Fash-In - Name of the Project

Synopsis

Fashion is an ever-evolving industry that plays a crucial role in self expression culture and lifestyle with increasing reliance on digital platforms for shopping and styling inspiration, the demand for an interactive and user-friendly web application has surged.

Fash-in is a web based fashion and lifestyle platform designed to bridge the gap between fashion enthusiasts and online shopping. It allows user to explore the latest fashion trends, create outfit inspirations, shop from multiple brands, and engage with a fashion-forward community. The platform provides an easy and engaging way to stay updated on fashion without requiring artificial intelligence (AI) or complex algorithms.

The primary goal of fash-in is to provide user with a centralized hub for all things fashion and lifestyle. The application aims to:

supremo

* Offer a seamless intuitive browsing experience for discovering fashion trends

* Allow users to mix and match clothing items to create personalized outfits.

* Provide a convinient e-commerce shopping experience by integrating fashion-trends.

- Yash-in will feature a dedicated section for real-time fashion updates users can explore:
- Seasonal collections and trends from top fashion brands.
- Celebrity and influencer inspired look.
- It aggregate products from multiple fashion brands into one interface.
- Offer categorized browsing (clothing, accessories, footwear etc)
- Include an easy to use cart and checkout system.
- Feature product reviews, and ratings

Yash-in is designed to cater to a diverse audience, including:

Fashion ethusiasts who want to stay updated

supreme

with trends.

Online shoppers looking for a convinient way to purchase clothing and accessories

The homepage will feature:

- A visually appealing layout with trending fashion images
- Quick links to different sections such as shopping, style and community.
- A search bar for easy navigation

Yash-in is a dynamic fashion and lifestyle web based application designed to enhance the Online shopping and styling experience. By integrating an intuitive user interface, community driven engagement, and seamless e-commerce functionality, it aims to become a go-to platform for fashion lovers with an emphasis on user convinience. Yash-in is poised to make a significant impact in the digital fashion landscape.

13.01.2025

System Setup

1. Installation and setup of database (my sql / postgre SQL)

2. Installation and setup of java, JDK

3. Installation and system checking of VS code.

14.01.25

→ Environment setting of my SQL.

→ Installation of Springboot in VS code extension.

→ Basic idea about Springboot.

→ Demostration of spring boot project

→ Creating Zip folder of Dash-in, extraction of file.

supremo

→ Creating home.html on template folder.

→ Creating CSS file on static folder.

15.01.25

The landing page:

The landing page serves as the first point of interaction for users.

- Navigation bar: Added a responsive navigation bar with links to all the categories present.

16.01.25

Designing of landing page

- Ensured the landing page is responsive and adapting
- Designed home page.

- Added navbar and footer to il-

17.01.25

- Used html and CSS and added javascript for interactive elements such as smooth scrolling and hover effect.

20.01.25

Designing of PDP

* Worked on designing the Product Detail Page (PDP) for Yash-in web application.

* It displays the detailed information about a selected product, providing users with essential details to make informed purchasing decision which Include

-> Header and Navigation

supremo

21.01.25

Designing of PDP Page

- Added key details including Product Name, Price, Description, Ratings and Review.

- Added to add to cart button with quality selector, allowing user to specify the number of items they want to order.

- Included a visual confirmation message when an item is added to cart.

Smriti
22.01.25

22.01.25

* Admin Dashboard design

* Created Product, Category, widget, and Dashboard page in admin dashboard

* Applied icons to it to enhance user experience.

23. 01. 25

Admin Dashboard Design

* Worked on the Product, and Category coloumns of the admin dashboard

* Fixed CSS of Total page.

24. 01. 25

Admin Dashboard design

* Worked on Category design page.

* Applied product page.

25.01.25

Admin Dashboard Design

- Worked on overall design of admin dashboard, which incelude search bar and navigation bar.
- Added icons and hover effect.

26.01.25

Git and GitHub

- Installation and download of Git and Github
- learned how to push code on Git-hub.
- Gitpull and Gitpush commands to git and github
- learned how to create new branch and ticket generation

supremo

27.01.25

Creation Of Controller Page

- Worked on developing the Controller layer for Gush-in web application.
- The controller handels user request processes them and return appropriate respons.
- Created Controller Classes :

  * Admin Controller
  * User Controller
  * Product Controller
  * Order Controller
  * Category Controller.

28.01.25

Controller Classes

* Used http methods appropriately like
  - → GET (Retrieve data)
  - → POST (Add new data)
  - → PUT (update existing data)
  - → DELETE (Remove data)

Smruthi
29.01.25

29.01.25

Controller Classes

* Implemented input validation using spring boots @valid annotation to prevent incorrect data submission

- spring MVC is used
- Java for backend logic implemented

supremo

30.01.25

Entity Class

* Worked on developing the entity layer for Yash-in web application.

* The entity class represents database tables and map the data to java objects, Enabling smooth interaction with My SQL database through JPA.

31.01.25

Entity Class

- Developed separate entity class like Category entity, Product entity Order item, Order and etc.

01.02.25

Auditable Class

* Created auditable Entity class which extends to other classes.

* The getter, setter methods

03.02.25

Entity Class

-> Utilized jpa annotation for table mapping

-> @Entity -> specifies the class as a JPA Entity

• @Table (name = "Table_name")
Maps the class to the corresponding table in the database.

