HEXAWARE

RideShare Application

Table of Contents

1 Project Overview

RideShare application is an innovative, AI-powered application exclusively designed for Hexaware employees. This application aims to streamline the commuting experience by connecting employees who have similar travel routes and schedules, thereby reducing individual commuting costs, promoting environmental sustainability, and enhancing the overall sense of community within Hexaware.

1.1 Objectives

- **Cost Reduction**: By facilitating ride-sharing among employees, RideShare application aims to lower the overall commuting expenses for each participant. Employees can share the costs of fuel and vehicle maintenance, making daily travel more affordable.
- **Employee Connectivity**: RideShare application fosters a sense of community by enabling employees to connect and network during their commutes. This can lead to stronger workplace relationships and improved collaboration across different departments.
- **Sustainable Commuting**: By reducing the number of vehicles on the road, RideShare application contributes to a decrease in carbon emissions and traffic congestion. This aligns with Hexaware's commitment to sustainability and corporate social responsibility.
- **Safety and Convenience**: The application provides a secure platform for ride-sharing, with features such as verified profiles, real-time ride tracking, and AI-driven ride matching to ensure safety and convenience for all users.

1.2 Scope

RideShare application is designed to serve Hexaware employees across all locations. The application will support both intra-city and inter-city carpooling. It will be integrated with the company's existing IT infrastructure to facilitate seamless user onboarding and management.

1.3 Key Features

- **User Authentication**: Employees will log in using their corporate credentials, ensuring that only verified Hexaware employees can access the application.
- **AI-Powered Ride Matching**: The application uses advanced AI algorithms to match riders and drivers based on factors such as route similarity, travel schedules, and user preferences, optimizing the carpooling experience.
- **Flexible Ride Scheduling**: Employees can schedule rides in advance or find immediate matches for their travel needs.
- **Cashless Transactions**: RideShare application supports secure, cashless payments, with automatic fare calculation and cost splitting among ride participants.
- **Real-Time Notifications**: Users receive real-time notifications for ride confirmations, cancellations, and updates.

2 Stakeholders

In the RideShare application project, stakeholders are individuals or groups with a vested interest in the development, implementation, and success of the carpooling and ride-sharing application. Stakeholders can be categorized into primary and secondary groups based on their level of involvement and impact on the project.

2.1 Primary Stakeholders

These stakeholders have a direct influence on the project and are the primary users and beneficiaries of the RideShare application application.

- Hexaware Employees
    - **Riders**: Employees who use the application to find and join rides.
        - Impact: Riders benefit from reduced commuting costs, convenience, and enhanced networking opportunities.
        - Role: Provide feedback to improve the application and participate in ride-sharing.
    - **Drivers**: Employees who offer rides to fellow employees.
        - Impact: Drivers benefit from shared commuting costs, the ability to contribute to sustainability efforts, and opportunities to network.
        - Role: Offer rides, maintain punctuality and safety, and provide feedback.
- Hexaware Management
    - Impact: Management gains insights into employee commuting patterns, cost savings, and environmental impact. They also benefit from enhanced employee morale and engagement.
    - Role: Provide strategic direction, ensure alignment with corporate sustainability goals, and promote the application within the organization.

2.2 Secondary Stakeholders

These stakeholders provide support, resources, and oversight to ensure the successful deployment and operation of the RideShare application application.

- IT and Support Teams
    - Impact: Responsible for the technical development, deployment, and maintenance of the application. They ensure the application is secure, reliable, and user-friendly.
    - Role: Provide technical support, perform regular maintenance, and address any technical issues that arise.
- HR Department
    - Impact: HR plays a crucial role in employee onboarding, promoting the application, and addressing any HR-related concerns that may arise from the use of the application.
    - Role: Facilitate user registration, promote the application through internal communications, and address any employee concerns related to ride-sharing policies.

3 Users and Roles

RideShare application caters to three primary user roles within Hexaware: Riders, Drivers, and Administrators. Each role has specific responsibilities and access permissions tailored to their needs within the application.

