Software Requirements Specification

for

Accountant Calendar

Version 1.0 approved

Prepared by Gennadiy Furduy

FMI, Plovdiv University

01 May 2019

Contents

[1. Introduction 1](#_Toc10552044)

[1.1 Purpose 1](#_Toc10552045)

[1.2 Document Conventions 1](#_Toc10552046)

[1.3 Intended Audience and Reading Suggestions 1](#_Toc10552047)

[1.4 Product Scope 1](#_Toc10552048)

[1.5 References 1](#_Toc10552049)

[2. Overall Description 2](#_Toc10552050)

[2.1 Context diagram 2](#_Toc10552051)

[2.2 ER Diagram 3](#_Toc10552052)

[2.3 Class diagram 4](#_Toc10552053)

[2.4 Design and Implementation Constraints 5](#_Toc10552054)

[2.5 Assumptions and Dependencies 5](#_Toc10552055)

[3. External Interface Requirements 6](#_Toc10552056)

[3.1 User Interfaces 6](#_Toc10552057)

[3.2 Hardware Interfaces 9](#_Toc10552058)

[3.3 Software Interfaces 9](#_Toc10552059)

[3.4 Communications Interfaces 10](#_Toc10552060)

[4. System Features 10](#_Toc10552061)

[4.1 Registration UC 1.1 10](#_Toc10552062)

[4.2 Login UC 1.2 11](#_Toc10552063)

[4.3 Choose Taxes and reports UC 1.3 12](#_Toc10552064)

[4.4 Buy Premium UC 1.4 13](#_Toc10552065)

[5. Acceptance Criteria 16](#_Toc10552066)

[5.1 Registration UC 1.1 16](#_Toc10552067)

[5.2 Login UC 1.2 16](#_Toc10552068)

[5.3 Choose Taxes and reports UC 1.3 16](#_Toc10552069)

[5.4 Buy Premium UC 1.4 16](#_Toc10552070)

[6. Other Nonfunctional Requirements 16](#_Toc10552071)

[6.1 Performance Requirements 16](#_Toc10552072)

[6.2 Safety Requirements 17](#_Toc10552073)

[6.3 Security Requirements 17](#_Toc10552074)

[6.4 Software Quality Requirements 17](#_Toc10552075)

[6.4.1 QR1: Quality of service 17](#_Toc10552076)

[7. Price and development time 17](#_Toc10552077)

[8. Appendix A. Figures 19](#_Toc10552078)

Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
|  |  |  |  |
|  |  |  |  |

# Introduction

## Purpose

The purpose of this document is to build an online system to help to accountants of different organization remember the deadlines and the terms and conditions of reports and taxes.

## Document Conventions

This document uses the following conventions.

|  |  |
| --- | --- |
| DB | Database |
| ERD | Entity Relationship Diagram |
| AC | Accountant Calendar |

## Intended Audience and Reading Suggestions

This document is for university teachers, students and mentors. It was made within the subject *“*Практикум по софтуерни технологии*”*

## Product Scope

The purpose of the AC is to help accountants and to create professional calendar with notification. AC based on a cloud-based database and services and works on mobile device with Android and iOS operation systems.

## References

<https://firebase.google.com/docs>

<http://sfs.gov.ua/pk/>

# Overall Description

## Context diagram

A close up of a map

Description automatically generated

Figure Context diagram

**User**: uses calendar to obtain information, pays for Premium

**Premium User**: uses the calendar to obtain information, can choose particular entities to observe

**Payment Service**: is responsible for billing

**SFS API**: is responsible for giving actual information

**Firebase Auth**: is responsible for User authentication

## ER Diagram

A picture containing text

Description automatically generated

Figure . ERD

Tax: taxes and reports

* Id – unique ID
* Title – name of tax / report in system
* Description – small description of tax / report
* Link – http link to particular document on rada.gov.ua
* Deadline – end date of pay tax / give report
* Company type – Id of CompanyType

CompanyType:

* Id – unique ID
* Title – name of company type

Company: user’s companies

* Id – unique ID
* Name – name of company
* Company type – Id of CompanyType

SelectedTax: taxes selected for show and notification by Premium User

* CompanyId – Id of Company
* TaxId – Id of Tax

## Class diagram

A screenshot of text

Description automatically generated

Tax:

* Name: String
* Description: String
* Link: String
* Deadline: Date

Company:

* Name: String
* Type: Int

User:

* Name: String
* IsPremium: Boolean
* Companies: List<Company>
* SelectedTaxes: List<Tax>

Workers:

* showTaxesForCompanyType – show taxes by Company Type in ViewInformation Screen
* showSelectedTaxes – show taxes selected by premium user in view information screen
* sendNotification – send notification to users

## Design and Implementation Constraints

AC is developed in JavaScript with React Native. It uses Firebase cloud services for authentication. It uses a modular design where every feature is wrapped into separate module and the modules depend on each other through well-written API.

