

Vinyl Vibe - T3A2-A

Vinyl Vibe is a sophisticated e-commerce store built using the MERN stack. This platform seamlessly integrates secure payment processing, user authentication, and inventory management into a user-friendly shopping experience.

At its core, the application serves as a bridge between businesses and their customers, offering a modern, responsive interface for browsing products, managing purchases, and processing secure transactions. Built with scalability in mind, it accommodates both small businesses and larger enterprises, providing essential tools for inventory management, order processing, and customer engagement.

The platform combines React's powerful frontend capabilities with Express.js's robust backend, MongoDB's flexible data management, and industry-standard security practices. With features like Stripe integration for payments and automated email notifications through Resend, it delivers a comprehensive solution for modern e-commerce needs.

Key aspects include:

- Secure user authentication and authorization
- Comprehensive product management system
- Streamlined checkout process with Stripe integration
- Responsive design for all devices
- Admin dashboard for business operations
- Real-time inventory tracking
- Automated email notifications for orders and updates

Table of Contents

- [Features and Functionality](#)
 - [MVP Features](#)
 - [Additional Features](#)
- [Packages, Dependencies and Third-Party Services](#)
 - [Backend](#)
 - [Frontend](#)
 - [Development](#)
 - [Deployment Services](#)
- [Dataflow and Application Architecture](#)
 - [Dataflow Diagram](#)
 - [Application Architecture Diagram](#)
- [User Stories & Personas](#)
 - [Persona 1](#)
 - [Persona 2](#)
 - [Persona 3](#)
 - [Persona 4](#)
- [Site Map](#)
- [Wire Frames](#)

- Task Management and Project Progress
 - Trello Board

Features and Functionality

MVP Features

1. Shopping Experience

- Intuitive shopping cart
- Mobile-responsive design
- Cart persistence
- Product catalog with detailed information
- Multiple product images
- Basic product filtering
- Basic sorting options
- Search functionality

2. Payment and Checkout

- Handoff to Stripe for payment processing
- Basic email notifications/confirmations

3. User Authentication and Management

- Secure registration and login system
- JWT-based authentication
- Password reset via email
- Role-based access (Customer/Admin)
- User profile management
- Address management
- Order/transaction history

4. Admin Management Features

- Product CRUD operations
- Order CRUD operations
- Customer CRUD operations
- Search functionality

Additional Enhanced Features (Post-MVP)

1. Enhanced Shopping Experience Features

- Product reviews and ratings
- Wishlist functionality
- Recently viewed products
- Social sharing
- Real-time stock updates

2. Enhanced Payment and Checkout Features

- Custom checkout flow
- Multiple payment methods
- Discount code system

- Guest checkout option
- Shipping options

3. Enhanced User Authentication and Management Features

- Order tracking

4. Enhanced Admin Management Features

- Advanced analytics
- Sales reports
- Bulk product management
- Inventory management

Each feature is designed to enhance the overall shopping experience while providing businesses with powerful tools to manage their online presence effectively.

Packages, Dependencies and Third-Party Services

The E-commerce Store utilises carefully selected technologies that work together to create a robust online shopping experience. Here's a detailed overview of our technology choices and why they're essential for our project:

Backend

- [Express.js](#) (version 4.18.2)

- body-parser
- cors
- compression

We chose Express.js as our backend framework for its minimalist yet powerful approach. Its middleware system is perfect for our e-commerce needs, allowing us to easily implement authentication, request processing, and error handling. The large ecosystem of middleware packages helps us solve common e-commerce challenges like CORS for secure client-server communication and compression for faster response times.

- [MongoDB](#) (version 6.3.0)

- mongodb (version 6.3.0)
- mongoose (version 8.1.1)

MongoDB's flexible schema design is ideal for our e-commerce platform where product attributes can vary significantly. Mongoose adds a crucial layer of structure and validation, ensuring our product data, user profiles, and order information maintain consistency. The schema-based approach helps prevent errors in critical operations like order processing and inventory management.

- [JSON Web Token](#) (version 9.0.2)

- jsonwebtoken

JWT provides a secure and scalable solution for user authentication in our e-commerce platform. It enables stateless authentication, perfect for maintaining user sessions across multiple devices while accessing protected routes like user profiles, order history, and admin functions.

- [Stripe](#) (version 14.14.0)

For handling payments, Stripe is our go-to choice due to its robust security features and comprehensive API. It manages complex payment flows while ensuring PCI compliance, handling multiple currencies, and providing detailed transaction analytics - crucial features for any modern e-commerce platform.

- **Resend** (version 3.2.0)

Email communication is vital for e-commerce, and Resend provides a modern, developer-friendly solution. We use it for sending order confirmations, shipping updates, password resets, and marketing communications, with excellent delivery rates and easy-to-use templates.

