

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
“КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ”
ІПСА

Кафедра Системного проектування

Лабораторна робота 2

з дисципліни: «ПРОЕКТУВАННЯ ІНФОРМАЦІЙНИХ СИСТЕМ»

**Скласти опис передпроектної документації (Software
Requirement Specifications).**

Виконав:

студент групи ДА-71

Павлюк Вадим

Revision History

Date	Description	Author	Comments
20.11.2020	Version 0.1	Pavliuk	Create template
25.11.2020	Version 1.0	Pavliuk	Ready

Document Approval

The following Software Requirements Specification has been accepted and approved by the following:

Signature	Printed Name	Title	Date
	Pavliuk		20.11.2020

Table of Contents

REVISION HISTORY	2
DOCUMENT APPROVAL	2
1. INTRODUCTION	4
1.1 PURPOSE	4
1.2 SCOPE	4
1.3 GLOSSARY	4
1.4 REFERENCES	4
1.5 OVERVIEW	5
2. OVERALL DESCRIPTION	6
2.1 PRODUCT PERSPECTIVE	6
2.2 PRODUCT FUNCTIONS	6
2.3 USER CHARACTERISTICS	6
2.4 GENERAL CONSTRAINTS	6
3. SPECIFIC REQUIREMENTS	7
3.1 EXTERNAL INTERFACE REQUIREMENTS	7
<i>3.1.1 User Interfaces</i>	7
3.2 FUNCTIONAL REQUIREMENTS	12
3.3 USE CASE	12
3.4 NON-FUNCTIONAL REQUIREMENTS	13
<i>3.3.1 Performance</i>	13
<i>3.4.2 Availability</i>	13
<i>3.4.3 Security</i>	13
<i>3.4.4 Maintainability</i>	13
<i>3.4.5 Portability</i>	13
3.5 LOGICAL DATABASE REQUIREMENTS	14
4. TESTS	14
4.1 TEST 1	14
<i>4.1.1 Scenario 1</i>	14
<i>4.1.2 Scenario 2</i>	14
4.2 TEST 2	14
<i>4.2.1 Scenario 1</i>	14
4.3 TEST 3	15
<i>4.3.1 Scenario 1</i>	15
<i>4.3.2 Scenario 2</i>	15
<i>4.3.2 Scenario 3</i>	15
4.4 TEST 4	15
<i>4.4.1 Scenario 1</i>	15
<i>4.4.2 Scenario 2</i>	15

1. Introduction

1.1 Purpose

The purpose of this document is to present a detailed description of the PasswordKeeper mobile application. It will explain the purpose, features of this system, and user interface, what the system will do, constraints under which it must operate and how system will react. This document is intended for both stakeholders and developers.

1.2 Scope

This system is intend to be a mobile app for everybody who wants to keep their logins and passwords in the phone. The software products name is “PasswordKeeper”.

Whole system is a mobile app which is connected to database.

1.3 Glossary

Term	Definition
<i>App</i>	<i>“PasswordKeeper” aplication</i>
<i>Database</i>	<i>Collection of all the information which is stored in system.</i>
<i>User</i>	<i>person, who was logged into the system.</i>
<i>Item</i>	<i>This is collection of(website name, login, password) which user want to store in database</i>
<i>Software Requirements Specification</i>	<i>A document that completely describes all of the functions of a proposed system and the constraints under which it must operate. For example, this document.</i>

1.4 References

IEEE. *IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications*. IEEE Computer Society, 1998.

“Методичні вказівки до виконання лабораторних робіт для студентів напряму підготовки 6.050101 «Комп'ютерні науки», спеціальностей 8.05010102 «Інформаційні технології

проектування» та 8.05010103 «Системне проектування» денної та заочної форм навчання», Dph Konstantin Kharchenko.

