**INDEX**

|  |  |  |
| --- | --- | --- |
| **S.NO** | **CONTENT** | **PAGE.NO** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# INTRODUCTION

1. **About Hotel Chellappa**:

Hotel Chellappa is a well-established name in hospitality, known for its authentic cuisine and quality service. With a growing customer base, the hotel is embracing digital transformation to enhance guest convenience.

1. **Need for Online Menu Ordering**:

In the age of digital convenience, customers increasingly prefer quick and contactless ordering options. An online menu ordering system allows guests to browse, customize, and place orders directly from their smartphones or devices.

1. **Objective of the System**:

The primary goal is to simplify and modernize the food ordering process for dinein, takeaway, or room service, reducing wait times and minimizing errors in order-taking.

1. **Features and Benefits**:
   1. User-friendly digital menu interface
   2. Real-time menu updates (availability, pricing, specials)
   3. Easy customization of orders
   4. Integration with kitchen and billing systems
   5. Contactless payment options
   6. Enhanced customer experience and operational efficiency

1. **Target Users**:
   1. Hotel guests (for room service)
   2. Walk-in customers (for dine-in or takeaway)
   3. Hotel staff (for order management)

1. **Why It Matters**:

With evolving customer expectations and the impact of health-conscious practices (especially post-COVID), providing a seamless, digital menu ordering system helps Hotel Chellappa stay competitive and customer-focused.

# ABSTRACT

* **Purpose**:

This system is designed to digitalize the food ordering process at Hotel Chellappa, offering customers a seamless, contactless, and efficient way to place orders.

* **Problem Statement**:

Traditional menu ordering methods are time-consuming, prone to human error, and less adaptable to real-time changes in menu availability and pricing.

* **Proposed Solution**:

An online menu ordering platform that enables customers to view the menu, customize orders, and make payments through a web or mobile interface, integrated with the hotel’s kitchen and billing systems.

* **Key Features**:
  + Dynamic digital menu display o Real-time updates on item availability o Multi-language support for diverse guests o Order tracking and notification system o Secure and flexible payment options o **Technologies Used** (if needed for a technical abstract):

* Frontend: HTML, CSS, JavaScript

* Backend: PHP/ with MySQL database

* Platform: Mobile responsive web app

* **Benefits**:
  + Improves customer satisfaction through convenience and speed o Reduces staff workload and minimizes order errors o Enhances operational efficiency and service quality

* **Conclusion**:

The system provides a modern solution for food ordering at Hotel Chellappa, aligning with the hotel’s commitment to quality and innovation in guest services.

# OBJECTIVE

* **To develop a user-friendly online platform** that allows customers to browse the hotel menu and place orders seamlessly from their own devices.

* **To reduce the dependency on manual order-taking**, thereby minimizing human errors, miscommunication, and delays in service.

* **To provide real-time menu updates**, including item availability, prices, and daily specials, ensuring customers have the latest information when placing orders.

* **To streamline the food ordering process** for dine-in, takeaway, and room service, improving overall operational efficiency.

* **To enable secure and flexible payment options**, including digital wallets, credit/debit cards, and UPI integration.

* **To improve customer satisfaction and experience** through faster service, order accuracy, and modernized interaction.

* **To support multi-device compatibility**, ensuring the system works efficiently across desktops, tablets, and smartphones.

* **To integrate the system with kitchen and billing departments**, allowing automatic order routing and simplified invoice generation.

* **To collect and analyze order data** for generating reports, tracking customer preferences, and improving menu planning.

* **To promote contactless service**, aligning with modern hygiene standards and customer expectations post-COVID.

# EXISTING SYSTEM

* **Manual Order Taking**:

Orders are currently taken by waiters or front desk staff, which is timeconsuming and prone to human errors such as miscommunication or wrong entries.

* **Printed Menus**:

Hotel Chellappa uses physical menu cards, which are costly to update and cannot reflect real-time changes in item availability or pricing.

* **Limited Menu Visibility**:

Guests must be physically present or call the hotel to view the menu, which reduces convenience for room service and takeaway customers.

* **No Real-Time Updates**:

Menu changes, daily specials, or out-of-stock items are not updated instantly, leading to confusion or disappointment at the time of ordering.

* **High Staff Dependency**:

The system relies heavily on staff for taking and delivering orders, increasing workload and reducing efficiency during peak hours.

* **Lack of Order Tracking**:

Once the order is placed, customers have no way to track the status (e.g., being prepared, ready, or delivered), which can affect satisfaction.

