Ace for Hire

Specification document by Paul Robson

Introduction

“Ace for Hire” is the working title of a career mod add on to Infinite Flight. It will be browser based, and free open source software, currently under the MIT license.

I wanted to have some of the feel of Infinite Flight in, and I have been looking at some of the available add ons for other flight simulators.

With this in mind, I wanted it to be :

- Free, not payware.
- A Javascript (probably Typescript) app , so it should run on anything much.
- Relatively uncomplicated, so that the actual management doesn’t interfere with the game.
- Allow a fair amount of immediate choice. People like flying airliners, more than small GA craft. I didn’t want something that meant you had to start with a C172.
- Flexibility. I wanted three things here ; the ability to travel roughly anywhere you wanted, the ability to fly like a localised airline, and the ability to have some sort of sequence of flights.
- Users to be able to create route groupings and flight sequences and so on.
- Able to be expanded to include online features like leaderboards and so on.

Gameplay

As you might guess from the title, the pilot here is self employed who can pick up any of the jobs available for him at his current location, rather than employed by any single entity such as an airline (though it is possible to effectively behave like this).

Each pilot has a bank balance, an experience score, a rank (which relates directly to the experience score), a home airport, and a name. This describes their current state.

When they fly a mission, they are paid a rate to that, which is driven by things such as the length of the flight and their current rank. Against this, there are daily expenses (in real time) which are also driven by the rank (and possibly, later, the location they are staying in). The bank balance cannot be negative.

Routes

Basic routing is derived from the current location.

A random selection of flights will be created from the current location to assorted airports at various distances, including long East/West flights to allow crossing of continents. These will be a mixture of passenger and cargo missions, the capacity for the latter will be approximated. Locations will initially be limited to International airports. There is no limit on what you can fly based on rank.

If you are not at your home, there will always be a flight home. I also plan to have a “catch a lift home” button that effectively warps you home.

If you are not one flight from home, there will also be a flight to your previous location. This allows a two leg out two leg back flight to be automated. (Out, out, back, home)

If you are flying a specified sequence the next in sequence will be an option.

Routes can be time limited with additional rewards for completing them before the due date.

Special Routes

Airline Routes

Users should have the option of setting up their own routing tables – lists of “if you are here, you can fly to here”, or using other people’s. This can supplement (tbc) the randomly generated routes (which would exclude anything in that routing table). Thus patterns flown by real airlines can be mimicked.

Sequence Routes

These are the closest things to missions. This would involve a sequence of maybe 3-5 flights to different airports. These could be generated randomly or manually, and would attract bonus XP/Salary on completion. They may also possibly have some sort of fabricated “mission” attached to them.

If your last flight was part of a sequential flight, then the “next leg” will be added to the list of the available routes.

Implementation

General

Phase 1 would be a Typescript application probably using Phaser, running on any browser. The interface would resemble an iPad very loosely, with icons or similar

indicating the various actions. All data would be stored locally, with the option of extracting the state data (name, xp, balance, location) out in an encrypted form so it can be transferred.

Phase 2 would allow the submission of this data to an encrypted hub for leaderboards and so on.

Phase 3 would move the storage of this data onto a server (if wanted) so it could be played from any location. In Phase 1 it is rather like an old console game state, you can't store your progress in the cloud.

Activities

They various activities would be accessed through touch/mouse clicks dependent on the target platform. This would include route selection and a graphical image of any selected route.

Interface with Infinite Flight

Following selection of a flight, or continuation of a sequential flight, the app would access the server to validate you are at the start airport. The app would periodically (5-10s ?) interrogate the server to check that the you have not reached the end airport and speed zero, which would be announced via audio. If no response was received for say one minute, the app would assume you have abandoned the flight, you would then be scored in experience points and paid for the trip.

Scoring

Scoring would be scaled not only by rank and flight, but by server, Casual, Training and Expert scoring by different amounts. Receiving a violation would either zero or reduce the scoring (I would have to check the violations in question). Payment and Points would be immediate and not at the end of the month.

Other possible scorings are for landings, though this has the problem of (I) interrogating the server significantly and (ii) not differentiating between manual and autoland / ILS.

Revisions

Date	Version	Notes
4/9/21	1.0	First draft