Abstract
This Software Transfer Document (STD) for the ClubNet system entails all the information required for a successful transfer of the project to our customer, Intuitive Technologies B.V. The document includes build and installation procedures and a report of the acceptance test of July 17th, 2016. Our encountered problems are also described in this document. This document adheres to the ESA [12] standard for software development documentation.
# Contents

1 Introduction .......................................................... 6
   1.1 Purpose ......................................................... 6
   1.2 Scope .......................................................... 6
   1.3 Definitions and Abbreviations .............................. 7
      1.3.1 Definitions .............................................. 7
      1.3.2 Abbreviations .......................................... 7
   1.4 List of References ............................................ 7
   1.5 Overview ..................................................... 8

2 Build procedure .................................................. 9
   2.1 Server .......................................................... 9
   2.2 Android app .................................................. 10
   2.3 iOS app ....................................................... 10

3 Installation procedure .......................................... 11
   3.1 Development .................................................. 11
   3.2 Android app .................................................. 11
   3.3 iOS app ....................................................... 12

4 Configuration item list .......................................... 13
   4.1 Documentation ............................................... 13
   4.2 Test plans .................................................... 13
   4.3 Software ...................................................... 13

5 Acceptance test report summary ............................... 14

6 Software problem reports ...................................... 15
   6.1 Heroes module ............................................... 15
   6.2 Sponsoring module ......................................... 15
   6.3 Forgot password app ....................................... 15
   6.4 CoachAssist connection ................................... 15
   6.5 Packaging .................................................... 15

7 Software change requests ....................................... 16

8 Software modification reports ................................ 17
DOCUMENT STATUS SHEET

GENERAL

Document title: Software Transfer Document v1.0.0
Identification: STD/1.0.0
Authors: J. Verhagen, T. Komar
Document status: Draft version

DOCUMENT HISTORY

<table>
<thead>
<tr>
<th>Version</th>
<th>Date</th>
<th>Author(s)</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0.1</td>
<td>02-06-2016</td>
<td>J. Verhagen</td>
<td>Initial document structure</td>
</tr>
<tr>
<td>1.0.0</td>
<td>30-06-2016</td>
<td>J. Verhagen, T. Komar</td>
<td>Wrote document</td>
</tr>
</tbody>
</table>
## DOCUMENT CHANGE RECORDS

<table>
<thead>
<tr>
<th>Section</th>
<th>Reason</th>
</tr>
</thead>
<tbody>
<tr>
<td>3</td>
<td>Added environment configuration</td>
</tr>
</tbody>
</table>
1 INTRODUCTION

1.1 PURPOSE

This is the Software Transfer Document (STD) of the ClubNet application. This document describes the configuration of items transferred and the procedure for building and installing the application. It will also report problems encountered and requirements not met.

1.2 SCOPE

ClubNet is a software system containing the ClubNet mobile application and a web interface. The ClubNet mobile application is designed for smartphones and tablets, and the web interface is designed for all modern web browsers. The entire ClubNet software system is conceived by Intuitive Technologies B.V. and developed by The Brofessionals. The purpose of the ClubNet system is to assist coaches and PR managers at football clubs in organizing training and club related activities in an efficient manner. The ClubNet application provides a controlled communication mechanism to help coaches arrange activities while the web interface will be used by PR managers to manage the activities happening in the club. Even though the current scope is only about football, the ClubNet system can be extended to be used for all type of sports in the future.
1.3 DEFINITIONS AND ABBREVIATIONS

1.3.1 DEFINITIONS

Android | A mobile operating system mainly developed by Google
AngularJS | An open-source web application framework mainly maintained by Google.
Brofessionals | The development team of ClubNet
CoachAssist | An independent software system developed by Intuitive Technologies B.V. [13]
Cordova | A framework for creating cross-platform mobile apps
Intuitive Technologies B.V. | A software engineering company situated in the Netherlands serving the role of client.
Ionic | A complete open-source SDK for hybrid mobile app development [15]
iOS | A mobile operating system developed by Apple Inc.
Meteor | A full-stack JavaScript solution for web apps written using Node.js [16]
MongoDB | A cross-platform NoSQL database [17]

