# Holiday Travel Insurance

**AN INTERNSHIP REPORT**

Submitted by

|  |
| --- |
| **Name of the Student:** Ashwin Ravi  **Register Number:** 3122 21 5001 014 |
| **Degree:** BE-CSE  **Year of Study:** 4th Year(2021-2025)  **College Name:** SSN College of Engineering |

**Under the guidance of**

**Supervisor Name: Naganand Shenoy**

**Designation: Business Architect**

**Company Name: Areteans**

**September 2024**

**Introduction**

The Bharat Insurance Company is developing a Travel Insurance application using the Pega 24.1 Platform to provide financial protection for customers against unforeseen situations during their holiday vacations. This application will enable customers to input their insurance details, with the option for Customer Service Representatives (CSRs) to create policies on behalf of customers. The application will require the inclusion of supporting documents and will feature a managerial review process for document verification, which must be completed within a one-hour timeframe. The travel insurance policy will only be applicable if the combined cost of tickets and accommodations exceeds ₹500; otherwise, an auto-rejection rule will apply. Customers will have the ability to choose from existing insurance plans—Basic, Economy, or Premium. Future enhancements will allow for dynamic policy suggestions based on travel specifics like mode of transport, distance, number of travellers, and duration of stay, along with a comparison chart of all available plans.

The application will automatically apply a 10% discount to customers, in future to offer additional discounts for first-time users and the option to purchase a recurring discount for an additional fee of ₹299, which would apply for a year. Customers will be shown the total payable amount, along with comprehensive policy details, including the validity period, which aligns with their holiday dates. The application will also allow customers to exit the booking process mid-way, requiring them to provide a reason for doing so. Furthermore, it includes the option for customers to cancel their policy before the journey start date, in which case a full refund will be processed after deducting a standard processing fee of ₹100. Finally, a PDF containing all policy details and validity will be sent to the customer, completing the process.

Along with that, I had 3 tasks to complete. The tasks were named as Pega Express Delivery, Low-Code App Builder and Low-Code App Builder Extended. The Pega Express Delivery mission page outlines a structured approach for efficiently delivering Pega projects. It highlights the Pega Express Methodology, which is designed to accelerate project timelines and ensure successful outcomes through best practices and streamlined processes. The methodology emphasizes a collaborative, iterative approach to project delivery, focusing on rapid development, continuous feedback, and adaptive planning to meet client needs effectively. By adhering to these practices, teams can optimize their project execution, enhance product quality, and achieve faster time-to-value for their Pega solutions.

Low Code App Builder is designed to help you rapidly and easily develop powerful, working apps with little or no code, using Pega's visual tools and model-driven dev environment. It includes essential topics like case types and user interfaces, which are building blocks for establishing a solid foundation on the integration of data from different sources along with useful tips about creating intuitive apps that work well. The mission will be perfect to kick start with your objectives on how you can utilise Pega platform in building applications which are business centric, workflow oriented & improve users experience irrespective of technical background.

The Low Code App Builder Extended mission expands on the foundational skills taught in the "Low Code App Builder" course, providing a deeper dive into advanced features and functionalities of Pega's low-code platform. This mission is designed for users who want to enhance their application-building capabilities by exploring more complex topics such as creating and managing data models, configuring advanced decision-making processes, and implementing automation features to optimize workflows. It also covers how to build and customize reusable components, manage user roles and security, and extend application functionality using Pega's integration tools. By completing this mission, learners will gain the expertise needed to design more sophisticated, scalable, and efficient applications that meet complex business requirements, all while maintaining the low-code development approach.

**Technologies Learned**

1. **Low-Code Development:**

Low-code development is an approach of software development that involves little to no coding, thanks to its visual interfaces and drag-and-drop tools that work together to build applications. Pega's low-code platform uses model-driven development, wherein developers design applications by the configuration of components instead of writing complex code. This will, in turn, be helpful for business users and developers collaborating more effectively by rapid prototyping and deploying applications into production in order to solve business needs. Apart from that, maintaining an application is easier in low-code development, with fewer chances of errors, while time-to-market improves as iterative changes can be done without extended code changes.

**2. Pega PRPC Platform**:

PRPC by Pegasystems-fully known as Pega Rules Process Commander-is the very foundation of technology that makes up the Pega platform, which is for building scalable, robust, and complex business applications. PRPC offers a single environment to address end-to-end business processes by integrating various components such as case management, workflow automation, and data handling. It will then enable different business operations by making the rules-driven architecture dynamically respond to changes in business. PRPC also facilitates management over process workflow, creating reusable components, automatically implementing decision management, and managing data integration of various systems for end-to-end business transformation.

