

UNIVERZITA PALACKÉHO V OLOMOUCI

PEDAGOGICKÁ FAKULTA

Katedra technické a informační výchovy

## Bakalářská práce

David Veselý

Návrh a tvorba aplikace pro operační systém Android

Olomouc 2016

Vedoucí práce: Mgr. Jan Kubrický, Ph.D.

**Prohlášení:**

Prohlašuji, že jsem bakalářskou práci zpracoval samostatně a použil jen prameny uvedené v seznamu literatury.

V Olomouci dne .....

.....

David Veselý

**Poděkování:**

Děkuji vedoucímu bakalářské práce, Mgr. Janu Kubrickému, Ph.D., za jeho čas, cenné připomínky a podněty při vzniku této práce. Také děkuji zaměstnancům a vedení společnosti T.R.I., s.r.o. za poskytnutou podporu.

# Obsah

Úvod .....	6
1 Aplikace a systém Android .....	7
1.1 Android .....	7
1.1.1 Historie Android .....	7
1.1.2 Verze Android .....	8
1.1.3 Rozšířenost OS Android .....	8
1.1.4 Architektura systému .....	9
1.1.5 Zařízení s Android OS .....	10
1.2 Aplikace .....	10
2 Vývojová prostředí pro Android .....	12
2.1 Android Studio .....	13
2.1.1 Užitečné funkce .....	13
2.1.2 Java .....	14
2.1.3 XML .....	15
2.1.4 Propojení mezi jazyky Java a XML .....	15
3 Návrh a tvorba aplikace „Lodní deník“ .....	17
3.1 Základní informace .....	17
3.2 Související aplikace .....	17
3.2.1 BoatBook Sailing Log .....	17
3.2.2 BoatLogger .....	18
3.3 Návrh demonstrativní aplikace .....	18
3.3.1 Lodní deník .....	18
3.3.2 Návrh lodního deníku .....	19
3.4 Příprava vývojového prostředí .....	19
3.5 Použité prvky .....	19

3.5.1	Fragmenty .....	19
3.5.2	Recycler View .....	20
3.5.3	Material design .....	20
3.5.4	SQLite.....	21
3.6	Vytvoření databáze .....	21
3.6.1	Řazení dat dle data.....	22
3.7	Založení nového projektu .....	22
3.8	Tvorba aktivity.....	24
3.8.1	Životní cyklus aktivity.....	24
3.8.2	První aktivita .....	24
3.9	Objekty.....	25
3.10	Texty s využitím String-array.....	26
	String-Array.....	26
3.11	Ikona .....	27
3.12	Budoucí vývoj aplikace .....	27
4	Publikace aplikace .....	28
4.1	Možnosti distribuce.....	28
4.1.1	Google Play .....	28
4.1.2	QR kód („Quick Response“) .....	28
4.2	Vývojářský účet .....	29
4.3	Google developer console.....	29
4.3.1	Přidání a schvalování aplikace .....	29
4.3.2	Statistické informace .....	31
	Závěr.....	32
	Referenční seznam.....	33
	Seznam příloh.....	35
	ANOTACE.....	39

## Úvod

Chytré mobilní telefony či tablety jsou stále více rozšířené, čímž se dostávají aplikace do denního používání milionu uživatelů. Všechna tato zařízení fungují na operačních systémech, přičemž nejrozšířenějším užívaným systémem je OS Android [1], kterým se zabývá tato bakalářská práce.

Cílem práce je prvně charakterizovat základní vlastnosti operačního systému Android v kontextu vývoje jeho aplikací a informovat o základní problematice, historii a fungování tohoto systému. Pro pochopení celé problematiky je nutné blíže rozebrat jednotlivá vývojová prostředí pro Android, která slouží právě pro samotné naprogramování celé aplikace.

Pro hlubší pohled do problematiky aplikací a srozumitelnost programování v OS Android, práce pojednává o vytvoření a publikaci aplikace „Lodní deník“. Tato aplikace byla vybrána proto, aby zjednodušila práci posádkám na lodích, ať už na mořích či řekách ve vnitrozemí. Aplikace by měla uživateli poskytnout databázi všech svých lodních deníků i se všemi záznamy a informacemi o posádce.

Naprogramováním kompletní aplikace však práce programátora nekončí, také je nutné ji dát k dispozici široké veřejnosti. Proto se poslední kapitola zabývá publikací samotné aplikace, aby byla dostupná ke stažení všem uživatelům na veřejně dostupných serverech (např. Google Play), ať už zdarma nebo za poplatek. Ani tento krok by se neměl podceňovat, proto je potřebné vědět, jak správně aplikaci publikovat a jaké podmínky je nutné splnit.

Tato práce by měla komplexně objasnit fungování a vývoj aplikace, aby si každý uživatel dokázal představit, co se skrývá za každým klepnutím na jeho dotykovou obrazovku. Samotná aplikace má svému uživateli poskytnout jednoduché a přitom kompletní vedení lodního deníku podle stanovených pravidel a zároveň být zajímavou databází všech cest každého „námořníka“.

# 1 Aplikace a systém Android

První kapitola se zabývá obecnou terminologií, která přiblíží základní pojmy využívané v této práci. Zaměřuje se především na charakteristiku mobilní aplikace a systému Android. Podstatné je si uvědomit, že Android je tvořen právě jednotlivými aplikacemi, které zajišťují funkčnost celého systému.

## 1.1 Android

Android je především open-source mobilní operační systém vyvíjen americkou společností Google, Inc. Avšak pod pojmem Android se kromě úspěšného operačního systému skrývá také open-source<sup>1</sup> vývojářská platforma pro vytváření mobilních aplikací do OS Android. V dnešní době se OS Android používá nejen v mobilních telefonech, ale také v tabletech, televizích, počítačích, chytrých hodinkách apod.

### 1.1.1 Historie Android

V roce 2003 Andy Rubin, Rich Miner, Nick Searsem a Chris White založili akciovou společnost Android, Inc. Následně v roce 2005 byla společnost Android, Inc. koupena společností Google Inc., která z ní vytvořila svou dceřinou společnost. Vedoucím společnosti se stal bývalý zakladatel Andy Rubin, za jehož působení začal vývoj mobilního operačního systému. [9]

Jedním z nejvýznamnějších okamžiků společnosti byl 5. listopad 2007, kdy byla vytvořena tzv. OHA (Open Handset Alliance), což bylo seskupení výrobců mobilních telefonů za účelem vytvoření standardu pro mobilní zařízení. V dnešní době je konsorcium OHA tvořeno 84 společnostmi jako například Google, Intel, Samsung, HTC, Dell, T-mobile a další. [9]

První telefon na světě s operačním systémem Android byl představen firmou HTC v říjnu 2008. Od té doby se OS Android velmi změnil a zmodernizoval. Jednotlivé verze OS Android jsou pojmenovávány podle amerických sladkostí již od verze 1.5 a tato tradice dále přetrvává. [9]

---

<sup>1</sup> Open-source software znamená, že autor softwaru vydal zdarma k dispozici i zdrojové kódy, které uživatelé mohou dále upravovat.

### 1.1.2 Verze Android

Android je vyvíjen jak pro mobilní telefony, tak pro tablety, televize, počítače apod. Dnes se používá jedna verze pro všechny platformy. Výjimkou byla pouze verze 3.0 (HoneyComb), která byla vyvíjena pouze pro tablety a na mobilních telefonech byla nepoužitelná. Od verze 4.0 se vývoj vrátil zpět k univerzálnímu využití pro všechny platformy. S každou novou verzí systému vychází také nové aplikační rozhraní API<sup>2</sup>. V Tabulce 1-1 je aktuální přehled všech oficiálních doposud vydaných veřejných verzí OS Android.

Tabulka 1-1 Přehled verzí Android

<b>Přehled verzí Android platných k 4. 11. 2015</b>			
<b>Verze</b>	<b>Název</b>	<b>Rok vydání</b>	<b>API</b>
1.5	Cupcake	2009	3
1.6	Donut	2009	4
2.0 – 2.1	Eclair	2009	5-7
2.2	Froyo	2010	8
2.3 – 2.4	Gingerbread	2010	9,10
3.0 – 3.2	Honeycomb	2011	11-13
4.0 – 4.0.3	Ice Cream Sandwich	2011	14, 15
4.1 – 4.3	Jelly Bean	2012	16-18
4.4 – 4.4.4	KitKat	2013	19, 20
5.0 – 5.1.1	Lollipop	2014	21, 22
6.0	Marshmallow	2015	23

Zdroj: vlastní tvorba dle [18]

### 1.1.3 Rozšířenost OS Android

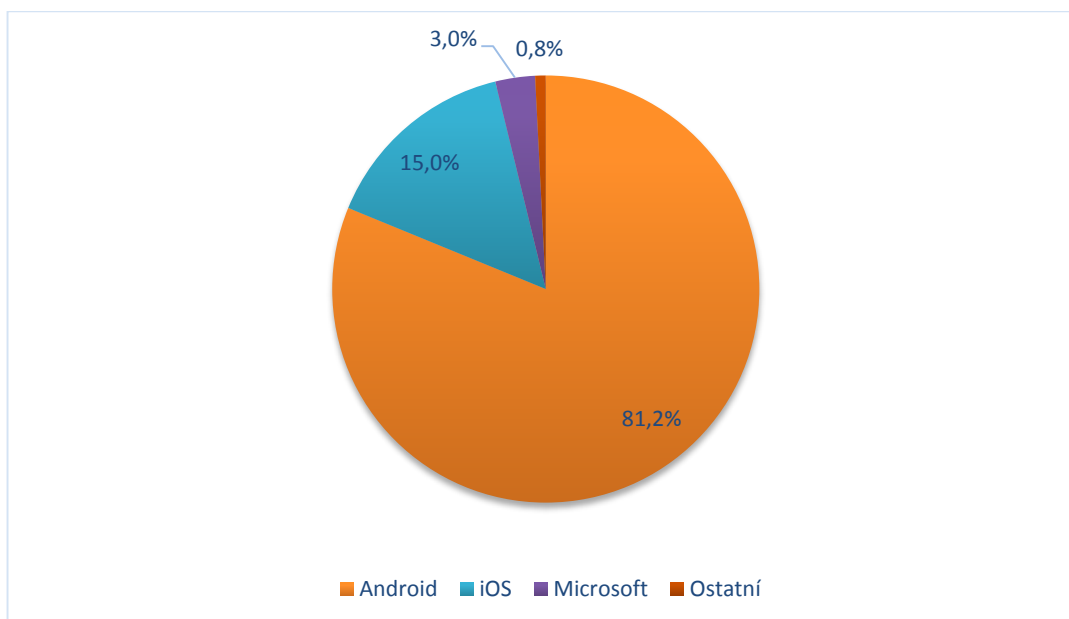
Tím, že je Android open-source neboli otevřený software, který do svých mobilních telefonů používá většina výrobců, zastává prvenství na světovém trhu mobilních telefonů. Podíl operačních systémů v jednotlivých zařízeních shrnuje Graf 1-1.