## Assumptions and Dependencies

AC is developed for mobile device operated by Android version 6 or higher and iOS version 11 and higher. AC needs Internet access.

# External Interface Requirements

## User Interfaces

A screenshot of a cell phone

Description automatically generated

Figure . UI. Login Screen

A screenshot of a cell phone

Description automatically generated

Figure . UI. Register Screen

A screenshot of a cell phone

Description automatically generated

Figure . UI. View Information Screen

A screenshot of a cell phone

Description automatically generated

Figure . UI. Select taxes Screen

## Hardware Interfaces

AC support device from Apple (iOS) and various other manufacturers (Android).

## Software Interfaces

1. Firebase Auth.
2. Payment Service

## Communications Interfaces

AC requires an internet connection for update data, receive notification and make payments.

# System Features

A close up of a map

Description automatically generated

Figure . Use Case Diagram

## Registration UC 1.1

Before use the AC User must register in AC.

User should register by enter E-mail and Password and press Register button.

* If E-mail does not exist in AC and Password matched security requirements, AC navigate User to View Information screen.
* If E-mail exists in AC, AC show message “You already registered, please Login” and navigate User to Login screen.
* If E-mail does not exist in AC and Password don’t matched security requirements, AC show message “Your password is weak. Please choose stronger”.

## Login UC 1.2

|  |  |  |
| --- | --- | --- |
| Number | Section | Content |
| 1 | ID | UC 1.2 |
| 2 | Title | Login |
| 3 | Author | Gennadiy Furduy |
| 4 | Priority | High |
| 5 | Critical | High |
| 6 | Source | Gennadiy Furduy |
| 7 | Responsible | Gennadiy Furduy |
| 8 | Description | User enter E-mail and Password. AC checks credentilas and navigate User to View Information screen |
| 9 | Trigger event | User press Login button |
| 10 | Actors | User, Firebase Auth |
| 11 | Preconditions | AC show Login screen |
| 12 | Postconditions | AC show View Information screen |
| 13 | Target | User see View Information screen |
| 14 | Main success scenario | 1. AC show Email and Password fields and Login button 2. User enter valid Email and Password and press Login button 3. AC make request to Firebase Auth 4. Firebase Auth find user and response with user information 5. AC navigate User to View Information screen |
| 15 | Extensions | 2a. User enter invalid Email or Password and press Login button 3a. AC show message "Please enter valid Email / Password" and go to 1.  4b. Firebase Auth can’t find user and response with error 5b. AC show message "Please enter valid Email / Password or Register" |
| 16 | Exceptions | Trigger event: AC have no internet connection |
| 17 | Quality | QR1: Quality of service |

## Choose Taxes and reports UC 1.3

A close up of a map

Description automatically generated

Figure . Use case 1.3 Action Diagram

## Buy Premium UC 1.4

A close up of a map

Description automatically generated

Figure . Use Case 1.4 Successful diagram

A screenshot of a cell phone

Description automatically generated

Figure . Use Case 1.4 Fail diagram 1

A screenshot of a computer

Description automatically generated

Figure . Use Case 1.4 Fail diagram 2

# Acceptance Criteria

## Registration UC 1.1

1. User enter valid Email and Password and press Register button. Email was not already registered in Firebase Auth. AC navigate User to View Information screen.
2. User enter valid Email and Password and press Register button. Email was already registered in Firebase Auth. AC show message “You already registered, please Login” and navigate User to Login screen.
3. User enter valid Email and Password and press Register button. Email was not already registered in Firebase Auth. Password don’t matched security requirements. AC show message “Your password is weak. Please choose stronger”.

## Login UC 1.2

1. User enter valid Email and Password and press Login button. AC navigate User to View Information screen.
2. User enter invalid Email or Password and press Login button. AC show message "Please enter valid Email / Password"

## Choose Taxes and reports UC 1.3

1. User press button “Select Taxes and Reports” and has Premium status. AC navigate User to Select Taxes and Reports screen. After select taxes and reports User press Save button and AC navigate him to View Information screen. Information screen show taxes and reports selected by User only.
2. User press button “Select Taxes and Reports” and has no Premium status. AC navigate User to Buy Premium screen.

