Epic 1: User Authentication and Authorization  
  
Feature 1.1: Login Page  
User story 1.1.1: As a user, I want to be able to access the login page to enter my credentials.  
  
Acceptance criteria:  
1. The login page should have input fields for username and password.  
2. The login page should have a "Login" button to submit the credentials.  
3. The login page should have a "Forgot Password" link for password recovery.  
4. The login page should display appropriate error messages for invalid credentials.  
  
Feature 1.2: User Authentication  
User story 1.2.1: As a user, I want to be able to authenticate my credentials to access the system.  
  
Acceptance criteria:  
1. Upon entering valid credentials and clicking the "Login" button, the user should be authenticated.  
2. Upon entering invalid credentials, the user should not be authenticated and should see an error message.  
3. The system should enforce password complexity rules during authentication.  
  
Feature 1.3: User Authorization  
User story 1.3.1: As a user, I want to be authorized to access specific functionalities based on my role.  
  
Acceptance criteria:  
1. Users with the role of "Admin" should have access to all functionalities.  
2. Users with the role of "User" should have limited access to certain functionalities.  
3. Unauthorized users should not be able to access restricted functionalities.  
4. The system should enforce role-based access control for different user roles.  
  
UI for Warranty Product Field:  
User story 1.3.2: As a user, I want to input warranty product details for a policyholder.  
  
Acceptance criteria:  
1. The UI should include fields for Policyholder's Name, Address Line 1, Address Line 2, City, State, Country, Postal Code, Mobile Number, Email, VIN, License Plate, Make, Model, Mileage, First Registration Date, and Maker Warranty End Date.  
2. All fields should have appropriate validation rules (e.g., required fields, format validation).  
3. The system should save the warranty product details accurately in the database.  
4. The UI should provide feedback on successful submission of warranty product details.

Epic 2: Vehicle Warranty Management  
  
Feature 2.1: Coverage selection for vehicle  
  
User story 2.1.1: As a user, I want to be able to select the coverage plan for my vehicle so that I can choose the appropriate warranty option.  
  
Acceptance criteria:  
1. User should be able to see a list of available coverage plans for their vehicle.  
2. User should be able to select one coverage plan from the list.  
3. User should receive confirmation of the selected coverage plan.  
  
Feature 2.2: Vehicle information capture  
  
User story 2.2.1: As a user, I want to input my vehicle information so that the warranty plan can be tailored to my specific vehicle.  
  
Acceptance criteria:  
1. User should be able to input the following vehicle information: VIN, License Plate, Make, Model, Mileage, First Registration Date.  
2. All fields should be validated for correct format and data type.  
3. User should receive confirmation once all information is successfully input.  
  
Feature 2.3: Coverage plan options  
  
User story 2.3.1: As a user, I want to view and select from different coverage plan options for my vehicle warranty.  
  
Acceptance criteria:  
1. User should be able to see a list of coverage plan options.  
2. Each coverage plan option should display details such as coverage period, cost, and included services.  
3. User should be able to select one coverage plan option.  
4. User should receive confirmation of the selected coverage plan option.

Epic 3: UI Customization and Theme Selection  
  
Feature 3.1: UI Configuration  
User story 3.1.1: As a system administrator, I want to be able to configure the layout of the warranty product field UI.  
  
Acceptance criteria:  
1. The system administrator should be able to rearrange the order of the fields in the UI.  
2. The system administrator should be able to add new fields to the UI.  
3. The system administrator should be able to remove existing fields from the UI.  
4. Changes made to the UI configuration should be reflected immediately.  
  
Feature 3.2: Customizable User Interface  
User story 3.2.1: As a policyholder, I want to customize the appearance of the warranty product field UI to suit my preferences.  
  
Acceptance criteria:  
1. The policyholder should be able to choose a color scheme for the UI.  
2. The policyholder should be able to select a font style and size for the UI.  
3. The policyholder should be able to set a background image for the UI.  
4. The customized UI settings should be saved for future visits.  
  
Feature 3.3: Theme Selection Options  
User story 3.3.1: As a user, I want to be able to select from different themes for the warranty product field UI.  
  
Acceptance criteria:  
1. The user should be presented with at least three different theme options.  
2. Each theme should have a distinct color scheme and layout.  
3. The user should be able to preview each theme before selecting one.  
4. Once a theme is selected, it should be applied to the entire UI consistently.  
5. The selected theme should persist across sessions for the user.

Epic 4: Reporting and Analysis Enhancements  
  
Feature 4.1: Reporting services integration  
User story 4.1.1: As a system administrator, I want to integrate reporting services to generate comprehensive reports for analysis.  
Acceptance criteria:  
1. The reporting services should be able to pull data from the database accurately.  
2. The reports should be customizable based on user preferences.  
3. The integration should support scheduling of automated report generation.  
4. Error handling mechanisms should be in place to handle any issues during report generation.  
  