---

<sup>2</sup> API je sada nástrojů a protokolů pro vytváření aplikací. Slouží k usnadnění práce programátorovi.



Graf 1-1 Podíl operačních systémů v mobilních zařízeních v roce 2014



Zdroj: vlastní tvorba dle [1]

#### 1.1.4 Architektura systému

Operační systém Android je rozdělen do 5 samostatných vrstev, které mezi sebou spolupracují. [19]

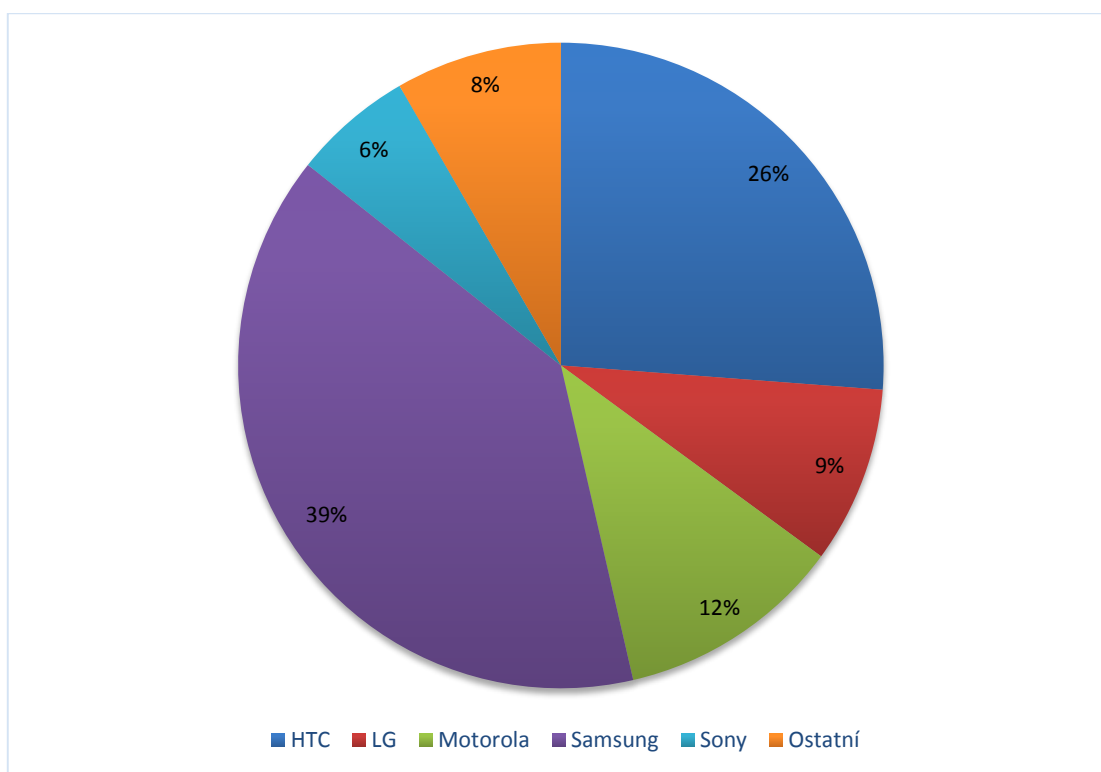
- **Linux Kernel** je nejnižší vrstvou architektury a tvoří základ (jádro) operačního systému. Slouží především ke komunikaci mezi hardwarem a softwarem, která je zajištěna pomocí ovladačů (driverů). Android nemá sám o sobě vlastní jádro, ale využívá linuxové jádro ve verzi 2.6.
- **Libraries** neboli v překladu knihovny, zajišťují základní funkce systému, které je poté možné použít při tvorbě vlastní aplikace. Patří sem např. video kodeky nebo odlehčená verze databáze SQLite.
- **Android Runtime** je vrstva sloužící ke správnému běhu aplikací a obsahuje knihovny programovacího jazyka Java. Vzhledem k tomu, že Java není nativním jazykem pro OS Android, je nutný virtuální stroj, který celou aplikaci překládá, což zajišťuje právě tato vrstva. Dříve Android používal virtuální stroj Dalvik Virtual Machine, avšak v posledních verzích Androidu byl již plně nahrazen ART (Android Runtime).
- **Application Framework** patří k nejdůležitější vrstvě pro programování aplikací. Obsahuje další užitečné knihovny, které již jsou napsané v Javě. Umožňuje vývojáři přístup k mnoha prvkům jako např. kontakty či fotoaparát. Pomocí těchto knihoven je možné vytvářet efektivnější a praktičtější aplikace.

- **Application** je nejvyšší vrstvou a patří sem jednotlivé aplikace, které jsou uživateli dostupné buď v oficiálním obchodě Google Play nebo je možná instalace pomocí aplikačních souborů APK.

### 1.1.5 Zařízení s Android OS

OS Android může do svých zařízení instalovat jakýkoliv výrobce a to díky tomu, že je univerzální a cenově nejvýhodnější. I když na americkém trhu zastává Android prvenství jen o pár procent, tak v celosvětovém měřítku je Android nejpoužívanější OS pro mobilní zařízení. Využití OS Android u jednotlivých výrobců znázorňuje Graf 1-2.

Graf 1-2 – Přehled nejpoužívanějších výrobců s OS Android k 15. 2. 2016



Zdroj: vlastní tvorba dle [13]

## 1.2 Aplikace

Slovo aplikace má v informatice více významů, avšak mezi nejčastější charakteristiku patří např. dle počítačového webového portálu PCMAG, že za aplikaci můžeme považovat veškerý software, který byl vytvořen pro operační systém. [2] Do aplikací můžeme zařadit např. MS Word, který je součástí kancelářského balíku Microsoft Office, avšak operační systém např. Windows, Linux apod. za aplikaci považovat nemůžeme. Moderním označením především pro mobilní aplikace je zkratka „app“ z anglického „application“ – aplikace.

První taková mobilní aplikace přišla na svět v roce 1998, kdy tehdejší společnost Nokia (dnes Microsoft) vydala legendární hru Snake (Had) či kancelářské aplikace „kalendář“ a „kalkulačka“.

Základem aplikace v OS Android jsou jednotlivé aplikační komponenty. Každá komponenta má vlastní životní cyklus a zajišťuje odlišný způsob přístupu systému k aplikaci. V OS Android existují čtyři typy aplikačních komponent. [20]

1. **Activity** (dále jen „aktivita“): představuje jednu obrazovku s uživatelským rozhraním.
2. **Services**: jedná se o komponentu, která není poskytována uživatelskému rozhraní a běží na pozadí, aby uživatel mohl pracovat na něčem jiném.
3. **Contents providers**: zajišťuje správu a sdílení aplikačních dat, jako např. databáze SQLite.
4. **Broadcast receivers**: slouží k naslouchání z aplikace nebo ze zařízení. Při zjištění daného stavu dokáže spustit požadovanou reakci. Například pokud je spuštěná aplikace pro chat a zařízení ztratí internetové připojení, spustí se akce, která na tuto skutečnost uživatele upozorní.

## 2 Vývojová prostředí pro Android

Vývojové prostředí IDE (integrated development environment) je programovací prostředí, které je v jednom balíku a tvoří aplikační software. Většinou se skládá z editoru, překladače, debuggeru a uživatelského rozhraní. [5]

V současné době lze využívat mnoha vývojových prostředí. Mezi nejznámější vývojová prostředí pro Android OS patří:

- **Android Studio:** je moderní vývojové prostředí, které vyšlo v červnu 2013 a je k dispozici zdarma. Jedná se o společné dílo společnosti Google, Inc. a JetBrains<sup>3</sup>, vytvořené na IntelliJ IDEA<sup>4</sup>. V Android studiu se využívá programovací jazyk Java spolu s XML.
- **Eclipse:** patří k nejpoužívanějším vývojovým prostředím pro tvorbu Java aplikací. Do roku 2013 byl nejlepším prostředím pro vývoj aplikací pro Android OS. Pomocí mnoha doplňků je možné v Eclipse programovat mimo Javy i v C, C++ či PHP aj. Eclipse původně vzniklo ve společnosti IBM, od roku 2004 jej vyvíjí nadace Eclipse foundation.[8]
- **Xamarin studio:** je moderní multiplatformní vývojové prostředí od společnosti Xamarin. Oproti Android studio a Eclipse se Xamarin liší tím, že není zdarma, ale k používání je zapotřebí si zaplatit paušální licenci. Cena se liší podle edice a jednotlivých doplňků. Dalším rozdílem je, že ve Xamarinu se programuje v jazyce C# a je možné aplikaci zkompileovat do zařízení s operačním systémem Android, iOS a Windows mobile. V dubnu 2016 koupila společnost Microsoft celý projekt Xamarin a slibuje, že příští vydání bude již bezplatné.
- **Netbeans:** je oficiální vývojové prostředí vytvořeno jazykem Java. Velkou výhodou je multijazyčnost prostředí. Lze zde programovat v jazycích jako je Java, C, C++, PHP apod. Mimo to je možné připojit i Android SDK a vyvíjet mobilní aplikac zde. Avšak pro Android OS se Netbeans příliš nepoužívá. [7]

---

<sup>3</sup> JetBrains – Společnost, která vznikla v Praze v roce 2000 a dnes patří k leaderům na trhu s vývojářskými nástroji.

<sup>4</sup> IntelliJ IDEA – vývojové prostředí společnosti JetBrains, které inteligentně pomáhá a kontroluje kód v reálném čase.

