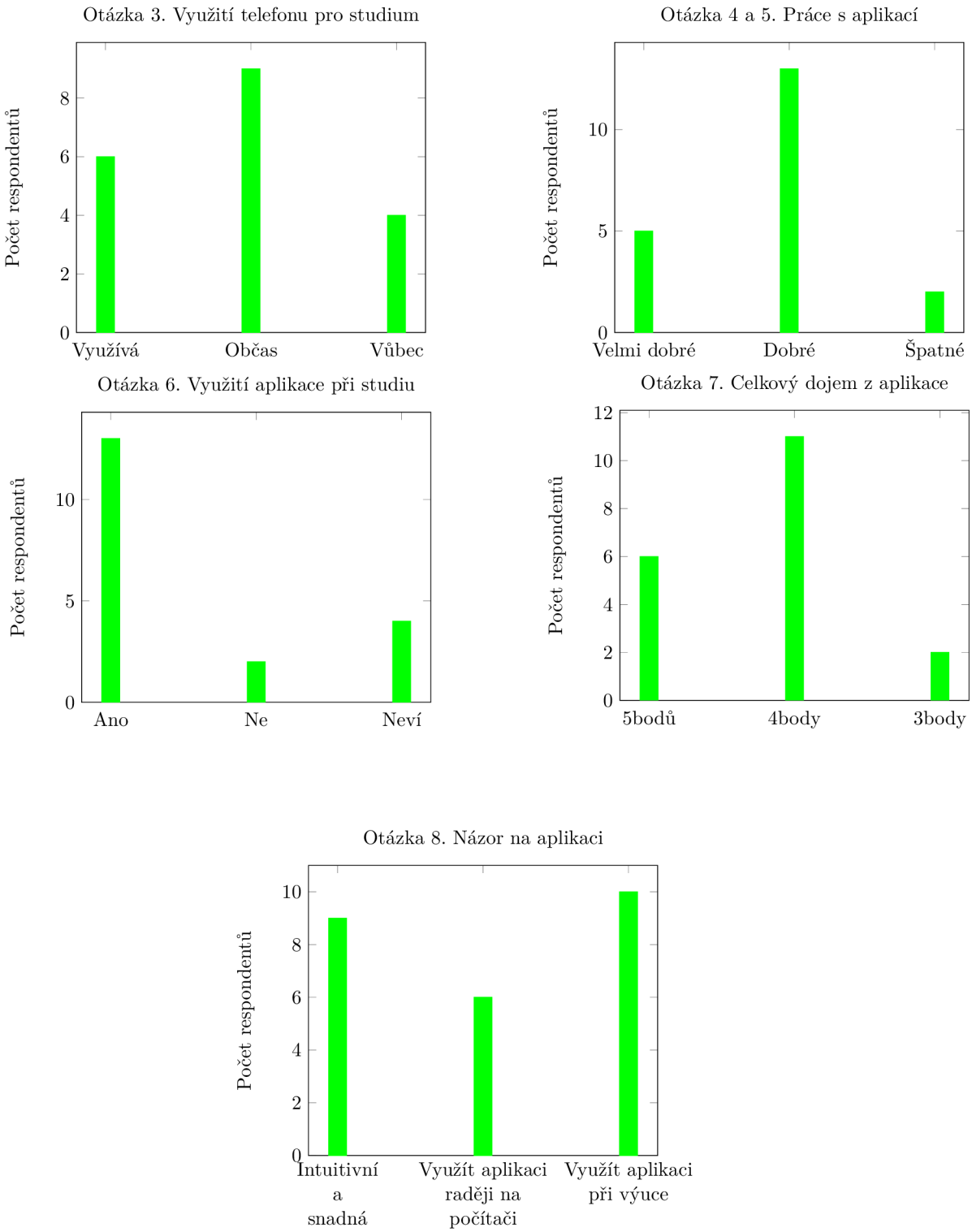
Velmi důležitou částí po vytvoření aplikace je samotné testování. V této části bude čtenář seznámen s provedeným testováním a s výsledky, které testování přineslo.

5.1 Uživatelské testování

Při návrhu bylo provedeno krátké testování designu se studenty víceletého gymnázia, které bylo zaměřeno na tyto oblasti:

1. Jaké informace v aplikaci vyhledáváte a jak rychle se Vám tyto informace hledají?
2. Obsahuje aplikace potřebné a dostatečné informace ze školy a pro studium?
3. Co se Vám na aplikaci líbí, co se Vám naopak nelíbí?
4. Co byste v aplikaci uvítali navíc oproti dosud existujícím aplikacím?

