**Une image contenant blanc, conception, noir et blanc, outil

Description générée automatiquementUne image contenant texte, Police, conception, guide

Description générée automatiquement *BytebiteDeliverySystem***

**Zyadfri, jaafar yefou and oumaima Deliverable 4**

Une image contenant dessin humoristique, texte, clipart, habits

Description générée automatiquement

***Team Members***

Une image contenant texte, Visage humain, personne, capture d’écran

Description générée automatiquement

The ByteBite Delivery Engine is a comprehensive web-based food delivery system designed to provide users with an efficient and user-friendly experience. This project encompasses multiple pages, each serving a specific purpose in the food ordering process. From user authentication to restaurant selection, menu browsing, and order management, the system aims to streamline the online food ordering process.

**1. Landing Page (homePage.php):**

The project begins with an inviting and aesthetically pleasing landing page. The landing page is meticulously designed to provide an intuitive interface for users. It features a login form for existing users and a registration option for new customers. The page includes error handling for incorrect login attempts, ensuring a seamless and frustration-free user experience.

**2. User Registration Page (sign.php)**

For new users, the registration page offers a straightforward form where they can create an account. The registration process collects essential user information, including email and password, ensuring a secure and personalized experience. The page incorporates validation checks to ensure the accuracy and completeness of user-provided data.

**3. Login Page (login.php):**

The login page serves as the gateway for existing users to access the ByteBite Delivery Engine. Upon successful login, users are redirected to the restaurant selection page. The login mechanism involves secure password validation, safeguarding user accounts from unauthorized access.

**4. Restaurant Selection Page (UserDashboard.php):**

After authentication, users are presented with a visually appealing restaurant selection page. This page dynamically fetches restaurant data from the database, showcasing a variety of options for users to choose from. Each restaurant is accompanied by enticing imagery and a brief description, creating an immersive experience for users.

**5. Menu Page (menu.php):**

The menu page provides an extensive showcase of the selected restaurant's offerings. Featuring a grid layout with vibrant images and detailed meal descriptions, users can easily browse through the available menu items. The page includes an "Add to Cart" button for each meal, enabling users to effortlessly assemble their orders.

**6. Shopping Cart Page (ViewCart.php):**

The shopping cart page is a pivotal element in the ordering process. It dynamically displays the selected meals, allowing users to review and modify their orders. The page features a responsive design, ensuring an optimal viewing experience across devices. Users can remove items from the cart, providing flexibility and control over their orders.

**7. Payment Page (payment.php):**

The payment page facilitates a secure and efficient transaction process. Users can enter their payment details with confidence, as the page adheres to industry-standard security practices. Integration with secure payment gateways ensures the protection of sensitive financial information.

**8. Order Confirmation Page (order\_confirmation.php):**

Upon successful payment, users are redirected to the order confirmation page. This page serves as a digital receipt, summarizing the order details and confirming the successful completion of the transaction. Users can view their order history and track the status of their deliveries.

.

**10. Database Interaction (conn.php):**

The project's backend seamlessly interacts with a relational database. The database stores user information, restaurant data, and order details, ensuring data integrity and enabling efficient retrieval of information.

***Project Achievements:***

**User-Centric Design:** The project prioritizes an exceptional user experience, featuring an intuitive interface and visually appealing design elements.

**Security Measures:** Robust security measures, including password encryption and secure payment gateways, safeguard user information.

**Responsive Design:** The project employs responsive design principles, ensuring a consistent and optimal user experience across various devices.

**Database Integration:** The seamless integration with a relational database enhances data management and retrieval capabilities.

**Order Tracking:** Users can conveniently track the status of their orders, providing transparency and improving the overall user experience.

**Error Handling:** The system incorporates comprehensive error handling mechanisms, minimizing user frustration and enhancing system reliability.

Flexibility and Control Users have the flexibility to customize their orders and remove items from the shopping cart, providing: a high level of control over their transactions.

**Project Learning Experience:**

**1. Technology Mastery:**

**PHP Development:** This project served as a practical introduction to PHP, a server-side scripting language. Through implementing dynamic web pages, handling user authentication, and interacting with databases, we gained proficiency in PHP.

**JavaScript Integration:** The incorporation of JavaScript enhanced the project's interactivity, contributing to a more engaging user experience. This included client-side form validation and dynamic content updates without requiring page reloads.

**2. Database Management:**

**MySQL Database:** Our project involved the design and utilization of a MySQL relational database. We learned to create normalized tables, establish relationships, and execute complex queries. This experience deepened our understanding of data management in web development.

**3. Collaborative Development:**

**Group Dynamics:** Collaborative development was a key aspect of this project. Working as a group, we practiced effective communication, task distribution, and version control using platforms like Git. This **experience honed our collaborative skills and prepared us for future teamwork scenarios.**

**Code Integration:** Coordinating the integration of individual code segments into a cohesive application provided insights into the importance of code consistency and modularity.

**4. User-Centric Design:**

**UI/UX Principles:** The project emphasized the significance of user-centric design. We learned to prioritize user experience, incorporating intuitive navigation, responsive design, and visually appealing elements to enhance overall satisfaction.

**Error Handling Strategies:** Implementing comprehensive error handling mechanisms improved the robustness of the application. We learned to anticipate user interactions and provide informative error messages for a smoother user experience.

**5. Security Practices:**

**Password Encryption:** Security was paramount in our project. We implemented password encryption techniques to protect user credentials. This hands-on experience heightened our awareness of security considerations in web development.

