App Specification Document

1. **Introduction**

   - This App is meant to be used by GB Advisors clients to create, view and reply to tickets in our instance from the client's Freshdesk Account.
   - The purpose of this application is to make communication with GB Advisors clients more efficient so they don't have to leave Freshdesk and write an email to contact our support team, they can do it right from their Freshdesk account.

2. **Features**

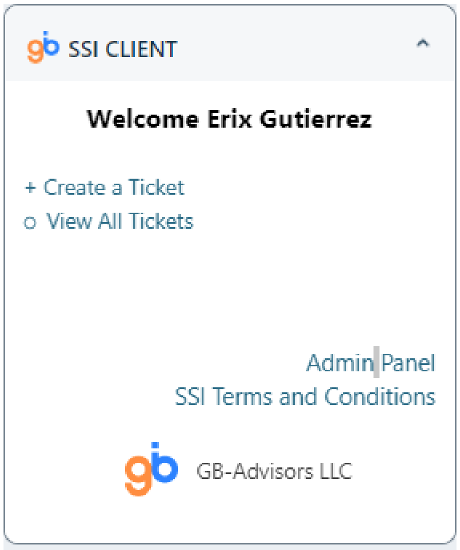
   - List of features in the application.
     - View open tickets.
     - Create new tickets.
     - Reply to existing tickets.
     - Manage users that are allowed to use this application.
   - Description of each feature.
     - Ticket Creation: Clients can create new tickets in the system. Each ticket includes details like a title, description, and a category or priority level.
     - Ticket Viewing: Clients can view their existing tickets. This includes not just the original details entered during ticket creation but also any updates or changes to the ticket status. Clients can see if their ticket has been assigned to a support agent, if it's been resolved, or if there are any updates or replies from the support team.
     - Ticket Replies: Clients can reply to their tickets. This allows for a back-and-forth conversation between the client and the support team. Clients can provide additional information, answer questions from the support team, or respond to proposed solutions.
     - User Management: The application includes functionality for the app administrator to manage the user accounts and decide who inside the Fresh platform can create tickets.
     - Security: The application ensures that only authorized users can create, view, and reply to tickets. This involves user authentication and also role-based access control.

3. **User Interface**

   - Description of the user interface.

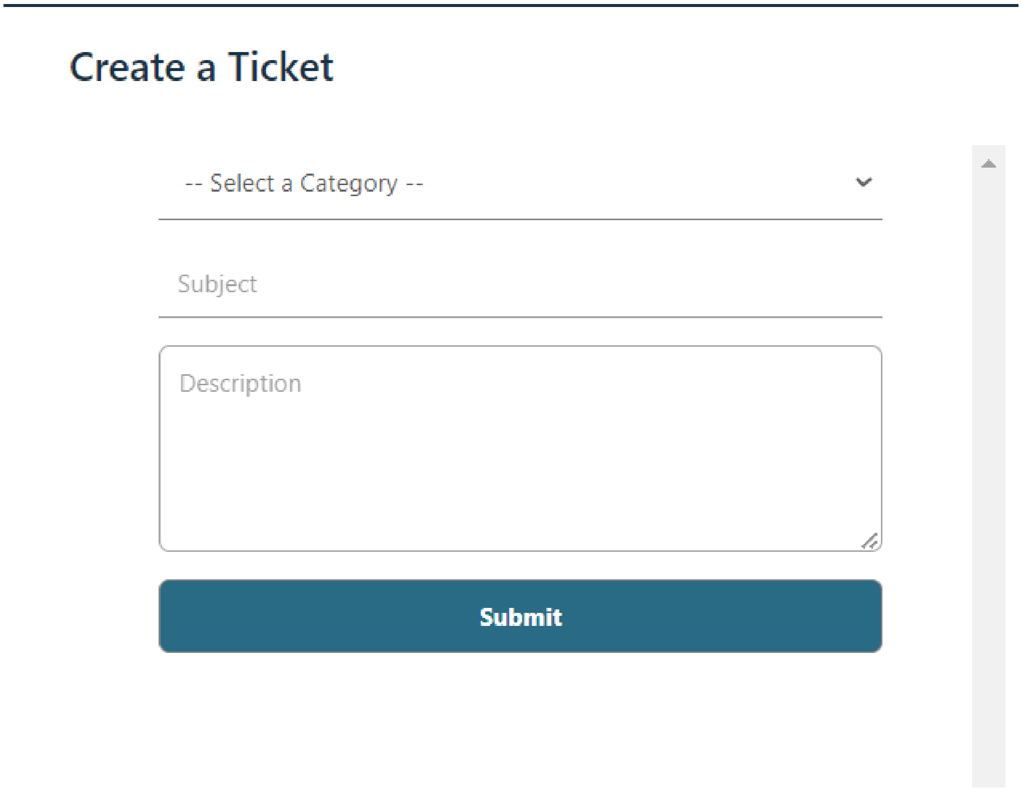
     It's a simple UI with 4 links that each open up their respective Modal, each Modal has specific information for example the View Tickets Modal has a list of all the tickets created, the Create Ticket Modal has a form that the user can fill to create a new ticket and so on.

