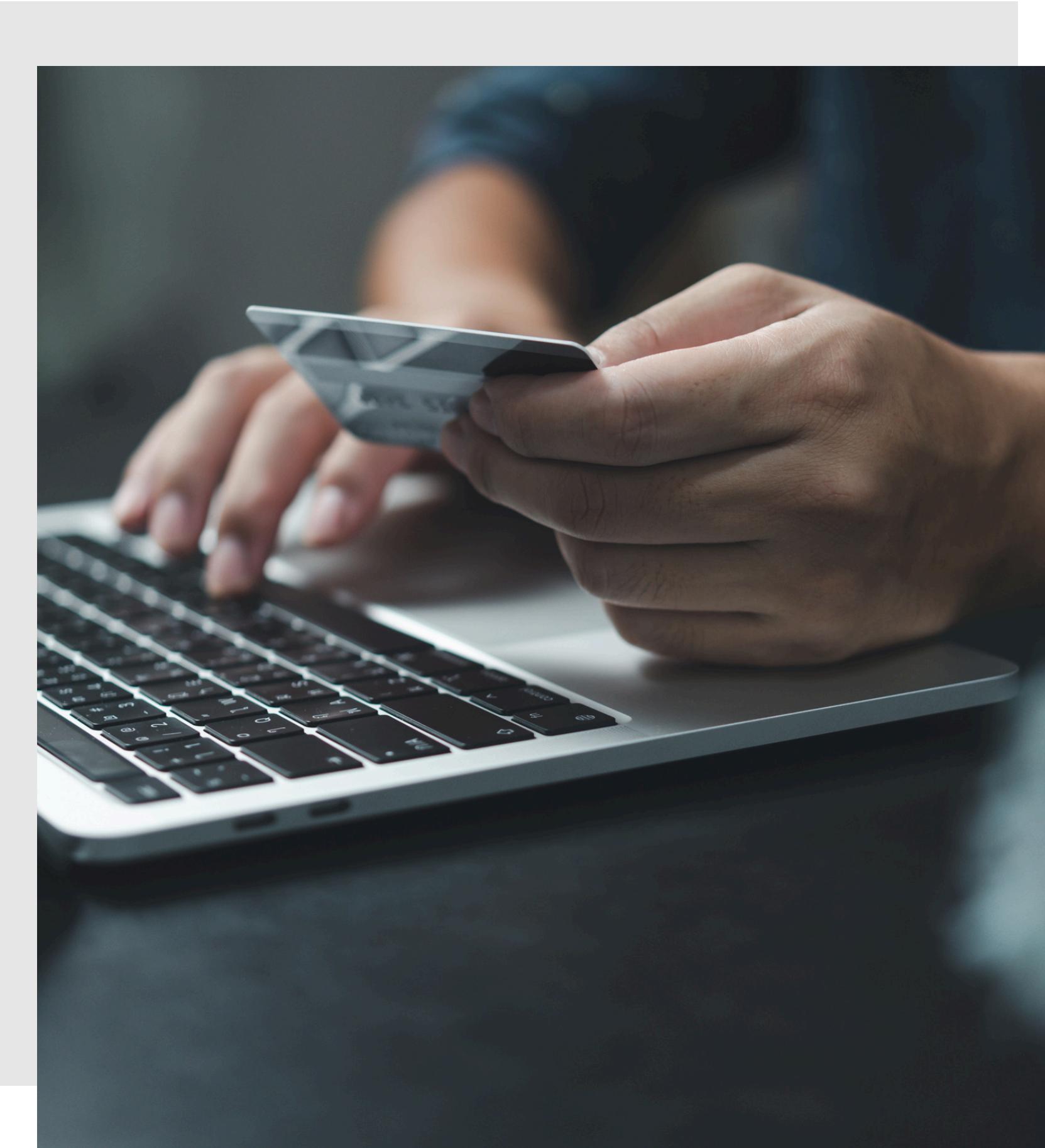


Shaddow Bazaar

E-COMMERCE





Introduction

WHAT IS SHADDOW BAZAAR?