**3. Case Management:**

Pega case management is all about designing and managing 'cases' that are structured collections of tasks and activities representative of business processes. The case management capabilities of Pega support defining a case lifecycle from creation to resolution while ensuring adherence to business policies. It includes features like task assignments, SLAs, process automation, and capturing data. Pega offers a suite of components which enables developers to build case-based applications, where the workflow will automatically change according to the latest input, presents decisions to the end-user, and manages an exception in cases where systems may not be able to automate the process or decision logic. It serves to ensure structured automation of processes and achieves frictionless operational efficiency through better visibility of work in progress and outcomes.

**4. User Interface Design:**

The Pega platform is integrated with a set of tools designed for building user-friendly visually attractive interfaces - UIs. Design tools allow developers to create responsive layouts, consistent styles, and intuitive navigation flows-keeping in mind the needs of the end-user. Pega promotes a set of reusable UI components and templates that hold the best practices in usability and accessibility so that experiences are smooth on different types of devices. The platform supports dynamic and context-sensitive UI changes, allowing the apps to change according to user roles, preferences, and real-time inputs. It ensures ease in application and alignment with business objectives.

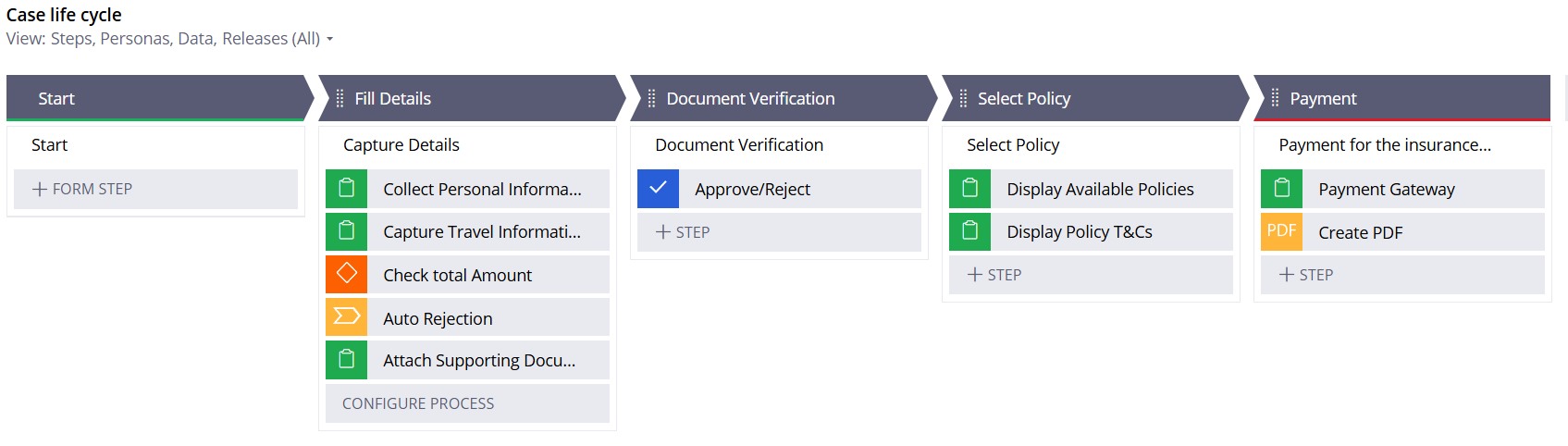
**5. Agile Methodology:**

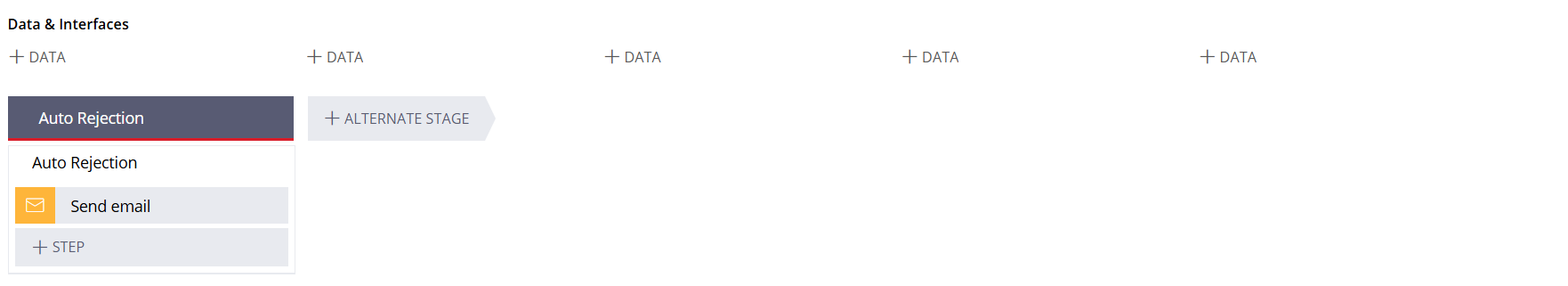
Agile methodology is an iterative approach to project management and software development that maintains continuous improvement, flexibility, and collaboration. As such, Agile principles in Pega projects include Iterative Development, Continuous Feedback, and Adaptive Planning, all put into place to further enhance efficiency in project delivery. This means that it breaks a project down into small increments-sprints-so a team can churn out the functionality more quickly and receive feedback from stakeholders on a regular basis. Practices like sprint planning, daily standups, and retrospectives keep the alignment among the team members, focusing on potential hiccups and adjusting toward the continuously changed requirements of the project. This approach entails continuous improvement of work culture for reduced time-to-market and increased project success rates.

