

SCRAP EASE

A Report

Submitted in partial fulfillment of the requirement for the degree of

B.Tech.

in

Artificial Intelligence & Data Science

Under the Supervision of

Mr. Sanjay Kumar Singh

(Assistant Professor)

by

Roli (2104231630022)

Danish Khan (2104231630015)

Ashutosh Shukla (2104231630010)



Department of Computer Science & Engineering School of
Management Sciences, Lucknow

Affiliated to Dr. A.P.J. Abdul Kalam Technical
University, Lucknow

May 2025

CERTIFICATE

This is to certify that Project Report entitled “**SCRAP EASE**” which is submitted by Roli, Danish Khan, Ashutosh Shukla in partial fulfilment of the requirement for the award of degree B. Tech. in Department of Computer Science and Engineering of School of Management Sciences Lucknow, affiliated to Dr. A.P.J. Abdul Kalam Technical University, Lucknow is a record of the candidates own work carried out by them under my/our supervision. The project embodies the result of original work and studies carried out by the students themselves and the contents of the project do not form the basis for the award of any other degree to the candidate or to anybody else.

Mr. Sunit Kumar Mishra
Head of Department
CSE Department,
SMS, Lucknow

Mr. Sanjay Kumar Singh
Assistant Professor
CSE Department
SMS, Lucknow

Date:

DECLARATION

We hereby declare that this submission is our own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person nor material which to a substantial extent has been accepted for the award of any other degree or diploma of the university or other institute of higher learning, except where due acknowledgment has been made in the text.

Name: Roli

Roll No.: 2104231630022

Name: Danish Khan

Roll No: 2104231630015

Name: Ashutosh Shukla

Roll No: 2104231630010

Date:

ACKNOWLEDGEMENT

It gives us a great sense of pleasure to present the report of the B.Tech. Project undertaken during B.Tech. Final Year. We owe special debt of gratitude to our project supervisor Mr. Rahul Chaturvedi, Department of Computer Science and Engineering, School of Management Sciences Lucknow, for his constant support and guidance throughout the course of our work. His sincerely, thoroughness and perseverance have been a constant source of inspiration for us. It is only his cognizant efforts that our endeavors have seen light of the day.

We also take the opportunity to acknowledge the contribution of Professor Mr. Sanjay Kumar Singh Sir, Head, Department of Computer Science & Engineering, School of Management Sciences Lucknow, for his full support and assistance during the development of the project.

We also do not like to miss the opportunity to acknowledge the contribution of all faculty members of the department for their kind assistance and cooperation during the development of our project. Last but not the least, we acknowledge our friends for their contribution in the completion of the project.

Name: Roli

Roll No.: 2104231630022

Name: Danish Khan

Roll No: 2104231630015

Name: Ashutosh Shukla

Roll No: 2104231630010

Date:

ABSTRACT

Scrapease is a web-based platform developed to streamline the process of booking scrap vendors for the collection and sale of various recyclable materials such as metal, plastic, and paper. The primary aim of the project is to provide an organized and user-friendly interface that connects individuals and businesses with verified scrap vendors in their locality. Through this system, users can schedule pickups, track vendor responses, and contribute to sustainable waste management practices. The platform is designed with responsiveness, real-time communication, and ease of use as its core principles.

This project not only digitizes an otherwise informal sector but also promotes environmental awareness by encouraging responsible disposal of recyclable waste. The system architecture follows a client-server model, ensuring efficient data handling and secure transactions between users and vendors. The successful implementation of Scrapease demonstrates the feasibility of leveraging technology to enhance the efficiency and transparency of scrap collection services, ultimately aiming to create a cleaner and more sustainable environment.

LIST OF FIGURES

- Fig 2.1. – 0 Level DFD.
- Fig 2.2. – 1 Level DFD.
- Fig 2.3. – 2 Level DFD.
- Fig 2.4. – E-R Diagram.
- Fig 2.5. – Process Flow Diagram.
- Fig 2.6. – Behavioral Diagram.
- Fig 3.1. – Signup Authentication Module.
- Fig 3.1.1. – Login Authentication Module.
- Fig 3.2. – How It Works Module.
- Fig 3.3. – Scrap Categories Module.
- Fig 3.4. – Track Module.
- Fig 3.5. – Dealer Dashboard.
- Fig 3.6. – Contact Us Module.

LIST OF TABLES

- Fig 5.2. – Project Schedule

LIST OF ABBREVIATIONS

REACT	---	JavaScript Library
RAM	---	Random Access Memory
GB	---	Giga Byte
SSD	---	Solid State Drive
UI/UX	---	User Interface/ User Experience
DOM	---	Document Object Model
CSS	---	Cascading Style Sheets
ORM	---	Object – Relational Mapping
JWT	---	Json Web Token
NODE	---	JavaScript Runtime Environment
HTTP	---	Hyper Text Transfer Protocol
ETC	---	Et Cetera
SMTP	---	Simple Mail Transfer Protocol
CRUD	---	Create Read Update Delete

TABLE OF CONTENTS

CERTIFICATE	ii
DECLARATION	iii
ACKNOWLEDGEMENT.....	iv
ABSTRACT.....	v
LIST OF FIGURES.....	vi
LIST OF TABLES	vi
LIST OF ABBREVIATIONS	vii
CHAPTER 1 (INTRODUCTION)	
1.1. Literature review	1
1.2. Problem definition.....	2
1.3. Brief introduction of project.....	2
1.4. Proposed modules.....	3
1.5. Hardware Requirements	5
1.6. Software Requirements.....	6
CHAPTER 2 (SYSTEMS ANALYSIS AND SPECIFICATION)	
2.1. A functional model.....	8
2.2. A data model	11
2.3. A process-flow model.....	12
2.4. A behavioral model	13
2.5. System Design.....	14
2.5.1 Technical feasibility.....	14
2.5.2 Operational feasibility.....	15
2.5.3 Economic feasibility	15

CHAPTER 3 (MODULE IMPLEMENTATION & SYSTEM INTEGRATION)

3.1. Signup/Login Process Module.....	16
3.2. How It Works Module	18
3.3. Scrap Categories Module.....	19
3.4. Track Module.....	20
3.5. Dealer Dashboard Module	21
3.6. Contact Us Module	22

CHAPTER 4 (TESTING AND EVALUATION)

4.1 Types of Testing Performed.....	23
4.2. Evaluation Criteria	24

CHAPTER 5 (TASK ANALYSIS AND SCHEDULE OF ACTIVITIES)

5.1 Task decomposition.....	26
5.2 Project schedule.....	28
5.3 Task specification.....	28

CHAPTER 6 (PROJECT MANAGEMENT)

6.1 Major risks and contingency plans	30
REFERENCES.....	32

CHAPTER 1 – INTRODUCTION

1.1. Literature Review

The Existing literature on waste management and recycling largely focuses on formal systems, with limited attention given to the application of digital platforms within the informal scrap sector. While studies acknowledge the importance of informal recycling, there's a gap in research exploring dedicated online solutions to connect waste generators directly with scrap vendors. However, insights from the burgeoning e-commerce and sharing economy offer valuable frameworks. The success of on-demand service platforms highlights the potential of digital interfaces to enhance efficiency, transparency, and accessibility by directly linking service providers and users. These models provide principles for user acquisition, trust-building, and seamless transactions, directly relevant to Scrapease's aim of connecting individuals and businesses with verified scrap vendors.

Furthermore, research in Human-Computer Interaction (HCI) and online platform design emphasizes the critical role of user-centricity. Intuitive navigation, clear information architecture, and reliable communication are vital for fostering user engagement and trust in online marketplaces. For Scrapease, understanding these principles is crucial to create an effective and user-friendly platform for both sellers and buyers of scrap materials. Studies on trust in online peer-to-peer marketplaces also underscore the need for mechanisms ensuring vendor reliability.

The current literature reveals a significant gap in digitally-driven solutions specifically tailored to the intricacies of the scrap collection and trading process. Scrapease aims to address this by creating a dedicated web-based platform, drawing inspiration from successful on-demand models while adapting to the specific needs of the recyclable materials market. Future research on user behavior within this ecosystem, effective vendor management, and the socio-economic impact of digitalizing this traditionally informal sector will be crucial for Scrapease's growth and its potential to promote sustainable waste management practices.

1.2. Problem Definition

Train Individuals and businesses seeking to dispose of recyclable materials like metal, plastic, and paper frequently encounter challenges in connecting with reliable scrap vendors. These problems include the difficulty in locating verified buyers, a lack of transparent pricing for different materials, and the often time-consuming and informal nature of arranging for collection.

Existing methods for selling scrap are often decentralized, lack standardized processes, and can lead to uncertainty regarding fair pricing and reliable service. Furthermore, scrap vendors may face issues such as inconsistent supply, difficulties in reaching a wider network of potential sellers, and a lack of efficient logistical planning tools.

The absence of a structured system hinders the smooth flow of recyclable materials back into the recycling stream. The current lack of a dedicated digital platform to efficiently connect sellers of recyclable materials with verified scrap vendors contributes to a fragmented and often inefficient recycling process. Scrapease aims to address these problems by providing a user-friendly web platform to facilitate seamless booking and connection for scrap collection services, thereby promoting a more organized and sustainable approach to waste management.

1.3. Brief Introduction of Project

Scrapease is a web-based platform designed to revolutionize how individuals and businesses in India manage the disposal and sale of their recyclable materials, such as metal, plastic, and paper. Recognizing the common difficulties faced by users in finding dependable and fairly priced scrap vendors, Scrapease provides a user-friendly platform to seamlessly connect sellers with local, verified scrap buyers, whom we refer to as "Scrap Partners."

This project introduces a convenient online system that allows users to easily request a pickup by providing essential details such as the type and estimated quantity of scrap materials and their location. Registered users benefit from personalized experience, including pickup history and profile management. Simultaneously, "Scrap Partners" can register on the platform to receive and accept pickup requests, effectively managing their collection services and material acquisition.

By establishing a direct link between sellers and scrap vendors, Scrapease aims to bring transparency, reliability, and efficiency to the traditionally informal process of scrap collection and sale. The platform facilitates scheduled pickups and aims to provide a clear channel for communication and potential price negotiation. By digitizing this process, Scrapease lays the groundwork for a more organized and accessible scrap recycling ecosystem, starting with a focus on simplifying the responsible disposal and profitable sale of recyclable waste.

1.4. Proposed Modules

1.4.1. User Management Module

User Signup (Image 4): Enables new users (individuals and businesses) to create accounts by providing their name, email, and password.

User Login (Image 2): Allows registered users to securely access their accounts using their email and password.

User Profile Management (Implicit): While not explicitly shown as a dedicated page, this module would allow users to manage their contact information, pickup addresses, and potentially notification preferences.

1.4.2. Booking/Pickup Management Module

Scrap Category Display & Booking Initiation (Image 3): Presents the different categories of recyclable materials (Metal, Plastic, Paper) with potential pricing and a "Book Pickup" button to start the request process.

Pickup Request Details (Implicit): This would involve pages or forms where users specify the type and estimated quantity of scrap, preferred pickup date and time, and their location details.

Order Tracking (Image 5): Allows users to monitor the status of their scheduled pickups, including details like Order ID, material, weight, status (Pending, In Progress, Completed), and pickup date.

Pickup History (Implicit): Enables users to view a record of their past scrap pickups.

