

Coding Standards Document

July 18, 2018

Team Name: Binary Ninjaz

System Name: Harvest



Client Name: Barry Christie

Team Members:

Teboho Mokoena (u14415888)

Sizo Duma (u15245579)

Letanyan Delon Arumugam (u14228123)

John Ojo (u15096794)

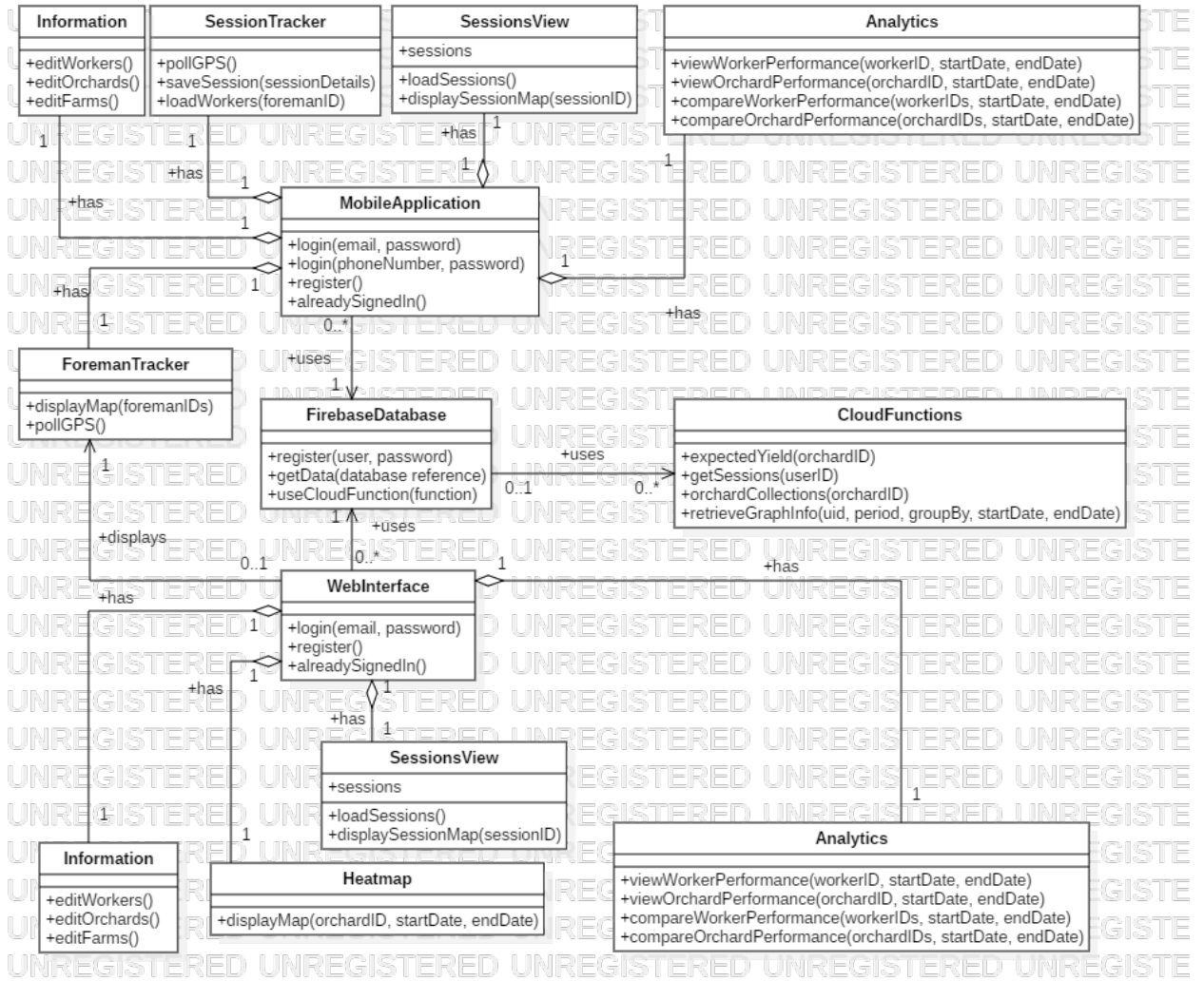
Kevin Reid (u15008739)

Shaun Yates (u16007493)

Contents

1	Detailed System Design	3
2	Coding Conventions	4
2.1	HTML/CSS	4
2.2	JavaScript	4
2.3	Java	4
2.4	Swift	4
3	File Structure	5
3.1	Branching	5
3.2	Naming	6
4	Code Review Process	6

1 Detailed System Design



2 Coding Conventions

The project and design we've undertaken means that we will need to use multiple languages and IDE's to complete the project. As such we have decided that the best common practice and standards of each language shall be the standard that is used for that respective language/environment.

2.1 HTML/CSS

The HTML and CSS standards we shall follow will be the ones of [Google's HTML/CSS style guide](#)

2.2 JavaScript

For JavaScript we will be using [Google's JavaScript style guide](#).

