Course Title: Software Project III
Course Code: CSE316

Submitted to:

Mr. Md. Mahbubur Rahman

Lecturer, Department of CSE, DIU

Submitted by:

Name: Redita Islam Katha
ID: 221-15-5328
Section:61_R

Department of CSE,
Daffodil International University.

Submission date:24/05/2024

1.Overview

Mathbazar stands as a premier food courier service, distinguished by its commitment to punctual delivery of an extensive array of culinary offerings. As a trusted cornerstone in the transportation and delivery sector, Mathbazar transcends mere logistical transactions. Our platform boasts a comprehensive suite of services, spanning local deliveries to international shipments, ensuring unparalleled accessibility to gastronomic delights. With an unwavering dedication to reliability, efficiency, and client contentment, Mathbazar epitomizes excellence in the realm of food delivery services.

key features:

Punctual Delivery Commitment:_Mathbazar's unwavering dedication to delivering orders on time, ensuring customer satisfaction and loyalty.

Extensive Culinary Offerings: The platform offers a wide variety of culinary options, ranging from local favorites to international delicacies, catering to diverse tastes and preferences.

Trusted Reputation: Mathbazar is recognized as a trusted cornerstone in the transportation and delivery sector, known for reliability, efficiency, and quality service.

Comprehensive Service Suite: Mathbazar offers a comprehensive suite of services, including local deliveries and international shipments, providing customers with convenient access to gastronomic delights worldwide.

Client-Centric Approach: The company prioritizes client contentment, actively seeking feedback and implementing measures to address customer needs and preferences.

Efficiency and Optimization: Mathbazar optimizes its operations to maximize efficiency in the delivery process, utilizing advanced logistics technology and strategic planning.

Quality Control Standards: Stringent quality control measures ensure the freshness, safety, and integrity of food items throughout the delivery process, maintaining high standards of quality.

Accessible Platform: Mathbazar's user-friendly platform makes it easy for customers to place orders, track deliveries, and provide feedback, enhancing the overall customer experience.

Global Accessibility: Mathbazar facilitates international shipments, enabling customers to enjoy gastronomic delights from around the world, expanding its reach and accessibility.

Continuous Improvement: The company is committed to continuous improvement and innovation, seeking ways to enhance services, optimize efficiency, and exceed customer expectations.

By emphasizing these key features in the project report, stakeholders can gain a comprehensive understanding of Mathbazar's value proposition, operational strengths, and commitment to excellence in the food delivery services industry.

Selection:

1. Market Demand: There is a growing demand for food delivery services, driven by changing consumer lifestyles, increased urbanization, and the convenience of ordering food online. Mathbazar capitalizes on this trend by offering a comprehensive platform that meets the diverse culinary needs of customers.
2. Unique Value Proposition: Mathbazar distinguishes itself from competitors by emphasizing punctual delivery, a wide array of culinary offerings, and a commitment to customer satisfaction. This unique value proposition positions Mathbazar as a premier choice in the food delivery market.
3. Business Potential: The food delivery market presents significant growth opportunities, especially with the expansion of e-commerce and the increasing adoption of online ordering platforms. Mathbazar's comprehensive suite of services, including local and international deliveries, ensures scalability and long-term viability.
4. Technological Advancements: Advances in logistics technology, such as GPS tracking, route optimization algorithms, and mobile app development, enable Mathbazar to streamline operations, enhance efficiency, and improve the overall customer experience.
5. Exceptional Service Delivery: Mathbazar goes beyond traditional food delivery services by prioritizing reliability, efficiency, and customer contentment. This exceptional service delivery sets Mathbazar apart and fosters customer loyalty and repeat business.