Feature 4.2: CLUE Report generation  
User story 4.2.1: As an insurance agent, I want to generate CLUE (Comprehensive Loss Underwriting Exchange) reports for policyholders.  
Acceptance criteria:  
1. The CLUE reports should include detailed information about the policyholder's claims history.  
2. The reports should be generated in a standardized format.  
3. The system should validate the accuracy of the data before generating the report.  
4. CLUE reports should be accessible only to authorized personnel.  
  
Feature 4.3: Record Report generation  
User story 4.3.1: As a claims adjuster, I want to generate detailed record reports for each claim filed.  
Acceptance criteria:  
1. The record reports should include information about the claimant, claim details, and resolution status.  
2. Reports should be generated in a timely manner to assist in claim processing.  
3. The system should ensure data privacy and security while generating record reports.  
4. Record reports should be easily exportable for further analysis.  
  
Feature 4.4: Reliability Report generation  
User story 4.4.1: As a quality assurance manager, I want to generate reliability reports to assess the performance of our services.  
Acceptance criteria:  
1. Reliability reports should include metrics such as uptime, response times, and error rates.  
2. The reports should provide insights into the overall reliability of the system.  
3. Historical data should be included in the reports for trend analysis.  
4. Reliability reports should be generated on a regular basis for continuous monitoring.  
  
Feature 4.5: Sentiment Analysis integration  
User story 4.5.1: As a customer service representative, I want to integrate sentiment analysis to understand customer feedback.  
Acceptance criteria:  
1. Sentiment analysis should be able to process customer feedback in real-time.  
2. The system should categorize feedback into positive, negative, or neutral sentiments.  
3. Sentiment analysis results should be used to improve customer service processes.  
4. The integration should support multilingual sentiment analysis for a diverse customer base.

Epic 5: Vehicle Data Management  
  
Feature 5.1: Fetching vehicle model factors  
  
User story 5.1.1: As a user, I want to be able to fetch vehicle model factors so that I can analyze and compare different vehicle models effectively.  
  
Acceptance criteria:  
1. The system should provide a list of available vehicle models with their corresponding factors such as make, model, year, and price.  
2. The user should be able to filter and search for specific vehicle models based on different criteria.  
3. The system should display accurate and up-to-date information for each vehicle model factor.  
  
Feature 5.2: API integration for vehicle data  
  
User story 5.2.1: As a developer, I want to integrate APIs for fetching vehicle data so that I can access real-time information about vehicles.  
  
Acceptance criteria:  
1. The system should have APIs that allow developers to retrieve vehicle data such as make, model, year, and price.  
2. The APIs should be well-documented and easy to integrate into external applications.  
3. The system should handle API requests efficiently and provide timely responses.  
  
Feature 5.3: Data validation and normalization  
  
User story 5.3.1: As a user filling out warranty product information, I want the data to be validated and normalized to ensure accuracy and consistency.  
  
Acceptance criteria:  
1. The system should validate all input fields for the warranty product form, including Policyholder's Name, Address, City, State, Country, Postal Code, Mobile Number, Email, VIN, License Plate, Make, Model, Mileage, First Registration Date, and Maker Warranty End Date.  
2. Data validation should include checks for required fields, data formats (e.g., email validation), and length restrictions.  
3. The system should normalize data such as converting input to uppercase, formatting dates consistently, and standardizing phone numbers.  
4. Error messages should be displayed for any validation failures, guiding users on how to correct the input.

Epic 6: Warranty Coverage Management  
  
Feature 6.1: Standard based coverages  
  
User story 6.1.1: As a policyholder, I want to view the standard warranty coverages available for my product so that I can understand the protection provided.  
  
Acceptance criteria:  
1. The system should display a list of standard warranty coverages available.  
2. Each standard coverage should include details such as coverage type, duration, and exclusions.  
3. Policyholder should be able to select a standard coverage option.  
4. The system should calculate the premium based on the selected standard coverage.  
  
Feature 6.2: Coverage customization options  
  
User story 6.2.1: As a policyholder, I want to customize my warranty coverage options to suit my specific needs.  
  
Acceptance criteria:  
1. The system should allow policyholders to add additional coverage options beyond the standard ones.  
2. Policyholder should be able to adjust coverage duration and coverage limits.  
3. The system should provide a summary of the customized coverage options before finalizing the selection.  
4. Policyholder should be able to save and edit customized coverage options for future reference.  
  
Feature 6.3: Compliance with industry standards  
  
User story 6.3.1: As a compliance officer, I want to ensure that the warranty coverages offered meet industry standards and regulations.  
  
Acceptance criteria:  
1. The system should validate that all warranty coverages adhere to industry regulations and standards.  
2. Compliance officer should be able to review and approve new warranty coverage options before they are made available to policyholders.  
3. The system should generate compliance reports showing the adherence to industry standards.  
4. Any changes to industry standards should trigger a review of existing warranty coverages for compliance updates.