**6. Business Rules Engine:**

Pega's Business Rules Engine is a powerful tool to define business rules and keep it run independently from the application logic. It makes use of multi-types of rules: decision tables, decision trees, and expressions to apply complex logics at business. BRE enables fast updating and changing of business policy with no heavy coding for quick response to changes in the market or to any regulatory requirement. Centrally managed, the rules become reusable across numerous applications. In addition, they can be updated easily with a user-friendly interface that reduces the chance for errors, making them very maintainable and flexible to keep the business logic current with organizational objectives.

**Design of the solution**

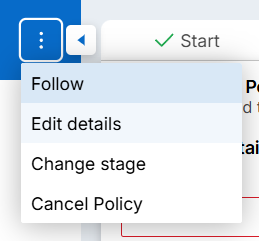
****

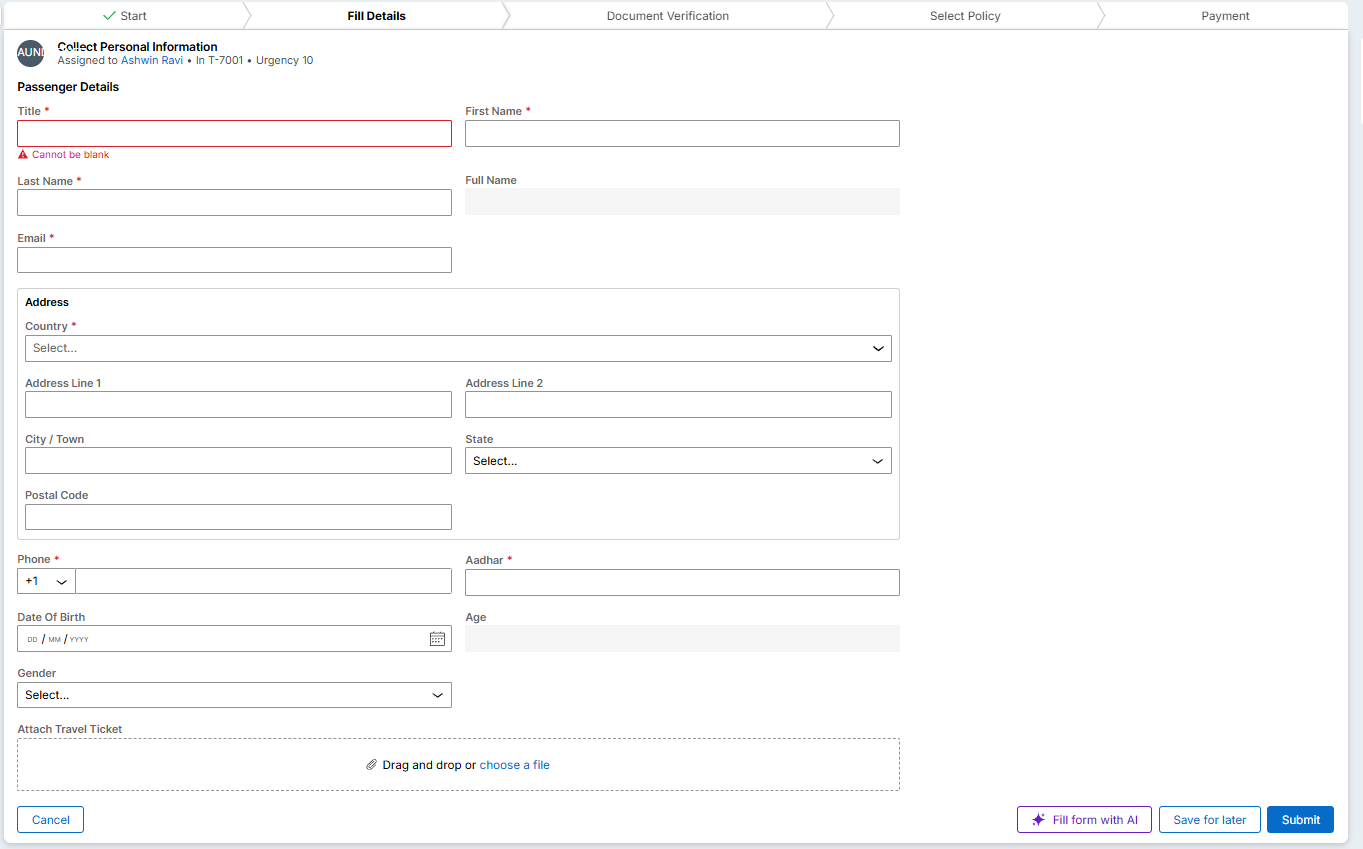


1. **Collect Customer Details:**
   * **Action**: Create a form in the Customer Portal to collect all necessary personal and travel details.
   * **Details to Collect**:
     + Personal information (Title, First Name, Last Name, Full Name).
     + Contact details (Address, Email, Phone).
     + Identification details (Aadhar/PAN/Passport).
     + Travel details (Source & Destination, Date of travel, Accommodation details).
     + Passenger details (Full name, Gender, Date of birth).
     + Supporting documents (Ticket, Accommodation proof).
   * **Explanation**: Ensure all the required details are captured correctly and validations like length checks on identification fields are in place to prevent data entry errors.
2. **Document Verification:**
   * **Action**: Set up a process in the Manager Portal for document verification.
   * **Details to Include**:
     + Allow the manager to review the documents submitted by customers.
     + Set a service level agreement (SLA) to complete the document verification within 1 hour.
     + Notify the manager of any breach of timelines.
     + Update customers about the approval or rejection status.
   * **Explanation**: This step ensures that all documents are verified quickly, adhering to regulatory requirements and providing timely responses to customers.
3. **Select Insurance Policy:**
   * **Action**: Display available insurance plans (Basic, Economy, Premium) to the customer.
   * **Details to Include**:
     + Show the cost and benefits of each plan.
     + Allow the customer to select one of the available options.
   * **Explanation**: This enables customers to choose the policy that best fits their needs. Future enhancements can dynamically display policies based on inputs like travel mode, distance, etc.
4. **Review and Confirm Details:**
   * **Action**: Provide a review screen summarizing all the collected information.
   * **Details to Include**:
     + Display the premium amount, discount, taxes, and the final payable amount.
     + Show the policy details, terms, and conditions.
     + Provide options to edit, cancel, or proceed with the booking.
   * **Explanation**: Ensuring that the customer reviews all details before finalizing helps avoid errors and confirms the customer's intent.
5. **Payment Collection:**
   * **Action**: Set up a payment gateway to collect payment details and process the transaction.
   * **Details to Include**:
     + Capture card details securely.
     + Verify and confirm payment once policy terms and conditions are accepted by the customer.
   * **Explanation**: This step finalizes the transaction, ensuring secure payment handling and compliance with financial regulations.
6. **Policy Allotment:**
   * **Action**: Generate and send a PDF document containing policy details.
   * **Details to Include**:
     + Policy ID, start date, and end date.
     + Name of the customer, passenger names, coverage amount, and validity.
     + Terms and conditions.
   * **Explanation**: This provides a documented proof of policy purchase to the customer, which can be referenced or used for claims.
7. **Policy Cancellation:**
   * **Action**: Enable customers to cancel the policy through the Customer Portal.
   * **Details to Include**:
     + Allow cancellation any time before the journey start date.
     + Process a full refund with a standard deduction of 100 currency units.
     + Capture the reason for cancellation.
   * **Explanation**: Providing an easy cancellation option ensures customer satisfaction and trust. Documenting reasons helps in future process improvements.
8. **UI/UX Design:**
   * **Action**: Design a user-friendly interface with consistent branding.
   * **Details to Include**:
     + Clear instructions and status updates on every page.
     + Ensure accessibility and responsiveness for various devices.
   * **Explanation**: A well-designed UI/UX improves customer engagement and reduces abandonment rates.
9. **Reporting Requirements:**
   * **Action**: Create a reporting dashboard in the Manager Portal.
   * **Details to Include**:
     + Show the number of policies issued in the last week.
     + Display the final status of each policy.
   * **Explanation**: This helps managers track performance and make informed decisions.
10. **Future Enhancements:**

* **Dev Studio Tasks**:
  + Implement dynamic policy display based on user inputs.
  + Add a comparison chart for all plans.
  + Develop a discount code mechanism for loyal customers.
  + Integrate a wallet feature for handling payments and refunds.
* **Explanation**: These enhancements add more functionality and flexibility to the application, improving customer experience and operational efficiency.