* **No Digital Payments Integration**:

Most transactions are cash-based or handled manually, lacking the convenience and security of digital or contactless payments.

* **No Data Collection**:

Customer preferences, popular items, and order history are not tracked or analyzed, missing opportunities for data-driven decisions or marketing.

* **Longer Service Time**:

Due to manual processes, customers often experience longer wait times from order placement to food delivery.

# PROPOSED SYSTEM

* **Digital Menu Platform**:

Introduce an interactive, web-based or mobile-friendly menu that allows customers to view available items with images, descriptions, and prices.

* **Real-Time Menu Management**:

Enable hotel staff to update the menu instantly, including item availability, pricing, and special offers, ensuring customers always see the latest information.

* **Self-Service Ordering**:

Allow customers to place their own orders directly from their devices, reducing reliance on staff and minimizing order errors.

* **Multiple Service Modes**:

Support ordering for dine-in, takeaway, and room service, with customized options based on the customer’s location or preference.

* **Order Tracking and Notifications**:

Provide real-time updates to customers on their order status (e.g., confirmed, in preparation, ready, delivered).

* **Secure Digital Payments**:

Integrate various payment gateways to support UPI, credit/debit cards, net banking, and digital wallets for smooth, secure transactions.

* **Admin Dashboard**:

Equip hotel management with an admin panel for managing menu items, monitoring orders, tracking sales, and generating reports.

* **Data Collection and Analytics**:

Automatically store customer order history and preferences to help with menu planning, marketing strategies, and service improvement.

* **Multi-Language and Accessibility Support**:

Offer menu options in multiple languages to cater to a diverse customer base, including accessible features for better usability.

* **Contactless Experience**:

Promote hygiene and safety by eliminating the need for physical menus or direct interactions during the ordering and payment process.

# FEATURES

* **Interactive Digital Menu**:

Customers can browse a visually rich, categorized menu with images, descriptions, prices, and availability status.

* **Real-Time Order Placement**:

Orders can be placed instantly for dine-in, takeaway, or room service, with realtime confirmation.

* **Customizable Orders**:

Users can select portion sizes, add/remove ingredients, and leave special instructions for their meals.

* **Multi-Platform Access**:

The system is accessible via smartphones, tablets, and desktops, with responsive design for all screen sizes.

* **Live Order Tracking**:

Customers receive updates on their order progress—from placement to preparation to ready/delivery status.

* **Digital Payments Integration**:

Supports multiple payment methods including UPI, credit/debit cards, wallets, and net banking for contactless transactions.

* **Multi-Language Support**:

The interface is available in multiple languages to serve a wider range of local and international guests.

* **Admin Panel for Staff**:

Hotel staff can manage orders, update the menu, set item availability, and generate sales reports from a secure dashboard.

* **Customer Feedback Option**:

Built-in feedback system allows customers to rate their food and service, providing valuable insights to hotel management.

* **Push Notifications and Alerts**:

Notifies users about order status, special offers, and menu changes in real-time.

**Software Specification**

# SOFTWARE SPECIFICATION

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Feature** | **Description** |  |
| 1 | User  Registration &  Login | Secure login for customers and staff with role-based access. |  |
| 2 | Menu  Management | Admin can add, edit, delete menu items, and update availability/prices. |  |
| 3 | Order  Placement | Customers can browse menu, customize, and place orders. |  |
| 4 | Order Tracking | Real-time status updates: confirmed, preparing, ready, or delivered. |  |
| 7 | Admin  Dashboard | Manage orders, users, reports, and view analytics. |  |
| 8 | Multi-Service Support | Supports dine-in, takeaway, and room service ordering modes. |  |

**FLOW DIAGRAM**

**Customer Access**

**↓**

**View Menu**

**↓**

**Select Items**

**↓**

**Add to Cart**

**↓**

**Order Confirmation**

**↓**

**Kitchen Prepares Order**

**↓**

**Serve/Deliver Order**

**ENTITY RELATIONALSHIP DIAGRAM**

1. **Customer**:

* 1. Customer\_ID (Primary Key)
  2. First\_Name
  3. Email
  4. Phone

2. **Menu\_Item**:

1. Menu\_Item\_ID (Primary Key)
2. Name
3. Description
4. Price
5. Category (e.g., appetizers, main course, drinks, desserts)
6. Availability\_Status (e.g., available, out of stock) 3

. **Order**:

1. Order\_ID (Primary Key)
2. Customer\_ID (Foreign Key to Customer)
3. Order\_Date
4. Total\_Price
5. Status (e.g., pending, in-progress, completed, cancelled)
6. Payment\_Status (e.g., paid, pending)

