

GRADO EN INGENIERÍA INFORMÁTICA DE GESTIÓN Y
SISTEMAS DE INFORMACIÓN
TRABAJO FIN DE GRADO

SIMULADOR DE BOLSA

Alumno/Alumna: Martín Casanova, Miguel

Director/Directora: Ferrero Martín, Begoña

Curso: <2018-2019>

Fecha: <2019, julio>

RESUMEN

En este proyecto se desarrollará un juego educativo para introducir al usuario en el mundo de la bolsa. Para ello proveerá documentación e información para guiarle en sus primeros pasos y unos simuladores en los que el usuario podrá poner a prueba lo aprendido haciendo inversiones simuladas en bolsa con datos reales, como si jugase en la bolsa autentica.

Empezaremos viendo la idea que dio lugar a esta aplicación, como funciona dicha idea y que elementos tiene. También, veremos cuál es la metodología que se va a seguir, cuál será el plan para desarrollar la aplicación y los principales riesgos que podemos encontrar al desarrollar una aplicación de estas características.

Seguidamente, haremos un repaso de algunas aplicaciones similares a la que se ha desarrollado. Miraremos, entre otras cosas, su manejabilidad, los datos que utilizan, su organización y hacia qué tipo de usuario están enfocadas.

Tras esto, estudiaremos que datos tendrá que manejar la aplicación y que requisitos debe cumplir. Además, analizaremos minuciosamente los elementos que compondrán la aplicación, veremos cómo deberán interactuar entre ellos y los datos que se usarán para lograr los objetivos de la aplicación.

Una vez explicado como es el funcionamiento de la aplicación, pasaremos al desarrollo donde se relata el proceso que se ha seguido para crear la aplicación y como se solucionaron los problemas que aparecieron. Por último, revisaremos las pruebas de evaluación a las que se expondrá el trabajo.

ABSTRACT

This project develops an educational game that will teach the user the basics of the stock market. In order to achieve it, documents and data will be provided alongside two simulators where the user will be able to test what has learnt by investing in a simulation of the stock market, just like is done in real life.

We will begin by introducing the idea that gave birth to this application, how it works and what elements are part of it. Also, we will see what methodology will be followed, the planning of the development process and the main risks that making this kind of application could face.

Afterwards, we will have a look to some similar applications. Alongside other things, we will exhibit their manageability, the data they use to work, the way they are organized and for what kind of users they were designed.

Then, we will study what data needs the application and what conditions it needs to meet. Furthermore, we will carefully analyse the elements that will be part of the application, how they interact with each other and what data will be used for achieving the objectives of the application.

Once the behaviour of the application has been explained, we will then see how the process of developing the application has been and how were solved the problems that came out. Finally, we will display how the work has been evaluated by making it undergo several tests.

Agradecimientos

Este proyecto ha sido una aventura para mí y quiero dar las gracias a todos los que me han ayudado durante su desarrollo, estoy seguro de que sin su apoyo todo esto no se hubiera llevado a cabo. Sois estupendos, muchas gracias a todos.

En segundo lugar, me gustaría agradecer a todos mis amigos, familiares y sobre todo a aquellas personas que durante la realización de este trabajo se han volcado en animarme y ayudarme cuando lo necesitaba. Han sido un apoyo constante, sin estas personas todo esto no hubiera sido posible. Mil gracias a todos ellos.

En tercer lugar, me gustaría agradecer la ayuda de todos los profesionales, profesores y tutores que se han implicado conmigo durante la carrera. Han demostrado tener paciencia y dedicación conmigo y con todos sus alumnos. En sus clases se nota la voluntad de enseñar y la vocación con la que viven la enseñanza. Gracias en especial a mi tutora del proyecto María Begoña del Pilar Ferrero, a Juan Pedro Reyes por ayudarme en los temas relacionados con economía y a Juan Antonio Pereira por prestarme un servidor para el trabajo.

Por último, gracias también a todas las personas de Bankinter que me han ayudado y enseñado y a los profesionales de Micro Focus, que me han formado, prestado su ayuda y aguantado con paciencia todas mis preguntas.

Gracias a todos.

ÍNDICE DE CONTENIDOS

1. Introducción	7
1.1 Equipo del proyecto.....	7
1.2 Objetivos principales del proyecto	7
1.3 Estructura de la memoria	8
2. Planteamiento inicial	9
2.1 Teoría	9
2.2 Simuladores	12
2.3 Herramientas	12
2.4 Pedagogía:	12
2.5 Metodología de trabajo	12
2.6 Gestión de riesgos	12
2.7 Tecnología	15
3. Antecedentes	17
4. Captura de requisitos	21
4.1 Jerarquía de Actores.....	22
4.2 Casos de uso.....	22
4.3 Análisis de Casos de Uso	23
5. Análisis y diseño.....	25
5.1 Modelo del dominio	26
5.2 Arquitectura de la aplicación	29
6. Desarrollo del proyecto	33
7. Pruebas.....	37
7.1 Recolección de datos de bolsa en tiempo real	37
7.2 Representación gráfica de los datos de bolsa.....	38
7.3 Pruebas de comunicación con COBOL.....	40
8. Conclusiones y trabajo futuro	43
8.1 Conclusiones	43
8.2 Trabajo futuro	45
Referencias.....	49
ANEXO: Manual de usuario.....	51

ÍNDICE DE ILUSTRACIONES

Ilustración 1. Diagrama de Gantt	13
Ilustración 2. Herramienta de la BBK	18
Ilustración 3. Casos de uso	22
Ilustración 4. Tablas de la base de datos	28
Ilustración 5. Arquitectura de la aplicación.....	29
Ilustración 6. Ejemplo de grafico	29
Ilustración 7. Ejemplo de grafico	33
Ilustración 8. Html de "El economista"	34
Ilustración 9. Resultado final	35
Ilustración 10. Prueba 1.....	37
Ilustración 11. Prueba 2.....	39
Ilustración 12. Prueba 3.....	40
Ilustración 13. Pantalla de login	52
Ilustración 14. Pantalla de inicio.....	52
Ilustración 15. Grafico	54
Ilustración 16. Análisis	56
Ilustración 17. Ejemplo de la calculo.....	57
Ilustración 18. Teoría	58
Ilustración 19. Selección de teoría	59
Ilustración 20. Documentacion.....	60
Ilustración 21. Selección de pruebas.....	61
Ilustración 22. Prueba	62
Ilustración 23. Pantalla de inicio.....	63

1. Introducción

Este proyecto plantea desarrollar un simulador de inversión en bolsa que funcione en tiempo real. La aplicación llevará un registro actualizado con la información de las inversiones simuladas por el usuario utilizando datos de la bolsa real. Además, enseñará al usuario estrategias de inversión que le ayudarán en su aprendizaje.

El trabajo a realizar utilizará COBOL para hacer todos los cálculos técnicos, y Java para la implementación de la interfaz gráfica y el web scrapping¹.

1.1 Equipo del proyecto

Los roles oficiales han sido:

Miguel Martín Casanova – Autor del proyecto.

María Begoña del Pilar Ferrero Martin – Tutora del proyecto.

1.2 Objetivos principales del proyecto

El objetivo de este trabajo es desarrollar una aplicación capaz de simular inversiones en la bolsa y ayudar al usuario a que aprenda y mejore sus habilidades a la hora de invertir.

Para llegar a conseguir el objetivo planteado, se han definido distintos hitos:

¹ Técnica usada para extraer los datos de la bolsa en tiempo real

- I. Captura de datos: el programa debe ser capaz de capturar los datos de la bolsa real a través de internet.
- II. Análisis de datos: los datos recogidos deben ser mostrados gráficamente y se le debe permitir al usuario operar sobre ellos.
- III. Almacenamiento de datos: se debe guardar en una base de datos el nombre del usuario, así como las inversiones que este realiza.
- IV. Enseñanza: al usuario se le ofrecerán documentos relacionados con las inversiones en bolsa para que pueda estudiarlos y ampliar sus conocimientos.

1.3 Estructura de la memoria

Esta memoria consta de ocho capítulos con los siguientes contenidos:

- Capítulo 1: Introducción del trabajo.
- Capítulo 2: Ofrece una visión global del proyecto, describiendo sus objetivos, la planificación del trabajo a realizar y estimaciones del tiempo que tomará su realización, así como los posibles riesgos inherentes a su desarrollo.
- Capítulo 3: Hace un repaso de las aplicaciones ya existentes parecidas a esta.
- Capítulo 4: Documentación de la captura de requisitos que debe cumplir la aplicación a desarrollar y sus casos de uso.
- Capítulo 5: Análisis y diseño de la aplicación.
- Capítulo 6: Proceso de desarrollo del proyecto incluyendo una explicación breve de los problemas encontrados.
- Capítulo 7: Pruebas realizadas y los resultados obtenidos en cada una.
- Capítulo 8: Conclusiones a las que se ha llegado tras realizar el proyecto. Finaliza presentando el trabajo futuro, en el que se resaltan algunos aspectos a añadir y posibles mejoras al trabajo realizado.

2. Planteamiento inicial

La finalidad principal de esta aplicación es la enseñanza. La estrategia de aprendizaje que se utilizará principalmente será prueba y error, esa es la razón por la que se pretende ofrecer un entorno lo más realista posible. La aplicación está pensada para un solo usuario porque en la bolsa, aunque intervienen muchas personas, realmente no se interactúa con ellas, por esta razón la aplicación será monousuario.

La aplicación está planteada como un juego educativo con tres niveles de dificultad. El usuario empezará en el nivel 1 donde podrá acceder a los ejercicios más simples y a la documentación con el contenido más básico sobre la bolsa. Lo que aprenda podrá aplicarlo en unos simuladores de la bolsa a los que se tendrá acceso desde el momento en el que se inicie la aplicación. Cuando el usuario gane dinero en las simulaciones, obtendrá puntos de experiencia. Cuando acumule bastantes puntos subirá de nivel y podrá acceder a documentación y a ejercicios más complicados. A continuación, iremos explicando al detalle todos estos aspectos.

2.1 Teoría

El usuario podría no tener ninguna experiencia en el mercado de valores, por lo que se le ofrecerán documentos relacionados con la materia para que los estudie y ponga en práctica lo que vaya aprendiendo. En ellos podrá encontrar conceptos, estrategias e ideas que podrá aplicar en el simulador. El usuario no tendrá acceso a todos los documentos desde el principio, empezará con unos pocos y el resto irán desbloqueándose a medida que vaya subiendo de nivel. El acceso a los nuevos documentos abrirá las puertas a nuevas formas de obtener ganancias y experiencia, hecho que motivará al usuario para consultarlos y seguir aprendiendo de ellos para avanzar lo más posible. La complejidad de los

documentos disponibles irá aumentando según aumente de nivel del usuario. El contenido de los documentos estará guardado en archivos de texto.

Además, habrá un glosario al que se podrá acceder desde el inicio de la partida y que tendrá los significados de palabras que aparecen en la teoría que puedan resultar confusas.

2.2 Simuladores

La parte más importante de la aplicación será un simulador de bolsa que use datos reales en tiempo real, de esta forma, el usuario podrá simular compras y ventas en la bolsa como lo haría en el mundo real, poner a prueba sus ideas y aprender de los resultados de sus inversiones. El aprendizaje surgirá de la practica en un entorno realista. El usuario empezará con 10.000\$ para invertir. En cada partida se controlará la cantidad máxima de dinero acumulado por el jugador, como si fuera un récord. Cada vez que se bata un nuevo récord se comprobará la diferencia que había entre el récord nuevo y el anterior, cuanto mayor sea la diferencia entre ambos más experiencia se le dará al usuario. Si gana dinero y no bate el récord, no obtendrá nada de experiencia.

Pongamos un ejemplo: el usuario tiene como récord y como cantidad de dinero actual 10.000\$ pero pierde 2.000\$ y gana 1.000\$ habrá conseguido dinero, pero no beneficios y no obtendrá experiencia. En cambio, si pierde 1.000\$ y gana 2.000\$ tendrá ahora 11.000\$ en el bolsillo, se registrará un nuevo récord y obtendrá experiencia por esos 1.000\$ de beneficios.

También, habrá un simulador que no usará los datos de la bolsa en tiempo real, sino datos históricos recogidos en ficheros de texto. Con ellos se podrán plantear ejercicios al usuario para que pueda ganar experiencia sin tener que depender de los horarios del mercado de valores real.

En este simulador histórico no se usará el mismo dinero que se usa en el simulador con tiempo real: al principio del ejercicio, se prestarán al usuario 10.000\$ para que los invierta durante el ejercicio. El ejercicio utilizará los datos históricos de los ficheros de texto para emular el mercado de valores real durante un periodo de tiempo. Estos datos históricos se irán mostrando en orden cronológico en un gráfico. El usuario elige la velocidad a la que quiere que se vayan mostrando. Podrá comprar y vender acciones en el momento en el que le parezca oportuno a la vista de los datos que se están mostrando. El ejercicio acabará cuando se hayan mostrado todos los datos. Si cuando acabe el ejercicio tiene más dinero que cuando empezó, habrá superado el ejercicio y obtendrá puntos de experiencia. Independientemente de que gane o pierda, se retirará el dinero prestado al principio de la prueba.

En el simulador histórico los puntos de experiencia se ganan más despacio que en el real, ya que es preferible que el usuario use más este último, por ser más realista: no es posible hacerle trampas y está limitado por los horarios.

2.3 Herramientas

Los éxitos del usuario se recompensarán dándole acceso a herramientas que le facilitarán la realización de cálculos sobre los datos de la bolsa y pistas que le aporten información importante sobre las empresas. Solo podrán usarse en el simulador en tiempo real.

Por ejemplo, una de estas herramientas permitiría obtener la media de una serie de datos y mostrarla por pantalla en un gráfico de líneas. Una "pista" cogería un conjunto de datos, con los que haría una serie cálculos aportando información en base a sus resultados. Esta información podría ser, por ejemplo, si los datos indican que es un valor estable.

El usuario podrá reiniciar su partida en cualquier momento.

