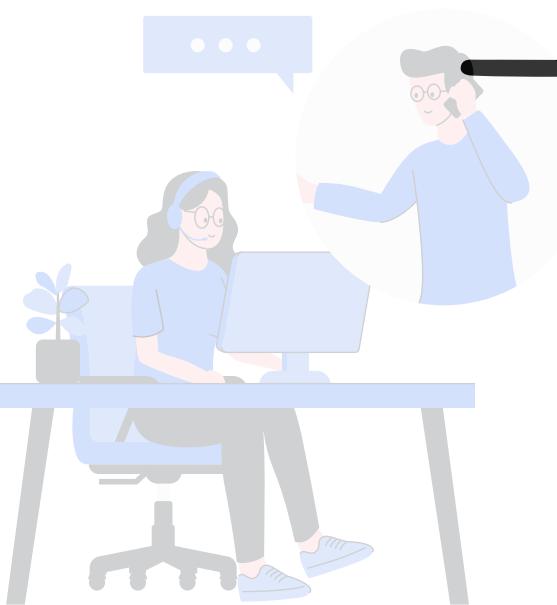




# **ONLINE COMPLAINT REGISTRATION AND MANAGEMENT SYSTEM**

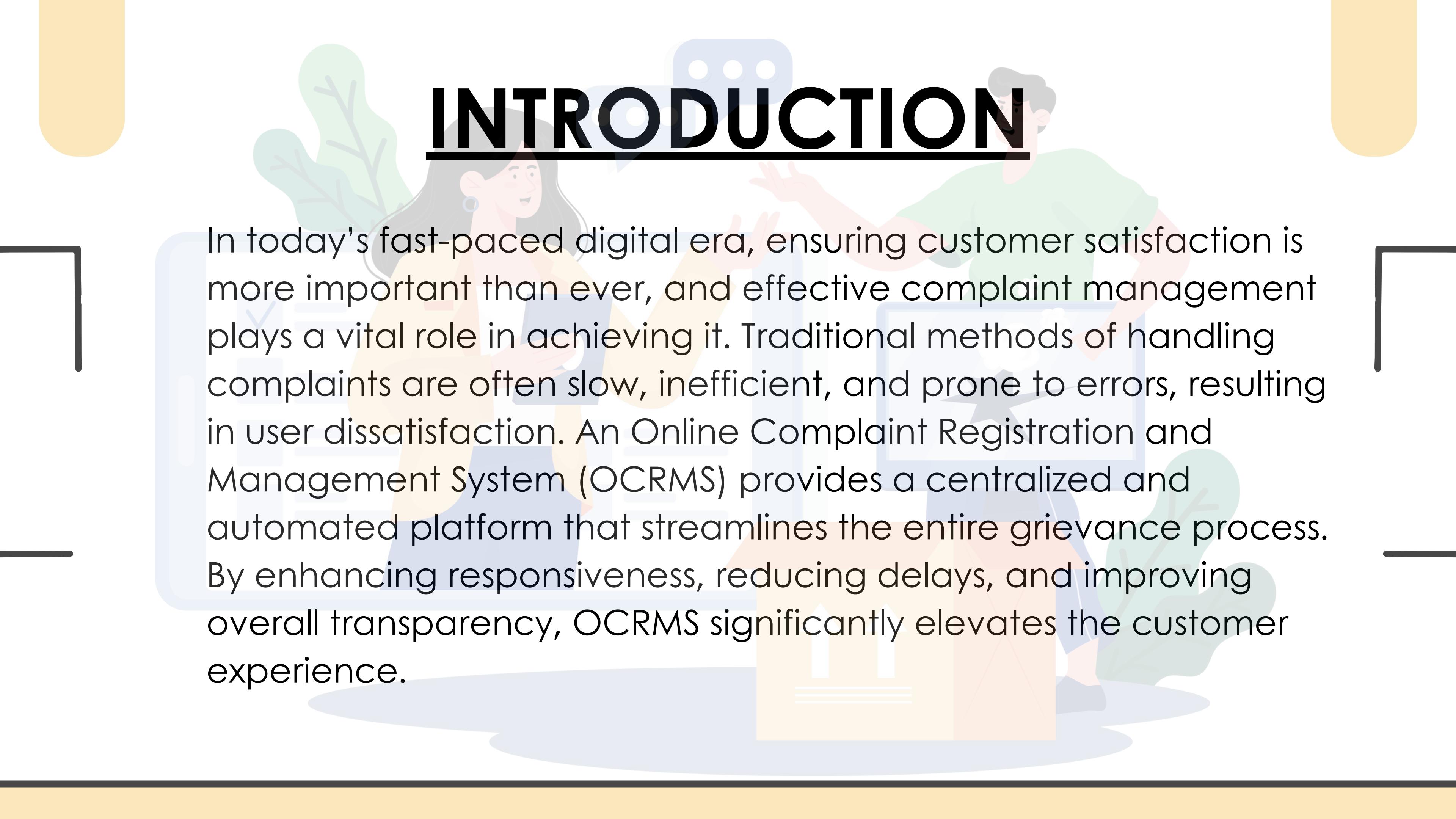
Using **MERN**



# OBJECTIVE

The **Online Complaint Registration and Management System** provides a user-centric, streamlined platform for lodging and tracking complaints with ease. It significantly improves response efficiency, fosters transparency, and ensures greater accountability throughout the process. By centralizing complaint handling, the system enables better resource utilization, enhances user satisfaction, and supports instant status updates. It modernizes conventional grievance redressal methods by offering a structured, intuitive interface that meets the growing need for fast, reliable, and transparent problem resolution.

# INTRODUCTION



In today's fast-paced digital era, ensuring customer satisfaction is more important than ever, and effective complaint management plays a vital role in achieving it. Traditional methods of handling complaints are often slow, inefficient, and prone to errors, resulting in user dissatisfaction. An Online Complaint Registration and Management System (OCRMS) provides a centralized and automated platform that streamlines the entire grievance process. By enhancing responsiveness, reducing delays, and improving overall transparency, OCRMS significantly elevates the customer experience.