4. **Order\_Item** (this is a junction table to handle many-to-many relationships between orders and menu items):

* 1. Order\_Item\_ID (Primary Key)
  2. Order\_ID (Foreign Key to Order)
  3. Menu\_Item\_ID (Foreign Key to Menu\_Item)
  4. Quantity
  5. Subtotal\_Price (Price \* Quantity) 5. **Payment**:
  6. Order\_ID (Foreign Key to Order)
  7. Payment\_Amount
  8. Payment\_Date
  9. Payment\_Status (e.g., paid, failed, pending) 6Table\_ID (Primary Key)

**Relationships:**

* **Customer ↔ Order**: A customer can place many orders. (1-to-many)
* **Order ↔ Order\_Item**: An order can contain multiple items, and an item can appear in multiple orders. (many-to-many, resolved by the **Order\_Item** table)
* **Menu\_Item ↔ Order\_Item**: A menu item can be included in many orders. (1-tomany)
* **Order ↔ Payment**: An order can have one payment associated with it. (1-to-1)
* **Staff ↔ Order**: Staff may be responsible for fulfilling the order. (1-to-many)
* **Staff ↔ Menu\_Item**: Staff can be responsible for preparing or serving menu items, especially chefs and waiters. (many-to-many)
* **Table ↔ Order**: A table can be assigned to an order (especially for dine-in orders). (1-to-many)

**Explanation of Relationships:**

* **Customer ↔ Order**: One customer can place many orders, hence a one-tomany relationship.
* **Order ↔ Order\_Item**: One order can contain multiple menu items, and each menu item can appear in multiple orders, hence the many-to-many relationship.
* **Menu\_Item ↔ Order\_Item**: One menu item can be part of multiple orders, hence a one-to-many relationship.
* **Order ↔ Payment**: Each order has a payment associated with it, hence a oneto-one relationship.
* **Staff ↔ Order**: Staff are assigned to orders, like a waiter taking the order, so a staff member can be associated with many orders.
* **Staff ↔ Menu\_Item**: Staff can be linked to menu items in terms of preparation or serving (e.g., chefs prepare certain items, waiters serve them).
* **Table ↔ Order**: A table is assigned to an order, usually for dine-in customers. One table can handle many orders.

# ADMIN MODULE

To define what functionalities admins have access to and how they manage core entities in the system like:

* Users (customers and staff)
* Menu management
* Order tracking
* Payments
* Reporting

✅ **Key Admin Roles (can vary by access level):**

1. **Super Admin** – Full system access (manage staff, menus, payments, reports)
2. **Manager** – Manage orders, tables, and limited staff
3. **Chef Admin** – View/manage menu items, mark dishes ready
4. **Billing Admin** – Manage payments, generate invoices **Admin Functional Modules:**

|  |  |
| --- | --- |
| **Module** | **Functions** |
| **User Management** | Add/edit/delete customers |
| **Menu Management** | Create/edit categories, add/edit/remove dishes |
| **Order Management** | View all orders, update statuses |
| **Reporting** | Generate reports for sales, popular items |
| **Table Management** | Manage seating availability for dine-in |
| **Access Control** | Assign roles and access levels to staff/admins |

1. **Real-Time Monitoring:**
   * **Real-Time Order Tracking**: Admins can view real-time status updates of ongoing orders (e.g., “In Progress,” “Out for Delivery,” “Served”) to monitor restaurant operations effectively.
   * **Live Chat Integration**: Enable admins to chat with staff members or customers (especially for real-time issue resolution).

1. **Menu and Category Management:**
   * **Dynamic Pricing**: Allow admins to create promotions or discounts on certain items or categories (e.g., "Happy Hour" discounts).
   * **Customizable Menu Items**: Allow menu items to have customizable options (e.g., different toppings or sides), with the ability to set prices for each variation.
   * **Seasonal Menu Control**: Ability to create and manage seasonal menus (e.g., summer specials, holiday menu).

## STUDENT MODULE

This module defines what a **student user** (or general customer in a student environment, such as a university hostel or canteen) can do in the system. It assumes students are **end-users/customers**, not admins.

# ✅ 1. Student Registration & Login

* Register as a new user with name, email, student ID, phone, and password.
* Secure login with email/username and password.
* Forgot password/reset password functionality.
* Optional OTP or email verification.

# ✅ 2. Profile Management

* View and update personal details (e.g., name, email, phone, hostel room number).
* Change password.
* Upload profile photo (optional).