2.4 Pedagogía

Como ya se ha comentado, la aplicación establece tres niveles de dificultad, siendo el nivel 1 el más básico. Para subir de nivel es necesario acumular experiencia, que se consigue superando pruebas en el simulador histórico o ganando dinero en el simulador en tiempo real.

La herramienta mide el conocimiento del usuario en función del dinero ganado ya que es posible que el usuario haga todo lo que tiene que hacer y aun así pierda dinero. Todos los expertos a los que se ha consultado opinan lo mismo. Esto ocurre en la vida real y, dado que el objetivo es hacer una simulación realista, lo lógico es dejar que el usuario se exponga a estos reveses y entienda que las cosas no son tan fáciles como lo que se muestra en las películas.

2.5 Metodología de trabajo

Seguiré la metodología de trabajo SCRUM por su flexibilidad. [1]

Para este proyecto, estimo que habrá cuatro Sprints. Al finalizar cada uno de los Sprints, enseñaré los resultados a mi tutora. Tras ello, podremos hacer una revisión del Sprint, su retrospectiva y la planificación del siguiente Sprint.

Ahora pasaremos a ver los Sprints y los hitos a cumplir en cada uno de ellos:

- Sprint 1: Estudiar las aplicaciones de bolsa ya existentes y los elementos que contienen: Antes de empezar a programar es necesario averiguar qué va a necesitar la aplicación, para ello observaré aplicaciones de bolsa ya existentes e imitaré sus características. Ya tengo, a grandes rasgos, una idea clara de lo que quiero que haga, pero es necesario ver cómo son las aplicaciones de verdad para poder añadirle más detalles. Estimo que durará una semana,

2.6 Gestión de riesgos

Este trabajo afronta varios retos, algunos pueden poner en peligro el proyecto entero y otros no son tan graves. En este apartado se exponen los principales problemas que pueden ocurrir junto con las estrategias que usaremos para lograr resolverlos:

1. No ha sido posible obtener los datos de bolsa en tiempo real:

Este es el mayor riesgo de todos. No se puede tener un simulador de bolsa en tiempo real sin los datos en tiempo real. Si esto llegase a ocurrir habría que idear otro proyecto. Otras soluciones como generar los datos aleatoriamente no funcionarían ya que perdería completamente el realismo. Tampoco valdría usar únicamente datos históricos o en diferido ya que sería demasiado fácil hacer trampas y no se aprendería nada realmente. Esto puede chocar con el hecho de que se planea implementar un simulador de datos históricos en la aplicación. La razón por la que esto no será un problema es porque dicho simulador sería usado como un complemento y estará más limitado para evitar trampas y atajos.

2. No ha sido posible establecer una comunicación entre Java y COBOL o no es posible programar con COBOL:

El hecho de que COBOL sea un lenguaje que no tiene nada que ver con todo lo que he visto hasta ahora conlleva riesgos. Por lo que he visto hasta este momento, hay opiniones tanto a favor como en contra sobre la manejabilidad de este lenguaje, así que hasta que no empiece a programar no sabré cómo es de complicado. En el peor de los casos, si no soy capaz de hacerme con él en un periodo razonable de tiempo no será un gran problema ya que podré seguir con el proyecto usando solo Java. Podría conllevar algún retraso por el tiempo perdido intentando usarlo, pero sería principalmente algo negativo para mi interés personal.

3. Complicaciones al intentar obtener documentación relacionada con el mercado de valores:

En este tema también soy un novato. Mis conocimientos sobre el mercado de valores son limitados. La diferencia entre esto y COBOL es que por lo menos he estudiado una carrera de programación y un lenguaje nuevo no debería representar un desafío tan grande como lo es aventurarme en economía y bolsa. En un principio, planeo usar los datos más fiables posibles. No descarto ir a los despachos de los profesores de economía a buscar información. Si no es posible, tendré que depender de lo que encuentre investigando en internet, aunque no esté seguro de que sea fiable. Al fin y al cabo, este es un proyecto de informática, no de economía.

2.7 Tecnología

Estas son las herramientas que se van a usar para desarrollar la aplicación:

- Java - Eclipse: se usará para el programa principal, para la interfaz gráfica y para recoger de forma continua los datos de la bolsa en tiempo real.
- COBOL - Visual COBOL de Micro Focus [2]: utilizado para realizar los cálculos con los datos obtenidos por Java.
- MySQL – MySQL workbench: almacenará los datos.

3. Antecedentes

Existen infinidad de aplicaciones de bolsa, prácticamente cada entidad bancaria tiene una. A diferencia de la que se presenta en este TFG cuentan con una larga lista de funciones e información que ponen a disposición del usuario.

Aunque una aplicación con tanta información pueda parecer muy útil, también puede acarrear problemas: pueden hacer que el usuario se sienta abrumado. Todas las aplicaciones que he visto estaban orientadas a un usuario profesional con experiencia y que sabe cómo invertir en bolsa. Esto para un usuario novato es sobrecogedor: tantos números y siglas que no entiende hacen que el usuario se sienta incapaz de manejar toda la información.

Cuando fui a Bankinter a pedir consejo sobre que podía introducir en la aplicación, me enseñaron la herramienta que ellos utilizan [3]. Ya había mirado la aplicación de Bankinter [4], pero esto era distinto: era su herramienta de análisis. Para acceder a ella era necesario tener una cuenta en Bankinter, razón por la cual no pude encontrarla en un principio. Se parecía bastante a las anteriores, pero aquí sí que vi más detalles que podrían guiar a un usuario con poca experiencia: la información que representaba estaba mejor explicada y además la propia herramienta analizaba cada valor y aconsejaba si era conveniente invertir, esperar, vender o comprar. En ese sentido se parecía bastante a las pistas que había planeado implementar en mi proyecto solo que más específico y usando más datos, por lo que iba por el buen camino.

Además de eso, contaba con un gráfico de barras que podía representar la volatilidad de una acción con el paso del tiempo o la cantidad de operaciones que se estaban realizando sobre esa acción.

A diferencia de la mía, y como ya he visto en muchas otras, tenía una tabla que mostraba los datos de todos los valores de un mercado y los actualizaba minuto a minuto. Es posible que con la práctica sea más fácil y más práctico moverse por todos esos números, pero a primera vista parecía demasiado complicado,

por lo que probablemente para un aprendiz sea más cómodo centrarse en los valores de uno en uno.

También, mire las aplicaciones de otros bancos. Por supuesto solo se podía acceder a ellas teniendo una cuenta. La única herramienta a la que pude acceder, aparte de la de Bankinter, fue la de la BBK [5] (Ilustración 2) que es donde yo tengo una cuenta:

The screenshot displays the BBK financial tool interface. At the top, there are tabs for 'Resumen' and 'Mi Resumen'. The main section is titled 'Índices' and shows the IBEX 35 index at 9,422.30, down 83.70 (-0.88%) from the previous day. It also highlights a 11.31% increase in the 2019 revaluation. A historical chart for the IBEX index is visible, showing a peak of 10,291.40 on 14/05/18 and a low of 8,286.20 on 27/12/18. Below the index, there is a table of 'Mayores subidas y bajadas' (Biggest gains and losses) for various companies like BANKIA, IAG, CAIXABANK, MEDIASET ESPAÑA, and SANTANDER. To the right, there are sections for 'Favoritos' (Favorites), 'Calendario' (Calendar) showing the date as Monday, April 29, 2019, and 'MIS COMUNICACIONES' (My Communications) with options like 'Correspondencia' and 'Buzón de avisos'. At the bottom right, there is an 'INFORMACIÓN' (Information) section with a dropdown menu for 'Productos' and other options.

Ilustración 2. Herramienta de la BBK

A primera vista, parecía más simple que la de Bankinter, pero luego me di cuenta de que para acceder a más información era necesario contratar más funciones y costaban dinero.

Todo lo que pude ver es como te mostraba los datos de los valores sin aportar ayudas para interpretarlos.

Mi aplicación tiene como objetivo mejorar la situación descrita, ofreciendo al usuario información adicional sobre cómo interpretar los datos que está viendo. Con ello pretende lograr que un usuario sin experiencia pueda invertir en bolsa y acumular experiencia en un entorno lo más realista posible utilizando un programa más accesible y ameno.

4. Captura de requisitos

Estos son los requisitos de la aplicación:

- Obtención de los datos en tiempo real: que la aplicación sea capaz de coger los datos del mercado de valores en tiempo real.
- Representación de los datos: que los datos de la bolsa se puedan ver fácilmente. Estos datos tienen que ser representados en un gráfico de líneas.
- Interacción con los datos: el usuario tiene que poder comprar y vender acciones. Se debe tener en cuenta que los precios de las acciones coincidan con el precio de la bolsa real en ese mismo momento.
- Manejo y cálculo de los datos: que el usuario pueda realizar cálculos con los datos con el fin de obtener más información para mejorar sus operaciones. También debe tener acceso a consejos y pistas relacionados con dichos datos.
- Almacenamiento de los datos: los datos deben actualizarse en la base de datos cada vez que realiza una operación.
- La aplicación será monousuaria: dadas las características de la aplicación, lo lógico es que solo haya un usuario. Se pedirá al usuario que introduzca su nombre para que la aplicación sea más acogedora. También se guardará el récord de dinero acumulado por el usuario. Habrá un récord que no se borrará cuando se reinicie la partida y otro que si se borrará. El primero servirá para que el usuario pueda ver su mayor logro y para que intente superarse a sí mismo, el segundo servirá para lo mismo en menor medida y también para controlar su progreso.

4.1 Jerarquía de Actores

Se identifica un único usuario que controla la aplicación y puede acceder a diferentes acciones en función de su nivel. Desde que inicia la partida podrá entrar al simulador en tiempo real donde tendrá disponibles las opciones de comprar y vender. Podrá acceder a los contenidos básicos, consultar la teoría y hacer pruebas, así como hacer cálculos sobre los datos de bolsa y recibir consejos adecuados a su nivel. También podrá reiniciar la partida o cambiar el idioma en cualquier momento.

4.2 Casos de uso

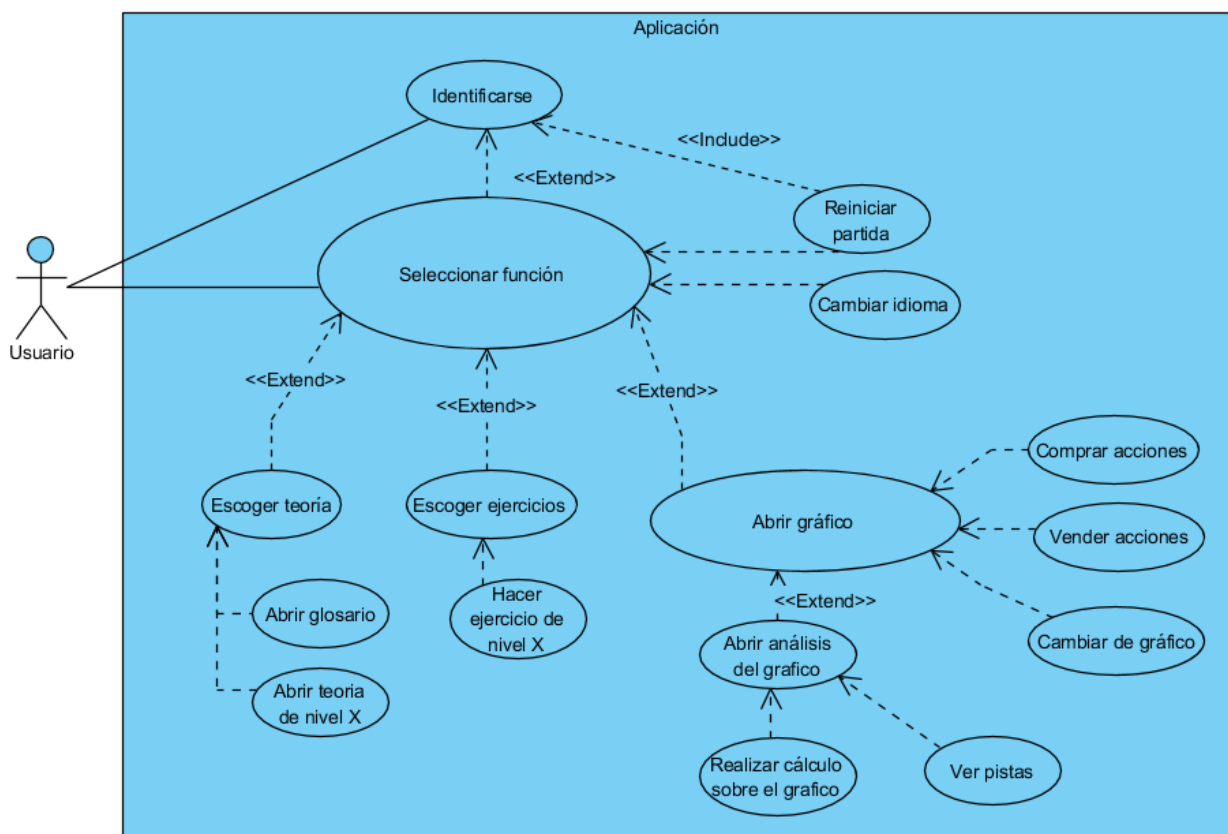


Ilustración 3. Casos de uso

4.3 Análisis de Casos de Uso

Aquí analizaremos los casos de uso de la Ilustración 3.

- **Identificarse:** pantalla de registro del usuario donde debe introducir su nombre y el idioma que prefiera usar en la aplicación. Sólo se le mostrará esta pantalla la primera vez que acceda a la aplicación o cuando quiera volver a comenzar tras haberse rendido.
- **Seleccionar función:** desde aquí el usuario puede acceder a los dos simuladores de la aplicación y a la teoría. También se puede cambiar el idioma de la aplicación y reiniciar la partida. Esta última opción borrará los datos de esta partida y llevará al usuario a la pantalla de registro.
- **Escoger ejercicios:** se muestran los ejercicios disponibles de cada nivel y el usuario puede hacer el ejercicio que desee si esta desbloqueado.
- **Escoger teoría:** se muestra la teoría disponible por cada nivel y el usuario puede acceder a la adecuada a su nivel.
- **Abrir gráfico:** tras seleccionar un mercado en la pantalla principal, se muestra el gráfico con los datos de ese mercado. Desde aquí se pueden comprar y vender acciones, pasar a ver los gráficos de empresas de ese mercado, cambiar el período de tiempo que muestra el gráfico o ir a la pantalla de análisis. Para cambiar el mercado es necesario volver a la pantalla de inicio.
- **Abrir análisis del gráfico:** aquí el usuario puede interactuar con los datos del gráfico: puede usar herramientas que le permitan ver datos del gráfico con mayor facilidad y pedir pistas que le puedan ayudar en su toma de decisiones.