Shaddow Bazaar is an e-commerce platform designed to provide a seamless shopping experience for users interested in purchasing and selling anything they want. This project aims to develop a comprehensive API, incorporating both back-end and front-end functionalities, as well as deployment using Docker containers.

Throughout this presentation, we'll delve into the project's purpose, key features, development process, and future prospects.



*"Innovative
e-commerce
platform."*



Project Overview

KEY FEATURES

Our platform boasts a robust back-end developed using the Symfony framework.

The front-end is powered by React, enhancing user interactivity and responsiveness.

Deployment is streamlined using Docker containers, ensuring scalability and efficiency.

Our Team



LUCAS FIXARI

*Versatile Back-End
Developer*



WILLIAM WOZIWODA

API Developer



PIERRE ROCHETTE

Front-End Developer

FOR THIS PROJECT

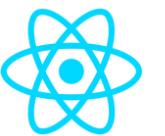
Dev Tools

Shaddow Bazaar's development was powered by a suite of essential tools. Symfony Framework provided a robust back-end structure, while React with TypeScript crafted dynamic user interfaces. Docker streamlined deployment, PostgreSQL managed data, and Stripe ensured secure payments. Composer managed PHP dependencies, Postman facilitated API testing, and NGINX acted as the web server. Make automated tasks, and Git enabled seamless collaboration. These tools, working in unison, drove the creation of Shaddow Bazaar's comprehensive e-commerce platform.



Symfony Framework

Symfony served as the backbone of our back-end development.



React with TypeScript

A winning combination for creating dynamic and responsive user interfaces.



PostgreSQL

PostgreSQL offered reliability, scalability, and advanced features for storing and retrieving data.



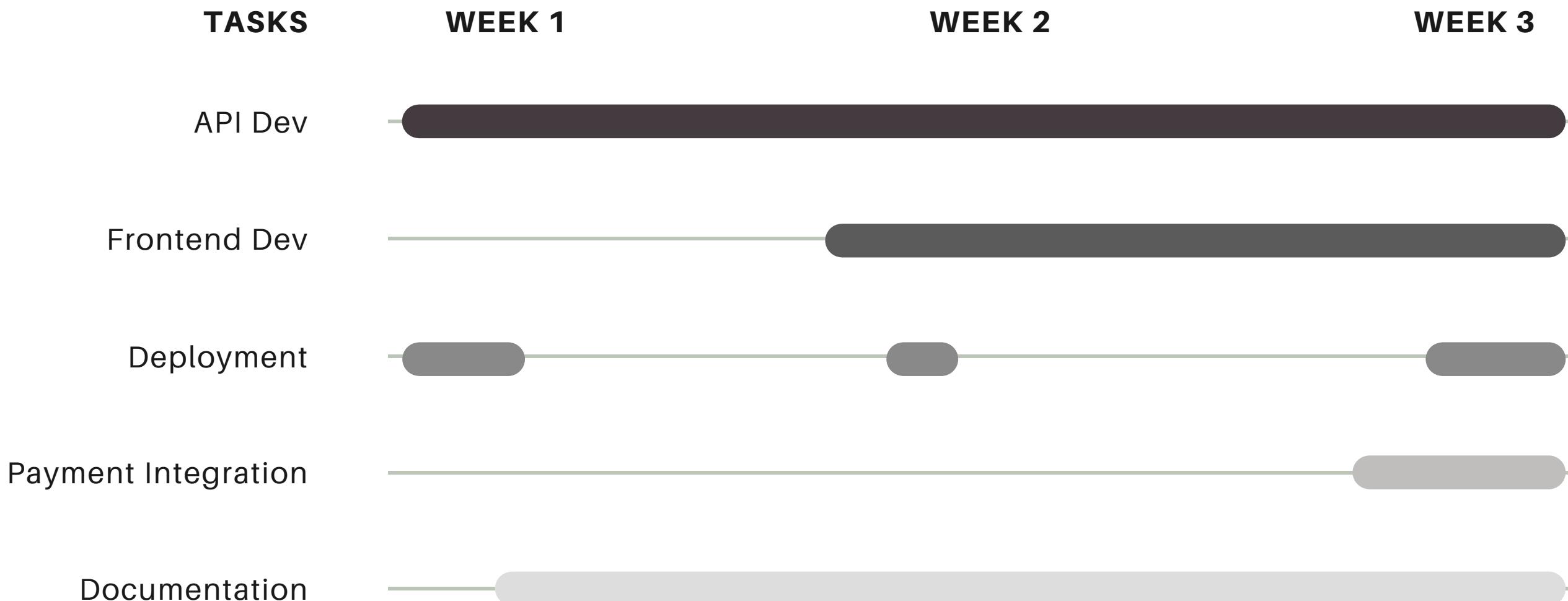
Docker

ensured consistent environments across development, testing, and production to enhanced scalability.