1.3.2 ABBREVIATIONS

PR | Public Relations
SDK | Software Development Kit: a set of development tools for the creation of applications for a certain framework
URD | User Requirements Document
SRD | Software Requirements Document
ADD | Architectural Design Document

1.4 LIST OF REFERENCES

References

1.5 OVERVIEW

The remainder of this document consists of seven chapters. Chapter 2 gives information regarding the environment in which the software was built. This includes which version was built and problems encountered. Chapter 3 describes information regarding the environment in which the software was installed. This includes what exactly was installed and problems encountered. Chapter 4 gives an overview of all configuration items transferred. Chapter 5 describes which user requirements were not met as well as a summary of test reports in the AT. Chapter 6 gives a list of problem reports raised during the transfer phase. Chapter 7 describes a list of software change requests raised during the transfer phase. Chapter 8 gives a list of software modification reports completed during the transfer phase.
2 BUILD PROCEDURE

The application consists of three parts. The part deployed on the server contains the back-end logic and the web interface. The Android app and iOS app are build separately. The build procedure as described in the DDD [4] are followed.

2.1 SERVER

The Clubnet application uses a Node.js server with Meteor framework installed. There are various hosting providers offering such a setup, and we have used Scalingo (http://www.scalingo.com) hosting provider.

To deploy Clubnet to the Scalingo server, one has to set up:

- A GitHub repository.

- A Scalingo account.
  Sign up link: https://scalingo.com/users/signup

- SSH keys to identify that one is allowed to push the application files to the Scalingo server.

- The Scalingo Command line tool.

When these things are set up these steps should be executed in the command line:

1. Go (command line: cd [Path]) to the Clubnet application root folder.
2. Type in: ‘git init’. 
3. Type in: ‘git commit –m “Clubnet application”’. 
   Now we have our Clubnet application initialized and committed to the git repository.
4. Create a Scalingo application.
   In the command line type in: ‘scalingo create application-name’. Note: application-name should be changed to the name of the application that will be used as a domain to access the application.
5. Create MongoDB .
   In the command line type in: ‘scalingo -a meteor-app addons-add scalingo-mongodb free’
6. Deploy the application from the latest commit in the master branch on GitHub:
   In the command line type in: ‘git push scalingo master’

7. The application is now accessible on http://application-name.scalingo.io.

### 2.2 ANDROID APP

To build an Android APK file one has to follow the following steps:

1. Open command line in the Clubnet root folder.

2. (If not installed yet) - type in `meteor install-sdk android`.

3. (If not installed yet) - type in `meteor add-platform android`.

4. Type in: `meteor build [Output directory path] --server=[Application server URL].`
   
   *Note:* [Output directory path] should be replaced with the absolute path of the directory to which one wants the APK file to be saved to.

   *Note 2:* [Application server URL] - has to be replaced with the URL of the deployed Scalingo application server - for example, http://clubnet.scalingo.io

### 2.3 IOS APP

To build an iOS APK file one has to follow following steps:

1. Open command line in the Clubnet root folder.

2. (If not installed yet) - type in `meteor install-sdk ios`.

3. (If not installed yet) - type in `meteor add-platform ios`.

4. Type in: `meteor build [Output directory path] --server=[Application server URL].`
   
   *Note:* [Output directory path] should be replaced with the absolute path of the directory to which one wants the IPO file to be saved to.

