**VUELA FÁCIL – SPRINT2**

**Integrantes:**

**Katherin Reyes Deantonio**

**Wilmer Orlando Daza Hernández**

**Airyance Mazo Ruidiaz**

**Daniel Barrera Barrera**

**Carlos Arturo Rodriguez Martínez**

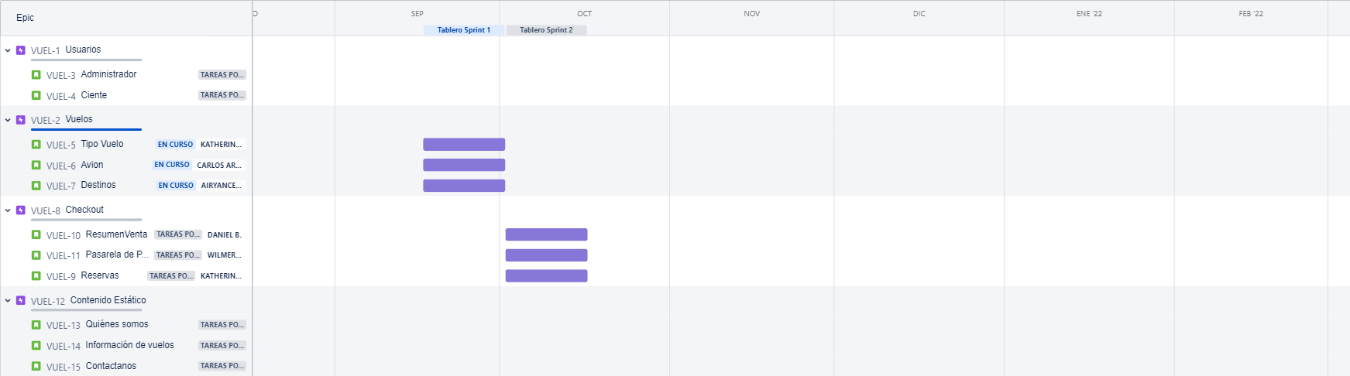
**Introducción:**

El proyecto consiste en construir una aplicación web que permita al usuario de una aerolínea llamada Vuela Fácil tener una mejor experiencia de usuario a la hora de comprar, reservar o cancelar sus vuelos, como también para consultar todo tipo de información y servicios que la aerolínea ofrece. Para realizar este aplicativo se hace uso de la metodología ágil SCRUM por medio del entorno JIRA, con el cual se plantea el esquema general del proyecto a realizar mediante las incidencias epic, y se diseñan los Sprints para llevar a cabo las tareas de cada uno de estos en un tiempo definido. A continuación, un esquema general del proyecto junto con el planteamiento del primer Sprint.

**Roles:**

* *Dueño del proyecto:* Katherin Reyes Deantonio, Wilmer Orlando Daza Hernández, Airyance Mazo Ruidiaz, Daniel Barrera Barrera y Carlos Arturo Rodriguez Martínez.
* *Scrum Master:* Juan David Argüello Plata / Carlos Arturo Rodríguez Martínez.
* *Desarrolladores:* Katherin Reyes Deantonio, Wilmer Orlando Daza Hernández, Airyance Mazo Ruidiaz, Daniel Barrera Barrera y Carlos Arturo Rodriguez Martínez.

