



Memòria del Projecte

Dan Maldonado
Samuel Hernán
Curs 2024 - 2025
2n DAW

1. Elements Preliminars

1.1. Resum Executiu

Síntesi del projecte

SODA és una aplicació web interactiva dissenyada per simular inversions en temps real. L'objectiu és oferir una experiència educativa i pràctica als usuaris que vulguin aprendre com funciona el mercat de valors. L'aplicació es connecta a través de WebSocket per mostrar el preu actualitzat de les accions, i permet operacions com comprar, vendre, seguir notícies, i gestionar el capital virtual de cada usuari.

Objectius principals

- Desenvolupar una eina formativa i entretinguda per a la simulació d'inversions.
- Permetre la compra i venda d'accions amb preus en temps real.
- Afavorir la interacció social amb funcionalitats de publicació, comentaris i "likes".
- Gestionar perfils d'usuari amb cartera, historial i saldo disponible.
- Proporcionar un entorn modern, responsiu i intuïtiu, mitjançant Next.js al frontend i Express al backend.

Resultats clau aconseguits

- S'ha implementat un sistema complet de **compra/venda d'accions** amb saldo virtual.
- El sistema de **WebSocket** transmet preus actualitzats de manera contínua.
- Cada acció disposa d'una **secció de notícies** relacionades amb l'empresa.
- Es disposa d'un **dashboard funcional**, cartera personal, i **historial de transaccions**.
- S'ha creat un apartat de **posts** amb sistema **CRUD**, likes i comentaris entre usuaris.
- El sistema de registre i perfil permet **gestionar saldo**, afegint o retirant fons.
- El projecte es troba publicat al repositori: <https://github.com/Daneteee/SODA>

1.2. Índex

1. Elements Preliminars	2
1.1. Resum Executiu	2
1.2. Índex	3
2. Cos Principal	4
2.1. Introducció	4
2.2. Planificació	6
2.3. Desenvolupament del Projecte	7
2.4. Avaluació i Resultats	11
3. Elements Finals	13
3.1. Conclusions	13

2. Cos Principal

2.1. Introducció

Context del projecte

En l'actualitat, l'interès per les finances personals i la inversió borsària ha augmentat de manera significativa, especialment entre joves i estudiants. Tanmateix, molts d'aquests potencials inversors no disposen de les eines o l'experiència per practicar sense riscos. **SODA** (Simulador d'Operacions Dinàmiques d'Accions) neix amb l'objectiu d'oferir una plataforma educativa i interactiva per aprendre a invertir en un entorn segur i en temps real.

Aquest projecte s'emmarca dins de l'àmbit educatiu i tecnològic, i busca fusionar l'aprenentatge pràctic amb una experiència d'usuari moderna. Ha estat desenvolupat com a treball de síntesi del cicle de Desenvolupament d'Aplicacions Web (DAW), amb una orientació clara cap a la simulació de mercats financers.

Objectius detallats

Els objectius específics que s'han definit per a aquest projecte són:

- **Simular el mercat de valors** amb dades de preus en temps real mitjançant connexions WebSocket.
- **Permetre l'operativa bàsica d'inversió:** compra i venda d'accions amb saldo virtual.
- **Crear un espai personalitzat per a cada usuari** amb cartera, historial de transaccions i perfil configurable.
- **Afavorir la interacció entre usuaris** mitjançant una secció de posts amb comentaris i likes.
- **Incloure notícies empresarials** associades a cada acció per fomentar decisions informades.
- **Fer possible la gestió de fons virtuals**, permetent afegir i retirar saldo.
- Desenvolupar una interfície **moderna, responsiva i intuïtiva** amb Next.js.

Abast del projecte

El projecte cobreix totes les funcionalitats bàsiques d'una plataforma d'inversió simulada, amb una clara divisió entre frontend (Next.js) i backend (Express.js amb WebSocket).