Planning

The project spanned three weeks, each member focusing on specialized tasks. Lucas, our Back-End Developer, Dockerized the app, handled documentation, and integrated the payment system. William, the API Developer, focused on backend functionality. Pierre, the Front-End Developer, crafted captivating UIs. Their coordinated efforts led to Shaddow Bazaar—a seamless fusion of backend expertise, API architecture, and engaging frontend design.



Project Organization

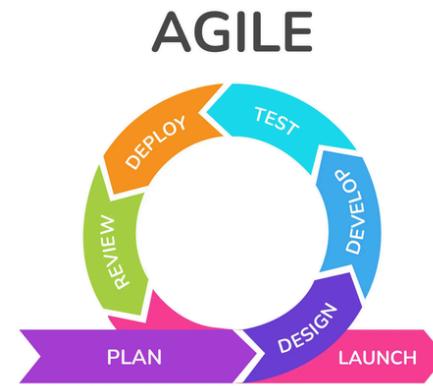
The organization of our project was meticulously orchestrated across multiple platforms, fostering seamless collaboration and efficient task management.

Discord served as our virtual workspace, facilitating real-time communication and brainstorming sessions.

GitHub, coupled with GitHub Projects, became our digital headquarters, housing our repository and meticulously curated project boards.

Face-to-face interactions further enriched our teamwork, allowing for dynamic discussions and rapid decision-making.

*“ C'est de la m*rde “*



Todo 0 / 5 Estimate: 0

This item hasn't been started

In Progress 2 / 5 Estimate: 0

This is actively being worked on

- Draft Client
- Draft Token must be invalid after time

Done 13 Estimate: 0

This has been completed

- T-WEB-600-STG_21 #3 index not found in route : no error return
- T-WEB-600-STG_21 #4 make base class Controller
- Draft Dockerisation
- Draft Users roles
- Draft

Rocket 10/05/2024 11:22
William tu parlais du PUT des products hier, ça ne fonctionnait pas que pour la photo ou il y a un problème plus général ?
J'ai des erreurs 500 sur le put là
J'ai test une fois avec et sans images

```
Data to send
name: Test 18/95
description: test
price: 28
stock: 20
PUT http://localhost:7777/api/Products/2 500 (Internal Server Error)
  > Artifexr (message: 'Request failed with status code 500', name: 'AxiosError', config: {}, response: XMLHttpRequest, ...)
```

EverydaySun 10/05/2024 11:27
Je crois pas qu'il as fait le edit d'un product (de ce que j'avais compris)

@Baald Juste avant que Lucas ne râle, non on peut pas PUT un produit ...

Rocket 10/05/2024 11:28
J'ai relu son message, il ne parle que de l'image,
Je vais mettre ça de côté, une erreur 500 c'est problème serveur,
dès qu'il rep je vois