- **Jest** (version 29.7.0)

- supertest

Testing is crucial for maintaining reliability in e-commerce systems. Jest provides a comprehensive testing solution for our backend, allowing us to verify critical paths like checkout processes, inventory updates, and user authentication flows with confidence.

- **Bruno** (version 1.35.0)

- bruno-api

Bruno is a tool that allows us to test our API endpoints and ensure they are working as expected. It provides a user-friendly interface for sending requests and viewing responses, making it easier to debug and develop our API.

Frontend

- **React** (version 18.2.0)

- react-dom
 - react-router-dom (version 6.22.0)

React forms the foundation of our frontend, chosen for its component-based architecture that perfectly suits our e-commerce UI needs. It enables us to create reusable components for product cards, shopping carts, and checkout forms while maintaining excellent performance through its virtual DOM system.

- **Vite** (version 5.1.0)

- @vitejs/plugin-react

Vite significantly improves our development workflow with its lightning-fast hot module replacement and optimised build process. This means faster development cycles and better performance for our customers, crucial for maintaining a competitive e-commerce platform.

- **Axios** (version 1.6.7)

For handling API communications, Axios provides a robust solution with features like request/response interceptors and automatic JSON transformation. This is crucial for maintaining smooth communication between our frontend and backend, especially for operations like cart updates and order processing.

- **Zustand** (version 4.5.4)

Zustand is a state management library that provides a simple and efficient way to manage state in our React application. It allows us to create and share global state variables across components, making it easier to maintain a consistent and

responsive user interface.

- [Tailwind CSS](#) (version 3.4.1)

- postcss
- autoprefixer

Tailwind CSS enables us to build a consistent and responsive design system efficiently. Its utility-first approach allows rapid UI development while maintaining a professional look across our product catalog, shopping cart, and checkout process. The built-in responsive design utilities ensure our store looks great on all devices.

- [shadcn/ui](#) (version 0.8.0)

- @radix-ui/react-* (various components)
- class-variance-authority
- clsx
- tailwind-merge

We leverage shadcn/ui to provide a consistent and accessible component library. Its integration with Tailwind CSS and focus on customization allows us to maintain brand consistency while providing a polished user experience across all interactive elements of our store.

- [Vitest](#) (version 1.2.0)

- @testing-library/react
- @testing-library/jest-dom

Vitest ensures our frontend components and user interactions work flawlessly. Its compatibility with Vite makes it perfect for testing critical user flows like add-to-cart functionality, checkout processes, and form validations in a way that mimics real user behaviour.

Development

- [nodemon](#) (version 3.0.3)

During development, nodemon dramatically improves our workflow by automatically restarting the server when code changes are detected. This is particularly valuable when working on API endpoints and backend business logic, ensuring we can quickly test and iterate on features.

- [dotenv](#) (version 16.4.1)

Security is paramount in e-commerce, and dotenv helps us manage sensitive configuration data like API keys and database credentials safely across different environments. It's essential for maintaining secure configurations between development, testing, and production environments.

Deployment Services

- [Netlify](#) For our e-commerce frontend, Netlify is the ideal choice due to its seamless integration with React applications. Its zero-configuration deployment process means we can focus more on developing features rather than dealing with deployment complexities. The platform's edge network ensures our store loads quickly for customers worldwide, while the automatic preview deployments for each pull request enable our team to confidently review changes before they go live. The built-in analytics help us understand user behaviour and optimise the shopping experience.

