



Departamento de Informática
Universidad Técnica Federico Santa María



Requisitos y Versión Inicial del Software

Proyecto: Valoración de Opciones sobre Acciones

Equipo: Hurricane Operation

Nombres y Apellidos	Email	ROL USM
Sebastián Gallardo Díaz	sebastian.gallard.14@sansano.usm.cl	201410006-K
Macarena Andrade Oyarzún	macarena.andradeo@sansano.usm.cl	201573067-9
Bastían Quezada Muñoz	bastian.quezada@sansano.usm.cl	201573033-4

Contexto del proyecto

En el mundo de las finanzas, la volatilidad de las acciones es impredecible, por lo cual es difícil saber a ciencia cierta si es que es conveniente invertir en un mercado o no. Para enfrentar esto, es por lo que existen las opciones sobre acciones. Estos son acuerdos en los cuales comprador y vendedor se comprometen a comercializar acciones a precio fijo en un plazo definido.

Yahoo! Finance es un medio el cual provee información financiera de empresas estadounidenses. Nos basaremos en este servicio web para lograr el objetivo del software.

Objetivo de proyecto

Nuestro objetivo es valorar opciones financieras sobre acciones automáticamente para decidir rápida y eficazmente la compra o venta de una acción asociada.

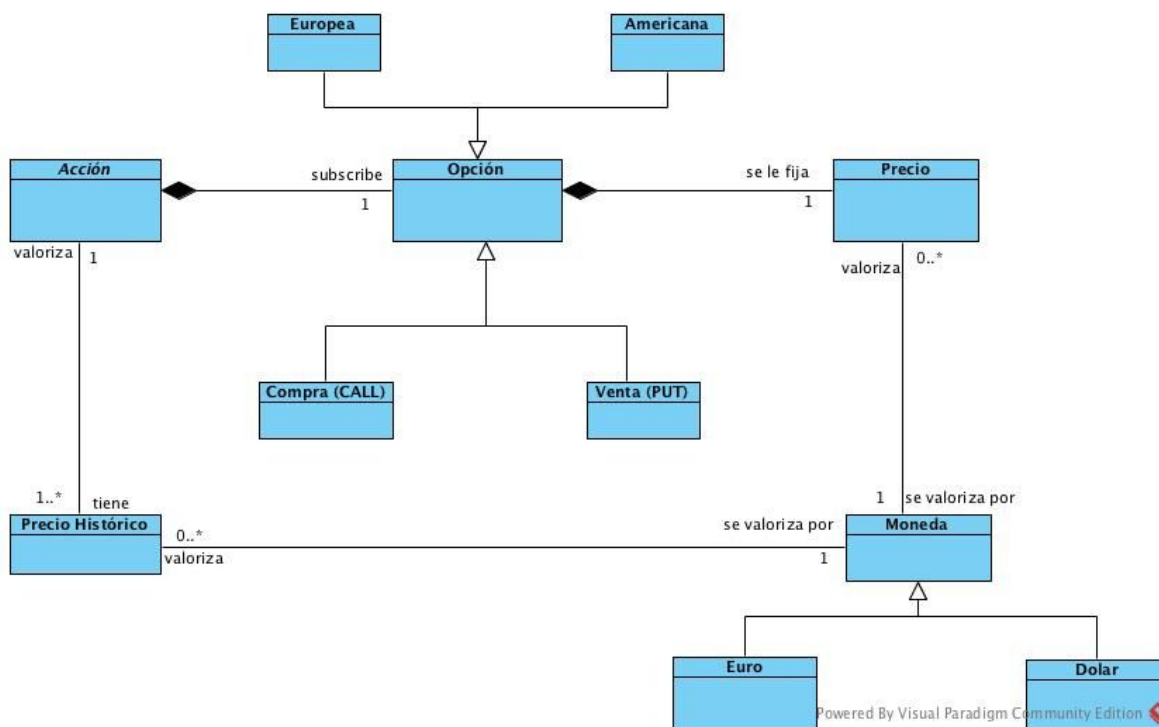
Resumen del Proyecto

Nuestro problema será obtener opciones sobre las acciones de un mercado. Para esto, nuestro software recibirá como dato de entrada la empresa a analizar y la fecha en cual queremos evaluar la acción. Éste entregará como resultado la estimación de la opción calculada para el tiempo dado. Además, se utilizará la API de Yahoo! Finance u otra (dependiendo de la disponibilidad de esta) para obtener los datos financieros de las empresas. Para verificar si el resultado es correcto, se realizarán varias pruebas comprobando si el resultado obtenido coincide o no con el que indica la página en la fecha dada.