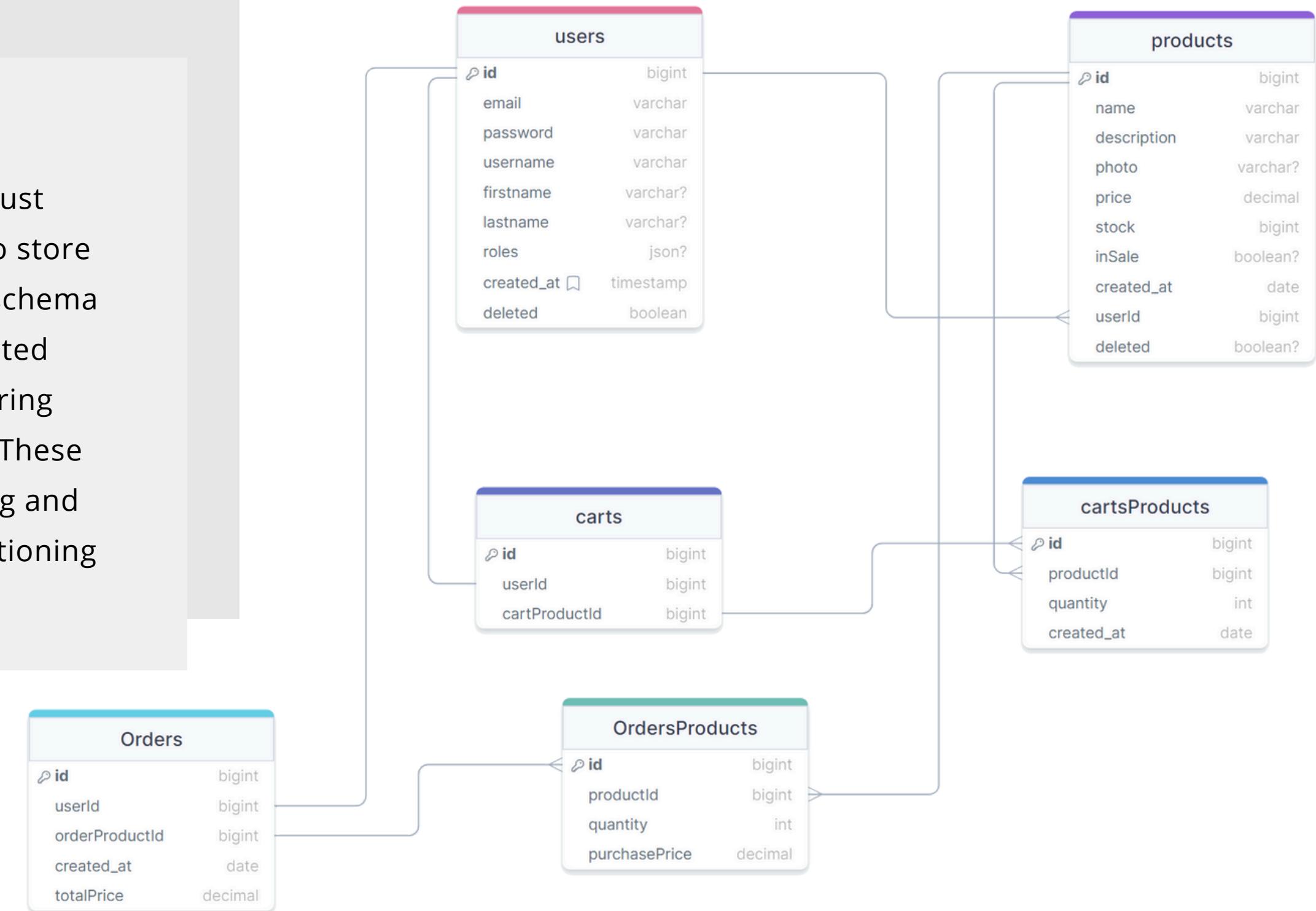
EverydaySun 10/05/2024 11:29
Du coup c'est pas moi qui ai râlé x)

Rocket 10/05/2024 11:30
moi je n'ai pas râlé j'ai posé une question cher Lucas
c'est une grande différence