## 2.1 Android Studio

Android Studio je oficiální vývojářské prostředí na platformu Android OS od společnosti Google Inc. K dispozici je zdarma pro zařízení s operačním systémem Windows, Linux a MacOS. Android studio vyžaduje nainstalovaný balík Java JDK od společnosti Oracle, který je zdarma ke stažení na oficiálním webu společnosti Oracle. Odkaz ke stažení prostředí Android Studio je k dispozici na: <http://developer.android.com/sdk/index.html>.

### 2.1.1 Užitečné funkce

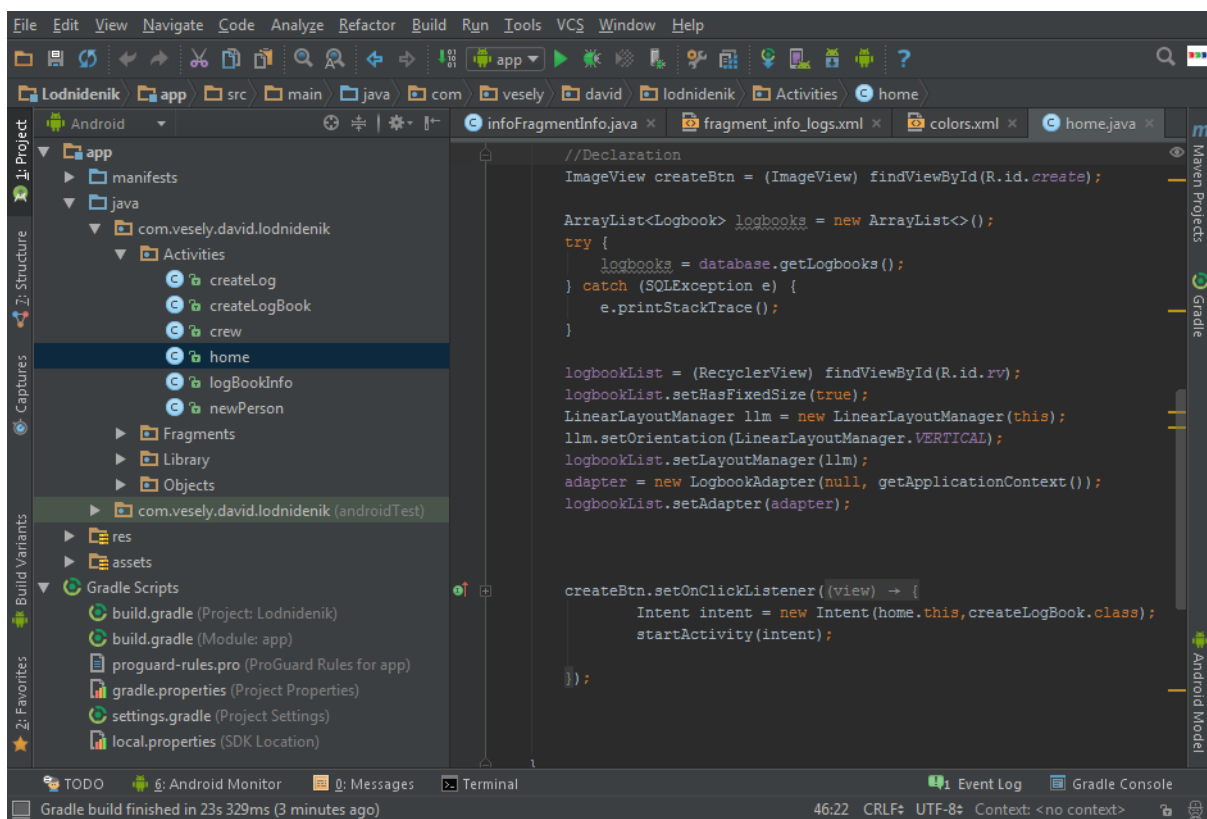
Android studio umožňuje vývojáři používat mnoho funkcí a komponentů pro zjednodušení práce. Mezi často používané funkce patří:

- **Kontrola verzí:** Android Studio umožňuje pracovat na jednom projektu v týmu lidí, kteří si mezi sebou automaticky vyměňují jednotlivé změny v projektu. Mezi nejpoužívanější služby patří Git a Subversion.
- **Import projektů:** je velmi příjemnou službou pro programátory, kteří přechází z vývojového prostředí Eclipse. Android Studio umožňuje přesun projektu včetně všech nastavení z Eclipse .
- **AVD manager:** slouží k vytváření a používání virtuálních zařízení. Android Studio přímo používá AVD manager z balíku Android SDK. To umožňuje vývojáři simulovat jednotlivá zařízení, avšak efektivita a rychlost těchto virtuálních zařízení je velmi špatná. Proto existují konkurenční programy jako například Genymotion<sup>5</sup>, který je pro nekomerční použití zdarma a je výrazně rychlejší.
- **Android monitor:** slouží k zobrazování informací o výkonu, logů a vytíženosti sítě v reálném čase během testování aplikace. Tento nástroj upozorňuje vývojáře o činnostech a chybách, které se dějí na pozadí aplikace.

---

<sup>5</sup> Genymotion je software pro virtualizaci Android zařízení od společnosti GenyMobile.

Obrázek č. 1 - Ukázka vývojového prostředí Android studio během vývoje aplikace "Lodní deník"



Zdroj: vlastní tvorba v Android Studio

## 2.1.2 Java

Java je objektivě orientovaný programovací jazyk, který není závislý na platformě a díky tomu zajišťuje univerzálnost použití. Nezávislost je zajištěna tím, že kompilace se nepřevádí přímo na strojový kód procesoru, ale do tzv. „Byte-kódu“. Tento kód je poté při spuštění Java aplikace rychle převeden na strojový kód aktuálního procesoru a operačního systému. Současná verze jazyku Java je 8.

Na obrázku č. 2 je zobrazena ukázka programovacího jazyku Java z aplikace *Lodní deník*.

Obrázek č. 2 - Ukázka Java kódu

```
public class Logbook {
    private int id;
    private String name;
    private String country;
    private String date;
    private String from;
    private String to;

    public Logbook(int id, String name, String country, String date){
        this.id = id;
        this.name = name;
        this.country = country;
        this.date = date;
    }
}
```

Zdroj: vlastní tvorba v Java

### 2.1.3 XML

Extensible Markup Language (XML) neboli rozšiřitelný značkovací jazyk je vytvořený konsorciem W3C. Jedná se o univerzální formát souborů, který je čitelný bez použití speciálních softwarů. Hlavní výhodou XML je univerzálnost a multiplatformovost. [10]

Ukázka kódu z jazyku XML je zobrazena na obrázku č. 3. Na obrázku č. 4 je grafická ukázka kódu po rendrování<sup>6</sup>.

### 2.1.4 Propojení mezi jazyky Java a XML

Propojení značkovacího jazyka XML s programovacím jazykem Java je možný dvěma způsoby:

- **Z XML:** pomocí metody „android:onClick“, která umožňuje volat metodu napsanou v jazyce Java. Tento způsob se doporučuje pouze u jednoduchých operací. Ukázka této metody je zobrazena na obrázku č. 3.
- **Z Javy:** Java umožňuje mnohem více operací s jednotlivými prvky, avšak každý grafický prvek musí být předem deklarován a přetypován na prvek z jazyka XML.

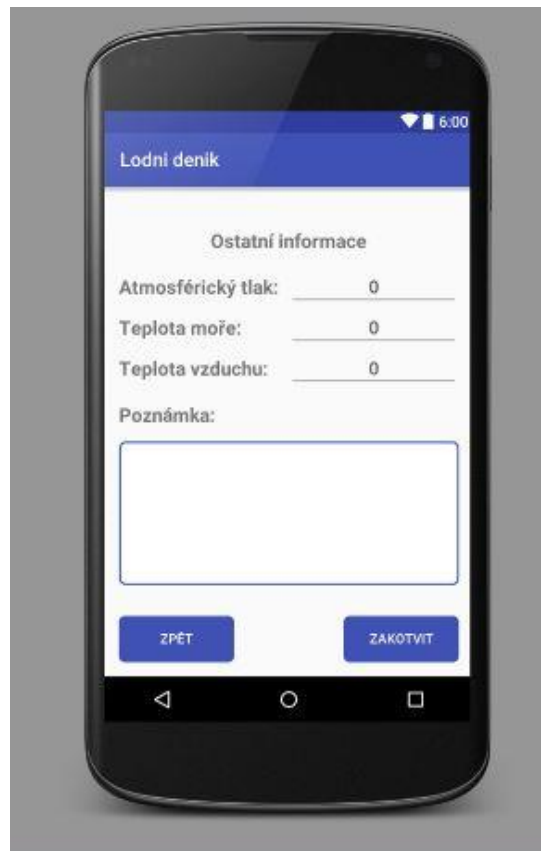
---

<sup>6</sup> Renderování (anglicky rendering) je proces, při němž ze zadaných dat vzniká cílový obraz (počítačová grafika).[14]

Obrázek č. 3 - Ukázka kódu z XML včetně metody „onClick“

```
<Button  
    android:id="@+id/button1"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="Button"  
    android:onClick="buttonClick" />/>
```

Obrázek č. 4 - Ukázka kódu XML po renderování





### 3 Návrh a tvorba aplikace „Lodní deník“

Následující text se zabývá samotným návrhem a postupem při tvorbě aplikace pro OS Android.

#### 3.1 Základní informace

Před začátkem tvorby aplikace si musí každý programátor nejprve zvolit v jakém vývojovém prostředí a programovacím jazyku chce pracovat. Toto rozhodnutí může ovlivnit několik faktorů, proto se doporučuje vytvořit si abstraktní návrh aplikace (ovládání, speciální prvky apod.) a podle toho si vhodně zvolit prostředí, ve kterém by tento návrh byl realizovatelný. Pro tuto práci bylo zvoleno vývojové prostředí Android Studio s programovacím jazykem Java a XML.

#### 3.2 Související aplikace

V obchodě Google Play již existuje několik aplikací, které svou funkčností připomínají lodní deník, avšak žádná z nich neobsahuje všechny náležitosti, které lodní deník musí obsahovat.

##### 3.2.1 BoatBook Sailing Log

Je jedna z nejlépe propracovaných aplikací na zapisování lodního deníku. Celá aplikace je v anglickém jazyce a obsahuje mnoho funkcí od ručních zápisů po automatické zaznamenávání mapy. Velkou výhodou aplikace je také vzdálený přístup k údajům z webu. Nevýhodou je, že aplikaci nelze používat v tuzemských vodách a je pevně vázána na GPS.