The Booking/Pickup Management module is central to the user experience, guiding sellers through the process of requesting scrap collection in a clear and structured manner. It encompasses the selection of scrap categories, the detailed input of pickup information, and the ability to track the progress of their requests. Users can also view and manage their past and current pickup orders through this module.

1.4.3. Vendor (Scrap Partner) Management Module (Dealer Dashboard - Image 7)

Vendor Signup (Implicit): Enables scrap vendors to register and create a profile on the platform, likely involving verification processes.

Vendor Login (Implicit): Allows registered vendors to securely access their dashboard.

Vendor Dashboard (Image 7): Provides an overview of total requests, pending requests, and completed requests. It also displays current material prices and a list of recent pickup requests with details like customer, material, weight, and status.

Request Management (Implicit from Dashboard): Enables vendors to view, accept, and manage incoming pickup requests.

Status Updates (Implicit from Dashboard): Allows vendors to update the status of pickups (e.g., Accepted, In Progress, Completed).

This module caters to the scrap vendors (Scrap Partners), enabling them to register and manage their presence on the platform. It allows vendors to log in, view and accept pickup requests, manage their schedules, and potentially update material prices. The vendor dashboard serves as their central hub for managing their operations on Scrapease.

1.4.4. Transaction/Information Display Module

Website Landing Page (Implicit): The main entry point for users, providing general information about Scrapease.

About Us (Implicit): Information about the platform's mission and goals.

Contact Us (Image 6): Provides contact information (email, phone, address) and a contact form for users and vendors to reach out.

How It Works (Image 1): Explains the step-by-step process of using Scrapease (Schedule a Pickup, Pickup at Your Address, Receive Payment).

This module provides essential content and communication channels for all users of the Scrapease platform. The landing page introduces the service, the "About Us" section provides more details, and the "Contact Us" page allows for inquiries and support. The "How It Works" section clearly outlines the user journey.

1.4.5.Pricing/Payment Information Module (Implicit)

Material Price Display (Image 3): Shows the current prices being offered for different categories of scrap materials (e.g., ₹45/kg for Aluminum Cans).

Invoice/Summary (Implicit): While not explicitly shown, this module would likely provide a summary of the collected scrap and the calculated payment to the user after the pickup is completed.

This module focuses on displaying pricing information for different recyclable materials to users. It would also handle the calculation and presentation of the payment amount due to the user based on the collected weight and material type.

1.5. Hardware Requirements

1.5.1. Minimum Requirements

For the Scrapease platform to function on the server-side, a basic setup would include at least a dual-core processor with a clock speed of 2 GHz, accompanied by a minimum of 4 GB of DDR4 RAM. The storage requirement starts at 40 GB, utilizing an SSD for better responsiveness. A standard 1 Gbps Ethernet connection is also necessary for network communication. For initial deployment, a basic Virtual Private Server (VPS) can serve as the hosting environment.

On the client-side, both users (individuals/businesses selling scrap) and "Scrap Partners" (vendors) would need a device with at least a single-core processor operating at 1 GHz, 1 GB of RAM, and sufficient free storage for the operating system and a web browser. The display should support a minimum resolution of 1024x768 pixels, and a basic internet connection (Wi-Fi or cellular data) along with a modern web browser are essential to access the platform.

1.5.2.Recommended Requirements

To ensure optimal performance and scalability for the Scrapease platform, the server infrastructure should ideally consist of a server-grade processor with at least four cores running at 2.5 GHz or higher, complemented by 8 GB or more of DDR4 RAM. The storage should be 80 GB or more of SSD in a RAID configuration for enhanced reliability and speed. A dedicated 1 Gbps Ethernet connection is preferred for consistent network performance. For hosting, a scalable VPS or an entry-level dedicated server or cloud instance would provide better resource management. On the client-side, a device equipped with a dual-core processor clocked at 1.5 GHz or higher and at least 2 GB of RAM is recommended for a smoother user experience. The display should support a resolution of 1366x768 or higher. A stable broadband internet connection (Wi-Fi, 4G, or 5G) and the latest version of a modern web browser are advised for optimal functionality and security.

1.6. Software Requirements

1.6.1.Frontend Technologies

The Scrapease website frontend utilizes React.js as the core JavaScript library for building the user interface. Its component-based architecture facilitates modular development, code reuse, and maintainability. Styling is primarily achieved through CSS, allowing for complete control over the visual design and a tailored user experience. Figma was likely used for the initial UI/UX design, providing a visual blueprint for the platform's layout and interactive elements. This combination of React.js for structure and custom CSS for styling enables a flexible and bespoke frontend development approach for Scrapease.

1.6.2.Backend and Database

The Scrapease website backend is built using Node.js, a JavaScript runtime environment that allows for server-side logic execution. This enables a consistent language across the frontend and backend development. The backend handles crucial functionalities such as user and vendor authentication, pickup request management, data processing, and communication with the database. Data persistence for the application is managed by MySQL, a robust relational database management system suitable for storing and retrieving user information, pickup details, vendor data, and other essential platform data. phpMyAdmin may be used as a web-based administration tool for the MySQL database, providing a user-friendly interface for database management and querying.

1.6.3.Authentication and Notification Services

Authentication for Scrapease likely employs JWT (JSON Web Tokens) stored in browser cookies to manage user and vendor sessions and implement role-based access control. This ensures that different user types (e.g., scrap sellers, scrap vendors, administrators) can only access relevant features and functionalities. For communication, Node Mailer or a similar service is likely used to handle email notifications such as account verification, password reset requests, pickup confirmations, and status updates. This enhances user and vendor communication and provides professional platform experience.

1.6.4.Development Tools

The Scrapease website's frontend interface is developed using React.js, leveraging its component-based structure for creating dynamic and interactive user interfaces. Styling is implemented using CSS, allowing for precise visual design and a unique platform aesthetic. Dynamic manipulation of UI elements within React.js components is achieved through interaction with the Virtual DOM, ensuring efficient updates and rendering. The backend logic, responsible for managing user and vendor data, pickup processes, and server-side operations, is built using Node.js, enabling scalable and efficient server-side development. Data for Scrapease is persistently stored and managed using MySQL, providing a structured and reliable way to organize information. The administration of the MySQL database is likely facilitated through phpMyAdmin. This combination of React.js and CSS for the frontend, and Node.js with MySQL for the backend, forms the technological foundation of the Scrapease web platform.

1.6.5.Browser and Operating System Compatibility

The Scrapease website prioritizes compatibility with the latest stable versions of major web browsers, including Google Chrome, Mozilla Firefox, Safari, and Microsoft Edge, to ensure optimal performance and rendering of its React.js frontend and CSS. This web-based approach inherently supports a wide range of operating systems, including Windows, macOS, Linux, and ChromeOS for desktop users, as well as Android and iOS for mobile users, provided they access the site through one of the recommended modern browsers. Users and vendors are encouraged to keep their browsers updated for the best experience and security.

CHAPTER 2 – SYSTEM ANALYSIS AND SPECIFICATION

2.1. Functional Model

The Scrapease platform operates on a functional model centered around connecting individuals and businesses needing to dispose of recyclable materials with registered scrap vendors ("Scrap Partners"). Users initiate the process by providing details about the type and quantity of their scrap and their location through an online booking/pickup request system. This request is then made available to verified "Scrap Partners" in their locality, who can accept the pickup request. Once accepted, the "Scrap Partner" schedules and carries out the pickup, and payment is facilitated based on the collected materials (details of the payment method would be defined in subsequent levels). The platform provides separate interfaces for users to manage their pickup requests and profiles, and for "Scrap Partners" to manage incoming requests and their service availability, facilitating an efficient and transparent exchange of services.

2.1.1. 0 Level DFD

The Level 0 DFD for Scrapease shows the entire website as one process. It illustrates how Customers (Scrap Sellers) send Pickup Schedule Requests and Login/Registration Data, receiving Pickup Confirmation in return. Scrap Partners (Vendors) would (implicitly in this diagram) send Login/Registration Data and receive Pickup Requests. The system also interacts with a Database (not explicitly shown in the provided "Connect Coolie" DFD but crucial for Scrapease) to fetch and store data. The provided DFD also shows interactions with a Payment Gateway (for potential online transactions) and a Scrapease Admin for platform management.

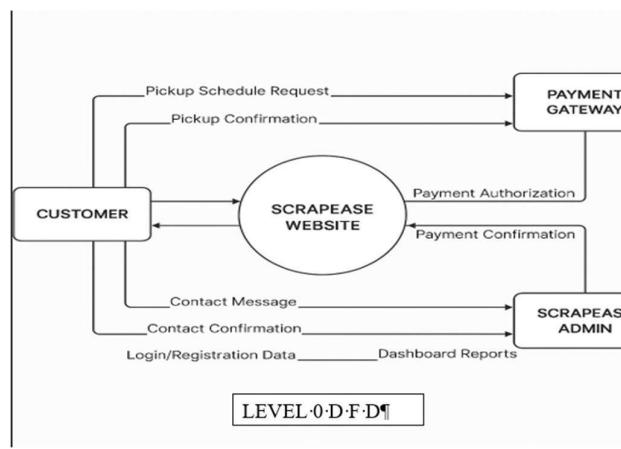


Fig 2.1. 0 Level DFD

2.1.2. 1 Level DFD

Scrapease's Level 1 DFD outlines key functions: User & Vendor Management (login/signup), Scrap Pickup Request Management (customer requests), Vendor Request Processing (vendor accepts), Pickup Scheduling & Tracking (status updates), Information & Support (platform info), and Payment Processing. Data flows between users, vendors, these modules, and the central Database to facilitate scrap pickups.

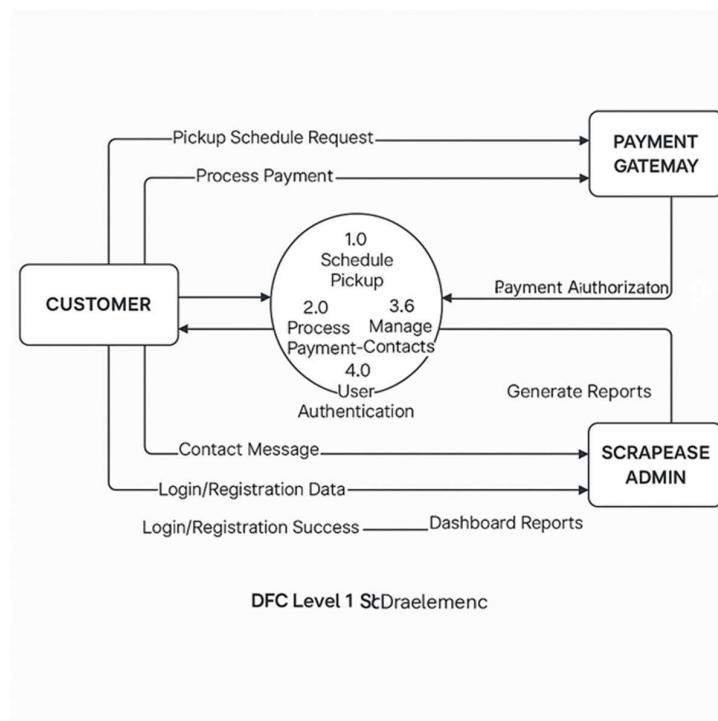


Fig 2.2. 1 Level DFD

2.1.3. 2 Level DFD

Scrapease's Level 2 DFD details user and vendor account management (signup, login, password recovery using respective databases), customer pickup request handling (submission, tracking, cancellation via Pickup DB), vendor processing of requests (viewing, accepting, updating Pickup DB), pickup scheduling and notifications, information and support via contact messages (Contact Msgs DB), and payment processing (recording payments, future gateway integration with Payments DB). These sub-processes illustrate the detailed data flow between users, vendors, the system's core functionalities, and specific data storage components.

