

RESTAURANT MANAGEMENT SYSTEM

Project Timeline (12 Weeks)

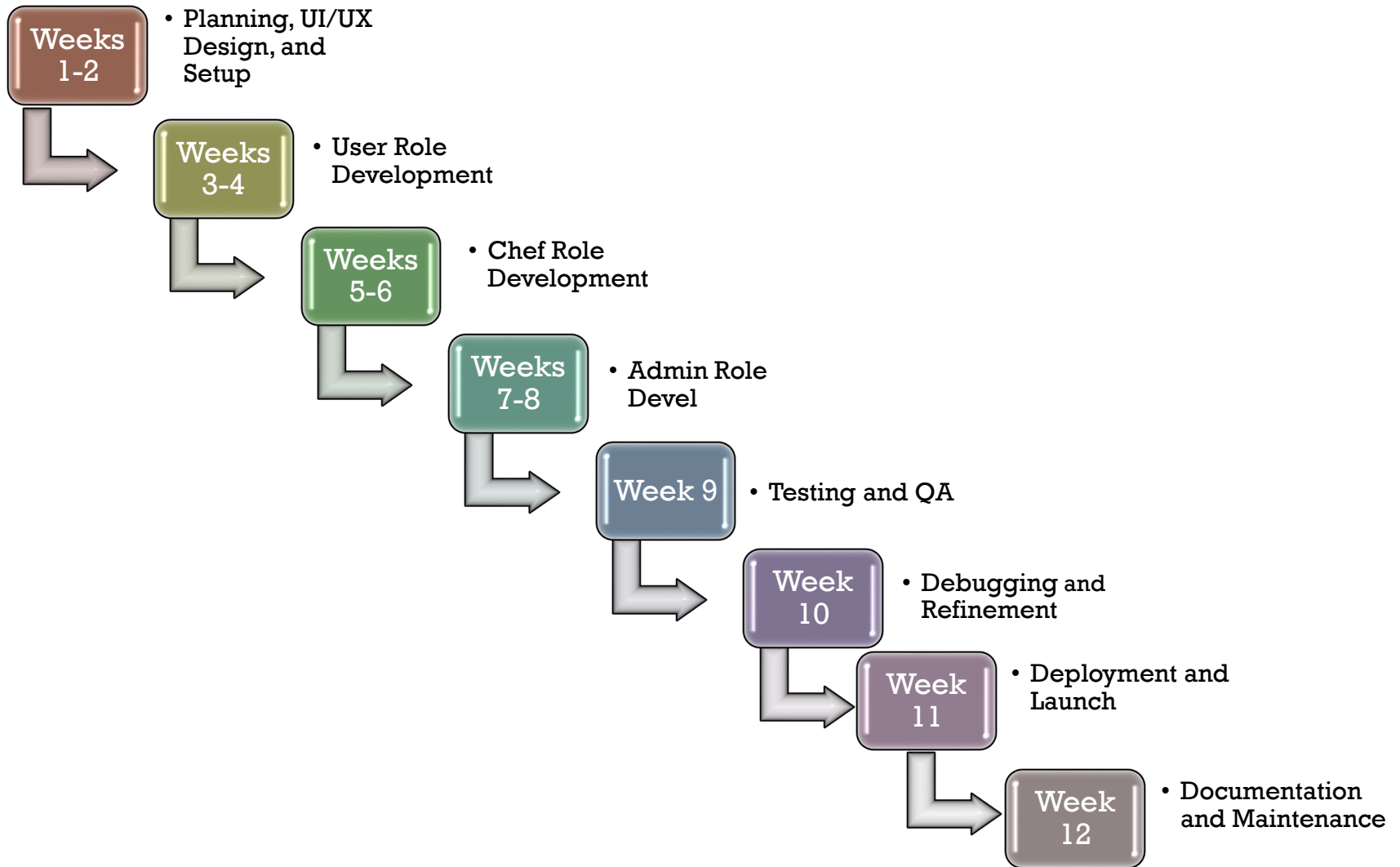
MERN Stack Development



MADE BY KUNTAL OJHA

PROJECT TIMELINE INDEX

- 1. Weeks 1-2: Planning, UI/UX Design, and Setup
- 2. Weeks 3-4: User Role Development
- 3. Weeks 5-6: Chef Role Development
- 4. Weeks 7-8: Admin Role Development
- 5. Week 9: Testing and QA
- 6. Week 10: Debugging and Refinement
- 7. Week 11: Deployment and Launch
- 8. Week 12: Documentation and Maintenance



WEEKS 1-2: PLANNING, UI/UX DESIGN, AND SETUP

1. Requirements Gathering & Finalizing Scope

- Define requirements for each role (user, chef, admin).
- Establish key features and functionalities.

2. UI/UX Design

- Create wireframes using Figma or Adobe XD.
- Design user flows for user, chef, and admin roles.

3. Tech Stack Setup

- Set up project repository (Git/GitHub).
- Initialize MERN stack: Express, React, MongoDB.
- Install necessary libraries.

WEEKS 3-4: USER ROLE DEVELOPMENT

1. QR Code Scanning and Menu Display

- Implement QR code scanning functionality.
- Create API to fetch menu data from MongoDB.
- Develop React component to display menu items.

2. Cart Functionality

- Build a cart component for adding menu items.
- Include name, count, table number, special requests.

3. Checkout and Payment Integration

- Develop checkout page and integrate payment gateway.
- Redirect to confirmation page post-payment.

WEEKS 5-6: CHEF ROLE DEVELOPMENT

1. Chef Login and Authentication

- Create login page for chefs with JWT-based authentication.
- Implement role-based access control.

2. Order Management for Chefs

- Build dashboard to display and manage orders.
- Implement functionality to set preparation timers.
- Display real-time countdown for ongoing orders.

WEEKS 7-8: ADMIN ROLE DEVELOPMENT

1. Admin Dashboard

- Create admin login page and authentication.
- Develop dashboard to view and manage orders.
- Implement filtering options (day, week, month).

2. Data Management

- Implement CRUD functionalities for menu management.
- Build features to generate billing reports and order history.

WEEK 9: TESTING AND QA

1. Unit Testing

- Write unit tests for key components using Jest or Mocha.
- Test API endpoints for consistency and error handling.

2. Integration Testing

- Perform end-to-end testing for all user flows.
- Use automated testing tools like Cypress or Selenium.

WEEK 10: DEBUGGING AND REFINEMENT

1. Debugging

- Identify and fix bugs from testing phase.
- Optimize performance for database queries and APIs.

2. User Experience Improvements

- Refine UI/UX based on feedback.
- Ensure mobile responsiveness and cross-browser compatibility.

WEEK 11: DEPLOYMENT AND LAUNCH

1. Prepare for Deployment

- Set up production environment (AWS, Heroku, DigitalOcean).
- Configure environment variables for production.

2. Launch

- Deploy the application to production server.
- Monitor for post-launch issues and resolve promptly.

WEEK 12: DOCUMENTATION AND MAINTENANCE

1. Documentation

- Write documentation for code, API, and database schema.
- Create user manuals for admin and chef roles.

2. Maintenance and Feature Roadmap

- Plan for regular updates and maintenance.
- Gather user feedback for future enhancements.

THANK YOU