3.1 Riders

- **Role Description:** Riders are Hexaware employees who seek rides to commute to and from the workplace. They utilize RideShare application to find and join rides offered by other employees (drivers) who are traveling along similar routes.
- **Responsibilities:**
  - **Search for Rides**: Enter travel details such as destination, date, and time to find available rides.
  - **Request Rides**: Send ride requests to drivers based on AI-suggested matches.
  - **Confirm Ride Details**: Confirm ride details, including pickup points and timings, once the driver accepts the request.
  - **Commute**: Join the ride as per the agreed schedule and location.
  - **Provide Feedback**: Rate and provide feedback on the ride experience to ensure service quality and safety.
- **Access Permissions:**
  - View available rides
  - Request to join rides
  - View ride details and driver information
  - Provide ride feedback and ratings

3.2 Drivers

- **Role Description**: Drivers are Hexaware employees who offer rides to other employees (riders). They post their travel plans on RideShare application and accept ride requests from riders traveling along the same route.
- **Responsibilities:**
  - **Post Ride Details**: Enter ride details, including destination, date, time, and available seats.
  - **Review Ride Requests**: Review and accept or reject ride requests from riders.
  - **Coordinate with Riders**: Communicate with riders to confirm pickup points and timings.
  - **Commute**: Provide the ride to the agreed-upon destination, ensuring timely and safe transportation.
  - **Provide Feedback**: Rate and provide feedback on riders to maintain service quality and trust within the community.
- **Access Permissions:**
  - Post and manage ride details
  - View and respond to ride requests
  - View rider information and contact details
  - Provide ride feedback and ratings

3.3 Administrators

- **Role Description**: Administrators are responsible for overseeing the entire RideShare application system. They manage user accounts, monitor ride activities, and generate reports for management review. They ensure the application runs smoothly and address any issues or disputes that arise.
- **Responsibilities**:
  - **User Management:** Add, remove, or update user details to maintain an up-to-date and accurate user database.
  - **Monitor Ride Activities**: Oversee active rides and user activities to ensure compliance with company policies and guidelines.
  - **Generate Reports**: Create and distribute reports on ride usage, cost savings, environmental impact, and user satisfaction.
  - **Issue Resolution**: Address any issues, disputes, or feedback raised by users to maintain a positive user experience.
  - **Policy Enforcement**: Ensure that all rides and user interactions comply with Hexaware's policies and guidelines.
- **Access Permissions**:
  - Full access to user management functions
  - View and monitor all ride activities and user interactions
  - Generate and access detailed reports and analytics
  - Manage application settings and configurations
  - Resolve issues and handle user disputes

4 Process Flow

The RideShare application application involves multiple process flows for different user roles, including riders, drivers, and administrators. Each process flow is designed to ensure a seamless and efficient ride-sharing experience, leveraging AI to enhance matching and user convenience.

4.1 Rider Process Flow

- **Login**:
  - Action: Rider logs into the RideShare application application using their corporate credentials.
  - UI Page: Login Page with fields for username and password.
- **Search for a Ride:**
  - Action: Rider enters their destination, date, and time to search for available rides.
  - UI Page: Search Ride Page with fields for destination, date, and time, and a search button.
- **Match Ride:**
  - Action: The AI algorithm processes the search request and provides a list of suitable ride matches based on route similarity, time preferences, and user ratings.
  - UI Page: Ride Results Page displaying a list of available rides with details (driver name, car type, seats available, estimated time, etc.).
- **Request Ride:**
  - Action: Rider selects a ride from the list and sends a request to the driver.
  - UI Page: Ride Details Page with an option to send a request.
- **Ride Confirmation:**
  - Action: The driver reviews and approves the ride request. The rider receives a notification confirming the ride.
  - Notification: Email and push notification sent to the rider.
- **Commute:**
  - Action: The rider joins the carpool as per the agreed schedule and pickup location.
  - UI Page: Trip Details Page showing real-time updates and ride status.
- **Payment:**
  - Action: The fare is automatically calculated and split among the riders. Payment is processed through the app.
  - UI Page: Payment Summary Page displaying the fare breakdown and payment confirmation.
- **Feedback:**
  - Action: After the ride, the rider provides feedback and rates the driver.
  - UI Page: Feedback Page with rating options and a comments section.

4.2 Driver Process Flow

- **Login:**
  - Action: Driver logs into the RideShare application application using their corporate credentials.
  - UI Page: Login Page with fields for username and password.