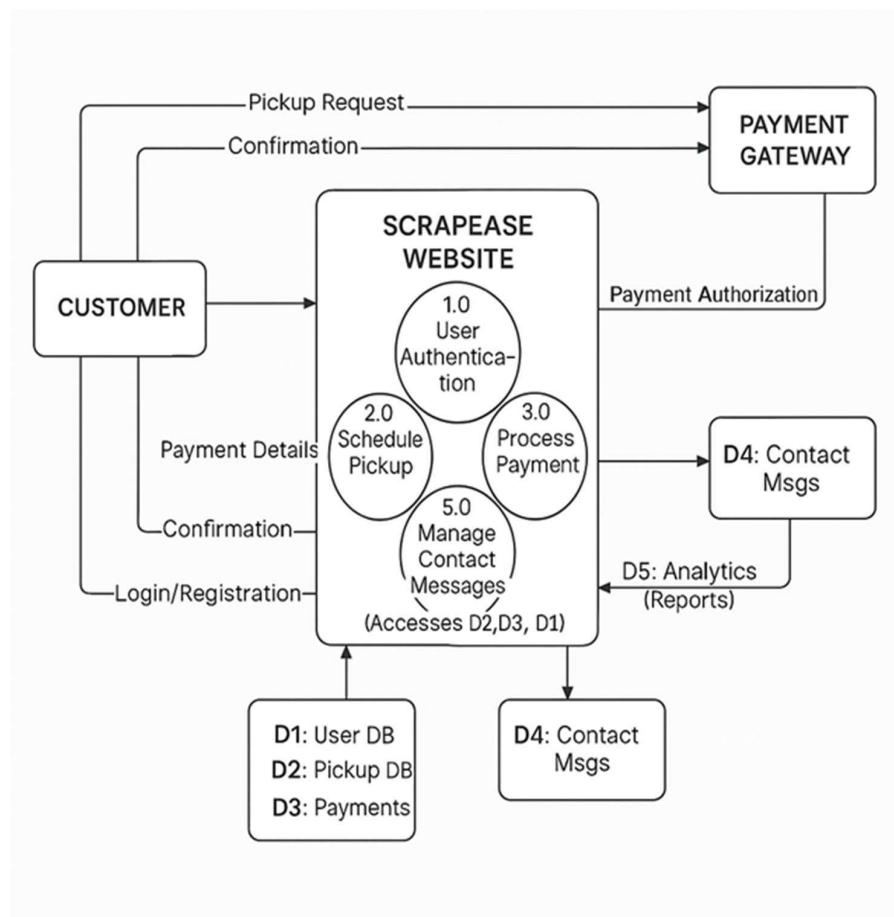


Fig 2.3. 2 Level DFD

2.2. Data Model

The Scrapease E-R diagram defines the fundamental data structure of the platform. It identifies key entities such as User, representing individuals or businesses selling scrap, and Vendor, representing the verified scrap buyers. Each entity has relevant attributes like name, contact information, and potentially location or specialization. The central entity connecting these two is Pickup Request, which stores crucial information about each transaction, including the requesting user, assigned vendor, pickup address, the types and estimated quantities of scrap materials involved, and the agreed-upon schedule. A related entity, Scrap Material, details the different categories and types of recyclable waste handled by the platform.

Furthermore, the E-R diagram models the financial aspect through the Payment entity. This entity links back to the Pickup Request, User, and Vendor, recording essential transaction details such as the amount, method of payment (initially likely cash, with potential for digital options), and the date of the transaction. The relationships established within the diagram are critical for maintaining data integrity and enabling the core functionalities of Scrapease, ensuring that the system accurately tracks user requests, vendor assignments, material details, pickup schedules, and payment records, thus facilitating a streamlined and transparent process for scrap collection and sale.

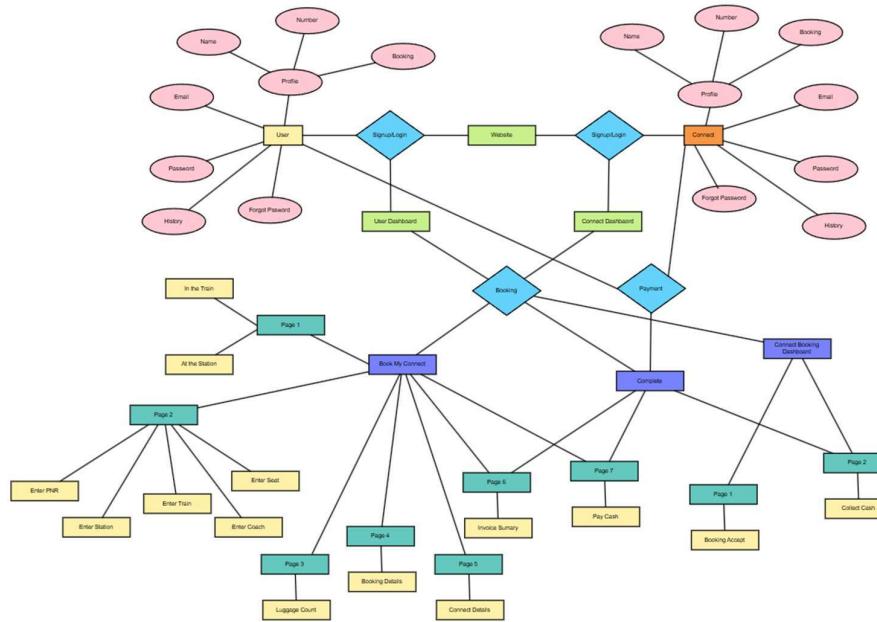


Fig 2.4. E-R Diagram

2

Fig 2.4. E-R Diagram

2.3. Process Flow Model

Scrapease streamlines scrap exchange: authenticated users (sellers) request pickups with material/location details and potential price estimates; verified vendors (buyers) accept, manage their service areas/availability, and complete pickups with payment and status updates. The platform features real-time notifications, a trust-building rating system, comprehensive admin control (user/vendor management, pricing, analytics, support), and a central database managing all data for efficient operation.

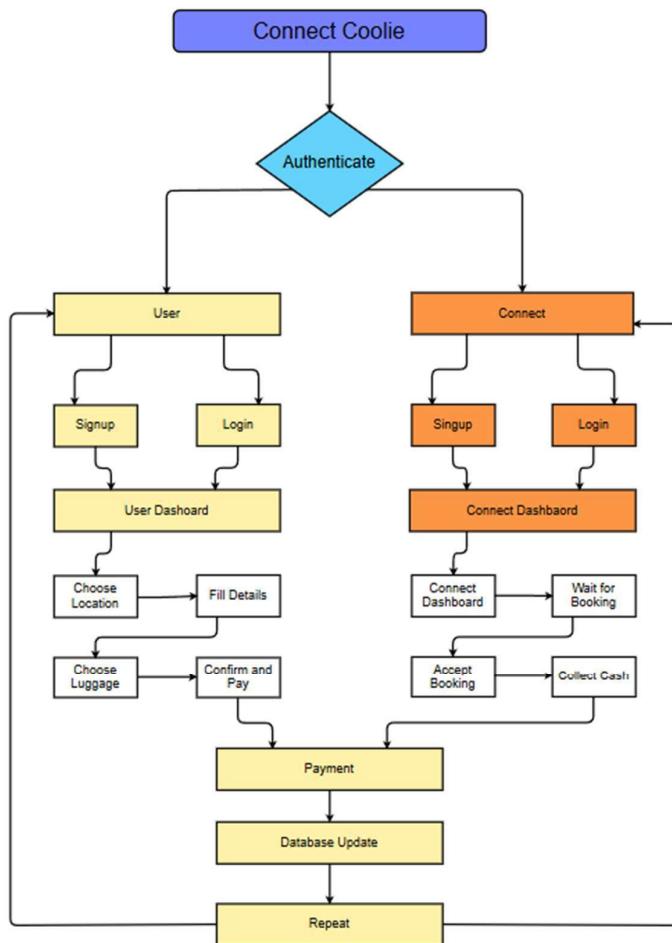


Fig 2.5. Process Flow Diagram

Fig 2.5. Process Flow Diagram

2.4. Behavioral Model

Scrapease's Behavioral Model shows customers (after login) requesting pickups with details, tracking status, and receiving payment after vendor collection. Vendors (after login and verification) view/accept requests, coordinate, collect scrap, pay customers, and update status. Admins manage users, categories, pricing, and provide support. The model outlines the sequential actions for each user type interacting with the platform.

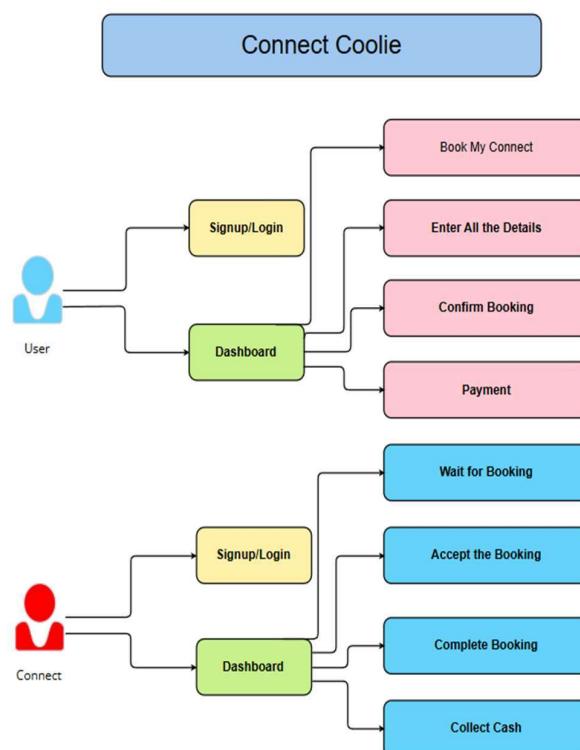


Fig 2.6. Behavioral Diagram

Fig 2.6. Behavioral Diagram

2.5. System Design

The Scrapease web application's system design outlines the collaboration of its components to serve both scrap sellers (Customers) and scrap buyers (Scrap Partners). Prioritizing scalability and maintainability, the system employs modular architecture. The front end, built with React.js, provides an intuitive user interface for scheduling pickups, tracking orders, and managing profiles. The backend, developed using Node.js, handles the core business logic, including user and vendor management, pickup request processing, matching vendors with requests, and potentially future payment integrations.

Data persistence is managed through a MySQL database. Secure authentication for both user types is implemented through session management or similar mechanisms. This design ensures a clear separation of concerns, facilitating efficient development and future enhancements to the Scrapease platform, ultimately aiming to streamline the scrap collection and recycling process.

2.5.1. Technical Feasibility

The Scrapease website leverages contemporary and reliable technologies to ensure scalable and secure development. The front end is built using React.js, a robust JavaScript library known for its component-based architecture and efficient UI rendering, providing a dynamic and user-friendly experience for both scrap sellers and vendors. Styling is achieved through CSS, allowing for a highly customized and unique visual experience aligned with the platform's branding.