- Render

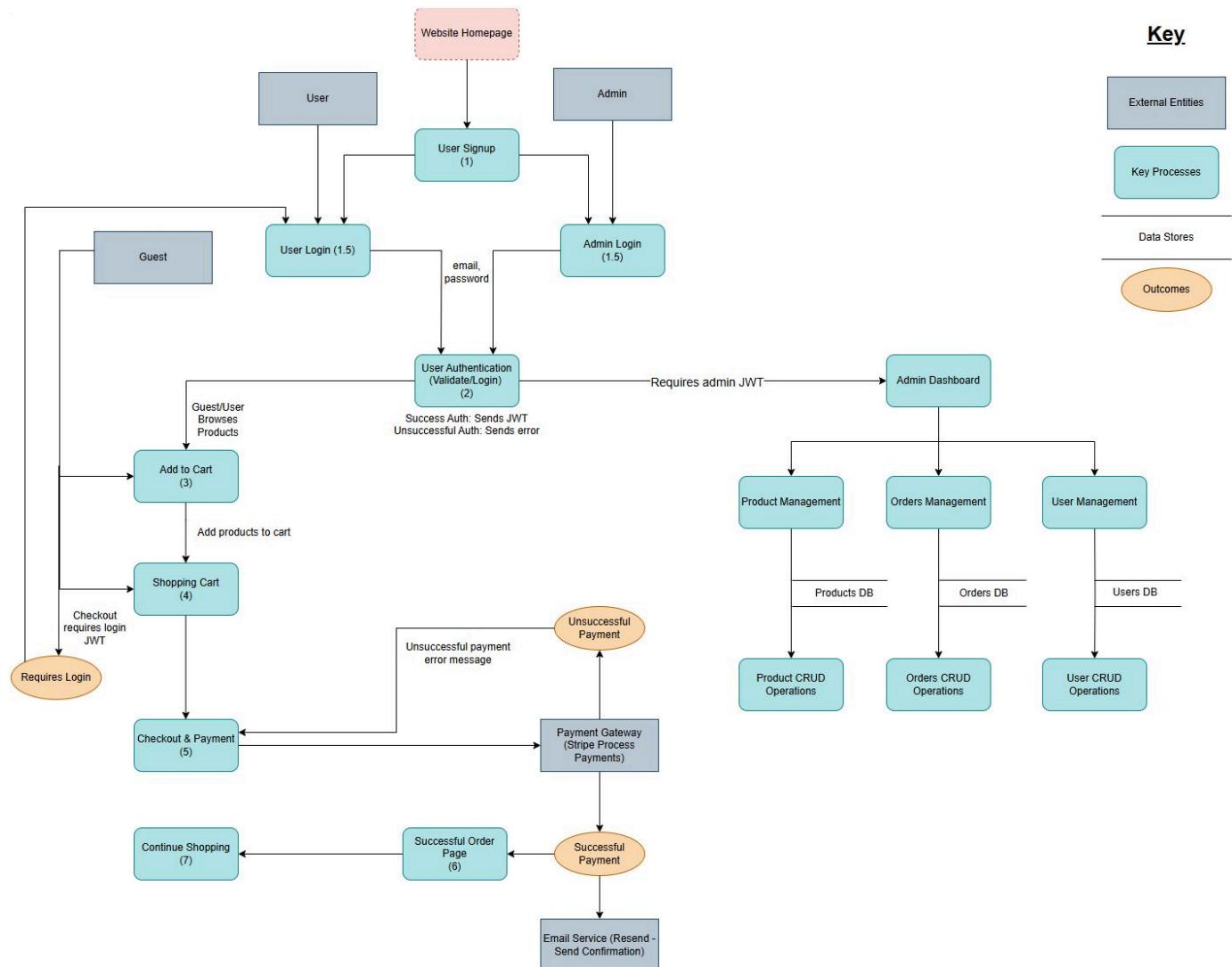
We chose Render for our backend deployment because it strikes the perfect balance between simplicity and power for a Node.js/Express API. Unlike more complex platforms like AWS, Render allows us to deploy our API with minimal configuration while still providing enterprise-grade features. Its automatic scaling capabilities ensure our store can handle traffic spikes during peak shopping periods or sales events. The seamless integration with MongoDB Atlas and built-in SSL security makes it ideal for handling sensitive customer data and payment processing through Stripe.

- MongoDB Atlas

For an e-commerce platform that needs to handle dynamic product catalogs, user sessions, and order processing, MongoDB Atlas provides the flexibility and scalability we need. Its document-based structure perfectly matches our data models for products, users, and orders, while the ability to perform complex queries helps with features like product filtering and search. The automated backups and security features protect our customer data, while the global distribution ensures fast database access regardless of customer location. The built-in monitoring helps us optimise performance and identify potential issues before they impact the shopping experience.

These packages and their dependencies work together to provide a robust, secure, and efficient full-stack e-commerce application, handling everything from database operations and payment processing to user interface components and testing.

Dataflow Diagram



External Entities

External entities represent the User/Admin or systems that interact with the application from the outside. These entities provide input or receive output from the system.

User

- Represents the customer interacting with the eCommerce store.
- Actions:
 - Authenticates to log in or register
 - Browse products
 - Add items to shopping cart
 - Proceed to checkout
 - Payment

Admin

- Represents the store administrator who can manage the store and perform CRUD operations.
- Actions:
 - Manages Product, Orders and Users via CRUD operations
 - View the admin dashboard tools

Guest

- Represents a guest browsing the store without logging in.
- Actions:
 - Browse products
 - Add items to shopping cart
 - Proceed to checkout (Checkout will then require login)

Payment Gateway (Stripe)

- Third party service used to securely process payments.
- Actions:
 - Processes payment details submitted by customer during checkout
 - Returns a payment confirmation or failure status

Email service (Resend)

- Third party email service used for sending transactional emails.
- Actions:
 - Sends order confirmation emails to the customer upon successful order payment
 - Sends additional updates related to orders or account activities

Processes (Customer Store)

Processes represent the core operations or functions that process, manipulate, or route data within the system.