Uživatelské testování výsledné aplikace poté proběhlo na základní škole se studenty devátého ročníku. Testování se účastnilo 19 studentů a jeden učitel, z toho 6 dívek a 8 chlapců, 5 respondentů své pohlaví neuvedlo. Aplikace byla žákům a učiteli nejprve komplexně představena pomocí krátké prezentace. Prezentace obsahovala základní popis aplikace a také přehled funkcí, které aplikace obsahuje. Studenti a učitel dostali 3 zařízení s předinstalovanou aplikací na vyzkoušení veškerých funkcí. Cílem testování bylo zjistit, jak je aplikace studenty a učitelem vnímána, dále, jak aplikace funguje a zda nedochází k chybám při funkčnosti. Následně bylo ověřeno, zda je aplikace uživatelsky přívětivá, intiutivní a rychlá při používání. K testování byl použit vlastní vytvořený dotazník v aplikaci SURVIO. Dotazník je k nahlédnutí umístěn v přílohách této práce A.3. Analýza výsledků z dotazníku je zobrazena v grafech 5.1.



Obrázek 5.1: Přehled odpovědí na otázky z dotazníku

Užitečné sekce: Respondentům jako nejvíce důležité sekce přišly zejména testy a domácí úkoly. Za nejméně důležité sekce byly považovány školní akce a novinky ze školy.

Výhrady, které respondenti uváděli, byly: Z pohledu učitele by bylo dobré, kdyby aplikace ukazovala služby ve třídě, zobrazení suplování, změny v rozvrhu a přidání notifikací o změnách a propojení s existujícím školním systémem.

Z pohledu žáka by bylo dobé zobrazení správných odpovědí po vyhodnocení testu, zobrazení školního řádu, přidání možnosti sledování vlastní docházky a zobrazení přehledu obědů ze školní jídelny spolu s jejich přihlašováním či odhlašováním.

Co se respondentům na aplikaci líbí: Respondentům se líbí, že je aplikace přehledná, snadná na ovládání, rychlá, dobře strukturovaná a vše je na jednom místě. Žákům se také líbila možnost psát testy a zobrazení výukových materiálů.

Co by respondenti v aplikaci vylepšili: Respondenti uváděli, že by si v aplikaci dokázali představit vylepšenou o aktuální rozvrh, suplování a změny. Dále by přidali zobrazení správných odpovědí u testů, umístili by klasifikaci na hlavní stránku a přidali zobrazení notifikací při změnách.

Kapitola 6

Závěr

Cílem této bakalářské práce bylo vytvořit podpůrnou mobilní aplikaci pro studenty i učitele, rozšířenou o možnosti, které nejsou u dosud existujích školních aplikací na trhu zatím dostupné. Zvolenou platformou pro vývoj mobilní aplikace byla platforma Jigx. Nově vzniklá mobilní aplikace nabízí dva uživatelské pohledy, a to pohled žáka a učitele. Oproti ostatním dosud existujícím aplikacím se vzniklá mobilní aplikace vyznačuje inovativními funkcemi, kterými jsou především: vytvoření a psaní zkušebních testů, přehled novinek ze školy ve stories a zobrazení dostupných studijních materiálů. Vytvořená aplikace byla ve finální verzi podrobena také testování u žáků a učitele na základní škole, což lze považovat za přidanou hodnotu vývoje celé aplikace.

Předem stanovené dílčí cíle práce byly splněny a aplikace byla úspěšně implementována. Aplikace byla otestována pouze v beta verzi, jelikož platforma Jigx, přes kterou byla tato aplikace vytvořena ještě není dostupná na trhu. Proto se zveřejnění aplikace dosud neuskutečnilo.

6.1 Pokračování ve vývoji aplikace

Další vývoj aplikace je výzva, která rozhodně není nerealizovatelná. Aplikace má kromě svých předností i mnohé limity, které by se daly dalším vývojem a vylepšením eliminovat. S ohledem na získané výsledky zpětné vazby při testování vznikly oblasti, ve kterých je možné aplikaci v budoucnu vylepšovat. Jedná se o zmíněné: přidání suplování a změn rozvrhů, zobrazení přehledu školní jídelny a možnost přihlašování, či odhlašování obědů. Dále by bylo možné přidání procvičovacích cvičení s možností získání ocenění za jednotlivá splnění a rozšíření aplikace o notifikace, informující o změnách či spuštění testu.

