Manzil Sprint Schedule

Sprint 1

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Monday | Tuesday | Wednesday | Thursday | Friday |
|  | Sprint Planning | Daily Standup | Daily Standup | Daily Standup | Daily Standup |
| Registration and authentication tasks  Assign Team roles | **Tasks Completed:**  Designed Sprint Backlog  Set up project structure  **Blockages:**  Confusion about best authentication method  API structure not fully defined | **Tasks Completed:**  API endpoint setup  Database schema finalized  Authentication handling initiated  **Blockages**  Error handling approach unclear  Authentication handling error | **Tasks Completed:**  User authentication API  Registration & login UI developed  **Blockages**  CORS issues during API calls  Role-based access control (RBAC) testing difficulties | **Tasks Completed:**  Database connection with backend  Registration & login features integrated  Testing environment setup  **Blockages:**  Database integration inconsistencies  Testing environment |
| Daily Standup | Daily Standup | Daily Standup | Daily Standup | Sprint Review |
| **Tasks Completed:**  Implemented user authentication  Frontend-backend integration    **Blockages:**  UI/UX inconsistencies affecting user experience  Authentication flow issues during integration  Frontend and backend synchronization delays. | ****Tasks Completed:**** Frontend and backend integration  Implemented role-based access control (RBAC)  **Blockages:**  Difficulties in authentication validation and session management  Role-based access not working correctly for certain user roles. | ****Tasks Completed:**** Implemented role management system  Integrated OAuth/Social Login authentication  **Blockages:**  OAuth integration causing token validation issues  Role-based permissions not correctly applied for all users. | ****Tasks Completed:**** Implemented user registration functionality  Conducted final testing of all authentication features  **Blockages:**  Edge case failures in user registration (e.g., duplicate emails not handled properly)  Inconsistent error messages during authentication failures. | Review what works well , challenges faced and improvement for next sprint  **Sprint Retrospective**  **Do's**  Ensure clear communication about daily tasks. Review backlog & finalize API early in the sprint. Keep API documentation updated for seamless integration.  **Don'ts**  Avoid delaying backend-finalization discussions. Do not ignore testing in the initial sprint phase. |
|  | Sprint 2 |  |  |  |  |
|  | Sprint Planning | Daily Standup | Daily Standup | Daily Standup | Daily Standup |
| Ride Booking Functionality  Real time vehicle Tracking  Fare Estimation | **Tasks Completed:**  Developed backend architecture for ride booking  Implemented Ride Booking API and initial endpoints  **Blockages:**  Ride-matching logic optimization causing delays  API response inconsistencies affecting frontend integration | ****Tasks Completed:**** Started **backend development** for ride booking  Set up **initial database schema and API structure**  **Blockages:**  **Complexity in ride management logic** (matching drivers and riders)  **Backend performance issues** under high-load scenarios | **Tasks Completed:**  Completed ride booking backend development  Designed UI for the ride booking feature  **Blockages**  UI inconsistencies affecting user experience  Backend-frontend integration delays due to API adjustments  User feedback on ride booking flow required further refinements | ****Tasks Completed:**** Integrated **backend with frontend**  Optimized **API responses for better performance**  **Blockages:**  **Delayed API response**, affecting real-time updates  **Frontend-backend synchronization issues,** causing minor glitches |
| Daily Standup | Daily Standup | Daily Standup | Daily Standup | Sprint Review |
| **Tasks Completed:**  Implemented ride status updates (Confirmed, In-Progress, Completed)  Developed logic for handling state transitions  **Blockages:**  Delayed status updates due to API response latency  State transition inconsistencies causing incorrect ride statuses  Edge cases (e.g., ride cancellation or driver unavailability) not fully handled | **Tasks Completed:**  Integrated vehicle tracking functionality  Finalized Location API for accurate geolocation services  **Blockages:**  Inconsistent GPS updates, causing delays in real-time tracking  Location API optimization needed for better accuracy and response time. | **Tasks Completed:**  Implemented fare estimation based on distance and vehicle type  Integrated distance calculation logic for accurate pricing  **Blockages:**  Inaccurate distance calculations in some edge cases (e.g., rerouting, poor GPS signal)  Dynamic fare adjustments based on traffic conditions not fully optimized  Vehicle type-based pricing variations need further validation | **Tasks Completed:**  Tested the ride booking and tracking process  Identified major UI/UX bugs affecting user experience  **Blockages:**  Inconsistent ride tracking updates due to GPS delays  UI glitches affecting the booking flow and ride status visibility  Slow response times in updating ride statuses in real-time | Demo ride-booking, tracking and fare estimation. ****Sprint Retrospective****  Successful **backend-frontend integration** and real-time tracking  Strong **team collaboration** in resolving UI/UX issues  **Challenges Faced:**  **Delayed API confirmation** caused integration slowdowns  **Late UI/UX bug identification** affected sprint timeline  **Scalability concerns** in real-time tracking |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sprint Planning | Daily Standup | Daily Standup | Daily Standup | Daily Standup |
| Defined ride-sharing workflow  UI/UX requirements  fare calculations | **Task Completed:**  Developed backend logic for shared ride allocation  Defined API endpoints  **Blockages:**  API response delays in real-time ride allocation | **Task Completed:**  Design UI for ride-sharing feature  optimize ride allocation logic  **Blockages:**  Issues handling multiple passengers in a single ride | **Task Completed:**  Implemented ride-matching for shared rides  fare-splitting logic  **Blockages:**  Errors in fare calculation when multiple passengers book simultaneously | **Task Completed:**  Backend-frontend data exchange setup  Early testing  **Blockages:**  UI updates lagging during booking requests |
| Daily Standup | Daily Standup | Daily Standup | Daily Standup | Sprint Review |
| **Task Completed:**  Integrated ride-sharing backend with frontend  Resolved API issues  **Blockages:**  API performance issues during heavy usage | **Task Completed:**  Tested ride-sharing feature in real-world cases  **Blockages:**  UI glitches when multiple users book at once | **Task Completed:**  Optimized UI/UX for seamless multi-passenger ride booking  **Blockages:**  Dynamic fare adjustments not working as expected | **Task Completed:**  Finalized ride-sharing discounts and optimized pricing structure  Conducted final tests, identify & fix bugs  **Blockages:**  Ride-sharing notifications not triggering correctly | Demo ride-sharing functionality, real-time updates, and fare-splitting  **Sprint Retrospective:**  **Successful ride-sharing backend-frontend integration Strong team collaboration in resolving UI/UX inconsistencies**  **Challenges Faced:** **Delayed API responses** affected ride-matching speed  **Late UI/UX refinements** required last-minute changes  **Scalability concerns** in handling multiple ride-sharing requests |