User Signup (1)

- Users can signup with a new login to the website
- Inputs: New login credentials (email and password)
- Outputs: User account creation enabling login access

User Login (1.5)

- The standard user login which can access all areas of the website excluding the admin dashboard.
- Inputs: Login credentials (email and password)
- Outputs: Successful JWT token for a User login

Admin Login (1.5)

- Admin role able to access all areas of website including admin dashboard
- Inputs: Login credentials (email and password)
- Outputs: Successful JWT token for Admin login

User Authentication (2)

- Validates user credentials during login or registration
- Inputs: User credentials (email and password)
- Outputs: Authentication token (JWT) on success or error message on failure

Add to Cart (3)

- Adds products to shopping cart to view and use for checkout
- Inputs: Product data
- Outputs: Adds product data to cart

Shopping cart (4)

- Handles the user's shopping cart operations. Shows a list of products that have been added to cart, with product details (Price, SKU, Quantity, etc)
- Inputs: Product selections (add, remove, modify quantities)
- Outputs: List of selected items with quantities and pricing

Checkout and Payment (5)

- Manages the checkout process and interfaces with the payment gateway (via stripe)
- Inputs: Shopping cart details and payment details
- Outputs: Order confirmation on success or error message on failure

Successful Order Page (6)

- Displays a confirmation message and order details upon successful checkout

Continue Shopping (7)

- Allows users to return to browsing and shopping with a reset cart after successfully completing a checkout
- This process ensures users can continue their shopping journey with a refreshed cart, enabling them to explore and add new products

Processes (Admin Dashboard)

Admin Dashboard

- Provides the admin with tools for managing the store via CRUD operations
- Inputs: Admin actions for product, user and order management
- Outputs: Updated product, user and order details displayed and stored

Product Management

- Handles the management of product inventory, including creating, updating, viewing, and deleting products
- Inputs: Product data provided by the admin for CRUD operations (e.g. product name, price, stock quantity)
- Outputs: Updated product records stored in the database and reflected in the store's product listings

Order Management

- Manages the lifecycle of orders, from creation to tracking and fulfilment
- Inputs: Shopping cart details, payment confirmation from the payment gateway and user information
- Outputs: New order records stored in the database and a confirmation email sent to the user via Resend

User Management

- Oversees the management of user accounts within the system, including creating, updating, viewing and deleting user records
- Inputs: User data provided by the admin for CRUD operations (e.g. user profile information)
- Outputs: Updated user records stored in the database and reflected in the admin dashboard

Data Stores

Where the data is stored in the system and is accessed or updated by the processes.

User Database

- Stores user account information (name, email)
- Contents:
 - User Profiles (name, email, address)
 - Login credentials (hashed passwords)
 - Order history

Products Database

- Stores product information
- Contents:
 - Product details (ID, SKU, name, description, price, stock level)

Orders Database

- Stores records of all orders placed by users
- Contents:
 - Order details (items purchased, quantities and total price)
 - Payment status

Application Architecture Diagram

Front End



React



Vite



Vitest

Frontend Framework and Build Tool

Testing



Frontend and Backend Languages

API Requests ad Data Fetchig



Styling and UI Components



Hosting



Testing

Database ODM



Authentication and Authorisation

Hosting

Database



Database Management System

Database Hosting

Payment Processing



Payment Gateway and API



Front End

The front end is where users interact with the Vinyl Vibe website. They make requests to our API, that will be handed to the backend. Once the request has been processed, the result is sent back to the frontend and displayed to the user.

React (`react-dom`, `react-router-dom`)

React's component-based architecture is perfect for our e-commerce UI, enabling reusable elements like product cards and checkout forms, with high performance ensured by the virtual DOM.

Vite (`@vitejs/plugin-react`)

Vite speeds up development with its fast hot module replacement and optimised builds, delivering a smooth experience for both developers and customers.

Tailwind CSS (`postcss`, `autoprefixer`)

Tailwind's utility-first approach streamlines the creation of a consistent, responsive design system, ensuring a polished look across all devices.

shadcn/ui (`@radix-ui/react-`, `class-variance-authority`, `clsx`, `tailwind-merge`)

shadcn/ui provides a customisable, accessible component library that integrates seamlessly with Tailwind to maintain brand consistency and enhance usability.

Vitest (`@testing-library/react`, `@testing-library/jest-dom`)

Vitest, paired with Testing Library, ensures the reliability of key user flows like cart updates and checkout through fast, realistic component testing.

Axios

Axios facilitates smooth API communication with features like JSON transformation and interceptors, critical for cart updates and order processing.

Netlify

We deploy our application on Netlify, leveraging its fast, reliable hosting and CI/CD pipelines to ensure seamless updates and optimal performance.

Back End

The backend powers the Vinyl Vibe website by managing requests, handling data, and processing payments via the Stripe API. It ensures smooth product browsing, cart updates, and secure checkouts.

Node.js:

Node.js runs the server in a MERN stack, efficiently handling requests and connecting the frontend to the database.

Express.js (body-parser, cors, compression):

Express.js streamlines backend tasks with middleware like body-parser (parsing requests), CORS (secure communication), and compression (faster response times).

