A

Mini Project Report

on

ParentShieldAI

Submitted in partial fulfillment of the requirements for the

degree

Third Year Engineering – Computer Science Engineering (Data Science)

by

Nishigandha Sawant	**23107047**
Nidhi Shettigar	**23107041**
Abhishek Sali	**23107004**

Under the guidance of

Ms. Poonam Pangarkar

DEPARTMENT OF COMPUTER SCIENCE ENGINEERING (DATA SCIENCE)

A.P. SHAH INSTITUTE OF TECHNOLOGY

G.B. Road, Kasarvadavali, Thane (W)-400615

UNIVERSITY OF MUMBAI

Academic year: 2025-26

CERTIFICATE

This to certify that the Mini Project report on **ParentShieldAI** has been submitted by Nishigandha Sawant(23107047), Nidhi Shettigar(23107041), and Abhishek Sali(23107004) who are bonafide students of A. P. Shah Institute of Technology, Thane as a partial fulfillment of the requirement for the degree in **Computer Science Engineering (Data Science)**, during the academic year **2025-2026** in the satisfactory manner as per the curriculum laid down by University of Mumbai.

Ms. Poonam Pangarkar
Guide

Dr. Pravin Adivarekar
HOD, CSE(Data Science)

Dr. Uttam D. Kolekar
Principal

External Examiner:
1.

Internal Examiner:
1.

Place: A. P. Shah Institute of Technology, Thane
Date:

ACKNOWLEDGEMENT

This project would not have come to fruition without the invaluable help of our guide **Ms. Poonam Pangarkar**. Expressing gratitude towards our HoD, **Dr. Pravin Adivarekar**, and the Department of Computer Science Engineering (Data Science) for providing us with the opportunity as well as the support required to pursue this project. We would also like to thank our project coordinator **Ms. Aavani Nair** and **Ms. Richa Singh** who gave us her valuable suggestions and ideas when we were in need of them. We would also like to thank our peers for their helpful suggestions.

TABLE OF CONTENTS

Abstract

1. Introduction……..1

   1.1.Purpose.....1

   1.2.Problem Statement……...2

   1.3.Objectives..... 2

   1.4.Scope.....…...3

2. Literature Review…….....4

3. Proposed System…….....7

   3.1. Features and Functionality……...7

4. Requirements Analysis..... 9

5. Project Design.....10

   5.1.Use Case diagram…… …...10

   5.2.DFD (Data Flow Diagram) ……11

   5.3.System Architecture…….12

   5.4.Implementation…….13

6. Technical Specification……..…….....14

7. Project Scheduling.....15

8. Results……..…….16

9. Conclusion…….....17

10. Future Scope…….18

References

ABSTRACT

ParentShieldAI is a unified web-based platform designed to protect Indian parents and less tech-savvy users from digital fraud through real-time threat detection and educational resources. The system addresses the critical gap in digital safety by providing on-demand analysis of suspicious messages and financial transactions at the moment of decision-making.

The platform consolidates message analysis, transaction verification, and a comprehensive Digital Safety Library into a single, intuitive interface accessible through web browsers.

Developed using React.js for the frontend and Flask for the backend, with Scikit-learn for machine learning implementation, the system provides immediate verdicts on potential threats with clear explanations, empowering users to make informed decisions.

Chapter 1

Introduction

The internet and a vast ecosystem of connected devices have become truly integral to modern childhood development, shaping how children learn, socialize, and interact with the world. This digital immersion, however, does not come without significant risk. Young users are now exposed to a complex and rapidly evolving spectrum of dangers that extend far beyond simple inappropriate content. The core problem this project addresses is the lack of a centralized, intelligent, and, most importantly, accessible resource for parents to effectively understand and mitigate these contemporary digital safety threats.

Traditional safety guides, often distributed as static pamphlets or blog posts, are fundamentally ill-equipped for this challenge. They become outdated almost as soon as they are published, failing to utilize modern analytical tools or account for the dynamic nature of online risks. Today's threats are global, subtle, and increasingly psychological. They involve sophisticated social engineering, personalized phishing campaigns that prey on a family's specific data, and emerging forms of digital fraud, making basic parental supervision or simple content filters insufficient. There exists a profound and widening knowledge gap between digital natives (children), who intuitively navigate these spaces, and their guardians, who are often left trying to apply outdated safety models to a new and invisible battlefield.

