**Team 19 Project Installation Guide**

This document contains the information necessary to install, configure and use the system correctly. It is necessary to read this documentation so that you may configure the system correctly when deploying it to test it. Any assumptions for your installation environment will be stated. A copy of a deployed backend will be made available and preconfigured on *homepages.cs.ncl.ac.uk* but it may be useful to have a local copy of the backend for testing out various scenarios. This document also provides critical information relating to some account information that will be necessary to use the application to its fullest.

**Installing the Backend & Website**

We assume that a version of MySQL is installed on your local machine which you wish to deploy the backend copy to. We also assume that you have a version of an Apache server installed with PHP/MySQL support as well as a configured mail-server within the PHP installation for email notification services to work. Under the *Accounts* section in this document we provide an example service account for this purpose if you need to set up a mail-server. We also assume that services such as SSL, cURL and a list of certificate authorities are enabled within your PHP installation. These are necessary for some features of the application to successfully work and are server configuration items. It is then necessary to create a database named *lloydsapp* with your Database Management System of choice. Within the file *backend*/*config.php* modify the documented variables to match your local machine’s installation environment.

You will need to also modify the root URL of your installation in *backend/config.php*; we recommend using *ngrok* [1] to provide a tunnel to your localhost if you are not installing the backend on a device which is globally reachable via the Internet. The tunnel system will provide a domain for your base URL and a way to make your localhost site globally reachable and hence allows the android application to communicate with it. We also recommend installing the backend on a root directory, rather than a relative directory.

It will also be necessary to modify the *backend/.htaccess* file to match your base url. If the backend is installed on a root directory, then this can be ignored. However if the backend is installed on a relative directory such as *localhost/backend/* rather than just *localhost* containing the files, then the *RewriteBase* must be modified to reflect that as documented in the *.htaccess* file.

Finally, our backend and website system uses a dependency injector, namely Composer [2] to inject dependant libraries into the project. It will be required for you to download this tool and run *composer install* in the root directory of the backend. This will automatically create a vendor folder in the directory structure containing all of the required libraries needed to run the backend system.

Once you have completed all of these steps, the backend should be ready to run. Accessing the index page will bring up the home page of the site and also on first-run create an administrator account for you to use to populate the database with users. The details of this account are provided in the *Accounts* section.

**Installing the Android Application**

We assume in this section that you are using Android Studio version 1.1 or above. It is important that the *debug.keystore* file provided in Android Studio must replace the *debug.keystore* file within your locally installed Android Studio version. This is done by finding your Android Studio configuration directory (on Windows, this is *C:/Users/<Your Name>/.android/* and on Unix based systems it is *~/.android/*) and overwriting the file with the one we provide. This will allow the application to be authenticated for Google Play services.

Once this step is complete, you may import the project as normal, and the Gradle build system will resolve any dependencies and libraries needed for the project will be automatically imported – it is necessary to have the latest versions of the SDK and its components all installed as well on your machine. Once this is complete, configure the *strings.xml* file’s *api\_base\_url* to match the installation URL of your backend. Once this is done, recompile the project and you will be able to run and use the application. The application will run both on Android devices and emulators – due to some known glitches within emulator packages, Google Play services may not fully work depending on what type of emulator you use. A physical Android phone is recommended for these features. For further limitations and requirements refer to the specification document.

**User Accounts**

The following user account is issued as an administrator account on the backend website. This is useful for the purpose of creating users and user bank accounts for the purpose of the backend. This is done via *User Management* after logging in.

**Username:**  [admin@lloyds.com](mailto:admin@lloyds.com)  
**Password:** admin  
**Security Code:** lloyds

The following user account is to be used for your Google Play services account. This is a limitation of Google services, as they require only validated tester accounts to be able to use Google Play features before an application is published to the application store. This account is also required in the Google+ sign-in feature on the website’s leaderboard and achievements pages for similar reasons.

**Username:** [team.19.service@gmail.com](mailto:team.19.service@gmail.com)  
**Password:**  Team1919

The above user account is also recommended to be used if you need to set up a mail-server on your local backend, though it is not a requirement if you already have one set up.

**References**

* [1] Ngrok (2015) *Ngrok: Introspected tunnels to localhost.* Available at: https://ngrok.com/ (Accessed 01/04/2015)
* [2] Composer (2015) *Dependency Manager for PHP.* Available at: https://getcomposer.org/ (Accessed 02/04/2015)