Project Overview

SB Foods is a cutting-edge food website created to provide each and every one of its consumers with a smooth and interesting experience. The platform sets itself apart by delivering three different interfaces that are customized for patrons, managers, and restaurant partners, each of which has specialized features to meet their particular requirements.

Customer Interface

For foodies looking for a quick ordering process, the consumer interface is the main point of contact. A visually beautiful display of several food categories, including Indian, Italian, and desserts, along with highlighted and well-liked eateries, greets guests as soon as they land on the homepage. A large search bar makes it easy to navigate and find the foods or categories you're looking for. With thorough menus for every restaurant and extensive filtering and sorting options (including price, dietary preferences (vegetarian, non-vegetarian, or both), and ratings, the menu section is made to be both comprehensive and easy to use, enabling users to make well-informed decisions with ease. Customers can also easily manage their selections by adding or removing things from a virtual shopping cart through the interface.

Administrator Interface

The foundation of the SB Foods platform is the administrator interface, which offers crucial capabilities for system management and upkeep. By giving administrators the ability to add, remove, and monitor customer accounts, this extensive control panel makes effective customer administration possible. Similar to this, features that enable the addition of new restaurant partners, the deletion of out-of-date listings, and the maintenance of important restaurant details streamline restaurant administration. Administrators can effortlessly access, trace, and manage all customer orders with the use of a centralized order management system, guaranteeing timely and precise order fulfillment. Strong promotion management features are also included in the interface, giving administrators the ability to design, implement, and monitor a range of promotional offers and discounts, hence increasing consumer value. face.

Restaurant Interface

The restaurant interface was created especially for restaurant partners, providing a customized setting for managing their daily operations and online presence on SB Foods. By adding new items, updating current entries (such as names, descriptions, prices, and availability), or eliminating items as necessary, this interface helps restaurants manage their menus effectively. One important function is real-time order management, which gives eateries a quick overview of incoming orders and speeds up order processing. Based on their present operations, restaurants can then decide whether to accept or reject these

requests. Restaurants can also change important information including contact data, working hours, and visual images thanks to the interface's capability for profile management.

Purpose

Creating a comprehensive and user-friendly online food ordering ecosystem that meets the specific requirements of three important stakeholders—customers, administrators, and restaurant partners—is the main goal of the SB Foods website. SB Foods is intended to improve user experiences on all fronts and expedite operations by dividing the platform into discrete interfaces.

Customers may easily browse through a range of culinary categories on the website, including desserts, Indian and Italian cuisines, and more, in an interactive and visually appealing environment. Customers can easily find the cuisines and meals they want thanks to the user-friendly interface, which is enhanced by showcases of featured and well-liked restaurants and a powerful search engine. A safe and transparent order placement system strengthens consumer confidence and happiness, while comprehensive restaurant menus combined with customizable filtering and sorting options make the meal choosing process quick and individualized.

The website functions as a central control center from the administrator's point of view. Its goal is to streamline backend operations by offering effective solutions for managing orders, restaurants, and customers. In order to keep the platform current, dynamic, and in line with market trends, this interface also facilitates category and promotional administration.

 Last but not least, the goal of the restaurant partner interface is to enable eateries to control their online appearance. In order to ensure that food service operations are responsive and flexible, this segment enables partners to track orders in real time, alter their menus, and immediately manage consumer demands.
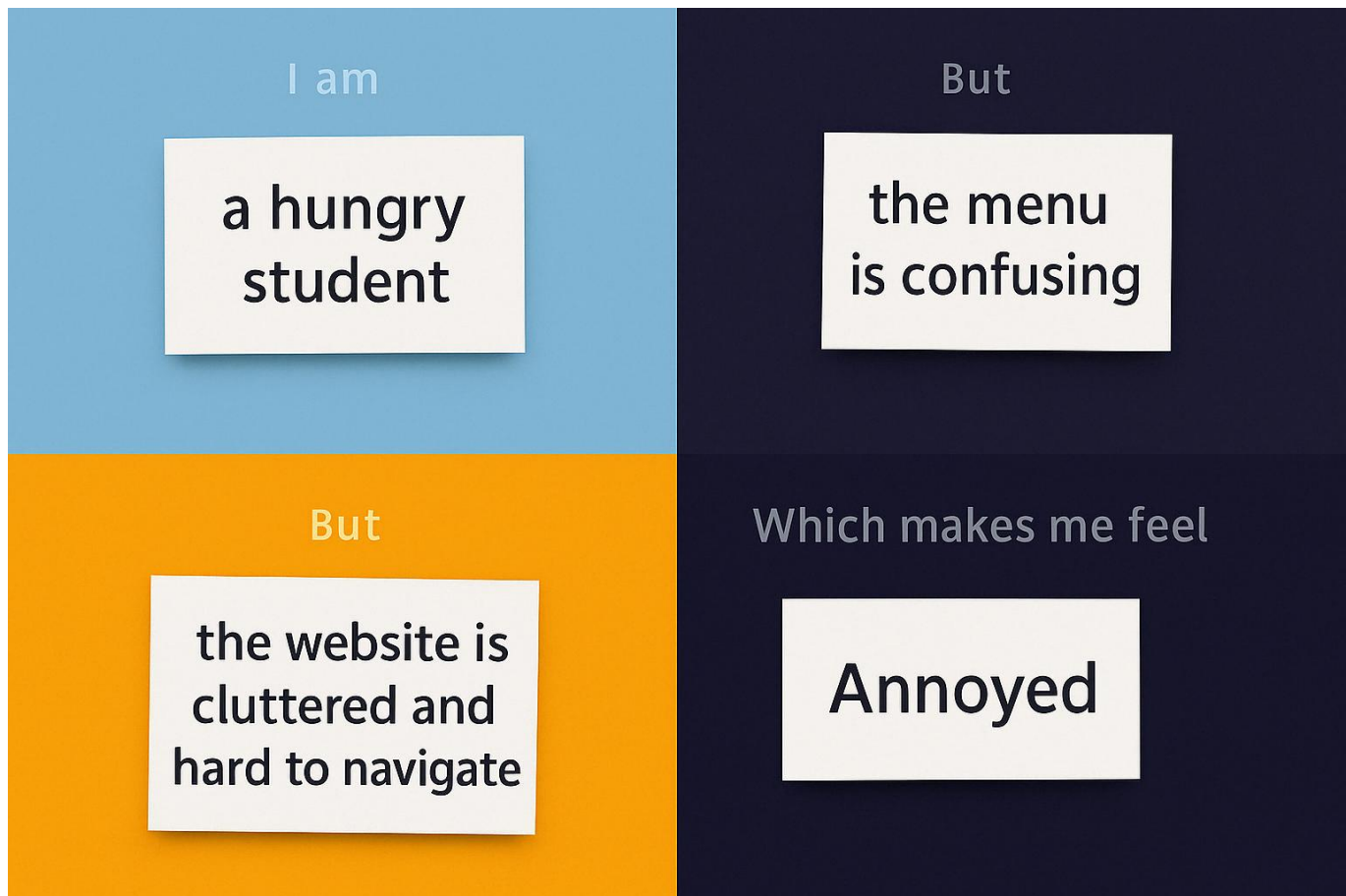
SB Foods essentially wants to combine these disparate experiences into a single, unified platform in order to provide a smooth service that benefits all users, boosts operational effectiveness, and cultivates a thriving online community for food ordering.