Mongoose:

Mongoose adds structure and validation to MongoDB, ensuring consistency in product data, user profiles, and order processing.

JSON Web Token (jsonwebtoken):

JWT enables secure, scalable user authentication and access to protected routes like profiles, order history, and admin features.

Jest:

Jest ensures backend reliability by testing key processes like checkout, inventory updates, and authentication flows.

Resend:

Resend simplifies email notifications (order confirmations, shipping updates, password resets) with templates and high delivery rates.

Render:

Render deploys the application, offering fast, reliable hosting and seamless updates with CI/CD pipelines.

Database

MongoDB (mongodb):

MongoDB's flexible schema design is ideal for managing product attributes and dynamic data in an e-commerce platform.

MongoDB Atlas:

Atlas provides a scalable solution for dynamic catalogs, user sessions, and orders, with features like complex queries, automated backups, robust security, and global distribution for fast access.

Stripe API

For handling payments, Stripe manages complex payment flows while ensuring PCI compliance, handling multiple currencies, and providing detailed transaction analytics - crucial features for any modern e-commerce platform.

User Stories & Personas

Vinyl Vibe is an expanding Record Store with a *cult following!*

After many years of struggling to keep the doors open as **Groovy Records**, the resurgence of the Vinyl Record market in the past 5 years has seen them outgrow the crude, hardcoded, HTML page that owner Norm Gleeson's nephew created in the 90s, attached below (image reference 1).

Norm has a large inventory of rare and hard to find records from both famous and obscure Australian artists, including highly sort after records that never made it into International markets. He is the go to guru for all things Aus Music!

Word of the treasure trove of rare and hard to find records at **Vinyl Vibe** has travelled far and wide through Facebook and Reddit. They've identified a growing market of customers who want buy from Norm, *but don't live in the area.*

Norm needs an effective modern website that can connect and sell to their customers, be they local, interstate, or international!

Persona 1



Norm Gleeson
Store Owner

Goals:

- Share his passion for music with the expanding community of music lovers, collectors, and audiophiles who purchase records and turntables from his shop.
- Expand his business to provide a more stable income for himself and his staff.
- Build a brand for his business to allow him to increase the size of his professional network.
- Expand his professional network to allow him to better serve his customers in the exchanging of hard to find and rare records.

Demographic:

- 66 years old
- Divorced
- No children
- Former Musician and Roadie for The Screaming Jets turned Record Store Owner.

Motivations:

- To help customers rediscover the joy and artistry of vinyl records, celebrating the unique sound quality, nostalgic charm of hiss and crackle, and the stories embedded in the music.
- To expand his business and foster an international community of music lovers and record store owners who value the craftsmanship and distraction-free experience of vinyl records.

Frustrations:

- Norm's current website is outdated, stuck in the 90s like his roadie memories. Building a new one is beyond his skillset, and he struggles with learning new systems.
- With 80-100 milk crates of records, Norm has no clear inventory system. Many records go unsold because they're not listed online, and his outdated website gets little traffic due to poor SEO.

User Stories:

As the owner of Vinyl Vibe,

- I want to attract new customers from outside my local area with a method of purchasing my stock for shipping.
- I need branding that reflects my desire to be viewed as a modern business with decades of expertise in my field.
- I need a method to properly catalogue my ever changing stock of records, turntables, parts, and accessories.



Ruby Harper

Store Manager/Web Master

Demographic:

- 38 years old
- Married
- 3 children
- Former JB Hi Fi Store Manager.
- Amateur Musician and Artist

Motivations:

- To thrive in a store focused on music, not like JB Hi-Fi, which she sees as a "Bing Lee with records."
- To feel pride, ownership, and community at work while fueling her passion for music.
- To connect with local musicians, offering them a platform to stock and sell their music.
- To become a go-to destination for touring artists' promotions and signings.

Frustrations:

- So much unrealised potential in this business!
- Incomplete database of stock, orders, and customers.
- A very outdated website that is quite obnoxious and ineffective with no procedures for marketing or retargeting customers.

User Stories:

As Vinyl Vibe's manager,

- I want to grow our business through serving our customers much better than we have been.
- I want a simple to use backend for the website that allows me to catalogue and advertise the vast stock of records, turn tables, parts and accessories we have at any given time.
- I want to be able to retarget our previous customers to advertise sales, small events, and listening parties that will get people into the store where they're more likely to buy records.
- We need to be able to effectively advertise to the people outside of our local area chasing very hard to find records that we currently have in stock, but not advertised.



Atticus Hastings-DuToit

Customer

Demographic:

- 29 Years old
- Single
- 2 x "Furr Children"
- Record collector, Audiophile, and Coffee Enthusiast.
- Amateur DJ and Producer.

Motivations:

- Always looking to expand his record collection, especially with rare and hard to find records that he can boast about. The more obscure the better.
- Obscure records offer him an opportunity to unearth songs that haven't been sampled before when creating beats.

