



Dr. Vishwanath Karad

**MIT WORLD PEACE
UNIVERSITY | PUNE**

TECHNOLOGY, RESEARCH, SOCIAL INNOVATION & PARTNERSHIPS

**School of Computer Science and Engineering
Department of Computer Engineering and Technology
Third Year B. Tech. CSE (Cybersecurity and Forensics)**

FULL STACK DEVELOPMENT

MINI PROJECT REPORT ON

“ECOCOLLECT: YOUR GREEN COMPANION”

Group Details:

NAME	ROLL NO	PRN
Rishika Agarwal	1032232953	63
Khushi Singh	1032233940	77
Meghshyam Kimmkar	1262240002	78
Vishakha Poojari	1262240013	79

Abstract

EcoCollect is a full-stack web application designed to promote environmental awareness and sustainable recycling practices. It combines a responsive React-style frontend with a robust Node.js and Express backend, using MongoDB for data persistence. The platform features interactive dashboards, gamified quizzes, a recycling pickup scheduler, educational guides, and community features to motivate eco-conscious behaviours. By providing real-time tracking, visual analytics, and social comparison, EcoCollect encourages user engagement and reinforces environmentally responsible actions. This report details the system's full-stack architecture, API design, database models, and practical implementation.

Introduction

1.1 Purpose of the Project

EcoCollect aims to bridge the gap between environmental knowledge and actionable behaviour. Recycling is essential for resource conservation, reducing carbon emissions, and minimizing landfill waste. Many individuals fail to consistently adopt sustainable behaviours due to lack of motivation or awareness. EcoCollect addresses this through gamification, interactive learning, and social accountability, built on a modern full-stack foundation.

1.2 Objectives of EcoCollect include:

- Educating users on eco-friendly practices and environmental concepts.
- Tracking user actions and visualizing environmental impact via a dynamic dashboard.
- Motivating engagement through gamified features such as XP, badges, and streaks.
- Encouraging social accountability through community leaderboards and pledges.
- Providing a seamless user experience with a responsive frontend and a secure, scalable backend.

From a theoretical perspective, EcoCollect leverages behavioural psychology principles, including positive reinforcement, habit formation, and social learning, to increase long-term adherence to sustainable behaviours.

System Overview

2.1 Architecture

EcoCollect employs a MERN-like stack (MongoDB, Express, React-style frontend, Node.js) architecture:

- **Frontend:** Built with vanilla HTML, CSS (Bootstrap), and JavaScript, providing a single-page application (SPA) experience. Chart.js provides interactive data visualizations.
- **Backend:** A RESTful API built with Node.js and Express, handling authentication, data processing, and business logic.
- **Database:** MongoDB Atlas (cloud) for persistent storage of users, pickups, pledges, and stats.
- **Authentication:** Uses bcryptjs for secure password hashing.

This architecture ensures a responsive, scalable, and visually appealing platform with a clear separation of concerns between the client and server.

2.2 Features Overview

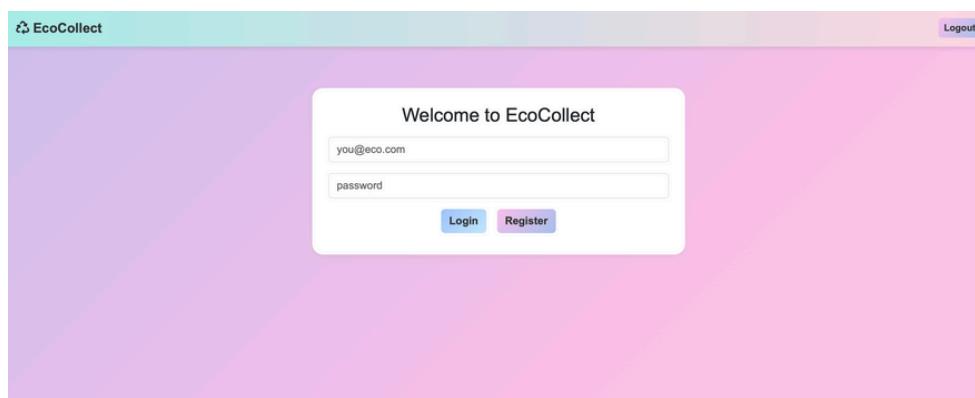
EcoCollect integrates several interconnected modules, all powered by the backend API:

- **Dashboard:** Visualizes user recycling statistics, XP, streaks, and achievements with dynamic charts.
- **Schedule Pickup:** Manages planning, tracking, and completion of waste collections.
- **Eco Guide:** Offers animated carousels with theoretical and practical sustainability knowledge.
- **Eco Quiz:** Tests knowledge with immediate feedback, streak tracking, and XP rewards.
- **Fun Interaction Game:** A hands-on drag-and-drop recycling categorization game with scoring.
- **Community Module:** Leaderboards, collective carbon goals, and public pledges.