The backend is powered by Node.js, a scalable JavaScript runtime environment capable of handling asynchronous operations efficiently, crucial for managing user and vendor interactions, pickup request processing, and potential future real-time communication features. Data persistence is handled by MySQL, a well-established relational database offering data integrity and structured querying for managing user data, vendor information, pickup requests, and transaction records. phpMyAdmin provides a user-friendly interface for database management.

Secure authentication for both users and vendors are implemented through session management or similar security methods, ensuring data privacy and platform integrity. The use of a modern Integrated Development Environment (IDE) like Visual Studio Code, along with Git for version control and a platform like GitHub for collaboration, ensures a streamlined and maintainable development process.

2.5.2.Operational Feasibility

The Scrapease website demonstrates strong operational feasibility by directly addressing the existing challenges and inefficiencies in the current informal scrap collection sector. The platform streamlines the process for individuals and businesses to find and connect with verified scrap vendors, eliminating the need for unreliable searches and unclear pricing. By supporting two primary roles – Customers (Scrap Sellers) and Scrap Partners (Vendors) – the system ensures that each user group interacts with features relevant to their needs. The intuitive web interface is designed for ease of use, minimizing the learning curve for both user types. Customers can conveniently schedule pickups by providing details about their scrap materials and location, while Scrap Partners can efficiently manage pickup requests and their service availability.

The platform's accessibility through various devices enhances convenience for all users, promoting wider adoption. By centralizing the connection process, Scrapease reduces the uncertainty and potential difficulties associated with traditional scrap disposal, offering a more organized and transparent solution that can be readily integrated into waste management practices. This makes the system not only feasible but also highly advantageous for simplifying the process of selling recyclable materials and fostering a more efficient and sustainable recycling ecosystem.

2.5.3.Economic Feasibility

The utilization of open-source technologies for Scrapease significantly reduces upfront costs, making the initial investment economically viable. Affordable hosting solutions further bolster its financial feasibility during the development and deployment phases.

The platform's long-term economic benefits stem from its ability to create a more efficient and transparent marketplace for recyclable materials. Sellers can easily connect with buyers, potentially securing better prices, while vendors gain access to a consistent supply of scrap, reducing their search costs and improving profitability. The potential integration of digital payments will further enhance transaction efficiency and security, solidifying Scrapease's economic advantages for all stakeholders.

CHAPTER 1 – INTRODUCTION

1.1. Literature Review

The Existing literature on waste management and recycling largely focuses on formal systems, with limited attention given to the application of digital platforms within the informal scrap sector. While studies acknowledge the importance of informal recycling, there's a gap in research exploring dedicated online solutions to connect waste generators directly with scrap vendors. However, insights from the burgeoning e-commerce and sharing economy offer valuable frameworks. The success of on-demand service platforms highlights the potential of digital interfaces to enhance efficiency, transparency, and accessibility by directly linking service providers and users. These models provide principles for user acquisition, trust-building, and seamless transactions, directly relevant to Scrapease's aim of connecting individuals and businesses with verified scrap vendors.

Furthermore, research in Human-Computer Interaction (HCI) and online platform design emphasizes the critical role of user-centricity. Intuitive navigation, clear information architecture, and reliable communication are vital for fostering user engagement and trust in online marketplaces. For Scrapease, understanding these principles is crucial to create an effective and user-friendly platform for both sellers and buyers of scrap materials. Studies on trust in online peer-to-peer marketplaces also underscore the need for mechanisms ensuring vendor reliability.

The current literature reveals a significant gap in digitally-driven solutions specifically tailored to the intricacies of the scrap collection and trading process. Scrapease aims to address this by creating a dedicated web-based platform, drawing inspiration from successful on-demand models while adapting to the specific needs of the recyclable materials market. Future research on user behavior within this ecosystem, effective vendor management, and the socio-economic impact of digitalizing this traditionally informal sector will be crucial for Scrapease's growth and its potential to promote sustainable waste management practices.

1.2. Problem Definition

Train individuals and businesses seeking to dispose of recyclable materials like metal, plastic, and paper frequently encounter challenges in connecting with reliable scrap vendors. These problems include the difficulty in locating verified buyers, a lack of transparent pricing for different materials, and the often time-consuming and informal nature of arranging for collection.

Existing methods for selling scrap are often decentralized, lack standardized processes, and can lead to uncertainty regarding fair pricing and reliable service. Furthermore, scrap vendors may face issues such as inconsistent supply, difficulties in reaching a wider network of potential sellers, and a lack of efficient logistical planning tools.

The absence of a structured system hinders the smooth flow of recyclable materials back into the recycling stream. The current lack of a dedicated digital platform to efficiently connect sellers of recyclable materials with verified scrap vendors contributes to a fragmented and often inefficient recycling process. Scrapease aims to address these problems by providing a user-friendly web platform to facilitate seamless booking and connection for scrap collection services, thereby promoting a more organized and sustainable approach to waste management.

1.3. Brief Introduction of Project

Scrapease is a web-based platform designed to revolutionize how individuals and businesses in India manage the disposal and sale of their recyclable materials, such as metal, plastic, and paper. Recognizing the common difficulties faced by users in finding dependable and fairly priced scrap vendors, Scrapease provides a user-friendly platform to seamlessly connect sellers with local, verified scrap buyers, whom we refer to as "Scrap Partners."

This project introduces a convenient online system that allows users to easily request a pickup by providing essential details such as the type and estimated quantity of scrap materials and their location. Registered users benefit from personalized experience, including pickup history and profile management. Simultaneously, "Scrap Partners" can register on the platform to receive and accept pickup requests, effectively managing their collection services and material acquisition.

By establishing a direct link between sellers and scrap vendors, Scrapease aims to bring transparency, reliability, and efficiency to the traditionally informal process of scrap collection and sale. The platform facilitates scheduled pickups and aims to provide a clear channel for communication and potential price negotiation. By digitizing this process, Scrapease lays the groundwork for a more organized and accessible scrap recycling ecosystem, starting with a focus on simplifying the responsible disposal and profitable sale of recyclable waste.

1.4. Proposed Modules

1.4.1. User Management Module

User Signup (Image 4): Enables new users (individuals and businesses) to create accounts by providing their name, email, and password.

User Login (Image 2): Allows registered users to securely access their accounts using their email and password.

User Profile Management (Implicit): While not explicitly shown as a dedicated page, this module would allow users to manage their contact information, pickup addresses, and potentially notification preferences.

1.4.2. Booking/Pickup Management Module

Scrap Category Display & Booking Initiation (Image 3): Presents the different categories of recyclable materials (Metal, Plastic, Paper) with potential pricing and a "Book Pickup" button to start the request process.

Pickup Request Details (Implicit): This would involve pages or forms where users specify the type and estimated quantity of scrap, preferred pickup date and time, and their location details.

Order Tracking (Image 5): Allows users to monitor the status of their scheduled pickups, including details like Order ID, material, weight, status (Pending, In Progress, Completed), and pickup date.

Pickup History (Implicit): Enables users to view a record of their past scrap pickups.

The Booking/Pickup Management module is central to the user experience, guiding sellers through the process of requesting scrap collection in a clear and structured manner. It encompasses the selection of scrap categories, the detailed input of pickup information, and the ability to track the progress of their requests. Users can also view and manage their past and current pickup orders through this module.

1.4.3. Vendor (Scrap Partner) Management Module (Dealer Dashboard - Image 7)

Vendor Signup (Implicit): Enables scrap vendors to register and create a profile on the platform, likely involving verification processes.

Vendor Login (Implicit): Allows registered vendors to securely access their dashboard.

Vendor Dashboard (Image 7): Provides an overview of total requests, pending requests, and completed requests. It also displays current material prices and a list of recent pickup requests with details like customer, material, weight, and status.

Request Management (Implicit from Dashboard): Enables vendors to view, accept, and manage incoming pickup requests.

Status Updates (Implicit from Dashboard): Allows vendors to update the status of pickups (e.g., Accepted, In Progress, Completed).

This module caters to the scrap vendors (Scrap Partners), enabling them to register and manage their presence on the platform. It allows vendors to log in, view and accept pickup requests, manage their schedules, and potentially update material prices. The vendor dashboard serves as their central hub for managing their operations on Scrapease.

1.4.4. Transaction/Information Display Module

Website Landing Page (Implicit): The main entry point for users, providing general information about Scrapease.

About Us (Implicit): Information about the platform's mission and goals.

Contact Us (Image 6): Provides contact information (email, phone, address) and a contact form for users and vendors to reach out.

How It Works (Image 1): Explains the step-by-step process of using Scrapease (Schedule a Pickup, Pickup at Your Address, Receive Payment).

This module provides essential content and communication channels for all users of the Scrapease platform. The landing page introduces the service, the "About Us" section provides more details, and the "Contact Us" page allows for inquiries and support. The "How It Works" section clearly outlines the user journey.

1.4.5.Pricing/Payment Information Module (Implicit)

Material Price Display (Image 3): Shows the current prices being offered for different categories of scrap materials (e.g., ₹45/kg for Aluminum Cans).

Invoice/Summary (Implicit): While not explicitly shown, this module would likely provide a summary of the collected scrap and the calculated payment to the user after the pickup is completed.

This module focuses on displaying pricing information for different recyclable materials to users. It would also handle the calculation and presentation of the payment amount due to the user based on the collected weight and material type.

1.5. Hardware Requirements

1.5.1. Minimum Requirements

For the Scrapease platform to function on the server-side, a basic setup would include at least a dual-core processor with a clock speed of 2 GHz, accompanied by a minimum of 4 GB of DDR4 RAM. The storage requirement starts at 40 GB, utilizing an SSD for better responsiveness. A standard 1 Gbps Ethernet connection is also necessary for network communication. For initial deployment, a basic Virtual Private Server (VPS) can serve as the hosting environment.

On the client-side, both users (individuals/businesses selling scrap) and "Scrap Partners" (vendors) would need a device with at least a single-core processor operating at 1 GHz, 1 GB of RAM, and sufficient free storage for the operating system and a web browser. The display should support a minimum resolution of 1024x768 pixels, and a basic internet connection (Wi-Fi or cellular data) along with a modern web browser are essential to access the platform.

1.5.2.Recommended Requirements

To ensure optimal performance and scalability for the Scrapease platform, the server infrastructure should ideally consist of a server-grade processor with at least four cores running at 2.5 GHz or higher, complemented by 8 GB or more of DDR4 RAM. The storage should be 80 GB or more of SSD in a RAID configuration for enhanced reliability and speed. A dedicated 1 Gbps Ethernet connection is preferred for consistent network performance. For hosting, a scalable VPS or an entry-level dedicated server or cloud instance would provide better resource management. On the client-side, a device equipped with a dual-core processor clocked at 1.5 GHz or higher and at least 2 GB of RAM is recommended for a smoother user experience. The display should support a resolution of 1366x768 or higher. A stable broadband internet connection (Wi-Fi, 4G, or 5G) and the latest version of a modern web browser are advised for optimal functionality and security.

1.6. Software Requirements

1.6.1.Frontend Technologies

The Scrapease website frontend utilizes React.js as the core JavaScript library for building the user interface. Its component-based architecture facilitates modular development, code reuse, and maintainability. Styling is primarily achieved through CSS, allowing for complete control over the visual design and a tailored user experience. Figma was likely used for the initial UI/UX design, providing a visual blueprint for the platform's layout and interactive elements. This combination of React.js for structure and custom CSS for styling enables a flexible and bespoke frontend development approach for Scrapease.