Tomaremos en cuenta los dos tipos de opciones, Americana y Europea. La Americana puede ser ejercida a lo largo del tiempo transcurrido hasta la fecha cotizada, en cambio la Europea sólo puede ser ejercida en la fecha cotizada. Utilizaremos el método de Black-Scholes para calcular el valor estimado.

Para validar si se cumple el objetivo del software, se le permitirá su uso al usuario y luego de un tiempo se les hará una encuesta para saber si está satisfecho con este, así, se utilizará el feedback para hacer las mejoras solicitadas.

Modelo de Dominio (Inicial)



Entidad	Descripción (1 línea)
Acción	Unidad de propiedad en una empresa que se puede poner a la venta
Opción	Contrato que da derecho al comprador o vendedor para comprar/vender una acción a un precio fijado anteriormente.
Precio	Valor fijado de la opción, en alguna moneda
Moneda	Unidad básica de forma de pago, aceptable como medio de cambio
Precio Histórico	Precio anterior de una acción (que no corresponde al periodo actual)
Compra (CALL)	Tipo de Opción, correspondiente a derecho a compra de una acción
Venta (PUT)	Tipo de Opción, correspondiente a derecho a venta de una acción
Euro	Tipo de Moneda, válida en la Unión Europea.
Dólar	Tipo de Moneda, válida en los EE.UU.
Opción Americana	Tipo de Opción, la cual puede ser ejercida en un tiempo menor o igual al tiempo de madurez
Opción Europea	Tipo de Opción, la cual puede ser ejercida solamente al terminar su tiempo de madurez.

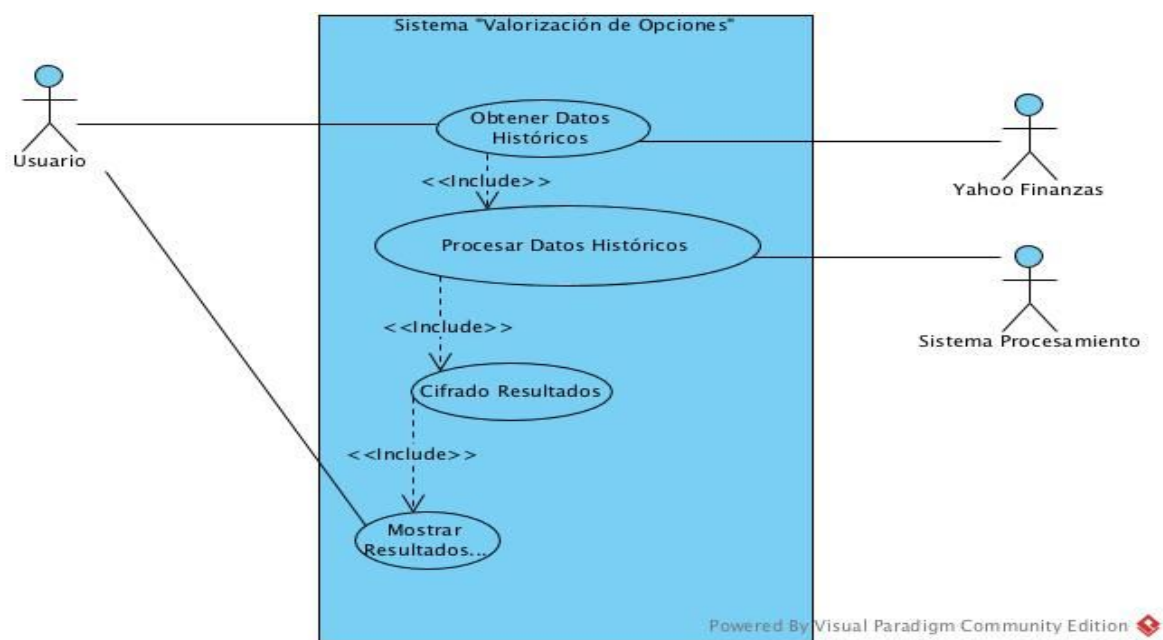
Requisitos clave funcionales y extra-funcionales