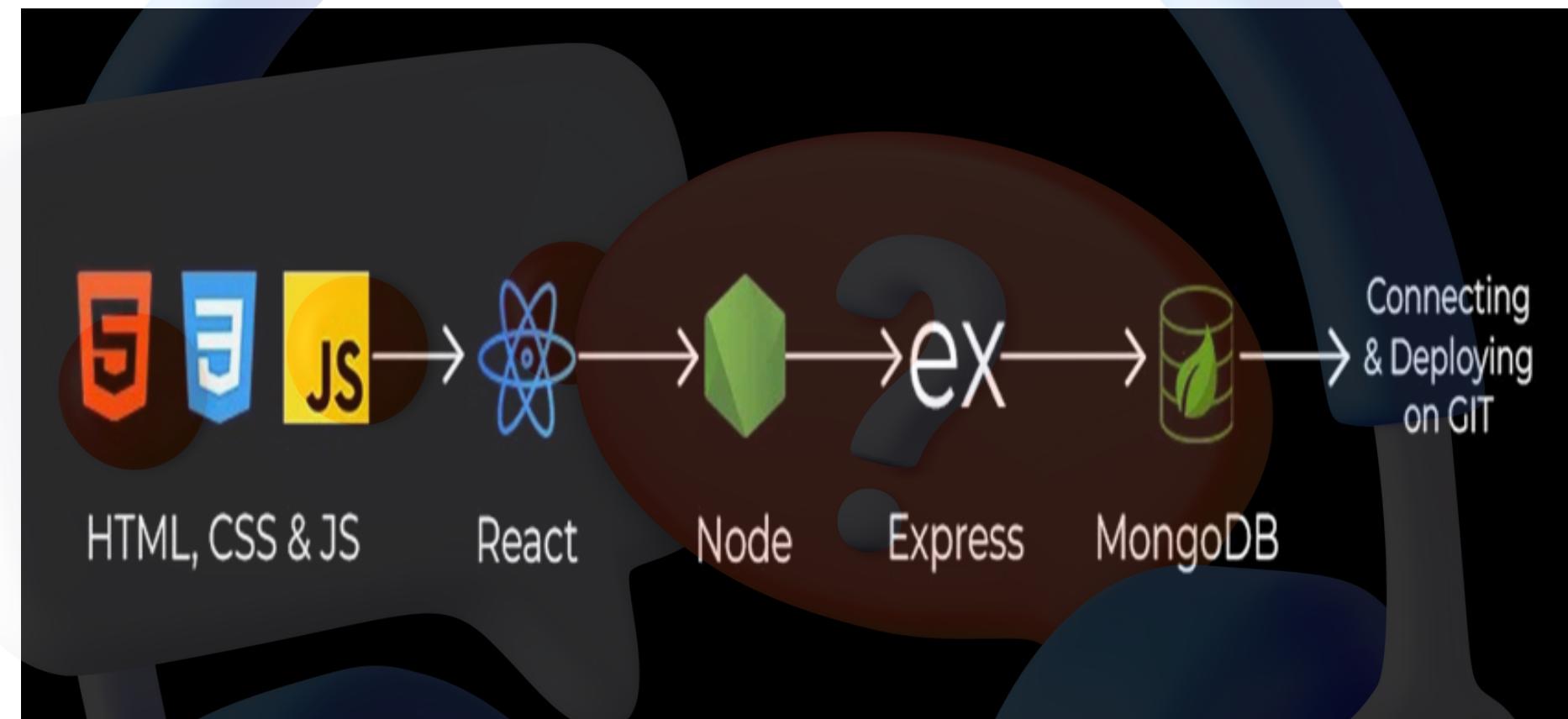
# **PROBLEM STATEMENT**

- Traditional systems are manual and slow.
- They lack transparency and efficiency.
- Complaints are filed through paper, calls, or emails.
- Tracking progress becomes difficult.
- Delays and miscommunication are common.
- There's no centralized way to ensure resolution.

# **SOLUTION**

- The aim of an online complaint registration and management system is to address these issues by providing a digital platform that allows users to easily submit complaints, track their progress, and receive timely updates