- Screenshots or mockups of the interface.
  - Home Page



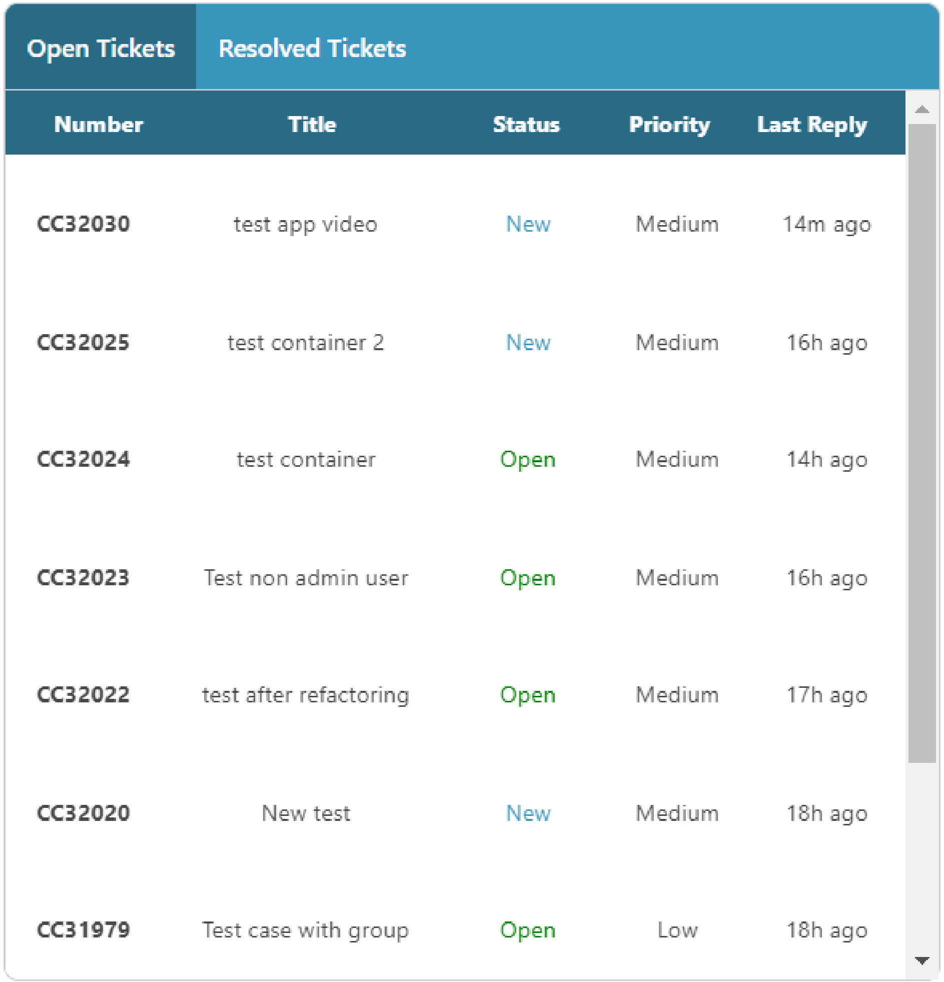
Welcome Erix Gutierrez

  - Create Ticket Modal



Create a Ticket

- View Tickets Modal



Number	Title	Status	Priority	Last Reply
CC32030	test app video	New	Medium	14m ago
CC32025	test container 2	New	Medium	16h ago
CC32024	test container	Open	Medium	14h ago
CC32023	Test non admin user	Open	Medium	16h ago
CC32022	test after refactoring	Open	Medium	17h ago
CC32020	New test	New	Medium	18h ago
CC31979	Test case with group	Open	Low	18h ago

4. **User Interaction**

   ○ Description of how the user interacts with the application.