Req. funcional	Descripción y medición
FR01: Obtención de datos financieros históricos de las acciones	El sistema debe obtener datos financieros de acciones desde Yahoo Finanzas o similar, a partir de parámetros de tipo de acción, categoría, tasa libre de riesgo y tiempo de madurez. Se valida revisando los outputs respecto a los inputs descritos.
FR02: Aplicación de modelos matemáticos a datos financiero.	El sistema debe aplicar el método de Black Scholes y sus ecuaciones a los datos obtenidos, además de obtener los resultados de la aplicación de este método. Se valida si el método recibe los datos históricos y entrega el precio de compra/venta de la opción que se estudia.
FR03: Despliegue de información obtenida y gráficos	El sistema debe mostrar la información del precio estimado de la opción de acuerdo al método aplicado, así como gráficos de volatilidad de la acción y de interés del precio. Se valida con la visualización clara de la información obtenida.
FR04: Subir archivo CSV de datos históricos	El sistema debe permitir al usuario cargar un archivo CSV o similar con los datos históricos de cierta acción. Se valida con pruebas sobre archivos CSV y su correcta lectura por parte del sistema.
FR05: Guardar resultados finales.	El sistema debe guardar los resultados finales obtenidos en una base de datos, para tener un registro de opciones consultadas. Se valida revisando los registros de la base de datos.

Req. extra-funcional	Descripción y medición
NFR01: Uso del lenguaje estadístico R	Se debe usar el lenguaje estadístico R en la aplicación del modelo matemático a los datos históricos. Se valida revisando la tecnología de implementación del modelo.
NFR02: Desempeño del procesamiento de datos	El sistema debe obtener los resultados de la aplicación del método en un tiempo adecuado (menor a 30 [s]). Se valida midiendo el tiempo total de obtención de datos.
NFR03: Aplicación de Escritorio	El sistema se debe ejecutar en un ambiente no Web (Software Ejecutable StandAlone). Se valida con la entrega misma del ejecutable.
NFR04: Seguridad de los Datos	El sistema debe cifrar los resultados obtenidos durante el proceso, hasta la visualización de ellos. Se valida con pruebas de acceso a los datos, los cuales no deben ser leídos con facilidad.

NFR05: Exactitud	El sistema debe entregar los resultados con un mínimo de 3 cifras significativas para los campos numéricos. Se valida observando la exactitud del resultado.
NFR06: Integridad de los datos	Los datos del sistema deben ser correctos y completos. Se valida verificando que la información obtenida es suficiente para ejercer una opción o no.

Casos de Uso y Diagrama



Nombre	Obtener Datos Históricos
Actores	Usuario, Sistema Proveedor de Datos históricos
Pre-condiciones	1. El sistema proveedor de datos históricos está operativo
Post-condiciones	El sistema obtiene los datos históricos de la acción ingresada, listos para procesar.
Flujo normal	<ol style="list-style-type: none"> 1. El usuario ingresa Fecha de compra de la acción, nombre de la acción, categoría (Compra o Venta), tipo (Europea o Americana), y tasa libre de riesgo. 2. El sistema consulta al Sistema Proveedor de datos históricos por el nombre de la acción, fecha de inicio y

