Customer Feedback Tracker

" Custrack "

Customer Feedback Tracker - *Custrack*

A tool to collect, categorize, and track customer feedback (complaints, suggestions, compliments). Admins can manage feedback status and generate basic reports to identify trends and areas for improvement in customer service.

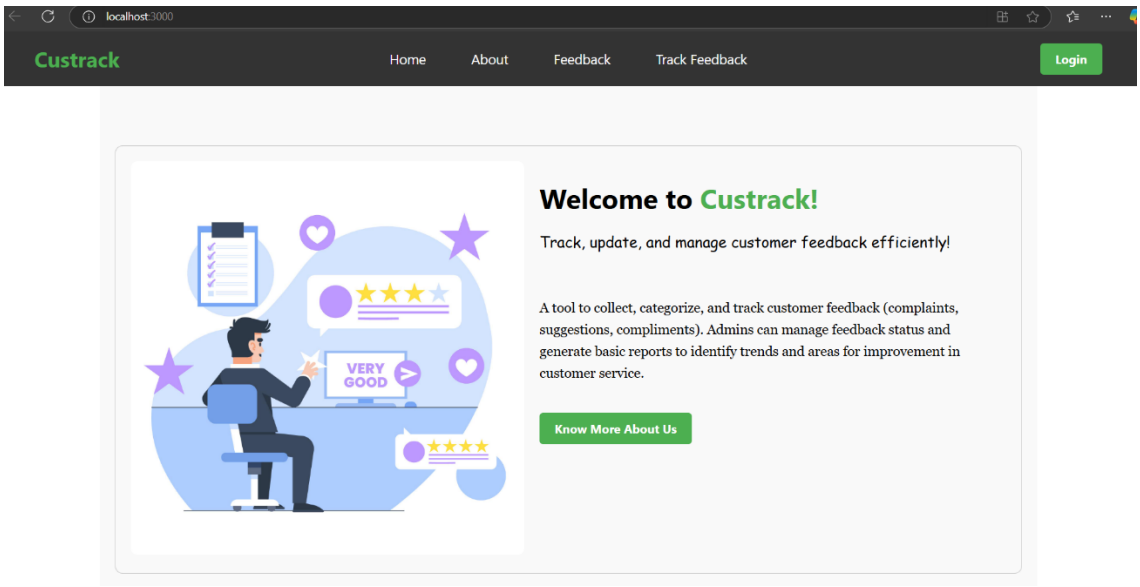
Project Overview

Custrack is a web application developed to streamline the process of collecting, categorizing, and managing customer feedback. The platform allows customers to submit different types of feedback, such as complaints, suggestions, and compliments, providing businesses with valuable insights into customer satisfaction and areas for improvement.

The application is designed to facilitate seamless interactions between customers and admin users. It provides a user-friendly interface for customers to submit their feedback and track its progress, while also offering powerful administrative controls to efficiently manage and analyze the feedback.

Key features of **Custrack** include:

- **Customer Feedback Submission:** Customers can easily submit complaints, suggestions, or compliments, which are categorized for appropriate action.
- **Feedback Tracking:** Each submission is assigned a unique track ID, allowing customers to monitor the status of their feedback.
- **Admin Dashboard**: Admins can manage, update, and categorize feedback, ensuring timely responses and resolutions.
- **Reporting and Analytics:** Admins can generate reports to identify trends and areas for improvement, helping drive data-informed decisions.



Custrack aims to improve communication between businesses and customers, offering an efficient system for managing feedback and enhancing customer service. By providing both customers and admins with real-time updates and powerful management tools, Custrack enhances transparency, responsiveness, and accountability in customer relations.

Technology Stack

- **Frontend:** React.js
- **Backend:** Node.js with Express
- **Database:** MongoDB
- **Authentication:** JSON Web Tokens (JWT) for secure login and session management
- **Styling:** CSS
- **Additional Libraries:** React Router (for routing), Axios (for API calls), etc.

Custrack utilizes **React.js** for the frontend, enabling the development of a dynamic, responsive, and user-friendly interface. The **backend** is powered by **Node.js** with **Express**, which ensures the server is fast, scalable, and efficient in handling API requests. **MongoDB**, a NoSQL database, is used to store customer feedback and related data, providing flexibility and scalability.