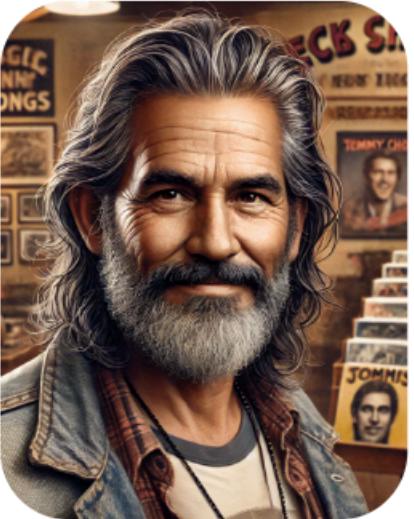
Frustrations:

- Can't make it into Vinyl Vibe in person as often as he would like to.
- Wants to be able to search a catalogue of Norm's records from home, and purchase online.
- Has a network of collectors that he would happily refer to Vinyl Vibe, but they don't live in the area.

User Stories:

As Vinyl Vibe's best customer,

- I want to be able to search their catalogue from my phone and make purchases from wherever I am.
- I want to be able to recommend them to other people from the record collecting community so the store can thrive.
- I want Vinyl Vibe to grow even bigger so I can say that I knew about and used to go there before it was mainstream.....



Diego Navarro

Owner of The Needle Drop

Demographic:

- 64, married, with 4 kids.
- Former Pearl Jam roadie who met Norm on a Screaming Jets tour in the 90s.
- Opened The Needle Drop, a similar store in Seattle, USA.
- Decades of industry knowledge and a vast, uncatalogued record collection.
- Potential customer for a website like Vinyl Vibe's.

Goals:

- Same goals as Norm, but specifically and in addition:
- Create a website that can reciprocate the service that Vinyl Vibes website will provide for him.

Motivations:

- Grow his business to the cult like status that Vinyl Vibes has achieved.
- Much like Norm, share his knowledge and passion for music with the record enthusiast community.
- Establish a resource-sharing agreement with Vinyl Vibe. Diego offers access to rare early Grunge and Alternative records, while Norm provides sought-after Australian Pub Rock, New Wave, and Underground records prized by Seattle collectors.

Frustrations:

- Regularly in contact with Norm (by phone call) to trade and find records for his customer base.
- Would buy from Norm more regularly if it were easier to see his stock levels on an up-to-date and easy to navigate website. Not having to call at 3am would be a bonus too.

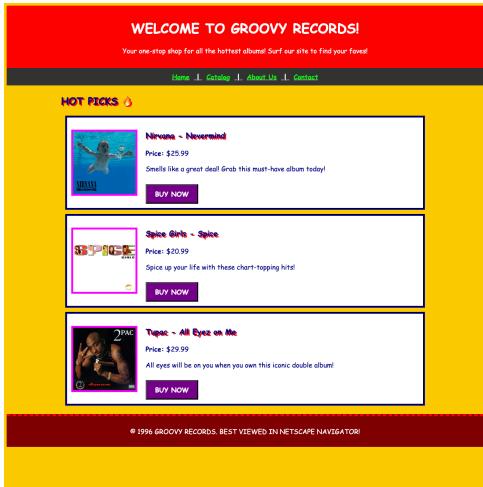
User Stories:

As the owner of The Needle Drop in Seattle,

- I want to see how a modern website works out for Vinyl Vibe before I commit to the investment myself.
- I want a website that is going to allow me to advertise to my potential customer base all over the world.
- I want to keep support the industry that has supported me and enriched my life.
- I want to expand my business so that I can move to a larger premises and create more product lines and income streams.
- I want a business that my family can run without me micro managing! I just want to oversee the store, hangout, and sell.

NB: Please find the first text based draft of Persona 1's user story below in Reference 2.