	<p>fecha de fin (fecha de hoy), los cuales definen el intervalo de obtención de datos.</p> <p>3. El Sistema Proveedor de datos históricos envía los datos históricos de la acción ingresada, entre las fechas ingresadas (Fecha, Precios High, Low y Close, Precio ajustado, y volumen).</p>
Cursos alternativo	<p>1 El usuario ingresa una tasa libre de riesgo inválida (mayor a 1 o menor a 0).</p> <p>1.1. El sistema informa al usuario que ingrese una tasa entre 0 y 1.</p> <p>3. El sistema proveedor de datos históricos no encuentra datos de la acción ingresada.</p> <p>3.1. El sistema proveedor de datos históricos envía una respuesta vacía al sistema</p> <p>3.2. El sistema informa al usuario que no se tienen datos de la acción ingresada.</p>

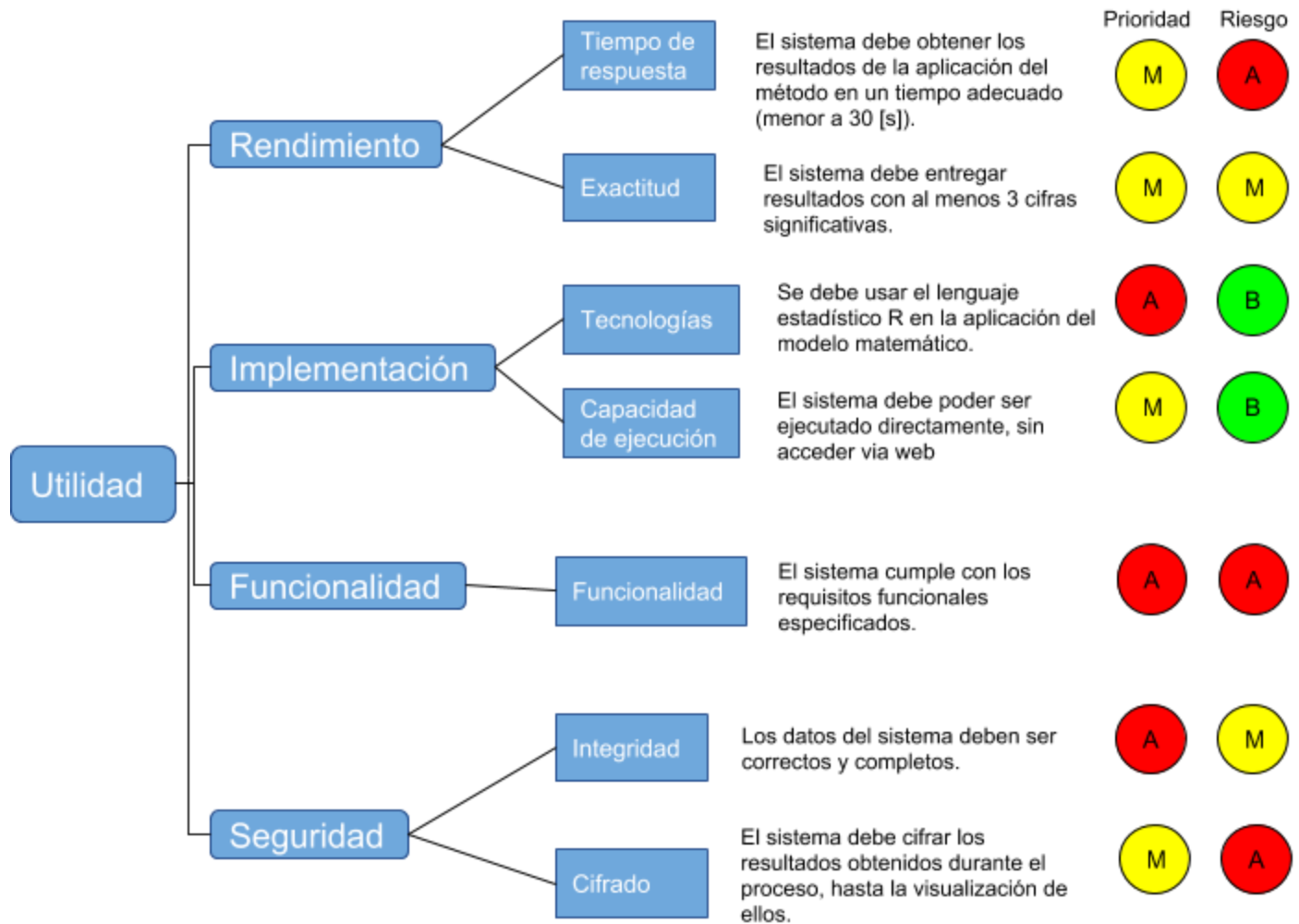
Nombre	Subir Archivo Datos Históricos
Actores	Usuario
Pre-condiciones	1. El usuario cuenta con un archivo CSV formateado que el sistema acepte.
Post-condiciones	El sistema lee los datos del archivo CSV y los tiene listos para procesar.
Flujo normal	<p>1. El usuario ingresa Fecha de compra de la acción, nombre de la acción, categoría (Compra o Venta), tipo (Europea o Americana), y tasa libre de riesgo.</p> <p>2. El usuario sube al sistema su archivo CSV.</p>
Cursos alternativo	<p>1. El usuario ingresa una tasa libre de riesgo inválida (mayor a 1 o menor a 0).</p> <p>1.1. El sistema informa al usuario que ingrese una tasa entre 0 y 1.</p> <p>2. El usuario sube un archivo inválido o mal formateado.</p> <p>2.1. El sistema informa que debe subir un archivo válido insistiendo en el formato de éste.</p>