For **secure authentication**, **JSON Web Tokens (JWT)** manage user sessions and login security. The **CSS** framework is applied to achieve a modern and clean design. **React Router** facilitates seamless navigation across the app, while **Axios** enables smooth communication with the backend through asynchronous API calls. These technologies ensure the Custrack application is not only efficient but also secure, scalable, and user-friendly.

Features

1. **Customer Feedback Submission:**

   Customers can easily submit complaints, suggestions, or compliments through a simple and intuitive form. Each submission is categorized for appropriate handling.

2. **Feedback Tracking:**

   After submission, customers receive a unique track ID, enabling them to monitor the status of their feedback in real time. This feature ensures transparency and keeps customers informed about their feedback's progress.

3. **Admin Dashboard:**

   The admin interface allows users to efficiently manage feedback by updating statuses, responding to customer inquiries, and categorizing issues based on priority. This ensures prompt and effective resolution of feedback.

4. **Reporting and Analytics:**

   Admins can generate detailed reports to analyze feedback trends and identify areas for improvement, helping drive data-driven decisions to enhance customer service and business performance.

These features combine to offer a seamless experience for both customers and admins, making Custrack an efficient tool for managing and improving customer feedback.

Implementation

The implementation of Custrack involves integrating the frontend, backend, and database components, ensuring smooth communication between them. Here's a detailed breakdown of how each module is implemented:

1. Frontend

- **Framework**:React.js
  The frontend of Custrack is developed using React.js, which enables the creation of a dynamic and responsive user interface.
- **Key Components**:
  - **HomeScreen**: Displays static and dynamic content, with links to other sections. Built using React functional components and styled with CSS.
  - **AboutScreen**: Implements a structured layout to inform users about Custrack's purpose and usage.
  - **FeedbackForm**:
    - Includes input fields for customer details, feedback category, and message.
    - Uses React's useState for form data and Axios for API calls to submit feedback to the backend.
  - **TrackFeedback**: Implements a search bar for Track ID and fetches feedback status via API. Displays results dynamically using React's state and lifecycle hooks.
  - **AdminDashboard**:
    - Displays feedback data fetched from the backend using Axios.
    - Allows filtering and updating of feedback statuses with React Router for navigation and state management.
  - **React Router**: Manages navigation between pages without reloading the app.
  - **Styling**: CSS is used to create a modern, user-friendly interface with responsive design principles.

2. Backend

- **Framework**: Node.js with Express
  - The backend provides RESTful APIs to handle frontend requests.
  - Routes and controllers are created for different functionalities:
    - /api/feedback for submitting and fetching feedback.
    - /api/track/:id for tracking feedback by its unique Track ID.
    - /api/admin for managing feedback and generating reports.

- Middleware functions are implemented to handle errors, validation, and authentication.

- **Key Features**:
  - **Feedback Submission**: Data is validated before storing in MongoDB.
  - **Feedback Management**: Admins can update feedback status and add replies.
  - **Authentication**: JWT ensures secure access to admin features, with tokens generated upon login.
  - **Reporting**: Feedback data is aggregated using MongoDB queries for generating reports.

3. Database

- **Database**: MongoDB
  - A NoSQL database is used to store feedback data.
  - **Schema Design**:
    - **Feedback Collection**: Includes fields for customer details, feedback message, category, unique Track ID, status, and timestamps.
    - **Admin Collection**: Stores admin credentials and access levels.

- **Sample Feedback Schema**:

```
const mongoose = require('mongoose');
const feedbackSchema = new mongoose.Schema({
  customerName: { type: String, required: true },
  email: { type: String, required: true },
  category: { type: String, enum: ['Complaint', 'Suggestion', 'Compliment'], required: true },
  message: { type: String, required: true },
  trackID: { type: String, unique: true, required: true },
  status: { type: String, enum: ['Pending', 'In Progress', 'Resolved'], default: 'Pending' },
  createdAt: { type: Date, default: Date.now },
});
module.exports = mongoose.model('Feedback', feedbackSchema);
```

4. API Integration:

Axios is used for making API calls to the backend for submitting feedback, retrieving track details, and managing feedback via the admin panel. Example API call for feedback submission:

```
const submitFeedback = async (feedbackData) => {
  try {
    const response = await axios.post('/api/feedback', feedbackData);
    alert(`Feedback submitted successfully! Track ID: ${response.data.trackId}`);
```

```
  } catch (error) {
    console.error('Error submitting feedback:', error);
  }
};
```

5. Authentication (JWT)

- Admins log in using credentials verified against a User collection in MongoDB. On successful login, a JWT is issued.
- Admin actions require this token for authentication, ensuring secure access to feedback management features.