5. Análisis y diseño

Para alcanzar los objetivos definidos en el apartado 1.2 se requerirá gestionar información proveniente de diferentes orígenes:

- Extraídos de internet: De aquí se obtendrán los datos del mercado de valores. Estos serán los datos que se representarán en el simulador de tiempo real. Estarán conformados por los precios de diferentes valores a lo largo del tiempo hasta la fecha de hoy.

Para obtener el precio actual de un valor, se ha implementado una función que usa el Html de las páginas web para obtener el precio actual del valor. Para obtener los precios históricos de un valor se usará otra función similar que se repetirá varias veces hasta haber obtenido todos los datos que necesite el programa en ese momento.

Para lograr hacer un simulador en tiempo real, he conseguido acceso a un servidor de la universidad, en el he instalado el programa que recoge los diferentes precios de cada valor minuto a minuto todos los días que abre la bolsa guardándolos de forma organizada en archivos de texto. De esta forma, cada vez que se quieran ver los datos en directo solo hay que acceder al servidor y consultar el fichero del valor que se quiera ver.

- Almacenados en ficheros de texto: En ellos se guardará la teoría y los datos históricos de la bolsa. La teoría contendrá la documentación didáctica sobre el mercado de valores; los datos históricos serán los precios de distintos valores durante un periodo de tiempo ya pasado.
- Almacenados en la base de datos: Aquí se guardará el nombre del usuario, así como los datos de las inversiones que realiza. Estos datos

se usarán para seguir el progreso del usuario. Algunos de estos datos estarán definidos por el sistema y otros serán elegidos por el usuario.

5.1 Modelo del dominio

En la aplicación se han definido las siguientes entidades:

- Usuario: para representar a un usuario de la aplicación se necesita la siguiente información.
 - Nombre: es el identificador del usuario y se guarda cuando inicie la aplicación por primera vez.
 - Nivel: representa el nivel alcanzado por el usuario en la partida en curso. Empieza valiendo "1", pero se irá incrementando a medida que el usuario consiga puntos de experiencia. El usuario podrá acceder a nuevas opciones y herramientas dependiendo de su nivel.
 - Experiencia: indica los puntos de experiencia del usuario. Funcionarán de manera similar al nivel, empieza siendo "0" e irá aumentando cuando el usuario gane dinero o supere pruebas.
 - Dinero: representará la cantidad de dinero que tiene el usuario en dólares. Empezará teniendo como valor "10.000" y se usará para comprar y vender acciones simuladas.

- Valor: contendrá los datos relacionados con las acciones de una empresa que posee el usuario. Un usuario podrá tener cero o más valores, pero un valor siempre tendrá un único usuario que será su dueño. El valor está caracterizado por la siguiente información:
 - Empresa: guardará el nombre de la empresa a la que pertenezca el valor.
 - Cantidad: número de acciones de esa empresa que tiene el usuario.
 - Precio de compra: cuánto han costado adquirir dichas acciones. Si el usuario comprase acciones de una empresa por un precio X y luego

volviera a comprar valores de esa empresa por un precio Y , X e Y se sumarían y se guardarían en el precio de compra. Esto servirá para ayudar al usuario a llevar el control de lo que ha invertido.

- Reinicios: en cualquier momento, el usuario tiene la opción de reiniciar la partida en caso de que pierda tanto dinero que ya no pueda seguir jugando en bolsa. Se llevará la cuenta de las veces que se ha reiniciado la partida junto con otros datos relacionados con el progreso del usuario que se describen a continuación:
 - Cuenta: recoge el número de veces que se ha reiniciado la partida.
 - RecordP: representa el récord de dinero acumulado en la partida en curso.
 - RecordF: es similar a RecordP, solo que éste representa el récord de dinero acumulado sin importar que haya sido en una partida anterior.
 - Idioma: contiene el idioma en el que se mostrará la aplicación.
- NombresCorregidos: servirá para guardar y organizar mejor los nombres de las empresas que se mostrarán en la aplicación. En las páginas de economía suelen referirse a las empresas de dos formas: el nombre original y una especie de "diminutivo". Por ejemplo, para la empresa Tesla el nombre original es Tesla Motors y el "diminutivo" es TESLA-MTRS. Ambos serán necesarios para acceder a la paginas de economía y recoger los datos.
 - NombreO: guardará los nombres originales de las empresas que usará la aplicación.
 - NombreP: guardará el "diminutivo".

La aplicación organiza toda esta información en las cuatro tablas que se muestran en la Ilustración 4:

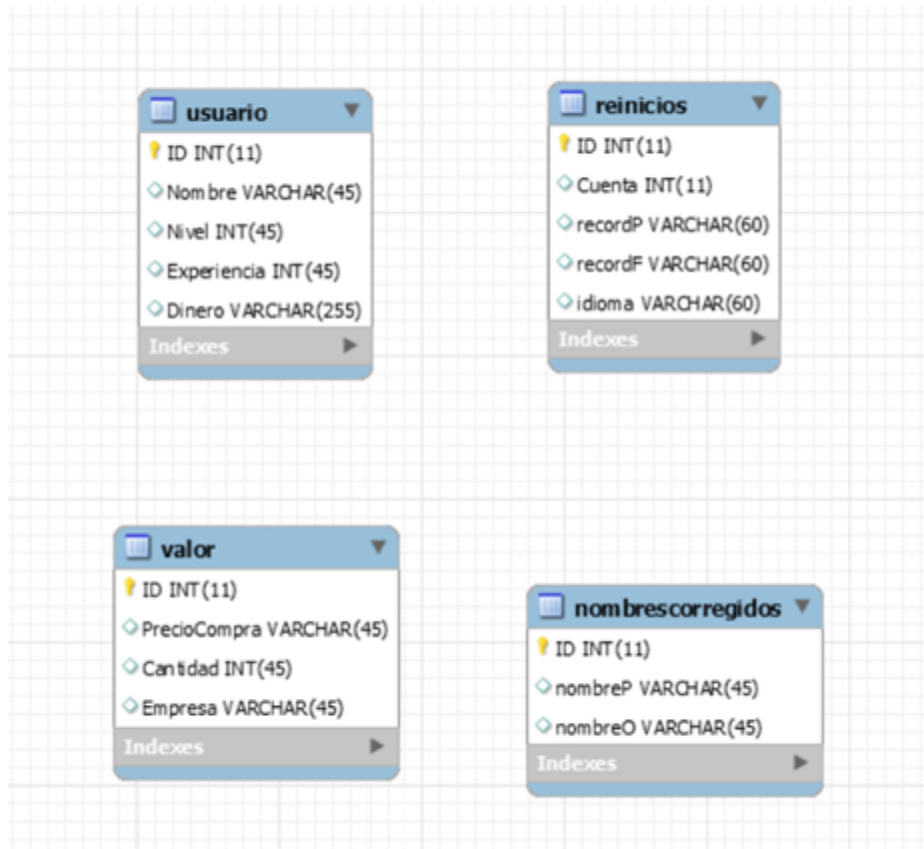


Ilustración 4. Tablas de la base de datos

5.2 Arquitectura de la aplicación

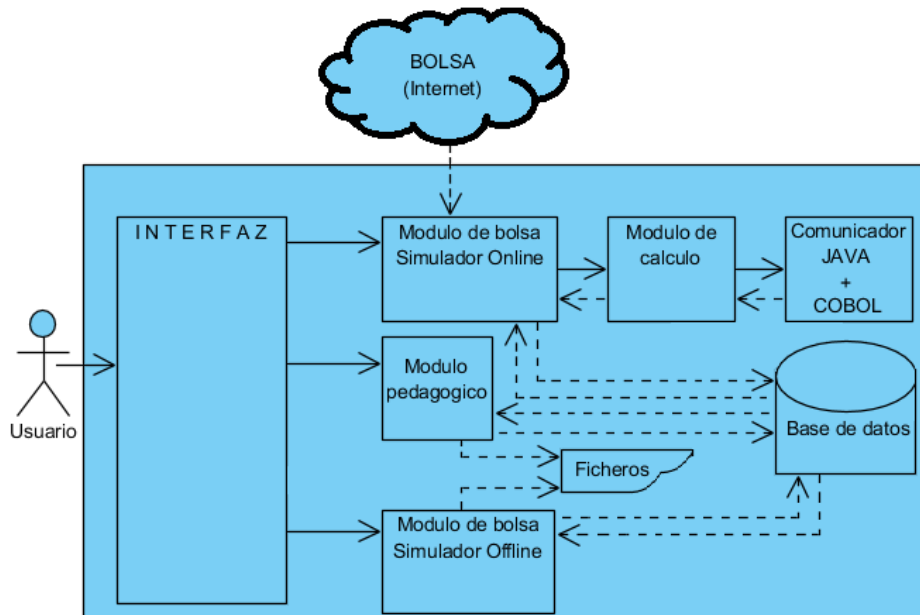




Ilustración 5. Arquitectura de la aplicación

-  Flujo de datos.
-  Comunicación entre módulos.

A continuación, se explica el funcionamiento de los módulos representados en la Ilustración 5:

- Módulo de bolsa, simulador online: este es el módulo encargado de gestionar el simulador de tiempo real y las operaciones de compra y de venta que se realizan en él. Mostrará un gráfico con datos del mercado de valores real de una empresa como lo haría cualquier página o aplicación sobre la bolsa (Ilustración 6).



Ilustración 6. Ejemplo de grafico

El usuario podrá cambiar la empresa que se está mostrando y el periodo de tiempo que muestran los datos. Si el usuario pide ver los datos en tiempo real, este módulo se comunicará a través de Internet con el servidor de la universidad que se ha preparado para obtener dichos datos. Si los datos que se piden son históricos, accederá a una página de bolsa para obtener de allí los datos. Este módulo también se comunica con la base de datos para obtener el dinero y las acciones que posee el usuario y los actualiza en caso de que haga compras, ventas o gane experiencia.

Desde aquí se puede acceder al módulo de cálculo, que se describe a continuación.

- Módulo de cálculo: gestiona los cálculos realizados sobre gráficos. El usuario podrá seleccionar cualquiera de los gráficos que el modulo anterior le haya mostrado y con este módulo podrá realizar cálculos sobre él. Este módulo solo permitirá al usuario aplicarle cálculos que ya haya desbloqueado subiendo de nivel. Cada vez que se haga un cálculo, este módulo mostrara el grafico con los resultados.
- Comunicador JAVA + COBOL: se encarga de gestionar la comunicación entre Java y COBOL. Se accederá a este módulo cada vez que el usuario solicite hacer algún cálculo sobre el grafico que el módulo de cálculo este mostrando. Se enviarán los datos del grafico a COBOL que es el que realizará las operaciones y este devolverá los resultados a JAVA. Hubo problemas para que COBOL leyera correctamente los números que tuvieran más de un dígito, ya que los multiplicaba por potencias de diez. La solución fue dividir los números en dígitos y escribirlos en línea por línea de forma ordenada usando letras para indicar donde había una coma y cuando se pasaba al siguiente número. Por ejemplo, si quisiéramos pasar los números 10 y 2,4 el fichero seria así:

1
0
X
2
C
4
X

De esta forma se logró establecer una comunicación entre JAVA y COBOL.

- Modulo pedagógico: es el encargado de gestionar la teoría que se le enseña al usuario sobre el mercado de valores. Se comunica con la base de datos para obtener el nivel del usuario. En función del nivel del usuario desbloquea unos módulos u otros, de forma que solo pueda acceder a los que adecuados a su nivel. Si el usuario quiere acceder a uno de ellos, este módulo se comunicará con los ficheros para obtenerlo y se lo mostrará por pantalla.

- Módulo de bolsa, simulador offline: este módulo se encarga de gestionar las pruebas con datos históricos. Cada nivel tiene sus pruebas asignadas, cuanto más alto sea el nivel más complicada será la prueba. De la misma forma que en el anterior, se consulta la base de datos para conocer el nivel del usuario y, en base a este, se le permite acceder a las pruebas de dicho nivel y de los inferiores.

Tras elegir uno de los niveles de dificultad desbloqueados, este módulo le llevará a una de las pruebas de dicho nivel. La prueba será elegida de manera aleatoria de entre todas las asignadas al nivel que ha seleccionado el usuario.

Este módulo se encargará de gestionar la prueba: llevará el control del dinero que tiene el usuario durante la prueba, las acciones que posee, las operaciones de compra y de venta que realiza y se encargará de que los datos se muestren de forma cronología y al ritmo que le indique el usuario.

Si el usuario logra superar la prueba, el modulo se volverá a comunicar con la base de datos para sumar los puntos de experiencia que ha ganado y para actualizar los datos del usuario que sean necesarios.

6. Desarrollo del proyecto

Sprint 1:

Como ya se vio en el apartado 2.2, el primer Sprint consistía en observar varias aplicaciones de bolsa y tomar nota de lo que usaba cada una. Tras ir a los despachos economía para pedir consejo sobre por dónde empezar, me sugirieron que copiase a los mejores así que solo mire las aplicaciones oficiales de los bancos. Tras haber observado aplicaciones como la de Bankinter y páginas web como "El economista" [6] decidí lo que la aplicación iba a necesitar, así que lo redacté en la planificación y en el diseño del proyecto. Esto llevo una semana como tenía planeado.

Sprint 2:

En el Sprint 2 tenía que obtener datos del mercado de valores en tiempo real para poder representarlos gráficamente como se suele ver en cualquier aplicación o página sobre economía (Ilustración 7).