   - Ticket Creation: Clients can create new tickets in the system. Each ticket includes details like a title, description, and a category or priority level.
   - Ticket Viewing: Clients can view their existing tickets. This includes not just the original details entered during ticket creation but also any updates or changes to the ticket status. Clients can see if their ticket has been assigned to a support agent, if it's been resolved, or if there are any updates or replies from the support team.
   - Ticket Replies: Clients can reply to their tickets. This allows for a back-and-forth conversation between the client and the support team. Clients can provide

additional information, answer questions from the support team, or respond to proposed solutions.

5. **Technical Specifications**

    - Description of the technologies used in the application.

      This App was created using:

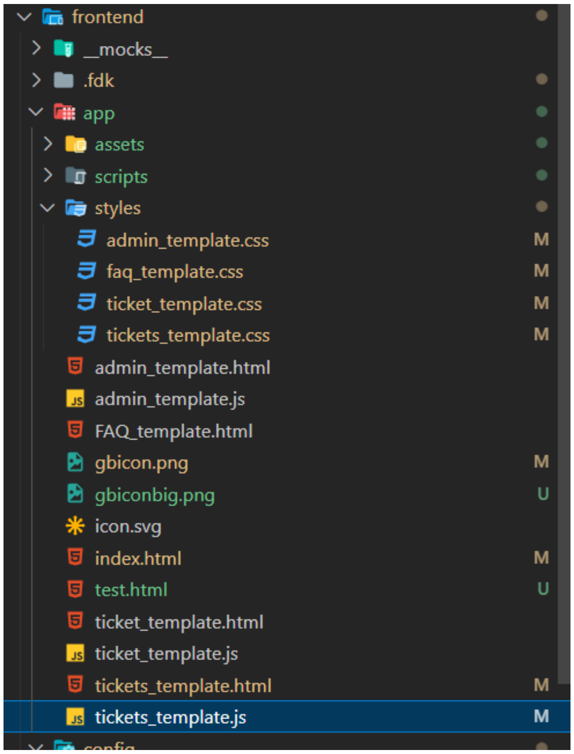
        - FDK CLI
        - HTML
        - CSS
        - JavaScript
        - React
        - Request Method (From Freshdesk)
        - IParams

    - Explanation of how these technologies are used.

        - HTML CSS JavaScript are the technologies used in the core of this application and all of the Modals are written in VanillaJS.
        - React to build the Home page.
        - FDK CLI to create, test and build this application.
        - Request Method to do all of the API calls and IParams to set the required information from the client.

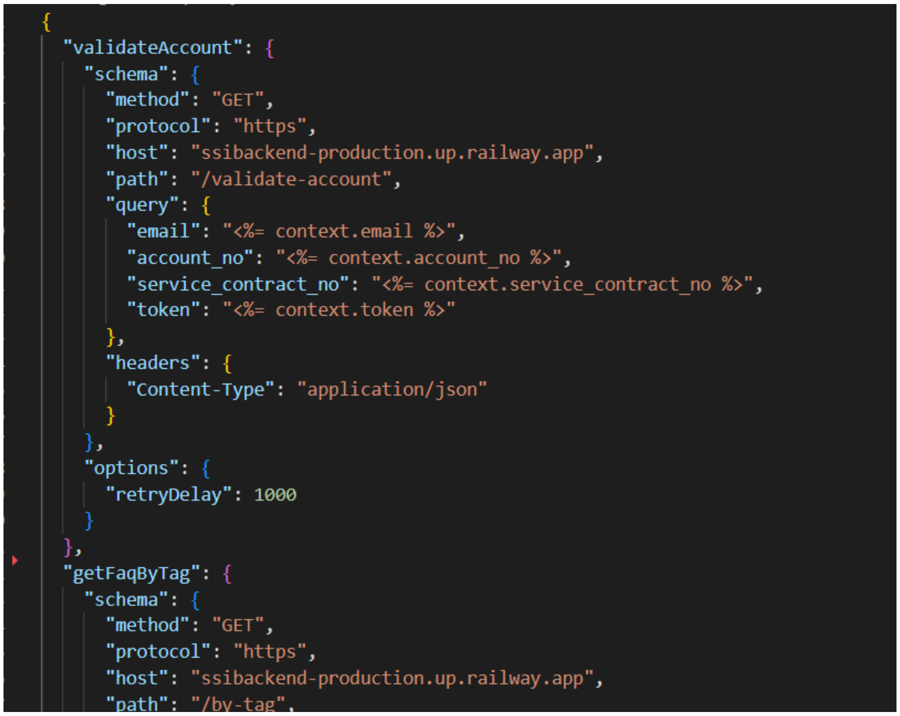
6. **Code Snippets**

- Important code snippets from the application.