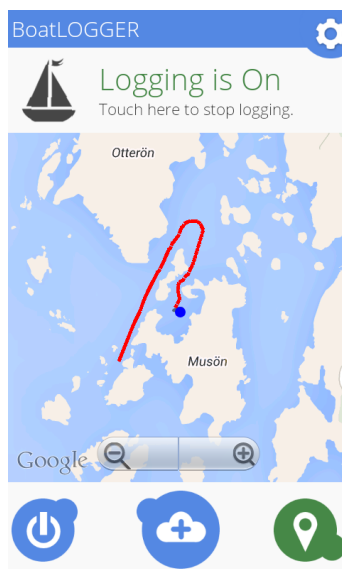
Obrázek č. 5 – Ukázka aplikace BoatBook Sailing Log z GooglePlay



### 3.2.2 BoatLogger

BoatLogger je aplikace, která byla vytvořena pro „BoatLogger ecosystem“. Aplikace je anglickém jazyce a umožňuje nahrávání trasy a přidávání informací o plavbě. Avšak stejně jako aplikace *BoatBook Sailing Log* vyžaduje pro používání registraci a plný přístup k GPS.

Obrázek č. 6 – Ukázka aplikace BoatLogger z GooglePlay



### 3.3 Návrh demonstrativní aplikace

Výstupem této práce je aplikace, která může v reálném životě nahradit během plavby klasické papírové lodní deníky a tím zapisovateli ušetřit čas.

#### 3.3.1 Lodní deník

Povinnost vést lodní deník je stanovena v zákoně č. 114/1995 Sb., o vnitrozemské plavbě a zákoně č. 61/2000 Sb., o námořní plavbě. Lodní deník musí minimálně obsahovat tyto náležitosti: [21], [22]

- pořadové číslo lodního deníku,
- jméno lodi,
- evidenční označení,
- informace o posádce,
- podrobné informace o plavbě v každém záznamu.

### 3.3.2 Návrh lodního deníku

Aplikace *Lodní deník* obsahuje kromě povinných náležitostí také mnoho dalších možností pro přehlednost a univerzálnost deníku. Účelem této aplikace je, aby ukládala záznamy o jednotlivých plavbách a uživatel se tak mohl jednoduše podívat na kteroukoliv ze svých plaveb.

## 3.4 Příprava vývojového prostředí

Každé vývojové prostředí pro tvorbu aplikací na operační systém Android vyžaduje mít k dispozici balíček SDK (Software Development Kit), který obsahuje sadu vývojových nástrojů sloužících k vytváření aplikací pro daný OS. Balík SDK je již ve výchozí instalaci vývojového prostředí Android Studio. V nástrojové liště Android Studia je umístěn odkaz na balík SDK, ve kterém je zapotřebí si zvolit pro které API bude možné aplikaci zkompileovat.

Je potřeba si dobře rozmyslet, se kterými zařízeními má být aplikace kompatibilní. Takové rozhodnutí může programátorovi ušetřit spoustu času a budoucích problémů. Například tvořit komunikační aplikaci pro API 23 a zároveň pro API 3 je velmi neefektivní, složité a pravděpodobně si tuto aplikaci nikdo s API 3 nestáhne, jelikož se jedná o zastaralou verzi.

## 3.5 Použité prvky

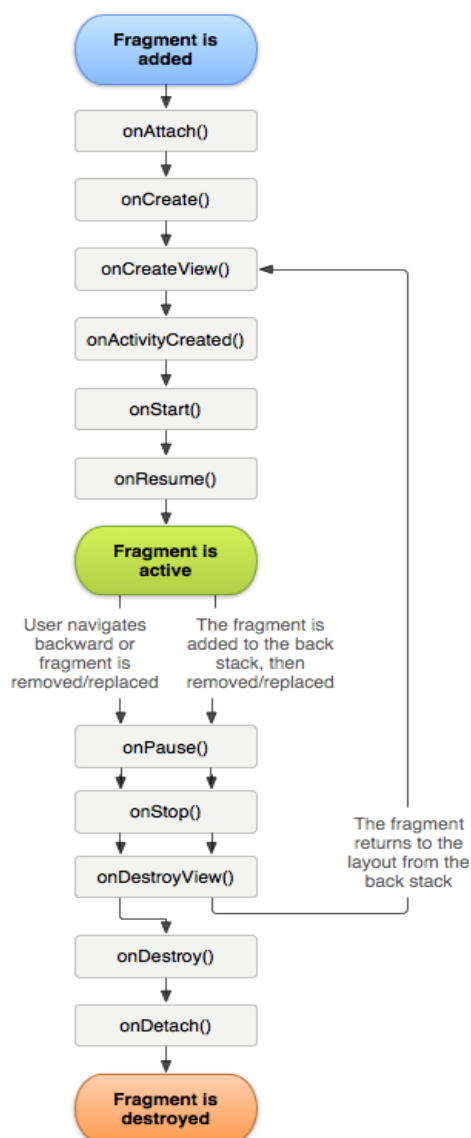
Kromě základních prvků programovacího jazyka Java bylo pro aplikaci použito i několik rozšiřujících doplňků, které aplikaci zajišťují plynulý chod a jednodušší ovládání.

### 3.5.1 Fragmenty

Fragmenty byly do systému Android přidány ve verzi 3.0. Jedná se o komponenty, které běží v kontextu aktivity a primárně byly vytvořeny za účelem kompatibilního zobrazení mezi telefony a tablety. Každý fragment má vlastní životní cyklus, který je zobrazen na obrázku č. 7. Jedna aktivita může obsahovat i více fragmentů.

Fragmenty lze do aplikace vkládat buď staticky přímo v XML nebo programově při použití metody `FragmentManager`. Velkou výhodou pro použití fragmentů je především rychlost a přehlednost. Při načítání více fragmentů se zpracovávají všechny fragmenty najednou, což znamená, že uživatel při procházení mezi jednotlivými fragmenty nemusí čekat na jejich načítání.

Obrázek č. 7 – Životní cyklus fragmentu



Zdroj: [6]

### 3.5.2 Recycler View

Je součástí material designu a jedná se o rozšířenou verzi klasického listView s lepším výkonem a dalšími výhodami. Používání RecyclerView je efektivnější a jednodušší než klasické listView.

### 3.5.3 Material design

Je designové 3D prostředí, což znamená, že se jednotlivé prvky pohybují po třech osách (x, y a z). Material design vydal Google spolu s API 21.

### 3.5.4 SQLite

Je malá, výkonná relační databáze, která ke své funkčnosti nepotřebuje žádný server ani konfigurační soubory. SQLite zastává jak funkce serveru, tak klienta a i přesto je výkon databáze vysoký a čtení dat má vyšší prioritu než zápis. SQLite se využívá i v jiných platformách než Android OS jako např. Windows, Mono, Python či iOS.

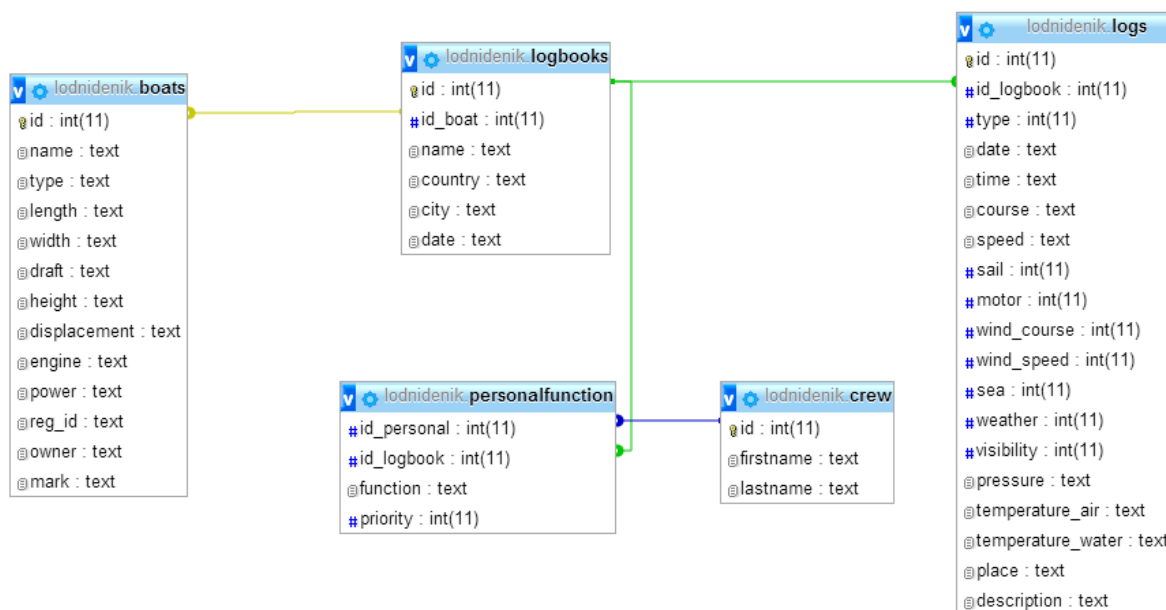
### 3.6 Vytvoření databáze

Jelikož účelem této aplikace je ukládání a načítání informací o plavbě, je potřeba vytvořit uložisko pro tato data. Nejjednodušším způsobem pro řešení je použití SQLite databáze, protože není potřeba přidávat žádné pomocné knihovny a databáze SQLite je součástí OS Android.

Tvorba databáze byla vytvořena pomocí open-source aplikace „DB Browser for SQLite“, která umožňuje vytváření SQLite databází v jednoduchém uživatelské prostředí. Tato aplikace umí mimo jiné také exportovat a importovat data mezi soubory pro SQL databáze a SQLite databáze.

V tomto projektu je soubor databáze pojmenován jako „database.db“ a je umístěn v aplikační složce „assets“, která je určena pro ukládání pomocných souborů. Návrh databáze včetně relací je zobrazen na obrázku č. 8.