Phase Two: Ideation

The SB Foods project's ideation phase is an essential stage where the framework for the website's functionality and design is established via methodical consideration and imaginative investigation. The main goals of this phase are to discover the platform's fundamental issues, comprehend user wants via empathy, and produce a number of creative solutions to these issues. The team has created a user-centered, incredibly effective, and dynamic web interface by following a methodical process that involves creating a problem statement, conducting an empathy map canvas to analyze users, and holding group brainstorming sessions.
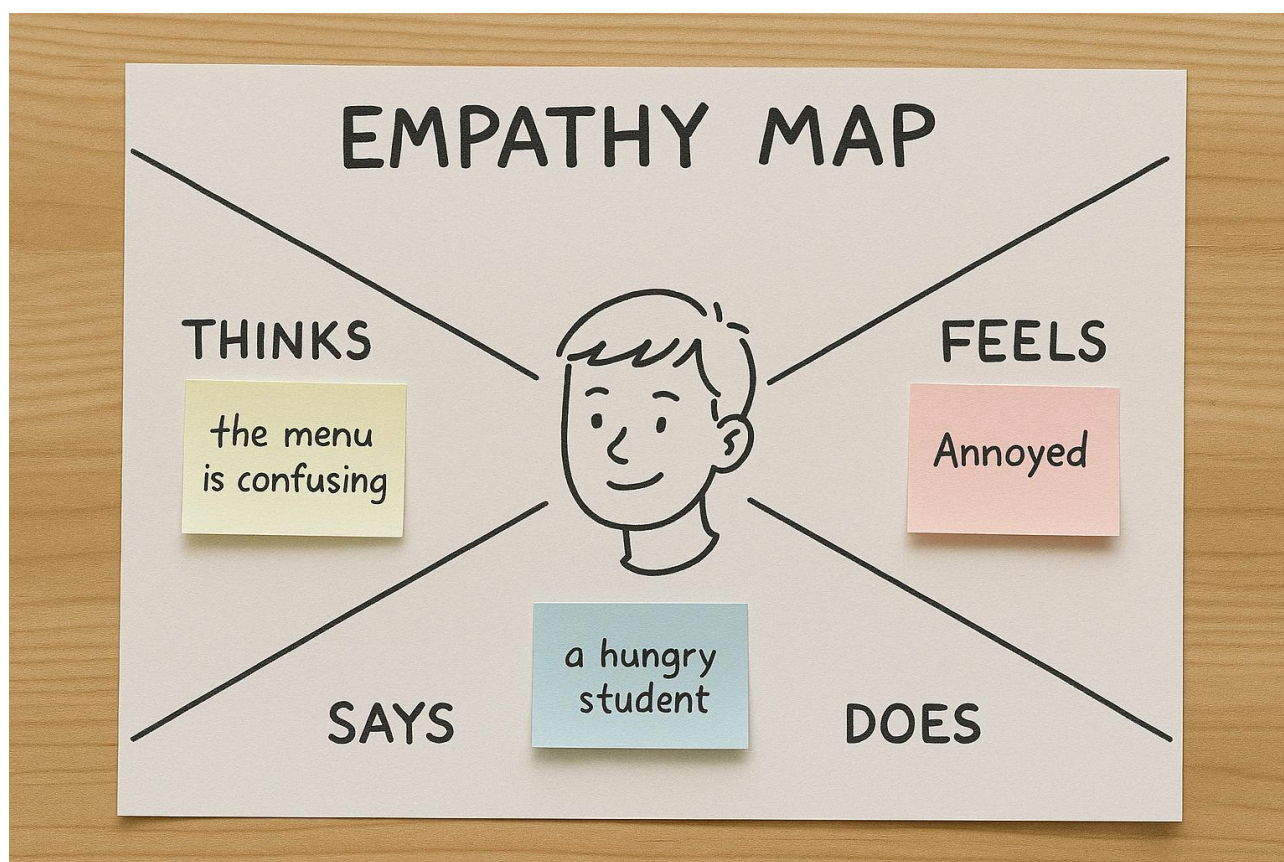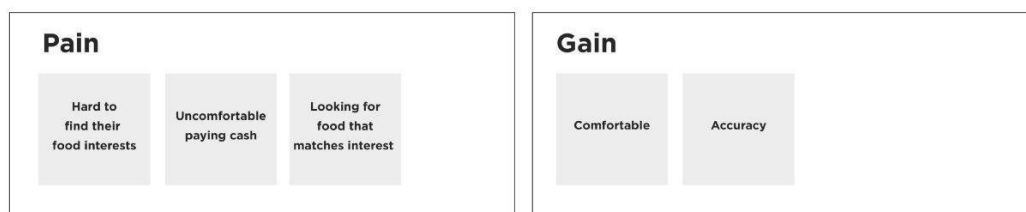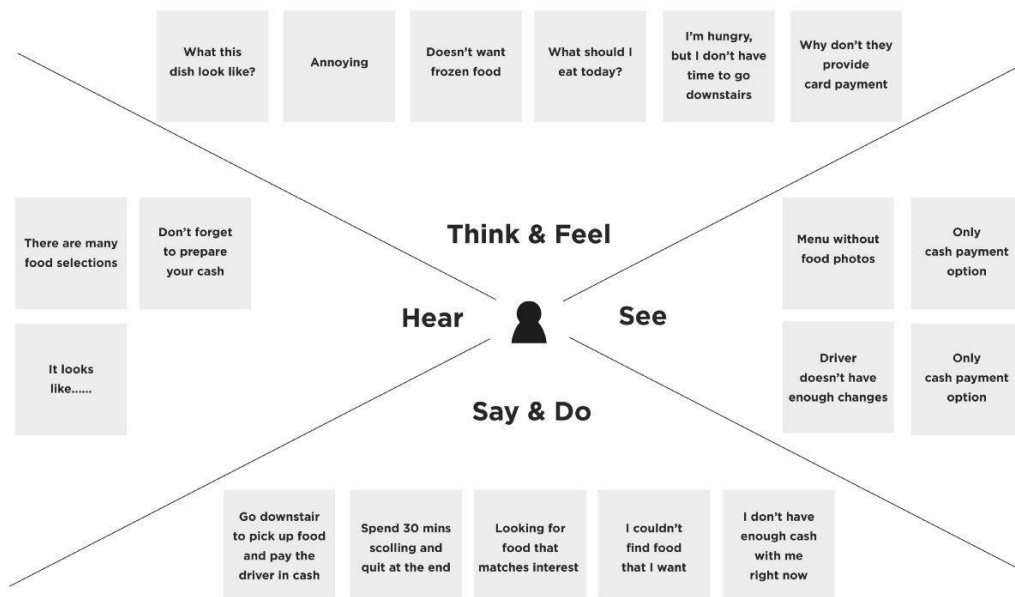
2.1 Problem Statement

Problem Statement (PS)	I am (Customer)	I'm trying to	But	Because	Which makes me feel
PS-1	a food lover who wants to quickly find and order my favorite dishes online	easily browse menus, filter by cuisine, and place an order without hassle	the website is cluttered, menus are hard to navigate, and filtering options are confusing	there are too many categories, inconsistent menu layouts, and limited sorting/filtering features	frustrated, overwhelmed, and less likely to complete my order
PS-2	an administrator responsible for managing restaurants, customers, and orders	efficiently update restaurant listings, manage user accounts, and monitor orders in real time	the admin panel is not centralized, and updating information is time-consuming	there are multiple disconnected modules and no unified dashboard	stressed, inefficient, and worried about errors or delays
PS-3	a restaurant partner who wants to manage my menu and process orders quickly	update menu items, change prices, and respond to orders in real time	the system is slow to update, and order notifications are delayed	the interface is not intuitive, and there is no real-time order management	powerless, anxious about missing orders, and dissatisfied with the platform

I am	But
a hungry student	the menu is confusing
But	**Which makes me feel**
the website is cluttered and hard to navigate	Annoyed

2.2 Empathy Map Canvas

The next crucial stage in the ideation process was to comprehend our users. To document each user group's experiences, expectations, and pain areas, the team created an Empathy Map Canvas. Deeper understanding of the everyday difficulties users encounter and the advantages they want to obtain from the SB Foods platform was made possible by this canvas.

Think & Feel

What this dish look like?

Annoying

Doesn't want frozen food

What should I eat today?

I'm hungry, but I don't have time to go downstairs

Why don't they provide card payment

Hear

There are many food selections

Don't forget to prepare your cash

It looks like......