6. Key Features Implementation

- **Feedback Submission**: Feedback is submitted through a form in the frontend. The backend validates and stores the feedback in the MongoDB database, generating a unique trackId for the customer.
- **Feedback Tracking**: Customers input their trackId in the TrackFeedback screen. The system fetches the corresponding feedback and displays its status and details.
- **Admin Management**: The admin dashboard enables viewing and filtering feedback. Updates to feedback status are sent via PUT requests to the backend.
- **Reporting**: Admins can view feedback trends by querying the database for categories and statuses.

Screens

The Custrack application includes several key screens for both customer and admin interactions. These screens facilitate feedback submission, tracking, and management, providing a streamlined process for addressing customer concerns and suggestions. The main screens include:

1. **AboutScreen**
2. **Admin Dashboard**
3. **FeedbackForm**
4. **HomeScreen**
5. **TrackFeedback**

About Each Screen and its Functions :

1. **AboutScreen:**

   The **AboutScreen** introduces users to Custrack and its purpose. It highlights the platform's goal to enhance customer service by efficiently managing feedback. This screen provides detailed sections such as:

   - **" About Custrack "**: An explanation of the platform's functionalities and benefits for customers and admins.

- **"What We Do"**: An overview of the core services offered, such as categorizing feedback, tracking its progress, and generating reports.
- **"Get Started"**: A call-to-action section guiding users on how to begin using the platform, either by submitting feedback or accessing the admin panel.

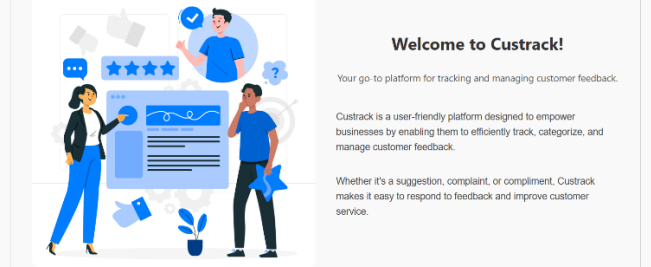


Figure - About Custrack

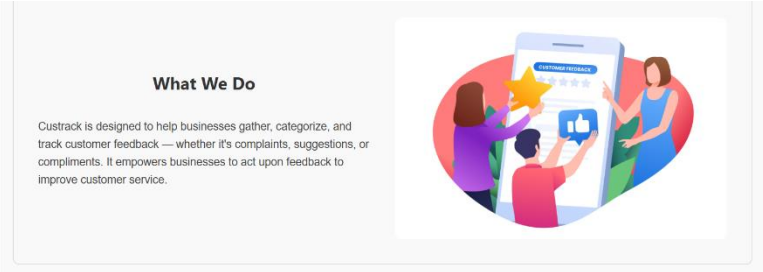


Figure – What we do

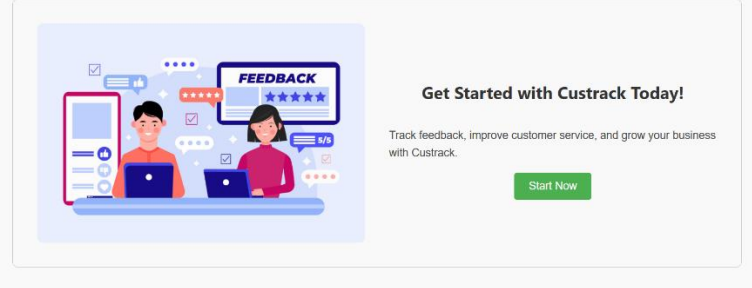


Figure – Get Started

2. **Admin Dashboard:**

The **Admin Dashboard** is the comprehensive control panel for administrators, featuring the following functionalities:

- **View All Feedback**: Admins can see a list of all feedback submissions, categorized as complaints, suggestions, or compliments.
- **Filter Options**: Filters allow admins to sort feedback by categories (e.g., complaint, suggestion), status (e.g., pending, resolved), and priority.