User Interface Design

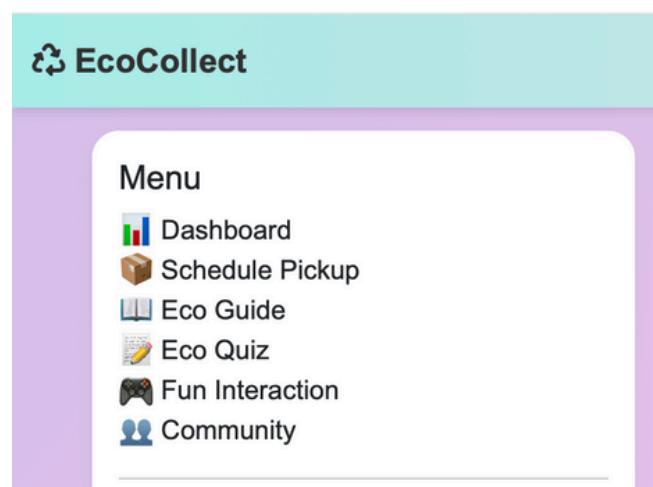
3.1 Theme & Aesthetic

The platform employs pastel gradients, smooth hover effects, emojis, and animations to create an inviting visual experience. Confetti and chime effects celebrate milestones and achievements, reinforcing user motivation. Design choices follow cognitive load theory by reducing visual strain and guiding attention to key elements.



3.2 Navigation

A sidebar menu allows smooth, SPA-like navigation between modules without page reloads. Carousel design in the Eco Guide and interactive dashboards ensure intuitive transitions while maintaining user orientation.



3.3 Accessibility

EcoCollect is fully responsive across devices using Bootstrap. Readable fonts, high-contrast colours, and consistent visual cues improve accessibility. The platform also supports interactive carousels and drag-and-drop elements to engage users with diverse learning preferences.

Dashboard Module

The Dashboard serves as the central hub for user engagement. Users can track recycled items, total weight, carbon offset, and XP earned, all fetched in real-time from the backend API (`/api/stats/:userEmail`).

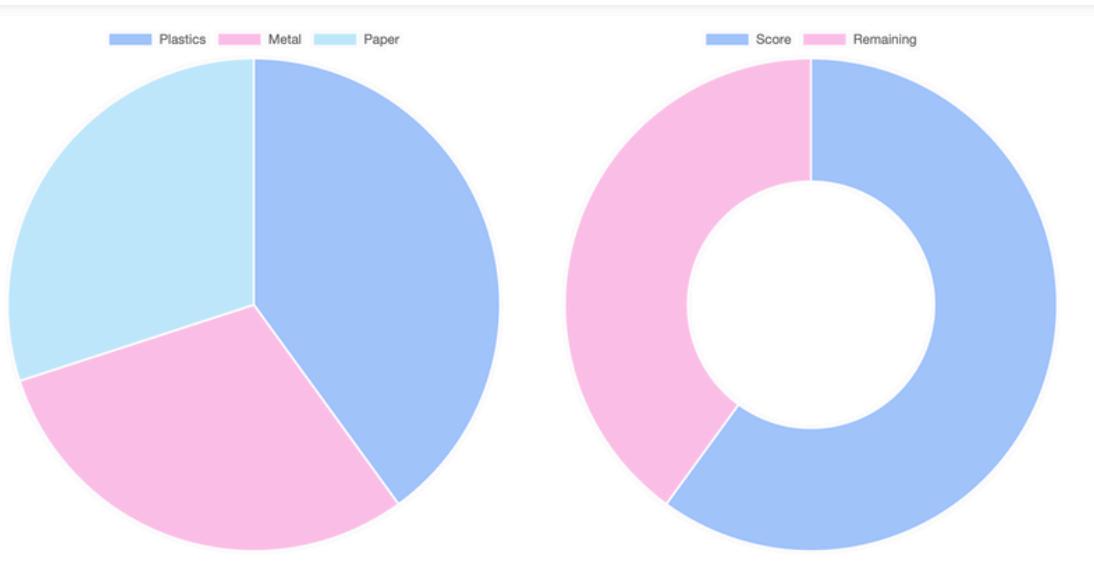
Charts & Visualizations:

- Monthly bar chart for recycling trends.
- Pie chart showing materials distribution.
- Doughnut chart reflecting quiz performance.

Progress Tracking:

- Carbon offset progress bars.
- Environmental impact equivalents: trees saved, water conserved, energy saved.
- Achievements and streak counters to reward consistent engagement.





Carbon Offset Progress

17%

Trees saved: 0 | Water saved: 20L | Energy saved: 8 kWh

Total pickups completed: 2

Badges earned: Plastic Buster, Paper Saver

Consecutive engagement streak: 7 days

Theoretical Basis:

Visual representation of data leverages information visualization theory, helping users interpret abstract environmental metrics effectively. Achievements and progress bars employ behavioural psychology principles, reinforcing habits through positive feedback.

Schedule Pickup Module

The Schedule Pickup module allows users to plan, monitor, and track recycling activities via the backend API. Input fields include date, time, location, pickup type, phone, materials, and notes. The Pickup Tracker table displays all scheduled pickups, allowing users to mark tasks as completed, which triggers XP rewards.

- Interaction & Feedback: Notifications and sound effects reinforce positive behaviour.
- Extended Functionality: Users can view historical pickups, check completion status, and track efficiency over time.