Epic 7: Rating System Enhancement  
  
Feature 7.1: High level requirement of the Rating on the Factors  
User story 7.1.1: As a system user, I want to understand the factors that contribute to the overall rating of a product.  
  
Acceptance criteria:  
1. The system should provide a clear list of factors that are considered in the rating calculation.  
2. Each factor should have a description explaining its importance in the rating process.  
3. The factors should be easily accessible to the user through the system interface.  
  
Feature 7.2: Rating calculation logic  
User story 7.2.1: As a system user, I want the rating calculation to be accurate and consistent based on the defined factors.  
  
Acceptance criteria:  
1. The system should accurately calculate the rating based on the input values for each factor.  
2. The rating calculation should be consistent across different product entries.  
3. The system should handle edge cases such as missing or invalid input values gracefully.  
  
Feature 7.3: Factor weightage configuration  
User story 7.3.1: As a system administrator, I want to be able to configure the weightage of each factor in the rating calculation.  
  
Acceptance criteria:  
1. The system should allow the administrator to set the weightage for each factor through a user-friendly interface.  
2. The weightage configuration should be saved and applied to the rating calculation process.  
3. The system should validate the total weightage to ensure it sums up to 100%.  
4. Changes in factor weightage should reflect in real-time on the rating calculation results.  
  
UI for Warranty Product Field:  
User story 7.3.2: As a system user, I want to input and view detailed information about a warranty product.  
  
Acceptance criteria:  
1. The UI should include fields for Policyholder's Name, Address Line 1, Address Line 2, City, State, Country, Postal Code, Mobile Number, Email, VIN, License Plate, Make, Model, Mileage, First Registration Date, and Maker Warranty End Date.  
2. The UI should provide validation for mandatory fields and format checks for fields like Mobile Number, Email, VIN, and First Registration Date.  
3. The system should store the entered warranty product information accurately and securely.  
4. The UI should display the entered information for reference and editing purposes.

Epic 8: AI Integration for Warranty Product  
  
Feature 8.1: Gen AI enabled features  
User story 8.1.1: As a user, I want to be able to generate AI-enabled features for the warranty product field.  
  
Acceptance criteria:  
1. The system should be able to generate AI-enabled features based on the provided data fields.  
2. The AI-enabled features should be accurate and relevant to the warranty product field.  
3. The generated AI features should be displayed in a user-friendly format for easy understanding.  
  
Feature 8.2: AI-driven recommendations  
User story 8.2.1: As a user, I want to receive AI-driven recommendations for the warranty product field.  
  
Acceptance criteria:  
1. The system should provide AI-driven recommendations based on the input data fields.  
2. The recommendations should be tailored to the specific needs of the warranty product field.  
3. The recommendations should be updated in real-time as new data is entered.  
4. The system should provide explanations for the recommendations to enhance user understanding.  
  
Feature 8.3: AI-powered decision support  
User story 8.3.1: As a user, I want AI-powered decision support for the warranty product field.  
  
Acceptance criteria:  
1. The system should provide AI-powered decision support to assist users in making informed decisions related to warranty products.  
2. The decision support should take into account all relevant data fields provided by the user.  
3. The system should provide clear reasoning behind the recommended decisions.  
4. The AI-powered decision support should be customizable based on user preferences and requirements.

Epic 9: Industry Compliance and Regulatory Requirements  
  
Feature 9.1: Industry standard compliance  
User story 9.1.1: As a policyholder, I want to provide my personal information for industry standard compliance.  
  
Acceptance criteria:  
1. The system should have fields for Policyholder's Name, Address Line 1, Address Line 2, City, State, Country, Postal Code, Mobile Number, Email.  
2. All fields should have proper validation to ensure accurate data entry.  
3. The system should display appropriate error messages for missing or incorrect information.  
  
Feature 9.2: Regulatory requirements implementation  
User story 9.2.1: As a policyholder, I want to provide my vehicle information to meet regulatory requirements.  
  
Acceptance criteria:  
1. The system should have fields for VIN, License Plate, Make, Model, Mileage, First Registration Date, Maker Warranty End Date.  
2. VIN and License Plate fields should have proper validation to ensure correct format.  
3. The system should validate the Maker Warranty End Date against the First Registration Date.  
4. The system should display relevant warnings if the vehicle information does not meet regulatory standards.  
  
Feature 9.3: Certification process integration  
User story 9.3.1: As a policyholder, I want to complete the certification process seamlessly.  
  
Acceptance criteria:  
1. The system should integrate with the certification process to validate the provided information.  
2. Upon successful certification, the policyholder should receive confirmation of compliance.  
3. In case of any issues during certification, the system should provide clear instructions on how to resolve them.  
4. The system should maintain a log of certification activities for auditing purposes.