- **Manage Feedback**: Admins can update the status of feedback, mark issues as resolved, or assign them for further action.
- **Reply Functionality**: Directly respond to customers to acknowledge feedback or provide updates.
- **Reports Section**: Generate and view graphical or tabular reports to identify trends, common issues, and areas for improvement. This feature supports decision-making based on data insights.

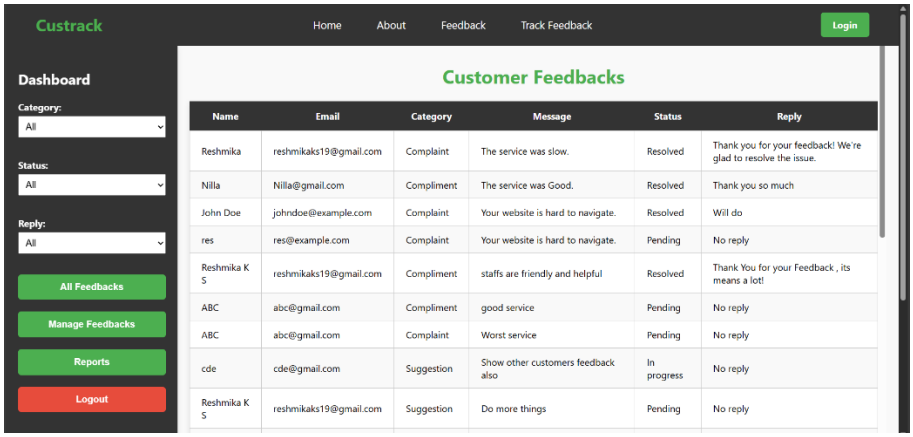


Figure – Customer Feedbacks

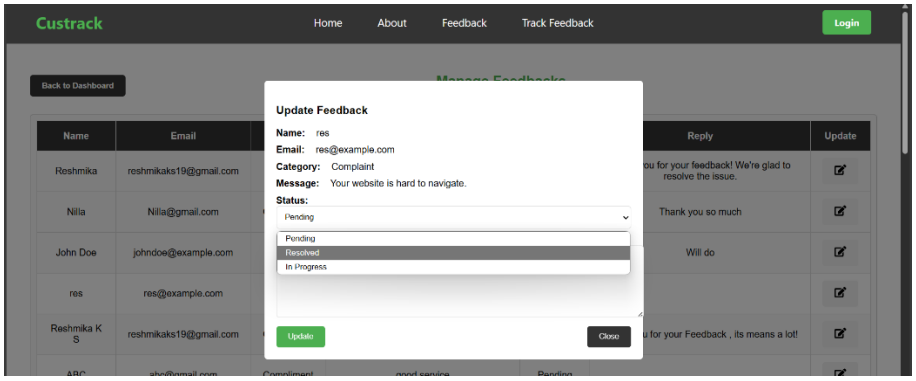


Figure – Managing Feedback

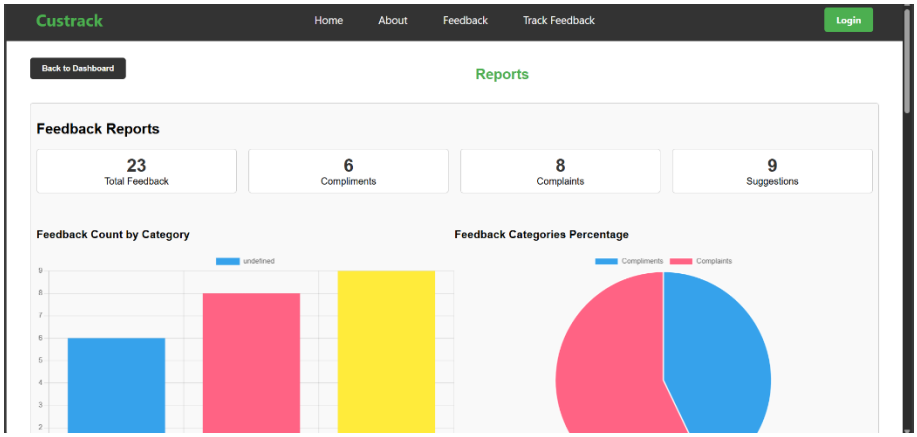


Figure - Reports

3. **FeedbackForm:**

The **FeedbackForm** is the primary interface for customers to submit feedback. Key elements include:

- **Input Fields**: Customers provide their name, email address, and feedback message, and select a category (complaint, suggestion, compliment).
- **Category Selection**: Ensures proper routing and management of feedback by categorizing the input.
- **Track ID Generation**: Once feedback is submitted, the system generates a unique Track ID for each submission, displayed to the user. This ID allows customers to track the progress of their feedback later.
- **User Guidance**: Clear instructions guide users through the submission process to ensure completeness and accuracy.