Obrázek č. 8 - Grafický diagram databáze v aplikaci "Lodní deník"



Zdroj: Vytvořeno autorem v aplikaci phpMyAdmin

### 3.6.1 Řazení dat dle data

Databáze SQLite umožňuje pracovat pouze s těmito datovými typy: [16]

- **INTEGER** – hodnota celého čísla nebo 8 bajtů v závislosti na velikosti hodnoty,
- **TEXT** – textový řetězec, který je v databázi kódován pomocí UTF-8 a UTF-16,
- **BLOB** – umožňuje zapisovat obecná data v binárním tvaru,
- **REAL** – hodnota reálných čísel do 8 bajtů.

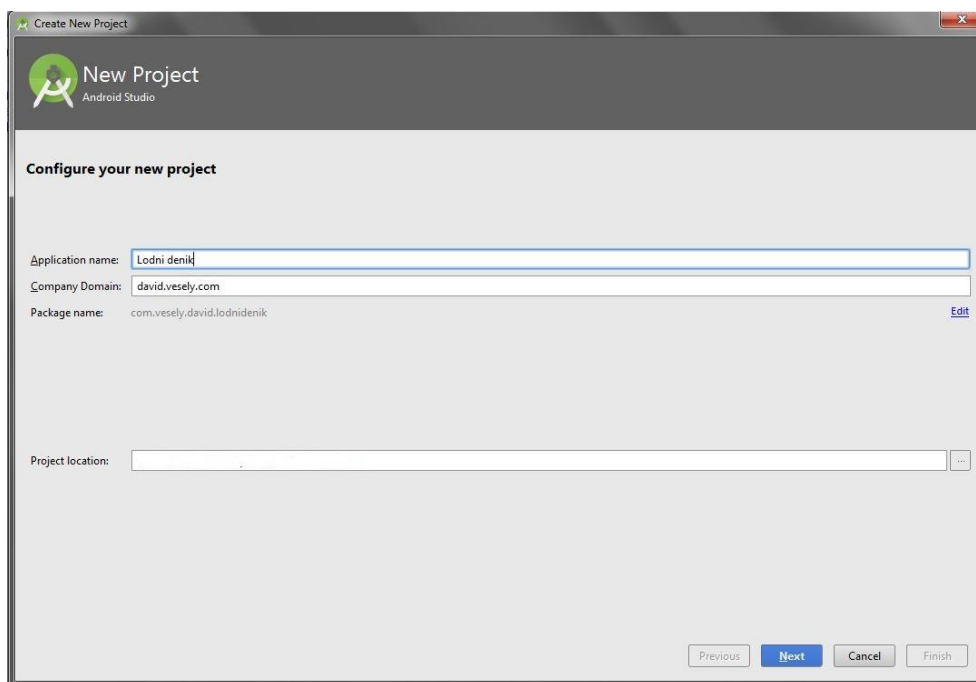
Problém u aplikace *Lodní deník* je ten, že jednotlivé záznamy v lodním deníku jsou řazeny podle času. Práci s časem a datem v jazyku Java zajišťuje datový typ „date“. Při zapisování informací z datového typu „date“ do datového typu „string“ v SQLite databázi dochází v případě řazení k problému. Datový typ „string“ během řazení porovnává jednotlivé znaky, nikoliv celá slova. Z toho plyne problém, že při řazení dat vzestupně bude například 29. 04. 2016 po 01. 05. 2016. Tento problém je v aplikaci *Lodní deník* vyřešen tím, že se informace z datového typu „date“ přetypují na datový typ „long“ a poté se nahrají do datového typu „string“ v databázi. Načítání dat z databáze poté pracuje na stejném principu, kdy se z databáze přetypuje datový typ „string“ na „long“ a dále na „date“ dle požadovaných parametrů.

### 3.7 Založení nového projektu

V prvním kroku při vytváření nového projektu je potřeba vyplnit název aplikace, název společnosti a adresář v počítači, do kterého se budou ukládat zdrojové kódy aplikace. Název společnosti slouží jako identifikátor a musí být jedinečný. Poté se vytvoří balík aplikace, jehož název se skládá z názvu společnosti, za který je doplněna tečka a názvu aplikace bez diakritiky.

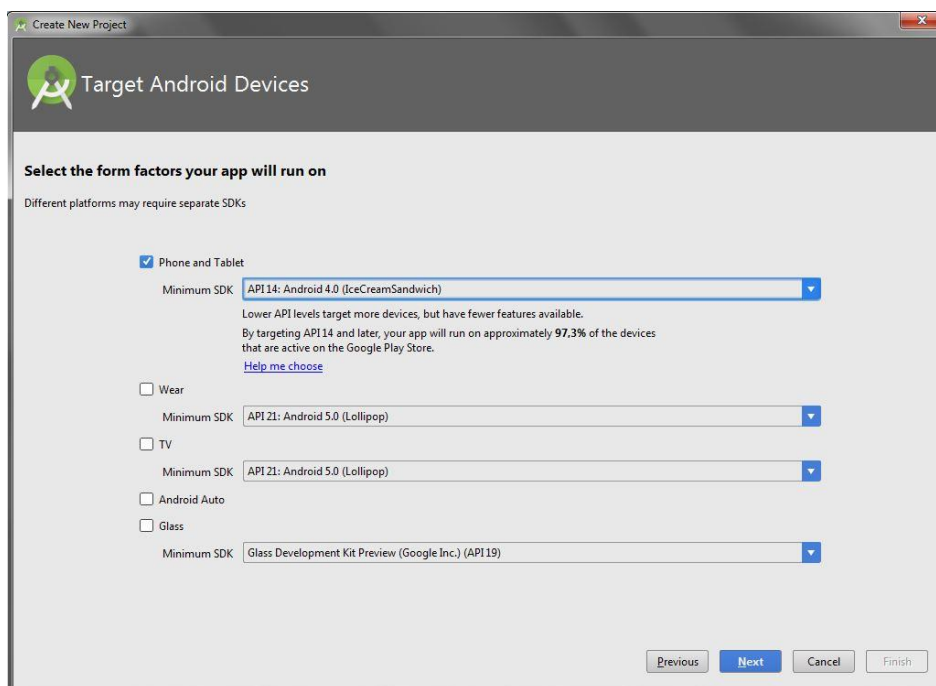
Ve druhém kroku je možné si zvolit, pro které API má být aplikace tvořena s ohledem na balík SDK. Mimo jiné je zde uvedeno, pro kolik procent zařízení bude aplikace kompatibilní. Tato aplikace je určena pro zařízení s API 14 a vyšší. Třetí krok se už zabývá pouze vytvořením první aktivity a je popsán v další podkapitole. Celý postup založení projektu je zobrazen na obrázku č. 9 a 10.

Obrázek č. 9 – První krok založení projektu - z aplikace „Lodní deník“



Zdroj: vlastní tvorba v Android Studio

Obrázek č. 10 – Druhý krok založení projektu – z aplikace „Lodní deník“



Zdroj: vlastní tvorba v Android Studio

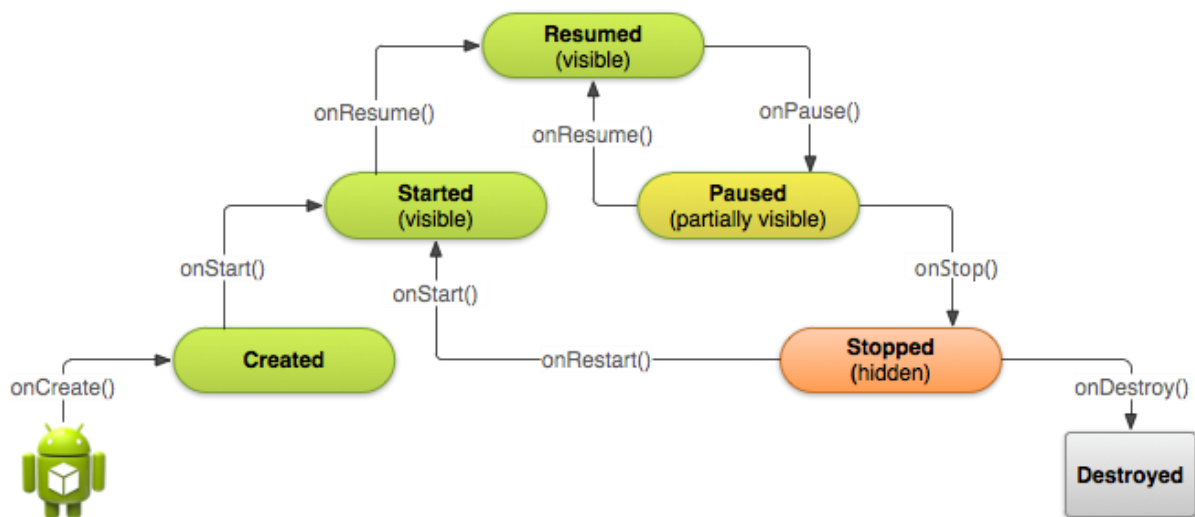
### 3.8 Tvorba aktivity

V celém projektu se dodržuje pravidlo jedné aktivity pro jednu zobrazenou obrazovku, což zajišťuje větší přehlednost a ochranu před ukončením aplikace pomocí stisknutí přístrojového tlačítka zpět.

#### 3.8.1 Životní cyklus aktivity

V jedné zapnuté aplikaci může běžet několik aktivit současně, avšak pouze jedna aktivita může být aktivní, ostatní jsou pozastaveny a uloženy do zásobníku, který je řízen mechanismem LIFO (Last In First Out). Pokud tedy uživatel stiskne tlačítko zpět, tak se aktuální aktivita pozastaví a předchozí aktivita se obnoví. Celý životní cyklus aktivity je zobrazen na obrázku č. 11.

Obrázek č. 11 – Životní cyklus aktivity



Zdroj: [4]

#### 3.8.2 První aktivita

Vytvoření první aktivity je možné už v posledním kroku při zakládání projektu. Pro vytvoření aktivity je zapotřebí zvolit si, zda má být aktivita částečně předvyplněná některými prvky či nikoliv. Pokud není dáno jinak, tak je první vytvořená aktivita automaticky nastavena jako spouštěcí neboli první, která se zobrazí po zapnutí aplikace. Často se první aktivita používá k vykreslování uvítací obrazovky. V tomto projektu nese první aktivita název „home“ a rovnou slouží k výpisu všech plaveb. Tento výpis plaveb je proveden pomocí prvku RecyclerView.