As for the exceptions being made in this project, it's essential to clarify any departures from the norm or any innovative approaches being employed. Here are a few potential exceptions that Mathbazar might be making:

1. International Shipments: While many food delivery services focus primarily on local deliveries, Mathbazar's inclusion of international shipments represents an exception. This expansion into global markets requires additional logistical considerations and regulatory compliance.
2. Emphasis on Punctuality: While punctual delivery is a crucial aspect of any delivery service, Mathbazar's exceptional commitment to punctuality might represent an exception in the industry. This dedication requires rigorous planning, real-time monitoring, and effective communication with customers.
3. Quality Assurance Standards: Mathbazar's stringent quality control measures to ensure the freshness and safety of food items may exceed industry standards. This exception reflects Mathbazar's commitment to delivering high-quality culinary experiences to customers.
4. By clearly articulating why Mathbazar is chosen as the project and identifying any exceptions or innovative approaches, stakeholders can better understand the strategic rationale behind the initiative and its potential for success in the food delivery market.

Reference:

The two best projects are chaldal and food panda which have a great hip in market after the corona pandemic. When people can't go outside because of the pandemic. Those services make life easy for the people. They provide great services to others and create great publicity to the country. The basic type of those project given below,

1. Foodpanda:

- Type: Foodpanda is primarily known as an online food delivery platform.
- Services: It connects users with local restaurants and allows them to order food online for delivery or takeout.
- Business Model: Foodpanda typically partners with various restaurants, offering their menus on the platform. Users can browse through the available options, place orders, and have food delivered to their doorstep.
- Geographical Presence: Foodpanda operates in numerous countries globally, providing a convenient solution for those looking to order food online.

2. Chaldal:

- **Type**: Chaldal is an online grocery and household items delivery platform.
- **Services**: Chaldal enables users to order groceries, fresh produce, household essentials, and more through their online platform. They deliver these items directly to the customer's doorstep.

- **Business Model**: Chaldal often sources products directly from suppliers and has its inventory. Users can browse through a wide range of products, add them to their cart, and schedule a delivery time.
- **Geographical Presence**: Chaldal primarily operates in Bangladesh, serving customers in cities like Dhaka. It focuses on providing a convenient way for users to shop for groceries without physically visiting a store.

Both Foodpanda and Chaldal represent the growing trend of online platforms catering to the needs of consumers by offering convenient and efficient ways to access food and groceries. These platforms leverage technology to streamline the ordering process, provide a variety of choices, and improve overall customer convenience.

Keep in mind that the details mentioned here are based on information available up to January 2022, and there may have been changes or expansions in the services or geographical reach of these platforms since then. If you're looking for the most current and accurate information, it's advisable to check the respective official websites or contact the companies directly.

Project Report: Django Web Application with Bootstrap 5 and SQLite3

Research on Scalable Web Application Architectures:

Introduction: Scalability is a critical consideration for any web application, particularly for services aiming to cater to a large user base with varying demands. Mathbazar, as a premier food courier service, requires a robust architecture capable of handling its expected user base, traffic patterns, data volume, and future growth projections. This research aims to analyze different scalable web application architectures, including microservices, monolithic, serverless, and others, to determine the most suitable architecture for Mathbazar.

1.Monolithic Architecture:

Monolithic architecture is a traditional approach where all components of the application are tightly integrated into a single codebase.

While relatively simple to develop and deploy initially, monolithic architectures may face challenges in scalability as the application grows.

Scaling a monolithic application typically involves replicating the entire application stack, which can be inefficient and costly.

Maintenance and updates to monolithic applications can become complex and risky as the codebase grows larger.

2.Microservices Architecture:

Microservices break down the application into smaller, independently deployable services, each responsible for specific business functions.

This architecture promotes scalability as each service can be scaled independently based on demand.

Microservices are well-suited for handling complex applications and accommodating future growth.

However, implementing microservices architecture requires careful design and management of inter-service communication, data consistency, and deployment processes.

- 3.Serverless Architecture:
- Serverless architectures abstract the infrastructure management away from developers, allowing them to focus on writing code.
- They offer automatic scaling based on demand, which is beneficial for handling variable traffic patterns and accommodating growth.
- Serverless can be cost-effective, as you only pay for the resources used during execution.

- However, serverless architectures might introduce limitations in terms of execution time, resource constraints, and integration with certain third-party services.