# Roadmap to MERN



**M - MongoDB**

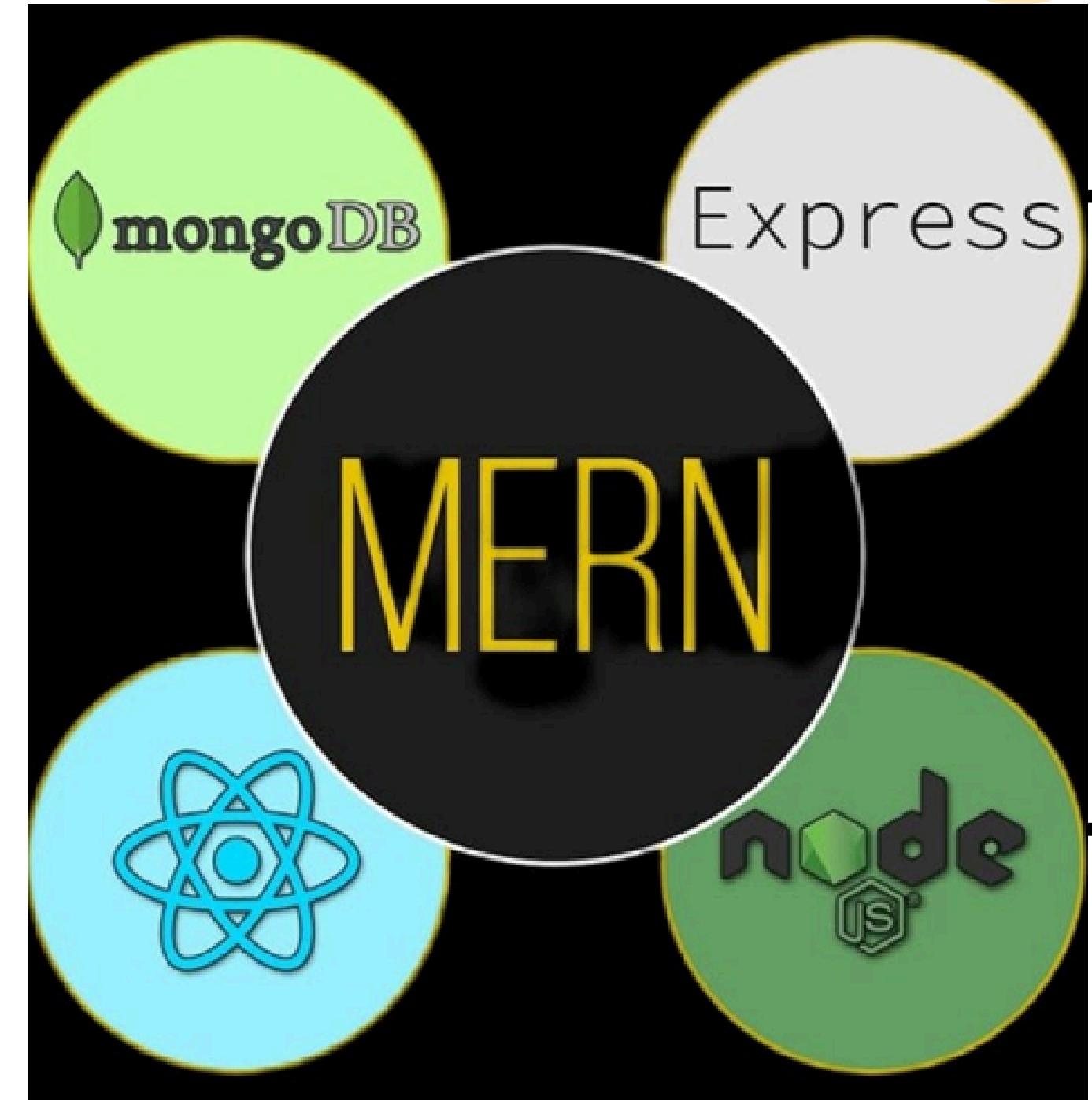
**E - Express.js**

**R - React.js**

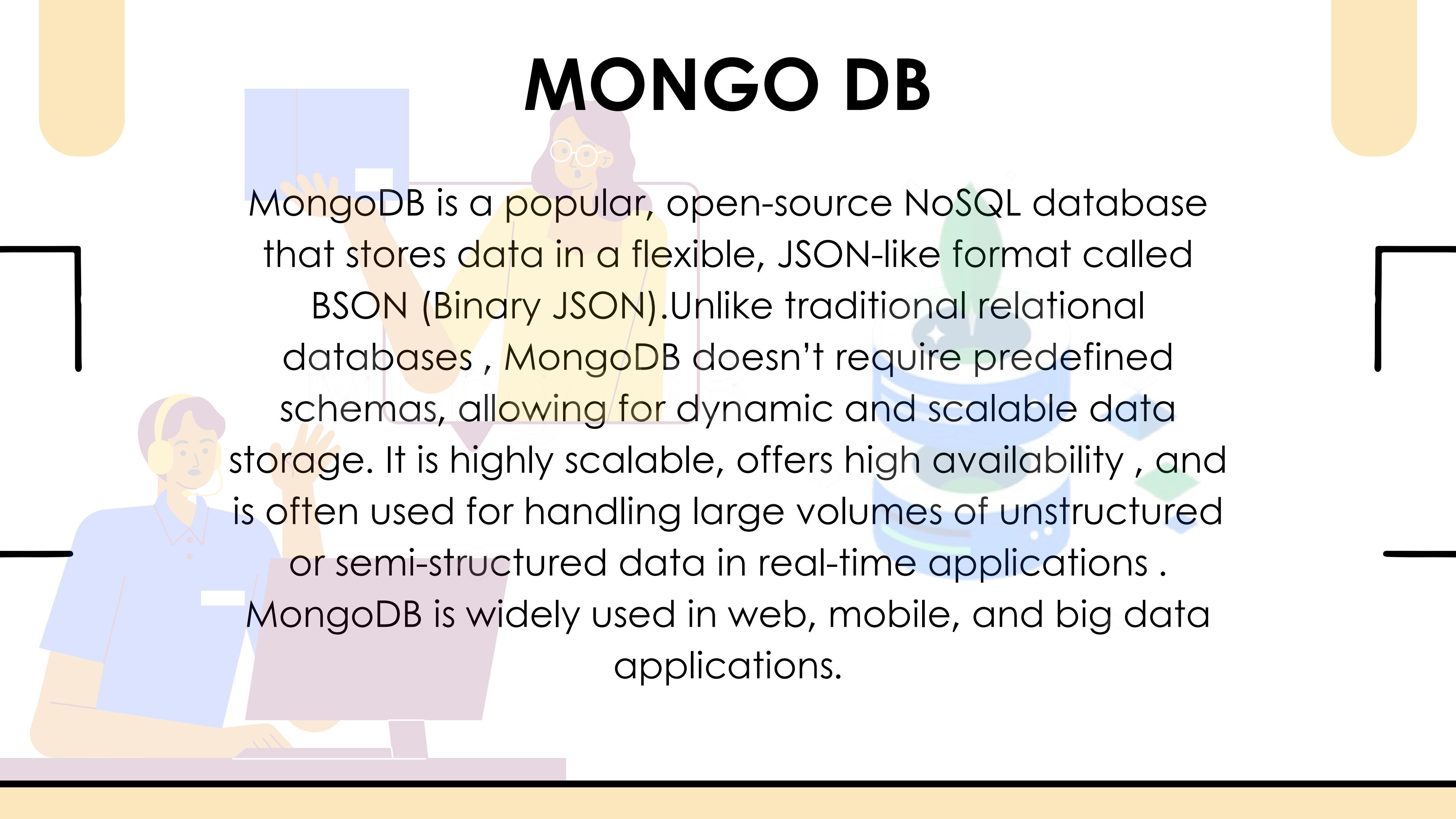
**N - Node.js**

# TOOLS USED

Using the MERN stack—MongoDB, Express.js, React, and Node.js—creates a robust complaint management system. MongoDB stores complaint and user data, while Express.js and Node.js handle backend operations and API routes for smooth data flow. React provides a dynamic, user-friendly interface, enabling users to submit and track complaints seamlessly.



# MONGO DB



MongoDB is a popular, open-source NoSQL database that stores data in a flexible, JSON-like format called BSON (Binary JSON). Unlike traditional relational databases , MongoDB doesn't require predefined schemas, allowing for dynamic and scalable data storage. It is highly scalable, offers high availability , and is often used for handling large volumes of unstructured or semi-structured data in real-time applications .

MongoDB is widely used in web, mobile, and big data applications.