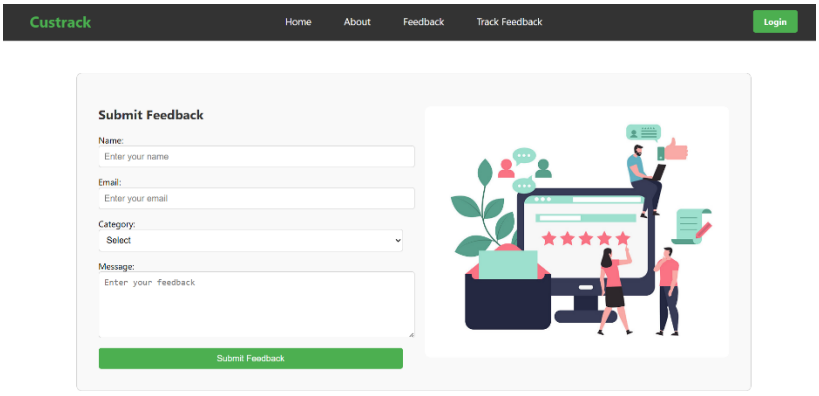


Figure – Submit Feedback

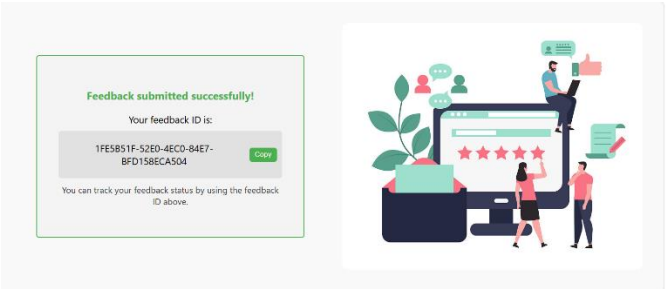


Figure – After submission

4. **HomeScreen:**

The **HomeScreen** serves as a welcoming and informational hub, structured with the following sections:

- **Introduction**: A brief welcome message that highlights the purpose of Custrack and how it simplifies feedback management.

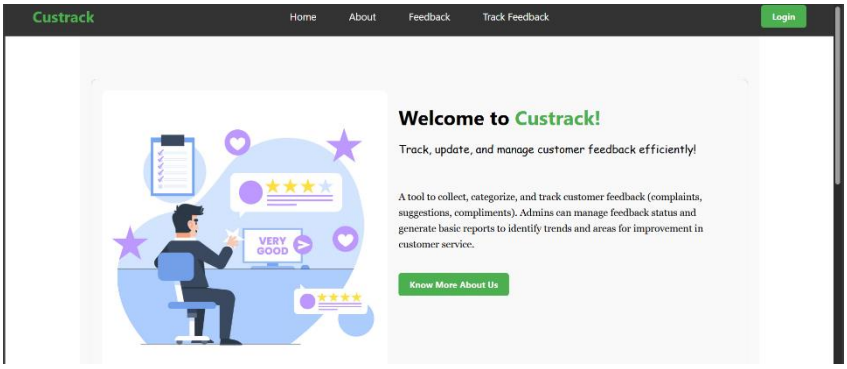


Figure - About Custrack

- **How to Submit Feedback**: Explains the steps customers need to follow to provide feedback using the platform.
- **How to Track Feedback**: Demonstrates how customers can use their unique Track ID to monitor the status of their submissions.
- **Admin Panel Overview**: A section for admins, explaining dashboard features like managing feedback, replying to customers, and generating reports.
- **Quick Links**: Navigation shortcuts to the Feedback Form, Track Feedback page, and Admin Dashboard.