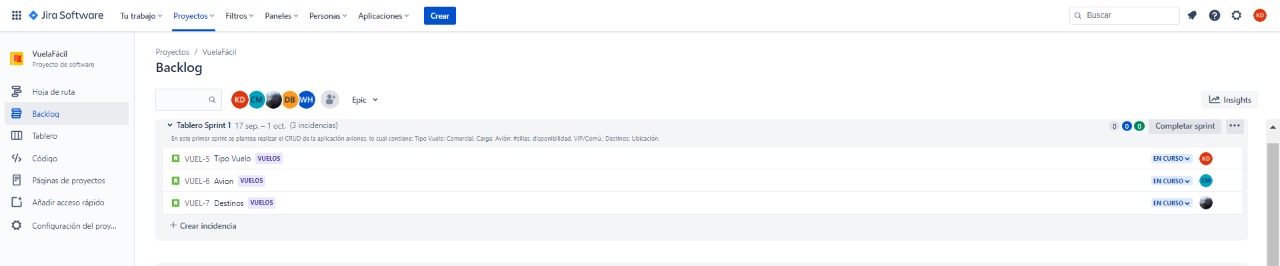
**1. Ruta de trabajo**

****

En los objetivos primarios presentados en el primer sprint se plantea la elaboración del paquete usuario y de vuelos, así, se plantea asignarles una incedencia de tipo epic en las que se desarrollan los modelos internos de cada paquete. Para el paquete *Usuarios* se dividirán las tareas para el desarrollo del usuario tipo Administrador y el usuario tipo Cliente, como los dos perfiles principalmente necesarios para la aplicación. De ese modo, para el paquete *Vuelos* se asignan tareas divididas en Tipo de vuelo, Avión y Destinos, como los elementos fundamentales para llevar a caba la lógica de la aplicación evocada a vuelos.

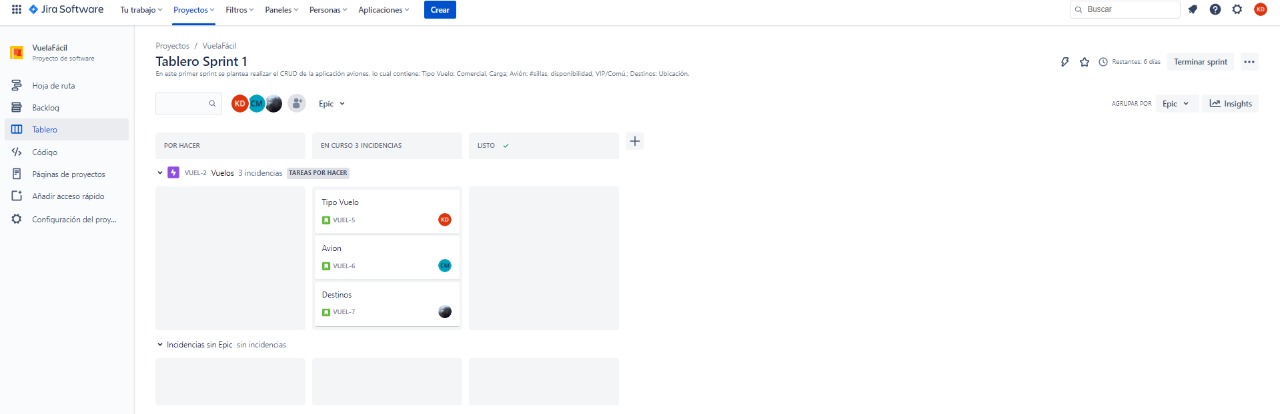
En los objetivos especificados en el anterior Sprint, se plantea la parte de interfz de usuario y las herramientas técnicas para elaborar el proyecto, dado que parte de esto ya se ha elaborado con las anteriores incidencias, se cubrirá lo que falta con las siguientes incidencias: *Checkout* y el *Contenido Estático*, las cuales requieren una interacción visual para con el usuario y el uso de todas las herramientas del FrontEnd y el BackEnd. Las tareas planteadas son Reservas, ResumenVenta y Pasarela de pago para el checkout, y Quiénes somos, información de vuelos y Contactanos serán las tareas destinadas a cumplir con el contenido estático.

**2. Backlog**



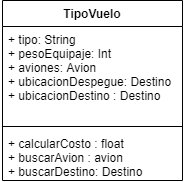
Dentro de la planificación de este sprint cabe desarrollar las tareas relacionadas con el paquete *Vuelos,* dado que son una parte de la creación de las clases CRUD del BackEnd, se pueden trabajar con Python-Django. Así se pueden realizar las primeras pruebas de funcionamiento sobre la lógica de las clases relacionada a los vuelos, por ejemplo; puede haber dos tipos de vuelos como los son el vuelo comercial y el de carga. Los aviones se pueden definir en base a sus características más importantes como lo son el número de asientos, su disponibilidad, VIP/Común, etc… y por último los destinos a los cuales los aviones aterrizan o despegan, con sus respectivas características. Estas pruebas de funcionamiento inicial pueden hacerse con ayuda del Python Shell.

