Insert here your thesis' task.

FACULTY
OF INFORMATION
TECHNOLOGY
CTU IN PRAGUE

Bachelor's thesis

StudyPad - Android Client

Roman Levinzon

Department of . . . (SPECIFY)
Supervisor: Ing. Miroslav Balík, Ph.D

December 10, 2018

Acknowledgements

THANKS (remove entirely in case you do not with to thank anyone)

Declaration

I hereby declare that the presented thesis is my own work and that I have cited all sources of information in accordance with the Guideline for adhering to ethical principles when elaborating an academic final thesis.

I acknowledge that my thesis is subject to the rights and obligations stipulated by the Act No. 121/2000 Coll., the Copyright Act, as amended, in particular that the Czech Technical University in Prague has the right to conclude a license agreement on the utilization of this thesis as school work under the provisions of Article 60(1) of the Act.

In Prague on December 10, 2018 .....

Czech Technical University in Prague
Faculty of Information Technology

Citation of this thesis

Levinzon, Roman. *StudyPad - Android Client.* Bachelor's thesis. Czech Technical University in Prague, Faculty of Information Technology, 2018.

Abstrakt

StudyPad je kombinace služby pro porižovaní poznámek a socíální sítě s cílem pomoci studentum zapamatovat si ruzné informace. Cílem práce je vyvinout aplikaci pro OS Android, která bude sloužit jako klient. Tento text uznává stávající řešení, obsahuje analýzu domén a požadavku, popis a výběr architektury aplikace a její implementace

Klíčová slova Android, Kotlin, MVVM

Abstract

StudyPad is a combination of note-taking service and a social network, aimed to help students to memorise different pieces of information. The goal of this thesis is to develop an application for Android OS which will serve as client. This text acknowledges existing solutions, contains domain and requirements analysis, description and choise of application's architecture and it's implementation

Keywords Android, Kotlin, MVVM

Contents

List of Figures

Introduction

Chapter 1

Analysis

1.1 System description

1.2 Existing solutions

1.3 Domain model

1.4 Android Platform

1.5 Requirements

It is important to establish all functional and non-functional requirements for StudyPad. This section contains all requirements designed before the start of the development

1.5.1 Non-functional requirements

- **N1: Native Android application** Application will be written using native Android Sdk
- **N2: API 21 support** Application minimal SDK version will be 21 (a.k.a Android Lollipop)
- **N3: Material Design** Application user interface will follow latest Material design guidelines and best practises
- **N4: Scalable app architecture** Application's architecture must be scalable and easy testable
- **N5: Tablet & Phone support** Application's GUI must be well suited for multiple screen sizes
- **N6: Multiple language support** Application will support multiple languages

1.5.2 Functional requirements

User Authentication

- **F1: Registration/Login using email** Access to StudyPadis possible by creating an account using email address/password combination
- **F2: Registration/Login using Facebook** Users will be able to use their Facebook account to access StudyPad
- **F3: Registration/Login using Google** Users will be able to use their Google account to access StudyPad
- **F4: Store OAuth token** API Authentication Token will be stored in device memory
- **F5: Token refreshment** API Token will be refreshed when needed, so user won't have to login again
- **F6: University selection** As a part of user registration flow, user will be able to select his university

Library Management (Notes & Notebooks)

- **F7: Notebook creation** User will be able to create new notebooks with the name they choose
- **F8: Notebook deletion** User will be able to delete existing notebooks
- **F9: Notebook name edition** User will be able to edit notebooks names
- **F10: Note creation** User will be able to create a note with specific title and content.
- **F11: Note edition** User will be able to edit existing note, or completely delete it
- **F12: Show Notebooks** : User will be able to view all the notebooks he created
- **F13: Show Notes**: By clicking on notebook item, user will be able to view the list of notes that are assigned to this notebooks

Sharing Hub

- **F14: View published notebooks** User will be able to view notebooks published by other users
- **F15: Search through published books** User will be able to search through the published notebooks by applying different filters (such as author, university and subject)
- **F16: Browse through published notebook** User will be able to see notes inside the notebook that's been published
- **F17: View comments** User will be able to view others users comments discussing a notebook that's been published
- **F18: Leave a comment** User can comment on other user published notebook
- **F19: Delete a comment** Application will let user to delete his/her comment.
- **F20: Save published notebook** User will be able to save published notebook to his library
- **F21: Publish notebook** User will be able to publish his notebook

- **F22: Update published notebook** Author of the published notebook will be able to update it's information
- **F23: Delete published notebook** Author of the published notebook will be able to delete the his notebook
- **F24: Share notebook** User will be able to share his notebook by generating a deeplink

Study Hub

- **F25: Start a basic self-check** User will be able to use an interactive way to look through his notes
- **F26: Start a written test** User will be able to participate in a written test based on one of his notebooks to test his knowledge
- **F27: Start a quiz** User will be able to participate in quiz challenge that will be based on one of his notebooks

Settings

- **F28: View Profile Information** User will be able to view his profile information such as first name, last name and his university
- **F29: Edit Profile Information** User will be able to edit his profile information
- **F30: Logout** User will be able to logout from the application.

1.6 Platform-Independent Model

Chapter 2

Design

Chapter 3

Implementation

3.1 Choice of technologies

3.2 Component diagram

3.3 Installation

Chapter 4

Testing

Conclusion

Appendix A

Acronyms

GUI Graphical user interface

XML Extensible markup language

Appendix B

Contents of enclosed CD

- `readme.txt` ..... the file with CD contents description
- `exe` ..... the directory with executables
- `src` ..... the directory of source codes
  - `wbdcm` ..... implementation sources
  - `thesis` ..... the directory of LaTeX source codes of the thesis
- `text` ..... the thesis text directory
  - `thesis.pdf` ..... the thesis text in PDF format
  - `thesis.ps` ..... the thesis text in PS format