Literatura

[1] ADESINA, R. Mobile Operating Systems and Application Development Platforms: A Survey. *Int. J. Advanced Networking and Applications* [online]. Engg Journals Publications. August 2014, sv. 6, č. 1, s. 2195–2201, [cit. 2023-03-05]. ISSN 0975-0290. Dostupné z: https://www.researchgate.net/publication/279954676_Mobile_Operating_Systems_and_Application_Development_Platforms_A_Survey.

[2] ALEEM, F. *Layered Architecture used by iOS and it's Performance Portability* [online]. ResearchGate, July 2019 [cit. 2023-03-10]. DOI: 10.13140/RG.2.2.22845.20968. Dostupné z: https://www.researchgate.net/publication/334233122_Layered_Architecture_used_by_iOS_and_it's_Performance_Portability. Path: HomeComputer; SoftwareSystem; SoftwareComputer; ScienceOperating; SystemsiOS.

[3] BHASKAR et al. General Principles of User Interface Design and Websites. *International Journal of Software Engineering (IJSE)* [online]. 2011, sv. 2, č. 3, s. 45–60, [cit. 2023-03-30]. Dostupné z: https://www.academia.edu/8877486/General_Principles_of_User_Interface_Design_and_Websites.

[4] DEACON, J. *Model-View-Controller (MVC) Architecture* [online]. Computer Systems Development, Consulting Training, August 1995. 2009 [cit. 2023-03-07]. Dostupné z: https://www.academia.edu/30077059/Model_View_Controller_MVC_Architecture?fbclid=IwAR13Dl1FWm_b787W7p7_5hLYFXr_zZYQmFQnw5C2MZNJUUB0Q_tvW_9Z8Sk.

[5] EGGER, R. a BUHALIS, D. Mobile systems (Introduction Chapter). In: *ETourism Casestudies* [online]. 1. vyd. London: Routledge, 2008, kap. 6, s. 417–425 [cit. 2023-02-10]. ISBN 9780080942865. Dostupné z: https://www.researchgate.net/publication/268389759_Mobile_systems_Introduction_Chapter.

[6] GILLIS, A. DEFINITION native app. *TechTarget* [online]. Gillis,Alexander. 2022. 2022 [cit. 2023-03-20]. Dostupné z: https://www.techtarget.com/searchsoftwarequality/definition/native-application-native-app. Path: Home; Software design and development.

[7] NEUMAJER, O., ROHLÍKOVÁ, L. a ZOUNEK, J. *Učíme se s tabletem - využití mobilních technologií ve vzdělávání: kniha s online podporou.* Praha: Wolters Kluwer, 2015. 188 s. ISBN 978-80-7478-768-3.

[8] RASHEDUL, I. a MAZUMDER, T. Mobile application and its global impact. *International Journal of Engineering and Technology* [online]. Engg Journals

Publications. January 2010, č. 6, s. 104–11, [cit. 2023-02-19]. ISSN 0975-4024. Dostupné z: https://www.researchgate.net/publication/308022297_Mobile_application_and_its_global_impact.

[9] Sheikh, A. et al. Smartphone: Android Vs IOS. *The SIJ Transactions on Computer Science Engineering its Applications (CSEA)* [online]. India: The Standard International Journals (The SIJ). October 2013, sv. 1, č. 4, s. 141–146, [cit. 2023-03-17]. DOI: 10.9756/SIJCSEA/V1I4/0104600401. ISSN 2321 – 2381. Dostupné z: https://www.researchgate.net/publication/335221195_Smartphone_Android_Vs_IOS.

[10] Sinha et al. Best Practices For Improving User Interface Design. *International Journal of Software Engineering Applications (IJSEA)-ERA Indexed* [online]. San Diego, California, USA: Department of Engineering and Computing National University. September 2019, sv. 10, č. 5, s. 71–83, [cit. 2023-03-27]. DOI: 10.5121/ijsea.2019.10505. Dostupné z: https://www.academia.edu/40965105/Best_Practices_For_Improving_User_Interface_Design.

[11] Stauffer, B. What Are 21st Century Skills? *Applied Educational Systems* [online]. January 2022. 2023 [cit. 2023-04-12]. Dostupné z: https://www.aeseducation.com/blog/what-are-21st-century-skills?fbclid=IwAR2C2IknERMvW2pbW1uW7IB8BLClTuyUJOhj0IRFyMgYnfi6mx_diRIukMs.