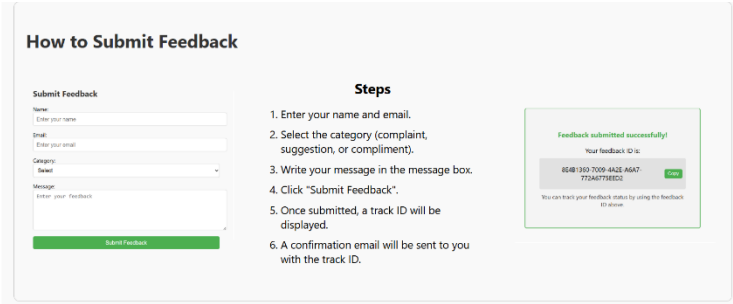


Figure – How to submit

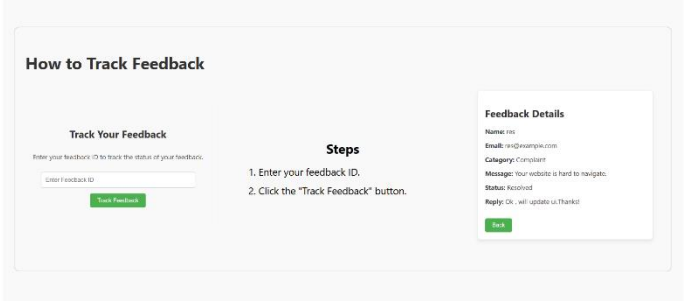


Figure – How to Track

5. **TrackFeedback:**

The **TrackFeedback** screen allows customers to monitor the status of their submitted feedback using the unique Track ID provided at the time of submission. Details include:

- **Track ID Input Field**: A search bar where customers enter their Track ID to retrieve the status of their feedback.

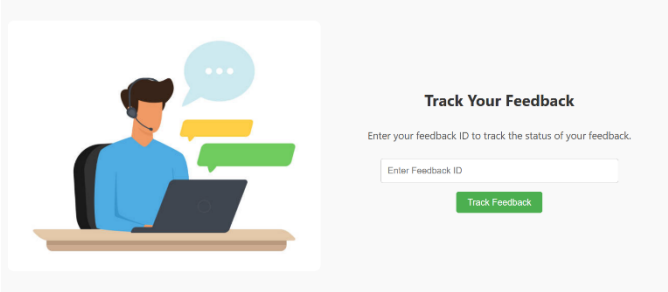


Figure – Track Feedback

- **Real-Time Status Updates**: Displays the current status (e.g., Pending, In Progress, Resolved), ensuring transparency in feedback handling.

- **Feedback Details**: Provides additional information, such as the date of submission, feedback category, and any admin responses or updates.

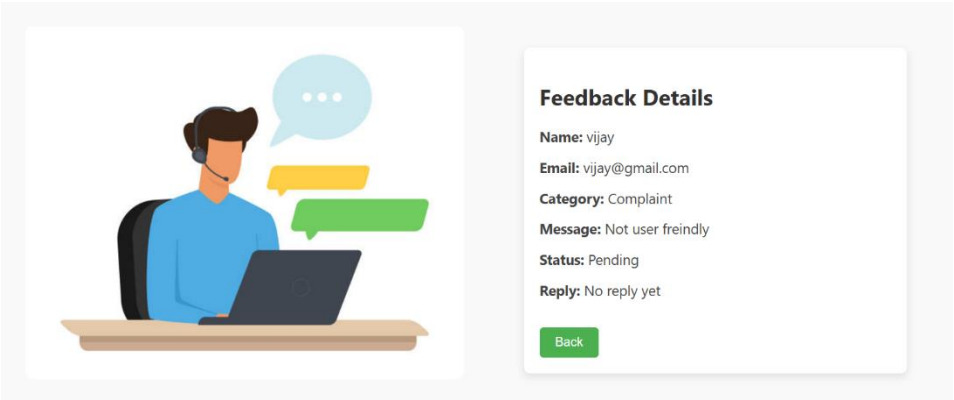


Figure –Feedback Details

Conclusion

Custrack is a robust and user-friendly web application designed to bridge the gap between customers and businesses by effectively collecting, categorizing, and managing feedback. With its seamless customer interface for feedback submission and tracking, combined with an efficient admin dashboard for managing and analyzing feedback, Custrack enhances transparency, communication, and customer satisfaction.

The implementation of modern technologies such as React.js, Node.js, and MongoDB ensures scalability, efficiency, and security, making it adaptable to various industries. By providing valuable insights through data-driven reports, Custrack empowers businesses to identify trends, address concerns, and continuously improve their customer service.

This system demonstrates how technology can streamline the feedback process, fostering better relationships between businesses and their customers while driving organizational growth and excellence.