**Secure Transactions**: The integration of secure payment gateways taught us the intricacies of handling sensitive financial information

**6. Flexibility:**

**User Customization:** The project's shopping cart functionality allowed users to customize their orders. Learning to implement dynamic and flexible features empowered us to provide users with a high degree of control over their transactions.

**Order Modification:** Enabling users to remove items from the shopping cart showcased our ability to implement dynamic updates to the user interface, contributing to a seamless and responsive design.

**7. Project Management:**

**Timeline Adherence:** Managing project timelines and milestones was a valuable learning experience. We developed skills in project planning, task prioritization, and meeting deadlines, ensuring the successful completion of the project within the stipulated timeframe.

**8. Order Tracking and Transparency:**

**Real-Time Updates:** Implementing order tracking functionality provided insights into real-time updates and data synchronization. This knowledge is transferable to projects requiring real-time information display and user feedback.

**Collaborative Group Dynamics:**

In our journey through the ByteBite Delivery Engine project, we embraced a collaborative ethos, where responsibilities were shared dynamically, fostering a unified team approach. Each member of our group played an integral role in shaping and implementing the project, creating an environment of shared ownership and collective responsibility.

***1. Collective Ideation and Planning:***

**Brainstorming Sessions:** From the inception of the project, we engaged in extensive brainstorming sessions. As a team, we collectively conceptualized the application's features, defining the scope and envisioning the user experience. Every team member contributed ideas, building a foundation that encapsulated diverse perspectives.

**Project Planning:** Collaboratively, we delineated project milestones, allocated tasks, and established timelines. This collaborative planning phase allowed us to leverage the strengths of each team member, ensuring an efficient distribution of responsibilities.

***2. Unified Decision-Making:***

**Consensus Building:** Every major decision, from the selection of technologies to the design of the user interface, was made through consensus. The team engaged in open discussions, weighing the merits of various options and arriving at decisions that reflected the collective vision.

***3. Shared Learning Curve:***

**Technology Exploration:** As we delved into new technologies such as PHP and JavaScript, the learning curve was a shared experience. we actively participated in learning sessions, knowledge-sharing, and collaborative problem-solving. This shared learning environment facilitated a collective mastery of the technologies employed.

**Cross-Functional Understanding we**  developed a cross-functional understanding of various aspects of the project. Whether it was database design, front-end development, or security considerations, we collectively ensured that everyone had exposure to and comprehension of each domain.

***4. Task Execution and Support:***

**Dynamic Task Allocation:** Tasks were not rigidly assigned; instead, we embraced a dynamic approach where we naturally gravitated toward tasks aligning with our strengths and interests. This flexibility empowered individuals to contribute where we felt most impactful.

**Mutual Support:** Throughout the development process, we actively supported one another. Code reviews, debugging sessions, and collaborative problem-solving were shared responsibilities, reinforcing a sense of collective achievement.

**Enhancement: Introduction of DeliveryPersonInfos Table**

As a part of our continuous efforts to enhance the functionality and user experience of our food delivery application, we introduced a new database table named DeliveryPersonInfos. This table is designed to store essential information about delivery persons, enabling them to log in and efficiently manage the delivery process.

**Table Structure:**

The DeliveryPersonInfos table includes the following fields:

**Pid**: A unique identifier for each delivery person.

**PName**: The name of the delivery person.

**Password**: A secure field storing the password associated with the delivery person's account.

**Purpose:**

**Login Functionality:** The primary purpose of introducing the DeliveryPersonInfos table is to provide a secure and personalized login mechanism for delivery persons. With their unique identifier (Pid) and password, delivery persons can log in to the application, ensuring accountability and security.

Efficient Delivery Management: By having dedicated accounts for delivery persons, we aim to streamline and enhance the efficiency of the delivery process. Each delivery person can access a personalized dashboard, view assigned orders, and update the delivery status in real-time.

**Implementation:**

The implementation involves integrating the new table into our existing database structure. The login mechanism is updated to validate credentials against the DeliveryPersonInfos table, ensuring a seamless and secure login experience.

**Transaction and Procedure:**

The transaction **AddOrderAtRestaurant**, within a restaurant delivery system. It initiates a transaction with a REPEATABLE READ isolation level, ensuring consistent data reads. The procedure checks for the existence of the specified customer, restaurant, and valid meal. If all conditions are met, it inserts order details into the 'Orders' table and associates the meal with the order in the 'Contains' table. In case of any discrepancies, the transaction is rolled back, and an error is signaled. Overall, the script establishes a robust mechanism for adding orders, maintaining data integrity through careful validation and transaction handling.

**ByteBite Delivery Engine A Comprehensive Restaurant Delivery System : Overview**

The project, collectively developed by Zyad Fri, Jaafar Yeffou, and Oumaima Kouhail, represents a dynamic restaurant delivery system. The project unfolds with a visually appealing homepage, "ByteBite," featuring a user-friendly interface. The homepage showcases the brand, navigation options for home, about us, and contact sections, and facilitates user engagement through sign-up and login buttons.

Moving to the sign-up and login pages, users seamlessly register and access the system, ensuring secure authentication. The project ingeniously employs PHP and JavaScript.

As users log in, the system excels in its versatility. A pivotal feature is the restaurant search functionality, where customers can explore and select restaurants effortlessly. A dedicated "SearchResults.php" page dynamically displays restaurant options based on user input. A substantial modification introduces a new "DeliveryPersonInfos" table, broadening the application's scope. Notably, the system demonstrates database management proficiency.

In the heart of the project lies the restaurant selection process, leading customers to the "ChooseRestaurant.php" page. The meticulous design ensures a smooth transition from restaurant selection to the respective menu. Here, each meal presents an "Add to Cart" button, seamlessly integrating with the ViewCart page.