# EXPRESS JS

- Express is a fast, minimal web application framework for Node.js, used to build server-side applications and APIs. It simplifies handling HTTP requests and responses, routing, and middleware, making it easier to develop robust, scalable applications. Express provides a flexible structure that allows developers to add various modules, making it ideal for building RESTful APIs and full-stack applications. It's a popular choice in the MERN (MongoDB, Express, React, Node) stack for web development.

# REACT JS

**React** is a popular, open-source JavaScript library developed by Facebook for building dynamic, interactive user interfaces, especially for single-page applications. Using a component-based architecture, React enables developers to create reusable UI components that update efficiently in response to data changes. Its virtual DOM feature optimizes rendering, making applications fast and responsive. React is widely used in modern web development, often paired with libraries and frameworks to build full-featured applications, and is a core technology in the MERN (MongoDB, Express, React, Node) stack.

# NODE JS

**Node.js** is an open-source, server-side runtime environment that allows developers to run JavaScript code outside the browser, primarily for building scalable network applications. Built on Chrome's V8 JavaScript engine, Node.js is known for its event-driven, non-blocking I/O model, which makes it efficient and well-suited for real-time, data-intensive applications. It's commonly used for backend development, particularly in building APIs and web servers, and is a key component of the MERN (MongoDB, Express, React, Node) stack.

# FRONT-END FUNCTIONALITY

## 1. USER-INTERFACE :

The login page allows users, administrators, and agents to securely access the system using credentials, with JWT for session management and live validation for a smooth user experience. The complaint registration page enables users to file complaints, capturing required details and supporting documents, with real-time validation. The user dashboard displays filed complaints, allowing users to track status and filter by various criteria. The admin and agent dashboards provide centralized complaint management, allowing for assignment, resolution, and real-time updates through tables, charts, and search functionalities.