1.6.2.Backend and Database

The Scrapease website backend is built using Node.js, a JavaScript runtime environment that allows for server-side logic execution. This enables a consistent language across the frontend and backend development. The backend handles crucial functionalities such as user and vendor authentication, pickup request management, data processing, and communication with the database. Data persistence for the application is managed by MySQL, a robust relational database management system suitable for storing and retrieving user information, pickup details, vendor data, and other essential platform data. phpMyAdmin may be used as a web-based administration tool for the MySQL database, providing a user-friendly interface for database management and querying.

1.6.3.Authentication and Notification Services

Authentication for Scrapease likely employs JWT (JSON Web Tokens) stored in browser cookies to manage user and vendor sessions and implement role-based access control. This ensures that different user types (e.g., scrap sellers, scrap vendors, administrators) can only access relevant features and functionalities. For communication, Node Mailer or a similar service is likely used to handle email notifications such as account verification, password reset requests, pickup confirmations, and status updates. This enhances user and vendor communication and provides professional platform experience.

1.6.4.Development Tools

The Scrapease website's frontend interface is developed using React.js, leveraging its component-based structure for creating dynamic and interactive user interfaces. Styling is implemented using CSS, allowing for precise visual design and a unique platform aesthetic. Dynamic manipulation of UI elements within React.js components is achieved through interaction with the Virtual DOM, ensuring efficient updates and rendering. The backend logic, responsible for managing user and vendor data, pickup processes, and server-side operations, is built using Node.js, enabling scalable and efficient server-side development. Data for Scrapease is persistently stored and managed using MySQL, providing a structured and reliable way to organize information. The administration of the MySQL database is likely facilitated through phpMyAdmin. This combination of React.js and CSS for the frontend, and Node.js with MySQL for the backend, forms the technological foundation of the Scrapease web platform.

1.6.5.Browser and Operating System Compatibility

The Scrapease website prioritizes compatibility with the latest stable versions of major web browsers, including Google Chrome, Mozilla Firefox, Safari, and Microsoft Edge, to ensure optimal performance and rendering of its React.js frontend and CSS. This web-based approach inherently supports a wide range of operating systems, including Windows, macOS, Linux, and ChromeOS for desktop users, as well as Android and iOS for mobile users, provided they access the site through one of the recommended modern browsers. Users and vendors are encouraged to keep their browsers updated for the best experience and security.

CHAPTER 2 – SYSTEM ANALYSIS AND SPECIFICATION

2.1. Functional Model

The Scrapease platform operates on a functional model centered around connecting individuals and businesses needing to dispose of recyclable materials with registered scrap vendors ("Scrap Partners"). Users initiate the process by providing details about the type and quantity of their scrap and their location through an online booking/pickup request system. This request is then made available to verified "Scrap Partners" in their locality, who can accept the pickup request. Once accepted, the "Scrap Partner" schedules and carries out the pickup, and payment is facilitated based on the collected materials (details of the payment method would be defined in subsequent levels). The platform provides separate interfaces for users to manage their pickup requests and profiles, and for "Scrap Partners" to manage incoming requests and their service availability, facilitating an efficient and transparent exchange of services.

2.1.1. 0 Level DFD

The Level 0 DFD for Scrapease shows the entire website as one process. It illustrates how Customers (Scrap Sellers) send Pickup Schedule Requests and Login/Registration Data, receiving Pickup Confirmation in return. Scrap Partners (Vendors) would (implicitly in this diagram) send Login/Registration Data and receive Pickup Requests. The system also interacts with a Database (not explicitly shown in the provided "Connect Coolie" DFD but crucial for Scrapease) to fetch and store data. The provided DFD also shows interactions with a Payment Gateway (for potential online transactions) and a Scrapease Admin for platform management.

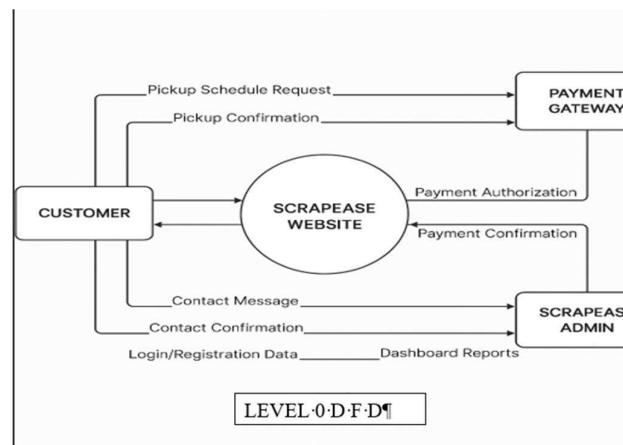


Fig 2.1. 0 Level DFD

2.1.2. 1 Level DFD

Scrapease's Level 1 DFD outlines key functions: User & Vendor Management (login/signup), Scrap Pickup Request Management (customer requests), Vendor Request Processing (vendor accepts), Pickup Scheduling & Tracking (status updates), Information & Support (platform info), and Payment Processing. Data flows between users, vendors, these modules, and the central Database to facilitate scrap pickups.

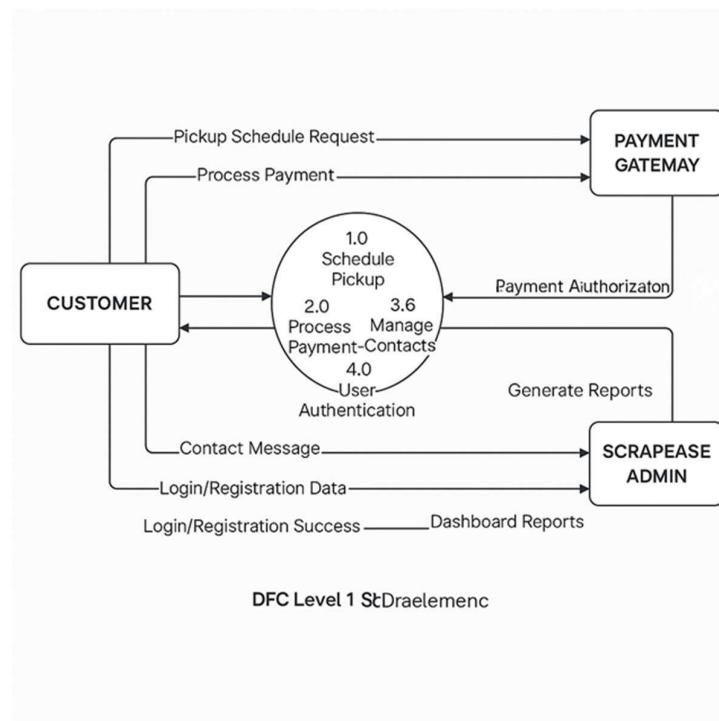


Fig 2.2. 1 Level DFD

2.1.3. 2 Level DFD

Scrapease's Level 2 DFD details user and vendor account management (signup, login, password recovery using respective databases), customer pickup request handling (submission, tracking, cancellation via Pickup DB), vendor processing of requests (viewing, accepting, updating Pickup DB), pickup scheduling and notifications, information and support via contact messages (Contact Msgs DB), and payment processing (recording payments, future gateway integration with Payments DB). These sub-processes illustrate the detailed data flow between users, vendors, the system's core functionalities, and specific data storage components.

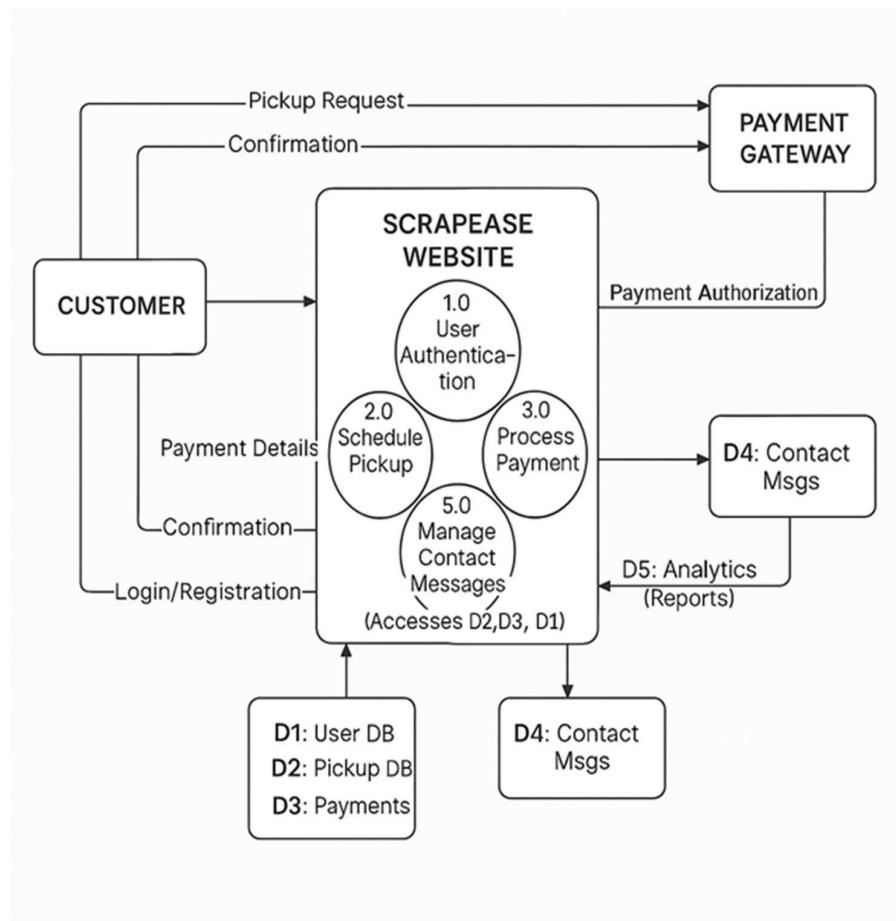


Fig 2.3. 2 Level DFD

2.2. Data Model

The Scrapease E-R diagram defines the fundamental data structure of the platform. It identifies key entities such as User, representing individuals or businesses selling scrap, and Vendor, representing the verified scrap buyers. Each entity has relevant attributes like name, contact information, and potentially location or specialization. The central entity connecting these two is Pickup Request, which stores crucial information about each transaction, including the requesting user, assigned vendor, pickup address, the types and estimated quantities of scrap materials involved, and the agreed-upon schedule. A related entity, Scrap Material, details the different categories and types of recyclable waste handled by the platform.

Furthermore, the E-R diagram models the financial aspect through the Payment entity. This entity links back to the Pickup Request, User, and Vendor, recording essential transaction details such as the amount, method of payment (initially likely cash, with potential for digital options), and the date of the transaction. The relationships established within the diagram are critical for maintaining data integrity and enabling the core functionalities of Scrapease, ensuring that the system accurately tracks user requests, vendor assignments, material details, pickup schedules, and payment records, thus facilitating a streamlined and transparent process for scrap collection and sale.