The screenshot shows the 'EcoCollect' application interface. On the left, there's a sidebar with a 'Logout' button at the top, followed by a 'Menu' section containing links to 'Dashboard', 'Schedule Pickup', 'Eco Guide', 'Eco Quiz', 'Fun Interaction', and 'Community'. Below the menu is an email address: 'demo@ecopickup.com'. The main content area is titled 'Schedule a Pickup'. It includes input fields for 'Date' (30-09-2025), 'Location' (Nagpur), 'Pickup Type' (represented by a car icon), 'Time' (07:28), and 'Phone' (5534266789). There are also checkboxes for 'Plastics' (unchecked), 'Metal' (checked), and 'Paper' (checked). A 'Notes (optional)' field is present, and a blue 'Save Pickup' button is at the bottom.

The screenshot shows a 'Pickup Tracker' table with a header row labeled: Date, Time, Location, Type, Phone, Materials, Notes, Status, and Action. The table contains three data rows:

Date	Time	Location	Type	Phone	Materials	Notes	Status	Action
2025-09-17	undefined	undefined	undefined	undefined	Metal, Paper	undefined	undefined	<button>Complete</button>
2025-09-28	23:09	Pune	Doorstep	12342343456	Plastics, Metal	Completed	Completed	<button>Complete</button>
2025-09-30	07:28	Nagpur	Drop-off	5534266789	Metal, Paper	Scheduled	Scheduled	<button>Complete</button>

Theory Applied:

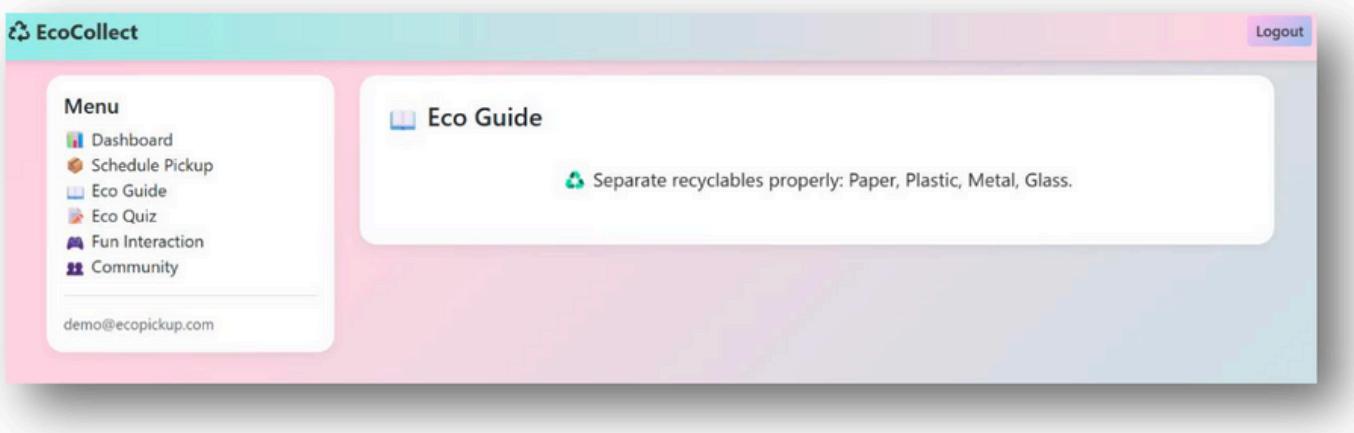
Tracking progress supports self-monitoring theory, which increases adherence by visualizing accomplishments and promoting accountability.

Eco Guide Module

The Eco Guide provides interactive, educational content about sustainability, recycling processes, and environmental impact. Larger fonts and a Bootstrap carousel design enhance readability and engagement.

Content Coverage:

- Recycling hierarchies (Reduce, Reuse, Recycle).
- Life-cycle assessment of materials.
- Practical tips for everyday sustainable practices.
- Environmental impact calculations (carbon, water, energy).



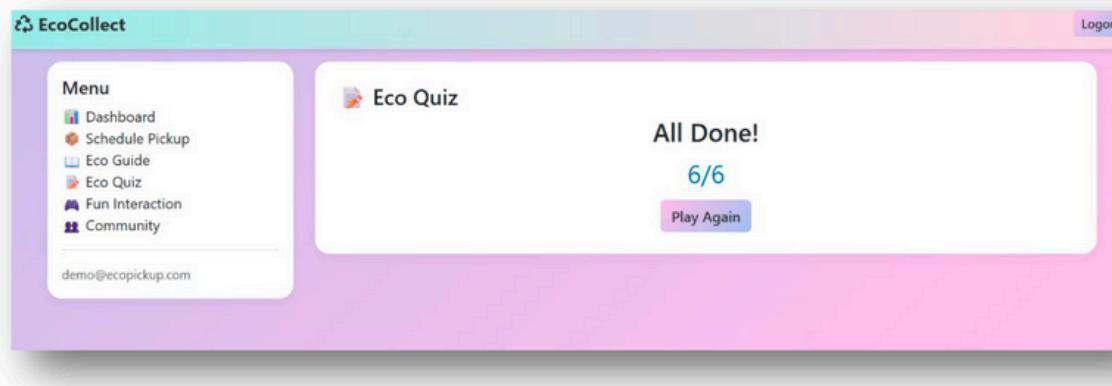
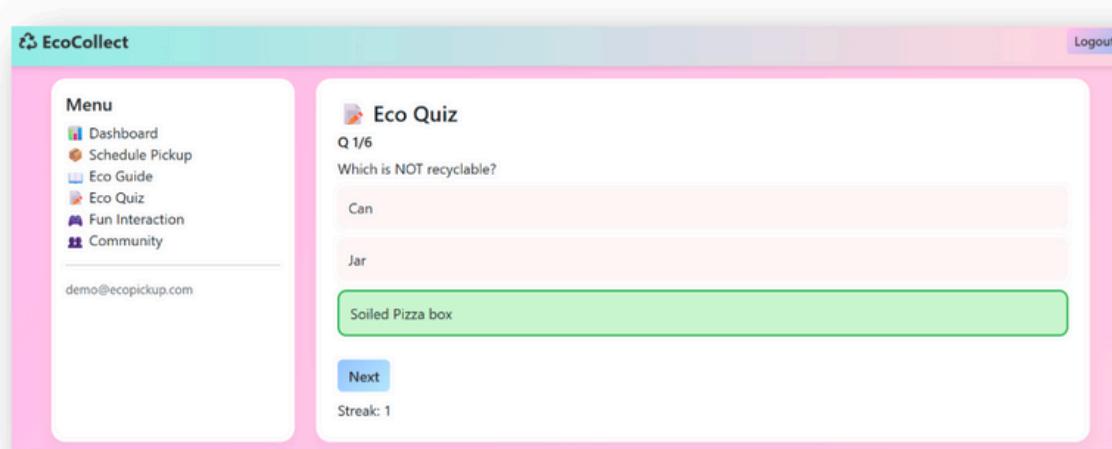
Theoretical Foundation:

The module uses multimedia learning theory to enhance comprehension and retention by integrating visuals, text, and interactivity. The combination of theory and practice supports constructivist learning, allowing users to apply knowledge actively.

Eco Quiz Module

The Eco Quiz tests knowledge with multiple-choice questions, immediate feedback, and XP rewards. Quiz streaks incentivize consecutive correct answers. The frontend manages the quiz state, while the backend could be extended to track performance.

- Gamification: Confetti, chime sounds, and visual feedback reinforce correct answers.
- Integration: Quiz performance is celebrated immediately on the frontend.



Theory Applied:

Quizzes utilize retrieval practice and game-based learning theory, which enhance memory retention and engagement by encouraging active recall and immediate reinforcement.

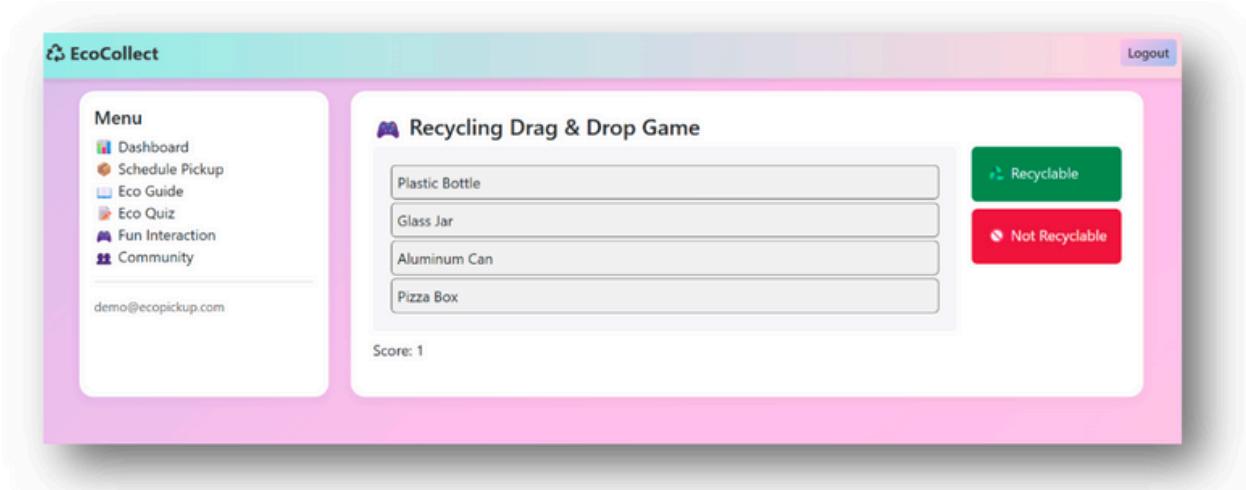
Fun Interaction Module

The drag-and-drop recycling game engages users by categorizing items as recyclable or non-recyclable.

Gameplay Mechanics:

- Immediate scoring for correct and incorrect actions.
- Positive and negative visual reinforcement.

Learning Outcome: The game bridges theoretical concepts from the Eco Guide to practical experience.