- **Post Ride:**
  - Action: Driver enters ride details such as destination, date, time, and available seats.
  - UI Page: Post Ride Page with fields for destination, date, time, and seats available, and a submit button.
- **Match Requests:**
  - Action: The AI algorithm matches rider requests to the driver's ride based on route and time compatibility.
  - UI Page: Ride Requests Page displaying a list of rider requests with details.
- **Review Requests:**
  - Action: Driver reviews the requests and approves or rejects them.
  - UI Page: Request Details Page with options to approve or reject requests.
- **Ride Confirmation:**
  - Action: Approved riders receive notifications confirming the ride.
  - Notification: Email and push notification sent to the riders.
- **Commute:**
  - Action: Driver picks up and drops off the riders according to the schedule.
  - UI Page: Trip Details Page showing real-time updates and ride status.
- **Payment:**
  - Action: The fare is automatically calculated and split among the riders. Payment is processed through the app.
  - UI Page: Payment Summary Page displaying the fare breakdown and payment confirmation.
- **Feedback:**
  - Action: After the ride, the driver provides feedback and rates the riders.
  - UI Page: Feedback Page with rating options and a comments section.

4.3 Administrator Process Flow

- **Login:**
  - Action: Admin logs into the RideShare application application using their corporate credentials.
  - UI Page: Login Page with fields for username and password.
- **Manage Users:**
  - Action: Admin can add, remove, or update user details to maintain an accurate user database.
  - UI Page: User Management Page with options to add, edit, or delete users.
- **Monitor Rides:**
  - Action: Admin monitors active rides and user activity to ensure compliance with policies and guidelines.
  - UI Page: Ride Monitoring Dashboard displaying real-time ride data and user activities.
- **Generate Reports:**
  - Action: Admin generates reports on ride usage, cost savings, environmental impact, and user satisfaction.
  - UI Page: Reports Page with options to generate and download various reports.
- **Issue Resolution:**
  - Action: Admin addresses any issues or disputes raised by users to maintain a positive experience.
  - UI Page: Issue Management Page with tools to track and resolve issues.
- **Policy Enforcement:**
  - Action: Admin ensures that all rides and user interactions comply with Hexaware's policies and guidelines.
  - UI Page: Policy Management Page with options to view and update policies.

5 UI Pages

RideShare application's user interface is designed to provide a seamless and intuitive experience for all user roles: Riders, Drivers, and Administrators. Each UI page is carefully crafted to ensure ease of use, accessibility, and functionality.

5.1 Rider UI Pages

- **Login Page**
  - **Components**: Fields for username and password, login button, forgot password link.
  - **Functionality**: Allows riders to log in using their corporate credentials. The forgot password link redirects to the password recovery process.
  - **Design**: Simple, clean design with Hexaware branding, secure login prompt.
- **Home Page**
  - **Components**: Search/Post Ride options, upcoming rides section, user profile link.
  - **Functionality**: Main navigation hub. Riders can quickly access ride search or post options, view their upcoming rides, and access their profile settings.
  - **Design**: Dashboard layout with easy navigation, highlighted sections for quick access.
- **Search Ride Page**
  - **Components**: Fields for destination, date, time, search button.
  - **Functionality**: Riders input their travel details to search for available rides. The search button triggers the AI algorithm to find matches.
  - **Design**: Form layout with prominent search button, user-friendly input fields.
- **Ride Results Page**
  - **Components**: List of available rides, ride details (driver name, car type, seats available, estimated time, etc.), request ride button.
  - **Functionality**: Displays a list of matched rides based on the rider's search criteria. Riders can view detailed ride information and send ride requests.
  - **Design**: List view with expandable ride details, clear call-to-action buttons.
- **Ride Details Page**
  - **Components**: Detailed ride information (route, driver profile, pickup points, estimated time), send request button.
  - **Functionality**: Provides comprehensive details about a selected ride. Riders can review the information and send a ride request to the driver.
  - **Design**: Detailed view with sections for each piece of information, prominent request button.
- **Trip Details Page**
  - **Components**: Real-time updates (pickup points, estimated time of arrival, current location), ride status.
  - **Functionality**: Allows riders to track their ride in real-time, view status updates, and estimated arrival times.
  - **Design**: Map view with live tracking, status indicators, clean and informative layout.
- **Payment Summary Page**
  - **Components**: Fare breakdown, payment confirmation.