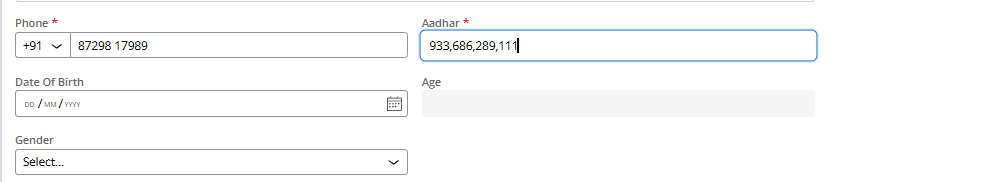
**Implementation**

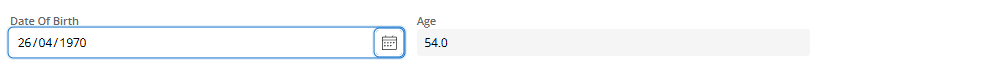
****

****

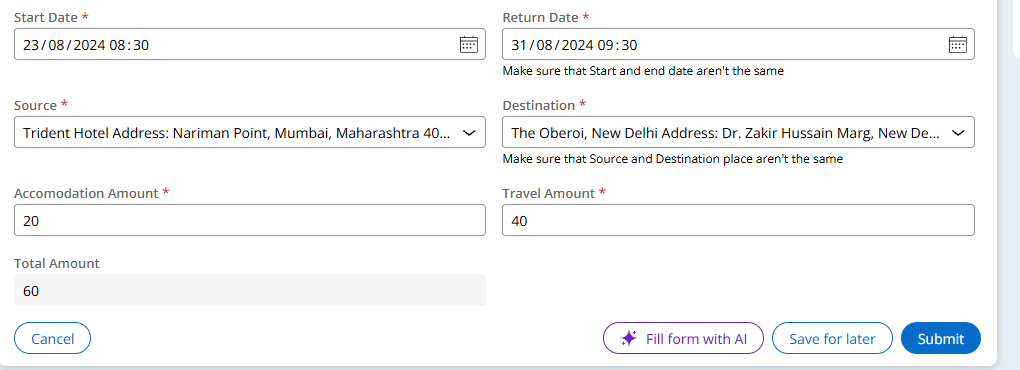
****

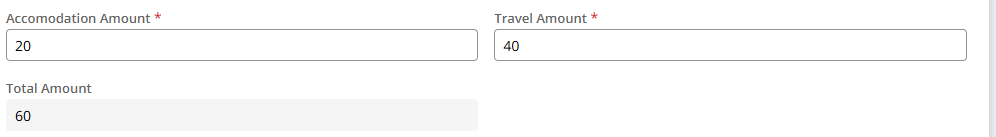
As the aadhar card format is wrong, it gives a prompt saying it is of the wrong type. In this case Aadhar format is wrong so it gives an alert to the user saying it is wrong.



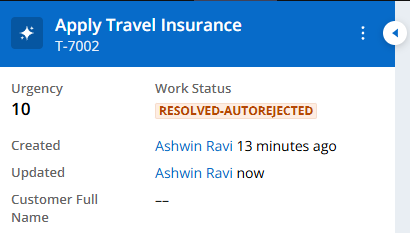


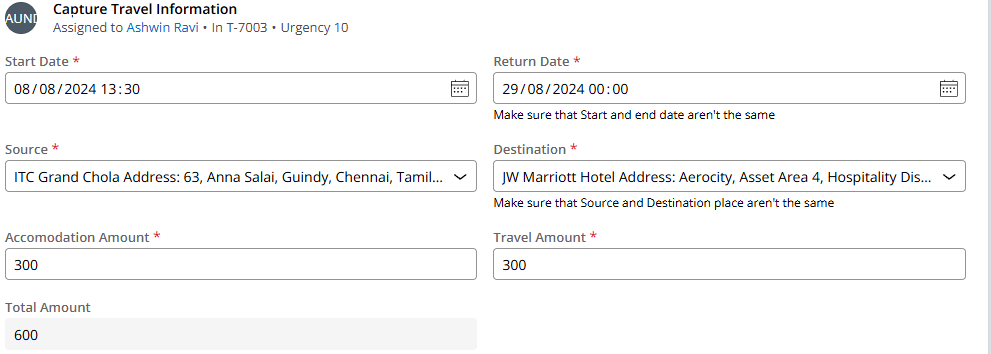
Age is automatically calculated from the DOB.



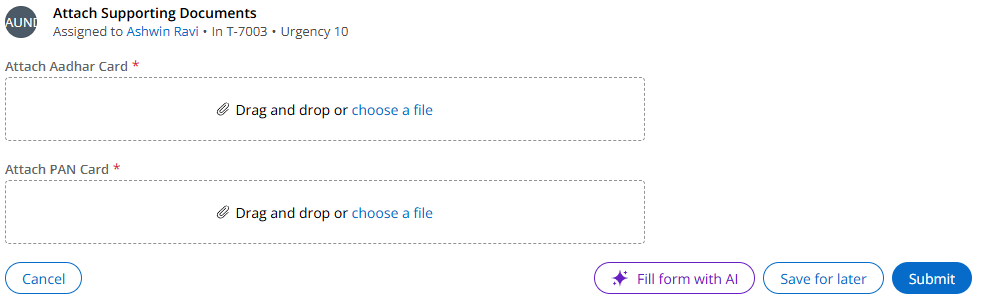


As the total amount is less than 500, it will automatically reject.