Ilustración 7. Ejemplo de grafico

La dificultad estaba en que los datos del mercado en tiempo real no son gratuitos. Las empresas y las páginas que los muestran han tenido que pagar o llegar a algún acuerdo con alguien que pueda proveerles los datos y, por supuesto, no van a dejar que otros cojan lo que a ellos les ha costado tanto

conseguir. En los Html de las páginas de economía no se podía acceder a todos los precios del día de hoy, pero si se podía ver el precio actual de cualquier valor (Ilustración 8).

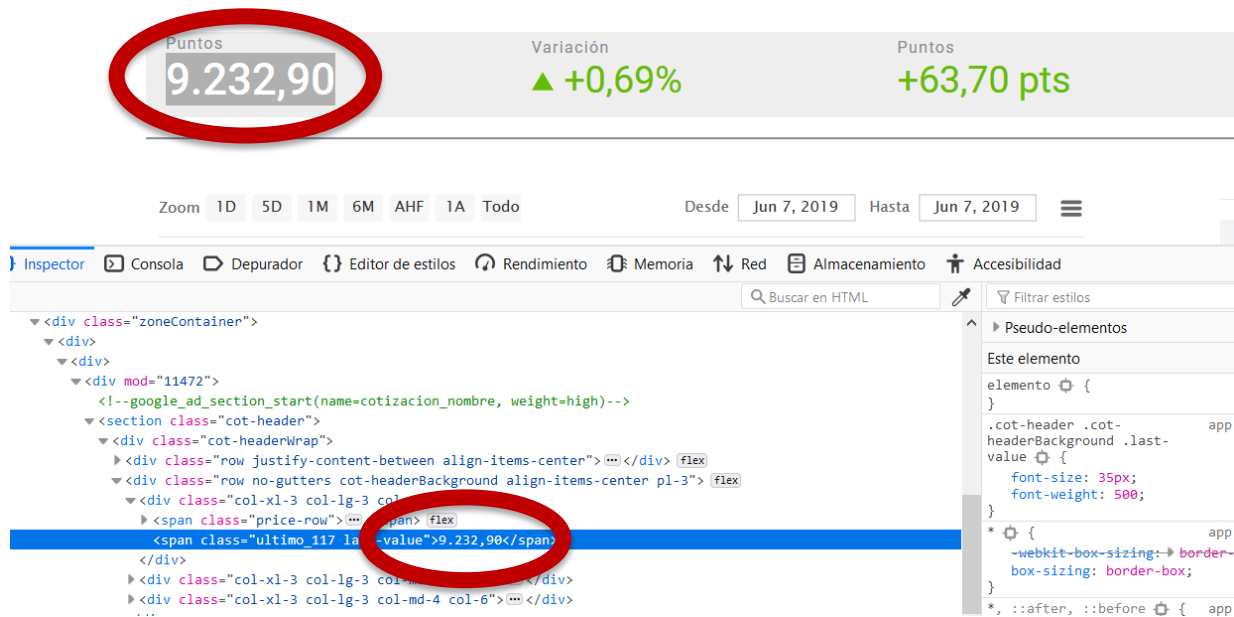


Ilustración 8. Html de "El economista"

La solución fue usar un servidor de la universidad e implementar en él un programa de Java que, usando los Html de las páginas de economía, recogiera minuto a minuto el precio actual de los valores de la bolsa y guardase de forma ordenada los datos de cada valor en ficheros de texto. De esta forma los ficheros se iban actualizando con los datos del día y servían como medio para obtener los datos en tiempo real. Esto presento algunas dificultades: las páginas de economía se están actualizando constantemente y, en momentos puntuales, esto creaba problemas al leer el Html y el programa se bloqueaba, por lo que se hicieron varias pruebas hasta lograr controlar todas las excepciones que aparecían.

Tras resolver este problema, hubo dificultades para representar los datos del mercado de valores. La idea era imitar lo visto en las páginas y en las aplicaciones de economía (Ilustración 7). Java no dispone de una forma de cumplir ese objetivo, por lo que hubo que añadirle librerías externas para lograrlo. Encontré varias librerías, pero tenían defectos y eran difíciles de usar:

desde no poder controlar el zoom hasta representar las líneas demasiado pequeñas. Viendo que podría no encontrar una librería lo bastante buena, decidí poner una tabla a la izquierda del gráfico con los datos que se están mostrando, de esta forma compensaría los defectos del gráfico. Poco después, encontré una librería que tenía lo que necesitaba [7] y pude representar los datos como tenía planeado. Al final decidí conservar la tabla que había puesto anteriormente ya que servía como un apoyo al gráfico. (Ilustración 9)

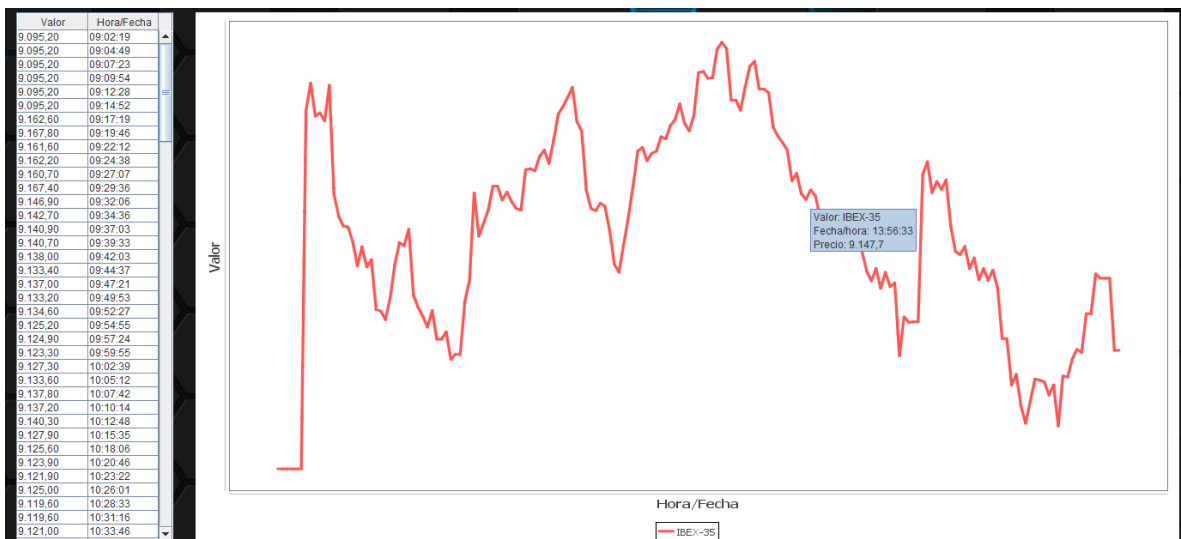


Ilustración 9. Resultado final

El segundo Sprint había llevado más tiempo de lo esperado: 9 semanas. Según lo planificado debería haber durado 5 semanas.

Sprint 3:

En el tercer Sprint había que establecer la comunicación entre Java y COBOL e implementar la parte de la aplicación que no dependa ni de la documentación sobre la bolsa ni de los cálculos sobre el mercado de valores.

Como prueba, quería lograr que Java le pasase a COBOL un fichero de texto con varios números, que COBOL calculase la media de esos números y se la devolviera a Java en un fichero de texto. Hubo problemas para hacer que COBOL leyera los ficheros de texto. Por alguna razón, al leer los números los multiplicaba por potencias de cien, pero si le pasaba números menores de 10 no tenía problemas en leerlos. La solución fue escribir en el fichero los números

digito por digito y colocar una marca cada vez que empezaba un número nuevo o había una coma, como si estuvieran “desmontados”. De esta forma, COBOL los leía y, usando a una función, los “montaba” de nuevo. Calcular la media en COBOL y devolver el resultado en un fichero no dio problemas.

Al final el Sprint 3 se cumplió en el tiempo esperado: 8 semanas.

Sprint 4:

En el Sprint 4 tenía que obtener la documentación sobre el mercado de valores que se usaría para enseñar al futuro usuario, los cálculos que se harían sobre el gráfico cuando se estuviera interactuando con él y las pistas que se le podrían dar al usuario. Lo que encontré por Internet no era fiable, así que use mis contactos y conseguí hablar con un empleado de Bankinter que me proporciono lo que andaba buscando. A cambio de la información, y de forma tácita al menos, me comprometí a asistir a unos cursos gratuitos que impartían en Bankinter sobre el mercado de valores. Introduje la documentación en la aplicación e implemente los cálculos y las pistas que había obtenido.

El Sprint paso de durar de 6 a 7 semanas. Recordemos que habíamos establecido que una semana de trabajo eran aproximadamente 30 horas.

Extras:

Durante el desarrollo de la aplicación, tuve ideas con las que podría mejorar la aplicación. Como cuando termine aún tenía tiempo, implemente algunas de ellas:

- Hice que la aplicación fuera Multiidioma añadiéndole inglés.
- Añadí botones de ayuda que expliquen al usuario la ventana en la que se encuentra, mejorando así la usabilidad.
- En la pantalla del gráfico, coloque el horario del mercado que se estaba viendo para que el usuario sepa cuando tiene que prestarle atención.
- Implemente un botón en la pantalla de inicio llamado “Mi cartera” que sirve para acceder más rápido a valores en los que se han comprado acciones.

7. Pruebas

En este apartado, se explicarán las diferentes pruebas que se han hecho para comprobar el funcionamiento adecuado del entorno desarrollado.

Durante el desarrollo de la aplicación se han ido realizando pruebas en cada iteración de forma constante para comprobar su funcionamiento. A continuación, se mostrarán las pruebas más representativas que se han realizado:

7.1 Recolección de datos de bolsa en tiempo real

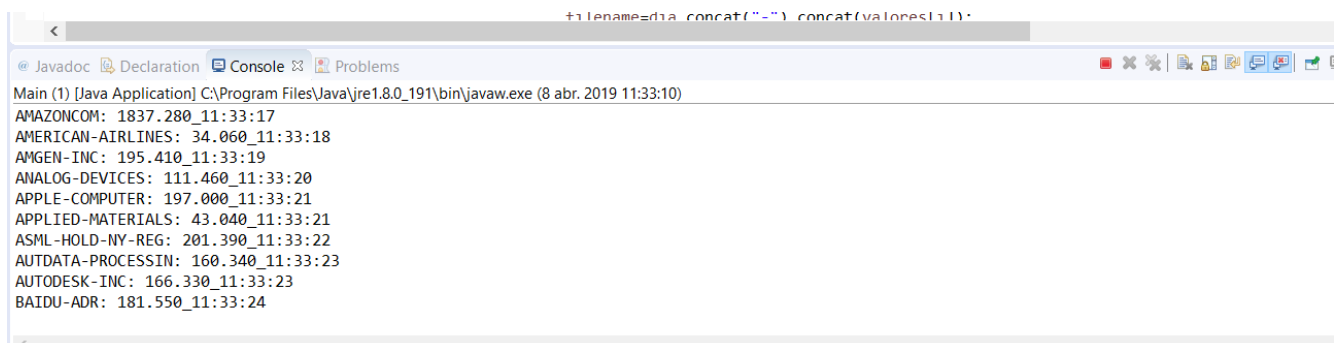
Descripción: Comprobar si los datos se recogen correctamente en tiempo real.

Diseño de la prueba: Obtener en tiempo real los datos del mercado de valores para representarlos gráficamente. La aplicación debe ser capaz de acceder al HTML de páginas que tengan estos datos para extraerlos y guardar el precio más reciente de cada valor en un fichero de texto.

Entorno de la prueba: Java.

Ejecución de la prueba:

La Ilustración 10 muestra el resultado exitoso de la prueba.



```
filename=dir.concat("-").concat(valores[1]);
Main (1) [Java Application] C:\Program Files\Java\jre1.8.0_191\bin\javaw.exe (8 abr. 2019 11:33:10)
AMAZONCOM: 1837.280_11:33:17
AMERICAN-AIRLINES: 34.060_11:33:18
AMGEN-INC: 195.410_11:33:19
ANALOG-DEVICES: 111.460_11:33:20
APPLE-COMPUTER: 197.000_11:33:21
APPLIED-MATERIALS: 43.040_11:33:21
ASML-HOLD-NY-REG: 201.390_11:33:22
AUTODESK-PROCESSIN: 160.340_11:33:23
AUTODESK-INC: 166.330_11:33:23
BAIDU-ADR: 181.550_11:33:24
```

Ilustración 10. Prueba 1

Cada línea que se muestra es el precio a una hora determinada de una acción. Esa misma línea se guarda en un documento de texto. Cada acción tiene su propio fichero de texto en el que se van recogiendo los precios de dicha acción en tiempo real.

Resultado: Este programa va a estar funcionando continuamente en un servidor, es vital que no se detenga y que los datos que recoja sean correctos. Por ello, se dedicó mucho tiempo a esta prueba y a la corrección de los errores detectados.

7.2 Representación gráfica de los datos de bolsa

Descripción: Facilitar la interpretación de los datos mediante su representación gráfica.

Diseño de la prueba: Para hacer este tipo de gráficos en Eclipse tuve que usar librerías externas, las que probé fueron: XChart [8], Charts4j [9], Krysalis JCharts [10], JTStand Chart [11] y JFreeChart. El objetivo es representar los datos de forma clara y que al pasar el ratón por sobre la línea se visualicen los valores correspondientes a ese punto.

Entorno de la prueba: Java.

Ejecución de la prueba:

La Ilustración 11 representa el resultado final. Este grafico cumple con lo requerido.