        - **Functionality**: Displays the cost breakdown of the ride, confirms the payment process, and splits the fare among riders.
        - **Design**: Summary layout with clear breakdown, confirmation message.
    - **Feedback Page**
        - **Components**: Rating options (stars or thumbs up/down), comments section.
        - **Functionality**: Allows riders to provide feedback and rate their ride experience.
        - **Design**: Simple rating system, optional comments field, submit button.

5.2 Driver UI Pages

- **Login Page**
    - **Components**: Fields for username and password, login button, forgot password link.
    - **Functionality**: Allows drivers to log in using their corporate credentials. The forgot password link redirects to the password recovery process.
    - **Design**: Consistent with the rider login page, secure login prompt.
- **Home Page**
    - **Components**: Post Ride option, upcoming rides section, user profile link.
    - **Functionality**: Main navigation hub for drivers. They can post new rides, view their upcoming rides, and access their profile settings.
    - **Design**: Dashboard layout, easy navigation, highlighted sections for quick access.
- **Post Ride Page**
    - **Components**: Fields for destination, date, time, seats available, submit button.
    - **Functionality**: Drivers enter details about their ride and post it to find riders.
    - **Design**: Form layout with clear input fields, prominent submit button.
- **Ride Requests Page**
    - **Components**: List of rider requests, request details (rider name, pickup location, etc.), approve/reject buttons.
    - **Functionality**: Displays incoming ride requests. Drivers can review details and approve or reject requests.
    - **Design**: List view with expandable details, clear action buttons.
- **Request Details Page**
    - **Components**: Detailed rider information (profile, pickup location, contact details), approve/reject buttons.
    - **Functionality**: Provides comprehensive details about each rider request. Drivers can make informed decisions to approve or reject.
    - **Design**: Detailed view, clear call-to-action buttons.
- **Trip Details Page**
    - **Components**: Real-time updates (pickup points, estimated time of arrival, current location), ride status.
    - **Functionality**: Allows drivers to track their ride in real-time, view status updates, and estimated arrival times.
    - **Design**: Map view with live tracking, status indicators.

- **Payment Summary Page**
  - **Components**: Fare breakdown, payment confirmation.
  - **Functionality**: Displays the cost breakdown of the ride, confirms the payment process.
  - **Design**: Summary layout, clear breakdown, confirmation message.
- **Feedback Page**
  - **Components**: Rating options (stars or thumbs up/down), comments section.
  - **Functionality**: Allows drivers to provide feedback and rate their rider experience.
  - **Design**: Simple rating system, optional comments field, submit button.

5.3 Administrator UI Pages

- **Login Page**
  - **Components**: Fields for username and password, login button, forgot password link.
  - **Functionality**: Allows administrators to log in using their corporate credentials.
  - **Design**: Consistent with other login pages, secure login prompt.
- **User Management Page**
  - **Components**: Options to add, edit, delete users, search functionality.
  - **Functionality**: Admins manage user accounts, ensuring the database is up-to-date and accurate.
  - **Design**: Administrative dashboard, list view of users with action buttons.
- **Ride Monitoring Dashboard**
  - **Components**: Real-time data on active rides, user activities, alerts, and notifications.
  - **Functionality**: Admins monitor ongoing rides and user interactions, ensuring policy compliance.
  - **Design**: Real-time dashboard, visual indicators for alerts, and ride status.
- **Reports Page**
  - **Components**: Options to generate, view, and download various reports (ride usage, cost savings, environmental impact).
  - **Functionality**: Admins generate and review reports to gain insights into the application's performance and impact.
  - **Design**: Report generation tools, downloadable formats, visual charts, and graphs.
- **Issue Management Page**
  - **Components**: Tools to track, manage, and resolve user issues and disputes.
  - **Functionality**: Admins handle user complaints and issues, ensuring a positive user experience.
  - **Design**: Issue tracking system, status updates, resolution tools.
- **Policy Management Page**
  - **Components**: Options to view, update, and enforce company policies related to ride-sharing.
  - **Functionality**: Admins manage and enforce policies to ensure all rides comply with Hexaware’s guidelines.
  - **Design**: Policy documents, edit tools, compliance indicators.