Funcionalitats implementades:

- Registre i inici de sessió d'usuaris
- Compra i venda d'accions
- Visualització del preu en temps real
- Gestió de cartera i historial
- Secció de publicacions: crear, editar, esborrar, comentar i donar like
- Gestió de saldo virtual
- Panell de control (dashboard) amb informació rellevant de l'usuari

Metodologia utilitzada

Per al desenvolupament del projecte, s'ha seguit una **metodologia iterativa i incremental**, basada en els principis àgils:

- **Divisió del projecte en funcionalitats (mòduls)**, desenvolupades de forma independent i integrades progressivament.
- Ús de **GitHub** com a sistema de control de versions i repositori col·laboratiu.
- Revisions periòdiques i proves manuals per validar cada mòdul.
- Tecnologies utilitzades:
 - **Frontend:** Next.js, Tailwind CSS, DaisyUI, Context API, WebSocket
 - **Backend:** Node.js amb Express, WebSocket, MongoDB (per persistència)
- Enfocament en l'**experiència d'usuari (UX)** i en una interfície intuïtiva i responsiva.

2.2. Planificació

Des que vam començar a treballar al projecte a mitjans de febrer no hem parat cap setmana. Teníem fet el diagrama de Gantt, però hem trigat molt menys del que pensàvem a completar cada tasca i haver d'anar fent diagrames constantment ens hauria endarrerit molt.

Per la qual cosa hem treballat tenint al cap tot el que calia fer i es podria dividir en les següents seccions:

Setmana 1	Sistema d'autenticació
Setmana 2	Trobar APIs per aconseguir les dades de les accions. Mostrar en una taula les accions.
Setmana 3	Compra / Venda d'accions Mostrar el rendiment del portafolis al dashboard.
Setmana 4	Portafolis amb seguiment Cartera virtual Historial de transaccions
Setmana 5	Millores de codi i rendiment
Setmana 6	Secció de notícies
Setmana 7	Configuració de l'usuari
Setmana 8	Pàgines varies (contacte, sobre nosaltres, privacitat, etc.)
Setmana 9	Millores visuals i disseny adaptatiu
Setmana 10	Millorar connexions del WebSocket Obertura de mercat
Setmana 11	Xat

Setmana 12	Sistema de posts
Setmana 13	Desplegament
Setmana 14	Desplegament Documentació

2.3. Desenvolupament del Projecte

Descripció de la solució implementada

La solució SODA es divideix en dos blocs principals:

- **Backend:** una API REST construïda amb **Node.js** i **Express**, que gestiona l'autenticació (JWT), la base de dades (MongoDB via Mongoose), les transaccions (compres/vendes i ingressos/retirs de fons) i el CRUD de posts. A més, hi ha un servidor **Socket.io** que emet en temps real les actualitzacions de preus d'accions i les notificacions de like/comentari.
- **Frontend:** una aplicació **Next.js** que consumeix tant l'API REST com el canal WebSocket per construir una interfície interactiva i responsiva.

Tota la comunicació està protegida amb JSON Web Tokens, i les dades es desen a **MongoDB** en col·leccions separades per a usuaris, transaccions, accions simulades i posts.

Tecnologies utilitzades

- **Backend**
 - **Node.js** (v18+) amb **Express.js**
 - **Socket.io** per a la comunicació en temps real
 - **MongoDB** amb **Mongoose**
 - **JSON Web Tokens** per a autenticació
- **Frontend**
 - **Next.js** (React)
 - **Socket.io-client** per rebre actualitzacions en temps real
 - **Tailwind CSS** i **DaisyUI** per a l'estil i components ràpids
 - **React Query** per a la caché i sincronització de dades
- **Eines de desenvolupament**
 - **Git/GitHub** per al control de versions
 - **VS Code** com a IDE.
 - **Postman** per a proves d'API

Arquitectura de la solució

1. Client (Next.js)

- Pàgines i components React que utilitzen l'store per a l'estat (usuari, cartera, posts).
- Connexió simultània a l'API REST i al WebSocket per a dades "live" i en memòria.