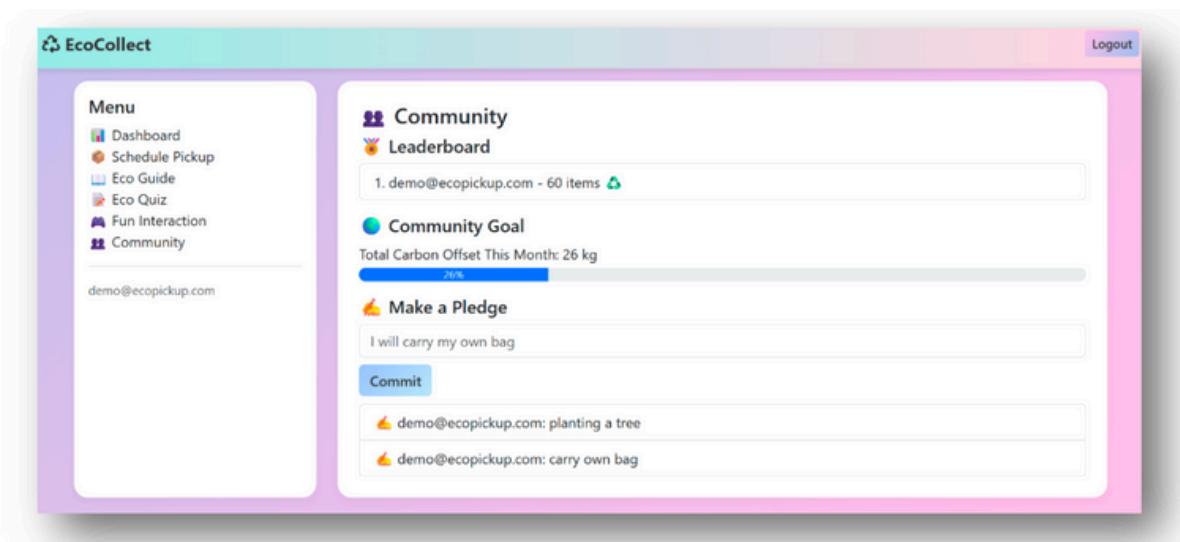


Theoretical Foundation:

Based on Kolb's experiential learning theory, this module promotes hands-on learning. Feedback mechanisms employ behaviourist principles to reinforce learning through reward and correction.

Community Module

The Community Module fosters social engagement and accountability. Users can compare performance on leaderboards, track collective carbon offset goals, and commit to pledges, all powered by backend API calls (/api/community/leaderboard, /api/pledges).



Interactive Features:

- Leaderboards ranked by user XP.
- Community goals with progress bars.
- Eco pledges for sustainable commitments.

Theory Applied:

Social cognitive theory and peer comparison are used to motivate user behaviour. Community features increase adherence to eco-friendly practices through social reinforcement and collaboration.

Backend API & Server Architecture

The backend, built with Node.js and Express, provides a secure and scalable foundation for the EcoCollect platform.

10.1 Core Dependencies:

- express: Web server framework.
- mongoose: MongoDB object modeling.
- bcryptjs: Password hashing.
- cors: Enable Cross-Origin Resource Sharing.
- dotenv: Environment variable management.

10.2 Key API Endpoints:

- POST /api/register: User registration.
- POST /api/login: User authentication.
- POST /api/pickups: Schedule a new pickup.
- GET /api/pickups/:userEmail: Get user's pickups.
- PUT /api/pickups/:id/complete: Mark pickup as completed.
- GET /api/stats/:userEmail: Get user stats and dashboard data.
- GET /api/community/leaderboard: Get top users by XP.
- POST /api/pledges: Add a new community pledge.

10.3 Data Persistence:

User data, pickups, and pledges are stored in MongoDB Atlas, ensuring persistence across sessions. The database connection is managed via mongoose.

Data Storage and Management

All user data, pickups, and pledges are stored in MongoDB collections, ensuring persistence across sessions. Actions dynamically update stats, charts, and trackers via API calls, providing immediate feedback. This approach supports self-monitoring theory, helping users visualize progress and maintain engagement.

MongoDB Schemas:

- User: email, password (hashed), xp, streak, badges[]
- Pickup: user, date, location, pickupType, time, phone, materials[], notes, status
- Pledge: user, pledge, createdAt

The screenshot shows the Compass MongoDB interface. On the left, the sidebar lists databases: '+ Create Database' (disabled), 'Search Namespaces', 'ecocollect' (selected), 'pickups' (highlighted), 'pledges', 'users', and 'sample_mflix'. The main panel title is 'ecocollect.pickups'. It displays storage details: STORAGE SIZE: 36KB, LOGICAL DATA SIZE: 258B, TOTAL DOCUMENTS: 1, INDEXES TOTAL SIZE: 36KB. Below this are tabs for 'Find', 'Indexes', 'Schema Anti-Patterns', 'Aggregation', and 'Search Indexes'. A button 'INSERT DOCUMENT' is on the right. A search bar says 'Generate queries from natural language in Compass'. A query input field contains 'Type a query: { field: 'value' }'. A 'Reset' button and an 'Apply' button are next to it. The results section shows 'QUERY RESULTS: 1-1 OF 1' with a single document:

```
_id: ObjectId('691062c4e888e5cf1f76b8')
user: "Khusht@eco.com"
date: "2025-11-10"
location: "Vanaz"
pickupType: "Doorstep"
time: "11:30"
phone: "8429862984"
materials: Array (2)
notes: ""
status: "Completed"
createdAt: 2025-11-09T09:45:40.793+00:00
__v: 0
```

