**Project Setup and Running Instructions**

**Prerequisites**

* Node.js and npm installed on your machine.
* MongoDB Atlas account for the database.
* Google Cloud Platform account for OAuth credentials.
* Intercom account for customer service integration.

**Step-by-Step Setup**

1. **Clone the Repository**

git clone https://github.com/your-username/customer-service-platform.git

cd customer-service-platform

1. **Backend Setup**
   1. Navigate to the backend directory:

cd backend

* 1. Install dependencies:

npm install

* 1. Create a .env file in the backend directory with the following content:

GOOGLE\_CLIENT\_ID=1051288013989-1g5o74qgqlsbslsafjnjk26chvl6m22o.apps.googleusercontent.com

GOOGLE\_CLIENT\_SECRET=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

INTERCOM\_ACCESS\_TOKEN=dG9rOjI4OTUwYjEwXzBjOWVfNDkzM185YWZhXzUxMTE4NjMzOTI1ZDoxOjA=

INTERCOM\_APP\_ID=e8w2cqnd

INTERCOM\_CLIENT\_ID=6355ae36-e762-4084-895a-0a8a282d0b9e

INTERCOM\_CLIENT\_SECRET=6355ae36-e762-4084-895a-0a8a282d0b9e

MONGODB\_URI=mongodb+srv://yeletinandureddy:g20PKvWUj0bvIH38@cluster0.nulhaoc.mongodb.net/?retryWrites=true&w=majority&appName=Cluster0

SESSION\_SECRET=4f7a60c15d3ea3b1a7e9324fc7c9cb2c

* 1. Start the backend server:

node src/app.js

1. **Frontend Setup**
   1. Navigate to the frontend directory:

cd ../frontend

* 1. Install dependencies:

npm install

* 1. Create a .env file in the frontend directory with the following content:

GOOGLE\_CLIENT\_ID=1051288013989-1g5o74qgqlsbslsafjnjk26chvl6m22o.apps.googleusercontent.com

INTERCOM\_APP\_ID=e8w2cqnd

* 1. Start the frontend server:

npm start

**End-to-End Flow Demonstration**

**1. Google OAuth Login**

* Navigate to the frontend URL (usually http://localhost:3000).
* Click on the "Login with Google" button.
* Sign in with your Google account.
* After successful authentication, you will be redirected to the home page of the application.

**2. Customer Service Interaction**

**2.1 Submit a Customer Service Request:**

On the home page, you will see a form to submit customer service requests.

Select a category (General Queries, Product Features Queries, Product Pricing Queries, or Product Feature Implementation Requests).

Enter additional comments and submit the form.

The request will be sent to the backend and stored in the database.

**2.2. View Submitted Requests:**

After submitting a request, you can see a list of submitted requests categorized accordingly.

The frontend will fetch and display these requests from the backend.

**3. Intercom Integration**

The Intercom widget will appear on the frontend.

You can use this widget to interact with customer service representatives.

The submitted requests will also be sent to Intercom using their API.

**Key Files and Directories**

**Backend:**

* backend/src/app.js: Main application file, initializes the server.
* backend/src/routes/requests.js: API endpoints for customer service requests.
* backend/src/routes/auth.js: API endpoints for Google OAuth authentication.
* backend/src/config/intercomConfig.js: Configuration for Intercom client.
* backend/src/models/Request.js: Mongoose schema and model for requests.

**Frontend:**

* frontend/src/components/LoginForm.jsx: Component for Google OAuth login.
* frontend/src/components/RequestForm.jsx: Component for submitting customer service requests.
* frontend/src/components/RequestList.jsx: Component for displaying submitted requests.
* frontend/src/App.js: Main React application file, contains routing and main layout.

By following these instructions, you should be able to set up, run, and demonstrate the complete flow of the customer service platform application, including Google OAuth login and interaction with Intercom.com.