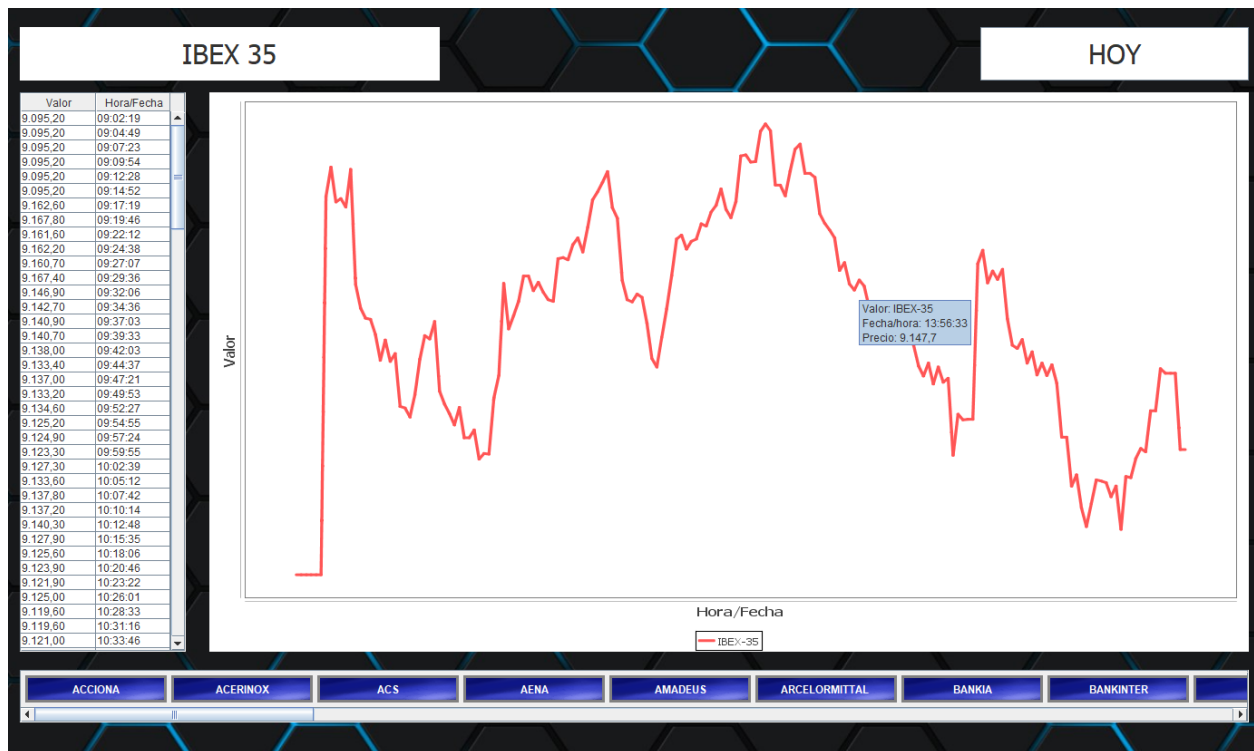


Ilustración 11. Prueba 2

Resultado: Probé varias librerías, pero en todas había algo que no funcionaba correctamente: había casos en los que hacer un zoom para ver los datos más de cerca era difícil de hacer, otros en los que no me permitían cambiar la escala de la tabla y hacia que los datos se vieran como una línea recta, entre otros problemas. De todas, JFreeChart fue la que dio los resultados deseados y la que finalmente se usó en la aplicación. El grafico es una parte vital de la aplicación, por lo que se dedicó mucho tiempo a la investigación de las librerías disponibles para implementarlo.

7.3 Pruebas de comunicación con COBOL

Descripción: Verificar la transferencia de datos entre Java y COBOL.

Diseño de la prueba: El objetivo de esta prueba es verificar que el envío de datos entre Java y COBOL es correcto. La prueba se dará como superada si Java es capaz de mandarle una serie de números a COBOL en un fichero de texto y COBOL es capaz de leerlos, calcular la media de los números y devolver el resultado a Java en otro fichero de texto.

Entorno de la prueba: COBOL.

Ejecución de la prueba:

La Ilustración 12 muestra el resultado final de la prueba.

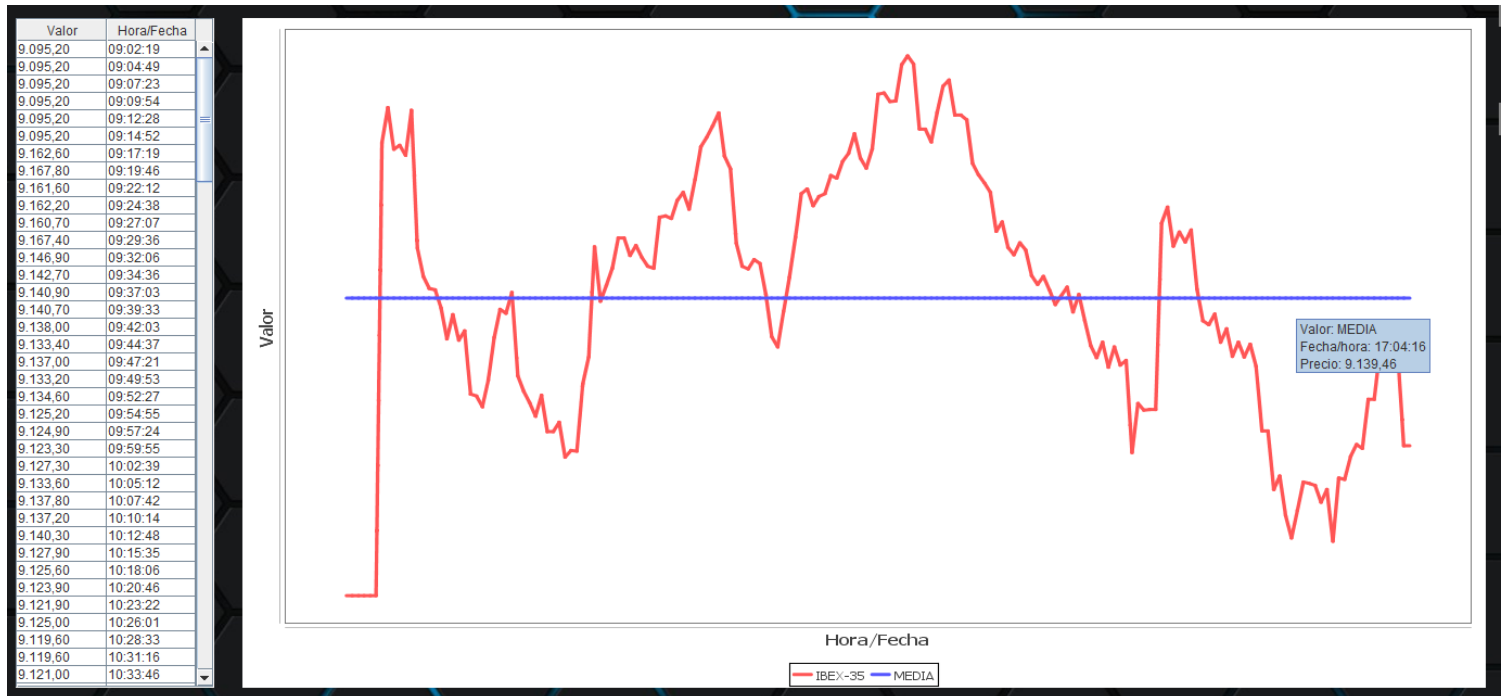


Ilustración 12. Prueba 3

Resultado: En un principio, COBOL tuvo problemas a la hora de leer los datos mayores de 10.

Se hizo que en el fichero se escribieran los números digito a digito y se indicasen las comas y cuando se acaba cada número.

De esta forma, COBOL fue capaz de leerlos y se logró crear la comunicación entre Java y COBOL, completando satisfactoriamente la prueba.

8. Conclusiones y trabajo futuro

8.1 Conclusiones

Para empezar, lo más destacable del proyecto es la cantidad de elementos nuevos a los que me he enfrentado.

Desde el principio tuve interés en COBOL, dediqué mucho tiempo a investigarlo antes de empezar oficialmente este proyecto ya que tenía claro que iba a usarlo. El problema era que no tenía ninguna base desde la que partir ni ninguna herramienta donde poder programar con COBOL.

En un principio estuve investigando su historia y sus usos y posteriormente me puse a buscar academias que lo enseñasen. Solo encontré una en Bilbao y me quise apuntar, pero por alguna razón no tenían un profesor disponible. Era como si no esperasen que alguien se pusiera a estudiar COBOL por lo raro que era y yo les hubiera cogido por sorpresa. Al final tuve aprender por mi cuenta yendo a la biblioteca de la universidad y estudiando los libros que tenían sobre COBOL. Cuando llegó el momento de programar no sabía que herramienta usar para hacerlo. La primera idea fue ir a las oficinas de IBM de Neguri ya que ellos podrían saber algo, pero esas oficinas se encargaban sobre todo de la atención al cliente así que no me pudieron ayudar. Posteriormente di con la página de Micro Focus y encontré su herramienta para programar con COBOL en Java. Estaba pensada para empresas, pero decidí contactar con ellos para ver si me podrían permitir usar su herramienta para el proyecto. Poco después me respondieron y me presentaron su curso académico de COBOL [12] que me daría acceso gratis a su herramienta durante un año y documentación [13] para aprender a usarla. Así fue como pude hacer que COBOL fuera parte del proyecto. Tras usarlo, puedo decir que su fama de ser muy eficiente haciendo cálculos numéricos está justificada.

Tenía COBOL y una herramienta para trabajar con él, pero no tenía claro en que iba a consistir el proyecto. Por lo que había aprendido, COBOL se usaba en empresas grandes con muchos cálculos numéricos. Usarlo en algo relacionado con economía podría ser interesante. Entonces recordé un pequeño juego de la bolsa en el que participe cuando estaba cursando economía y se me ocurrió la idea de un simulador de bolsa. Como no sabía mucho sobre ese tema, fui a despachos de profesores de economía y administración de empresas para coger ideas. A más de uno le sorprendió ver aparecer a un alumno de ingeniería buscando información sobre el mercado de valores. Allí fue donde me dieron la idea de una aplicación que enseñará al usuario a invertir en bolsa ya que las aplicaciones ya existentes están orientadas a usuario con conocimiento previo.

Tiempo después volví a visitar los despachos. Lo que necesitaba esta vez eran documentos sobre como aprender a invertir en bolsa, datos importantes y estrategias de inversión que podría enseñar al usuario de la aplicación. Me propusieron ponerme en contacto con empleados de Bankinter que iban a venir a dar una charla a alumnos de otro grado y a hablarles de un curso sobre la bolsa que imparten gratuitamente en Bankinter [14]. Asistí a la charla y conocí a los empleados con lo que quedé días después para enseñarles el proyecto y decirles lo que me hacía falta. Lo que había hecho les pareció interesante. Me dieron algunos documentos, consejos sobre que podía añadir y también me propusieron asistir al curso de bolsa. Irónicamente, de todos los alumnos que fueron a la charla, fui el único que finalmente asistió al curso a pesar de que la charla inicial no estaba pensada para alumnos de ingeniería informática. De hecho, acabe abriéndome una cuenta bróker para acceder a su aplicación e invertir en serio algún día. El curso fue muy interesante y buena parte de lo que aprendí en él pude aplicarlo en el proyecto, obteniendo así los documentos y la información que buscaba. Además, me di cuenta de que se le debería enseñar más sobre bolsa a los alumnos de informática ya que nuestros conocimientos y nuestra capacidad de automatizar tareas podría darnos una gran ventaja en ese campo.

Relacionado con la automatización, está el problema que supuso lograr obtener los datos de bolsa en tiempo real. Los datos no son gratuitos y las empresas que disponen de ellos han tenido que llegar a un acuerdo con alguien que les facilite esos datos. Fui testigo de ello durante el curso en Bankinter, donde nos presentaron su último acuerdo para obtener datos de empresas europeas. Para lograr obtener los datos en tiempo real tuve que pedir acceso a un servidor de la universidad y hacer un programa que descargue el precio más reciente de distintos valores, lo guarde minuto a minuto y que haga esto todos los días que abra el mercado de valores. Resulto ser mucho más difícil de lo que esperaba.

Como conclusión final puedo decir que he disfrutado del reto de desarrollar este proyecto y estoy muy satisfecho con los resultados obtenidos. También ha sido emocionante explorar materias nuevas.

8.2 Trabajo futuro

Sin duda me gustaría ampliar el contenido. Tras asomar la cabeza en el mundo de la bolsa para realizar este proyecto, me he dado cuenta de lo grande que es y las posibilidades que ofrece. Lo que he desarrollado es solo una pequeña parte en comparación y además me he quedado con ganas de más.

Durante el desarrollo se me han ocurrido algunas ideas y, junto con lo que he aprendido en el curso de Bankinter, he acabado con una buena lista de mejoras que podría implementar en la aplicación:

- Hacer el simulador histórico más especializado: en su estado actual, el simulador histórico solo sube de dificultad cuando se sube de nivel. Esto es algo muy general. Una mejora sería localizar momentos históricos en

los que sea conveniente usar una estrategia en concreto y convertirlos en una prueba para enseñar al usuario a usar esa estrategia en ese momento en particular.

- Desbloquear herramientas y pistas cuando el usuario demuestre entenderlas: actualmente, las pistas y herramientas se desbloquean avanzando en la partida. Se podría buscar la forma de averiguar si el usuario realmente entiende para que sirven y, si es así, las desbloquee como recompensa.
- Estrategias bajistas: en el mercado de valores también es posible ganar dinero cuando los precios están bajando. Para usar estrategias en un mercado bajista no bastaría solo con usar los botones de compra y venta. Se podrían añadir a la aplicación los elementos necesarios para usar dichas estrategias.
- Añadir más valores negociables: las acciones no son lo único que hay en el mercado de valores. Los futuros, los bonos y las permutas, entre otros, también son valores negociables con los que se puede operar y ganar dinero.
- Apalancamiento: es una forma de inversión que se podría añadir. Permite multiplicar los beneficios, pero también las pérdidas.
- Stop loss y stop gain: son dos herramientas muy usadas a la hora de invertir. El stop loss vende las acciones automáticamente si su precio baja demasiado para minimizar pérdidas y el stop gain vende las acciones automáticamente cuando su precio ha alcanzado un valor concreto. Para implementarlas en la aplicación habría que aumentar la comunicación

entre el servidor y la aplicación: el servidor tendría que saber las acciones que ha comprado el usuario en todo momento y vigilar su precio para aplicar el stop loss o el stop gain si es necesario.