POSTGRES

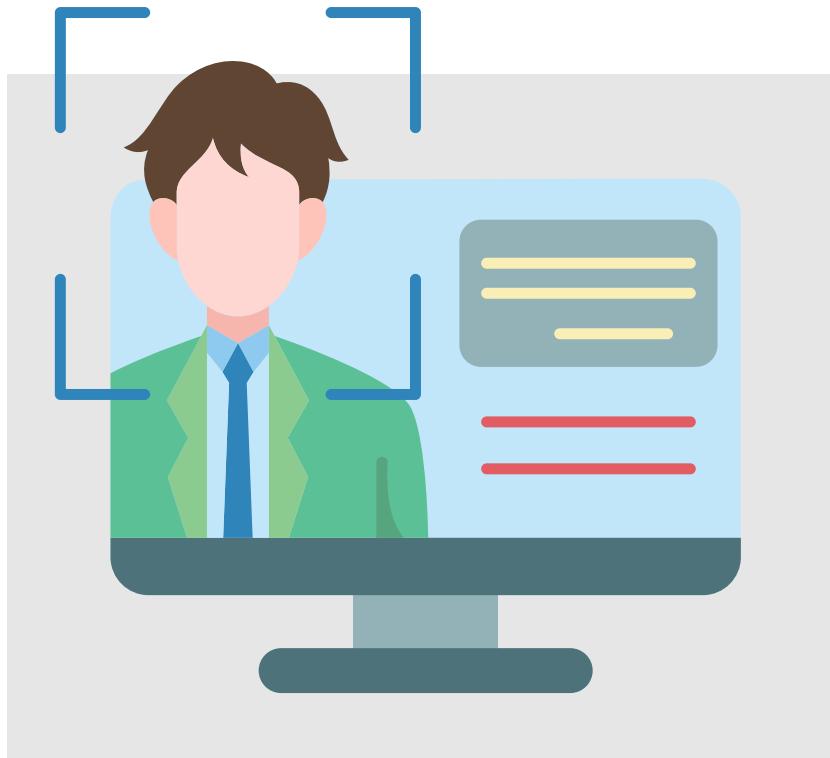
Database Overview

Our project leverages PostgreSQL, a robust relational database management system, to store and manage data efficiently. The database schema encompasses various tables interconnected through well-defined relationships, ensuring seamless data retrieval and manipulation. These relationships enable intricate data querying and manipulation, facilitating the seamless functioning of our application.



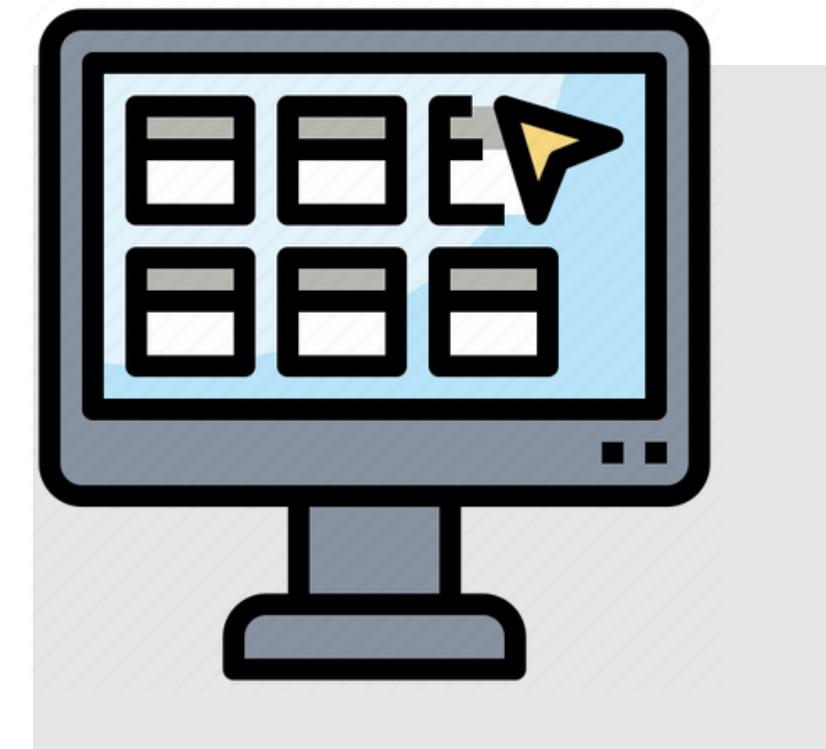
Exploring the API

Shaddow Bazaar's API is meticulously designed to adhere to REST standards, ensuring efficiency and robustness in data exchange. Implemented with Symfony framework, it offers comprehensive endpoints for user management, product handling, and order processing. With clear status codes and detailed documentation.



Users

Manages user registration, authentication, and profile updates efficiently.



Catalog & Cart

Facilitates seamless browsing, retrieval, addition, modification, and deletion of products.



Orders

Streamlines order placement, retrieval, and detailed order information for users.