- Reference 1



- Reference 2

```
### Persona 1
#### Shop Owner
Name: Norm Gleeson
#### Demographic:
- 66 years old
- Divorced
- No children
- Former Musician and Roadie for The Screaming Jets turned Record Store Owner.

#### Goals:
- Share his passion for music with the expanding community of music lovers, collectors, and audiophiles who purchase records and turntables from his shop.
- Expand his business to provide a more stable income for himself and his staff.
- Build a brand for his business to allow him to increase the size of his professional network.
- Expand his professional network to allow him to better serve his customers in the exchanging of hard to find and rare records.

#### Motivations:
- To help his customers discover the simple joy of listening to a vinyl record. To learn to appreciate the work that has gone into a piece of art. The stories that are shared. The sound quality, the hiss and crackle, and the experience of not being distracted by skip buttons, fast forward and the overwhelm of endless options and advertising on streaming services.

#### Frustrations:
- His current website (much like most of his fondest memories of being on the road) is stuck in the 90s and is well overdue for an update, but Norm takes a while learning new systems and building a website is well outside his skillset.
- With an estimated 80-100 milk crates full of records in his existing inventory, Norm isn't even sure how many records he has at any given moment. It's also difficult for him to keep track of what comes and goes. There are many records that could be getting sold, but aren't even advertised on his website. Not that it would matter, as the SEO for his website is almost non-existent and it gets very little traffic.

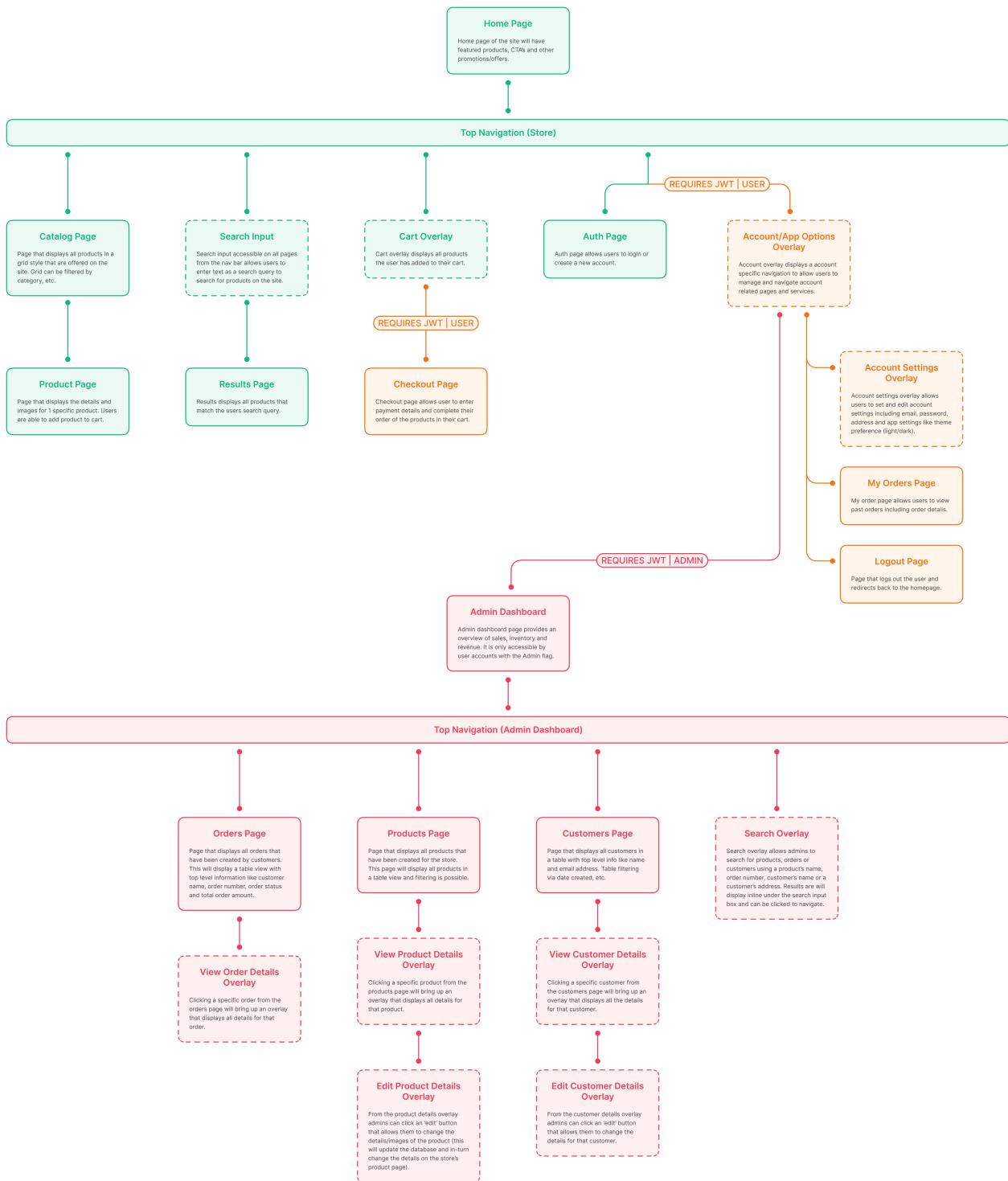
#### User Story 1
<strong>As the owner of Vinyl Vibe,</strong>
- I want to attract new customers from outside my local area with a method of purchasing my stock for shipping.
- I need branding that reflects my desire to be viewed as a modern business with decades of expertise in my field.
- I need a method to properly catalogue my ever changing stock of records, turntables, parts, and accessories.
```

Site Map

Our e-commerce platform's structure is designed to provide intuitive navigation for both customers and administrators. The site map illustrates the hierarchical organization of pages and features, ensuring a logical flow from browsing products to completing purchases. We've organised the navigation to minimise the number of clicks required to reach any destination while maintaining a clear separation between customer-facing pages and administrative functions.