See

Menu without food photos

Only cash payment option

Driver doesn't have enough changes

Only cash payment option

Say & Do

Go downstair to pick up food and pay the driver in cash

Spend 30 mins scolling and quit at the end

Looking for food that matches interest

I couldn't find food that I want

I don't have enough cash with me right now

Pain

Hard to find their food interests

Uncomfortable paying cash

Looking for food that matches interest

Gain

Comfortable

Accuracy

EMPATHY MAP

THINKS — the menu is confusing

FEELS — Annoyed

a hungry student

SAYS

DOES

. For Clients:

Says: "I want to easily and quickly locate and order my favorite dishes."

Considers: "This website should be quick, visually appealing, and simple to use."

Does: Looks through a variety of categories, filters menus, weighs possibilities, and places orders; uses the virtual shopping cart a lot.

Feelings: Joy when trying out new dishes, annoyance while navigating convoluted checkout procedures, and contentment when orders are handled efficiently. These observations advocate for an interface that maximizes discoverability, usability, and simplicity while minimizing friction.

.Regarding Administrators:

declares: "I need a tool that makes it easier to manage a lot of data and operations."

believes: "Efficiency and dependability are crucial; mistakes in order or customer management can cause the ecosystem as a whole to collapse."

Does: Runs promotions, changes menus and categories, keeps an eye on user and restaurant activity, and fixes problems as they come up.

Feels: Stress from managing several features at once while aiming for a flawless system, and finally, relief when difficult jobs are simplified with user-friendly backend tools. Administrators can focus on creating an interface that centralizes control functions and streamlines complicated processes by using the empathy map.

.Regarding Restaurant Partners:

"I need an interface where I can see and process orders instantly while updating my menu on the fly," the statement reads.

Thinks: "Keeping my menu and profile current has a direct impact on how clients view and use my services."

Does: Maintains operational details such as working hours and restaurant imagery; updates menu items, keeps track of incoming orders, and responds to consumer demands.

Feelings: annoyance when antiquated procedures impede operational responsiveness, and empowerment when the system is dependable and responsive. The restaurant partners'

empathy map highlights the significance of personalized profiles, real-time order management, and an agile system that allows for frequent modifications.

The team was able to directly address each user's functional and emotional demands throughout the design process by mapping these user viewpoints. This gave a strong basis for setting feature priorities and improving the user interfaces in line with those priorities.

2.3 Brainstorming

Step-1: Team Gathering, Collaboration and Select the Problem Statement
Team Members Present:

- Mayank Yadav (UI/UX Designer)

- Ketak Singh (Backend Developer)

- Ramaditya Chaudhary (Frontend Developer)

- Yash Dhanaji Powar (Database Developer)

Collaboration Tools Used:

- Google Meet

- Github

Selected Problem Statement:
How can we create a cohesive food ordering platform that effectively meets the diverse needs of customers, administrators, and restaurant partners, ensuring usability, efficient content management, and prompt order processing?

Step-2: Brainstorm, Idea Listing and Grouping
Raw Ideas Generated:

- Real-time order tracking for customers and restaurants

- Search bar

- Admin dashboard with live data and full control

- Menu management with instant updates for restaurants

- Personalized restaurant recommendations

- In-app notifications for order status

- Secure payment gateway integration

- Automated promotional offers

Grouping of Ideas:

- Customer Experience: Real-time tracking, search bar

- Admin Efficiency: Live data dashboard, automated promotions

- Restaurant Partner Tools: Instant menu updates, order management, profile customization

- Platform Security: Secure payments, data privacy, role-based access

Step-3: Idea Prioritization
Prioritization Criteria:
- User Impact

- Technical Feasibility

- Business Value

- Implementation Effort

3. REQUIREMENT ANALYSIS

Customers, administrators, and restaurant partners are the three different user groups that the SB Foods platform is intended to service. Each interface is customized to provide a particular set of features. The structure, procedures, and underlying technologies required to provide a seamless, effective, and secure experience for all stakeholders are outlined in this requirement analysis. A thorough customer journey map, solution requirements, a data flow overview, and a description of the technological stack comprise the many elements that comprise the analysis.

3.1 Customer Journey Map

The steps a user takes from the moment they land on the website until they place an order are graphically represented by the customer journey map. By precisely identifying important touchpoints and chances to raise engagement and happiness, it is intended to maximize the user experience.

Knowledge & Access: The homepage is the first thing that customers see when they come. With its prominent and well-liked restaurant parts, the interface is aesthetically pleasing and displays a variety of culinary categories, including Indian, Italian, desserts, and more. Users can easily find what interests them thanks to a prominent search bar.
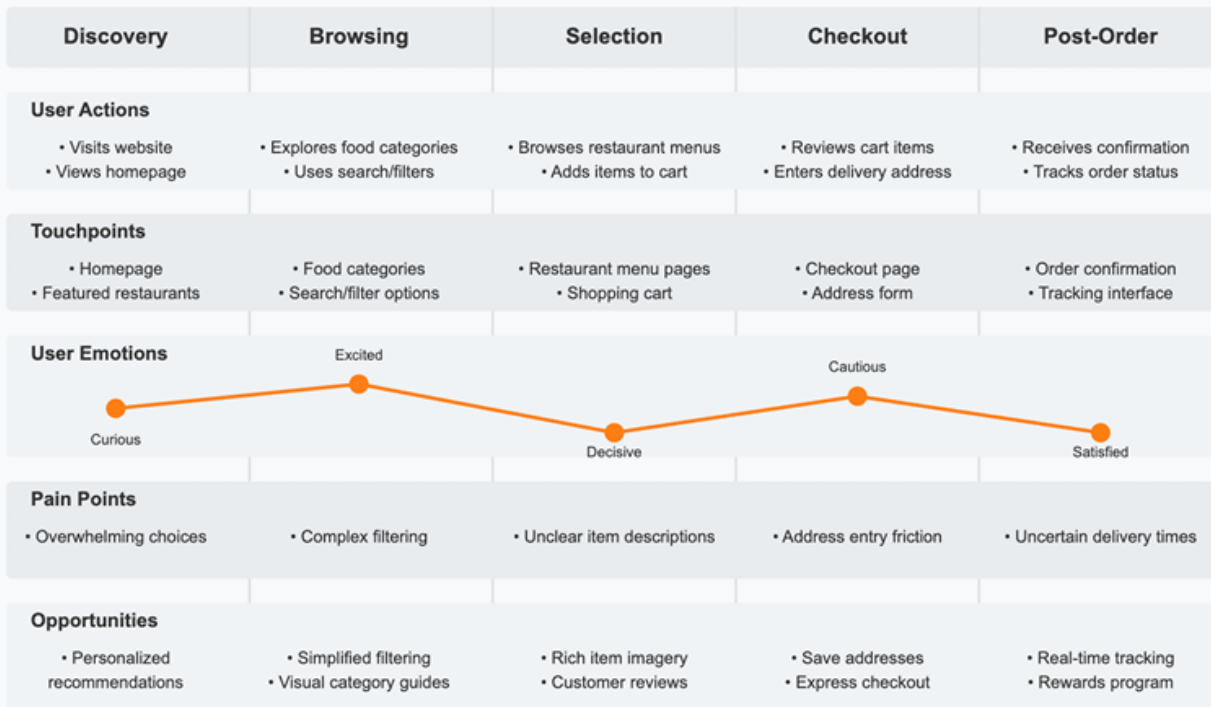
Selection & Exploration: After accessing the website, consumers start looking through particular menus. In addition to providing extensive filtering options including sorting by pricing (low to high/high to low), dietary choices (vegetarian/non-vegetarian), and customer ratings, the interface displays comprehensive restaurant menus. Additionally, this stage offers interactive features like the ability to add products to a virtual shopping cart, giving consumers a smooth approach to choose what they might like to buy.

Order placement and checkout: Following menu item selection, clients examine their order in a secure checkout section that shows an itemized list with the relevant pricing and taxes. Customers can choose or add a delivery address that has already been saved. The final purchase step is made simpler by the order summary and total cost being presented clearly, giving users confidence in their choices before confirming the transaction.

Engagement After Purchase: After a successful order placing, real-time order tracking and notifications are provided. From purchase confirmation until delivery, the system is made to keep users informed about the progress of their goods, promoting constant interaction and recurring platform usage.