Spouštěcí aktivita je pro tuto aplikaci velmi důležitá. Při vytvoření této aktivity se musí zkontrolovat existence aplikační databáze. Pokud není nalezena, tak se musí automaticky vytvořit a otevřít. V tomto případě je použita vlastní třída „DatabaseHelper“.

### 3.9 Objekty

Veškerá uživatelská data jsou aplikací uložena do databáze. Po načtení dat z databáze je důležité s těmito informacemi vhodně pracovat. Právě proto se v aplikaci všechna načtená data načítají do jednotlivých objektů, u kterých můžeme využívat jejich metody. Ukázka třídy objektu „Person“ je zobrazena na obrázku č. 12, kde je také například vidět metoda „getFirstname“, která vrací jméno člena posádky. Jednotlivé objekty jsou mezi sebou spojeny pomocí identifikačních čísel podobně jako relace u databáze. Seznam objektů v aplikaci je následující:

- **Logbook** – objekt, který obsahuje základní informace o jednom lodním deníku,
- **Logs** – objekt, který obsahuje informace o jednom záznamu ke konkrétnímu lodnímu deníku,
- **Person** – tento objekt obsahuje veškeré informace o členovi posádky,
- **Boats** – obsahuje základní a technické informace o konkrétní lodi.

Obrázek č. 12 - Ukázka třídy objektu Person z aplikace „Lodní deník“

```
public class Person {
    private int id;
    private String firstname;
    private String lastname;
    private String function;

    public Person(int id, String firstname, String lastname, String function){
        this.id = id;
        this.firstname = firstname;
        this.lastname = lastname;
        this.function = function;
    }

    public String getFirstname(){return firstname;}
    public String getLastname(){return lastname;}
    public String getFunction(){return function;}
}
```

Zdroj: vlastní tvorba v Android Studio

### 3.10 Texty s využitím String-array

Většina textů z aplikace je uložena v složce „res/values/“, v souboru „strings.xml“. Ukládání textu do souborů XML má velkou výhodu při využívání více cizích jazyků. Pokud by aplikace byla přeložena do dalšího cizího jazyka, stačilo by pouze přepsat texty v souboru „strings.xml“ a aplikace by rázem byla přeložena.

Nevýhodou tohoto způsobu psaní textu je, že pro každé umístění textu se musí pomocí funkce „findById“ najít proměnná s daným textem, což zabere více času, než když se pouze napíše text k danému prvku.

#### String-Array

Jedná se o pole, do kterého lze zapsat položky s textovými řetězci. V aplikaci *Lodní deník* se toto pole využívá především pro zápis textů, které má uživatel na výběr při rozbalení např. nabídky s počasím.

Každá položka v poli je zapisována pod unikátním indexem, který začíná nulou. Díky tomu se do vnitřní databáze nezapisuje celý text jako např. u zadávání stavu moře „Velmi vysoko vzedmuté“, ale zapíše se pouze index tohoto textu, což je v daném případě „8“. Překlad těchto textů funguje stejně také u načítání dat z databáze, kdy se z databáze stáhne pouze číslo indexu a poté podle něho se přiřadí daný text z prvku „String-Array“.

Obrázek č. 13 - Ukázka String-Array z aplikace „Lodní deník“

```
<string-array name="crew_function">
  <item>Kapitán</item>
  <item>1. Palubní důstojník</item>
  <item>1. Strojní důstojník</item>
  <item>2. Strojní důstojník</item>
  <item>Strojní důstojník strážní služby</item>
  <item>Radiodůstojník</item>
  <item>Elektrodůstojník</item>
  <item>Radio-asistent</item>
  <item>Elektrotechnik</item>
  <item>Lodník</item>
  <item>Strojní pomocník</item>
  <item>Lodní kuchař</item>
  <item>Lodní steward</item>
</string-array>
```

Zdroj: vlastní tvorba v Android Studio

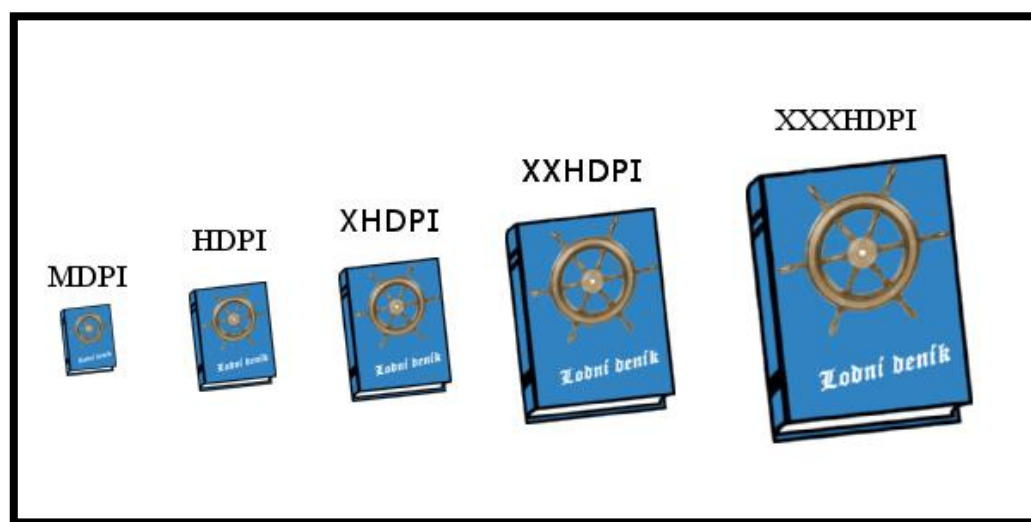
### 3.11 Ikona

Aplikace pro Android nemají pouze jednu spouštěcí ikonu, ale mají jich celkem pět, a to pro každé rozlišení. Ikona pro aplikaci *Lodní deník* byla vytvořena v grafickém programu GIMP a poté upravena do všech povinných rozměrů.

Rozměry ikon pro jednotlivá rozlišení: [17]

- MDPI – 48 x 48 px
- HDPI – 72 x 72 px
- XHDPI – 96 x 96 px
- XXHDPI – 144 x 144 px
- XXXHDPI – 192 x 192 px

Obrázek č. 14 - Ukázka ikon z aplikace "Lodní deník"



Zdroj: vlastní tvorba

### 3.12 Budoucí vývoj aplikace

Aplikace *Lodní deník* by měla být v blízké budoucnosti ještě vylepšována. Momentálně aplikace dokáže nahradit pouze informace lodního deníku. Rád bych přidal do aplikace více interaktivních funkcí jako např. widget na plochu, automatické zaznamenávání trasy či zobrazování celé plavby na mapách Google.

Spolu s těmito funkcemi bych rád vytvořil také webovou podporu se synchronizací, aby každý záznam byl ihned uložen do internetové MySQL databáze a každý zaregistrovaný uživatel by si mohl spravovat své plavby nejen pomocí svého mobilního telefonu, ale také pomocí jakéhokoli zařízení s webovým prohlížečem. Aplikaci *Lodní deník* bych rád publikoval v obchodě Google Play až se všemi funkcemi, aby potenciální uživatele neodradila první verze aplikace.

## 4 Publikace aplikace

Tato kapitola se zabývá možnostmi, jakými je možné publikovat vlastní aplikaci ostatním uživatelům.

### 4.1 Možnosti distribuce

Pokud je aplikace hotová a otestovaná, je důležité se rozhodnout, jakou cestou bude aplikace k dispozici ostatním uživatelům.

- **Svépomocí:** soubor apk může být volně distribuován například na vlastním webovém stránce. Toto je vhodné pro aplikace, které jsou určeny jen pro pár uživatelů, kteří aplikaci chtějí. Sám o sobě by tento způsob byl efektivní u webových stránek s velkou návštěvností, kde by nebyl problém se získáváním nových uživatelů. Avšak největším problémem této distribuce je ten, že samostatný soubor apk můžeme do telefonu nainstalovat, pouze pokud máme v telefonu povolené instalace z neznámých zdrojů a již máme v telefonu aplikaci pro prohlížení souborů.
- **Pomocí obchodu:** je nejčastější způsob distribuce aplikací. V dnešní době existuje několik obchodů s aplikacemi pro OS Android. Největším a neznámějším obchodem je však Google Play.

#### 4.1.1 Google Play

Je univerzální internetový obchod (dříve znám pod názvem Google Market) vyvinutý společností Google Inc. Jedná se v podstatě o trh, do kterého vstupují uživatelé Android v roli kupujících a vývojáři aplikací v roli prodávajících.

Google Play nabízí mimo aplikací a her také širokou nabídku filmů, písniček a elektronických knížek. Veškerý obsah umístěný v obchodě Google Play musí projít schvalovacím řízením a díky tomu dochází k ochraně uživatele před nežádoucími či škodlivými kódy, které by mohly poškodit uživatele či jeho zařízení.

#### 4.1.2 QR kód („Quick Response“)

Jedná se o typ čárového kódu od japonské společnosti Denso-Wave, který je uznán jako ISO standard a jeho užívání je bezplatné. QR kód umožňuje uložit až 4296 znaků. Dříve byl používán v automobilovém průmyslu na označování součástek. V dnešní době se QR kódy používají především v odvětví chytrých zařízení, ukládají se do něho například vizitky,

informace o poloze či internetové odkazy. V případě, že je pro aplikaci vytvořena také webová stránka, tak vložení QR kódu s odkazem na Google Play (popřípadě Apple Appstore, Windows Store) je jedna z možností, jak upoutat návštěvníkovu pozornost. [15]

Obrázek č. 15 – Ukázka QR kódu



Zdroj: mobilní telefon Xiaomi

## 4.2 Vývojářský účet

Pro možnost publikování aplikací v obchodě Google Play je zapotřebí, aby vývojář vlastnil svůj uživatelský účet od Google. Vytvoření tohoto účtu je velmi jednoduché a lze jej provést na webu <https://accounts.google.com/>.

Po vytvoření účtu je nutné, aby uživatel propojil svůj Google účet s vývojářskými funkcemi, avšak tento krok je zpoplatněn částkou 25 \$ na jeden účet. Poté se uživatel může přihlásit do Google developer console.

## 4.3 Google developer console

Google developer console je služba, která zajišťuje vývojáři veškeré nástroje pro práci s jeho aplikací, jako například přidávání nových aplikací, statistika jednotlivých aplikací, propojení projektu s ostatními lidmi apod.

