



ESCUELA POLITÉCNICA NACIONAL
FACULTAD DE INGENIERIA DE SISTEMAS
SOFTWARE
CONSTRUCCIÓN Y EVOLUCIÓN DE SOFTWARE
PROYECTO [2B]
GRUPO N°6

Integrantes: Erick Cabezas, Kenny Pinchao, Kevin Cano

Fecha: 6/03/2024

ANÁLISIS DE CAMBIOS

I. OBJETIVOS:

- Identificar los artefactos de la primera entrega del producto.
- Realizar el análisis de cambio para agregar la actualización solicitada a los artefactos de la primera entrega.

II. DESARROLLO

Artefactos del sistema:

- Stakeholders.
- Alcance del proyecto.
- Configuración del proyecto.
- Diseño de base de datos.
- Artefactos XP (Historias de usuario)
- Planificación reléase.
- TDD.

Descripción de los cambios realizados.

Alcance del proyecto:

El alcance del proyecto de software se refiere a la definición y delimitación de todas las actividades, características, funcionalidades y entregables que serán incluidos en el desarrollo del software, por lo tanto, al realizar una actualización al producto software aumentan sus funcionalidades afectando así al alcance del proyecto.

El sistema deberá permitir en la segunda entrega:

- Registrar una nueva acción.
- Buscar una acción ya registrada.
- Visualizar una lista con todas las acciones registradas.
- Calcular el costo total de compra = Precio de compra por acción*cantidades de acciones.
- Visualizar el % de cambio de la acción comprada.
- Visualizar la ganancia o pérdida respecto al % de cambio

A continuación, se presentan los requerimientos que cubren las necesidades del usuario:

Tabla 1. Requerimientos funcionales para la aplicación en la segunda entrega.

Código	Requerimiento
Requerimientos funcionales	
RFP1	La aplicación debe permitir al usuario registrar la compra de acciones
RFP2	La aplicación debe permitir al usuario observar una lista de las acciones compradas
RFP3	La aplicación debe permitir buscar con su nombre una acción registrada por el usuario.
RFP4	La información que debe mostrarse de las acciones es: el nombre de la acción, fecha de compra, precio de compra por acción, cantidades de acciones y costo total de compra.
RFP5	La información que ingresa el usuario al registrar una acción es el nombre de la acción, fecha de compra, precio de compra por acción y cantidades de acciones
RFP6	El costo total de compra cuando del usuario registra una compra de acción se calcula automáticamente.
RFP7	La interfaz de usuario debe poderse cambiar a modo oscuro.

RFP8	Las acciones registradas también se podrán eliminar del registro.
RFP 9	Las acciones registradas deben tener un % de cambio respecto al precio que se compró inicialmente.
RFP10	Respecto al % de cambio la aplicación permite visualizar si existe una ganancia o pérdida de la acción respecto a su precio original.

Configuración del proyecto:

La arquitectura de la primera entrega basada en componentes se mantiene, sin embargo, en el back se conectó a un websocket para conseguir información en tiempo real de las acciones y poder hacer el cálculo de cambio, ganancia y pérdida de cada una de las acciones.

Arquitectura del WebSocket:

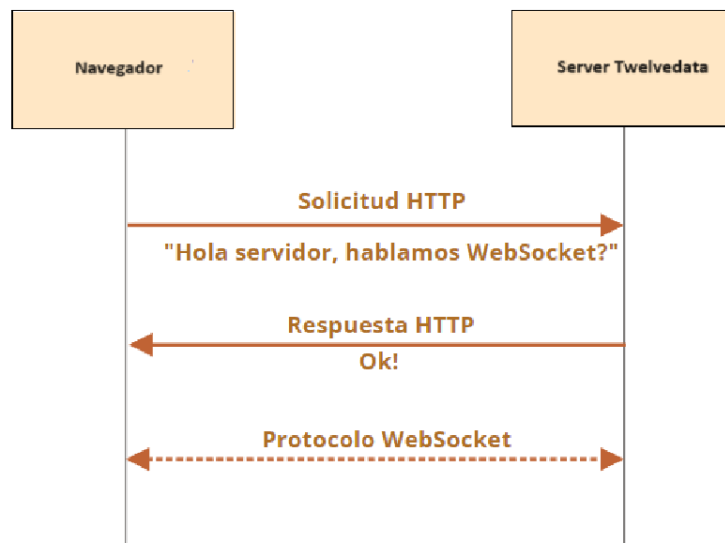


Ilustración 1. arquitectura websocket.