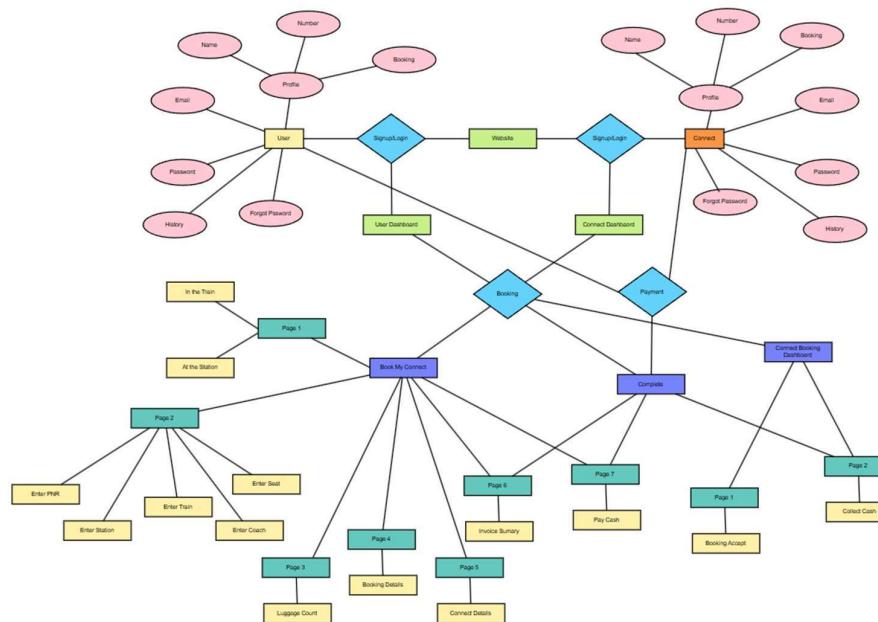


Fig 2.4. E-R Diagram

Fig 2.4. E-R Diagram

2.3. Process Flow Model

Scrapease streamlines scrap exchange: authenticated users (sellers) request pickups with material/location details and potential price estimates; verified vendors (buyers) accept, manage their service areas/availability, and complete pickups with payment and status updates. The platform features real-time notifications, a trust-building rating system, comprehensive admin control (user/vendor management, pricing, analytics, support), and a central database managing all data for efficient operation.

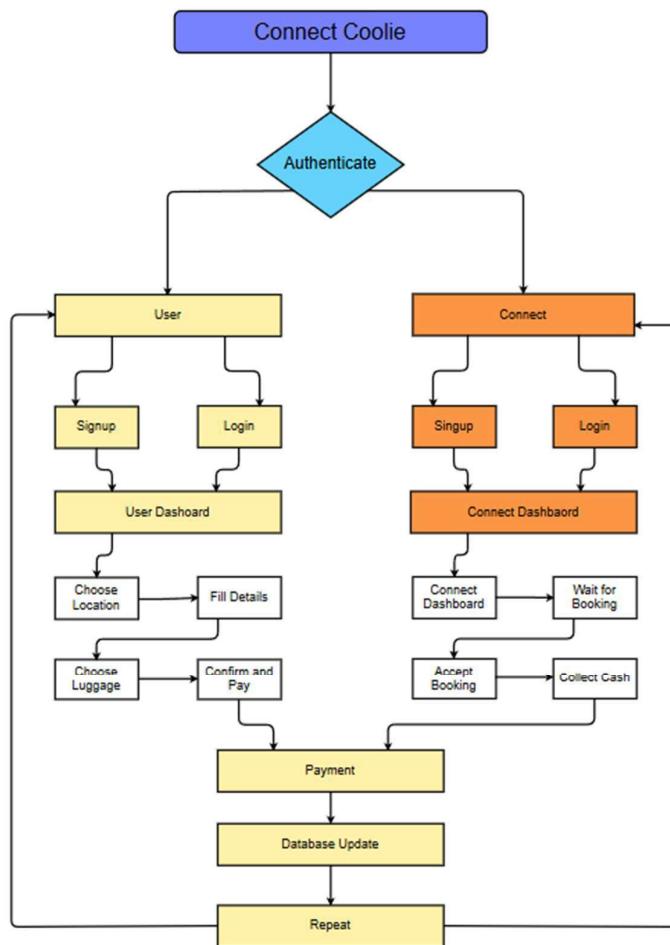


Fig 2.5. Process Flow Diagram

Fig 2.5. Process Flow Diagram

2.4. Behavioral Model

Scrapease's Behavioral Model shows customers (after login) requesting pickups with details, tracking status, and receiving payment after vendor collection. Vendors (after login and verification) view/accept requests, coordinate, collect scrap, pay customers, and update status. Admins manage users, categories, pricing, and provide support. The model outlines the sequential actions for each user type interacting with the platform.

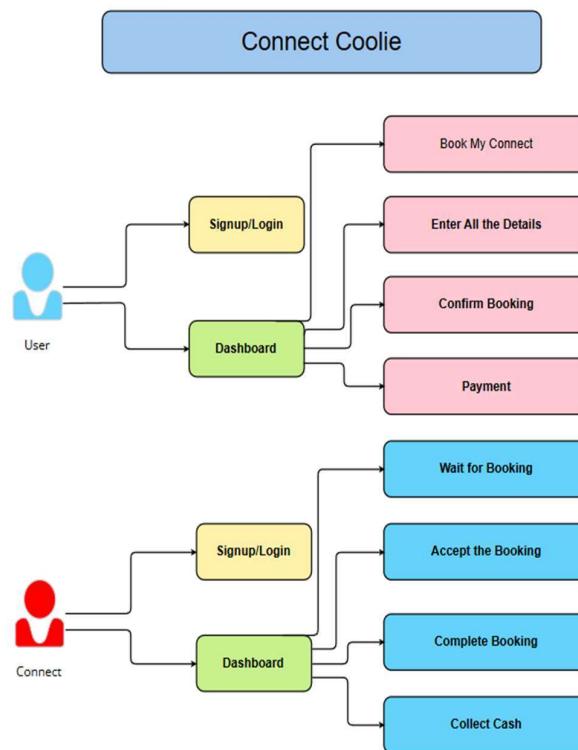


Fig 2.6. Behavioral Diagram

Fig 2.6. Behavioral Diagram

2.5. System Design

The Scrapease web application's system design outlines the collaboration of its components to serve both scrap sellers (Customers) and scrap buyers (Scrap Partners). Prioritizing scalability and maintainability, the system employs modular architecture. The front end, built with React.js, provides an intuitive user interface for scheduling pickups, tracking orders, and managing profiles. The backend, developed using Node.js, handles the core business logic, including user and vendor management, pickup request processing, matching vendors with requests, and potentially future payment integrations.

Data persistence is managed through a MySQL database. Secure authentication for both user types is implemented through session management or similar mechanisms. This design ensures a clear separation of concerns, facilitating efficient development and future enhancements to the Scrapease platform, ultimately aiming to streamline the scrap collection and recycling process.

2.5.1. Technical Feasibility

The Scrapease website leverages contemporary and reliable technologies to ensure scalable and secure development. The front end is built using React.js, a robust JavaScript library known for its component-based architecture and efficient UI rendering, providing a dynamic and user-friendly experience for both scrap sellers and vendors. Styling is achieved through CSS, allowing for a highly customized and unique visual experience aligned with the platform's branding.

The backend is powered by Node.js, a scalable JavaScript runtime environment capable of handling asynchronous operations efficiently, crucial for managing user and vendor interactions, pickup request processing, and potential future real-time communication features. Data persistence is handled by MySQL, a well-established relational database offering data integrity and structured querying for managing user data, vendor information, pickup requests, and transaction records. phpMyAdmin provides a user-friendly interface for database management.

Secure authentication for both users and vendors are implemented through session management or similar security methods, ensuring data privacy and platform integrity. The use of a modern Integrated Development Environment (IDE) like Visual Studio Code, along with Git for version control and a platform like GitHub for collaboration, ensures a streamlined and maintainable development process.

2.5.2.Operational Feasibility

The Scrapease website demonstrates strong operational feasibility by directly addressing the existing challenges and inefficiencies in the current informal scrap collection sector. The platform streamlines the process for individuals and businesses to find and connect with verified scrap vendors, eliminating the need for unreliable searches and unclear pricing. By supporting two primary roles – Customers (Scrap Sellers) and Scrap Partners (Vendors) – the system ensures that each user group interacts with features relevant to their needs. The intuitive web interface is designed for ease of use, minimizing the learning curve for both user types. Customers can conveniently schedule pickups by providing details about their scrap materials and location, while Scrap Partners can efficiently manage pickup requests and their service availability.

The platform's accessibility through various devices enhances convenience for all users, promoting wider adoption. By centralizing the connection process, Scrapease reduces the uncertainty and potential difficulties associated with traditional scrap disposal, offering a more organized and transparent solution that can be readily integrated into waste management practices. This makes the system not only feasible but also highly advantageous for simplifying the process of selling recyclable materials and fostering a more efficient and sustainable recycling ecosystem.

2.5.3.Economic Feasibility

The utilization of open-source technologies for Scrapease significantly reduces upfront costs, making the initial investment economically viable. Affordable hosting solutions further bolster its financial feasibility during the development and deployment phases.

The platform's long-term economic benefits stem from its ability to create a more efficient and transparent marketplace for recyclable materials. Sellers can easily connect with buyers, potentially securing better prices, while vendors gain access to a consistent supply of scrap, reducing their search costs and improving profitability. The potential integration of digital payments will further enhance transaction efficiency and security, solidifying Scrapease's economic advantages for all stakeholders.

CHAPTER 3 – MODULE IMPLEMENTATION AND SYSTEM INTEGRATION

The Scrapease project follows modular architecture, with distinct sections dedicated to Customer (Scrap Seller) functionalities and Scrap Partner (Vendor) functionalities, ensuring clarity and maintainability of the codebase.

The implementation phase involved building these modules using React.js for the user interface and Node.js for the backend logic. CSS was written to create a tailored and responsive design that caters to both desktop and mobile users. The backend interacts with a MySQL database to manage user data, vendor information, pickup requests, material categories, and potentially pricing details. Secure authentication mechanisms are implemented to manage access for both scrap sellers and vendors.

System integration is crucial to ensure seamless operation of Scrapease. This involves the smooth flow of pickup requests from Customers to available Scrap Partners, the updating of pickup statuses (e.g., pending, accepted, completed), and the recording of completed pickups and potential payment information in the database. The frontend and backend communicate to provide a real-time experience for both Customers (allowing them to track their requests) and Scrap Partners (allowing them to manage available pickups). Thorough testing is conducted to ensure that the integrated system functions effectively, enabling scrap sellers to easily connect with vendors and vendors to efficiently manage collection requests and their service.

3.1. Signup/Login Process Module

The Scrapease Signup/Login module allows both Customers and Scrap Partners to securely access the platform. Customer Signup (Image 2) requires Full Name, Email, and Password for new accounts. Customer Login (Image 1) uses Email and Password for returning users. Similar (implicit) signup and login processes exist for Scrap Partners, potentially with additional business details and verification. This module ensures secure access for all user roles to manage their profiles and utilize the platform's features.

