

PANTALLA DE AEROPUERTO**1. Especificación de Requisitos Funcionales.**

Nombre	R.F# 1. Generar un listado de vuelos
Resumen	El Sistema debe realizar un listado de vuelos generados aleatoriamente con diferentes fechas, horarios, diferentes aerolíneas, números de vuelos, ciudades de destino y puertas de embarque.
Entradas	
Número de vuelos.	
Resultados	
Listado de vuelos.	

Nombre	R.F# 2. Buscar un vuelo
Resumen	El programa buscará un vuelo predeterminado con determinados criterios de búsqueda que escogerá el usuario.
Entradas	
Vuelo que desea encontrar.	
Resultados	
El vuelo buscado.	

Nombre	R.F# 3. Pagar vuelos
Resumen	El programa debe mostrar los vuelos en pantalla mediante el control Pagination de javaFX, en caso de que no se permitan visualizar en pantalla
Entradas	
Ninguno.	
Resultados	
Subdividir la lista de los vuelos en páginas.	

Nombre	R.F# 4. Mostrar información de los vuelos
Resumen	El Sistema mostrara toda la información de los vuelos correspondientes en una tabla.
Entradas	
Ninguna	
Resultados	
El listado de vuelos.	

Nombre	R.N.F# 1. Ejecutar el programa con JavaFX.
Resumen	Utilizar JavaFX con Scene Builder para crear la interfaz gráfica de usuario.
Entradas	
Ninguna	
Resultados	
La interfaz grafica.	

Nombre	R.N.F# 1. Permitir que la pantalla sea interactiva
Resumen	El listado de los vuelos será interactivo, no solo los mostraran, si no que el usuario podrá modificarlos a su criterio
Entradas	
Ninguna	
Resultados	
Pantalla interactiva de acuerdo a los criterios establecidos por el usuario.	

1. Diseño

Diagrama de clases

2. Diseño de casos de prueba

Diseño de pruebas

3. Tabla de trazabilidad de requerimientos vs métodos

Requerimientos Funcionales	Metodo	Clase
R.F#1.	generateSimulation() getText() Simulation() generateSimulation() load() addListAirlines() addListCities() searchListAirline() searchListCities() PATH_AIRLINES PATH_CITIES Flight() DateFlight() Time()	ui/ FlightController ui/ FlightController model/Simulation model/Simulation model/Simulation model/Simulation model/Simulation model/Simulation model/Simulation data/airlines data/cities model/Flight model/DateFlight model/Time
R.F#2	searching() sortByAirline() sortByDestination() sortByGate() sortByDate() sortByTime() sortByCode() getSchedule() getDate() getNameAirline() getIdAirline() getDestinationCity() getBoardingGate() toString() toString()	ui/ FlightController model/Simulation model/Simulation model/Simulation model/Simulation model/Simulation model/Simulation model/Flight model/Flight model/Flight model/Flight model/Flight model/Flight model/DateFlight model/Time
R.F#3	generateSimulation() createPage()	ui/ FlightController ui/ FlightController
R.F#4	generateSimulation() createTable() createData() getFlightsToArray() createPage()	ui/ FlightController ui/ FlightController ui/ FlightController model/Simulation ui/ FlightController



Algorithms and Programming II

Unit Laboratory 5 - 2019-1