1.5 Overview

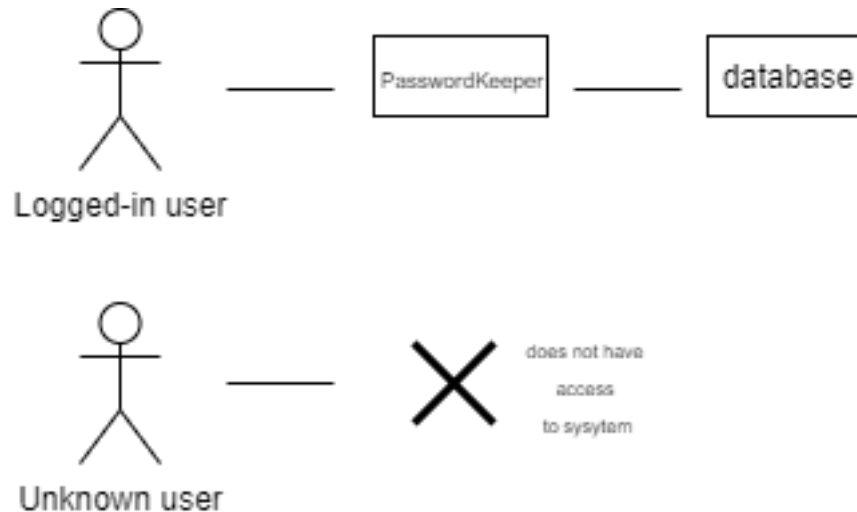
The next chapter, the Overall Description section, of this document gives an overview of the functionality of the product. It describes the informal requirements and is used to establish a context for the technical requirements specification in the next chapter.

The third chapter, Requirements Specification section, of this document is written primarily for the developers and describes in technical terms the details of the functionality of the product.

Both sections of the document describe the same software product in its entirety, but are intended for different audiences and thus use different language.

2. Overall Description

2.1 Product Perspective



The PasswordKeeper has only one type of user – logged-in. He communicates with database via mobile app and changes information which is stored in system. All he needs - is phone connected to the internet.

2.2 Product Functions

Product function is to provide a system for creating, changing and deleting item.

2.3 User Characteristics

Internet literate person, who knows how to use mobile apps and perform basic operations.

2.4 General Constraints

Mobile app will be written in Vue.js it is web frame-work, for database will be used FireBase. There aren't any restrictions in target product platform. All the user needs – are device with access to the internet.

3. Specific Requirements

3.1 External Interface Requirements

3.1.1 User Interfaces

Authorization page

Welcome to PasswordKeeper

Login

Password

Login

Sign Up

Copyright 2020

Here user is able to enter the system via login and password.
Also here is presented sign up option.

Sign up

Registration

Login

Enter login...

Password

Enter password...

Repeat Password

Repeat password...

Sign Up

Cancel


Copyright 2020

Here user is able to register.


Storage page


STORAGE

Create new





Apple
example@icloud.com







Apple
example@icloud.com







Apple
example@icloud.com



Favourites

Storage

Settings

Here user is able to watch all his items .

Storage page(change existed item)

The screenshot shows a mobile application interface for editing a stored item. At the top is a blue header with the word "STORAGE" in white. Below the header is a white bar containing an Apple logo and the text "Apple". The main content area has a light gray background and contains three text input fields: "Name of website" with the value "Apple", "Login" with the value "example@icloud.com", and "Password" with the value "Qwerty123". Below these fields are three buttons: a green "Save" button, a blue "Delete" button, and a red "Cancel" button. At the bottom is a navigation bar with three icons: a star for "Favourites", a shield for "Storage", and a gear for "Settings".