### 4.3.1 Přidání a schvalování aplikace

Přidání nové aplikace na publikování je rozděleno do čtyř povinných kroků, které je potřeba vyplnit:

- **APK:** je prvním krokem, který vyžaduje nahrání souboru apk. Po nahrání dojde k automatické kontrole verzí aplikace. Pokud je verze nového souboru apk

stejná nebo menší než aktuální nahraná verze, dojde k chybě a je potřeba vytvořit nový soubor apk.

- **Záznam v obchodu:** v tomto kroku je potřeba vyplnit veškeré informace o aplikaci jako je název, krátký a dlouhý popis apod. Tyto informace jsou později zobrazovány uživatelům v obchodě Google Play. Mimo textové podklady je potřeba zde nahrát screenshoty<sup>7</sup> z aplikace, ikonu aplikace a grafické záhlaví, které je vidět v obchodě Google Play.
- **Hodnocení obsahu:** je systém hodnocení her a aplikací, který má za úkol přinášet uživatelům relevantní hodnocení. Systém zahrnuje oficiální hodnocení od IARC<sup>8</sup>. Zde je zapotřebí, aby vývojář vyplnil dotazník, který je následně vyhodnocen a k aplikaci se automaticky připojí informace o výsledku vyhodnocení, které obsahuje například informace pro rodiče, zda je aplikace vhodná pro jejich potomka.
- **Cena a distribuce:** je posledním povinným krokem, ve kterém je potřeba, aby vývojář vyplnil informace o ceně a zemích, ve kterých bude aplikace publikována. Vývojář, který svou aplikaci chce nabízet za peníze, si musí nejprve vytvořit účet obchodníka, který mu umožňuje dostávat peníze za každou staženou aplikaci.

Po vyplnění všech kroků je k dispozici možnost odeslání aplikace k publikaci. Tento krok trvá cca 20 hodin a dochází během něho ke kontrole aplikace zaměstnanci Google. Pokud neshledají žádný problém, který by porušoval smluvní podmínky, tak je aplikace automaticky publikována v obchodě Google Play všem uživatelům, kteří spadají do podmínek, které vývojář vyplnil v kroku „Cena a distribuce“.

Pokud je během kontroly nalezena chyba, tak je aplikace dočasně pozastavena a Google poté může stáhnout celou aplikaci, i když jde pouze o nahrání aktualizace pro aplikaci. Vývojář poté musí opravit chyby, popřípadě doložit veškeré dokumenty týkající se autorských práv použitého obsahu a vyplnit žádost o znovu překontrolování aplikace. Při nedodržení podmínek může Google zablokovat celý účet vývojáře.

---

<sup>7</sup> Screenshot – zachycení a uložení obrazu ze zařízení.

<sup>8</sup> IARC (International Agency for Research on Cancer) – Mezinárodní agentura, která se specializuje na rakovinu pro světovou zdravotnickou organizaci.

### 4.3.2 Statistické informace

Při úspěšné publikaci aplikace má vývojář k dispozici statistické informace, ve kterých je zobrazen počet stažení s ohledem na zařízení apod., ale také informace o pádech, recenzích a finanční bilanci aplikace.

Obrázek č. 16 – Ukázka webového prostředí Google developer console na aplikaci „Lodní deník“

The screenshot shows the Google Developer Console interface for the application 'Lodní deník'. The interface is in Czech and includes a sidebar with navigation options like 'APK', 'Záznam v obchodu', 'Hodnocení obsahu', 'Cena a distribuce', 'Produkty v aplikacích', 'Služby a rozhraní API', and 'Tipy k optimalizaci'. The main content area is titled 'ZÁZNAM V OBCHODU' and 'PODROBNOSTI PRODUKTU'. It features a language selector set to 'Čeština - cs-CZ' and a 'Spravovat překlady' button. There are three input fields for product information: 'Název\*' (filled with 'Lodní deník', 11 z 30 znaků), 'Krátký popis\*' (filled with 'Lodní deník v chytrém zařízení', 30 z 80 znaků), and 'Úplný popis\*' (filled with a detailed description of the application). A note at the top right states 'Pole označená \* musí být před zveřejněním vyplněna.' Buttons for 'Uložit koncept' and 'Publikovat aplikaci' are visible in the top right corner.

Zdroj: Autorův účet na Google developer console[23].

## Závěr

Cílem této práce bylo charakterizovat základní vlastnosti operačního systému Android, vysvětlit jeho fungování a strukturu. OS Android byl charakterizován tak, aby každý byl schopen porozumět základům o vývoji aplikací pro tento operační systém, včetně souvislostí o historickém vývoji tohoto nejrozšířenějšího operačního systému, který má velký potenciál se i nadále rozvíjet.

Dalším cílem souvisejícím s OS Android bylo navrhnout a publikovat aplikaci „Lodní deník“ právě pro tento operační systém. Aplikace by měla být schopna nahradit klasický papírový lodní deník pro vnitrozemské i námořní plavby. Aplikace splňuje veškeré základní požadavky a náležitosti, které lodní deník musí splňovat a je tedy plně využitelná pro veškeré vodní plavby. Aplikace je plně kompatibilní se všemi zařízeními s OS Android 4.0.3 a novějšími.

Aplikace „Lodní deník“ by mohla být rozvíjena o další doplňkové funkce pro jednodušší a praktičtější používání aplikace, a interaktivní prvky, mezi které bude patřit např. grafické zaznamenávání trasy na mapě. Aplikace bude dostupná na Google Play, odkud si ji bude moci stáhnout každý uživatel využívající OS Android. Uživatelé na Google Play mohou také aplikaci hodnotit, což bude sloužit jako reálná zpětná vazba, ke které se při dalším vývoji aplikace bude určitě přihlížet.

Aktualizace a zlepšování jsou pro mobilní aplikace důležité k udržení a získávání nových uživatelů, jelikož vývoj mobilních aplikací je stále rychlejší. Už v dnešní době se u úspěšných aplikací programují další podpory, např. pro chytré televize či chytré hodinky s OS Android i iOS. Právě proto je do budoucna velký potenciál v programování a dalším vývoji těchto aplikací.



## Referenční seznam

- [1] Strategy Analytics: Android Shipped 1 Billion Smartphones Worldwide in 2014 -- BOSTON, Jan. 29, 2015 /PRNewswire/. [online]. [cit. 2015-11-08]. Dostupné z: <http://www.prnewswire.com/news-releases/strategy-analytics-android-shipped-1-billion-smartphones-worldwide-in-2014-300027707.html>.
- [2] Application Definition from PC Magazine Encyclopedia. [online]. [cit. 2015-11-08]. Dostupné z: <http://www.pcmag.com/encyclopedia/term/37892/application>
- [3] Krátké ohlédnutí za historií Androidu – svět Androida. [online]. [cit. 2015-11-08]. Dostupné z: <http://www.svetandroida.cz/kratke-ohlednuti-za-historii-androidu-201305>.
- [4] Starting an Activity – Android Developers [online]. [cit. 2016-02-20]. Dostupné z: <http://developer.android.com/training/basics/activity-lifecycle/starting.html>
- [5] What is integrated development environment (IDE)? - Definition from WhatIs.com. SearchSoftwareQuality [online]. 2007 [cit. 2016-03-14]. Dostupné z: <http://searchsoftwarequality.techtarget.com/definition/integrated-development-environment>
- [6] Fragments – Android Developers [online]. [cit. 2016-03-14]. Dostupné z: <http://developer.android.com/guide/components/fragments.html>
- [7] Overview – NetBeans IDE [online]. [cit. 2016-03-14]. Dostupné z: <https://netbeans.org/features/index.html>
- [8] Eclipse desktop & web IDE [online]. [cit. 2016-03-14]. Dostupné z: <https://eclipse.org/ide/>
- [9] PROFESSOR F. CLARK, John. History of Mobile Applications [online]. University of Kentucky [cit. 2016-03-14]. Dostupné z: <http://www.uky.edu/~jclark/mas490apps/History%20of%20Mobile%20Apps.pdf>
- [10] LACKO, Ľuboslav. *Vývoj aplikací pro Android*. 1. vyd. Brno: Computer Press, 2015. ISBN 978-80-251-4347-6.
- [11] ALLEN, Grant. *Android 4: průvodce programováním mobilních aplikací*. 1. vyd. Brno: Computer Press, 2013. ISBN 978-80-251-3782-6.
- [12] ANNUZZI, Joseph, Lauren DARCEY a Shane CONDER. *Advanced Android Application Development*. Addison-Wesley Professional, 2014. ISBN 01-338-9238-7.

- [13] PassMark – Android Devices & Smartphones Marketshare in the last 30 Days [online]. [cit. 2016-03-15]. Dostupné z: <http://www.androidbenchmark.net/30dayshare.html>
- [14] ABZ.cz - Renderování [online]. [cit. 2016-03-18]. Dostupné z: <http://slovník-cizich-slov.abz.cz/web.php/slovo/renderovani>
- [15] Proxima studio – 2D kódy [online]. [cit. 2016-04-1]. Dostupné z: <http://www.proxima.net/Digitalni-tisk/2D-carove-kody/>
- [16] Datatypes In SQLite Version 3[online]. [cit. 2016-04-2]. Dostupné z: <https://www.sqlite.org/datatype3.html>
- [17] Drawable – Android Developers [online]. [cit. 2016-04-5]. Dostupné z: <http://developer.android.com/reference/android/graphics/drawable/Drawable.html>
- [18] <Uses-sdk> - Android Developers [online]. [cit. 2016-04-05]. Dostupné z: <http://developer.android.com/guide/topics/manifest/uses-sdk-element.html>
- [19] Web o operačním systému Android [online]. [cit. 2016-04-06]. Dostupné z: <http://home.zcu.cz/~hodlova/>
- [20] Android Applications Components [online]. [cit. 2016-04-06]. Dostupné z: [http://www.tutorialspoint.com/android/android\\_application\\_components.htm](http://www.tutorialspoint.com/android/android_application_components.htm)
- [21] Zákon o vnitrozemské plavbě – č. 114/1995 Sb. – Aktuální znění. [online]. [cit. 2015-04-06]. Dostupné z: <https://zakonyprolidi.cz/cs/1995-114#cast5>
- [22] Zákon o námořní plavbě – č. 61/2000 Sb. – Aktuální znění. [online]. [cit. 2016-04-06]. Dostupné z: <http://www.zakonyprolidi.cz/cs/2000-61>
- [23] API Library - Google [online]. [cit. 2016-04-08]. Dostupné z: <https://console.developers.google.com/>