6 AI-Powered Features and Algorithms

RideShare application leverages various algorithms to ensure an efficient, reliable, and user-friendly experience. These algorithms play crucial roles in matching riders with drivers, optimizing routes, calculating payments, and personalizing user experiences.

6.1 Ride Matching Algorithm

Purpose: Match riders with suitable drivers based on destination, route, and time preferences.

- **Input Collection**:
  - Rider inputs: destination, date, time, and any preferences (e.g., only same-gender drivers, non-smoking cars).
  - Driver inputs: destination, route, date, time, and available seats.
- **Preprocessing**:
  - Normalize inputs for consistency (e.g., format date and time, standardize location data).
- **Similarity Calculation**:
  - Calculate the similarity score between rider and driver routes using geographic information system (GIS) algorithms (e.g., Haversine formula for distance calculation).
  - Consider time compatibility and preferences in the similarity score.
- **AI-enhanced Matching**:
  - Apply a machine learning model (e.g., random forest or neural network) trained on historical ride data to predict the best matches.
  - The model considers factors like past ride preferences, user ratings, and feedback.
- **Output**:
  - Generate a list of potential matches ranked by similarity score and AI prediction.
  - Present the top matches to the rider for selection.
- **Notification**:
  - Send notifications to the matched drivers for approval.

6.2 Route Optimization Algorithm

Purpose: Optimize the route for drivers to minimize travel time and distance while accommodating multiple riders.

- **Input Collection**:
  - Collect pickup and drop-off locations for all riders in the carpool.
  - Retrieve real-time traffic data and road conditions.
- **Initial Route Planning**:
  - Use a shortest path algorithm (e.g., Dijkstra's algorithm) to determine the initial route.
- **Route Optimization**:
  - Apply a traveling salesman problem (TSP) solver to optimize the sequence of pickups and drop-offs.
  - Use real-time traffic data to adjust the route dynamically.

- **AI-enhanced Adjustment**:
    - Incorporate machine learning models to predict traffic patterns and suggest route adjustments.
    - Continuously update the route based on real-time data and rider feedback.
- **Output**:
    - Provide the optimized route to the driver with estimated arrival times for each stop.

6.3 Payment Calculation Algorithm

Purpose: Calculate and split the ride fare among riders based on distance traveled.

- **Input Collection**:
    - Collect trip data: total distance, number of riders, individual pickup, and drop-off points.
- **Fare Calculation**:
    - Determine the base fare and additional charges (e.g., per kilometer rate).
    - Calculate the total fare for the trip.
- **Distance-based Split**:
    - Calculate the distance each rider traveled.
    - Split the total fare proportionally based on each rider's distance traveled.
- **Adjustments**:
    - Apply any discounts or promotional codes.
    - Adjust the fare for any additional preferences or requirements (e.g., express route).
- **Output**:
    - Generate the fare breakdown for each rider.
    - Process the payment through the integrated payment system.

6.4 Notification Algorithm

Purpose: Ensure timely and relevant notifications are sent to users (riders and drivers) about ride statuses and updates.

- **Event Detection**:
    - Monitor key events (e.g., ride request, ride confirmation, trip start, trip end, feedback submission).
- **Notification Rules**:
    - Define rules for when and how notifications should be sent based on event type and user preferences.
- **Message Generation**:
    - Generate notification messages tailored to the event and user (e.g., "Your ride request has been accepted", "Your driver is arriving in 5 minutes").
- **Delivery Optimization**:
    - Use AI to determine the best time and channel for delivery (e.g., push notification, email, SMS).
    - Ensure notifications are sent in a timely manner to maximize user engagement.
- **Output**:
    - Send notifications to users.

- Log notifications for tracking and analytics.

6.5 Feedback Analysis Algorithm

Purpose: Analyze user feedback to improve service quality and user satisfaction.

