**Requirements**

Key:

- ref. = reference

- qq# = questionnaire question #

- The style of the system is up to us as long as the system has all the functions required (ref. qq1)

- The method of storage is up to us (database is not required, text files are fine) (ref. qq2)

- 5 roles exist in the system: (ref. qq4)

- Customer - Register account, book flights, change personal details, view details, cancel flights.

- Travel agent - book flights for customers.

- IT manager (God) - Full priority to everything.

- Staff - create accounts for customers, book flights for customers, view customer details.

- Management (flight control, booking manager, service manager)

- No one can add new flight paths into the system.

- Registration and login is required before using the system. (ref. qq3)

- Passwords do not need to be encrypted.

- Passwords need to be hidden by asterisks on the screen.

- Customers must be able to see their booked flights' status' (food services, flight times, prices, seat# etc.)

- The only seating options are first class and business class (random seat).

**- STRETCH GOAL:** More specific choices available (isle/window, row, seat#).

- Customers may have a warning or no-fly status.

- If a customer with a warning status books a flight, a management class (booking manager?) must be alerted via the main screen when they log in.

- A customer with a no-fly status cannot book flights.

- Only the management class can generate a report.

- If customers cancel their booking they are charged a service fee.

- When customers cancel a flight, don't notify managers, but keep a record of the cancellation.

**- STRETCH GOAL:** Rescheduling functionality.

- Customers can change their own details (address, ph# etc). They cannot change data that the system uses to identify them(customer#) (ref. qq9)

- GUI interfaces are not required, but recommended.

- Whatever the interface is it MUST be useable/intuitive.

- Client/Server architecture is a stretch goal (ref. qq14)

- Program should run on windows and linux (subject to change) (ref. qq15)

- If a flight is cancelled by the system or a manage, all customers that booked that flight must be alerted when they next log in.

- Prices and other specific details are up to us as they don't make a difference on the functionality of the system.

- We need to create template reports to show the customer and assist us in achieving a mutual understanding of the desired requirements.

- Managers need to be able to generate reports.

- Calculate prices of services/flights.

- Stocking of food, drinks, technology etc.

- THINGS TO DO FOR THE NEXT CLIENT MEETING:

- Get the client to go through system process use cases step by step

- eg what is shown to user after login?

- eg what info/screens are shown to user during a successful flight booking?

- What are they shown if booking is unsuccessful?

- Create storyboards of system processes for client feedback.

- Create template reports for client feedback.