SB Foods Customer Journey Map
Food Ordering Website - Customer Interface

Discovery	Browsing	Selection	Checkout	Post-Order
User Actions				
• Visits website • Views homepage	• Explores food categories • Uses search/filters	• Browses restaurant menus • Adds items to cart	• Reviews cart items • Enters delivery address	• Receives confirmation • Tracks order status
Touchpoints				
• Homepage • Featured restaurants	• Food categories • Search/filter options	• Restaurant menu pages • Shopping cart	• Checkout page • Address form	• Order confirmation • Tracking interface
User Emotions	Excited		Cautious	
Curious		Decisive		Satisfied
Pain Points				
• Overwhelming choices	• Complex filtering	• Unclear item descriptions	• Address entry friction	• Uncertain delivery times
Opportunities				
• Personalized recommendations	• Simplified filtering • Visual category guides	• Rich item imagery • Customer reviews	• Save addresses • Express checkout	• Real-time tracking • Rewards program

3.2 Solution Requirement

scalability, security, and performance. Specific requirements for the solution include:

Requirements for Customer Interface:

Intuitive Design: A visually appealing homepage that highlights restaurant features and makes it simple to navigate through cuisine categories.

Dynamic Menu Display: Detailed menus that enable thorough sorting and filtering according to ratings, price, and dietary requirements.

A secure checkout procedure, an interactive shopping cart, a well-defined order summary, and customized shipping options are all components of seamless order management.

Requirements for the Administrator Interface:

Centralized Management: A control panel for tracking orders across all transactions, integrating restaurant data, and managing customer accounts.

Promotion & Category Tools: Features for creating and overseeing promotions, discounts, and regularly updated food categories to keep the site current.

Data Security & Performance: Measures taken to guarantee that private information, such as user profiles, transaction histories, and restaurant specifics, is managed safely and readily for an expanding user base.

Requirements for the Restaurant Partner Interface:

Restaurants can view incoming orders, approve or reject them, and alter order statuses in real time with the help of real-time order processing tools.

Menu and Profile Updates: The ability to regularly update their operation hours, restaurant photos, and menu items, as well as to add, alter, or remove them from their internet presence.

Capabilities for Integration: smooth communication with the main system to maintain service quality and give clients correct information.

3.3 Data Flow Diagram

The SB Foods system's information flow is depicted by the Data Flow Diagram (DFD), which makes sure that all of the modules are communicating effectively to facilitate user interactions.

Outside Parties and Contributions:

Consumers: Enter search terms, put products in their shopping carts, and finish checkout processes.

Administrators: Oversee orders, promotions, restaurant information, and customer data.

Restaurant Partners: Edit profile information, process orders, and update menus.

Procedures:

User authentication and data validation: confirms login information and guarantees that any data entered by patrons, administrators, or restaurant partners is secure and correct.

Order Processing and Tracking: Coordinates with payment gateways, updates statuses in all associated systems, and oversees transactions from order placing to real-time tracking.

Content and Menu Management: Takes care of the dynamic updates for the restaurant profiles, menu categories, and advertising banners.

Stores of Data:

User Database: Safely stores administrator and customer account information.

establishment Database: Holds operational data, menu information, and establishment profiles.

Transaction logs: Document each order that is processed, facilitating troubleshooting and analytics.

By guaranteeing smooth data flow between user interfaces and backend operations, the DFD lowers the possibility of bottlenecks and preserves the system's information integrity.

3.4 Technology Stack

The technology stack for SB Foods must incorporate contemporary tools and frameworks that complement the project's main goals in order to create a reliable and scalable solution.

Front-end programming:

The building blocks for creating an interactive, responsive, and aesthetically pleasing user interface are HTML5, CSS3, and JavaScript.

Contemporary Frameworks: React or Angular are used to create dynamic elements and guarantee seamless page and device navigation.

Back-End Development: Server-Side Languages: Python (Django/Flask) or Node.js are choices for managing the development of the API and business logic.

Application Server: Apache or Nginx for effective HTTP request management.

Systems for databases:

Relational databases, which are ideal for managing structured customer, order, and restaurant data, include MySQL and PostgreSQL.

NoSQL databases: MongoDB could be added to store session data or cache frequently asked queries to improve performance.

Gateways for payments and security:

To ensure safe data transfer throughout the platform, SSL/TLS encryption is used.

Payment Processor Integration: Utilizing services like PayPal or Stripe to guarantee safe and dependable transactions throughout the checkout process.

Extra Resources:

Version Control: Git for collaborative development and source code management.

Cloud services: scalable hosting options from AWS or Azure guarantee that the system will remain highly available even as user demand increases.

4. PROJECT DESIGN

The goal of SB Foods' project design is to convert customer needs into a concrete, fully operational online meal ordering system. The design phase focuses on drawing out a strong architectural framework that guarantees scalability, security, and a flawless user experience by carefully analyzing and creatively solving issue areas. The three main components of the design are the Solution Architecture, the Proposed Solution, and the Problem Solution Fit.

4.1 Problem Solution Fit

Delivering a consistent experience to three different user groups—customers, administrators, and restaurant partners—is the main issue SB Foods is attempting to solve. The design breaks down the issue into three main areas of pain: the necessity for restaurant partners to efficiently update and manage their orders in real time; administrative overburden from managing large datasets and operations; and a cluttered interface that could impede down customer interaction.

By creating a customized interface for every stakeholder, the problem-solution fit is accomplished:

Customers benefit from an easy-to-use homepage, comprehensive menu sections, and a safe, efficient checkout process. Adding dynamic filters (such pricing, dietary restrictions, and ratings) streamlines the decision-making process and improves navigation in general.

A robust backend system that centralizes order, restaurant, and customer management is advantageous to administrators. By facilitating quick data updates, promotional management, and strong reporting features, this lowers operational friction.

A specialized interface that prioritizes effective menu management, configurable profile controls, and real-time order updates empowers restaurant partners. This guarantees that eateries can keep an updated online presence and react quickly to incoming orders.

 This focused strategy not only tackles the unique difficulties faced by each group, but it also supports the main objective of developing a platform that is effective and user-focused.

1. CUSTOMER SEGMENT(S)	5, CUSTOMER CONSTRAINTS	5. AVAILABLE SOLUTIONS
Hungry students People looking for a fast, convenient meal delivery option	• Limited budget for meals • Time constraints from academic schedules • Limited patience for lengthy ordering processes	• Other food delivery apps (Uber Eats, DoorDash, etc.) • Takeout options from local restaurants • Campus dining facilities
2. JOBS-TO-BE-DONE / PROBLEMS	6. CUSTOMER ROOT CAUSE	7. BEHAVIOUR
Need to order meals quickly between classes or study sessions • Frustrated with limited variety or poor ordering experience	• Current menus are poorly organized • Overwhelming information leading to Indecision and errors	• Users abandon carrts or exit site it they are frustatred • They tend to favor simple, easy-to-use interfaces
4. EMOTIONS: BEFORE / AFTER	9. YOUR SOLUTION	10. CHANNELS & BEHAVIOUR
Before: Hungry, frustrated and rushed with complex odering process Atter: Students feel relieved and at ease with simplified.	• SB Foods displays menus in a clear, organized manner with categories and highlights • Search, filtering, and sorting functions aid quick navigation	• Website as the main platform, potential mobile app extension • Social media marketing and empura ements for initial user engagentent

4.2 Proposed Solution

Project team shall fill the following information in the proposed solution template
.

