Chapter 2

Requirements Analysis

The project aims to design and develop a usability centric flight tracking web application where a user can track flights globally on a map and view flights locally using augmented reality. The project will be achieved using a front-end web application where user requests can be made and visualised. Requests will be sent to a server with an API returning the flight data. The requirements specification has been derived from the literature carried out in the research conducted in the literature review. The areas of research conducted were ‘Flight Tracking Technology’, ‘Flight Tracking Application Programming Interface’, ‘Augmented Reality’ and ‘Usability’.

The requirements are composed using the MoSCoW method to prioritise the different requirements into ‘must’, ‘should’ and ‘could’. The highest priority requirement ‘must’ if not met the web application would not fulfil the purpose of the project. The second priority ‘should’ if not met does not affect the functionality of the project; however, does improve the final application. The last priority ‘could’ are requirements which are not necessary to the overall functionality of the application; however, if development time allows for their inclusion, they would give value. From the carried out research, the following requirements were created:

2.1 Functional Requirements

* Must use ADS-B data source
* The application must use the most relevant data source of the current time
* Accuracy of data
* Must use MLAT data source
* The application must use the traditional data source used by older aircraft
* Ensures diversity of aircraft to be visible to users
* Must use a Flight Tracking API
* Data must be requested from an API containing live flight data.
* The flight data source must have an extensive global coverage.
* APIs such as the OpenSky Network or FlightAware must be used.
* The API must be rigorously tested
* API must be tested to ensure correct data is retrieved from the API to the user. Testing must be conducted with different parameters being passed through the request to the server.
* Must be able to make user requests to an API
* Must be able to return flight data dependent on request parameters supplied by the user.
* Must have a usable interface in order to make requests.
* Must display requested flight data onto a map
* Could implement a 3D map to display flight data.
* Must use an Augment Reality library
* Must use a web application based AR library such as AR.js in comparison to a native application based AR library such as ARKit.
* Must have interlinking screens to navigate the application
* The application navigation must be able to transfer between screens without any issues.
* Should have the ability to return to previous pages saving prior requests made rather than users having to resubmit requests.
* Should have seamless transition from the main page to the camera function of the Augmented Reality.
* Should only request the permission of use of the users camera once throughout the current visit of the web application.
* Should display all flights initially on the map
* The application should display all flights globally before any user requests on specific flight data are made.
* Should be responsive to users
* The application should be able to deal with user input and handle interrupts and be able to return to previous states.
* Could be implemented to function on a smartwatch device
* The application could be implemented or have a separate application to function on smartwatches.
* Integrate the camera from the user’s mobile phone and view augmented reality footage on the smartwatch.
* The watch could view a map of local flights.

2.2 Non-Functional Requirements

* Must have a scalable user interface
* The user interface must scale to different mobile devices due to the variation of screen size.
* Must be available to be used at all times
* The application must not have little or no downtime and be available for users at any time of day.
* A percentage uptime of between 95% to 99% should be achieved.
* Must be user-accessible throughout the application
* Must use usability inspection models to determine the usability of the application in order to adapt prototype to a refined product based on the feedback given by the models.
* Such models to be used are the Heuristic Evaluation, Cognitive Walkthrough and System Usability Scale.
* Must use HTML, CSS and JavaScript for development
* Due to the product being a web-based application, HTML must be used for the foundation to build the application.
* Styling of the web application to be implemented using CSS to engage visual appeal to users.
* JavaScript should be used in order to develop functionality of the web application such as calls made to the API and general functionality of the application.
* Must run on a range of web browser
* Users must be able to use the application on web browsers such as Google Chrome and Firefox.
* Must be cross-platform accessible Augmented Reality
* The AR library must be accessible on different web browsers such as Google Chrome and Firefox.
* Should have the same theme throughout all pages of the web application
* The application should follow a theme through each individual page of the application.
* Should have application documentation
* Should provide documentation to user on how to use application.
* Should have a short processing time
* The application should request data from the server API in an appropriate time of 3 seconds
* Could implement a splash screen or visual loading progression mechanism to give the user the visual cue of the loading process of the data requested. Diverting the attention of the user focusing on time and directing onto the visual cue.