# ✅ 3. Browse Menu

* View categorized menu (e.g., Breakfast, Lunch, Snacks, Dinner).
* Search items by name or category.
* View item details (ingredients, price, availability, nutritional info).
* Filter by dietary preferences (e.g., veg, non-veg, gluten-free).

# ✅ 4. Place Order

* Add multiple items to cart.
* Select quantity for each item.
* Choose delivery option: o Dine-in
  + Takeaway
  + Room delivery (for hostels)
* Place order with notes (e.g., "no onions").
* View total price including tax and delivery (if any).
* Confirm order and choose payment option.

# ✅ 6. Order Tracking

* Track real-time order status:
  + Order placed o Preparing
  + Out for delivery / Ready for pickup o Delivered / Completed
* Receive notifications (email/SMS/in-app) about order progress.

# ✅ 7. Order History

* View all past orders with:

o Order ID o Items o Amount paid o Date/time o Status

* Reorder from history with a single click.

✅ 9. Offers and Promotions

* View current discounts, offers, or meal combos.
* Apply coupon codes at checkout.

# ✅ 11. Logout & Session Management

* Secure logout.
* Auto-session timeout for inactive users.

**FONT-END TECHNOLOGY**

🎨 **Front-End for Online Hotel Menu Ordering System**

# ✅ 1. Technologies You Can Use

* **HTML5** – Structure of web pages

**CSS3** – Styling (use frameworks like Bootstrap or Tailwind for responsiveness)

* **JavaScript** – Interactivity (vanilla JS or frameworks like React.js, Vue.js)

**Responsive Design** – Use media queries or frameworks for mobile/tablet support

# ✅ 2. Essential UI Pages / Components

|  |  |
| --- | --- |
| **Page** | **Functionality** |
| Home Page | Welcome message, featured menu items, login/signup options |
| Login / Signup Page | Student authentication with input validation |
| Menu Browsing Page | Show menu items, categories, filter/sort, item cards |
| Cart Page | View selected items, update quantities, remove items |
| Checkout Page | Confirm address/delivery type, choose payment method |
| Order Tracking Page | Show live order status (e.g., preparing, out for delivery) |
| Order History | List past orders, allow reorder, show invoice links |
| Review & Feedback | Allow users to leave ratings and reviews |
| Profile Settings | Update user info, change password |
| Offers / Promotions Page | Show current deals or student combos |

# ✅ 3. UI/UX Design Tips

* Use **consistent color schemes** (e.g., warm tones for food-related apps).
* Ensure **easy navigation** (hamburger menu on mobile, tabs, breadcrumbs).
* Keep **important buttons accessible** (e.g., “Add to Cart,” “Checkout”).
* **Highlight offers** and recommended dishes with labels or badges.
* Use **icons and images** for visual appeal (Font Awesome or SVGs).
* Implement **loading spinners** for smoother transitions.

# ✅ 4. Front-End Features to Include

* **Live Search**: Search bar with suggestions while typing

**Sorting & Filtering**: Price range, dietary filters (veg/non-veg)

* **Item Details Modal**: Click on a menu item to see more details (popup/modal)
* **Real-Time Cart Update**: Automatically update totals without page reload
* **Toast Notifications**: Show messages like “Item added to cart” or “Payment failed”
* **Theme Support**: Optional light/dark mode for better UX

# ✅ 5. Libraries & Frameworks (Optional)

* **React.js** – For SPA (single-page applications)
* **Bootstrap/Tailwind CSS** – For responsive layout
* **jQuery** – For simpler DOM manipulation (if not using React/Vue)
* **SweetAlert2** – For alert modals (success/error)
* **Chart.js / ApexCharts** – For admin-side analytics charts

# ✅ 6. Validation & Security (Client-Side)

* Form validation (e.g., required fields, valid email, password strength)
* Disable right-click or console access (light protection)
* Limit attempts on login fields (in addition to backend security)

# ✅ 7. Testing & Debugging

* Use browser DevTools to test responsive design (mobile/tablet view)
* Check for broken links or missing image files
* Test performance using Lighthouse (Chrome DevTools)

# ✅ 8. Good Practices

* Modularize your code: use components (especially with React)
* Use a **router** (e.g., React Router) for managing views/pages
* Keep CSS organized and scoped (or use CSS modules)
* Separate logic from layout (JS in separate files or folders)

**Example Folder Structure (React-based):**

bash CopyEdit

/frontend

/public

/src

/components

* Header.js
* MenuCard.js
* OrderSummary.js

/pages

* Home.js
* Menu.js
* Checkout.js

/assets

* images/
* styles/ App.js index.js