

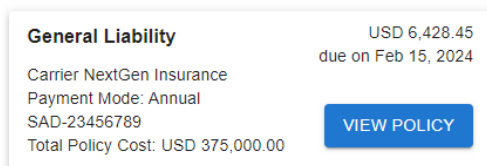
Functional Requirements:

1. Customer Details:

- Display customer information:

Sample

- Customer name
 - Email
 - Company name
- ### 2. Add Submission Inquiry function.
- Users can attach files and include explanations.
- ### 3. Requests for Quote:
- User can request a quotation to a certain product
- ### 4. About us
- Provide a button to easily contact the company.
 - Include various contact methods such as email and phone.
 - Info about PBIBINC
- ### 5. Payments:
- Facilitate payments. If Joe can do it on their program.
 - Payment status tracking
 - Users can select between seeking assistance from us or directly contacting finance.
- ### 6. Dashboard:
- Display All policies, using card with CTA.



7. Including status info of policies

- Intent:
 - User can see the intent information
- Renewal:
 - Track renewal statuses
- Pending Approval:
 - Display pending approval statuses.
- Cancellation Request Status:
 - Approved
 - Reviewed
 - Pending
 - Process
 - Waiting for Approval
 - Completed

8. DIY Certificates

Proof of Insurance:

- Allow users to edit personal information.
- Allow users to preview POI after submission of updated information.
- Once the user has already reviewed the document and confirms that all the information is correct, they can send it to their own email.

Specialized Certificates:

- Users can request certificates with specialized certifications.
- Allow users to preview certificate after the customer service representative approved the document.

9. Notifications

- All various notification such as statuses, reminders, update etc.

10. Technical Requirements:

Platforms:

- The application should be developed for both iOS and Android platforms.

User Interface:

- Design a user friendly and intuitive interface.
- Ensure easy navigation through the customer portal, payments, and notifications sections.

Security:

- Implement robust security measures to protect customer data.
- Encryption of sensitive data
- Ensure secure transactions for payments. (Possible to have payment)
- User privacy.

Performance:

- Optimize the application for fast loading times and smooth performance.
- Ensure the application works seamlessly on various devices and screen sizes.

Notifications:

- Implement push notifications to alert customers of important updates, renewals, and statuses.

Nonfunctional Requirements:

- Usability:
 - The application should be easy to use and accessible for all users.
- Provide clear instructions and tooltips where necessary.

Reliability:

- Ensure the application is reliable and functions correctly without frequent crashes or bugs.

Compatibility:

- Ensure compatibility with the latest versions of iOS and Android operating systems.

Maintainability:

- Design the application to be easily maintainable and updatable.

Development and Testing:

Development:

- Use agile development methodologies to ensure timely delivery and continuous improvement.
- Conduct regular code reviews and testing.

Testing:

- Perform thorough testing including unit testing, integration testing, and user acceptance testing.
- Ensure the application is free from critical bugs before release.

Deployment:

- Deploy the application to the respective app stores (Apple App Store and Google Play Store).
- Provide clear instructions for downloading and installing the application.

Feedback Mechanism

- In-app feedback form
- Regular Surveys

Documentation:

- Provide comprehensive documentation for both users and developers.
- Include user manuals, API documentation, and troubleshooting guides.