FRONT-END OVERVIEW

Design



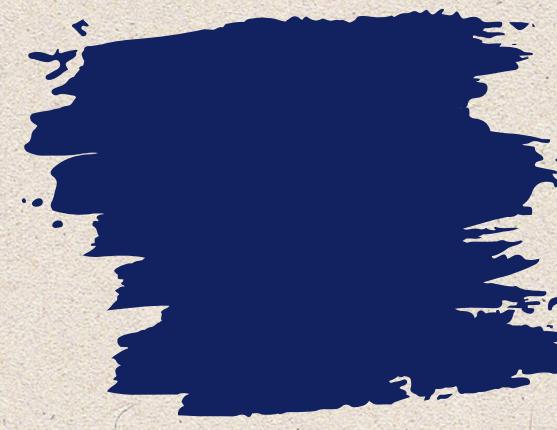
The screenshot displays the front-end interface of the Shadow Bazaar website. At the top, there's a dark header bar with the brand name "Shadow Bazaar" in orange. Below it, two buttons are visible: "Login" and "Register". The main content area features a grid of five book cards, each with a small thumbnail image, a title, a brief description, and a price. Each card also includes "Add to cart" and "View more" buttons.

| Book Title | Description | Price |
|--------------------------------------|-------------------------------------------------------------------|---------|
| placeat repudiandae | Madame Bovary, le soir, n'allait pas chez elle, Emma se com... | 85.36 € |
| necessitatibus mollitia qui | Elle en témoigna davantage à se ressaisir, d'une manière in... | 82.02 € |
| laudantium et sunt null | Quand son ouvrage était fini, il ne p... | 87.02 € |
| sapiente simili que ea | Alors, se rappelant à la chaleur de l'appartement, jointe à l... | 60.11 € |
| aut quis dolorumque eaque sequi odit | S'il le fallait, mon ami. -- N'étais-tu pas heureuse? Est-ce m... | 86.11 € |
| dolor sit | Emma, il la chérissait davantage. C'é | 21.18 € |

Shaddow Bazaar's front-end exudes a modern and minimalist aesthetic, aiming for a seamless user experience. The brand identity embodies sophistication and innovation, resonating with tech-savvy shoppers. The color scheme harmonizes shades of blue, yellow, and orange, evoking trust, warmth, and vitality. These colors were chosen to instill confidence in users while infusing energy and vibrancy into their shopping journey.

Crafting User-Centric Elegance

Our design philosophy revolves around clarity and elegance, ensuring effortless navigation and engagement. With a focus on user-centricity, we aim to deliver a visually captivating interface that enhances the shopping journey while reflecting the brand's identity and values.



ShadowBlue

bg-blue-500 — RGB(12, 22, 61) alias Blue



ShadowYellow

bg-yellow-400 — RGB(244, 197, 0) alias Amber



ShadowOrange

bg-orange-400 — RGB(226, 107, 0) alias Orange

FRONT-END OVERVIEW

Color Palette

The chosen color palette aligns with the e-commerce site's objectives by creating a visually appealing and emotionally engaging environment.

Blue symbolizes trust, essential for building credibility in online transactions.

Yellow adds warmth and friendliness, fostering a welcoming atmosphere for users.

Orange injects energy and excitement, encouraging users to explore products with enthusiasm.

Together, these colors evoke a sense of reliability, positivity, and vibrancy, enhancing the overall user experience and fostering trust in the brand.

Project Challenges

Implementing Stripe integration posed multifaceted challenges, from grasping its intricacies to configuring Nginx/Apache with separate Docker images. Crafting a functional Nginx configuration demanded meticulous attention, ensuring seamless communication between services. Additionally, adhering to Epitech directives added an extra layer of complexity, requiring precise alignment with academic guidelines.