Nombre	Procesar Datos Históricos
--------	---------------------------

Actores	Sistema Procesamiento
Pre-condiciones	1. El sistema cuenta con datos históricos y con todos los parámetros que se necesitan para obtener los datos.
Post-condiciones	El sistema de procesamiento entrega los resultados obtenidos.
Flujo normal	<ol style="list-style-type: none">1. Se hace entrega de los datos históricos y los parámetros al sistema de procesamiento.2. El sistema de procesamiento hace los cálculos necesarios.3. El sistema recibe los resultados..
Cursos alternativo	<ol style="list-style-type: none">2. El sistema de procesamiento no puede calcular la estimación con los datos dados.<ol style="list-style-type: none">2.1 Se informa que no se es capaz calcular en este momento el precio de la acción solicitada.

Priorización de requisitos

Para priorizar los requisitos se hará uso del árbol de utilidad.



Analizando el árbol hemos llegado a la siguiente conclusión con respecto a la priorización:

1. Funcionalidad.
2. Tecnologías.
3. Integridad.
4. Capacidad de ejecución.
5. Exactitud.
6. Tiempo de respuesta.
7. Cifrado.

Estimación

Para la estimación del esfuerzo utilizaremos el modelo de COCOMO. Aplicaremos el sub-modelo Intermedio y consideramos este proyecto del tipo Semi-Acoplado, ya que a pesar de no tener tanta complejidad, no tenemos mucho conocimiento del dominio del problema y por lo tanto, no poseemos tanta experiencia en este ámbito.

La fórmula del esfuerzo es:

$$E = a * KLDC^{\{e\}} * FAE$$

Donde:

- KLDC: Kilo Líneas de Código
- FAE: multiplicación de los valores evaluados en los diferentes 15 atributos.
- a, e: coeficientes predeterminados según el tipo de proyecto, donde este caso serían a=3.0 y e=1.12