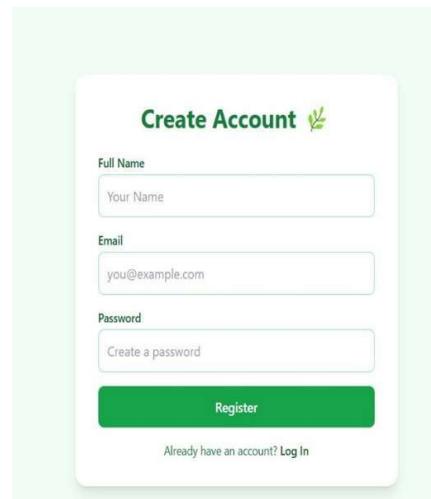


Fig 3.1. Signup Authentication Module

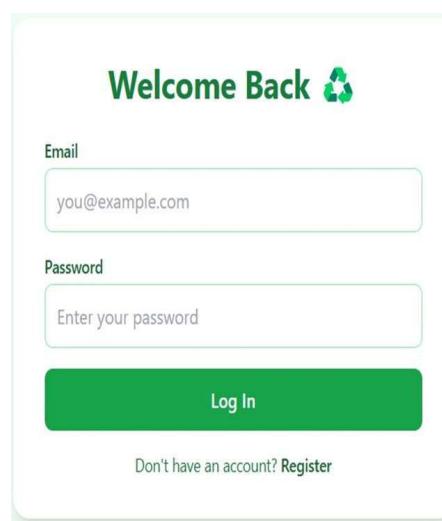


Fig 3.1.1. Signup Authentication Module

3.2. How it Works Module

The "How It Works" module on Scrapease (Image 1) clearly outlines the user journey in three simple steps. First, users "Schedule a Pickup" by browsing scrap categories (Image 3), selecting their materials, and providing details like quantity, preferred time, and address. This ensures efficient collection arrangement.

Second, "Pickup at Your Address" highlights the convenient door-to-door service where a verified Scrap Partner collects the recyclables directly from the user. Finally, "Receive Payment" signifies the user being compensated for their scrap, initially likely in cash, emphasizing the direct value conversion.

This module effectively communicates the ease and benefit of using Scrapease for a transparent and rewarding recycling experience.

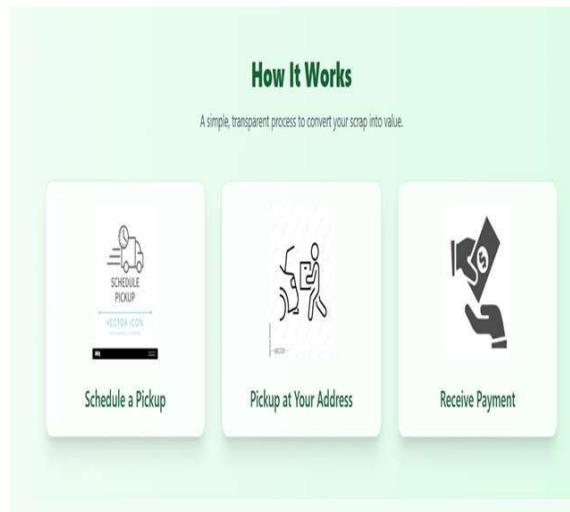


Fig 3.2. How it Works Module

3.3. Scrap Categories Module

The Scrap Categories module on Scrapease (Image 3) is thoughtfully designed to guide users through the types of recyclable materials accepted by the platform. Organized into logical groupings like Metals, Plastics, and Paper, the module presents a visually clear and easily navigable interface.

Each specific scrap item within these categories, such as Aluminum Cans or Plastic Bottles, is accompanied by a representative image, its classification (e.g., Metal, Plastic), and the current price offered per kilogram. This immediate display of pricing ensures transparency and allows users to quickly assess the potential value of their scrap.

The inclusion of a prominent "Book Pickup" button alongside each listed scrap item is a key feature of this module. This direct call to action streamlines the user experience by allowing them to initiate a pickup request for a specific material with a single click. Upon selecting "Book Pickup," users would likely be directed to a form or a series of pages where they can provide further details necessary for the collection, such as the estimated quantity of the chosen material, their preferred pickup date and time, and their location.

The Scrap Categories module thus serves as a crucial bridge between user identification of recyclable materials and the commencement of the scrap collection service offered by Scrapease.

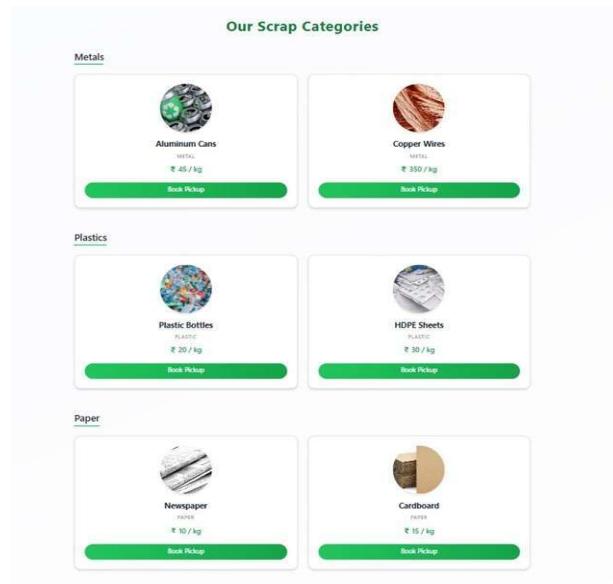


Fig 3.3. Scrap Categories Module

3.4. Track Module

The Track Orders module (Image 4) provides a transparent and informative interface for Scrapease customers to monitor the status of their scheduled scrap pickups. Upon accessing this module, users are presented with a clear list of their past and current orders, organized in a tabular format for easy readability. Each row in the table represents a specific pickup request and displays key details associated with that order.

The information presented for each order includes a unique "Order ID" for easy reference, the name of the "Customer" who placed the request, the type of "Material" scheduled for pickup (e.g., Aluminum Cans, Plastic Bottles, Cardboard), the recorded "Weight" of the material (in kilograms), the current "Status" of the order (e.g., Pending, In Progress, Completed), and the scheduled "Pickup Date." The "Status" field is particularly useful, providing users with real-time updates on the progress of their pickup. This module empowers customers with visibility into their scrap collection process, fostering trust and ensuring positive user experience by keeping them informed at every stage.

Track Your Orders					
Order ID	Customer	Material	Weight	Status	Pickup Date
ORD1245	Rahul Sharma	Aluminum Cans	15 kg	Pending	2025-05-10
ORD12346	Priya Singh	Plastic Bottles	8 kg	In Progress	2025-05-06
ORD12347	Amit Kumar	Cardboard	12 kg	Completed	2025-04-28

Fig 3.4. Track Module

3.5. Dealer Dashboard Module

The Dealer Dashboard module (Image 5), intended for Scrap Partners (vendors) on the Scrapease platform, serves as a central hub for managing their operations and engaging with pickup requests. Upon logging in, vendors are presented with a comprehensive overview of their activity, including key metrics displayed prominently at the top. These metrics include the "Total Requests" received, the number of "Pending Requests" requiring their action, and the number of "Completed Requests," providing a quick snapshot of their workload and performance.

Below the key metrics, the dashboard features a section displaying "Current Material Prices (₹/kg)" in a graphical format. This bar chart visually represents the current rates offered for different categories of scrap materials such as Aluminum, Copper, Plastic, and Paper. This information is crucial for vendors to make informed decisions when accepting pickups and potentially communicating pricing with customers.

Further down, a "Recent Pickup Requests" section lists incoming requests with essential details like the "Request ID," the "Customer" name, the type of "Material," the estimated "Weight," and the current "Status" of the request. This allows vendors to quickly assess and act upon new pickup opportunities, streamlining their workflow and enabling efficient management of their scrap collection services.

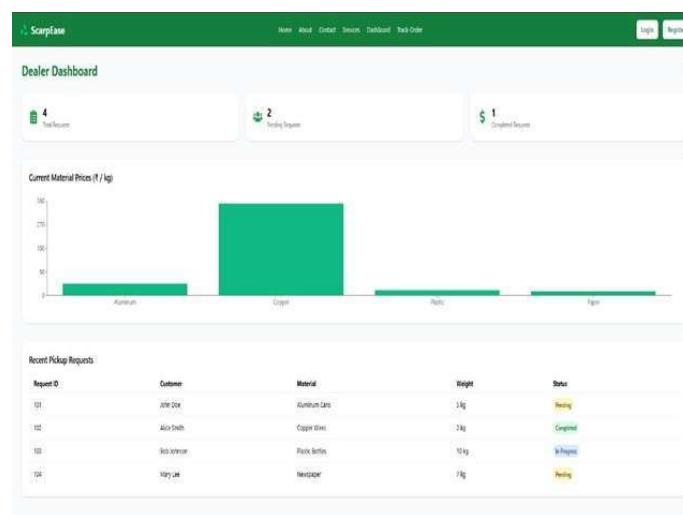


Fig 3.5. Dealer Dashboard Module.

3.6. Contact Us Module

The Contact Us module (Image 6) provides a direct communication channel for users and potential vendors to reach out to the Scrapease team for inquiries, support, or feedback. The page prominently displays essential contact information, including the official support email address (support@scrapease.in), a direct phone number (+91-9876543210), and a physical address (2nd Floor, GreenTech Building, Lucknow, India). This readily available information caters to users who prefer direct communication through these traditional methods.

In addition to the direct contact details, the module features a user-friendly contact form. This form includes fields for the user to enter their Name, Email address, and a Message detailing their inquiry or feedback. A clear "Send Message" button allows users to submit their communication directly through the website. This form provides a convenient way for users to get in touch without needing to leave the platform. The inclusion of both direct contact information and a contact form ensures that Scrapease offers multiple accessible avenues for user communication and support.

The screenshot shows a contact form titled "Contact Us". At the top, there is contact information: Email: support@scrapease.in, Phone: +91-9876543210, and Address: 2nd Floor, GreenTech Building, Lucknow, India. Below this, there are three input fields: "Name" with placeholder "Your Name", "Email" with placeholder "you@example.com", and "Message" with placeholder "Your message...". At the bottom is a green "Send Message" button.

Fig 3.6. Contact Us Module

CHAPTER 4 – TESTING AND EVALUATION

Testing is a crucial phase to ensure Scrapease functions as intended and meets the needs of both scrap sellers (Customers) and scrap buyers (Scrap Partners). Given the platform's role in connecting users and facilitating transactions of recyclable materials, various testing techniques were employed to verify its functionality, usability, and security.

4.1. Types of Testing performed

4.1.1. Unit Testing

Individual components, functions, and API routes within the Scrapease backend (built with Node.js) and frontend (built with React.js) were tested in isolation. For example, functions related to user and vendor authentication, pickup request handling, and vendor availability updates were tested independently to ensure they performed their logic correctly.

4.1.2. Integration Testing

Once individual units were verified, the integration between different modules was tested. For instance, the flow of a customer submitting a pickup request and it being displayed to available scrap partners was tested. Similarly, the interaction between the frontend pickup request form, the backend processing logic, and the MySQL database was validated to ensure data consistency regarding user information, pickup details, and material types.

4.1.3. Functional Testing

Core functionalities for both user roles were tested to ensure they met the specified requirements. This included customer signup and login, vendor signup and login, customer pickup request submission, vendor availability management, vendor accepting a pickup, and the basic information displayed to both parties during the pickup process (e.g., material type, estimated weight, pickup location). These tests were conducted manually through the browser, simulating real user interactions.

4.1.4. System Testing

The complete Scrapease system, including the frontend UI, backend logic, authentication, and MySQL database, was tested as a unified whole.