The screenshot shows the Compass MongoDB interface. On the left, the sidebar lists databases: '+ Create Database' (disabled), 'Search Namespaces', 'ecocollect' (selected), 'pickups', 'pledges' (highlighted), 'users', and 'sample_mflix'. The main panel title is 'ecocollect.pledges'. It displays storage details: STORAGE SIZE: 36KB, LOGICAL DATA SIZE: 11B, TOTAL DOCUMENTS: 1, INDEXES TOTAL SIZE: 36KB. Below this are tabs for 'Find', 'Indexes', 'Schema Anti-Patterns', 'Aggregation', and 'Search Indexes'. A button 'INSERT DOCUMENT' is on the right. A search bar says 'Generate queries from natural language in Compass'. A query input field contains 'Type a query: { field: 'value' }'. A 'Reset' button and an 'Apply' button are next to it. The results section shows 'QUERY RESULTS: 1-1 OF 1' with a single document:

```
_id: ObjectId('69106377e888e5cf1f76cf')
user: "Khusht@eco.com"
pledge: "I will not do unnecessary waste"
createdAt: 2025-11-09T09:48:39.230+00:00
__v: 0
```

```

{
  "_id": ObjectId("691061bbe888e5cf1f76ab"),
  "email": "Khushi@eco.com",
  "password": "$2a$10$FoK5tFLaTowq83QaU8XsWuqyKtc5P4bJfyONGFLai7yGX7Fos0b.",
  "xp": 35,
  "streak": 1,
  "badges": [],
  "createdAt": 2025-11-09T09:40:43.407+00:00,
  "__v": 0
}

{
  "_id": ObjectId("691061bbe888e5cf1f76b0"),
  "email": "Khushi@eco",
  "password": "$2a$10$QfIIv8cRbyvY3fs2Llpg/e66cfaBj9iw7LX/vlmz7VNw/s1/m0gxu",
  "xp": 0,
  "streak": 0,
  "badges": [],
  "createdAt": 2025-11-09T09:41:15.458+00:00,
  "__v": 0
}

```

Gamification and UX Enhancements

EcoCollect integrates gamification through badges, XP, streaks, and celebratory effects, all tracked by the backend.

Badges & Levels: Earned based on milestones (e.g., 'Plastic Buster' at 100 XP), motivating consistent participation.

Streaks & XP: Rewarded for repeated engagement with tasks like pickups, quizzes, and guides.

UX Enhancements: Hover effects, animations, sound effects, and confetti celebrate achievements and guide user attention.

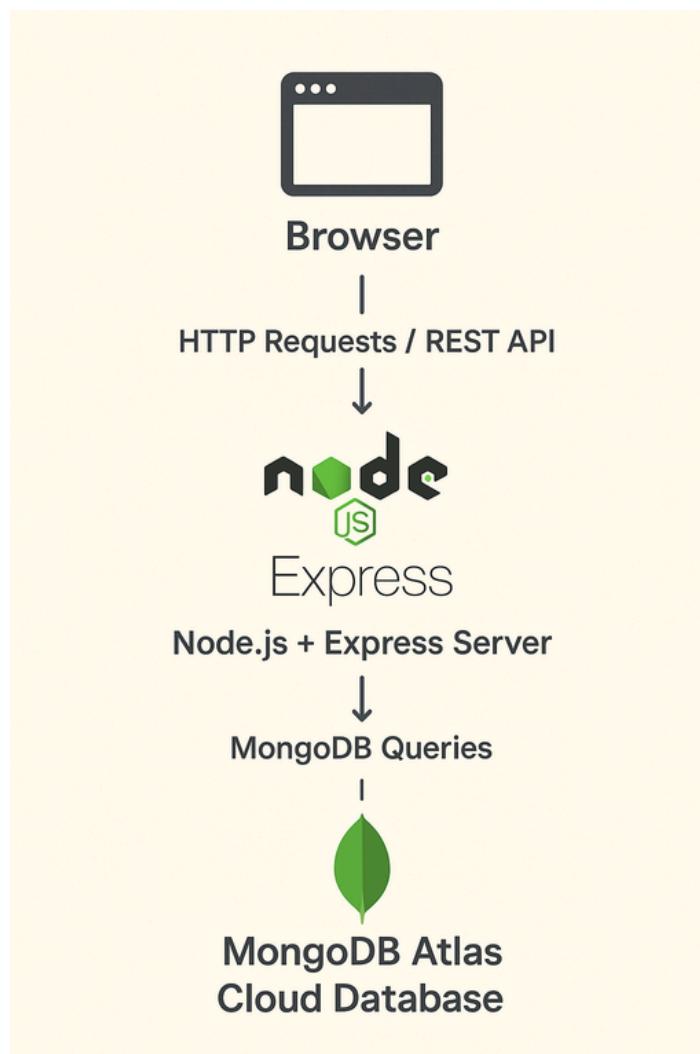
Theory Applied: Gamification leverages behavioural reinforcement, habit formation, and intrinsic motivation theory to enhance engagement and learning.