CORS Policy

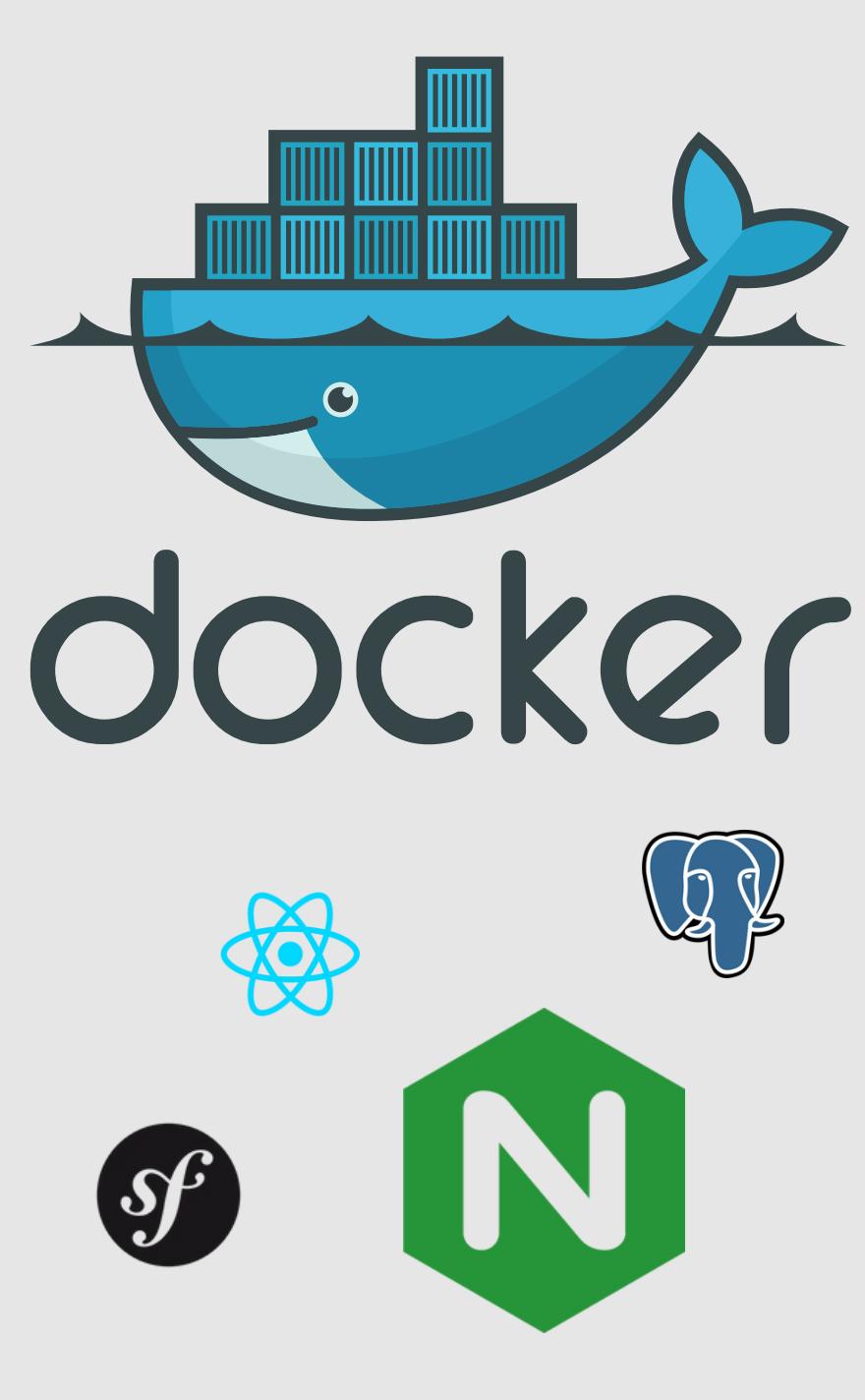
Overcoming the CORS policy hurdle between React and Symfony posed initial integration challenges, requiring strategic configuration to enable seamless communication between front-end and back-end components.