* @ID, @Generated value @coloumn etc.

04.02.25

- Added validation annotation to ensure data integrity.
  @NotNull
  @Size
  @E-mail.

05.02.25

Creation Of Repositories

Worked on developing the repositories layer for Tash-in web application.

→ Developed separate repositories for key Entities.

* Product repository
* Category repository

* widget repository
* User repository

Suresh
06.02.25

06.02.25

- Used spring data JPA to extend the jpa Repository interface,

Providing built in methods like

- findAll()
- findByID()
- Save(Entity)
- deleteBy()

07.02.25

- Added custom query methods to fetch specific data efficiently

  → findByCategory
  → findByStatus etc.

- Used springboot JPA for Object relation mapping (ORM)

08.02.25

→ Worked on Enum class

→ Created Category status, Product status, Order status, widget status for Enums.

09 | 02.25

-> Worked on ProductController class to fetch the data from the backend using get mapping

10.02.25

-> Created widget page under admin dashboard

11.02.25

-> Worked on database table in My SQL.

-> ~~Inserted~~ Made a script table containing ~~product~~ details of products

12. 02. 25

→ Inserted values to database

→ Worked on CSS of admin dashboard

→ Overall space adjustment and design

Smruti
13.02.25

13.02.25

learned about Entity Entity relationship

→ Collection interface and about list like Arraylist, linkedlist, vector, stack etc.

14.02.25

→ Learned about JDBC connection, which include http-request, dispatcher Servlet / frontController handler mapper, Controller, view resolver etc.

15.02.25

→ Learned about code reusability used fragment in the area needed.

→ Created a fragment file of html and css which can include all the comman codes which can be reused in another html files.

16.02.25

→ Merged code in git-hub

→ Learned how to manage conflict in it when occured.

17.02.25

→ Learned about UUID Universally Unique identifier which are 12-bit labels used to universally identify objects in Computer system.

→ Implemented it on Entity class as @ Generated value (strategy = UUID)

18.02.25

Worked on database tables of Category and Product.
Along with it fetched data from backend for Product using GetMapping and Post Mapping.

19.02.25

Worked on Category Page and Product Page on admin dashboard

→ Used Thymleaf springboot dependency to render the data

Smruti
20.02.25

20.02.25

* Completed the admin Category Page and admin Product Page

* Worked on its data table

21/02/25

* Implemented CRUD operations.

(Create, Read, update, delete)
to manage catagories and products - allowing seamless interaction with my-SQL data base.

22·02·25

→ Utilized Thymleaf template engine for rendering dynamic html contents and displaying real-time data from backend.

23·02·25

- Created branch in github
- Merged the existing code to main branch.

24.02.25

→ Developed dynamic tables on the Category and Product Page of the web application using Spring boot and Thymeleaf.

→ Utilized Hibernate ORM (Object-Relational Mapping) to interact with MySQL database.

25.02.25

Integrating custom search, for enabling users to locate easily for specific Products.

26.02.25

- Ensured data consistency and integrity using hibernate's cascading operation and validation annotations.

- Worked on css of homepage.

Suryansh
27.02.25

27.02.25

- Created widget table
- Learned about Spring-boot security
- Implement in CSRF Token

28.02.25

-> Worked on upload-image

-> Created a folder upload-image in the explorer.

-> Put it on gitignore.

-> upload images for Product page.

1.03.25

-> Worked on search button

-> Used JPA Repository < Product, String > to leverage CRUD operation and simplify database interaction

02. 03. 25

Implemented the FindByProductID containing Ignore case (String ProductName) method to perform case sensitive search for Products names

→ Used spring Data JPA method naming convention to automatically generate query implementations reducing the need for custom SQL quiries.

06. 03. 25

Worked on backend of widget page and widget Table.

07.03.25

Created widget Entity class

Link it to the database

08.03.25

→ Created widget repository

Render data from database to frontend.

Smruthi

04.02.25

09.02.25

-> Created widget list front-end page.

-> Created CSV file of widget list which have product-id and widget id.

10.02.25

-> On admin dashboard render the dynamic data from the database and show it on frontend.

12.03.25

→ Worked on spring security

→ Applied javascript in the front-end for design enhancement.

→ Worked on CSS of widget page under admin dashboard.

13.03.25

→ Added model to the widget html page to show the widget list page.

14.03.25

Created dto repository under repository folder to extend JPA repository

→ Worked on backend of widget and widget list page.

16.03.25

* Worked on the designing part of the home page

* Bought data of PLP and PDP from backend

supremo

17·03·25

Created dtorepository under Repository folder to extend jpa repository.

→ Worked on backend of widget and widget list page.

Smruti
18.03.25

18·03·25

Created edit and delete button on category and Product Page under admin dashboard.

→ Worked on delete button.

→ Added Security to the application.

supremo

19.03.25.

Created a Config folder and security config class under where used @Bean.

Bcrypt password Encoder (12)
which will convert the password to hash form.

-> Applied exception handling using orElseThrow()

21.03.25

Worked on backend of edit and delete.

Pushed the code to github.

22.03.25

Worked on the overall security of the project

Worked on the edit and delete button of Product

Completed Dashboard designed completely.

Suryansh
22.03.25