## 2.REACT:



The front-end functionality of an online complaint registration and management system plays a crucial role in providing users with an intuitive and responsive experience. Built using React, a widely popular JavaScript library for building user interfaces, the front end ensures that the application is fast, interactive, and easy to use. React's component-based architecture allows for modular development, making it easier to manage and update individual elements without affecting the entire system.

# BACK-END FUNCTIONALITY

The backend of the Online Complaint Registration and Management System, built with Node.js and Express, handles data processing, secure user authentication, and API routing.

## 1. Express :

manages routes for complaint submission, status updates, and real-time notifications.

## 2. Node.js :

ensures a fast and scalable server. Admin and agent-specific routes enable efficient complaint management, with robust security features to protect user data and support seamless front-end communication.

# DATABASE FUNCTIONALITY

**MongoDB** is a NoSQL. In a complaint management system, the database records complaint details, tracks statuses, manages user information, logs staff-user communications, and supports real-time updates. It also generates analytics for trend analysis and reporting, and triggers notifications for status changes, enabling efficient, transparent, and accountable management of complaint resolutions.

# DATABASE CONNECTIVITY

The screenshot shows the MongoDB Atlas Data Services Overview page for the project "sample-project". The top navigation bar includes "Atlas", "rithika's Org ...", "Access Manager", "Billing", "All Clusters", "Get Help", and "rithika". The left sidebar lists "sample-project" under "Data Services", with sections for "Overview", "DATABASE", "Clusters", "SERVICES" (Atlas Search, Stream Processing, Triggers, Migration, Data Federation, Programmatic Access), and "SECURITY" (Quickstart, Backup, Database Access). The main content area displays the "RITHIKA'S ORG - 2024-09-08 > SAMPLE-PROJECT" cluster. The "Clusters" section shows "Cluster0" with "Data Size: 135.78 MB". It features buttons for "Connect", "Edit configuration", "Browse collections", "Migrate data", "View monitoring", and "+ Add Tag". A "Toolbar" on the right provides links to "Featured Resources" (NODEJS: Aggregations in Node.js, Semantic Search with Atlas Vector Search, More Javascript Content) and "Sample Apps" (NODEJS: MERN Stack, MEAN Stack, Remix Stack).

# DISADVANTAGES BEFORE IMPLEMENTING ONLINE COMPLAINT CARE:

- 1. Limited Reach:** Traditional channels restrict easy complaint submission.
- 2. Slower Response Time:** Complaints take longer to resolve.
- 3. Higher Costs:** Manual handling increases operational expenses.
- 4. Lack of Transparency:** No visibility into complaint progress.
- 5. Limited Insights:** Missed data analysis for improvements.
- 6. Increased Workload:** Staff overwhelmed by manual processes.
- 7. Frustration:** Customers frustrated by slow resolutions.

# APPLICATIONS :



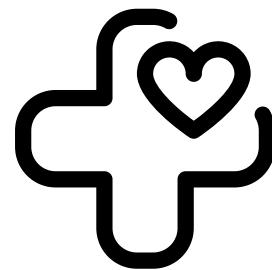
## Customer Service:

Businesses use it to track and resolve customer issues, improving service quality and satisfaction.