Testing and Validation

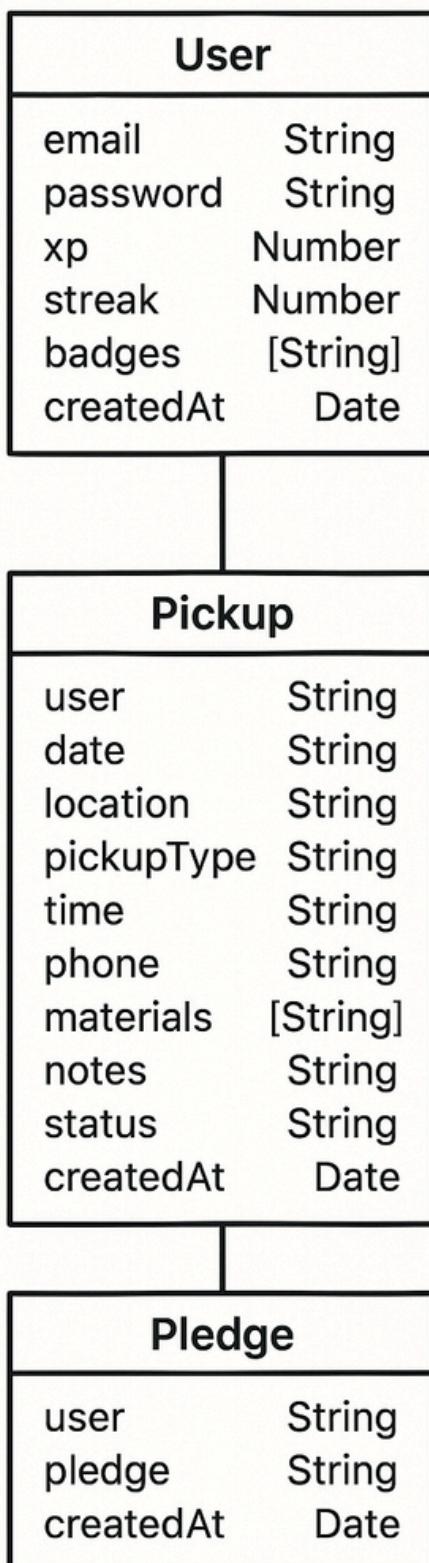
Functional testing confirmed correct operation of dashboards, quiz scoring, game mechanics, and pickup tracking via the API. UI/UX testing verified responsiveness, readability, and smooth animation. Backend API endpoints were tested for correct data creation, retrieval, and updates. Engagement metrics, including streaks, leaderboard rankings, and community participation, were monitored to ensure effective motivation and retention.