S.No.	Parameter	Description
1.	Problem Statement (Problem to be solved)	Students and working professionals often face difficulty in ordering meals conveniently and quickly, especially during tight schedules. Existing platforms are either too complex, slow, or lack user-friendly navigation, leading to ordering mistakes or customer dissatisfaction.
2.	Idea / Solution description	Our solution is a modern, responsive food delivery website, SB Foods, that offers an intuitive, well-organized menu structure, real-time order tracking, and multiple payment options. The platform focuses on

		simplicity and speed to provide a hassle-free food ordering experience.
3.	Novelty / Uniqueness	SB Foods distinguishes itself by offering a student-centric and time-efficient interface with smart recommendations, category filtering, and a seamless checkout process. Our real-time order updates and personalized suggestions improve user satisfaction and reduce ordering errors.
4.	Social Impact / Customer Satisfaction	The platform reduces food delivery stress for busy students and professionals by minimizing delays and improving the overall ordering experience. By offering transparent, quick, and reliable services, it encourages healthy eating habits and customer loyalty.
5.	Business Model (Revenue Model)	Revenue will be generated through commission on each order from partnered restaurants, premium subscription plans for free delivery, ad space for local eateries, and occasional sponsored offers.
6.	Scalability of the Solution	The solution can be easily scaled to new cities or campuses by onboarding local food vendors and cloud kitchens. It also allows smooth expansion into a dedicated mobile app and integration with third-party payment and delivery services.

4.3 Solution Architecture

SB Foods' underlying architecture has been thoughtfully created to enable its multi-interface strategy while guaranteeing excellent performance, dependability, and simplicity of maintenance. The front-end, back-end, and data management components make up the solution architecture:

The front-end layer This layer is in charge of providing a dynamic and rich user experience. The foundation of responsive and dynamic interfaces is made up of technologies like HTML5, CSS3, and JavaScript combined with contemporary frameworks like React or Angular. To make sure that the administrative dashboards, restaurant management tools, and customer-facing sites are unique and unified, the front-end is divided into multiple modules, one for each user group.

Back-End Layer: Using frameworks like Django or Flask, the back-end makes use of a microservices architecture constructed with server-side languages like Python or Node.js. APIs manage business logic, order processing, and real-time data updates, acting as the link between the different services. This layer uses a strong security mechanism to stop data breaches and unwanted access, and it communicates with external payment gateways to ensure safe transactions.

Integration and Management of Data: To effectively store, manage, and retrieve data, a mix of relational databases (MySQL or PostgreSQL) and NoSQL solutions (like MongoDB) are used. NoSQL databases are best suited for session data and caching frequently asked queries, while relational databases work better for structured data such as user accounts, orders, and restaurant details.

All system components process food orders and profile updates safely and dependably.

Cloud-Based Infrastructure and Extra Services: Cloud platforms like AWS or Azure host the application to provide scalability and high availability. By offering load balancing, automated backups, and elastic resources, these services make sure the system can withstand fluctuating loads. The development process is supported by version control systems such as Git, which facilitate cooperation and seamless updates via pipelines for continuous integration and deployment.

5. PROJECT PLANNING & SCHEDULING

5.1 Project Planning

Product Backlog, Sprint Schedule, and Estimation (4 Marks)

Sprint	Functional Requirement (Epic)	User Story Number	User Story / Task	Story Points	Priority	Team Members
Sprint-1	Registration	USN-1	As a user, I can register for the application by entering my email, password, and confirming my password.	2	High	Ramaditya Chaudhary, Ketak Singh
Sprint-1		USN-2	As a user, I will receive a confirmation email once I have registered for the application.	1	High	Mayank Yadav, Yash Dhanaji Powar
Sprint-2		USN-3	As a user, I can register for the application through Facebook.	2	Low	Ketak Singh, Yash Dhanaji Powar
Sprint-1		USN-4	As a user, I can register for the application through Gmail.	2	Medium	Ramaditya Chaudhary, Mayank Yadav
Sprint-1	Login	USN-5	As a user, I can log into the application by entering my email & password.	1	High	Ketak Singh, Yash Dhanaji Powar
Sprint-2	Dashboard	USN-6	As a user, I can view a personalized dashboard with order history, saved addresses, and payment methods.	3	High	Ramaditya Chaudhary, Mayank Yadav
Sprint-2		USN-7	As a user, I can edit my profile information including email, phone number, and address.	2	Medium	Ketak Singh, Yash Dhanaji Powar
Sprint-3	Menu	USN-8	As a user, I can view the menu items categorized by type (e.g., burgers, pizzas, desserts, etc.).	3	High	Ramaditya Chaudhary, Mayank Yadav
Sprint-3		USN-9	As a user, I can search for specific items by name or ingredients.	2	Medium	Ketak Singh, Yash Dhanaji Powar
Sprint-3	Cart	USN-10	As a user, I can add items to my cart and proceed to checkout.	2	High	Ramaditya Chaudhary, Mayank Yadav

Project Tracker, Velocity & Burndown Chart: (4 Marks)

Sprint	Total Story Points	Duration	Sprint Start Date	Sprint End Date (Planned)	Story Points Completed (as on Planned End Date)	Sprint Release Date (Actual)
Sprint-1	20	3 Days	5 April 2025	8 April 2025	20	15 April 2025
Sprint-2	20	5 Days	8 April 2025	12 April 2025	20	15 April 2025
Sprint-3	20	2 Days	13 April 2025	14 April 2025	20	15 April 2025
Sprint-4	20	2 Days	14 Mar 2025	15 Mar 2025	20	15 April 2025

Velocity:

$$AV = \text{Sprint Duration / Velocity} = 20/12 = 1.67$$

Burndown Chart:

A burn down chart is a graphical representation of work left to do versus time. It is often used in agile software development methodologies such as Scrum. However, burn down charts can be applied to any project containing measurable progress over time.

Burndown Chart - SB Foods Project
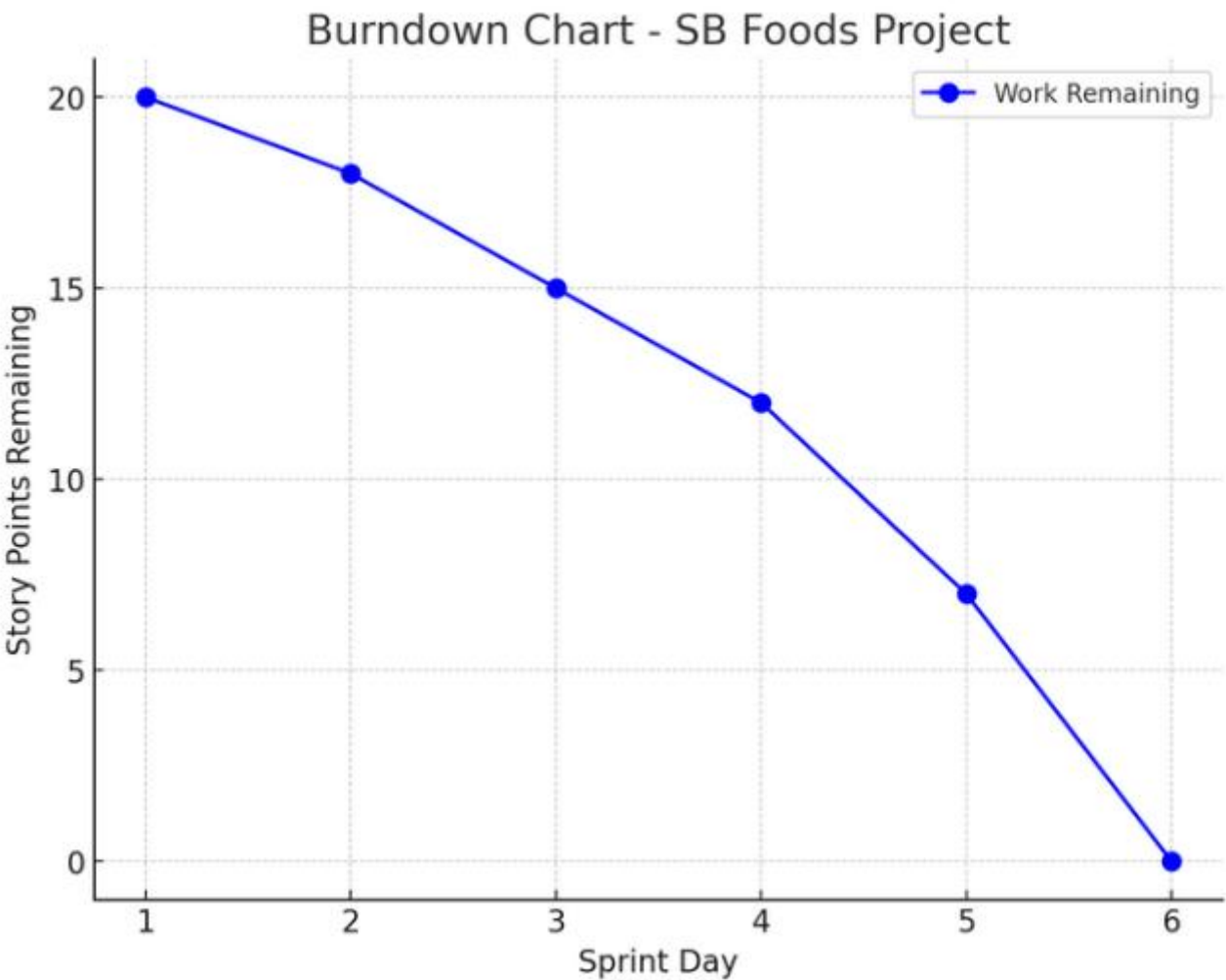
6. FUNCTIONAL AND PERFORMANCE TESTING