## Education:

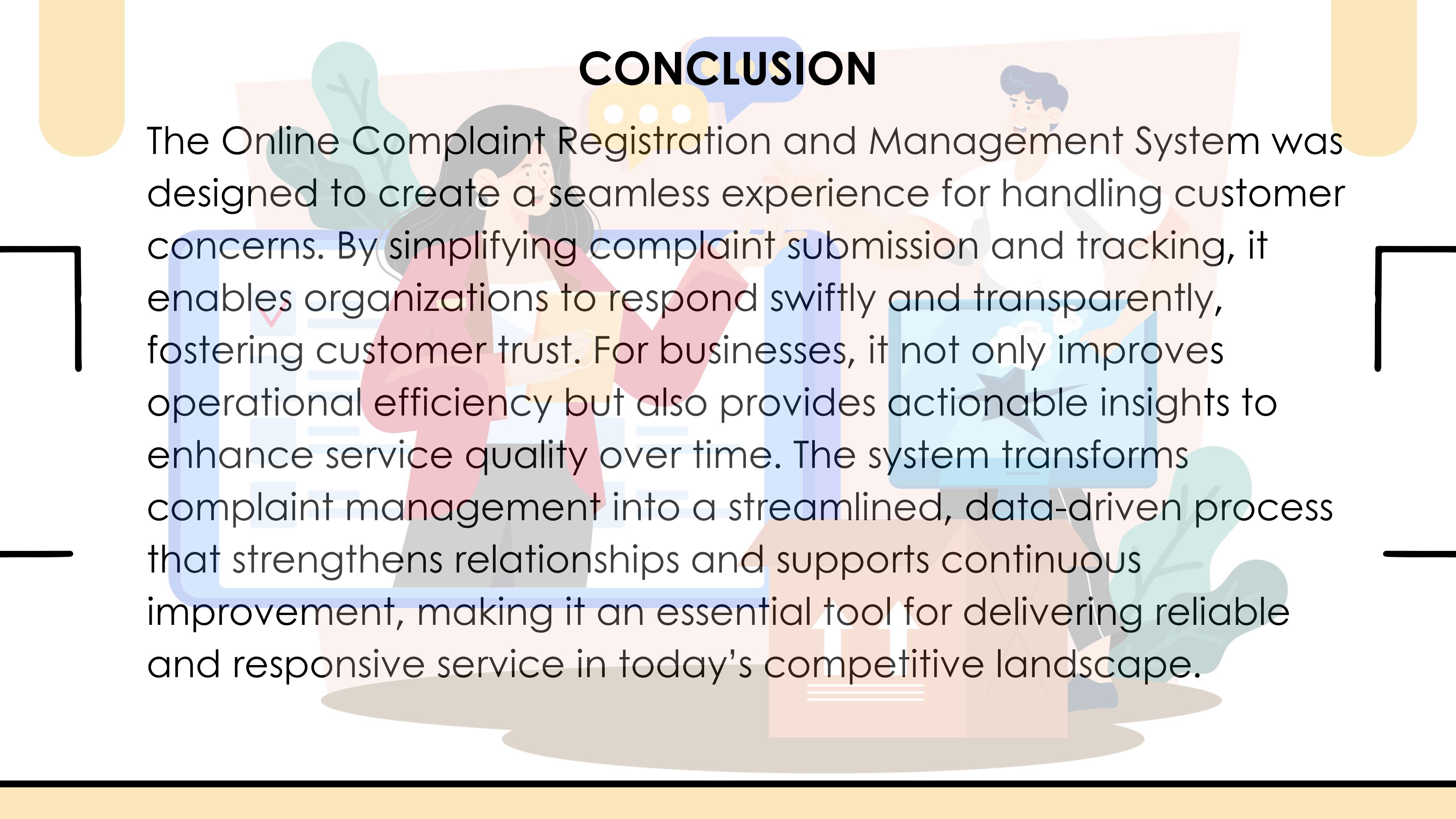
Educational institutions use it to handle complaints from students, faculty, and staff, addressing issues related to facilities, teaching, or administration.



## Healthcare

Hospitals and clinics manage patient complaints regarding care quality, appointment scheduling, and billing concerns.

# CONCLUSION



The Online Complaint Registration and Management System was designed to create a seamless experience for handling customer concerns. By simplifying complaint submission and tracking, it enables organizations to respond swiftly and transparently, fostering customer trust. For businesses, it not only improves operational efficiency but also provides actionable insights to enhance service quality over time. The system transforms complaint management into a streamlined, data-driven process that strengthens relationships and supports continuous improvement, making it an essential tool for delivering reliable and responsive service in today's competitive landscape.

# **THANK YOU**

**PRESENTED BY**

**BANDIKATTU CHAITANYA - 22BFA37015**

**ANANDARAJ ALWIN RAJ - 22BFA37008**

**KURUBA GAGANA - 22BFA37055**

**LEBAKA MAHIJA - 22BFA37057**

**DASARI TEJA - 22BFA37029**