Sprint 4:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sprint Planning | Daily Standup | Daily Standup | Daily Standup | Daily Standup |
| Bus booking workflow  Package delivery workflow | **Task Completed:**  Developed backend logic for bus reservations  Designed database schema  **Blockages**:  Handling seat availability in real-time | **Task Completed:**  Designed & implemented seat selection UI  Set availability logic **Blockages**:  UI inconsistencies in seat selection | **Task Completed:**  API integration for bus scheduling and Fare calculation  **Blockages**:  Duplicate seat bookings in simultaneous requests | **Task Completed:**  Conducted early testing  Fixed seat allocation bugs  **Blockages**:  Bus availability not updating in real-time |
| Daily Standup | Daily Standup | Daily Standup | Daily Standup | Sprint Review |
| **Task Completed:**  UI planning for package booking & tracking  **Blockages**:  Delays in tracking updates | **Task Completed:**  Backend API for package tracking,  Live updates  **Blockages**:  API communication issues between package & ride system | **Task Completed:**  Frontend integration with package status updates  **Blockages**:  Incorrect package statuses displayed | **Task Completed:**  Optimized package delivery flow & scheduling  **Blockages**:  Performance issues when handling multiple deliveries | **Demo of bus reservations & package tracking system**  **UI tested for seamless package booking & tracking**  **Sprint Retrospective:** Improve **real-time seat availability checks**  Optimize **package tracking API for faster updates**  Prevent **overlapping seat bookings** during high demand |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sprint Planning | Daily Standup | Daily Standup | Daily Standup | Daily Standup |
| Define payment security measures  Invoice layouts  UI wireframes | **Task Completed:**  Developed payment API  Integrated third-party payment gateways  Planned UI  **Blockages**:  Payment gateway response inconsistencies | **Task Completed:**  Implemented fare calculation & transaction validation logic  **Blockages**:  Handling failed transactions and refunds properly | **Task Completed:**  Added fraud detection logic for secure transactions  Encryption, and security compliance  **Blockages**:  Fraud detection false positives causing transaction failures | **Task Completed:**  Backend testing  Fixed transaction failures  **Blockages**:  Slow response time for high-volume transactions |
| Daily Standup | Daily Standup | Daily Standup | Daily Standup | Sprint Review |
| **Task Completed:**  Developed UI for payment processing & invoice generation screens  **Blockages**:  Real-time payment status updates lagging in frontend | **Task Completed:**  Frontend integration with backend APIs for live payment updates  **Blockages**:  Invoice generation errors for failed payments | **Task Completed:**  Tested refund processing & payment failure handling  **Blockages**:  Payment UI responsiveness issues on different devices | **Task Completed:**  Added fraud detection logic for secure transactions  Encryption, and security compliance  **Blockages**:  Fraud detection false positives causing transaction failures | **Demo of secure payments, invoice generation, and refund handling**  **Payment UI integrated with backend & tested successfully**  **Transaction logging, error handling, and notifications working smoothly**  **Sprint Retrospective:**  Improve real-time transaction updates in UI Reduce refund processing delays for better user experience  Ensure early-stage security & API testing to avoid last-minute issues |