[12] Taylor, P. Mobile Operating System Market Share Worldwide. *Statista* [online]. Germany: [b.n.], 21. February 2023 [cit. 2023-02-10]. Dostupné z: https://www.statista.com/statistics/272698/global-market-share-held-by-mobile-operating-systems-since-2009/. Path: Technology.

[13] Truong, D. How To Design a Mobile Application to Enhance Teaching and Learning? *International Journal of Emerging Technologies in Learning (iJET)* [online]. FL, USA: Embry Riddle Aeronautical University, Daytona Beach. 2014, sv. 9, č. 3, s. 4–11, [cit. 2023-02-15]. DOI: 10.3991/ijet.v9i3.3507. ISSN 1863-0383. Dostupné z: https://online-journals.org/index.php/i-jet/article/view/3507/3143?fbclid=IwAR37nX_4IaQIb7R-uSJTFcQEsAsWo3rE4ezwvjblWpMBCSsZc4_LvaoTu8M.

[14] Verma, S. Mobile phone. [online]. 1. vyd. 2008, s. 1–14, [cit. 2023-02-18]. Dostupné z: https://www.academia.edu/21262654/Mobile_phone.

[15] Vithani, T. a Kumar, A. Modeling the Mobile Application Development Lifecycle. *Proceedings of the International MultiConference of Engineers and Computer Scientists* [online]. Hong Kong: IMECS. March 2014, sv. 1, s. 12–14, [cit. 2023-03-24]. ISSN 2078-0966. Dostupné z: https://www.iaeng.org/publication/IMECS2014/IMECS2014_pp596-600.pdf?fbclid=IwAR1aCE3Ywax1reZE2OQhw4sr8cgRJrTvKU2KMfngNpQo-CCYxSXC4mwCpHI.

[16] Zounek, J. a Sudický, P. *E-learning: učení (se) s online technologiemi.* Praha: Wolters Kluwer, 2012. 226 s. ISBN 978-80-7357-903-6.

[17] Learning React Native by. *O'Reilly* [online]. Gillis,Alexander. O'Reilly Media, Inc., 2023. 2023 [cit. 2023-03-22]. Dostupné z: https://www.oreilly.com/library/view/

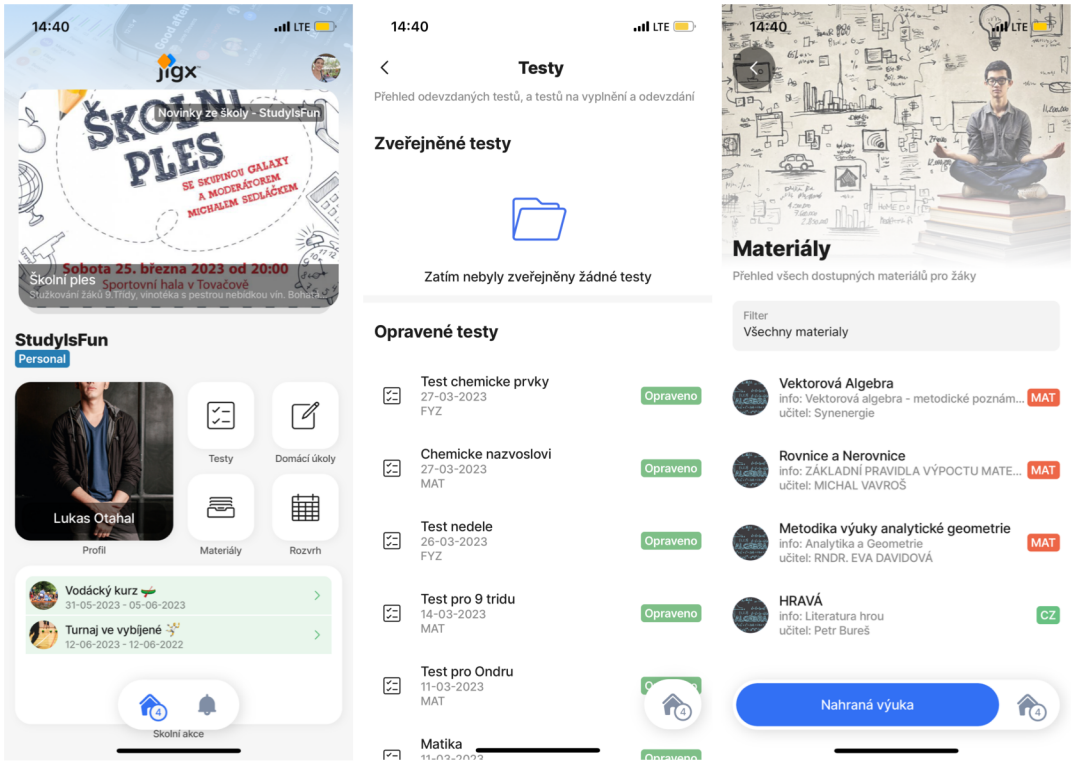
learning-react-native/9781491929049/ch01.html?fbclid=
IwAR3JRUyzK_cMHMrg2SZerdzJwH4GctaR1VgwuVYSdCxprBaFx850muoN-LQ.