The following site map outlines the complete structure of our application:

Sitemap



Wire Frames

We designed three breakpoints for the following devices:

- Mobile
- Tablet
- Desktop

You can view the Figma file [here](#).

Desktop view

E-Commerce Store | TA: admin / staff / store owner

Home Page // Desktop Catalog Page // Desktop Product Details Page // Desktop Auth Page // Desktop Search Results Page // Desktop

Store Management | TA: admin / staff / store owner

Store Dashboard // Desktop Orders Page // Desktop Products Page // Desktop

Customers Page // Desktop

E-Commerce Store | TA: admin / staff / store owner

Tablet view

E-Commerce Store | TA: admin / staff / store owner

Home Page // Tablet Catalog Page // Tablet Product Details Page // Tablet Auth Page // Tablet Search Results Page // Tablet

Store Management | TA: admin / staff / store owner

Store Dashboard // Tablet Orders Page // Tablet Products Page // Tablet Customers Page // Tablet

E-Commerce Store | TA: admin / staff / store owner

Mobile view

E-Commerce Store | TA: admin / staff / store owner

Home Page // Tablet

Catalog Page // Tablet

Product Details Page // Tablet

Auth Page // Tablet

Search Results Page // Tablet

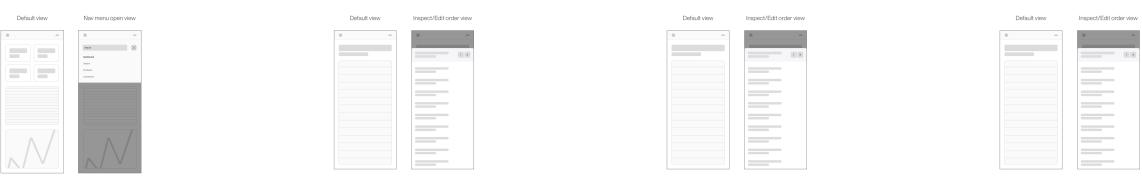
Store Management | TA: admin / staff / store owner

Store Dashboard // Mobile

Orders Page // Mobile

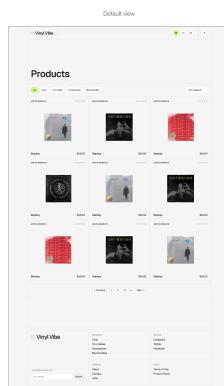
Products Page // Mobile

Customers Page // Mobile

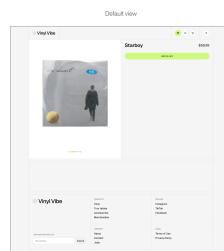


Hi-Fi Wireframe

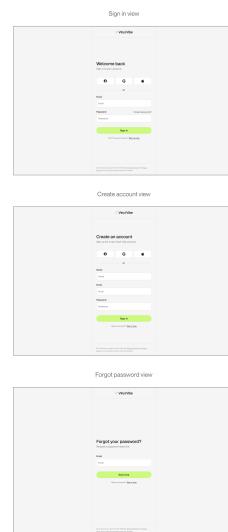
Catalog Page // Desktop



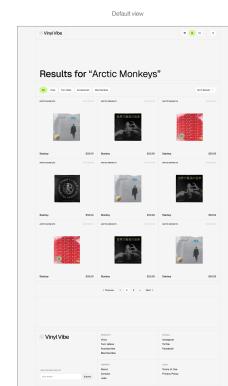
Product Details Page // Desktop



Auth Page // Desktop



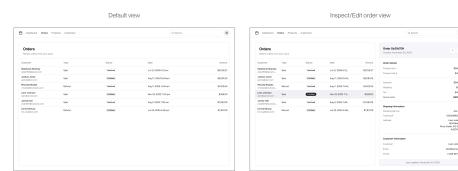
Search Results Page // Desktop



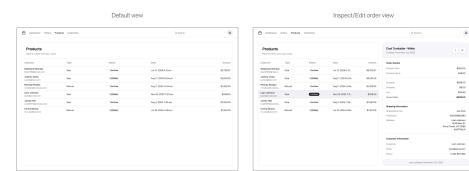
Store Dashboard // Desktop



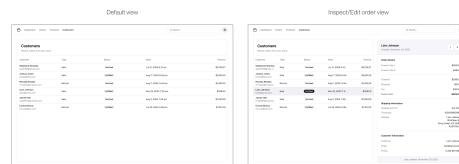
Orders Page // Desktop



Products Page // Desktop



Customers Page // Desktop



Task Management and Project Progress

Throughout the development of this project we utilised Trello as our primary task management tool. This approach allowed us to effectively organise, prioritise, and track the progress of various features and tasks.

Trello Board

You can view our project's Trello board [here](#).

The board is organised into the following lists:

- Backlog
- To Do
- In Progress
- Testing
- Done