Conductores de Coste	Muy Bajo	Bajo	Normal	Alto	Muy Alto	Extra Alto
Fiabilidad requerida del software	0.75	0.88	1.00	1.15	1.40	-
Tamaño de la Base de Datos	-	0.94	1.00	1.08	1.16	-
Complejidad del producto	0.70	0.85	1.00	1.15	1.30	1.65
Restricciones del tiempo de ejecución	-	-	1.00	1.11	1.30	1.65
Restricciones del almacenamiento principal	-	-	1.00	1.06	1.21	1.56
Volatilidad de la máquina virtual	-	0.87	1.00	1.15	1.30	-
Tiempo de	-	0.87	1.00	1.07	1.15	-

respuesta del ordenador						
Capacidad del analista	1.49	1.19	1.00	0.86	0,71	-
Experiencia en la aplicación	1.29	1.13	1.00	0.91	0.82	-
Capacidad de los programadores	1.42	1.17	1.00	0.86	0.70	-
Experiencia en S.O. utilizado	1.21	1.10	1.00	0.90	-	-
Experiencia en el lenguaje de programación	1.14	1.07	1.00	0.95	-	-
Prácticas de programación modernas	1.24	1.10	1.00	0.91	0.82	-
Utilización de herramientas software	1.24	1.10	1.00	0.91	0.83	-
Limitaciones de planificación de proyecto	1.23	1.08	1.00	1.04	1.10	-

$$FAE=1.4*0.94*1.15*1.11*1.0*1.0*1.15*0.71*0.91*1.00*1.10*0.95*1.00*0.91*1.10$$

$$\sim 1.3056$$

El LDC de Python no está determinado pero algunos documentos indican que es menor o igual al lenguaje de programación Perl, por lo que vamos a tomar como medida el valor para este que es 21.

Para el cálculo del PF, utilizamos el método de IFPUG-FPA¹:

Componente	Tipo de Componente	Nivel de Complejidad	Puntos de Función
Obtención de datos financieros históricos de las acciones	Entrada Externa	Bajo	3
Aplicación de modelos matemáticos a datos financiero	Salida Externa	Alto	7
Despliegue de información obtenida y gráficos	Consulta Externa	Medio	4
Subir archivo CSV de datos históricos	Entrada Externa	Bajo	3
Guardar resultados finales	Archivo Lógico Interno	Bajo	7

Así, nuestro PF será 24.

$$KLDC = PF * LDC / 1000 = 24 * 21 / 1000 = 0,504$$

$$E = a * KLDC^{\{e\}} * FAE = 3.0 * 0,504^{\{1.12\}} * 1.3056 \sim 1,8183 \text{ [personas/mes]}$$

¹ Método IFPUG-FPA (2011) <https://www.infor.uva.es/~manso/calidad/PFA-CLM-2011.pdf>

Identificación preliminar de riesgos para el proyecto

1. Requisitos poco claros: El cliente no tiene claridad de lo que desea, lo cual complica el levantamiento de requisitos de parte nuestra y pone en peligro todo el desarrollo del proyecto. Además, se tiene poca cantidad de requisitos funcionales, lo que implica que el sistema puede crecer en propiedades del sistema (requisitos no funcionales), los cuales (en general) son más complejos de abordar.
2. Complejidad del Dominio a estudiar: El procesamiento de los datos depende de matemática compleja, la cual el equipo de desarrollo no maneja del todo. Por lo mismo, se hace uso del lenguaje estadístico R especializado en ello, lo cual ayuda al procesamiento de los datos pero complejiza el sistema.
3. Dependencia de un sistema externo: Para la obtención de datos históricos a procesar por parte del sistema, se necesita un sistema externo que provea de una API, la cual nuestro sistema consumirá. En nuestro caso, se hará uso de los datos de Yahoo! Finance. Sin embargo, dado que es un sistema que no es controlado por nosotros, no es posible garantizar el uptime del mismo. Por lo tanto, es posible que el sistema no pueda obtener los datos históricos necesarios en algún momento.

Como equipo, el que consideramos el mayor riesgo, es el riesgo (1), debido a que elicitación pobre o mala puede perjudicar el valor final del producto por parte del cliente y/o stakeholders. La forma de mitigar esto es gracias al proceso de desarrollo **iterativo incremental**, con el cual cada ciertos periodos se validan requisitos, diseño y funcionalidades con el cliente y/o stakeholders.