

Requirement ID	Requirement Description	Observation	Related Classes	Related Methods	Notes
		This document is intended to link the requirements, with an observation. The observations are not intended to be evidence, just an explanation on how we handled the requirement.			
		Test Passed			
		Removed Requirement			
		Test Failed			
Requirement ID	Requirement Description	Observation	Related Classes	Related Methods	Notes
F1	The game GUI shall display the game airspace and active flights. The game shall initiate aircraft entering the airspace with a static flight plan. The static flight plan will be a route passing through waypoints from the planes entry to it's exit point.	A window is displayed containing a menu screen giving the option to start a new demo game. Starting a new demo game gives the user the ability to select a difficulty. Upon selecting a difficulty the main game screen is displayed to the user. This screen contains a map of the airspace featuring waypoints and flights between these points. When a flight enters the airspace the user has the ability to click on a plane, when clicked a aircraft displays its initial route between waypoints. Aircraft automatically fly from their start waypoint to the end waypoint through a set of waypoints	Demo Aircraft	aircraft.draw demo.draw	
F2	The game shall allow the player to send orders to aircraft to immediately alter their flight plan.	When an aircraft has been selected, the user has the option to drag the part of their flight plan which meets a waypoint, to another waypoint. This alters the flight-plan of the plane and causes the plane to take a new route around the waypoints.	Aircraft Waypoint	demo.createAircraft aircraft.alterPath	
F3	There shall be at least 3 entry and 3 exit points and the exit points will correspond to given destinations.	When the waypoints are generated, four corner waypoints are added to the list. These waypoints are positioned in the corner of the airspace and act as both entry and exit points. A pair of these waypoints will serve as both the entry and exit point of the flight-path.	Aircraft Demo	demo.mousePressed demo.mouseReleased	
F4	The game shall check aircraft separation regularly.	For each plane in the air-space, the planes are checked with each other plane in the air-space to check for two things. We check first to see if the planes are close enough for a collision, and in the second check we want to see if the aircraft are within separation rules.	Demo Demo Aircraft	main.update demo.update demo.checkCollision aircraft.updateCollisions	
F5	The game shall end when the distance between one aircraft and another is lower than the specified separation rules.	For each plane in the air-space, the planes are checked with each other plane in the air-space to check for two things. We check first to see if the planes are close enough for a collision, and in the second check we want to see if the aircraft are within separation rules.	Demo Aircraft GameOver	demo.checkCollisions aircraft.updateCollisions	
F6	The game shall track an aircraft's position and speed while it is within the airspace.	Each aircraft object has speed and position vector attributes, these are unique to each plane. The position of the plane is recalculated each time we want to update the screen, taking the speed and next waypoint into consideration.	Aircraft	aircraft.position aircraft.speed	
F7	The game shall calculate the score as a function of time played and successful flights.				Removed Requirement
F8	The game shall run on a PC that meets the Windows 7 minimum system requirements (GHz processor, 2GB of RAM).	The game has been tested on a variety of devices, and has worked as desired on hardware with specifications much lower than required. The majority of development and testing was carried out on the lab terminals, and works as desired. The mouse is used to navigate the menu screen, selecting planes and dragging their paths to new waypoints. The mouse-wheel is used for toggling the primary layer displayed and the right mouse button is used for deselecting planes. The left and right keys are used in manual control for steering the aircraft, space bar can be used to enter manual mode, and escapes can be used to exit to the menu.			
F9	The game shall allow the use of the keyboard and mouse as input devices.	When selected to play a demo game the user is prompted with a selection of difficulties to choose from. Increasing the difficulty of the game will increase the speed of the aircraft and increase the separation requirement between them.	Demo DifficultySelect Title	demo.mousePressed demo.mouseReleased demo.keyPressed demo.keyReleased	
F10	The game shall allow for separation rules of aircraft to scale against difficulty.	When the game is started, the user is presented with fourteen waypoints, of these waypoints, four are exiently points situated in the corners of the airspace. The remaining ten waypoints are scattered around the airspace, the position of these points is fixed and is the same every time the user plays.	Demo DifficultySelect	difficultySelect.start	
F11	The game shall provide at least ten fixed waypoints within the airspace.	The difficulty of the game will vary the speed in which aircraft will ascend and descend between altitude layers. A higher difficulty game will have a slower transition between altitude layers.	Demo Waypoint		
F12	The game shall allow aircraft to vary the rate at which they climb / descend.		Aircraft DifficultySelect	difficultySelect.start	
F13					
Requirement ID	Requirement Description	Observation	Related Classes	Related Methods	Notes
NF1	The game shall incorporate realism where possible/appropriate, but its primary goal is to provide a fun and enjoyable experience to the user.	The game tries to emulate the controls of a real air traffic controller, the game is to a less realistic fast pace. The user is provided with an entertaining backstory and environment. The game is fast paced and keeps the user wanting to see the longest they can survive for.			
NF2	The game shall be engaging.	Pen European Game Information (PEGI), have content guidelines that they use for determining the age rating of a video-game, these are available online. We feel our game has next to none of the elements which could lead to a higher age rating. We feel our game is suitable for the student demographic.			
NF3.1	The game will be appropriate for this market in terms of the graphics and language used. Target market being students.	The game looks similar to that of a real air traffic controller's monitor, however we have used basic plane images and simple circles for the waypoints. We have a light semi-realistic background image and particle effects are used when two aircraft collide.			
NF4	The graphics of the game shall not be restricted by reality, but will instead aim to provide a fun and engaging experience for the user.	A frames per second counter is displayed in the title bar of the game, from all of our tests we have never witnessed this value go below 50fps. This game is capped at 60 frames per second, the majority of the time the application will run at this framerate.			
NF5	The game GUI shall be updated regularly. The target for this is every 30ms.	The main menu displays a button labeled "help" which opens the user's web browser to display our website with simple to understand user documentation.	Window	Window.update	
NF6	The game shall be easy to play and clear instructions will be available when needed.	Mouse actions can be easily adapted for touch screen interfaces, all of user actions can be carried out with the mouse. Every keyboard control can be duplicated by clicking on certain areas of the screen. In the case of changing the displayed altitude, this currently uses the mouse wheel, but because of the nature of the action, a figure's gesture could be used.	Title		
NF7	The game shall be adaptable for touch input.	No sizes or units are given for any distances between planes. Relative to the size of the icons used to represent the aircraft, the separation rules are unrealistic, however it's important we can see the aircraft clearly. The waypoints are assumed to be airports, the separation radius relative to the distances between these is more realistic.			
NF8	The game shall use semi-realistic separation rules.	The game doesn't show the position of the aircraft as values to the user, the altitude of the current selected layer is given at the top of the user interface in feet. Each aircraft has a label indicating their altitude, this will be one of the two altitude layer values, this again will be in feet. The separation between planes is shown as a visual indicator with no numbers given.	Aircraft	aircraft.updateCollisions	
NF9	The game shall use units which fit the air traffic control domain, but which are not so obscure as to confuse the user.		Aircraft		