Pokémon Card Trading Platform

**Team Members:**

Dan Park

Angela Marie Tagudin Chua

Amany Omar

**Executive Summary**

Our project focuses on building a Pokémon card trading platform using real-time data from the Pokémon TCG API. The primary goal is to help users search, favorite, and explore trading cards through a clean, dynamic UI. Our solution addresses the problem of scattered card information by aggregating card data in one place and allowing users to manage personal favorites/history. Key deliverables include dynamic routing, protected login functionality, global state management with Jotai, and API integration.

**Project Tasks**

|  |  |  |  |
| --- | --- | --- | --- |
| **Person** | **Step** | **Details** | **Date** |
| Angela | Work on wireframes | Create basic layout, design, etc on Figma |  |
| Amany\_phase 0 | Organize product data, web navigation | Organize what product data will be shown to user, decide the architectural map for website | Done on March 26 |
| Dan | Organize website info | Decide on website name, description, info, etc, for each page |  |
| Dan | Create main page layout | Create navigational bar, components, etc... that will show for each site |  |
| Amany | Task 1 | Modeling Store Data |  |
| Dan | Task 3 | Authentication/Authorization |  |
| Angela | Task 4 | “Favorites” Functionality |  |
| Dan | Task 5 | “Route Guard” functionality |  |
| Amany | Task 6 | Publishing our App on Vercel |  |

Project Deliverables

|  |  |
| --- | --- |
| Phase/Date | Details (form, route) |
| Phase1 | UX/UI Wireframe, product list |
| Phase2 | Codebase with layout, product listing, pagination, basic functionality done |
| Phase3 | Codebase with user accounts, user authentication, favourites, done |
| Phase4 | MongoDB connected and tested with site, bug-fixes applied, website css applied |
| Final Phase | Deployment, final bugs fixed |

**Tasks:**

Task 1: Modeling Store Data

Task 2: WireFraming, Layout, Dynamic Content

Task 3: Authentication/Authorization

        Step 1: Create a “User” API

        Step 2: Update Next.js App (Login and Register Components)

Task 4: “Favorites” Functionality

Task 5: “Route Guard” functionality

Task 6: Publishing our App on Vercel

Task 7: (Bonus )  
                Additional functionality: site interact with dynamic data  
                Unit testing

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  
Initial Planning:

Structure: [Amany]

|  |  |  |
| --- | --- | --- |
| **Page** | **Route** | **Purpose** |
| Homepage | / | Intro, slogan, and link to explore cards |
| About | /about | Team members: resume, skills, hobbies, similar to assignment 1) |
| Product Listing | /cards | Displays Pokemon cards with filters, search, pagination |
| Card Details (Dynamic) | /cards/[id] | Detailed view of a selected card |
| Login/Register | /login, /register | User’s favorites/history |
| Favorites | /favorites | Shows user’s favorited cards |
| 404 | /404 | Not found page |

Folder Structure: [Amany]

**|-- public/** //for static files, image, icons

**|--src/**

**|-- pages/**  
**| |-- index.js  
| |--about.js  
| |-- cards/  
| | |--index.js  
| | |-- [id].js** dynamic for card details of selected card (id, picture, name, description, price, rarity, abilities, attacks, level) **| |-- login.js //form  
| |--register.js //form  
| |--favorites.js  
| |--404.js  
|--components/  
 CardList.js** //SWR **CardItem.js  
 SearchBar.js  
 Layout.js  
 NavBar.js  
|-- store/**favoriteAtom // Jotai store (Global state)  
**|-- styles/globals.css**

**|--utils/ fetcher.js** //SWR fetcher with .env API key **|--categories.js** (min 4 categories) (“d”, “description”) { "data": ["Colorless", "Darkness", "Dragon", "Fairy", "Fighting", "Fire", "Grass", "Lightning", "Metal", "Psychic", "Water"]}   
**|--products.js** //Product data ”mock” (id, name, description, price, discontinued (Boolean), categories[])

**|-- .env.local //API key**