**3. Tablero de trabajo**

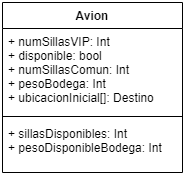


Para cada tarea particular, vamos a especificar las diferentes subtareas. Estas tendrán una plantilla de inicion bastante simple, con la posibilidad de que en la mediad de que el proyecto progrese, se preste para mejorar.

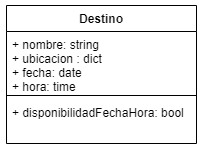
* TipoVuelo:
  + Contará con los siguientes elementos:
    - Atributos: Tipo, pesoEquipaje, aviones, ubicacionDespeque, ubicacionDestino.
    - Métodos: calcularCosto, buscarAvion, buscarDestino



* Avion:
  + La clase avion contará con:
    - Atributos: numSillasVIP, numSillasComun, disponible, pesoBodega, ubicacionInicial[]
    - Métodos: sillasDisnponibles, pesoDispobibleBodega,



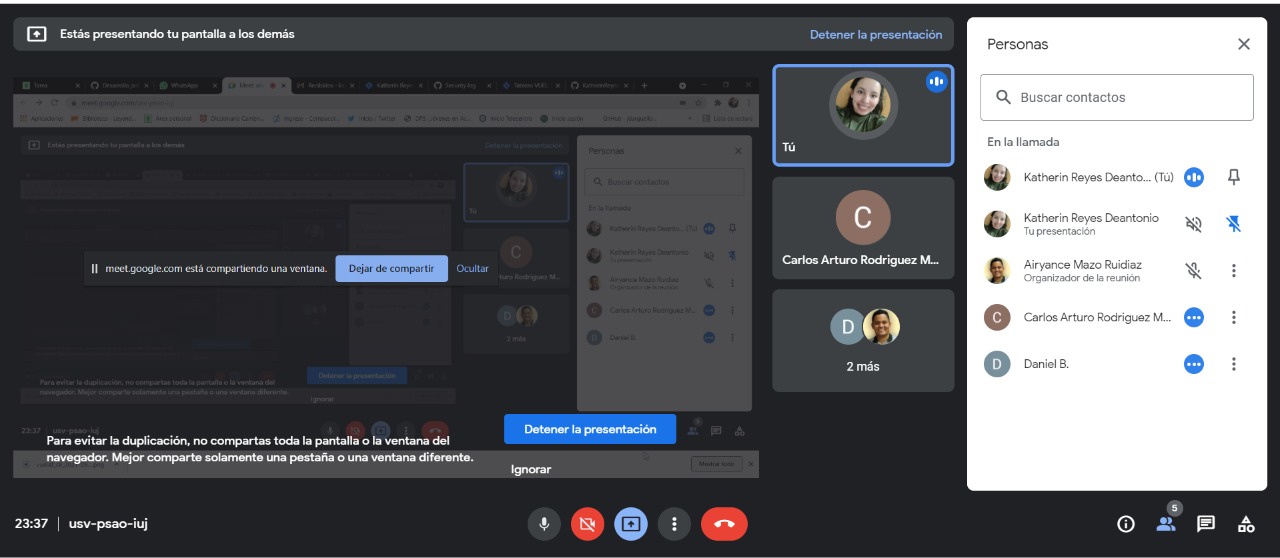
* Destinos:
  + La clase destinos tendrá los siguientes elementos:
    - Atributos: ubicación, capacidadVuelosDiaria, fecha, hora
    - Métodos: disponibiliadadFechaHora



**Nota:** por el momento se contemplan únicamente estas clases para el paquete vuelos, la lógica se plantea de tal manera que el Destino equivale al aeropuerto o lugar donde despegan y aterrizan aviones. El avión debe contar con la información de dónde estará en determinada fecha, junto con las características necesarias para saber cuántos pasajeros puede llevar. Uniendo esos conceptos se plantea que el tipo de vuelo se calcule el precio del vuelo acorde a la información disponible.

**Anexos:**

**Reuniones de equipo:**

****