Conexión con websocket al servidor de Twelvedata:

WebSockets es una tecnología avanzada que hace posible abrir una sesión de comunicación interactiva entre el navegador del usuario y un servidor. Con esta API, puede enviar mensajes a un servidor y recibir respuestas controladas por eventos sin tener que consultar al servidor para una respuesta [3].

```

import WebSocket from 'ws';

// Export the WebSocket server
export const wss = new WebSocket.Server({ port: 8081 });

// Handle incoming connections
wss.on('connection', function connection(ws) {
  console.log('Cliente conectado');

  // Handle incoming messages from the client
  ws.on('message', function incoming(message) {
    console.log('Mensaje recibido del cliente:', message);
  });
});

// Export the function to send messages to connected clients
export function sendMessageToClients(message: any) {
  // console.log('Enviando mensaje a los clientes:', message);
  const jsonMessage = JSON.stringify(message);

  wss.clients.forEach(function each(client) {
    if (client.readyState === WebSocket.OPEN) {
      //console.log('Enviando mensaje al cliente:', jsonMessage);
      client.send(jsonMessage);
    }
  });
}

```

Ilustración 2. conexión con websocket al servidor Twelvedata.

También se agregó un nuevo módulo para la configuración de un servidor websocket, para brindar servicio de websocket al front y este con JS realice los cambios correspondientes para que el usuario pueda visualizar los cambios del precio de la acción respecto a lo original y se presente la ganancia o pérdida.

```

import AccionesService from '../services/acciones.service';
import AccionesModel from '../db/models/acciones.model';
import { AccionesCreationAttributesI, AccionesAttributesI } from '../type';
import { sendMessageToClients } from '../websocket';
import WebSocket from 'ws';

const subscribedSymbols = new Set();

export const socket = new WebSocket('wss://ws.twelvedata.com/v1/quotes/price?apikey=2f8c9936a0ec4f6381a9cf6522e9f408');

socket.addEventListener('open', function (event) {
  AccionesModel.findAll().then((acciones: any) => {
    acciones.map((accion: any) => {
      subscribe(accion.siglas_accion);
    });
  });
});

socket.addEventListener('message', function (event) {
  const data = JSON.parse(event.data as any);
  const receivedSymbol = data.symbol;

  AccionesService.getAcciones().then((acciones: any) => {
    acciones.map((accion: AccionesAttributesI) => {
      if (accion.siglas_accion === receivedSymbol) {
        accion.cambio = (data.price - accion.precio_compra) / accion.precio_compra;
        accion.ganancia_perdida = accion.costo_total * (1 + accion.cambio);
        //console.log(accion);
      }
    });
  });
});

```

Ilustración 3. creación del servidor websocket para el front.

Diseño de base de datos:

Para la segunda entrega el sistema presenta el porcentaje de cambio y la ganancia o pérdida de la acción, por lo tanto, se aumentan las filas cambio y ganancia_perdida a la entidad Acción.

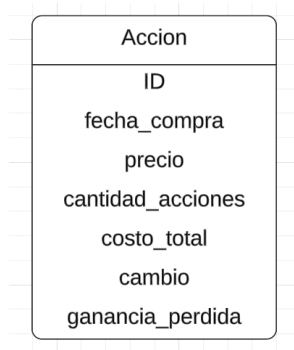


Ilustración 4. Modelo entidad relación de la base de datos del sistema.

Artefactos XP:

Al realizar la actualización del producto software se generan una nueva épica y de esta salen 2 nuevas historias de usuario que serán implementadas en el sistema.

Épica 4: Como usuario, deseo tener el porcentaje de cambio de las acciones compradas para saber si se da una ganancia o pérdida respecto al porcentaje dado.\

Título: Porcentaje de cambio	Número: 7
Historia de usuario: Como usuario, deseo que en el listado de las acciones registradas se muestre el porcentaje de cambio.	
Criterios de aceptación: Dado que ingrese a la aplicación. cuando inicie, Entonces se muestra las acciones compradas con el dato de porcentaje de cambio.	

Título: Ganancia o pérdida respecto al porcentaje de cambio	Número: 8
Historia de usuario: Como usuario, deseo que en el listado de las acciones registradas se muestre la ganancia o pérdida que genera mi acción respecto a su porcentaje de cambio.	
Criterios de aceptación: Dado que ingrese a la aplicación. cuando inicie, Entonces se muestra las acciones compradas con el dato de porcentaje de cambio y respecto a este el dato de ganancia o pérdida que me genera mi acción.	

Planificación reléase

Con la aparición de las 2 nuevas historias de usuario en los artefactos XP se reorganiza la planificación del release, primero el objetivo del release cambia y se aumenta una iteración más (iteración 4) en la que se implementaran estas 2 historias de usuario.

Objetivo del reléase: Automatizar en un 100% la administración de las acciones compradas por el usuario.

Iteración 4

Objetivo: Permitir al usuario visualizar la ganancia o pérdida de las acciones compradas con respecto al porcentaje de cambio de estas acciones en el mercado.

