**G52GRP 2017/18 Expected final documentation**

You will be expected to provide:

* A final report for your project as outlined in the pro forma below.
* The main purpose of the group report is to present your group’s work to your supervisor (and external supervisor). It should be written formally – as a report from a company that has been awarded a contract to the company that issued the initial call. Thus it should be written as a professional document and assessment will be negatively affected if you use informal language.
* Instructions for use for your software (user manual).
* Documentation of the software functionality
* To the point where a group following on could take the project forward
* For code this will include appropriate documentation such as
* Clear and consistent use of conventions for variables etc.
* Only appropriate levels of comments
* Clearly structured code where functions/methods have only one purpose.
* Evidence of refactoring to avoid over complex code structure

|  |  |
| --- | --- |
| **Group Project Report** | Group 20 |
| **Date of Submission** | 25th April 2018 |
| **Expected Submission Date** | 25th April 2018 |
| **Project Title** | Displaying Aircraft Movements and Positions in 3D Space |
| **Sponsoring Company (CS if local only)** | CS |
| **Contact Person** | Jason Atkin |
| **Contact Email** | [Jason.Atkin@nottingham.ac.uk](mailto:Jason.Atkin@nottingham.ac.uk) |
| **Project Supervisor <or as above for local>** | Jason Atkin |
| **Supervisor Email** | [Jason.Atkin@nottingham.ac.uk](mailto:Jason.Atkin@nottingham.ac.uk) |
| **Group members names and emails** | |
| Joel Cox [psyjc5@nottingham.ac.uk](mailto:psyjc5@nottingham.ac.uk) | |
| Ruiyi Lin [psyrl5@nottingham.ac.uk](mailto:psyrl5@nottingham.ac.uk) | |
| Rumyana Cherneva [psyrc5@nottingham.ac.uk](mailto:psyrc5@nottingham.ac.uk) | |
| Emmanuel Papadakis[psyep1@nottingham.ac.uk](mailto:psyep1@nottingham.ac.uk) | |
| Norah Alageel [psyna3@nottingham.ac.uk](mailto:psyna3@nottingham.ac.uk)  Hugo Ozanne[psyho1@nottingham.ac.uk](mailto:psyho1@nottingham.ac.uk) | |
| Zhi Fu Lim[psyzfl@nottingham.ac.uk](mailto:psyzfl@nottingham.ac.uk) | |
| **Project Background Information** | |
| <Fill in here the tender as you understood it. Changes to the initial specification and when and how they were introduced. If you have been following an agile methodology these changes may have occurred throughout the project as requested by the sponsor/supervisor>  The project description made it clear that we were asked to create a piece of software to display aircrafts movements and positions in 3D space and help visualise routes. The language and target platform were not specified so we could decide which one to use, however we agreed that we’d aim to have our project run on the three main Operating Systems: OSX, Windows and Linux. As a group we liked the idea that this software will be used by specialists who work on aerospace research projects and it will hopefully play a part in the future of airline travel.  Originally, we thought we were going to implement a data scraping tool to get information from web pages and store it in data files as one of the task’s requirements. After a discussion with our supervisor we decided at his suggestion that this feature should be removed from the requirements, as we were provided with a scraping tool already designed for the task and a lot of scraped data. For our software, we would need to be able to read the files and display them in 3D space, with planes’ specified locations from any provided data file (.csv format). It was clear that we will need a clean user interface to allow navigation tools such as: going forward or backwards in time, pause, zoom in and out, and to filter data. A simple user manual for reference was also recommended.  During the development process, we agreed with the sponsor to adjust the requirements slightly, as the concept for the end product changed. This included removing the aforementioned data scraping component, switching from accepting multiple data formats to only accepting Comma Separated Values (CSV) files, as well as redefining the expected time controls to be implemented. These features were considered either unnecessary or overcomplicated for the final product by the sponsor, and thus changed or removed.  A feature that was not included in the original tender but was added to the initial requirements was the implementation of trails following each individual aircraft.  The 3D representation aspect was written such that the program renders a 3D space, with a globe and aircraft around it. Included in this was a moveable camera, which allows for different viewing angles. All airports around the world are represented as bright green dots on the map so they can be distinguished from the aircraft trails.  **(500 words maximum)** | |
| **Progress on Work at Time of End of Contract** | |
| <Summarise here the extent to which you have achieved in your deliverable the objectives outlined above. Wherever objectives have not been achieved outline the extent to which you have not achieved the objectives and the reasons for it. Also give information on future plans for your project>  We have delivered all of the basic requirements listed in the requirements document, as well as several “expandability” requirements that we have highlighted below our core requirements.  Our data system was designed as a set of nested vectors, which are populated with data that are read from the input .csv files. These .csv files are selected at runtime, so a user can provide their own data files. The data files we have used are parsed JSON ADS-B exchange data in csv format for the aircraft locations and data points. We also used a .csv file for the airport locations.  The 3D Visualisation was built using the open source graphics library OpenGL. This includes rendering a sphere, calculating relative positions for latitude, longitude, and altitude. Our system plots each plane’s position and their previous positions, and then connects them to illustrate its flight path trail. We have plotted all of the airports on the earth as fixed green dots to allow for easy identification. We have also plotted ground textures on the 3D model of the earth outlining the respective regions.  We have also created lighting, shaders and other graphical enhancements to improve the render quality of the project, this allows for the final product to feel professional and look refined. A standout feature of this is the mouse driven camera control system, which provides a smooth viewing of the visualisation.  We have implemented a drop-down menu complete with several navigation buttons. First, camera controls which give the user to examine the aircraft traces as desired. Second, plane filters for separation of particular aircraft traces so they are easier to interpret. Lastly, time controls allowing for the ability to pause and go forwards and backwards.  Additionally, we have created a cleaning method for data input into the system, by removing any corrupt data entries; this will prevent unnecessary crashes.  One of our extendibility requirements was to provide 3D aircraft models on the end of each aircraft traces. A group member worked on them and managed to partially implement them, this has been completed however integration with the master system has proven problematic; this will be worked on in the following weeks left before the project delivery date.  Another extendibility requirement was to provide trails of the aircrafts’ flight paths over time. This feature has been implemented, and can be toggled on and off using the onscreen buttons in the navigation menu.  **(1000 words maximum)** | |

|  |  |
| --- | --- |
| Documentation of the project is here. | Git repo, folder in parent directory  /Documentation/ |
| The code repository is here | https://projects.cs.nott.ac.uk/Group20/Displaying-Aircraft-Movements-And-Positions |
| Other project management documents (e.g. Trello boards) are here. | https://trello.com/group2015 |