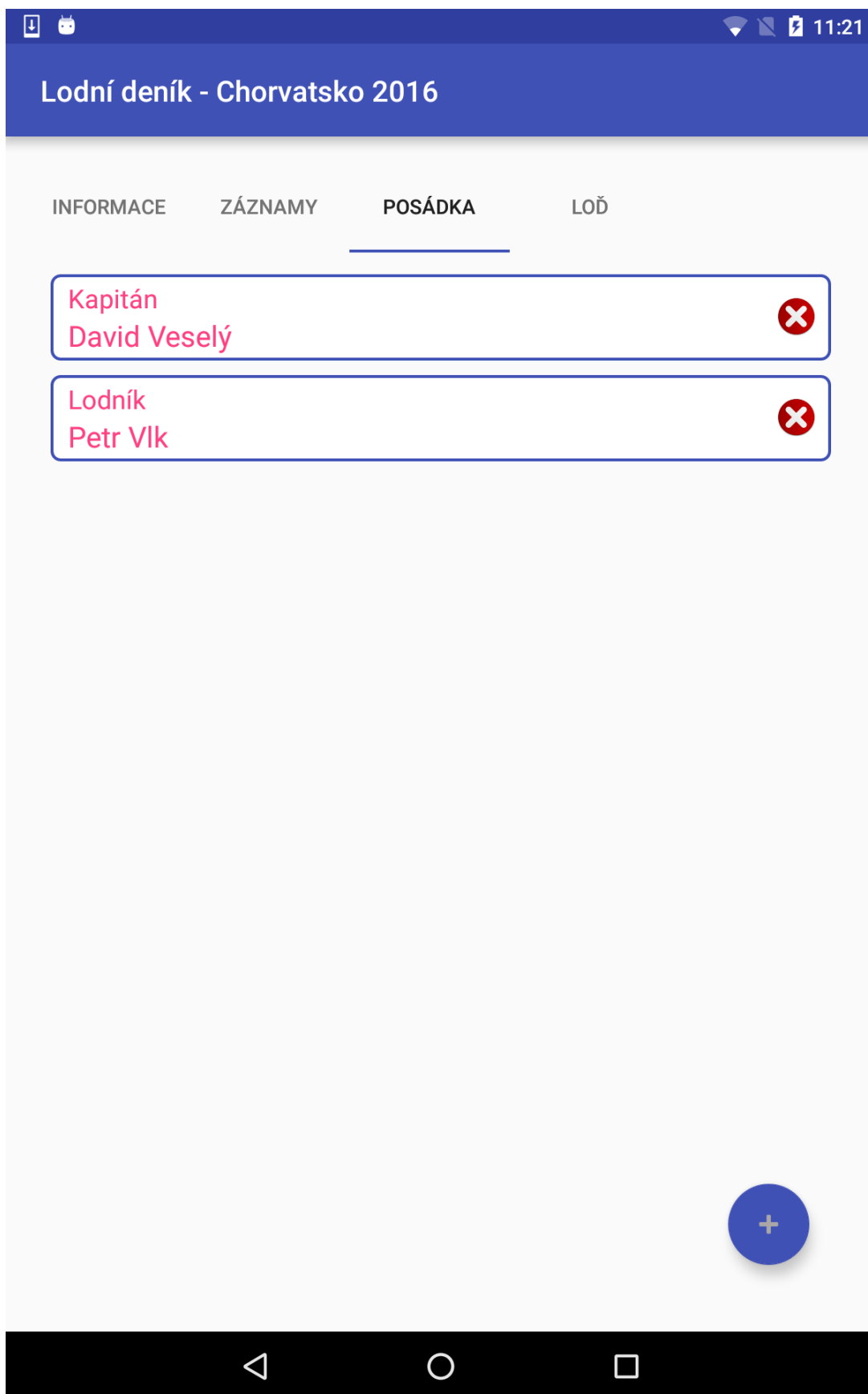
## **Seznam příloh**

Příloha č. 1 – Posádka – z aplikace „Lodní deník“

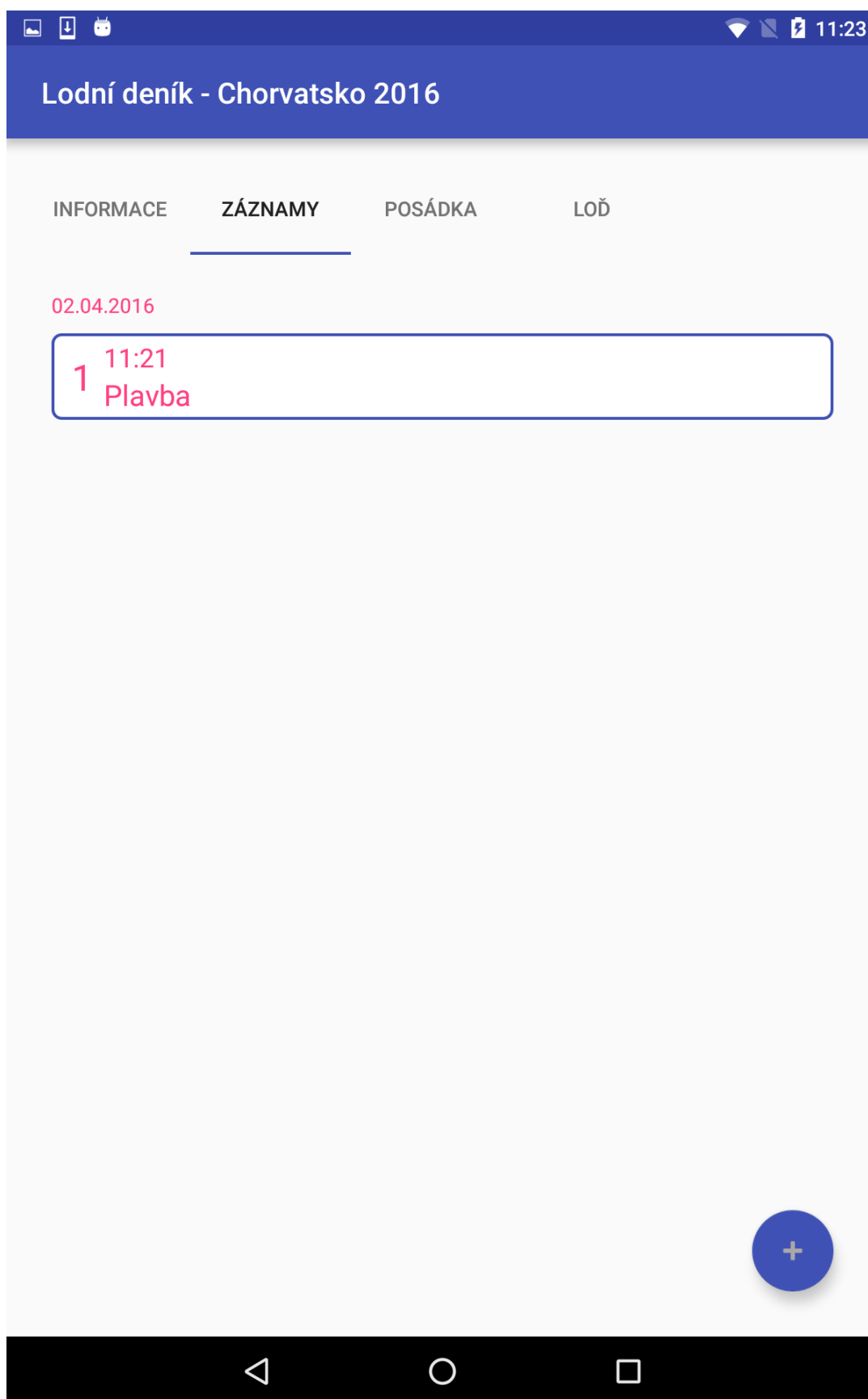
Příloha č. 2 – Záznamy – z aplikace „Lodní deník“

Příloha č. 3 – Vytvoření deníku – z aplikace „Lodní deník“

## Příloha č. 1 – Posádka – z aplikace „Lodní deník“



## Příloha č. 2 – Záznamy – z aplikace „Lodní deník“



### Příloha č. 3 – Vytvoření deníku – z aplikace „Lodní deník“

14:39     72%

## Nový lodní deník



Název plavby

První plavba

Výchozí země

ČESKÁ REPUBLIKA

Typ lodě

Motorová loď



## ANOTACE

<b>Jméno a příjmení:</b>	David Veselý
<b>Katedra:</b>	Katedra technické a informační výchovy
<b>Vedoucí práce:</b>	Mgr. Jan Kubrický, PhD.
<b>Rok obhajoby:</b>	2016

<b>Název práce:</b>	Návrh a tvorba aplikace pro operační systém Android
<b>Název v angličtině:</b>	Design and development of application for the Android operating system
<b>Anotace práce:</b>	Tato bakalářská práce se zabývá historií, charakteristikou a problematikou aplikací v operačním systému Android. Pro praktickou ukázkou programování aplikací v OS Android byla v rámci této práce vypracována mobilní aplikace "Lodní deník", která slouží jako plná náhrada papírového lodního deníku, který musí být povinně veden při každé plavbě.
<b>Klíčová slova:</b>	Android, vývoj aplikací, programování, chytré telefony
<b>Anotace v angličtině:</b>	This bachelor thesis is focused on history, characteristics and application in the Android operating system. For the practical demonstration of application programming in OS Android a mobile application "Lodní deník" was developed, serving as a full replacement of a paper log-book, which is a mandatory part of every voyage.
<b>Klíčová slova v angličtině:</b>	Android, develop application ,programming, smartphones
<b>Přílohy vázané v práci:</b>	Příloha č. 1 – Posádka – z aplikace „Lodní deník“ Příloha č. 2 – Záznamy – z aplikace „Lodní deník“ Příloha č. 3 – Vytvoření deníku – z aplikace „Lodní deník“
<b>Rozsah práce:</b>	39 s.
<b>Jazyk práce:</b>	český