4.Event-Driven Architecture:

- Event-driven architecture focuses on asynchronous communication between services through events.
- This architecture promotes loose coupling between services, enabling better scalability and flexibility.
- Event-driven architectures are well-suited for handling real-time data processing and responding to dynamic events.
- However, designing and managing event-driven systems require careful consideration of event sourcing, event routing, and fault tolerance.

In conclusion, for Mathbazar, a microservices architecture would be well-suited to meet scalability requirements. It allows for independent scaling of services, handles variable traffic patterns efficiently, accommodates future growth projections, and provides flexibility in managing the application's complexity. Additionally, leveraging serverless components within the microservices architecture for specific functionalities could further enhance scalability and cost-effectiveness

Scalability requirements:

Based on the research conducted on scalable web application architectures, the scalability requirements for Mathbazar, a premier food courier service, can be outlined as follows:

1. Expected User Base:
    - Mathbazar anticipates serving a large user base consisting of individual consumers, businesses, and potentially international customers.
    - The architecture should be capable of handling a significant influx of users without compromising performance or availability.
2. Traffic Patterns:
    - Mathbazar's platform is expected to experience variable traffic patterns, with peak hours during lunch and dinner times, especially on weekends.

- The architecture should be able to handle sudden spikes in traffic without downtime or performance degradation, ensuring smooth operations during peak periods.

3. Data Volume:
    - Mathbazar will manage a substantial volume of data related to user accounts, orders, menus, payment information, delivery tracking, etc.
    - The architecture should efficiently store, retrieve, and process large volumes of data, ensuring scalability as the data volume increases over time.
4. Future Growth Projections:
    - Mathbazar aims to expand its services locally and internationally, indicating a need for scalability to accommodate future growth.
    - The architecture should be flexible and adaptable, capable of scaling horizontally and vertically to meet the demands of a growing user base and expanding service offerings.
5. Reliability and Availability:
    - As a trusted cornerstone in the transportation and delivery sector, Mathbazar must prioritize reliability and availability.
    - The architecture should be resilient to failures, ensuring uninterrupted service even in the event of hardware or software failures.
6. Efficiency:
    - Mathbazar emphasizes efficiency in its operations, including order processing, delivery routing, and resource utilization.
    - The architecture should be optimized for efficiency, minimizing latency and maximizing resource utilization to deliver a seamless user experience.
7. Security:
    - Mathbazar handles sensitive user data, including payment information, personal details, and order history.
    - The architecture should implement robust security measures to protect against data breaches, unauthorized access, and other security threats.
8. Scalability Testing:
    - Regular scalability testing should be conducted to assess the performance of the architecture under various load conditions.
    - The architecture should be able to scale both horizontally (adding more instances of services) and vertically (increasing resources of existing instances) based on the results of scalability testing.

In summary, the scalability requirements for Mathbazar necessitate an architecture that can efficiently handle a large and variable user base, accommodate growing data volume, ensure reliability and availability, optimize efficiency, prioritize security, and undergo regular scalability testing to maintain optimal performance. A microservices architecture, potentially augmented with serverless components, appears to align well with these requirements, offering flexibility, scalability, and resilience to support Mathbazar's objectives effectively.

Architecture Design :

1. Overview:

Mathbazar, a premier food courier service, requires a scalable architecture to accommodate its expected user base, traffic patterns, data volume, and future growth projections. Leveraging a microservices architecture is the most suitable choice to address these scalability requirements while ensuring flexibility, agility, and resilience.

2. Architecture Components:

1. User Management Service:
Responsibilities:

- User registration, authentication, and authorization.
- Profile management, including user preferences and settings.
- Password management and security.

Interaction:

- Communicates with the Authentication Gateway for user authentication and authorization.
- Integrates with other services to retrieve user information and manage user-related operations.

2. Order Management Service:

Responsibilities:

- Handling the lifecycle of orders, from creation to delivery.
- Managing order details, including items, quantities, prices, and delivery addresses.
- Facilitating order tracking, status updates, and notifications.
- Interaction:
- Communicates with the User Management Service to retrieve user information for order processing.
- Integrates with the Menu Management Service to retrieve menu items and prices for order creation.
- Interacts with the Delivery Tracking Service to update order status and provide real-time delivery updates.