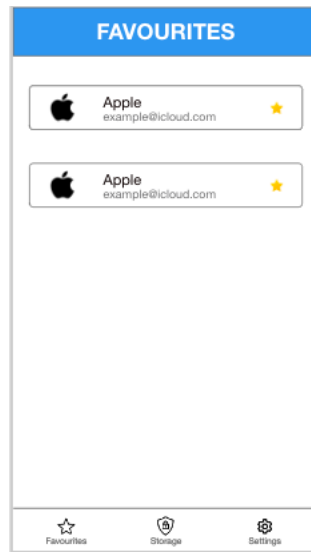
Here user is able to change: name of website, login and password or delete item at all.
Storage page(create new item)

The screenshot shows a mobile application interface for creating a new item. At the top is a blue header with the word "STORAGE" in white. Below the header is a white bar with the text "New item". The main content area has a light gray background and contains three text input fields: "Name of website" with the value "Apple", "Login" with the value "example@icloud.com", and "Password" with the value "Qwerty123". Below these fields are two buttons: a green "Create" button and a red "Cancel" button. At the bottom is a navigation bar with three icons: a star for "Favourites", a shield for "Storage", and a gear for "Settings".

Here user is able to create new item

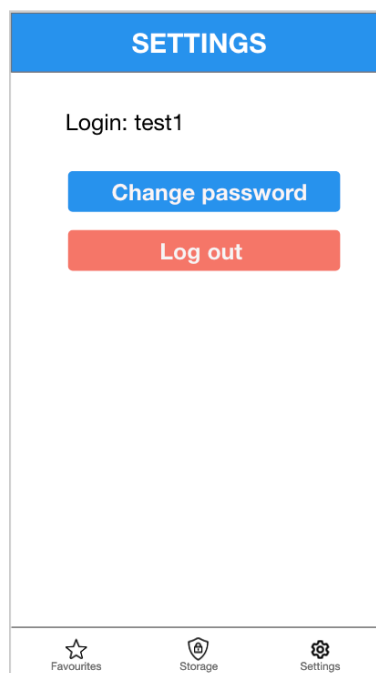
PasswordKeeper

Favourites page



Here user is able to watch all his favourites items or in other words most used.

Settings page



Here user is able to change password of account or log out.

New password page

NEW PASSWORD


Login: test1


Password


Repeat Password

Change

Cancel


Favourites


Storage


Settings

Here user is able to change password

3.2 Functional Requirements

Login: input user login and password.

Logout: button at the setting page.

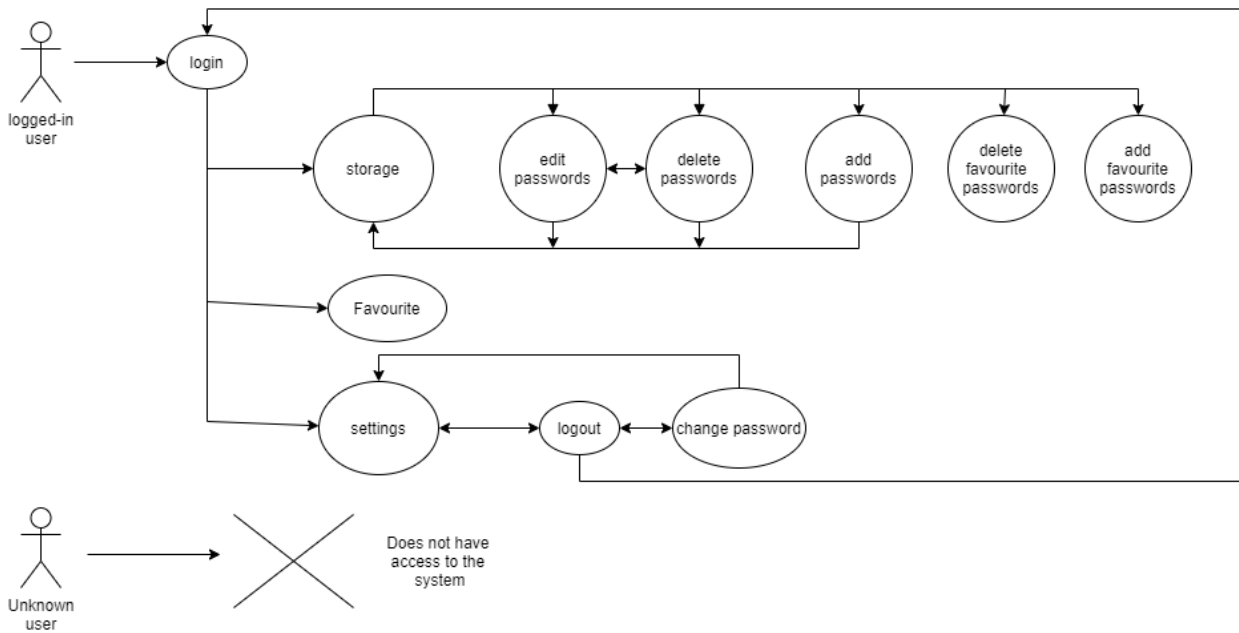
Change user password: form with (new password, repeat password) fields.