6.1 Performance Testing

To make sure the system satisfies the necessary speed, scalability, and reliability requirements while providing a smooth user experience across its three separate interfaces—Customer, Administrator, and Restaurant Partner—performance testing of the SB Foods website is essential. Verifying that the website can effectively manage fluctuating loads, sustain quick response times, and securely execute transactions in both typical and high-volume scenarios is the aim.

Important Goals:

Verification of Response Time: The testing will concentrate on gauging how quickly users can complete important tasks including filtering menu items, looking up food categories, and finishing a safe checkout process. We can make sure that pages—particularly the visually appealing homepage and comprehensive menu sections—load quickly and that important features like the search bar and order placement are extremely responsive by modeling a variety of user scenarios.

Load and Stress Testing: In order to assess SB Foods' resilience, load testing will replicate several users carrying out tasks at once. For example, the system's ability to sustain performance without lag or crashes will be evaluated during periods of peak usage, when a large number of consumers are concurrently perusing menus and placing orders. In order to determine the breaking point and create efficient recovery plans, stress testing will put the system under more strain than it usually does at its peak.

Scalability evaluations: The architecture of the website needs to grow with the number of users. All interfaces will be subjected to performance tests that simulate increased traffic, with a focus on the administrator and restaurant panels that manage frequent data updates and real-time order processing. These tests will assist in determining whether load balancers and auto-scaling mechanisms are set up correctly to divide traffic among servers in an equitable manner.

Database Performance: When handling large product catalogs, client orders, and restaurant information, database query efficiency is especially important. Performance testing involves determining how quickly CRUD (Create, Read, Update, Delete) operations are completed, confirming that data retrieval is streamlined, and making sure that the customer menu section's sophisticated filtering and sorting options function dependably under load.

Integration and Security Checks: Performance testing is necessary to make sure that interactions with external APIs and payment gateways don't cause latency or jeopardize security. Secure user authentication procedures and SSL/TLS-encrypted transactions are thoroughly tested to ensure they don't impair system performance as a whole.

Testing Methodology and Tools: To simulate realistic user behaviors across a range of usage scenarios, the performance testing strategy makes use of automated testing tools like Apache JMeter and LoadRunner. Real-time data collection and analysis is done on metrics including transaction rates, throughput, response times, and error rates. Furthermore, in order to identify bottlenecks and direct additional optimization, monitoring tools offer insights into the usage of system resources, such as CPU, memory, and network bandwidth.

7. RESULTS

7.1 Output Screenshots (with Description)

The SB Foods platform has been successfully developed and tested across all three user interfaces. Below is a detailed breakdown of the output observed during implementation:

Customer Interface

- Homepage: Displays food categories (Indian, Chinese, Continental, Desserts, etc.) with attractive images and a dynamic section for featured restaurants based on popularity and ratings.

- Search Bar: Provides real-time suggestions and filtering options based on typed keywords.

- Menu Page: Lists available dishes per restaurant. Users can filter by price, rating, and dietary preferences. A cart section dynamically updates as items are added or removed.

- Checkout Page: Includes an order summary, delivery address form, total bill, and a secure payment gateway simulation.

- Order Confirmation Page: Displays confirmation ID, estimated delivery time, and tracking link.

Search restaurants, cuisine, etc.,

No items found... try searching for Biriyani, Pizza, etc.

Delicious Food Delivered Fast

Order from your favorite local restaurants with just a few taps

Order Now

Breakfast

Biriyani

Pizza

Noodles

Burger

Popular Restaurants

Bella Italia
123 Main Street, New York, NY 10001

All restaurants

Bella Italia
123 Main Street, New York, NY 10001

Spice Garden
456 Park Avenue, New York, NY 10002

Dragon Wok
789 Broadway, New York, NY 10003
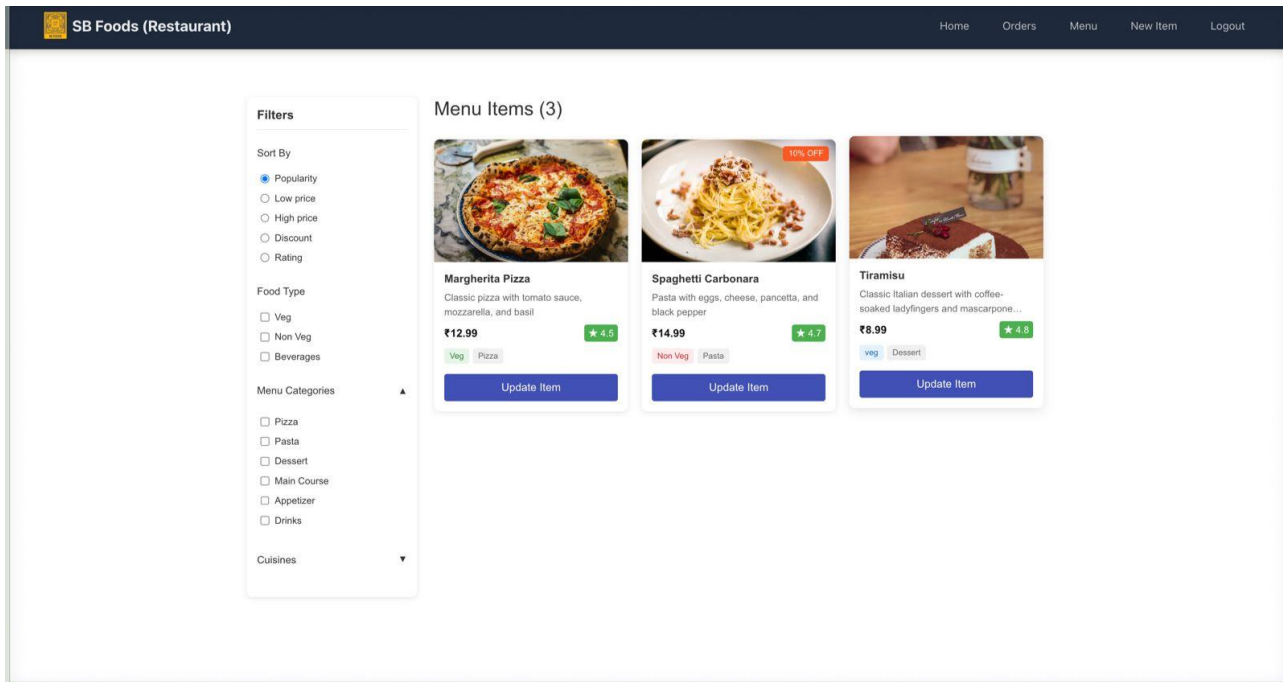
**Administrator Interface**

- Dashboard: Visual representation of total users, total orders, active restaurants, and promotional campaigns.

- Customer Management: Admins can view, add, delete, or modify customer information.

- Restaurant Management: Admins can onboard new restaurants, edit their menus, and disable accounts if needed.

- Order Management: Central panel for viewing all incoming orders with details like customer name, restaurant, and status (Pending, In Process, Delivered).