3. Menu Management Service:

Responsibilities:

- Managing restaurant menus, including items, categories, descriptions, and prices.
- Providing endpoints for menu retrieval, search, filtering, and modification.
- Interaction:
- Interacts with the Order Management Service to provide menu information for order creation.
- Communicates with the Authentication Gateway for authentication and authorization of menu management operations.

4. Payment Service:

Responsibilities:

- Handling payment processing, including payment gateway integration and transaction management.
- Managing payment methods, billing information, and order payment status.
- Interaction:
- Communicates with the Order Management Service to process payments for completed orders.
- Interacts with external payment gateways for payment authorization and settlement.

5. Delivery Tracking Service:

Responsibilities:

- Tracking the real-time location of delivery personnel and orders in transit.
- Providing delivery status updates to users, including estimated arrival times and route information.

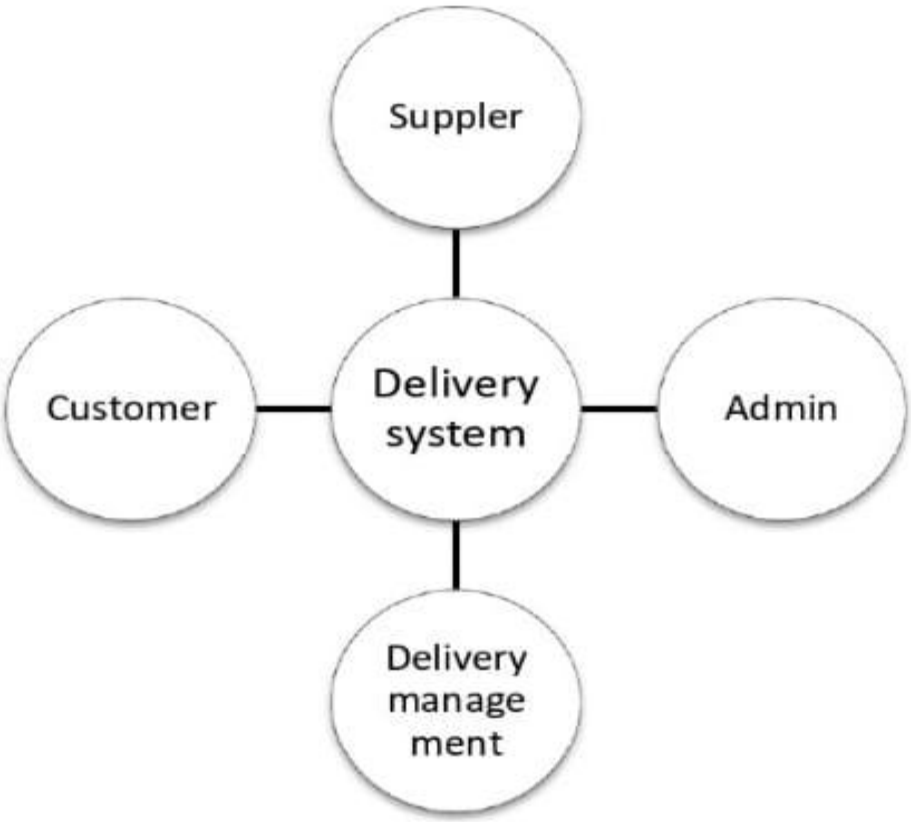
- Interaction:
- Communicates with the Order Management Service to update order status and delivery information.
- Interacts with external mapping services to optimize delivery routes and schedules.

6. Notification Service:

Responsibilities:
Sending notifications to users, restaurants, and delivery personnel regarding order updates, promotions, and events. Supporting various notification channels such as email, SMS, push notifications, and in-app messages.

Interaction:
Communicates with the Order Management Service to trigger notifications based on order status changes.
Interacts with external messaging services or APIs to deliver notifications through different channels.
7. Authentication Gateway:

Responsibilities:
Acting as a centralized authentication and authorization gateway for all microservices.
Providing authentication endpoints for user login, token generation, and validation.
Interaction:
Authenticates incoming requests from client applications and generates authentication tokens for authorized users.
Validates authentication tokens for protected resources before allowing access to microservices.



