Project Requirements

Project Name: Wordfinder

Team: 4

Dominik Seliner

Tim Kousadianos

Andres Wälchli

Domenico Iapello

Customer: Andrei Chis

Revision History

|  |  |  |
| --- | --- | --- |
| Version | Date | Revision Description |
| 1.0 | 02.10.2013 | Rough sketch |
| 1.1 | 23.10.2013 | Scenario addaption to changes statet in Customer meeting |
| 1.2 | 27.11.2013 | Scenario addaption to changes statet in Customer meeting |
| 1.4 | 08.12.2013 | Deletion of non required content of the SRS |

Ind§ex

1 Introduction 3

1.1 Purpose 3

1.2 Stakeholders 3

1.3 Definitions 3

1.4 Overview of Document 3

2 Overall description 4

2.1 Diagram 4

2.2 Use cases 5

2.2.1 1. Play Game 5

2.2.2 2. View Stats 5

2.2.3 3. Retry Board 6

2.2.4 View Tutorial 6

2.3 Actor characteristics 7

3 Specific Requirements 8

3.1 Functional Requirements 8

3.1.1 Use case 1. Play Game 8

3.1.2 Use case 2. View stats 8

3.1.3 Use case 3. Retry board 8

3.1.4 Use case 4. View tutorial 8

3.2 Nonfunctional Requirements 8

3.2.1 Use case 1. Play Game 8

3.2.2 Use case 2. Show stats 9

3.2.3 Use case 3. Retry board 9

3.3 General 9

# Introduction

## Purpose

The reason this document is written, is for the stakeholders and the developers to define an appropriate definition of the Android application “Wordfinder”. It should declare the application’s purpose and features, including the use cases and requirement specifications.

## Stakeholders

* User
* Client
* Developer

## Definitions

|  |  |
| --- | --- |
| Board / Matrix | Area where the player has to play in, the place where he has to find the words. |
| Dictionary |  |
| Friend | Another player who was added as “Friend” in the social section of the game. A friend allows to play other modi of the game |
| Highscore | Amount of all points, which a player collected in his playtime. |
| Language | Defines the language in which the game is played (language of the words). |
| Player | The person who uses the application. |
| Time, -r | A visual countdown that is used to determine the length of the game. |
| Word | Predefined chain of letters |
| Wordfinder | Name of the Android-application |
|  |  |

## Overview of Document

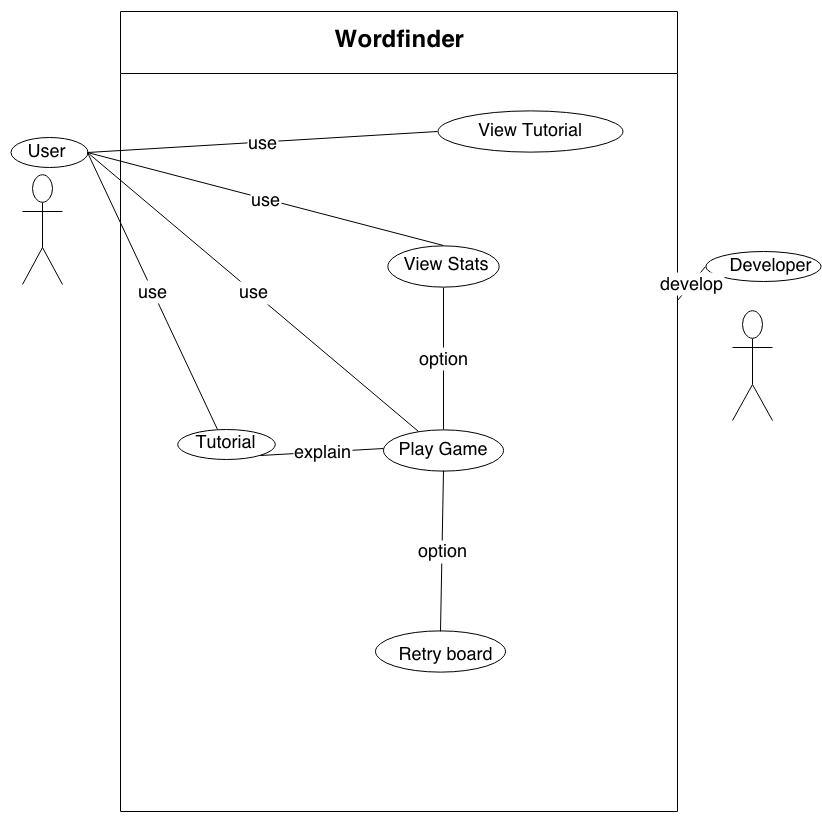
The next chapter, the Overall description, contains a diagram that present an overview and summary of all the use cases. Therefor the Overall description describes the application in the sight of an user, and simulates every action he should be able to perform.