- Promotions Panel: Section to add/edit discount codes, set expiration dates, and apply them to specific categories or restaurants.

Total users

4

View all

All Restaurants

3

View all

All Orders

2

View all

Popular Restaurants(promotions)

☑ Bella Italia
☑ Spice Garden
☐ Dragon Wok

Update

Approvals

Restaurant
restaurant_owner2   Approve   Reject

All Users   Add New User

User Id	User Name	Email Address	User Type	Status	
67fe7f51a39df18e3d8fb149	customer1	customer1@example.com	customer	approved	Delete User
67fe7f51a39df18e3d8fb14a	customer2	customer2@example.com	customer	approved	Delete User
67fe7f51a39df18e3d8fb14b	restaurant_owner1	owner1@example.com	restaurant	approved	Delete User
67fe7f51a39df18e3d8fb14c	restaurant_owner2	owner2@example.com	restaurant	pending	Delete User

---

Restaurant Interface

- Login and Dashboard: Restaurant partners log in to access their own dashboard.

- Menu Editor: Add, remove, and update dishes along with name, description, price, dietary label, and availability status.

- Incoming Orders View: List of active orders with timestamps and item details.

- Profile Settings: Allows restaurants to update contact details, open hours, images, and restaurant information.
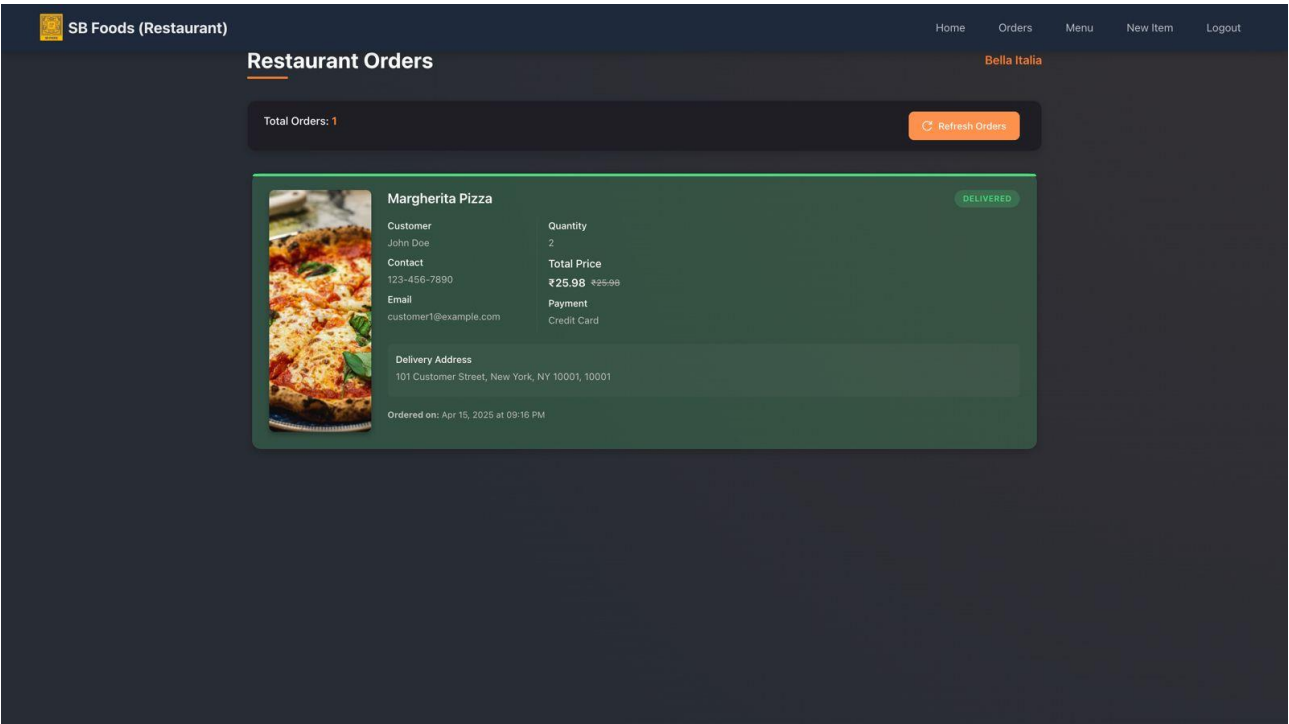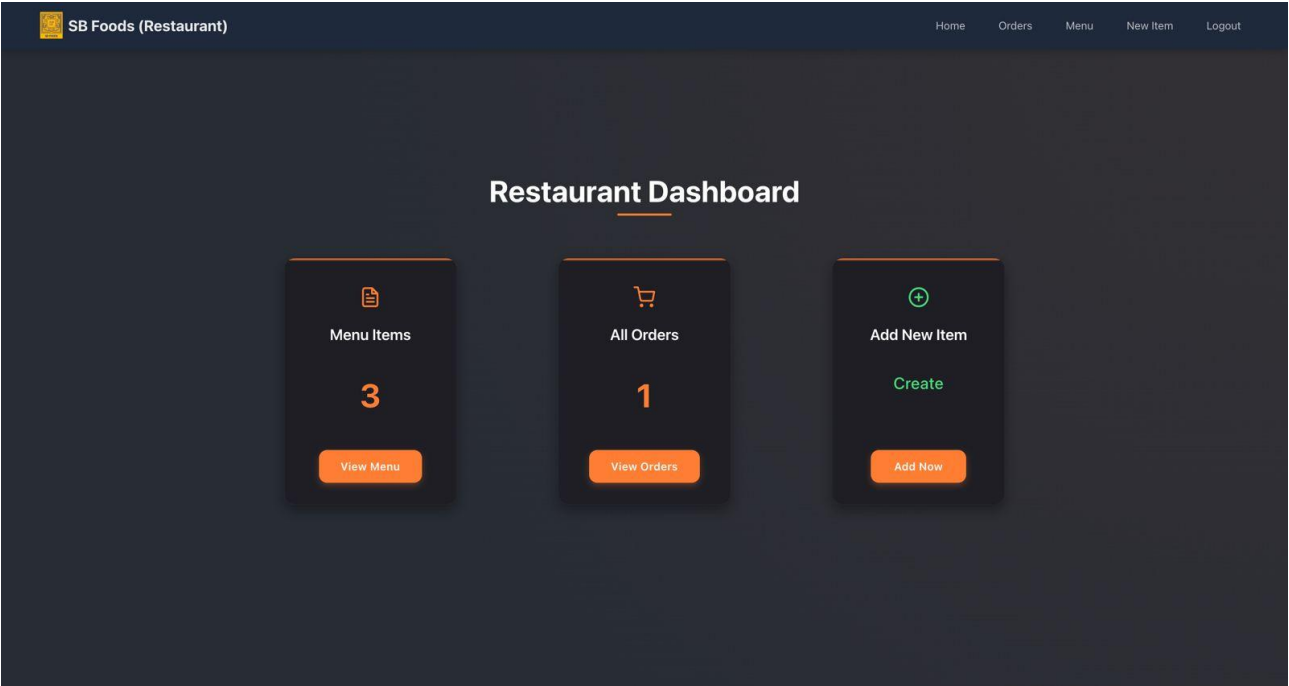
8. ADVANTAGES & DISADVANTAGES

1. **Advantages**

Role-Based Access Control (RBAC)
The platform is built with a solid role-based access control system, which guarantees that users can only access the features and data appropriate to their role. Customers, administrators, and restaurant partners all have their own specific dashboards and functionalities suited to their respective needs. This method improves security, reduces navigation complexity, and offers a more targeted user experience.

Enhanced User Experience and Responsive Layout
SB Foods has a sleek and responsive user interface with responsiveness in its design. The structure automatically resizes for desktop, tablet, and mobile layouts, making it accessible on every screen size. The uncluttered navigation path, interactive elements, and explicit call-to-action components ensure a seamless and enjoyable browsing and ordering experience.

Flexible and Dynamic Menu Management System
Restaurant partners are given full control over their menu. They are able to add new items on their own, edit descriptions, change prices, and switch between availability states—all in real-time. This eliminates the requirement of backend support and allows restaurants to respond rapidly to fluctuating inventories or consumer trends.