Interaction between Components:

Components interact with each other through lightweight protocols such as HTTP/REST or messaging queues.
Microservices communicate asynchronously, allowing for loose coupling and improved scalability.
Authentication and authorization are enforced by the Authentication Gateway, ensuring secure access to protected resources.
The API Gateway acts as a facade for client applications, abstracting the complexities of the underlying microservices architecture and providing a unified interface for interaction.
By defining clear responsibilities for each component/module and outlining their interactions, Mathbazar establishes a well-structured and scalable architecture that efficiently manages its business operations while delivering a seamless user experience.

Key Aspects of the Architecture:

- NoSQL databases (e.g., MongoDB, Cassandra) are utilized for flexible data models such as menu items and delivery tracking.
- Data storage solutions are chosen based on the specific requirements of each microservice, balancing factors like data consistency, scalability, and performance.
-

4. Communication between Components:

- Microservices communicate with each other through lightweight protocols such as HTTP/REST or messaging queues.1. Load Balancing: Load balancers distribute incoming traffic across multiple instances of microservices to ensure optimal resource utilization and prevent overloading of any single instance. Load balancing algorithms such as round-robin, least connections, or weighted round-robin are employed to evenly distribute traffic. The load balancer continuously monitors the health and availability of microservice instances and routes traffic accordingly. Load balancing is implemented at both the API Gateway level and individual microservice level to handle varying load patterns.

2. Caching Mechanisms:

- Caching is used to store frequently accessed data to improve response times and reduce the load on backend services.
- Each microservice may implement its caching layer using technologies like Redis or Memcached.
- Cached data may include user session information, frequently accessed menu items, or static content.
- Cache invalidation strategies, such as time-based expiration or event-based invalidation, are employed to ensure data consistency and freshness.

3. Data Storage:

- Data storage is distributed across multiple databases, each managed by the respective microservice.
- Relational databases (e.g., PostgreSQL, MySQL) are used for transactional data such as user profiles, orders, and payments.
-

- Asynchronous communication patterns like publish-subscribe or event-driven messaging are employed to decouple services and improve scalability.
- Service discovery mechanisms (e.g., service registries like Consul or Kubernetes Service Discovery) facilitate dynamic service discovery and communication.
- API contracts and versioning are enforced to ensure compatibility between microservices and prevent breaking changes during updates.

5. Service Mesh:

- A service mesh, such as Istio or Linkerd, may be employed to manage communication between microservices, handle load balancing, and enforce security policies.
- Service mesh provides features like circuit breaking, retries, timeouts, and distributed tracing to enhance reliability and observability.
- Sidecar proxies are deployed alongside microservice instances to intercept and manage traffic between services transparently.

6. Asynchronous Processing:

Asynchronous processing mechanisms, such as message queues (e.g., RabbitMQ, Kafka), are used for offloading time-consuming tasks and improving system responsiveness.
Long-running operations like order processing or image processing for menu items are delegated to background worker processes or dedicated microservices.
Event-driven architectures enable microservices to react to asynchronous events and trigger subsequent actions, enhancing scalability and responsiveness.
By incorporating these aspects into the microservices architecture, Mathbazar ensures scalability, reliability, and performance while efficiently managing its business operations. Load balancing, caching mechanisms, distributed data storage, and asynchronous communication patterns contribute to the overall scalability and resilience of the system, enabling Mathbazar to deliver a seamless and responsive user experience.

Technologies Used

For Mathbazar, the project incorporates a variety of robust technologies:
- Backend: Django
- Frontend: HTML, CSS, Bootstrap 5
- Database: PostgreSQL

Backend (Django) Mathbazar utilizes Django for the backend architecture, which is structured around key components such as models, views, and templates. Django's models are used to define the essential data structures, handling the intricacies of order management, user profiles, and transaction logs. Views facilitate the application logic, directing data between models and templates. The templates then render the necessary HTML content dynamically, ensuring that server-side processing is efficiently handled.
Frontend (Bootstrap 5) The frontend of Mathbazar uses Bootstrap 5 to create a responsive and accessible interface. Bootstrap's grid system and components like modals, dropdowns, and buttons are utilized to ensure the layout is intuitive and engaging. The styling is customized to align with Mathbazar's branding, providing a seamless visual experience for users across different devices.