The third chapter is primary written for the developers and describes in technical terms and the details of the functionalities of the application.

# Overall description

## Diagram

The Diagram includes also funktionalities that no longer are in the application. They will not be removed to show a possible overview of the funktionality that the app could provide in future.



## Use cases

### 1. Play Game

Actors

* Player (Bob)

Description

* Bob wants to play a local game

Trigger

* Presses the 'Start Game' button in the main screen

Pre-Conditions

Post-Conditions

1. New entry in personal recordings (Score, Board, ...)

Main scenario

1. Bob starts the 'word finder' app
2. Bob presses on 'start game' in the main screen
3. The System generates a new game
   1. The system tries to use as many words form the selected dictionary as possible[[1]](#footnote-1)
   2. The remaining board is filled up with words from the default dictionary (English)
4. Bob plays and finished the game
   1. The game is finished once Bob has found all the words in the board, or the timer[[2]](#footnote-2) has expired.
5. Score is displayed and saved in the personal recordings
6. returning to the main screen

Alternative scenarios

1. Game is interrupted (e.g. incoming call) or paused manually
2. Pause the game
3. Hide the board, show 'resume' button
4. Wait until 'resume' is pressed
5. resume the game and continue normal execution
6. Game is quit manually

* End game and return to main screen
* No entri in scores

Notes

* local games can be played without an internet connection

### 2. View Stats

Actors

* Player (Bob)

Description

* Bob want's to view his score.

Trigger

* Bob presses ‘Scores’, if he wants to see his scores

Main scenario

1. Bob selects 'Scores' to enter the scores menu
2. Bob is presented with a list of all his played Boards
3. Beside each board the personal best score is listet.

### 3. Retry Board

Actors

* Player (Bob)

Description

* Bob wants to play a game he already played before

Trigger

* Bob selects one of his scored boards.

Post-Conditions

1. New entry in personal recordings (Score, Board, ...)

Main scenario

1. Bob selects to replay a board
2. The board is loaded
3. The board's dictionary is loaded
4. The game is generated. The loaded board is used instead of a newly generated one
5. Bob plays and finished the game
6. Score is displayed and saved in the personal recordings
7. returning to the main screen

Alternative scenarios

1. Game is interrupted (e.g. incoming call) or paused manually
2. Pause the game
3. Hide the board, show 'resume' button
4. Wait until 'resume' is pressed
5. resume the game and continue normal execution
6. Game is quit manually
7. End game and return to main screen

Notes

* Local games can be played without an internet connection.

### 4. View Tutorial

Actors

* Player (Bob)

Description

* Bob wants to know how the game functions.

Trigger

* Bob selects the tutorial button in the main screen.

Main scenario

1. Bob selects tutorial in the main screen
2. The tutorial text is shown
3. Bob returns to the main screen

## Actor characteristics

Main actor of our use cases in Bob:

Bob is a informatics student at the university of Bern.

So Bob is experienced using software, and acquiring new skills.

Bob has owned Android phones since the first Android came out.

Bob so has a lot of experience in using Android software.

Second actor is Susie:

Susie has already installed and used our app.