También se podrían implementar variantes de estas herramientas como el stop loss dinámico.

- Órdenes de compra: el stop loss y el stop gain son herramientas que venden acciones automáticamente cuando ocurre un evento indicado por el usuario. De la misma forma existen herramientas que compran acciones automáticamente cuando ocurre un evento.
- Añadir más gráficos: ahora mismo solo se dispone de un gráfico de líneas sobre el que se pueden acceder a ciertos cálculos y obtener cierta información. Añadiendo más gráficos se podría ampliar tanto la información como los cálculos que se pueden usar.
- Añadir más operaciones: mientras que en la aplicación hay acceso a unos pocos cálculos, en el mundo profesional existen cientos. Añadir más ofrecería nuevo conocimiento al usuario. Por supuesto, habría que añadir también la teoría correspondiente sobre esos nuevos cálculos.
- Trabajar con varias ventanas: ya que al usarlas se suelen abrir muchos gráficos, las aplicaciones profesionales trabajan con varias ventanas para organizarse mejor y las usan todas a la vez. Si se añadieran nuevos gráficos, también habría que usar más de una ventana para facilitar el manejo de la aplicación.
- Herramienta para poner a prueba estrategias: es algo que descubrí en la aplicación de Bankinter. Es una funcionalidad que permite coger los datos históricos de una acción durante un periodo de tiempo, introducir la estrategia de compra y de venta y ver un informe de cuánto dinero habrías ganado, tu porcentaje de aciertos y número de operaciones que habrías

hecho en ese periodo de tiempo siguiendo esa estrategia. Me resulto muy interesante el hecho de que hubiera una opción avanzada que permitía introducir la estrategia más al detalle usando programación. Definitivamente, implementar algo así sería probablemente la mejora más complicada de todas.

Además de posibles mejoras, también he pensado en modificaciones que se podrían hacer a la aplicación para darle un uso distinto:

Lo primero que se me ocurrió fue añadirle inteligencia artificial a la aplicación para intentar que gane dinero por su cuenta. Las estrategias y las herramientas que anteriormente eran para enseñar al usuario a invertir, las podría usar la inteligencia artificial para tomar decisiones.

La segunda idea fue hacer que la aplicación sirviera de asistente para invertir en bolsa. La aplicación pasaría de enseñar al usuario a ayudar al usuario a invertir en bolsa. Estaría observando constantemente los cambios en el mercado de valores buscando oportunidades de compra y de venta. De hecho, ese tipo de aplicaciones son usadas ya por inversores profesionales para buscar oportunidades de inversión.

Referencias

- [1]<https://proyectosagiles.org/que-es-scrum/>
- [2]<https://www.microfocus.com/es-es/solutions/cobol-development/>
- [3]https://broker.bankinter.com/www2/broker/es/analisis/broker_grafico/informacion?gneisnt=on&vgnextnoice=1
- [4]<https://play.google.com/store/apps/details?id=com.gneis.bankinter.broker&hl=es>
- [5]https://portal.kutxabank.es/cs/Satellite/kb/es/banca_personal/productos/bolsa_3
- [6]<https://www.eleconomista.es/indice/IBEX-35>
- [7]<http://www.java2s.com/Code/Java/Chart/JFreeChartLineChartDemo5showingtheuseofacustomdrawingsupplier.htm>
- [8]<https://knowm.org/open-source/xchart/>
- [9]<https://code.google.com/archive/p/charts4j/>
- [10]<http://jcharts.sourceforge.net/>
- [11]<https://mvnrepository.com/artifact/org.codehaus.jtstand/jtstand-chart>
- [12]<https://www.microfocus.com/es-es/cobol-academic-program/>
- [13]<https://www.microfocus.com/es-es/campaign/visual-cobol-book/>
- [14]<https://www.bankinter.com/banca/ahorro-inversion/mercados>

ANEXO: Manual de usuario

Introducción:

Esta es una guía sobre cómo usar el simulador de bolsa. Esta aplicación es un juego educativo que enseña los conceptos básicos sobre cómo invertir en bolsa. Una vez introduzcas tu nombre y tu idioma, empezaras una partida en el simulador.

Al inicio, estarás en el nivel 1, tendrás 0 puntos de experiencia y contarás con 10.000\$ para usar en el juego. Para avanzar tendrás que usar ese dinero en los simuladores de la bolsa que hay disponibles, invirtiéndolo y consiguiendo beneficios.

Existen dos simuladores en la aplicación: el primero funciona con los datos de la bolsa real en tiempo real y el segundo usa datos históricos de la bolsa.

Podrás invertir en cualquiera de los dos como si fuera la vida real. Cada vez que obtengas beneficios por una inversión se te darán puntos de experiencia. Al acumular los suficientes puntos de experiencia subirás de nivel.

Si sabes poco sobre el mercado de valores, hay varios documentos disponibles dentro de la aplicación con teoría sobre la bolsa que te pueden ayudar.

Subir de nivel y acumular experiencia sirven para desbloquear contenido. Al comienzo solo tendrás acceso al contenido más básico. Tendrás que subir de nivel para desbloquear nueva teoría, nuevos datos históricos para el simulador de datos históricos y herramientas y consejos que te puedan ayudar en el simulador en tiempo real.

En todo momento es posible reiniciar la partida y cambiar de nivel.

Identificarse:

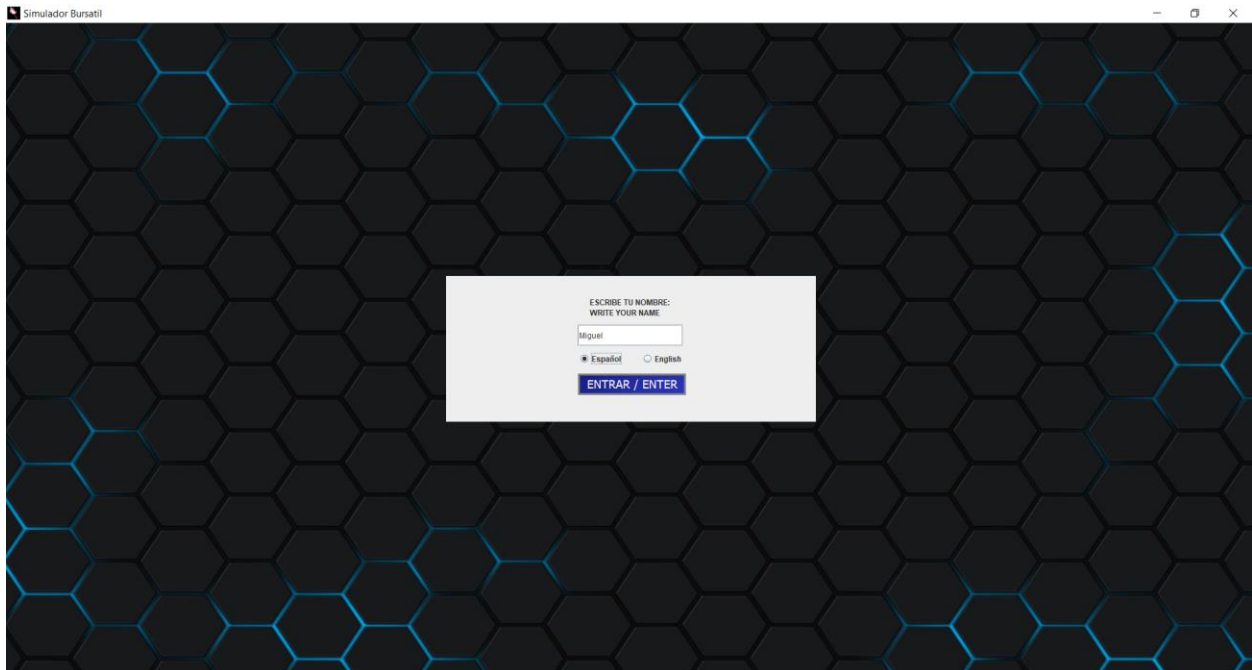


Ilustración 13. Pantalla de login

Al entrar por primera vez en la aplicación se te mostrará la pantalla de login. Tienes que introducir un nombre y elegir el idioma para poder entrar en la aplicación:

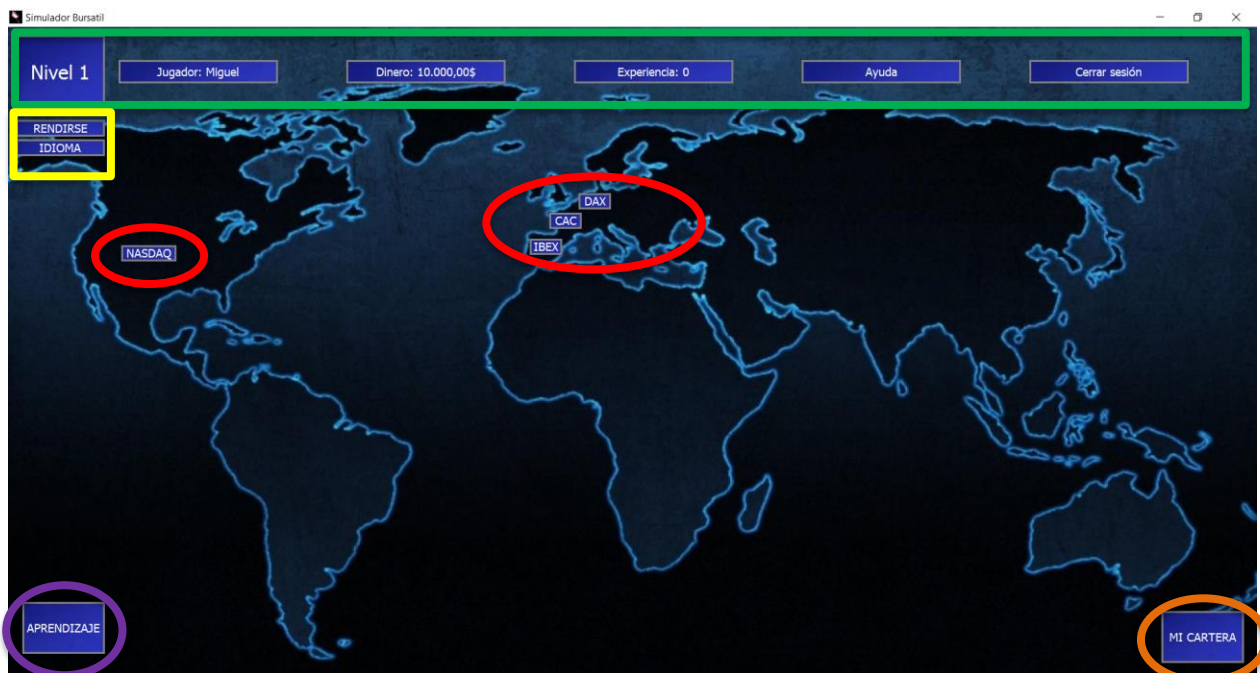


Ilustración 14. Pantalla de inicio

Una vez dentro serás saludado por el "Lobo", que será tu guía dentro de la propia aplicación.

Estarás en la pantalla principal en la que hay varios botones. En la Ilustración 14 podrás ver los botones rodeados por círculos y cuadrados de colores. Esto se ha hecho para poder dividirlos por grupos y explicarlos mejor:

- Los que tienen un círculo rojo representan mercados. Pulsar sobre cualquiera de ellos te llevará al mercado que representa. Como se puede ver, están esparcidos por un mapa del mundo. Cada botón está encima del país al que pertenece el mercado al que lleva. Al entrar en un mercado, podrás acceder a todos los valores que componen dicho mercado, pero no podrás acceder a valores de otros mercados. Para cambiar de mercado tendrás que volver a la página de inicio y seleccionar el mercado que quieras ver. En el apartado "Simulador en tiempo real" profundizaremos más en este tema.
- El que tiene un círculo naranja representa la cartera del jugador. Al pulsarlo te mostrará, si tienes alguna, las acciones que has comprado. Al hacer seleccionar una de ellas te llevará directamente al mercado al que pertenece y te mostrará su gráfico.
- Los que están en un cuadrado verde muestran información sobre el estado de tu partida, ofrecen ayuda y permiten cerrar la sesión. Estarán siempre disponibles y el botón de ayuda permite acceder a una pequeña guía sobre la pantalla que estés viendo.
- El botón con el círculo morado lleva a la teoría y al simulador con datos históricos. Hablaremos de ambos en los apartados "Teoría" y "Simulador de datos históricos".
- Los botones en el cuadrado amarillo permiten hacer cambios en la partida, se profundiza en ellos en el apartado "Ajustes".