Número	8	Estimado: 7 horas
Título	Porcentaje de cambio	
Historia de usuario	Como usuario, deseo que en el listado de las acciones registradas se muestre el porcentaje de cambio.	
Criterio de aceptación	Dado que ingrese a la aplicación. cuando inicie, Entonces se muestra las acciones compradas con el dato de porcentaje de cambio.	
Tareas	T2. Conectarse al servidor websocket para obtener el cambio de del precio de las acciones compradas. (Estimado: 3 horas) T1. Agregar en el listado de las acciones una columna para mostrar el porcentaje de cambio. (Estimado: 2 horas) T3. Actualizar automáticamente el porcentaje de cambio con el cambio de precio de las acciones obtenido desde el servidor websocket (Estimado: 2 horas).	

Número	9	Estimado: 5 horas
Título	Ganancia o pérdida respecto al porcentaje de cambio	
Historia de usuario	Como usuario, deseo que en el listado de las acciones registradas se muestre la ganancia o pérdida que genera mi acción respecto a su porcentaje de cambio.	
Criterio de aceptación	Dado que ingrese a la aplicación. cuando inicie, Entonces se muestra las acciones compradas con el dato de porcentaje de cambio y respecto a este el dato de ganancia o pérdida que me genera mi acción.	
Tareas	T1. Agregar la columna para mostrar la ganancia o pérdida de la acción respecto al porcentaje de cambio. (Estimado: 1 horas)	

	T2. Realizar el cálculo de ganancia o pérdida de la acción respecto a el porcentaje de cambio. (Estimado: 2 horas)
	T3. Actualizar automáticamente la columna de ganancia o pérdida respecto a los cambios de la columna de porcentaje de cambio de la acción. (Estimado: 2 horas)

Resumen de iteración:

No	Titulo	Estimado
8	Porcentaje de cambio	7
9	Ganancia o pérdida respecto al porcentaje de cambio	5
Estimación Total:		12
Tiempo Real:		40
Velocidad Total:		0.3

TDD

Con las nuevas funcionalidades se aumentan las pruebas unitarias, a continuación, se muestra la prueba unitaria que muestra la conexión de la app con el servidor websocket de Twelvedata.

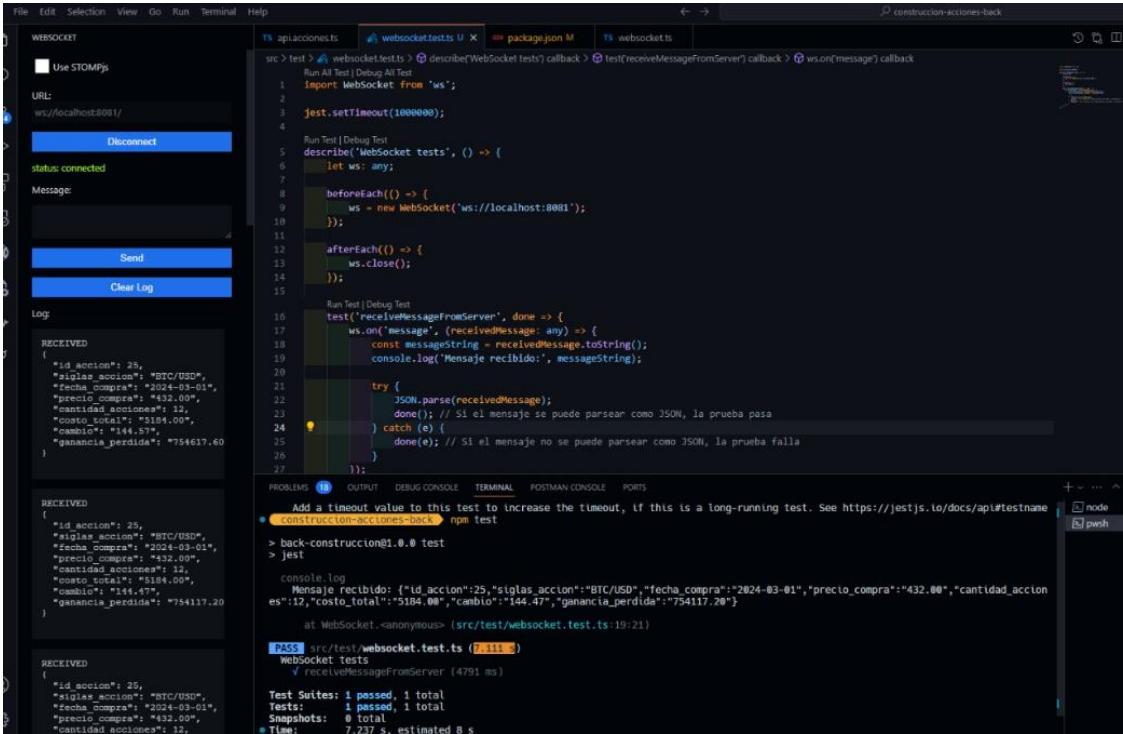


Ilustración 5. prueba unitaria para probar la conexión de la app con el websocket de twelvedata.