# Specific Requirements

## Functional Requirements

### Use case 1. Play Game

* Makes playing Word Finder game.
* Home-screen has a start game button to start game.
* User makes his profile at first opening, user database is created.
* Play game initializes a new 7x7 word game grid, sets time and score to initial value. (minimal word-count of 10 words and minimal word-length of 3 letters are assured).
* Initial value of score is 0.
* Initial value of time is the time set to the value defined in the settings.
* Internet connection for synchronization.
* Pause button stops the game (time halts) and shows pause screen.
* Pause screen shows a resume button.
* At end of game time and score is showed to player, and achievement of new hight score is shown to player.

### Use case 2. View stats

* Shows own scores.
* Shows friend’s scores.
* Score button in main screen.

### Use case 3. Retry board

* Play a known board once more.
* Retry button beside the scores of played board.
* A retrial count as a normal game, and thus manipulates the board recordings in the same way.

### Use case 4. View tutorial

* Play a known board once more.
* Retry button beside the scores of played board.

## Nonfunctional Requirements

### Use case 1. Play Game

* <Performance requirement> : <System start (home screen) under 10s.>
* <Portability> : <System portability has to be specified before installation and download of the App.>
* <Space requirement> : <Installation space have to be specified before download and installation. (kbyte)>
* <Space requirement> : <Minimal RAM space during use has to be specified before installation. And checked for availability before usage.>
* <Space requirement> : <Space used for offline recorded scores/times/Board/ is specified and checked before playing game.>
* <Performance requirement> : <System has to stop immediately when pushing pause, the screen has to be covered in an instant.>
* <Legislative requirement> : <Copyright and terms of use has to be shown to the User before first launch of the App (during download or installation).>
* <Legislative requirement> : <User has possibility to modify or correct his own user record database.>
* <Efficiency requirement> : <Internet connection is checked when opening the game. If no connection is established a offline game will be started and game history or statistics are saved in a external file for recovery when the connection is reestablished. Assures no continuous connection searching, and performance lost by this.>
* <Performance requirement> <Opening a new game should not take more than 1s.>

### Use case 2. Show stats

### Use case 3. Retry board

* <Space requirement> : <text file for the generation numbers of the last 20 boards and the top 5.>

## General

* <Backup> : <No backup is done fore the game in execution, excepted the stats sent to the server.>
* <Software Requiurements> : <App funktions on kitkat Android (API 19).>
* <Minimal Reqirements> : <Android API 8.>
* <Backup> : <Github for version control an backups.>
* <Interoperability> : <Word Finder Game is designed to work alone, and for synchronization with a server.>
* <Price> : <Set by "Customer".>
* <Copyright> : <"Customer".>
* <Open Source> : <Code is not open source.>
* <Priority> : <For all interactions with friends a internet connection has to be established, in order to synchronies the action. If not a, action is saved in a file for later synchronization.>
* <Fault tolerance> : <Every action modifying the friends list has to be confirmed at least once.>
* <Response time> : <average time to respond non during gaming is 10s-15s during gaming 1s for selecting max 5s for verification.>
* <Delivery requirement> : <Product delivery according to scheduled time by the course ESE HS 2013.>
* <Ethical requirement> : <No specification about a word filter.>
* <Testability> : <All requirements has to be tested.>
* <Testability> : <Test are done whit Jankins.>
* <Supportability requirement> : <Group 4 ESE HS 2013 is available for support questions and issues.>
* <Disaster recovery> : <System shuts down and no data is saved.>
* <Documentation> : <User manual or short tutorial is available.>
* <Extensibility> : < - >.
* <Usability> : <Specification comes with first prototype.>
* <Accessibility> : <App launched on Swiss stores for android apps, available to all without restrictions.>

Annotation: Tried not to write the same requirements more than one time, some points ar missing requirements but would be covered by redundancy of the other scenarios.

1. Dictionaries can be managed in the settings screen [↑](#footnote-ref-1)
2. The timer lenght can be set in the settings screen [↑](#footnote-ref-2)