Simulador en tiempo real:

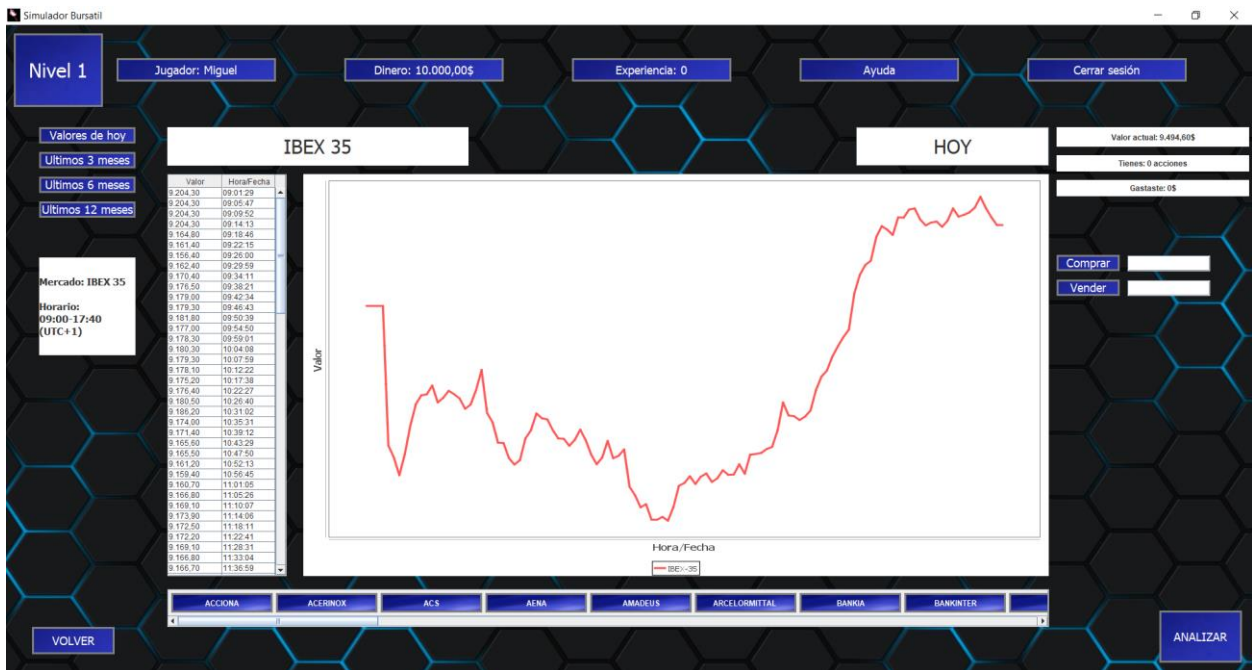


Ilustración 15. Grafico

Aquí nos muestra el grafico del IBEX 35, en el centro vemos un gráfico que nos muestra los cambios en el precio de la acción y junto a él tenemos una tabla que muestra esos mismos datos por escrito. Es posible también ver los datos en un punto concreto del grafico pasando el ratón por encima de la línea del gráfico.

En este momento estamos viendo los datos de hoy, como se indica encima de la esquina superior derecha del gráfico. A la izquierda del grafico podemos ver cuatro botones que nos permiten cambiar el periodo de tiempo que estamos viendo.

Justo debajo de esos botones tenemos el horario del mercado que estamos viendo. Es muy útil para saber cuándo hay que estar atento a la bolsa.

Debajo del grafico tenemos una larga fila de botones. Cada uno de ellos representa las empresas de este mercado. Al hacer click en uno de ellos te llevará al gráfico de esa empresa. Hay muchas empresas, así que se recomienda mover el scroll de izquierda a derecha para poder verlas todas.

A la derecha del gráfico se nos muestra el precio actual de la acción, cuántas acciones tienes y cuánto te costó comprarlas. También se nos muestran dos botones. Para adquirir acciones, pon un número en el campo que está al lado del botón "Comprar" y luego haz click en el botón. Para vender tienes que hacer lo mismo solo que usando el botón "Vender" y el campo que tiene al lado. Cada vez que obtengas beneficios se obtienen puntos de experiencia. Cuando tengas bastante experiencia, subes de nivel.

Puedes retroceder a la pantalla anterior pulsando en el botón "Volver" que hay en la parte de abajo a la izquierda.

Analizar gráficos:

Ahora pulsaremos el botón "Analizar" que esta abajo a la derecha para realizar cálculos sobre el grafico:

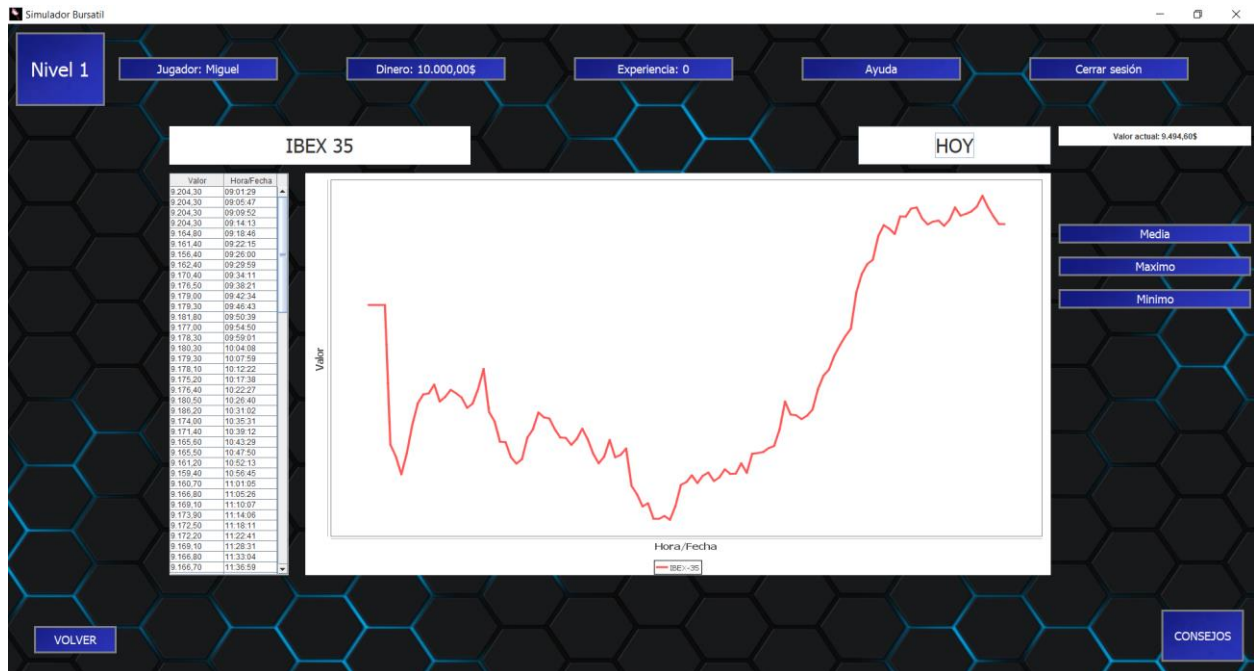


Ilustración 16. Análisis

Esta pantalla es muy similar a la anterior. Alrededor del grafico encontraremos botones que permiten realizar cálculos sobre el grafico. Vamos a probar a hacer el promedio del gráfico con el botón "Media".

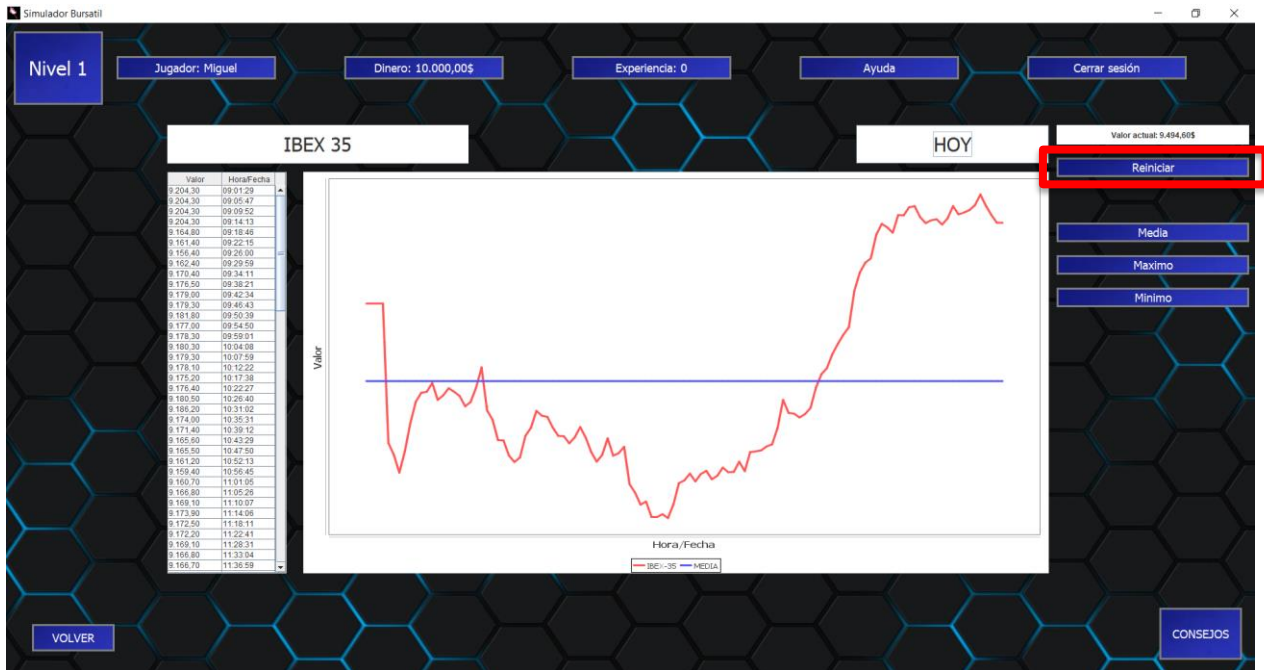


Ilustración 17. Ejemplo de la calculo

Tras unos instantes se mostrará el mismo grafico que antes solo que con una línea que representa la media. Además, ha aparecido un nuevo botón en el que pone "Reiniciar", que permite devolver al grafico a su estado original. Abajo a la derecha hay un botón en el que pone "Consejos". Si lo seleccionas, el "Lobo" te dará algunas pistas sobre la empresa.

Tanto las estrategias como las operaciones que se pueden realizar en el grafico se obtienen conforme se sube de nivel y se obtiene experiencia.

Ahora regresaremos a la pantalla de inicio pulsando los botones de "Volver" que hay en cada pantalla abajo a la izquierda.

Aprendizaje:

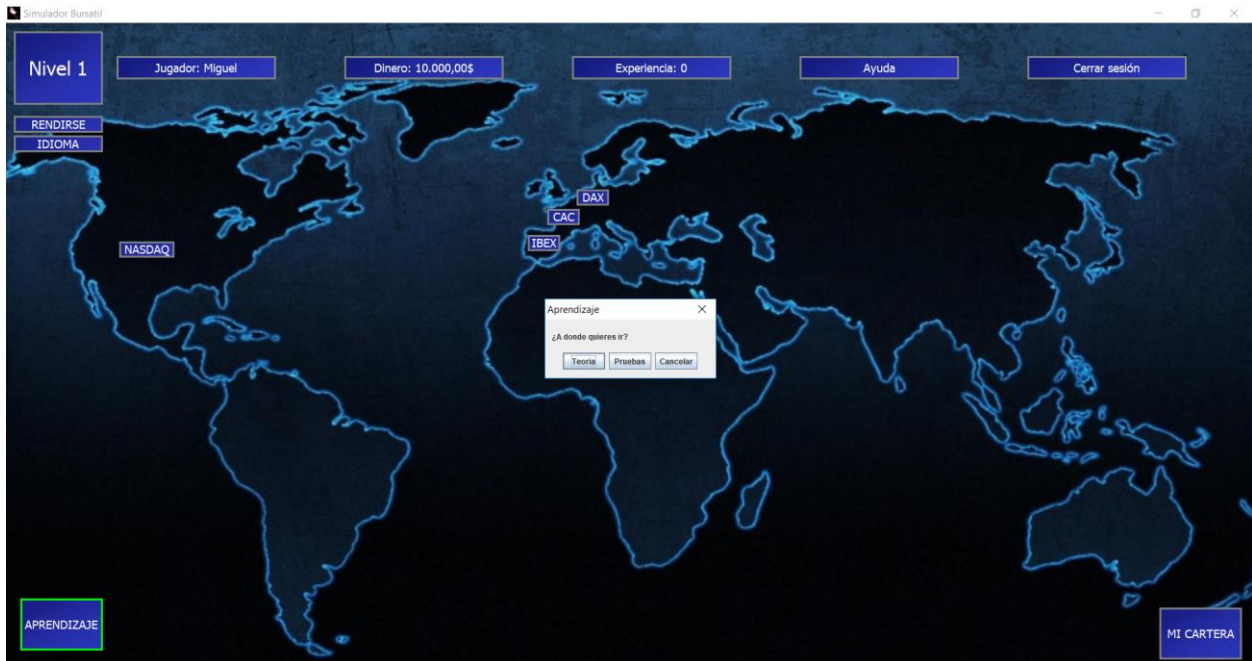


Ilustración 18. Teoría

Ahora nos fijaremos en el botón de "Aprendizaje" que hay en la esquina inferior izquierda. Al pulsarlo ofrece dos opciones: ir a la teoría o a las pruebas, que es el simulador con datos históricos.

Teoría:

Empezaremos yendo a la teoría:

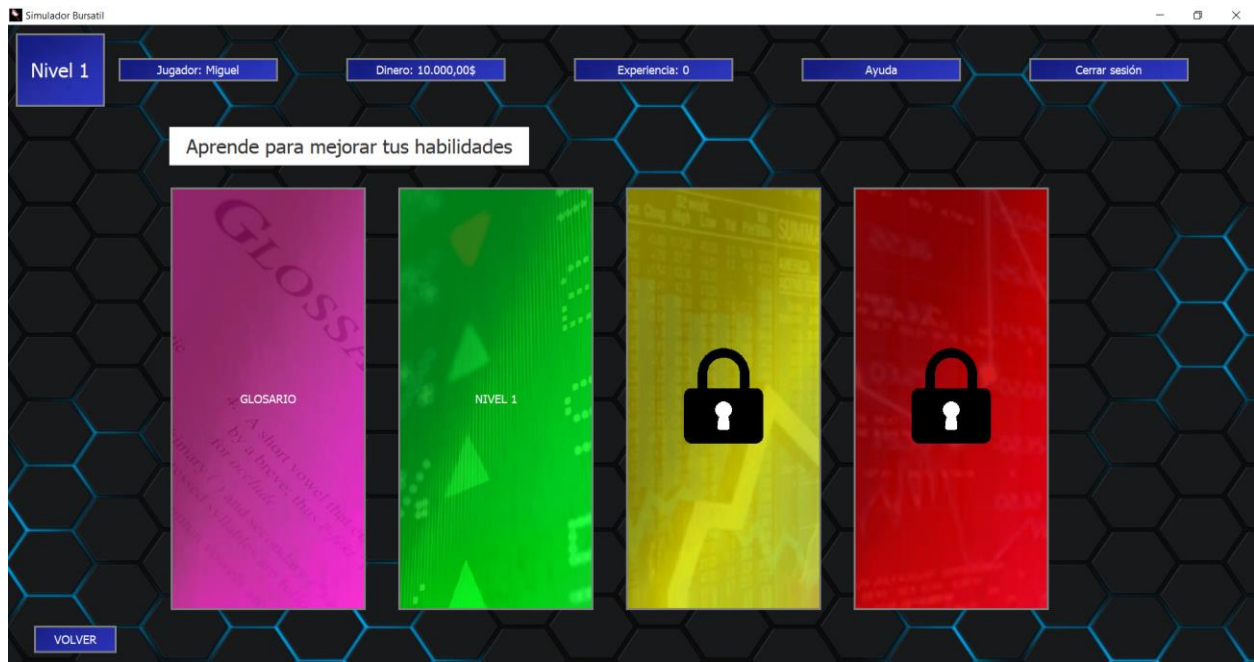


Ilustración 19. Selección de teoría

Aquí se te mostrarán cuatro grandes botones. El de la izquierda lleva al glosario y el resto a la documentación correspondiente a su nivel. Cada vez que subas de nivel podrás acceder a una parte de la documentación nueva.