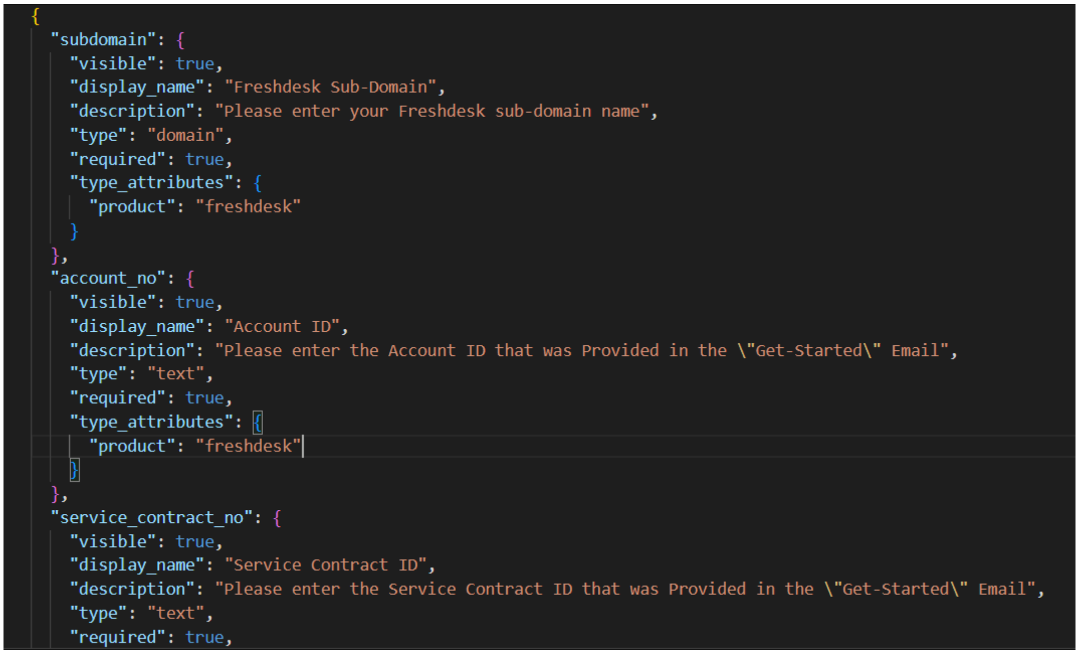
```
frontend
  __mocks__
  .fdk
  app
    assets
    scripts
    styles
      admin_template.css      M
      faq_template.css        M
      ticket_template.css     M
      tickets_template.css    M
    admin_template.html
    admin_template.js
    FAQ_template.html
    gbicon.png                M
    gbiconbig.png             U
    icon.svg
    index.html                M
    test.html                 U
    ticket_template.html
    ticket_template.js
    tickets_template.html     M
    tickets_template.js       M
  config
```

This is where all the Modals are created in the app folder



```
{
  "validateAccount": {
    "schema": {
      "method": "GET",
      "protocol": "https",
      "host": "ssibackend-production.up.railway.app",
      "path": "/validate-account",
      "query": {
        "email": "<%= context.email %>",
        "account_no": "<%= context.account_no %>",
        "service_contract_no": "<%= context.service_contract_no %>",
        "token": "<%= context.token %>"
      },
      "headers": {
        "Content-Type": "application/json"
      }
    },
    "options": {
      "retryDelay": 1000
    }
  },
  "getFaqByTag": {
    "schema": {
      "method": "GET",
      "protocol": "https",
      "host": "ssibackend-production.up.railway.app",
      "path": "/by-tag",
```

This is a sample of the requests.json file.



```
{
  "subdomain": {
    "visible": true,
    "display_name": "Freshdesk Sub-Domain",
    "description": "Please enter your Freshdesk sub-domain name",
    "type": "domain",
    "required": true,
    "type_attributes": {
      "product": "freshdesk"
    }
  },
  "account_no": {
    "visible": true,
    "display_name": "Account ID",
    "description": "Please enter the Account ID that was Provided in the \"Get-Started\" Email",
    "type": "text",
    "required": true,
    "type_attributes": {
      "product": "freshdesk"
    }
  },
  "service_contract_no": {
    "visible": true,
    "display_name": "Service Contract ID",
    "description": "Please enter the Service Contract ID that was Provided in the \"Get-Started\" Email",
    "type": "text",
    "required": true,
```

Sample of the iparams.json File.

7. Future Improvements

- List of potential future improvements for the application.
    - We may be adding a Survey screen soon so that clients can reply to closed tickets surveys from Freshdesk.