- **Feedback Collection**:
  - Collect feedback and ratings from riders and drivers after each trip.
- **Preprocessing**:
  - Clean and preprocess the feedback data (e.g., remove noise, standardize format).
- **Sentiment Analysis**:
  - Use natural language processing (NLP) techniques to analyze the sentiment of the feedback (positive, negative, neutral).
  - Apply sentiment analysis models (e.g., VADER, BERT) to categorize and score feedback.
- **Trend Analysis**:
  - Identify common themes and trends in the feedback (e.g., frequent complaints about punctuality, positive comments about vehicle cleanliness).
- **AI-enhanced Insights**:
  - Apply machine learning models to predict areas of improvement and suggest actionable insights.
  - Continuously train the models with new feedback data to enhance accuracy.
- **Output**:
  - Generate reports and insights for administrators.
  - Provide actionable recommendations to drivers and the RideShare application team.

7 Notification Services

RideShare application's notification services are designed to keep users informed, engaged, and up-to-date with real-time ride-related information, ensuring a smooth and efficient ride-sharing experience. Notifications are categorized based on their purpose, delivery channels, and the users they target (riders, drivers, and administrators).

7.1 Types of Notifications

- **Ride Request Notifications**
  - **Riders**: When a ride request is sent, accepted, rejected, or updated.
  - **Drivers**: When a new ride request is received, and when a ride request is accepted or rejected by a rider.
  - **Administrators**: Summary of daily ride requests for monitoring.
- **Ride Status Updates**
  - **Riders**: When a ride is confirmed, the driver is en route, approaching, starting the trip, or arriving at the destination.
  - **Drivers**: When a rider cancels a ride, the trip is about to start, ongoing trip status updates.
  - **Administrators**: Real-time monitoring of ongoing rides.
- **Payment Notifications**
  - **Riders**: Fare calculation, payment confirmation, receipt of payment.
  - **Drivers**: Payment received, payment summary for completed trips.
  - **Administrators**: Daily and monthly payment summaries for auditing.
- **Reminders and Alerts**
  - **Riders**: Upcoming ride reminders, ride cancellation alerts.
  - **Drivers**: Ride schedule reminders, vehicle maintenance alerts.
  - **Administrators**: Important policy changes, system maintenance alerts.
- **Feedback and Ratings**
  - **Riders**: Request for feedback after a ride, response to submitted feedback.
  - **Drivers**: Request for feedback after a ride, response to submitted feedback.
  - **Administrators**: Summary of user feedback and ratings for analysis.
- **Promotional and Informational Notifications**
  - **Riders**: New features, promotional offers, system updates.
  - **Drivers**: New features, promotional offers, system updates.
  - **Administrators**: Updates on system upgrades, new policies.

7.2 Delivery Channels

- **Push Notifications**
  - Instant notifications sent to the user's mobile device or web browser.
  - Used for real-time updates such as ride status changes, reminders, and urgent alerts.
- **Email Notifications**
  - Detailed notifications sent to the user's email address.
  - Used for payment receipts, feedback requests, promotional offers, and system updates.

- **SMS Notifications**
  - Short message notifications sent to the user's phone.
  - Used for critical updates such as ride confirmations, driver arrivals, and emergency alerts.

7.3 Notification Process Flow

- **Event Trigger**:
  - An event occurs within the system (e.g., a ride request is made, a payment is processed).
- **Notification Rules**:
  - The system checks predefined rules to determine the type of notification to send and the appropriate channel.
- **Message Generation**:
  - The system generates a notification message based on the event and the user's preferences.
  - AI models can personalize the content and timing of the message to increase relevance and engagement.
- **Notification Dispatch**:
  - The message is sent through the chosen delivery channel (push, email, SMS).
  - The system ensures the message is delivered promptly and reliably.
- **Confirmation and Logging**:
  - The system logs the notification event for tracking and analytics.
  - Users receive a confirmation that the notification has been sent.

7.4 Examples of Notification Scenarios

- **Ride Request Confirmation**:
  - **Trigger**: Rider submits a ride request.
  - **Notification**: Push notification to the driver, email confirmation to the rider.
- **Driver En Route**:
  - **Trigger**: Driver starts heading towards the pickup location.
  - **Notification**: Push notification to the rider with estimated arrival time.
- **Payment Receipt**:
  - **Trigger**: Payment is processed after the ride.
  - **Notification**: Email receipt to the rider, payment confirmation SMS to the driver.
- **Upcoming Ride Reminder**:
  - **Trigger**: Ride scheduled to start in 30 minutes.
  - **Notification**: Push notification reminder to both rider and driver.
- **Feedback Request**:
  - **Trigger**: Ride is completed.
  - **Notification**: Email request for feedback to both rider and driver.