Příloha A

Obsah CD

- /xotaha03-BP.pdf – elektronická verze písemné zprávy
- /doc.zip – zkomprimované zdrojové soubory písemné zprávy
- /source-code.zip – zkomprimované zdrojové kódy
- /video.mp4 – video k aplikaci
- /jigx.apk – instalační balíček pro Android
- /readme.md – dokumentace k instalaci a přihlášení

A.1 Ukázka finální aplikace



Obrázek A.1: Ukázka finální aplikace, kde na prvním obrázku je vidět domovská obrazovka aplikace, na druhém je vidět přehled testů a na třetím obrázku je přehled studijních materiálů

A.2 Schéma entit

Obrázek A.2: Schéma databáze a jednotlivé entity pro data v aplikaci

A.3 Ukázka dotazníku

StudyIsFun

Dobrý den,

jmenuji se Lukáš Otáhal a jsem bývalým studentem této školy. Momentálně studuji Fakultu informatiky na Vysokém učení technickém v Brně a ve své bakalářské práci se věnuji vývoji mobilní aplikace pro usnadnění školní výuky a komunikace se studenty.

Na základě shlédnutí úvodní prezentace, kde jsem mobilní aplikaci komplexně představil a Vašeho následného vyzkoušení jednotlivých funkcí aplikace, Vás prosím o vyplnění následujícího, krátkého dotazníku.

Mockrát děkuji!

1 Uveďte Vaši třídu

Nápověda k otázce: *(př. 6B apod.)*

[ ]

2 Uveďte Vaše pohlaví

Nápověda k otázce: *Vyberte jednu odpověď*

☐ Žena ☐ Muž ☐ Nechci uvést

3 Jak často využíváte mobil pro studium?

Nápověda k otázce: *Vyberte jednu odpověď*

○ Často ○ Občas ○ Téměř vůbec

4 Jak se Vám s aplikací pracovalo (uživatelská přívětivost)?

Nápověda k otázce: *Vyberte jednu odpověď!*

☐ Velmi dobře ☐ Dobře ☐ Špatně ☐ Velmi špatně

5 Jak hodnotíte rychlost aplikace? Z hlediska přechodů mezi stránkami, otevírání dokumentů a odkazů.

Nápověda k otázce: *Vyberte jednu odpověď!*

☐ Velmi dobře ☐ Dobře ☐ Špatně ☐ Velmi špatně

1

6 Myslíte si, že byste tuto aplikaci využili při Vašem studiu ve škole?

Nápověda k otázce: *Vyberte jednu nebo více odpovědí*

☐ Ano ☐ Nevím ☐ Ne

7 Na aplikaci mám názor:

Nápověda k otázce: *Vyberte jednu nebo více odpovědí!*

☐ Intuitivní (návodná) a snadná na ovládání
☐ Chaotická a složitá na ovládání
☐ Raději bych aplikaci využíval na počítači
☐ Raději bych se učil/a pomocí klasických metod (tužka, papír, učebnice apod.)
☐ Líbilo by se mi, kdyby se aplikace využívala při výuce ve škole
☐ Jiné připomínky doplňte prosím... _____

8 Váš celkový dojem z aplikace

Nápověda k otázce: *1 = nejhorší hodnocení ; 5 = nejlepší hodnocení*

☆☆☆☆☆ ☐ / 5

9 Které sekce Vám přišli nejvíce užitečné?

Nápověda k otázce: *Změňte pořadí položek dle svých preferencí (1. - nejdůležitější, poslední - nejméně důležitá)*

Novinky ze školy	☐
Profil s klasifikacemi	☐
Testy	☐
Domácí úkoly	☐
Materiály	☐
Rozvrh	☐
Školní akce	☐

10 Co Vám v aplikaci chybí?

11 Co se Vám v aplikaci líbilo?

12 Co byste v aplikaci vylepšili?