**Requerimientos funcionales**

|  |  |
| --- | --- |
| Nombre | RF#1 Mostrar la información de los vuelos |
| Resumen | Mostrar la información de un vuelo correspondiente a: hora de salida, aerolínea a la que pertenece el vuelo, código del vuelo, destino y puerta de embarque. |
| Entradas | Ninguna |
| Salidas | Información de los vuelos |

|  |  |
| --- | --- |
| Nombre | RF#2 Generar un listado aleatorio de vuelos |
| Resumen | Se debe poder generar un aleatorio de vuelos con información aleatoria. Por defecto el programa muestra un listado aleatorio y se debe poder generar uno nuevo cada vez que el usuario lo pida. |
| Entradas | Número de vuelos |
| Salidas | Listado nuevo de vuelos |

|  |  |
| --- | --- |
| Nombre | RF#3 Buscar un vuelo |
| Resumen |  |
| Entradas | Código de vuelo |
| Salidas | Si el código de vuelo existe, muestra el vuelo correspondiente, de lo contrario muestra un mensaje para informar que el código ingresado no corresponde a ningún vuelo actual. |

|  |  |
| --- | --- |
| Nombre | RF#4 Ordenar los vuelos |
| Resumen | Se debe poder ordenar los vuelos por los diferentes criterios (hora de salida, nombre de aerolínea, código de vuelo, destino y puerta de desembarque), el criterio debe ser escogido por el usuario. Por defecto el programa ordena los vuelos por hora de salida. |
| Entradas | Criterio de ordenamiento |
| Salidas | Vuelos ordenados |

**Requerimientos no funcionales**

|  |  |
| --- | --- |
| Nombre | RNF#1 Uso de JavaFX para la interfaz gráfica. |
| Resumen | Se hace uso de las clases de JavaFX y el software de SceneBuilder para construir la interfaz gráfica que se muestra al usuario. |

**Diagrama de clases**

**Trazabilidad**

|  |  |  |
| --- | --- | --- |
| Requerimiento | Método(s) | Clase(s) |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |