



## **INF-225: Ingeniería de Software**

### **Entregable 3**

**John Rodríguez**  
201573002-4

**Javier Zavala Torres**  
201573004-0

**Pablo Flores Repetto**  
201573001-6

## **1. Requisitos clave (Final)**

### **1.1. Requisitos funcionales**

| <b>Requisito</b> | <b>Descripción y Medición</b>   |
|------------------|---|
| FR1              | [Funcional] Descargar datos financieros desde yahoo finance.<br>Se mide la velocidad con que se descargan los datos y si están correctos.           |
| FR2              | [Funcional] Permitir que usuario suba su propio archivo csv con datos de acciones.<br>Se verifica que el archivo se suba correctamente.             |
| FR3              | [Funcional] Estimar el precio de una opción bajo parámetros dados.<br>Se analiza si el valor tiene sentido y es coherente con los parámetros dados. |
| FR4              | [Funcional] Mostrar gráfico acción vs tiempo.<br>Se verifica que el gráfico represente de manera correcta los datos.                                |

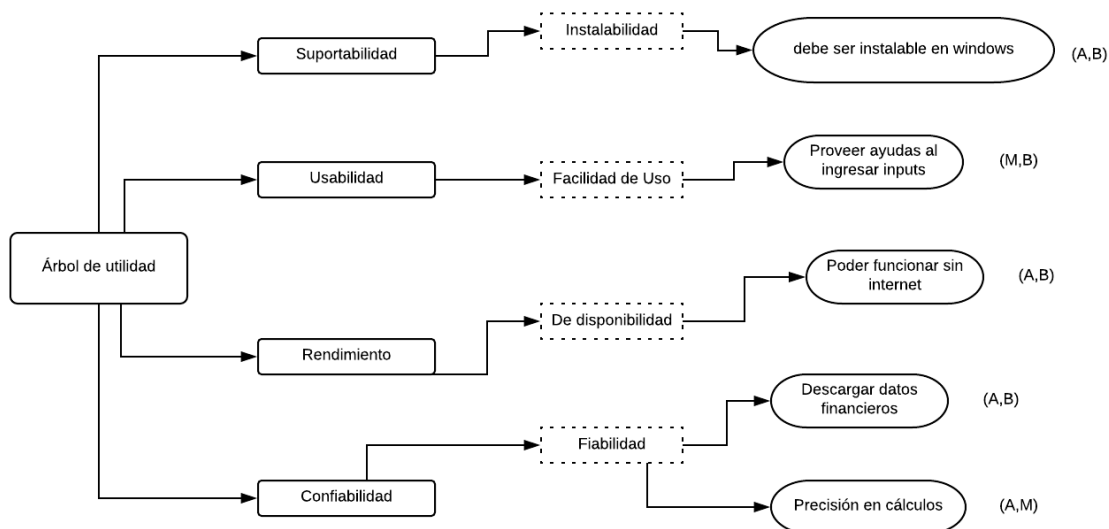
### **1.2. Requisitos extra-funcionales**

| <b>Requisito</b> | <b>Descripción y medición</b>   |
|------------------|---|
| EFR1             | [Interfaz] Software debe ser una aplicación de escritorio.<br>Se verifica que la aplicación se ejecuta correctamente como una aplicación de escritorio en el sistema operativo Windows. |
| EFR2             | [Fiabilidad] Precisión en los cálculos, minimizando errores numéricos.<br>Se comprueba que operaciones no pierdan significancia por overflow u otras imprecisiones.                     |
| EFR3             | [Extensibilidad] Sistema operativo: linux, windows o android.<br>Se testeará el funcionamiento de la aplicación en el sistema operativo escogido, ya sea linux, windows o android.      |
| EFR4             | [Disponibilidad] Poder funcionar offline.<br>Se medirá testeando que funcione sin conexión alguna a internet y con conexión wifi.   |

## **2. Árbol de Utilidad (Final)**

Para el árbol de utilidad, se reconocieron los siguientes atributos de calidad presentes en el software:

- **Suportabilidad:** Que sea instalable en windows, es un escenario de alta prioridad porque es un requisito funcional del software, y es algo no difícil de lograr, pues basta con generar un ejecutable del programa
- **Usabilidad:** Proveer ayuda al ingresar input es un escenario de mediana prioridad ya que es importante que no se dificulte el uso del software para el cliente, pero no es de vital importancia conseguirlo, y también es algo no difícil de lograr, ya que basta con poner las ayudas en la interfaz
- **Rendimiento:** que el software funcione sin internet es un escenario de prioridad alta ya que es un requisito extra funcional del software, y es algo fácil de implementar, ya que sólo se tiene que agregar la opción de importar los datos desde, por ejemplo, un csv enés de la API
- **Confiabilidad:** Descargar datos financieros, es un escenario de prioridad alta, ya que es vital para el funcionamiento del software y además es un requisito funcional, y es algo fácil de lograr ya que basta con implementar la conexión a la api y usar sus funciones para obtener los datos. Precisión en los cálculos es de prioridad alta ya que es necesario entregar el resultado de la opción con el menor error posible, y además de que es un requisito no funcional, y es algo medianamente difícil de conseguir, ya que hay que trabajar con el algoritmo que se usa para el cálculo de la opción, para ver como reducir el error, ya sea usando algún paquete numérico o investigando como se puede lograr



### 3. Modelo de Software (Final)

#### 3.1. Modelo de dominio

Las siguientes entidades son las que se reconocen en el modelo de dominio:

- **Usuario:** Persona que compra o vende acciones.
- **Acción:** Una acción.
- **Opción:** Derecho a la compra o venta de una acción luego de pasado un tiempo de madurez, al precio que se acordó.

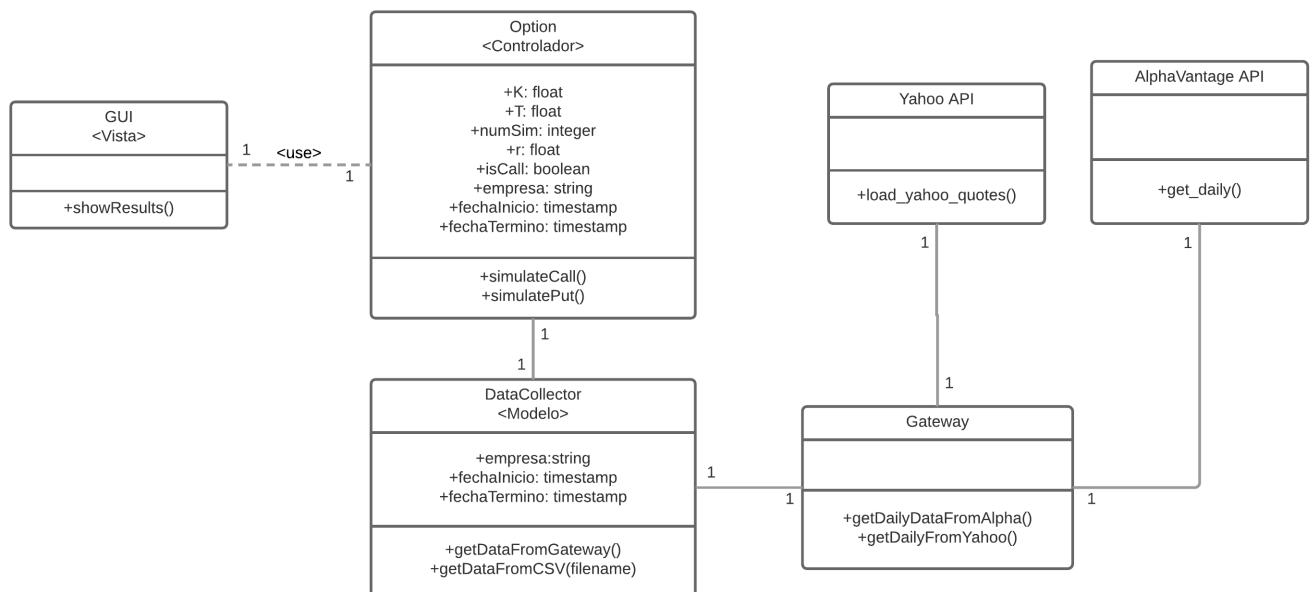
- Empresa: Una compañía con acciones.

Las cuales se relacionan de la siguiente forma:



### 3.2. Diagrama de clases

El software se ha desarrollado con la siguiente estructura:



Cada clase <nombre\_clase> mostrada en el diagrama esta dentro de su <nombre\_clase>.py. Tenemos las siguientes clases:

- GUI: Representa la vista de nuestro software. Depende de los resultados entregados por el controlador.
- Option: Controlador de la vista. Se encarga de realizar la simulación con los datos que obtiene del modelo. Pasa los resultados a la vista.
- DataCollector: Modelo. De aquí se obtienen los datos para realizar la simulación.
- Gateway: Obtiene los datos de las acciones de la API de Yahoo o de Alpha Vantage, y se los entrega al DataCollector.

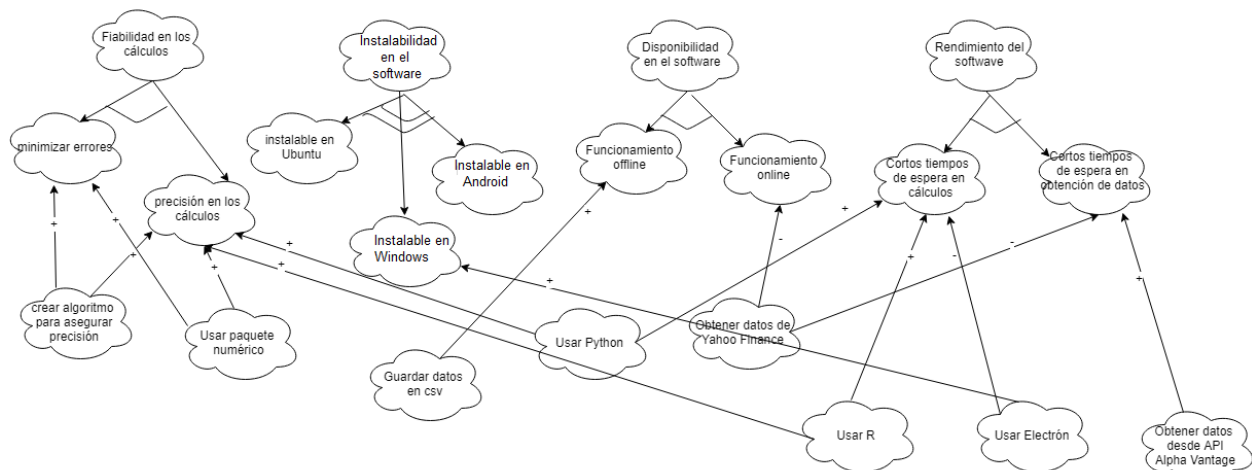
Y se tienen las siguientes API's:

- Alpha Vantage: API para obtener datos financieros. Decidimos incluirla debido a que es más rápida y mucho más estable que la obtención de datos a través de Yahoo Finance.
- Yahoo: No es una API en si pues la API de Yahoo ya no está funcionando. Es un script que hace web scraping al sitio de Yahoo Finance.

### 3.3. Patrones de diseño utilizados

| Intención  | Patrón de diseño | Razonamiento  |
|--|------------------|---|
| Mostrar los resultados del cálculo de las opciones | MVC              | MVC permite modularizar las distintas responsabilidades y mantener el código más ordenado y mejor gestionado. |
| Separar el acceso a yahoo finance                  | Gateway          | Gateway permite separar el acceso a recursos externos, que en este caso son la API de Alpha Vantage y Yahoo   |

## 4. Trade-offs



Para minimizar los errores y aumentar la precisión en los calculos, decidimos crear un algoritmo que nos ayudara a asegurar la precisión. Además utilizamos paquetes numéricos, puesto que es código muy probado y que nos permite disminuir los errores de cálculo.

El cliente nos solicito que la aplicación pudiera ser instalada en windows, ubuntu o android. Como equipo decidimos desarrollar la aplicación para windows y ubuntu, por que son las plataformas con las que estamos más familiarizados. Esto nos llevo a escoger tecnologías para estas plataformas.

Otro aspecto importante para el cliente es las diponibilidad del software. Para poder cumplir con esto implementamos la obtención de datos desde un csv, para que funcione de manera offline, y la obtención de datos desde Yahoo finance, puesto que el cliente solicito explicitamente que obtuvieramos los datos desde este sitio web. Sin embargo la API de Yahoo finance está descontinuada, por lo que debemos obtener los datos haciendo Web scraping, lo que no siempre funciona. Por esta razón decidimos añadir la API Alpha Vantage que también cuenta con una base de datos de las acciones de varias empresas. Esto nos permite mantener la aplicación siempre disponible.

El último aspecto importante para el cliente es el rendimiento. Para poder disminuir los tiempos de cálculos utilizamos R, puesto que es un lenguaje diseñado para hacer este tipo de cálculos y por lo tanto está muy bien optimizado. Para disminuir los tiempos en la obtención de datos utilizamos Alpha Vantage, puesto que es más rápido que hacer web scraping.

| Decisión          | Softgoal                 | Evaluación | Razonamiento  |
|-------------------|--------------------------|------------|---|
| R                 | Cortos tiempos de espera | ++         | Los paquetes de R nos permiten tener un mayor rapidez en los cálculos en comparación con Python, puesto que R está diseñado para este tipo de cálculos. |
| API Alpha Vantage | Funcionamiento Online    | ++         | La API Alpha Vantage tiene mayor probabilidad de estar disponible que hacer web scraping a la página de Yahoo Finance.                                  |
| Electron          | Instalabilidad           | +          | Electron permite compilar el código para distintos sistemas operativos.   |

## 5. Deuda Técnica

Las siguientes decisiones de desarrollo conllevan una deuda técnica:

| Ítem deuda técnica                               | Razonamiento   | Impacto  |
|--|--|--|
| GUI desarrollada en Python con el framework kivy | Desarrollar GUI en Python es más fácil que otras alternativas, como Electron | GUI no es agradable a la vista y software no es multiplataforma. |
| Uso de Python para la simulación y no R          | Tenemos mayor experiencia con Python   | Menor performance pues R es más rápido en cálculos estadísticos  |