Edit item: form with pre-filled data.

Delete item: button.

Create item: form with fields.

3.3 Use Case



3.4 Non-Functional Requirements

3.3.1 Performance

Speed of system execution depends mostly on Internet connection.

3.4.2 Availability

This app is available for everyone.

3.4.3 Security

All your personal information is secured and encrypted.

3.4.4 Maintainability

System will have regular updates that fix bugs and performance improvements. Besides that could be added new features.

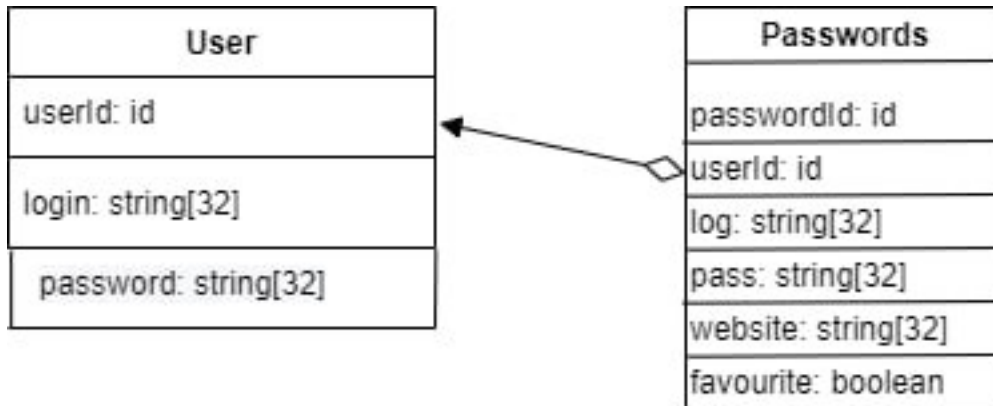
3.4.5 Portability

System targets wide range of devices:

- Android;
- IOS;

3.5 Logical Database Requirements

Logical structure of data to be stored in the Database is given below.



4. Tests

Internet connection is required

4.1 Test 1

- Open mobile app
- Login form should appear

4.1.1 Scenario 1

- Type login – admin, password – 1234
- Errors should appear “access denied”

4.1.2 Scenario 2

- Type login –admin2, password – 1234
- Storage page should be opened

4.2 Test 2

- Click “SIGN UP” button on the login page

4.2.1 Scenario 1

- Type exampleLogin, examplePass, examplePass
- Should appear success

4.3 Test 3

- Open «Storage» page

4.3.1 Scenario 1

- Click “Create new” button
- Should appear form with fields: website name, login, password

4.3.2 Scenario 2

- Click “item”
- Should appear form with pre-field data: website name, login, password

4.3.2 Scenario 3

- Click “star”
- Star should change color from black to yellow and be visible on Favourites page

4.4 Test 4

- Open “Settings” page

4.4.1 Scenario 1

- Click “Change password” button
- Should appear form with fields: password, repeat password

4.4.2 Scenario 2

- Click “Log out ” button
- Should appear starting page