2. Server (Express + Socket.io)

- Controllers, serveis i models Mongoose organitzats en capes.
- Middleware d'autenticació JWT per a rutes HTTP i connexions WS.
- Rooms de Socket.io per a cada acció i notificacions de posts.

3. Base de dades (MongoDB)

- Col·leccions per a **Users**, **Transactions**, **Stocks** i **Posts**.
- Esquema flexible per adaptació fàcil de nous camps.

Funcionalitats principals

1. **Autenticació i perfil**
 - Registre, login, canvi de contrasenya i dades de perfil.
2. **Compra/Venda d'accions**
 - Validació de saldo, actualització de cartera i registre d'operacions.
3. **Preus en temps real**
 - WebSocket que envia actualitzacions de preu cada segon.
4. **Dashboard personalitzat**
 - Rendiment i resum de posicions obertes.
5. **Mòdul de posts**
 - CRUD complet, likes i comentaris en temps real via WebSocket.
6. **Secció de notícies**
 - Fetch d'API externa o simulada per mostrar titulars rellevants per acció.

Decisions tècniques rellevants

- **Next.js API Routes:** permet emprar el mateix domini per a lògica de servidor (middleware, autenticació) i d'interfície, simplificant el desplegament.
- **MongoDB:** elecció per la seva escalabilitat, esquema dinàmic i integració senzilla amb Mongoose.
- **React Query:** sincronitza de forma automàtica les dades de l'API REST, amb caching i refetch intel·ligent.
- **Arquitectura MVC simplificada** al backend: separació clara de responsabilitats sense sobrecàrrega de frameworks més complexos.

2.4. Avaluació i Resultats

Anàlisi del grau d'assoliment dels objectius

Pràcticament, hem assolit tots els nostres objectius, encara que falten algunes millores que es comentaran més endavant.

Avaluació per punts:

- **Objectius assolits**

1. Compra / venda d'accions
2. Cartera
3. Portafolis
4. Configuració
5. Historial
6. Disseny adaptatiu
7. Notícies

- **Ampliacions realitzades**

Algunes ampliacions que no teníem contemplades afegir són:

1. Xat
2. Posts
3. Accions preferides

Incidències i resolucions

Hem trobat alguns problemes molt petits i molt pocs problemes de gran magnitud, un d'aquests últims serien els següents:

- **Problemes trobats**

1. Al principi hi havia un WebSocket per usuari, el problema és que l'API key era la mateixa llavors no podien haver-hi diversos usuaris al mateix temps.
2. Volíem desplegar el front-end a Vercel, una plataforma d'allotjament d'aplicacions NextJS el problema és que requeríem protocol HTTPS, NO autosignat, el qual no podíem aconseguir.

- **Solucions aplicades**

1. Per solucionar-ho hem deixat un WebSocket permanentment obert d'on agafem les dades, i aquestes dades es distribueixen als diferents usuaris
2. Hem allotjat la nostra aplicació al servidor web de Lacetània

3. Elements Finals

3.1. Conclusions

- Resum dels resultats

Hem pogut aconseguir completament l'objectiu principal de la pàgina: crear un simulador d'inversions real amb el qual puguis operar en un mercat actualitzat a temps real i veure notícies sobre l'acció per decidir com operar.

- Valoració personal

Ha sigut una experiència molt enriquidora haver fet aquest aplicatiu, poder haver treballat en un projecte completament nostre fent les nostres pròpies decisions ha sigut molt constructiu. Hem après molt sobre com fer les coses i sobre com no fer-les per futurs projectes.

- Propostes de millora

En un futur voldríem fer que el disseny sigui encara més adaptatiu i accessible per a tothom. Al manual tècnic s'expliquem les millores que voldríem afegir a cada apartat.

3.2. Enllaç al GitHub

<https://github.com/Daneteee/SODA.git>