Centralized Administrative Dashboard
The administrator interface is an operations control centre for the platform. It is a robust management tool for managing customers, restaurants, orders, and campaigns. Admins are able to monitor platform activity, view order streams, onboard/offboard partners, and even switch on/off specific food categories from being visible, making operational control effective and seamless.

Scalable and Modular Architecture
Developed with the MERN stack (Node.js, Express.js, React.js, MongoDB), the platform is extremely modular and scalable. New interfaces or features can easily be added without impacting the fundamental functionalities. Microservice-friendly architecture makes horizontal scaling possible, which is suited to manage more traffic or adding new cities and regions down the line.

2. Disadvantages

Lack of Real Payment Gateway Integration

Currently, the checkout process in SB Foods is simulated for demonstration purposes. There is no real integration with payment gateways such as Razorpay, Stripe, or PayPal. As a result, the system does not support actual monetary transactions, limiting its readiness for commercial deployment.

Absence of Live Delivery Tracking

Although customers receive confirmation and order details, the platform does not yet feature live GPS-based tracking of food delivery. This limits transparency and real-time updates for users, which are common expectations in modern food delivery systems like Swiggy or Zomato.

No AI-Powered Personalization or Recommendations

The system currently lacks advanced machine learning or AI algorithms that could provide personalized dish or restaurant recommendations. Features like "Recommended For You" or "People Also Ordered" are absent, which could enhance engagement and order frequency if implemented.

Time-Consuming Initial Setup and Data Population

Before full-scale operation, the platform requires a comprehensive setup process. This includes onboarding multiple restaurants, uploading detailed menus, and configuring operational hours and categories. The process can be labor-intensive, especially if scaled to multiple cities or regions.

Basic Security Implementations

While basic authentication and role management are in place, the current system does not include advanced security measures such as multi-factor authentication (MFA), end-to-end encryption, or secure API token handling. These are essential for preventing data breaches and ensuring compliance with data protection standards during real-world deployment.

9. CONCLUSION

The SB Foods website is a complete and scalable solution in the food delivery space, designed to make the interaction between customers, administrators, and restaurant partners easier. With its role-based functionalities, modular architecture, and intuitive interface design, the platform manages to provide a seamless digital food-ordering experience that is customized to the specific needs of each user group.

From a customer's viewpoint, the website is centered on simplicity, convenience, and customization. Customers can search through numerous cuisines, narrow down dishes according to their dietary needs, see popular restaurants, and order food with a few clicks. The minimalistic interface and responsive design make it accessible across devices, making it a more satisfactory experience for the user.

For administrators, the platform serves as a robust management console. Admins are able to monitor all operational facets of the application — from restaurant onboarding and user account management to monitoring order statuses and running promotional campaigns. The centralized dashboard enables increased transparency, operational effectiveness, and the capability to make informed decisions based on data.

The restaurant interface is designed with flexibility, allowing partners to control their digital storefronts without backend intervention. Restaurant owners are able to change menus, change item availability, process incoming orders in real-time, and change their profile information easily. This allows small and large food vendors to have an active and responsive online presence.

From a technical perspective, the system benefits from the ease of employing the MERN stack, which employs MongoDB, Express.js, React.js, and Node.js in a full-stack, single-language (JavaScript) setup. Dynamic routing, modular components, and the schema flexibility of MongoDB complement RESTful APIs in a system that is both maintainable as well as scalable to accommodate future development.

In addition, the project also showcases the merits of user-centered design principles. Every interface is specifically designed for its target user group to have a seamless experience and enhance task performance. This commitment to real-world usability further justifies the feasibility of commercial deployment with minimal modifications, including the incorporation of a secure payment gateway and real-time delivery tracking.

In summary, SB Foods is not just a college-level web development project — it is an operational prototype of a solution that actually exists in the real world and meets industry standards and user expectations. With slight improvements in security, personalization, and third-party integration, the platform can be a solid starting point for a full-fledged food ordering application.

10. FUTURE SCOPE

The SB Foods platform, which is based on a scalable and modular architecture, is a good starting point for substantial growth and functional upgrading. As technology and user expectations keep advancing, a number of advanced features and integrations can be implemented to upgrade the platform into a mature commercial-grade offering. The following are areas of future development:

1. Real-Time Delivery Tracking

To increase transparency and customer satisfaction, integration with a live delivery tracking system via APIs such as Google Maps, Mapbox, or OpenStreetMap can be implemented. This capability would enable customers to track their food delivery status in real-time — from preparation in the kitchen to delivery at the doorstep — building user confidence and platform reliability.

2. Payment Gateway Integration

As it stands, SB Foods emulates transactions. The next logical step would be the inclusion of secure, real-time payment gateways like Razorpay, Stripe, or PayPal. This would enable customers to make online payments immediately via different modes like UPI, credit/debit cards, and net banking, and the platform would be ready for actual business deployment.

3. Personalized Recommendation System

Utilizing machine learning algorithms, the system might examine customer order history, ratings, and browsing patterns to recommend restaurants or dishes. A recommendation engine would enhance user interaction and possibly drive sales by highlighting relevant choices based on individual tastes and trending popularity.

4. Multi-Language Support

In order to make the platform more inclusive and accessible throughout India's linguistic diversity, multi-language capability can be implemented. Supporting regional languages like Hindi, Tamil, Telugu, Marathi, and so on, SB Foods can address a larger group of users and enhance the overall user experience, particularly in rural and semi-urban regions.

5. Mobile Application Development

In a world dominated by mobile devices, having custom mobile apps for Android and iOS

built using Flutter or React Native will add accessibility and convenience. Mobile apps can offer push notifications, quick navigation, offline support for viewing menus, and location-based services to increase engagement.

6. Advanced Analytics Dashboard

For administrators and restaurant owners, an advanced analytics dashboard can be introduced to give insights into the key performance metrics.

7. Customer Feedback and Review System

An integrated feedback mechanism can enable users to provide ratings, reviews, and issue reports, making service quality and transparency better. Restaurant partners can use this feedback to enhance offerings, whereas administrators can use it to moderate and ensure platform quality.

8. Automated Notification System

An automated order update, delivery reminder, promotional offer, and seasonal discount sending system through email, SMS, or in-app push would enhance user communication. On-time alerts not only foster trust but also engage customers and keep them updated.

11. APPENDIX

A. Source Code

The entire project source code is available on GitHub and is structured according to the MERN Stack (MongoDB, Express.js, React.js, Node.js) architecture, enabling a full-stack JavaScript development environment. This ensures consistency in development, easy maintenance, and high scalability.

- **Frontend (Client-Side)**
  The frontend of the application is built using React.js, implementing component-based design principles and utilizing modern React hooks (e.g., useState, useEffect, useContext) for state management and side effects.
  Key features include:

    - Responsive UI with dynamic rendering of content

    - Separate views for customer, admin, and restaurant roles

    - Integrated search, filtering, and menu browsing functionalities

    - Cart and checkout handling with form validations

- **Backend (Server-Side)**
  The backend is developed using Node.js and Express.js, forming a RESTful API layer that communicates between the client and the database.
  Key backend functionalities:

    - Secure login/signup and role-based authentication (JWT)

    - CRUD operations for restaurants, users, menus, and orders

    - Middleware for route protection and error handling

    - API routes structured using MVC principles for clarity and reusability

- **Database (MongoDB + Mongoose)**
  The application uses MongoDB, a NoSQL database known for its flexibility and scalability. Mongoose is used to define schemas and manage relationships between collections.
  Database collections include:

- Users: Stores customer, admin, and restaurant partner data

- Orders: Contains order items, timestamps, status, and customer info

- Restaurants: Includes metadata, menus, ratings, and images

- Menus: Stores item name, price, type (veg/non-veg), availability, etc.

- **Deployment & Environment**

  - The project is suitable for deployment on platforms like Render, Vercel, or Heroku.

  - Environment variables are securely managed through .env files for sensitive information like database URIs and JWT secrets.