8 Document Generation

RideShare application includes various document generation features to support different aspects of the ride-sharing process. These documents provide detailed information for users, enhance transparency, and facilitate administrative tasks. The documents generated by the system include invoices, trip sheets, reports, and summaries, all of which are crucial for maintaining records and ensuring a smooth operational flow.

8.1 Invoices

Purpose:

- To provide riders with a detailed breakdown of the fare and payment information after a ride is completed.

Components:

- **Rider Details**: Name, contact information, Hexaware employee ID.
- **Trip Details**: Date and time, pickup and drop-off locations, distance traveled.
- **Fare Breakdown**: Base fare, distance fare, any additional charges (e.g., peak time surcharge, express route).
- **Payment Information**: Total fare, payment method, transaction ID.
- **Driver Information**: Name, vehicle details.

Process Flow:

- **Ride Completion**: The ride is marked as complete in the system.
- **Data Collection**: Gather trip details, fare calculation, and payment information.
- **Document Generation**: Create an invoice using a predefined template.
- **Delivery**: Send the invoice to the rider via email and make it available in the user's account on the app.

8.2 Trip Sheets

Purpose:

- To provide drivers with a detailed summary of each trip, including rider details, route, and any special instructions.

Components:

- **Driver Details**: Name, vehicle information.
- **Rider Details**: Name, contact information, employee ID.
- **Trip Details**: Date, time, pickup and drop-off locations, route map.
- **Special Instructions**: Any rider preferences or special requests (e.g., non-smoking, quiet ride).

Process Flow:

- **Ride Request**: A ride request is accepted by the driver.
- **Data Collection**: Gather trip details, rider preferences, and route information.
- **Document Generation**: Create a trip sheet using a predefined template.
- **Delivery**: Send the trip sheet to the driver via the app and make it printable if needed.

8.3 Daily/Monthly Reports

Purpose:

- To provide administrators with comprehensive summaries of ride activities, financials, and user feedback.

Components:

- **Summary of Rides**: Total number of rides, distance traveled, time taken.
- **Financial Overview**: Total revenue, average fare, payment methods.
- **User Feedback**: Summary of ratings and comments, identified issues, and satisfaction scores.
- **Operational Metrics**: Average ride acceptance time, driver availability, ride cancellations.

Process Flow:

- **Data Aggregation**: Collect data from all rides within the reporting period.
- **Data Analysis**: Analyze data to generate insights and identify trends.
- **Document Generation**: Create a report using a comprehensive template.
- **Delivery**: Send the report to administrators via email and make it available on the admin dashboard.

8.4 Ride History Summaries

Purpose:

- To provide users (riders and drivers) with a summary of their ride history for personal record-keeping and expense tracking.

Components:

- **User Details**: Name, contact information, employee ID (for riders).
- **Ride History**: List of rides with date, time, pickup/drop-off locations, fare, and payment status.
- **Total Distance and Fare**: Cumulative distance traveled and total fare paid/received.
- **Feedback Summary**: Summary of ratings given/received and feedback comments.

Process Flow:

- **Data Collection**: Gather historical ride data for the user.
- **Data Compilation**: Compile the data into a coherent summary.
- **Document Generation**: Create a ride history summary using a predefined template.
- **Delivery**: Make the summary available in the user's account on the app and send via email if requested.

HEXAWARE

About Hexaware

Hexaware is a global technology and business process services company. Our 28,400 Hexawarians wake up every day with a singular purpose; to create smiles through great people and technology. With this purpose gaining momentum, we are well on our way to realizing our vision of being the most loved digital transformation partner in the world. We also seek to protect the planet and build a better tomorrow for our customers, employees, partners, investors, and the communities in which we operate.

With 45+ offices in 19 countries, we empower enterprises worldwide to realize digital transformation at scale and speed by partnering with them to build, transform, run, and optimize their technology and business processes.

Learn more about Hexaware at www.hexaware.com