   *Note 2:* [Application server URL] - has to be replaced with the URL of the deployed Scalingo application server - for example, http://clubnet.scalingo.io
3 INSTALLATION PROCEDURE

3.1 DEVELOPMENT

Setting up development environment includes these steps:

1. Install Node.js 4.4.3
2. Install Meteor 1.3
3. Install Ionic: command line: `npm install -g ionic`
4. Install Cordova: command line: `npm install -g cordova`
5. Install Git.
6. Install IDE (example: Web Storm).
7. Import Clubnet project to the IDE with Git.
8. Open command line and go to the Clubnet project root folder.
9. Type `meteor add driftyco:ionic`, press enter (this adds Ionic to the project).
10. Type `Meteor add angular`, press enter (this adds AngularJS to the project).
11. Now remove some dependencies: `meteor remove blaze-html-templates`, enter, `meteor remove ecmascript`, enter.
12. Run project for the first time so it can initiate: `meteor` (this might take a while).
13. Stop meteor by pressing Ctrl+C twice.
14. Connect Android or iOS device.
15. Respectively type: `meteor run android-device` or `meteor run ios-device`.
17. Application starts on the connected device.

Note: When running on Android device, make sure that ‘Development mode’ is turned on.

3.2 ANDROID APP

For production - download the application from Google Play store and press install.
For development - go to the root directory of the Clubnet application and in the command line type: `meteor run android-device --server=[Remote server URL]`.
Note: [Remote server URL] should be replaced with the url of a remote server at which the application is running. If left blank, it defaults to `http://localhost:3000`.
3.3 iOS APP

For production - download application from iStore and press install.
For development - go to the root directory of the Clubnet application and in command line type: `meteor run ios-device -server=[Remote server URL].`

Note: [Remote server URL] should be replaced with the url of a remote server at which the application is running. If left blank, it defaults to `http://localhost:3000`.
4 CONFIGURATION ITEM LIST

This chapter gives an overview of all the items delivered. The documents are delivered in PDF format. For the software, the source files are delivered.

4.1 DOCUMENTATION

- URD[1]
- SRD[2]
- ADD[3]
- DDD[4]
- SUM[5]
- STD[6]

4.2 TEST PLANS

- UTP[7]
- ITP[8]
- ATP[9]

4.3 SOFTWARE

The ClubNet software system including:

- ClubNet mobile application
- ClubNet web interface
5 ACCEPTANCE TEST REPORT SUMMARY

Description All tests as described in the ATP version 1.0.0 [9] have been executed during the acceptance test.

Activity and event entries The acceptance test was performed on June 17, 2016 from 9:30 to 12:00. Attendees were the customer, the whole project team, the project managers and the project supervisor. The project supervisor had to leave early. All acceptance tests passed.
6 SOFTWARE PROBLEM REPORTS

6.1 HEROES MODULE

The heroes module is not completely implemented due to time constraints. It is possible for a coach to add a heroes item including text and image, but the PR side is not implemented at all. This means that a PR user cannot regulate how many heroes items can be created and that they cannot view/add/delete heroes items.

6.2 SPONSORING MODULE

The sponsoring module is also not completely implemented due to time constraints. Sponsoring items can be created by a coach, but other users are not able to indicate they want to sponsor an event. All interactions between a sponsoring item and a PR user are also not implemented. So PR users are unable to delete sponsor events or add external sponsors to it.

6.3 FORGOT PASSWORD APP

Handling a forgotten password through a link in a mail is not working due to technical constraints. To open the app from a link we need to use a custom url scheme. These links work in a browser, but mail clients disable them for security. To solve this issue, the user can copy paste the forgot password token in the app manually.

6.4 COACHASSIST CONNECTION

The system does not have a connection with the CoachAssist system at the moment. This is impossible because the necessary APIs are not available to us. Importing coaches and teams is thus impossible, they are now defined in our own system. Furthermore, the importing of exercises for a exercise voting is also not possible. As a replacement we use a local file in which exercises are defined.

6.5 PACKAGING

To use implemented functionality across the entire ClubNet system, the goal was to group the reusable and sharable code into packages. Packages are a built-in feature of Meteor that allows for better reuse of code. However, because of our dependency on AngularJS, it was not possible for us to get the packages to work correctly. We have decided to simply moduralize our code base as much as possible, as opposed to spending too much time on the packaging issues we faced.
7 SOFTWARE CHANGE REQUESTS

Not applicable
8 SOFTWARE MODIFICATION REPORTS

Not applicable