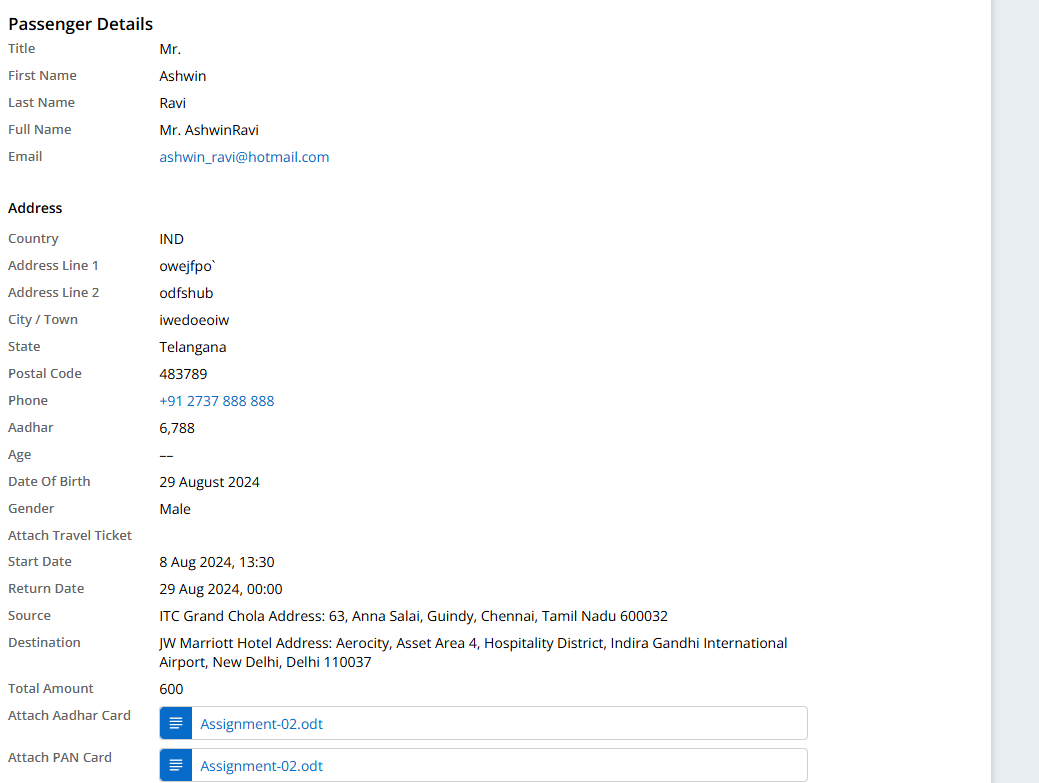




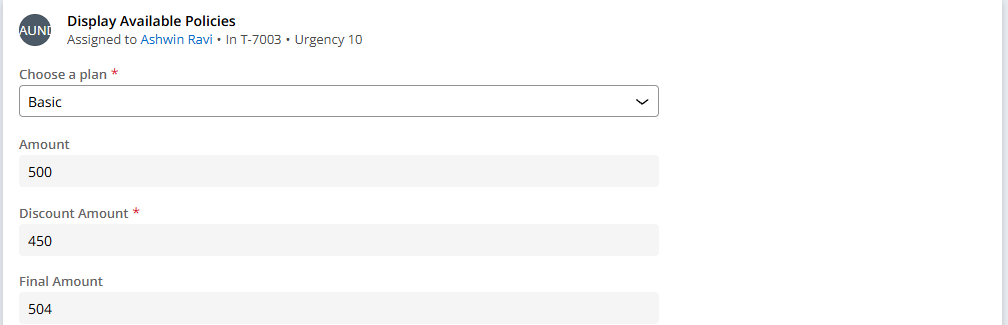
Attach supporting documents

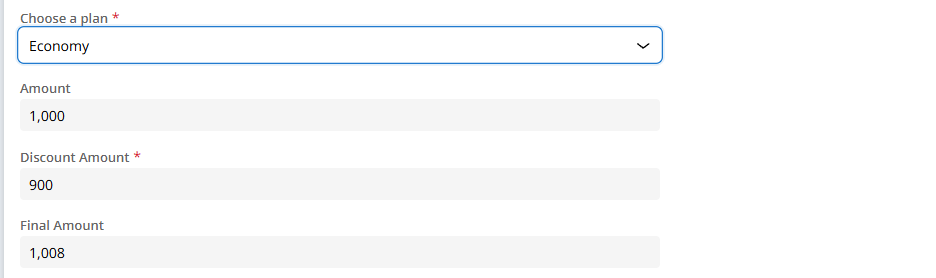


Then all the details in a consolidated way will be sent to the manager(Persona). And then manager gets to choose to accept/reject accordingly.