Para este ejemplo haremos click en el nivel uno.

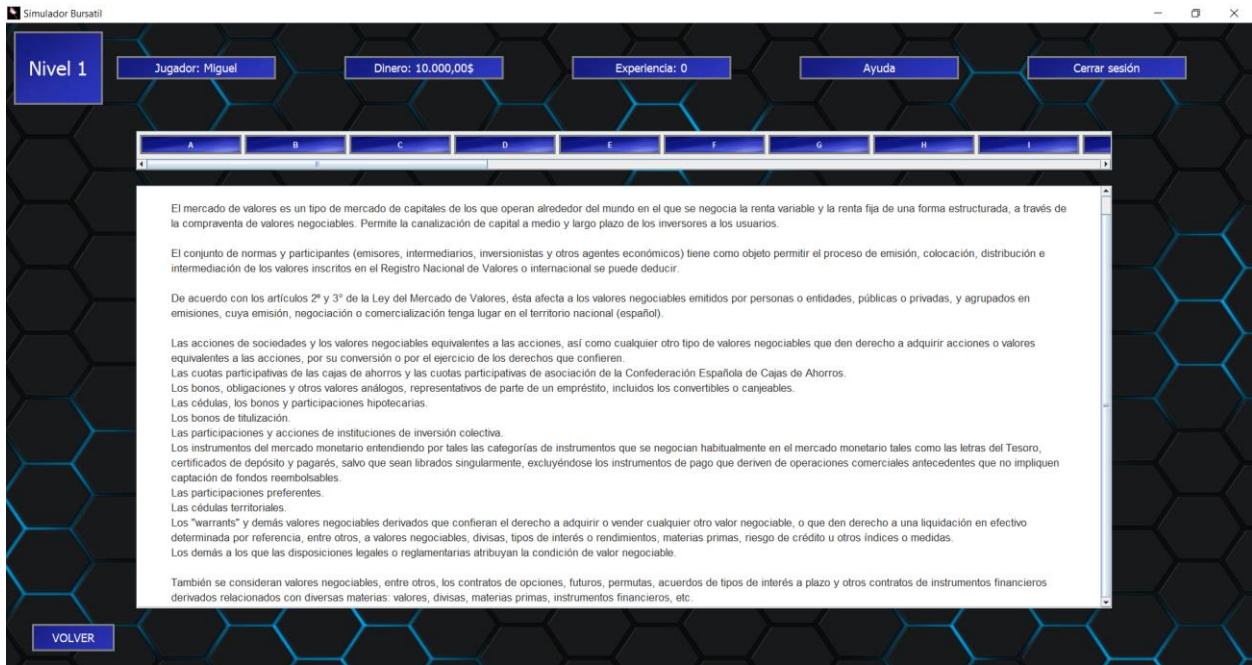


Ilustración 20. Documentación

El resto de niveles y el glosario tienen el mismo aspecto básico. En el centro está el texto y con los botones que hay encima puedes ir de un tema a otro.

Simulador de datos históricos:

Ahora iremos al simulador de datos históricos:

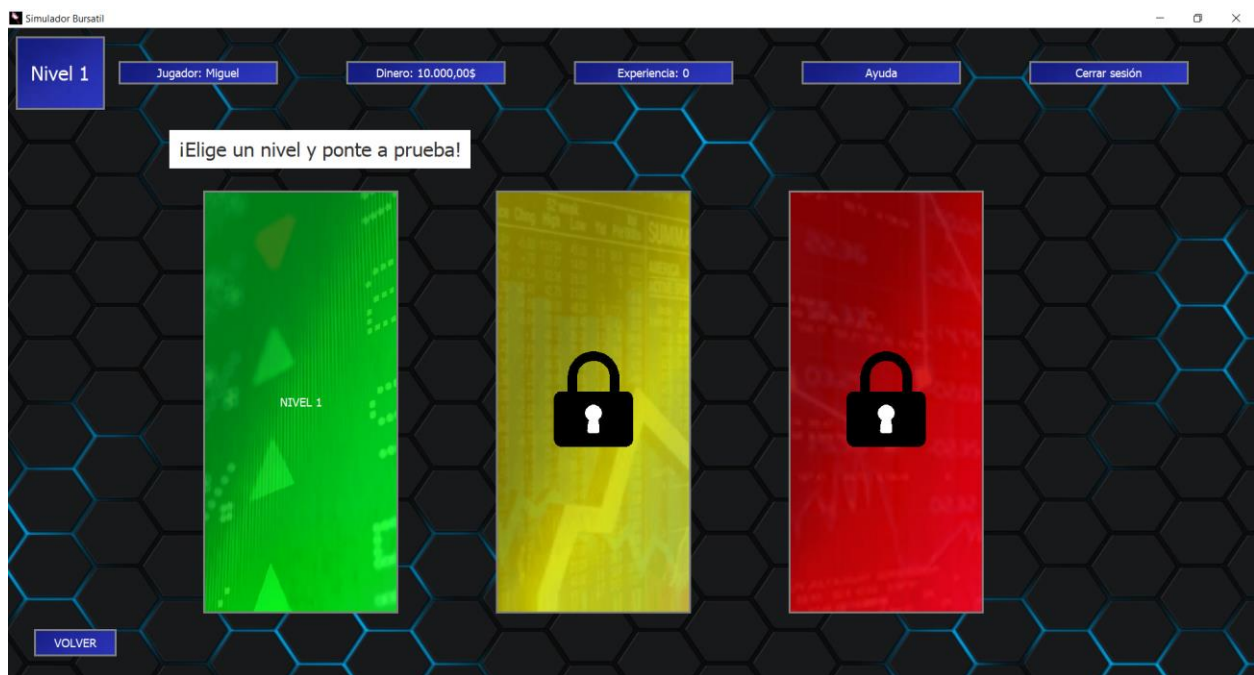


Ilustración 21. Selección de pruebas

Es muy similar a la pantalla de la teoría. En el simulador con datos históricos podrás realizar diferentes pruebas en las que simulas que inviertes usando un periodo de tiempo que ya ha pasado. Hay tres niveles disponibles, pero al iniciar la partida solo podrás entrar al nivel 1 que usa los datos históricos más predecibles. Cuando subes de nivel desbloqueas pruebas nuevas.

Veamos en que consiste una de esas pruebas, vamos a pulsar en el nivel 1:

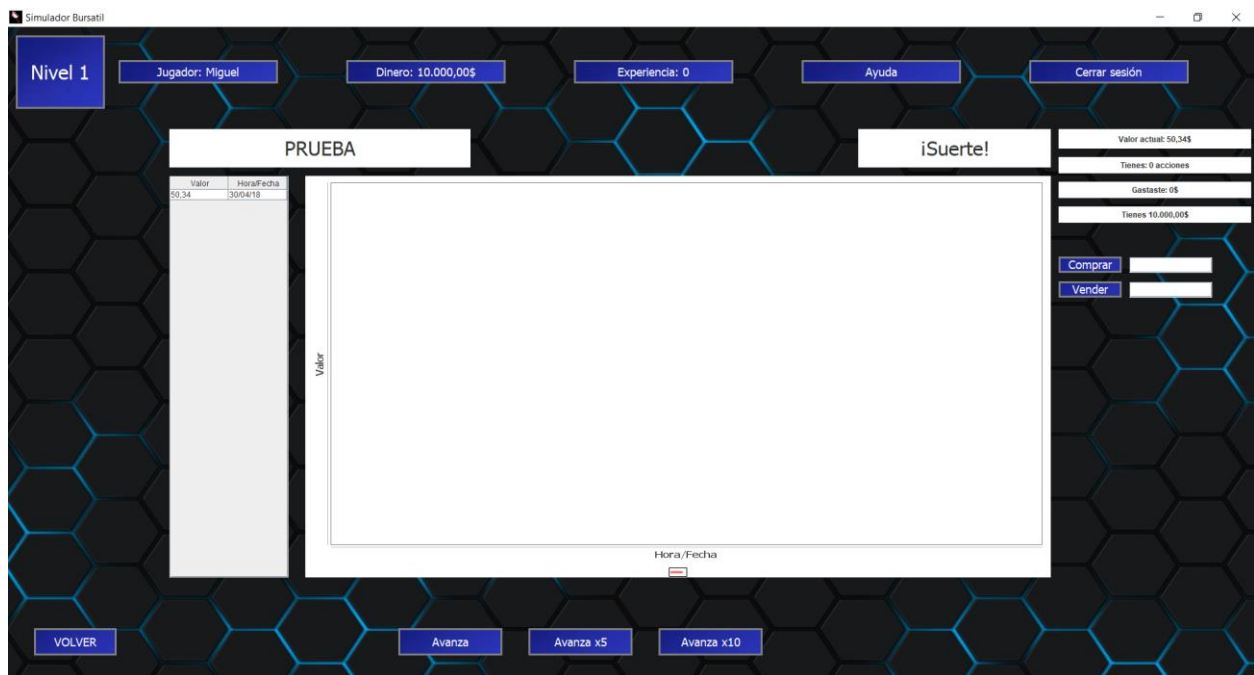


Ilustración 22. Prueba

Ahora podremos ver el simulador con datos históricos. La pantalla es muy similar a la del simulador en tiempo real.

Para superar la prueba hay que obtener beneficios. Para cada prueba se te prestarán 10.000\$ para que los inviertas. Se podrá ver de cuánto dinero dispones a la derecha del gráfico, junto con el valor actual de la acción, cuantas acciones tienes y cuanto te han costado. De la misma forma que antes puedes usar los botones de compra y de venta para obtener o deshacerte de acciones.

La mayor diferencia son los botones de la parte inferior. Al pulsar uno de ellos se revelarán más partes del gráfico. De esta forma puedes simular que el tiempo avanza. La prueba termina cuando se haya completado el grafico.

Si superas la prueba se te darán puntos de experiencia.

Ahora volveremos a la pantalla principal para ver las dos últimas funciones importantes.

Ajustes:

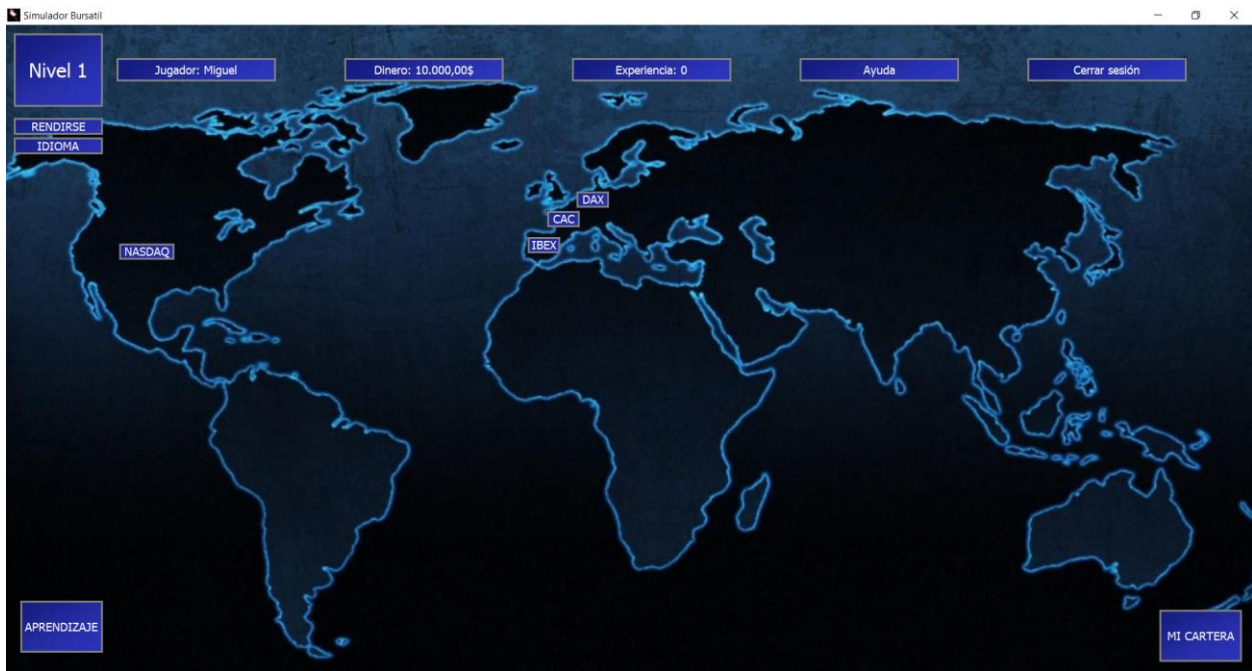


Ilustración 23. Pantalla de inicio

En la parte de arriba a la derecha debajo del nivel, vemos dos botones: "RENDIRSE" e "IDIOMA".

"IDIOMA" sirve para cambiar el lenguaje de la aplicación. Se puede alternar entre inglés y español.

El botón "RENDIRSE" permite reiniciar la partida. Esto se ha pensado principalmente para situaciones en las que estés en bancarrota. Al reiniciar, volverás a la pantalla de login que hemos visto al principio del manual y podrás empezar de nuevo.