Technical Diagrams

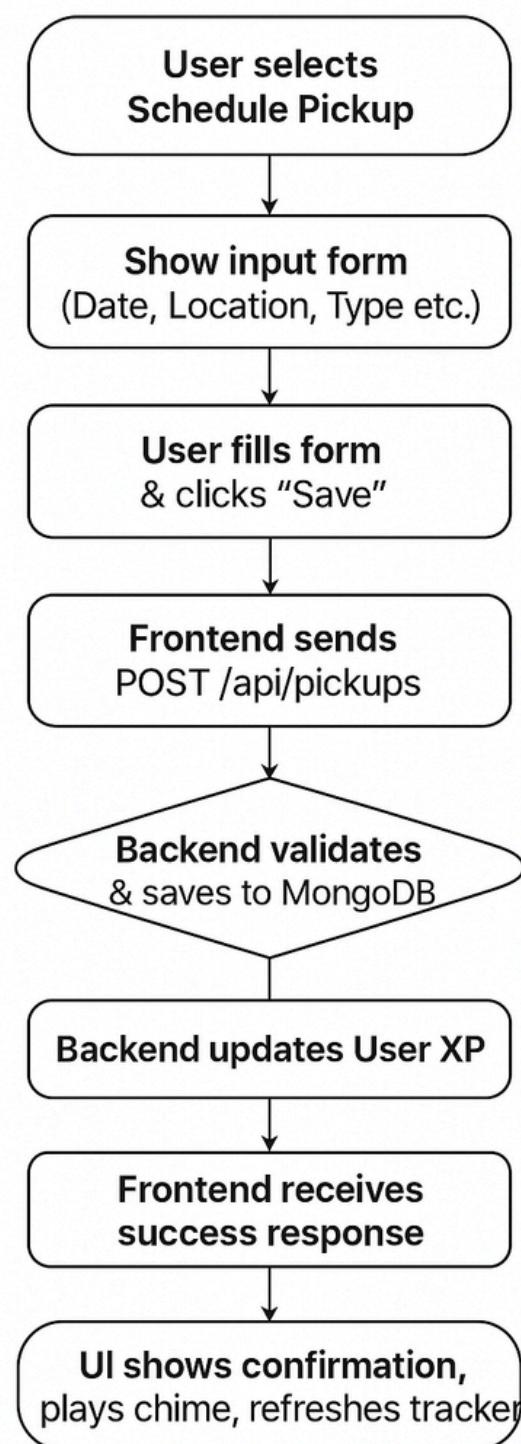
System Architecture Diagram



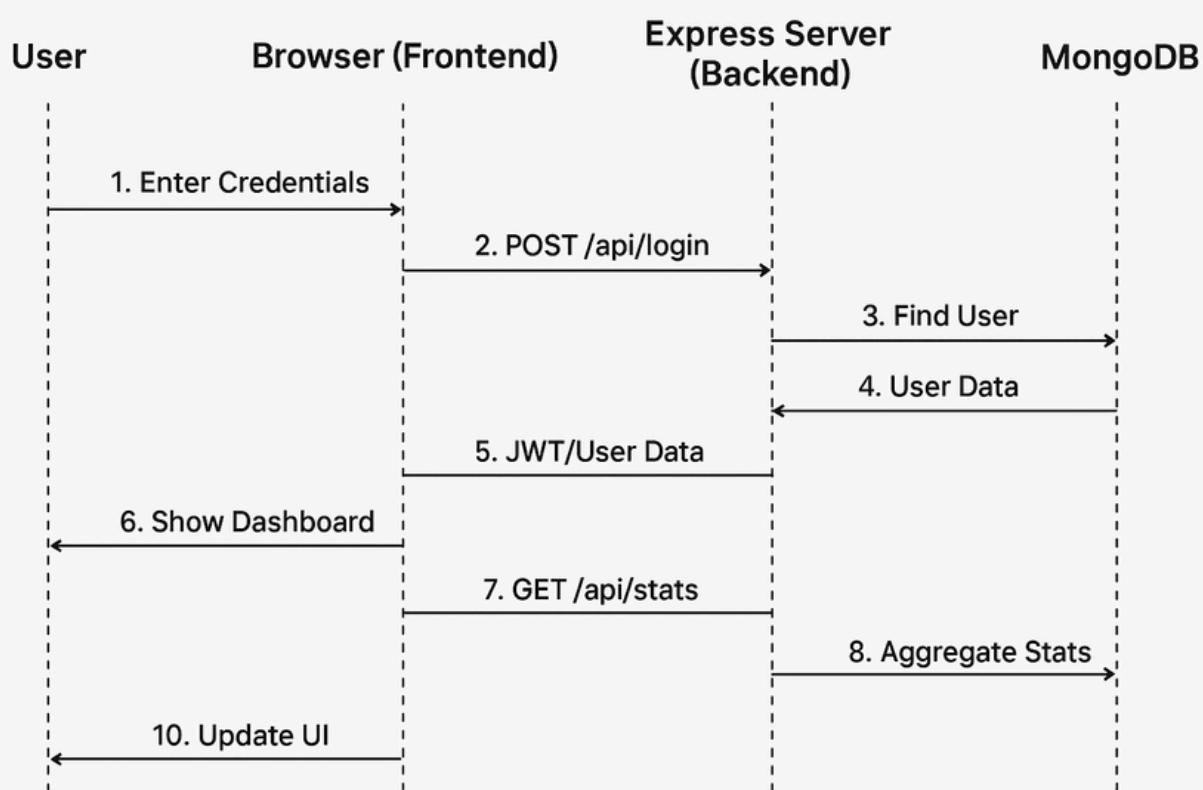
UML Class Diagram (Backend Models)



Activity Diagram: Schedule Pickup Flow



Sequence Diagram: User Login & Load Dashboard



Advantages

- Encourages adoption of environmentally responsible behaviours through a full-stack, interactive platform.
- Reinforces theoretical knowledge through interactive modules and immediate feedback.
- Fosters peer engagement and accountability via community features.
- Provides real-time tracking and visualization to motivate users.
- Supports habit formation through backend-tracked gamification and streak tracking.
- Scalable cloud-based architecture using MongoDB Atlas.

Real-World Applications

EcoCollect can be used in:

- Educational institutions to teach students about recycling and sustainability.
- Municipal recycling programs to coordinate pickups and track progress.
- NGO awareness campaigns for community engagement.
- Corporate sustainability initiatives to track employee eco-activities.

Importance of the Project

EcoCollect addresses the challenge of translating environmental awareness into action. By combining theory, education, gamification, and social reinforcement within a modern full-stack application, the platform motivates and sustains eco-friendly behaviours. Real-time tracking, community engagement, and gamified rewards provide measurable impact, fostering environmental responsibility at both individual and collective levels.

Conclusion

EcoCollect effectively integrates education, engagement, and gamification into a cohesive full-stack platform. The system's Dashboard, Schedule Pickup, Eco Guide, Quiz, Game, and Community features, powered by a robust Node.js backend and MongoDB database, collectively create a motivational and educational ecosystem. By connecting theory to practice and providing real-time feedback, EcoCollect encourages long-term sustainable behaviour. Future enhancements could include AI-driven eco tips, interactive maps for pickups, timeline visualizations, and push notifications to further increase engagement and scalability.

References

1. United States Environmental Protection Agency (EPA) - Recycling Basics:
<https://www.epa.gov/recycle>
2. United Nations Sustainable Development Goals - Goal 12: Responsible Consumption: <https://sdgs.un.org/goals/goal12>
3. Chou, Y. (2015). Actionable Gamification: Beyond Points, Badges, and Leaderboards. Octalysis Media. <https://yukaichou.com/gamification-book/>
4. Loo, R. (2016). Gamification in Education: Impact on Motivation and Learning. Journal of Learning Analytics. <https://learning-analytics.info/journals/index.php/JLA/article/view/>
5. Recycle Nation - Local Recycling Guidelines:
<https://www.recyclenation.com/>
6. McGonigal, J. (2011). Reality is Broken: Why Games Make Us Better and How They Can Change the World. Penguin Press.
<https://janemcgonigal.com/books/reality-is-broken/>
7. Express.js Documentation: <https://expressjs.com/>
8. MongoDB Documentation: <https://www.mongodb.com/docs/>
9. Mongoose ODM Documentation: <https://mongoosejs.com/>