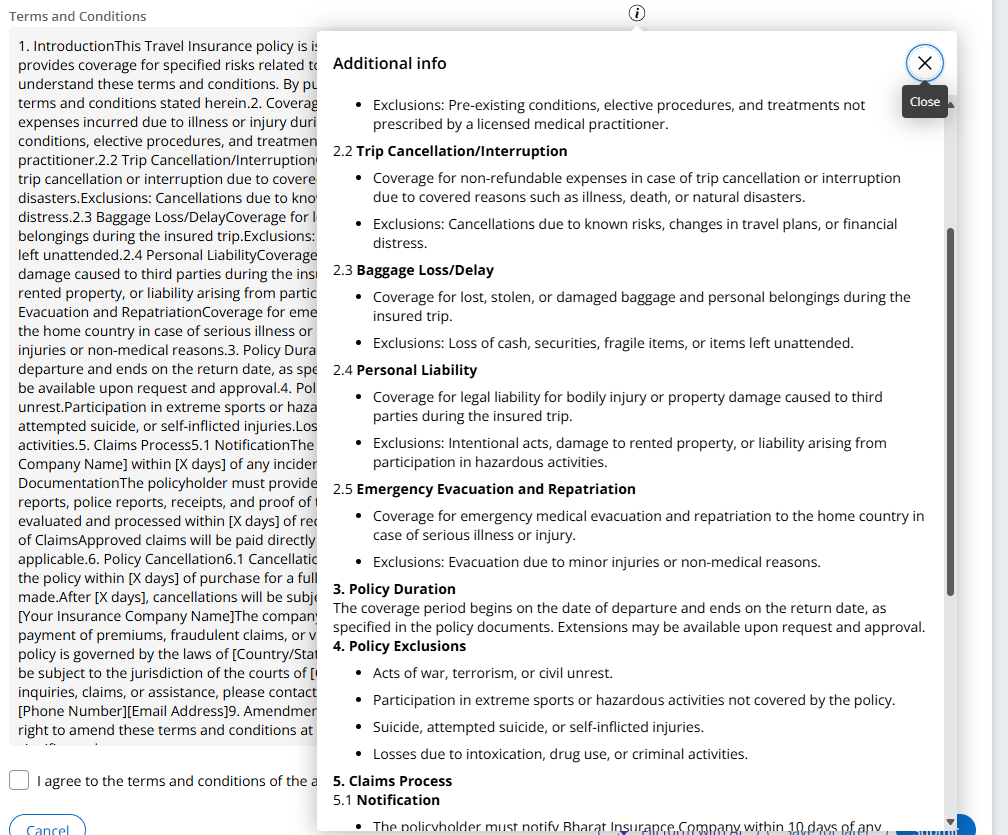
After that there will be an option for the user to choose a plan for the Insurance.



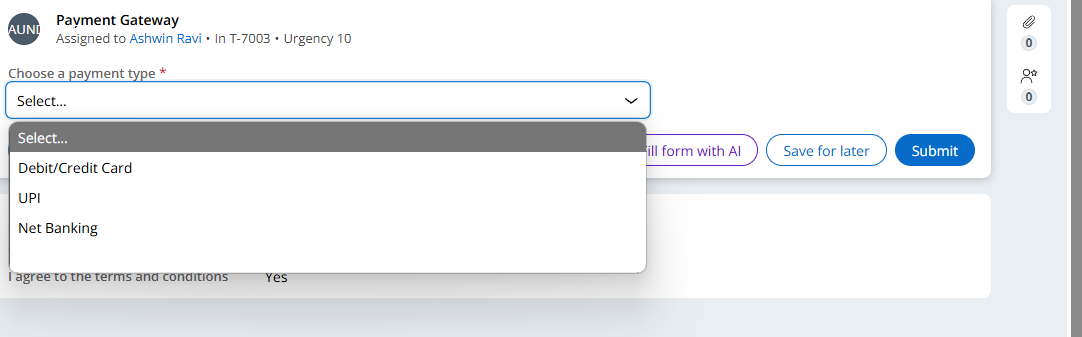




Then after that accept the T&C.

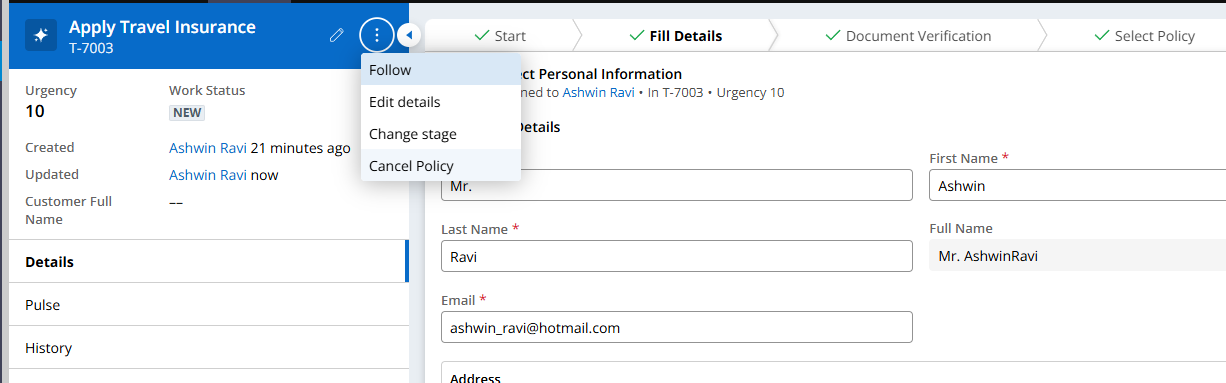


Then choose payment time.