Symfony Complexity

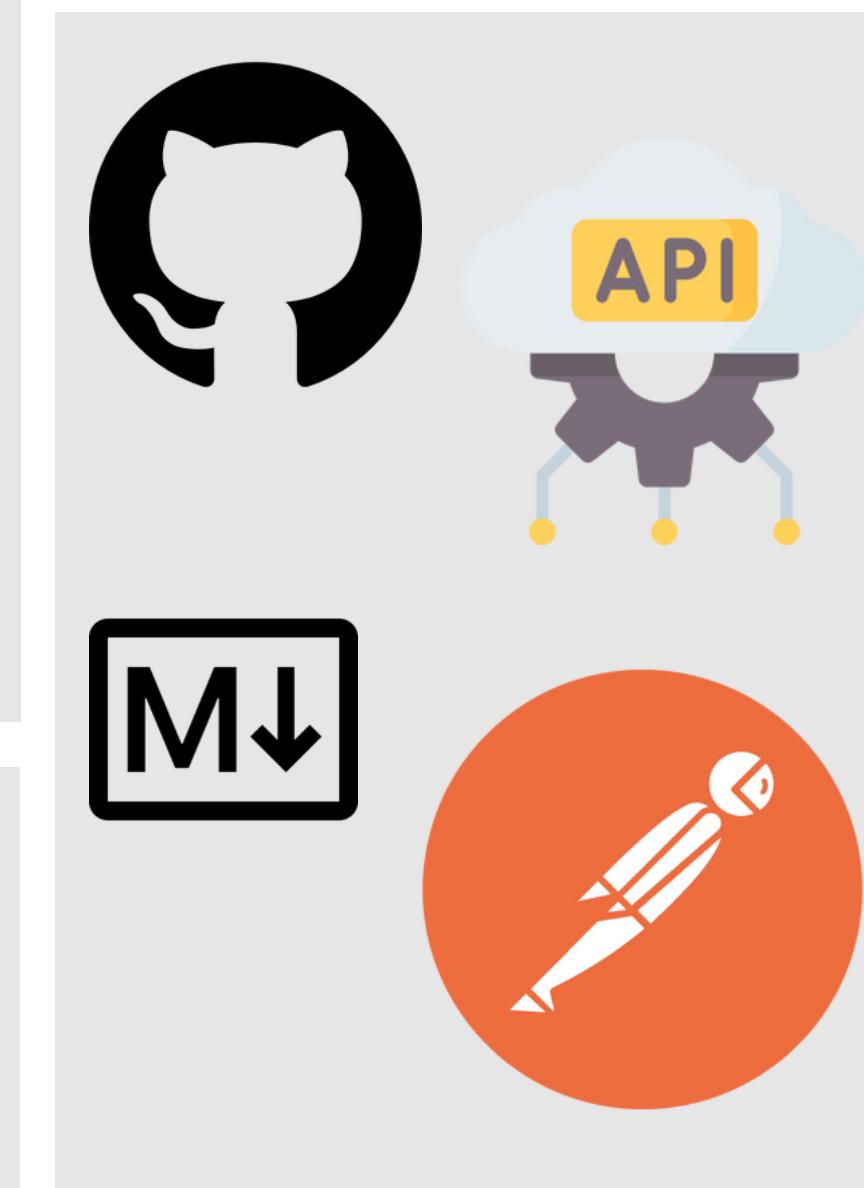
Navigating the intricacies of Symfony framework presented a learning curve, demanding meticulous troubleshooting and debugging to ensure smooth development progress.

Documentation Gap

Addressing the documentation gap in linking React with Symfony proved challenging, necessitating extensive research and experimentation to bridge the knowledge divide and implement effective solutions.



"Bonuses are requested."



Bonus Features

NOT MANDATORY IN THE PROJECT

Full Dockerized Application — Our project is fully containerized using Docker, encompassing both the backend and frontend components for seamless deployment.

Comprehensive API Documentation — We've meticulously documented our API, ensuring clarity and accessibility for developers. Additionally, a Postman collection is available on GitHub for convenient API testing.

Admin Page — We've developed an intuitive admin page to streamline product management, enhancing user experience and efficiency.



Thanks

We would like to express our gratitude to the [Symfony Documentation Team](#) for their invaluable resources and guidance throughout the development process. Their comprehensive documentation has been instrumental in understanding and implementing the Symfony framework effectively. We extend our sincere appreciation for their dedication to providing high-quality documentation, which has significantly contributed to the success of our project.

Contact

If you encounter any issues or have questions, feel free to reach out to the project maintainers for assistance.



<http://github.com>



<http://pg-admin.com>



www.shaddowbazaar.store