Database (PostgreSQL) Replacing SQLite3, PostgreSQL serves as Mathbazar's database solution to better handle higher volumes of transactions and complex queries efficiently. The database schema is designed to support robust relationships, such as between users, orders, and shipment details, optimizing data integrity and retrieval speed.

Features and Functionality

Core Features: Core functionalities in Mathbazar include user authentication, real-time order tracking, and comprehensive data CRUD (Create, Read, Update, Delete) operations. These features ensure a secure and user-friendly environment for managing courier orders.

Additional Features: Additional features include automated notifications, detailed analytics dashboards for users, and an interactive map interface for tracking deliveries. These enhancements aim to improve user engagement and provide added value

User Interface and Design :

Design Principles: The design principles focus on simplicity, consistency, and responsiveness. The user interface is crafted to be intuitive and efficient, minimizing user edicated worker services improves system responsiveness and resource utilization. By processing background tasks asynchronously, Mathbazar ensures that the main application thread remains available to handle user requests promptly.

By implementing these solutions, Mathbazar addresses key challenges such as scalability, real-time tracking, user authentication, security, and performance optimization. These measures ensure the project's success by providing a scalable, secure, and high-performing platform that meets the needs of its users while delivering a seamless and responsive user experience.
effort and maximizing usability.

Responsiveness: Using Bootstrap 5, Mathbazar's design is fully responsive, adapting seamlessly to different screen sizes and orientations. This ensures a consistent experience whether accessed from a mobile device, tablet, or desktop.

Challenges and Solutions: To overcome the challenges and ensure the success of the Mathbazar project, several solutions have been implemented:

Scalability: Microservices Architecture: Breaking down the application into smaller, independent services allows for easier management and scalability. Each service can be deployed and scaled independently, accommodating varying levels of demand. Containerization and Orchestration: Utilizing Docker for containerization and Kubernetes for orchestration streamlines the deployment and management of microservices. This ensures consistent operation across different environments and facilitates rapid scaling as needed.

Real-Time Tracking: WebSocket Communication: WebSocket communication enables real-time updates on order status changes without the overhead of continuous polling. This ensures that users receive instant notifications and accurate tracking information. Geospatial Databases: Implementing geospatial databases like PostGIS enhances the accuracy and efficiency of order tracking. These databases enable the storage and retrieval of geographic data, facilitating real-time location updates and route optimization.

3.Performance Optimization: Caching Mechanisms: Leveraging caching mechanisms like Redis reduces the load on the database and improves response times by storing frequently accessed data in memory. This ensures faster access to data and enhances overall system performance.
Asynchronous Processing: Offloading non-essential tasks to asynchronous processes or dedicated worker services improves system responsiveness and resource utilization. By processing background tasks asynchronously, Mathbazar ensures that the main

application thread remains available to handle user requests promptly. By implementing these solutions, Mathbazar addresses key challenges such as scalability, real-time tracking, user authentication, security, and performance optimization. These measures ensure the project's success by providing a scalable, secure, and high-performing platform that meets the needs of its users while delivering a seamless and responsive user experience.

Testing

Unit Testing: The backend components are rigorously tested using Django's built-in testing framework, which ensures reliability and functionality through comprehensive unit tests.

User Acceptance Testing: User acceptance testing was performed with a group of selected active users to ensure the application meets real-world usage requirements and to gather feedback for further refinement.

Future Enhancements:

In the future, Mathbazar plans to enhance its platform by focusing on user experience, expanding features, and integrating emerging technologies. This includes personalized recommendations, advanced order management, integration with blockchain for transparency, and AI-driven predictive analytics. Additionally, they aim to expand delivery options, improve analytics and reporting, and engage in community initiatives for sustainability. These enhancements will help Mathbazar stay competitive and meet evolving user expectations in the courier and food delivery industry.