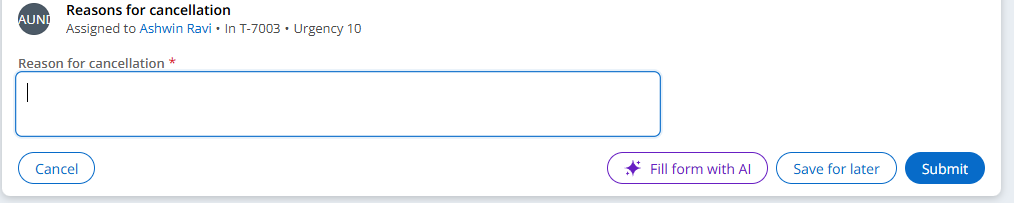


In that choose one and enter the details.

After that, PDF will be generated with all the details. If user wishes to cancel anywhere midway they can cancel.



Here select cancel policy.



After this the user will be getting an PDF where it contains a link to pay Rs. 110 for the cancellation charges.

This is how the application works.

The age extraction from DOB , Aadhar card verification, auto reject total cost below 500 are preset by writing regex , java code and operations for it.

**Best Practices**

* **Identify Business Needs**: Collaborate with stakeholders to capture and document the detailed requirements.
* **Consistent and Accessible UI/UX Design**: Ensure consistent and branded design, responsive to all types of users.
* **Data Validation and Security**: Enforce strong data validation; handle data securely and encrypt sensitive information.
* **Automated Workflows and SLAs**: Leverage automation capabilities and service level agreements provided by Pega for automating processes that reduce manual errors.
* **Centralized Business Rules Management**: Use Pega's Business Rules Engine for managing the rules at the center, which can be updated and reused easily.
* **Agile Development Practices**: Follow Agile development principles on iterated development, continuous feedback, and regular sprint cycles to ensure that the software delivers what business needs on time with agreed-upon quality.
* **Role-Based Access Control**: Use RBAC to ensure controlled and secured access to functionalities.
* **Regular Security Audits**: Regular security testing and auditing have to be conducted to detect potential vulnerabilities and mitigate them.
* **Full Documentation**: Detailed documentation has to be done related to architecture, process, and configurations of the application.
* **Performance Optimization**: The optimum tuning of the workflows and performing of load testing to maintain high performance while scaling up the application.
* **Continuous Integration and Deployment**: In project development, use continuous integration tools for frequent testing and rapid deployment of updates.
* **User Feedback Mechanisms**: Channels should be created for users to give their feedback for the identification of issues in enhancing application functionality.
* **Monitoring and Reporting**: Monitoring tools should be put in place to observe performances and usages for proactive improvements.
* **Regular Security Audits**: Conduct regular security audits and testing to identify and mitigate potential vulnerabilities.

**Conclusion**

By using the Pega 24.1 Platform, the development of Holiday Travel Insurance showed us the capacity of Low-Code Development in providing robust applications highly scalable towards user-friendly Business Solution. Features present in Pega by means of rules-driven architecture, automation of workflows, and agile developmental practices were able to implement an application fully satisfactory to the needs of the client. The project has also made us induce additional functionalities into one: real-time decision-making, data validation, and secure payment processing. In addition, the UI/UX experience underlined the role that continuous feedback, collaboration, and user-centricity play in creating quality applications. Despite all that has been done, the future enhancements proposed will go towards fine-tuning the application further to keep it adaptable and usable for the emerging business needs. The overall internship helped in garnering quite a lot about the Pega platform on how it applies to the real business scenario and importantly, how an organization works.

**References**

1. <https://academy.pega.com/mission/pega-express-delivery/v1>
2. <https://academy.pega.com/mission/low-code-app-builder/v6>
3. <https://academy.pega.com/mission/low-code-app-builder-extended/v6>
4. [how to calculate age when date of birth is given? | Support Center (pega.com)](https://support.pega.com/question/how-calculate-age-when-date-birth-given?)
5. <https://support.pega.com/support-doc/pdf-document-generation-patterns>?
6. <https://support.pega.com/question/paragraph-or-text-block-constellation>?
7. <https://support.pega.com/question/how-solve-expected-screenflow-assignment-label-be-create-select-offer-identify-transfer>
8. [Calculate Age from Date of Birth in Pega￼￼ - Pega Tech (pegateck.com)](https://pegateck.com/age-calculation-from-date-of-birth-in-pega/)

Link to the project:- <https://github.com/SolitudeAsh/Holiday-Travel-Insurance>