## Buy Premium UC 1.4

1. User press button “Buy Premium” and AC redirect User to Payment Service. After successful payment procedure AC navigate User to View Information screen and can choose taxes. Information about User status in Firebase Auth updated.
2. User press button “Buy Premium” and redirected to Payment Service. After fail payment procedure AC navigate User to View Information screen and User see message about failed payment.

# Other Nonfunctional Requirements

## Performance Requirements

AC requires at least 50Mb of free space on mobile device

## Safety Requirements

To ensure that no one of AC’s users loses any data while using AC (due to a crash or a bug of some kind) the developer team updates AC regularly. There is a bug tracker available where users can report any bugs they have encountered so that the developers can fix it in the next release.

## Security Requirements

AC does not have any security requirements, but Premium users needs authorization with login and password.

## Software Quality Requirements

### QR1: Quality of service

If user have internet connection all screens can show in 1000ms or faster

# Price and development time

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Function Count** | | | | | |
|  |  |  |  |  |  |
| **Item** | **Item Description** | **Complexity** | **Count** | **Weight** | **Weighted** |
| **Count** |
| 1 | Number of User Inputs | Simple | 9 | 3 | 27 |
| Average |  | 4 |  |
| Complex |  | 6 |  |
| 2 | Number of User Outputs | Simple | 3 | 4 | 12 |
| Average |  | 5 |  |
| Complex | 3 | 7 | 21 |
| 3 | Number of User Inquiries | Simple | 2 | 3 | 6 |
| Average |  | 4 |  |
| Complex |  | 6 |  |
| 4 | Number of Files | Simple | 4 | 7 | 28 |
| Average |  | 10 |  |
| Complex |  | 15 |  |
| 5 | Number of External Interfaces | Simple | 2 | 5 | 10 |
| Average |  | 7 |  |
| Complex |  | 10 |  |
| **Total Weighted Function Count (FC)** | |  |  |  | **104** |
|  |  |  |  |  |  |
| **Complexity Factor** | | |  |  |  |
|  |  |  |  |  |  |
| **Factor** | **Description** | **Rating** |  |  |  |
| **0=Irrelevant** |  |  |  |
| **5=Essential** |  |  |  |
| F1 | Reliability and backup recovery | 1 |  |  |  |
| F2 | Data communications | 5 |  |  |  |
| F3 | Distributed processing | 0 |  |  |  |
| F4 | Performance | 3 |  |  |  |
| F5 | Operate on existing system | 2 |  |  |  |
| F6 | On-line data entry | 0 |  |  |  |
| F7 | Data entry over multiple screens | 1 |  |  |  |
| F8 | Master files updated on-line | 2 |  |  |  |
| F9 | Complex inputs, outputs, files & inquiries | 0 |  |  |  |
| F10 | Complex internal processing | 1 |  |  |  |
| F11 | Code needs to be reusable | 2 |  |  |  |
| F12 | Need conversion and installation | 3 |  |  |  |
| F13 | Multiple installations of the system | 0 |  |  |  |
| F14 | Easy to change and use | 3 |  |  |  |
| **Complexity Factor (CF) = sum of ratings** | | **23** |  |  |  |
|  |  |  |  |  |  |
| **Function Points** | | |  |  |  |
|  |  |  |  |  |  |
| **Function Points (FP) = FC x (0.65 + 0.01 x CF)** | | **91.52** |  |  |  |
|  |  |  |  |  |  |
|  | Price per hour: | $50.00 |  |  |  |
|  | Hours/FP: | 3.00 |  |  |  |
|  | Development time [(FP \*Hours/FP) / 8 hours]: | 34 |  |  |  |
|  | AC price [Development time \* 8 hours \* Price per hour]: | $13,600.00 |  |  |  |

# Appendix A. Figures

[Figure 1 Context diagram 2](#_Toc10551511)

[Figure 2. ERD 3](#_Toc10551512)

[Figure 3. UI. Login Screen 6](#_Toc10551513)

[Figure 4. UI. Register Screen 7](#_Toc10551514)

[Figure 5. UI. View Information Screen 8](#_Toc10551515)

[Figure 6. UI. Select taxes Screen 9](#_Toc10551516)

[Figure 7. Use Case Diagram 10](#_Toc10551517)

[Figure 8. Use case 1.3 Action Diagram 12](#_Toc10551518)

[Figure 9. Use Case 1.4 Successful diagram 13](#_Toc10551519)

[Figure 10. Use Case 1.4 Fail diagram 1 14](#_Toc10551520)

[Figure 11. Use Case 1.4 Fail diagram 2 15](#_Toc10551521)