This involved testing end-to-end scenarios, such as a new customer signing up, making a pickup request, a vendor accepting it, and the simulated completion of the service (given the current likely cash-based payment). Edge cases, like incomplete pickup submissions or simultaneous user access, were also explored.

4.1.5. Security Testing

Basic security considerations for user and vendor authentication (signup and login credentials) were addressed. Input validation on forms was implemented to prevent basic data integrity issues. Future iterations with integrated payment gateways would necessitate thorough security testing.

4.2. Evaluation Criteria

4.2.1. Accuracy

Accuracy in Scrapease was evaluated by ensuring that the information displayed to customers and scrap partners (e.g., pickup details, material types, vendor availability) correctly reflects the data entered and stored in the system. Test cases verified that pickup information was accurately transmitted and stored.

4.2.2. Efficiency

The responsiveness of the Scrapease website was assessed, including page load times, the speed of pickup request submissions, and the efficiency of updating vendor availability. Backend queries to the MySQL database were considered for optimization.

4.2.3. Usability

The ease of use and intuitiveness of the Scrapease interface for both scrap sellers and vendors were evaluated. This involved observing user interactions and gathering feedback (if available) on the clarity of the pickup request process, dashboard navigation, and overall user experience.

4.2.4. Reliability

The stability of the Scrapease platform under normal and potentially stressed conditions was checked. Error handling for invalid inputs and basic resilience was considered.

4.2.5.Security

Basic security measures for user and vendor authentication were evaluated. Future development involving sensitive data or payment gateways would require more extensive security testing.

4.2.6.Scalability

The underlying architecture (React.js frontend, Node.js backend, MySQL database) was considered for its potential to handle increased user loads, vendor registrations, and pickup requests in future expansions.

4.2.7.Maintainability

The modular structure of the React.js frontend and Node.js backend, along with clear code organization, contributes to the maintainability of the Scrapease codebase for future updates and enhancements, such as adding new scrap categories or integrating payment options.

This evaluation confirms that Scrapease provides a functional and usable platform for connecting scrap sellers and vendors. Ongoing testing and evaluation will be crucial for ensuring the platform's continued reliability, efficiency, and security as it evolves and potentially integrates more features.

CHAPTER 5 – TASK ANALYSIS AND SCHEDULE OF ACTIVITIES

5.1. Task Decomposition

The development of Scrapease was broken down into smaller, manageable tasks to facilitate efficient development, clear assignment of responsibilities, and progress tracking. These tasks were primarily organized around the functionalities for Customers (Scrap Sellers), Scrap Partners (Vendors), and the core platform operations.

5.1.1. Requirement Analysis

- Identify the needs of individuals and businesses seeking to sell scrap and scrap vendors looking for material sources in Lucknow.
- Define functional requirements (user/vendor registration, pickup request process, vendor availability management, material category display, potential pricing information) and non-functional requirements (usability, reliability, transparency).
-

5.1.2. System Design

- Define the initial database schema using MySQL to store user, vendor, pickup request, and material data.
- Outline the customer and vendor access flows and basic system architecture, including frontend and backend interactions.

5.1.3. Frontend Development

- Build the user interface using React.js and custom CSS for styling a user-friendly and responsive design.
- Implement interactive elements for forms (signup, login, pickup request), dashboards (customer order tracking, vendor request management), and listings (scrap categories, vendor availability).
- Develop responsive pages for signup, login, scheduling pickups, tracking orders, and the vendor dashboard displaying requests and material prices.

5.1.4. Backend Development

- Use Node.js to create server-side logic and APIs to handle requests from the frontend.
- Implement user and vendor authentication and authorization.
- Develop APIs for handling pickup requests (submission, retrieval, acceptance), updating vendor availability, retrieving scrap categories and potential pricing, and managing user and vendor data.

5.1.5. Database Configuration

- Implement the signup and login processes for both Customers (scrap sellers) and Scrap Partners (vendors).
- basic dashboards for Customers to view their pickup requests and for Scrap Partners to view/manage incoming requests.

5.1.6. User and Scrap Partner Management

- Implement the signup and login processes for both user types.
- Create basic dashboards for users to view bookings and for connects to view/manage requests.

5.1.7. Pickup Management

- Develop the core pickup process flow, allowing Customers to input scrap details and submit requests.
- Implement the functionality for Scrap Partners to view and accept pickup requests.

5.1.8. Testing

- Perform basic unit tests on key functions and components.
- Conduct integration testing between the frontend pickup process and the backend pickup management.
- Perform manual functional testing of Customer and Scrap Partner workflows.

5.1.9. Deployment

- Deploy the website on a suitable hosting platform.
- Configure basic environment settings.

5.1.10. Documentation

- Create basic project documentation outlining the architecture and user flows.
- Write developer-level documentation for future maintenance.

5.2. Project Schedule

SDLC Activities	SEP	OCT	NOV	DEC	JAN	FEB	MAR	APR
Planning								
Requirement Analysis								
System Design								
Frontend								
Backend								
Testing								
Deployment								

Table 5.2. Project Schedule

5.3. Task Specification

Task specification involves clearly defining each major development activity in terms of its objectives, required resources, expected outcomes, and how it connects to other tasks. This helps manage time and resources efficiently while maintaining clarity of responsibility and progress tracking for the Scrapease project.

5.3.1. Planning

- **Planning:** Define project roadmap, allocate development team resources, and outline the initial project timeline (4-5 days).
- **Requirement Analysis:** Thoroughly understand the needs of scrap sellers (Customers) and scrap buyers (Scrap Partners) in the target region (Lucknow, Uttar Pradesh, India), and define the essential features for the platform (3 weeks).

- **System Design:** Design basic wireframes and user interface mockups for both customer and vendor platforms, outline the initial database structure using MySQL, and map out the fundamental user and vendor interaction flows (2 months).
- **Frontend:** Build the user interfaces for both Customers and Scrap Partners using React.js and custom CSS for styling and responsiveness (4.5 months).
- **Backend:** Develop the server-side logic and APIs using Node.js to handle core functionalities such as user/vendor authentication, pickup request processing, and data management (3 months).
- **Database:** Define the necessary tables and relationships within the MySQL database schema to store and manage user data, vendor information, pickup requests, and material details, and implement basic data storage and retrieval operations (3 weeks).
- **User/Scrap Partner Management:** Implement the signup and login processes for both Customer and Scrap Partner roles, and create basic dashboards for each to manage their respective activities (1 month).
- **Pickup Management:** Develop the functionality for Customers to submit detailed pickup requests (including material types, quantities, and location) and for Scrap Partners to view and accept these requests (2 months).
- **Testing:** Conduct basic unit tests on individual functions, integration tests between frontend and backend modules, and manual functional testing of core Customer and Scrap Partner workflows (3 months).
- **Deployment:** Prepare and deploy the Scrapease website on a suitable hosting platform (3 weeks).

CHAPTER 6 – PROJECT MANAGEMENT

Project management for Scrapease involved organizing, planning, executing, and monitoring the development process to meet the specific objectives of connecting scrap sellers (Customers) with scrap buyers (Scrap Partners) within a defined scope and timeline.

The project followed a phased approach, moving from understanding user needs to design, development, testing, and deployment of the web platform. Basic task breakdowns and timelines were utilized to track progress across the frontend (React.js), backend (Node.js), and database (MySQL) components. While not strictly adhering to a formal Agile methodology, an iterative approach allowed flexibility and adjustments based on initial testing and feedback. Simple task management tools were used to monitor progress and responsibilities.

Effective project management aimed to deliver a functional Scrapease platform that addressed the core user needs and facilitated a more efficient scrap collection and recycling process.

6.1. Major risks and contingency plans

Identifying potential risks early was crucial for the smooth development of "Connect Coolie." Below are some major risks and their contingency plans:

6.1.1. Requirement Misunderstanding

- **Risk:** Miscommunication about the exact needs of travelers and porters could lead to an inadequate platform.
- **Contingency Plan:** Conduct direct communication with potential users and porters to gather clear requirements and validate the developed features.

6.1.2. Technical Integration Issues

- **Risk:** Integrating the React.js frontend with the Node.js backend and the MySQL database might encounter compatibility or data flow problems.
- **Contingency Plan:** Implement and test API interactions early and frequently. Allocate time for debugging and ensure clear data structures.

6.1.3.Authentication & Security Failures

- **Risk:** Issues with user and connect login security could lead to unauthorized access.
- **Contingency Plan:** Implement secure authentication practices for both user types.

6.1.4.Data Loss or Corruption

- **Risk:** Potential data loss in the MySQL database.
- **Contingency Plan:** Implement regular database backups

6.1.5.Delay in Task Completion

- **Risk:** Unforeseen technical challenges or resource constraints might delay development.
- **Contingency Plan:** Maintain a flexible timeline and prioritize core functionalities.

6.1.6.Payment Process Issues (Future)

- **Risk:** If a payment gateway is integrated later, issues with its implementation could arise.
- **Contingency Plan:** Thoroughly research and test the payment gateway integration in a separate environment.

6.2. Principle Learning Outcomes

- The development of "Connect Coolie" provided several learning opportunities

6.2.1.End-to-End Web Development

Gaining practical experience in building a full-stack web application using React.js for the frontend and Node.js for the backend.

6.2.2. Role-Based Authentication & Authorization

Implementing secure login mechanisms for both travelers and porters.

6.2.3. Database Interaction

Understanding how to design and interact with a MySQL database for storing and retrieving application data.

6.2.4. API Development

Creating backend APIs using Node.js to handle frontend requests and manage data.

6.2.5. Error Handling and Debugging

Identifying and resolving issues during the development process.

6.2.6. Project & Time Management Skills

Learning to break down tasks and manage the development timeline.

6.2.7. Software Testing and Quality Assurance

Conducting initial unit, integration, and functional testing.

6.2.8. Collaboration and Communication

Working effectively within a development team.

6.2.9. Scalability and Maintainability Principles

Considering basic principles for future growth and updates.

6.2.10. User-Centric Development Approach

Designing interfaces and features with the needs of both travelers and porters in mind.

REFERENCES

- [1]. Sundararajan, A. (2016).
The Sharing Economy: The End of Employment and the Rise of Crowd-Based Capitalism. MIT Press. (Broader context of on-demand platforms).
- [2]. Botsman, R., & Rogers, R. (2010).
What's Mine Is Yours: The Rise of Collaborative Consumption. HarperCollins. (Principles of collaborative models).
- [3]. Zervas, G., Proserpio, D., & Byers, J. W. (2017).
The Rise of the Sharing Economy: Implications and Opportunities. Annual Review of Resource Economics, 9(1), 687-717. (Academic overview of the sharing economy).
- [4]. O'Reilly, T. (2011).
What's the future of work? O'Reilly Media. (Discusses the changing nature of work and online platforms).
- [5]. Nielsen, J. (1994).
Usability Engineering. Morgan Kaufmann. (Principles of user-centered design)ser-Centric Design.
- [6]. Norman, D. A. (2013).
The Design of Everyday Things. Revised and Expanded Edition. Basic Books. (Core concepts of good design).
- [7]. React Documentation.
Retrieved from react.dev (Frontend technology reference).
- [8]. Node.js Documentation.
Retrieved from nodejs.org (Backend technology reference).

[9]. MySQL Documentation.

Retrieved from dev.mysql.com (Database technology reference).

[10]. MDN Web Docs.

Retrieved from developer.mozilla.org (Web development standards).