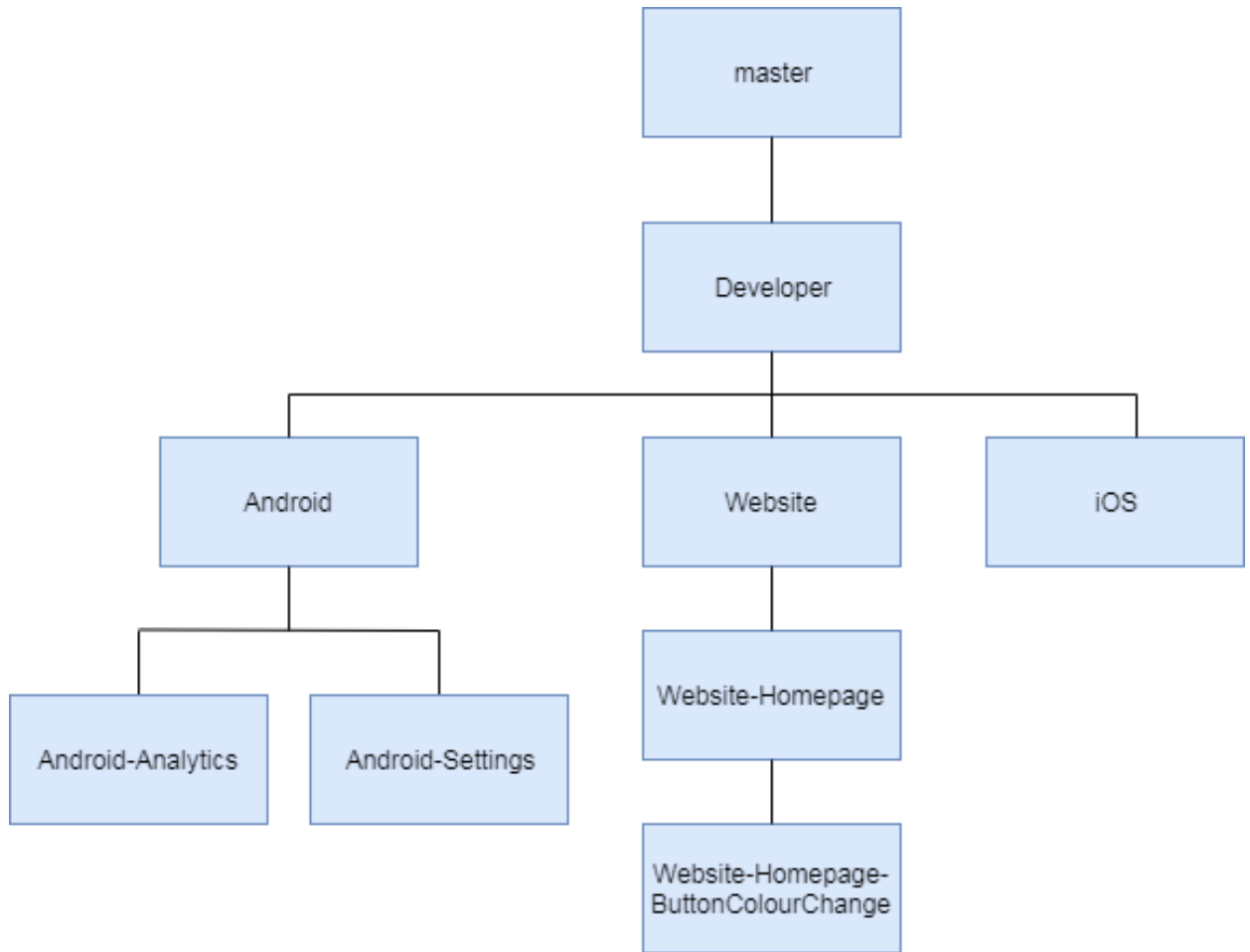
2.3 Java

Similarly we will use [Google's Java style guide](#).

2.4 Swift

For Swift we will use [Apple design guidelines](#).

3 File Structure



3.1 Branching

An example structure is shown above, showing that **master** is never worked on directly. Each subsystem has its own respective branch, and these branches branch off **Developer**, and in turn any changes done to a branch would then branch off that specific branch. A **Developer** branch is used, to act in much the same way as **master**, but as a proxy, so that before merging that branch to master, the entire system can be reviewed there before a more solid commitment is made.

3.2 Naming

A branch shall never contain an individuals name, rather it must be informative as to what the intention of the branch is whilst also remaining short. An example of a name that would be deemed unacceptable is `Joe`; another would be `JoeWebsiteChanges`, where the name contains a member name, and gives very little information as to what is going on in the branch. Bad, but barely acceptable names are `WebsiteChanges`, or `WebsiteExperimentation`. Ideal names are as follows: in the case of a branch where the website is ultimately being assembled, by merging other branches into it, and the only modifications taking place are to structure the files, or a quick fix, such as changing the colour of a button, `Website`; in the case that a new homepage is created for the website, where the homepage is created, and merged into `Website` on completion, `Website-Homepage`. Finally, if a user wants to propose a change to the `Website-Homepage` branch, where he wants to change the colour of a button, but the user has not been assigned, or is not the primary benefactor to `Website-Homepage`, he creates a new `Website-Homepage-ButtonColourChange` branch. Notice that in the ideal examples it is possible to track down the source of a branch, to understand exactly what it's purpose is.

4 Code Review Process

All code reviewing is done on GitHub, and occurs every time there are changes done to code, or if new code has been implemented. Currently, pre-commit reviews are utilized and although it is a slower process compared to post-commit reviews, it always for more people to be aware of changes to the code and could possibly lead to better implementation as more eyes usually means more ideas.

After changes have been made on the branch, the creator, who wants to merge, will then make a pull request on GitHub, and announce the request on Slack. After which it shall become the responsibility of the entire team, but more importantly that of the original branch maintainer to analyze and comment on the changes. The exact location of the discussion is trivial, however GitHub provides a better, and more concrete platform for this discussion to take place, where inline comments can be made. After a sufficient amount of time and discussion has passed, either the branch

maintainer (if it is a minor change), or the entire team (if it is to master, or a major change) shall decide to merge or not.