This project is motivated by the urgent necessity to bridge this critical gap. We aim to provide an innovative, all-in-one solution that not only demystifies complex digital security concepts but also makes them actionable for everyday parents. This mission aligns directly with global calls for responsible technology use, the push for greater digital inclusion, and the fundamental need to empower parents to protect their families in an increasingly connected world.

1.1 Purpose:

The primary purpose of ParentShieldAI is twofold. First, it seeks to establish a centralized, intelligent, and readily accessible educational platform that equips parents with up-to-date

digital literacy. This education is not delivered through passive articles alone, but through interactive and engaging formats, such as a quiz-based game feature, designed to build and test knowledge retention in a more compelling way.

Second, the project will utilize Artificial Intelligence (AI) for proactive, real-time threat analysis and content simplification. The goal is to create a practical tool that parents can use at the moment of need. This AI is being designed to not only analyze suspicious text but to also intelligently parse screenshots of messages to identify and predict the malicious intent of embedded URLs or links, a common vector for fraud. Furthermore, to maximize accessibility across India, the platform is being built with multilingual capabilities, enabling it to understand and analyze threat messages written in Hindi and Marathi, not just English.

By achieving this, the project aims to bridge the current knowledge and accessibility disparity between parents and the complex, rapidly evolving landscape of online risks, thereby directly contributing to a safer and more informed digital ecosystem for children.

1.2 Problem Statement:

Target Audience:

Indian parents who are increasingly dependent on digital platforms for banking, communication, and education but remain highly vulnerable to online fraud and deception. This audience is linguistically diverse, with many more comfortable in regional languages than in English, and often lacks a foundational vocabulary for digital security.

The Critical Gap:

Current security solutions operate passively in the background (like antivirus software) or provide generic, unengaging educational content (like static blogs). There exists no accessible, user-friendly platform that empowers parents with real-time, on-demand analysis of suspicious messages, screenshots, and transactions at the moment of decision-making. This gap is significantly widened by a language barrier, as the vast majority of existing safety resources are available only in English and fail to address region-specific or language-specific scams.

Our Solution:

ParentShieldAI is a unified web-based platform designed to fill this gap. It provides immediate, clear verdicts on potentially malicious content through intelligent analysis. A parent can upload a screenshot of a suspicious message, and the AI will analyze its text—whether in English, Hindi, or Marathi—and also scan any visible links to provide a clear fraud prediction. This immediate, on-demand protection is supported by an educational library that builds long-term digital safety awareness. By incorporating an interactive quiz feature, ParentShieldAI makes learning an engaging process, helping parents build the critical thinking skills needed to identify threats on their own.

1.3 Objectives:

The ParentShieldAI project is designed to achieve the following key objectives:

1. Deploy a Unified Web Platform Develop a single, intuitive web application that consolidates multiple security tools into one accessible interface, eliminating the need for parents to navigate multiple platforms or applications.

2. Implement Multilingual Malicious Content Detection Implement an intelligent system leveraging Optical Character Recognition (OCR) and machine learning. This system will analyze text from user-uploaded message screenshots in English, Hindi, and Marathi. It will be trained to identify patterns indicative of scams and phishing attempts, and will also predict the malicious intent of any embedded URLs or links found in the content.

3. Predict Transaction Fraud Deploy a machine learning model trained to recognize anomalous patterns in financial transaction data—analyizing factors such as transaction amount, frequency, account balances, and transaction types—to predict and flag potentially fraudulent payment requests.

4. Provide an Interactive Dashboard with Downloadable Reports Develop a user-friendly dashboard that presents the results of any analysis (e.g., from message or transaction inputs) in a clear, easy-to-understand format. This dashboard will also provide users with the ability to download a summary report of the findings for their records.

5. Establish Interactive Learning Resources & a User Feedback Channel Establish a comprehensive Digital Safety Library containing articles, guides, and educational content.

To make learning engaging, this library will also feature an interactive game-based quiz to test and reinforce parental knowledge. Additionally, a feedback mechanism will be implemented for continuous platform improvement based on user experiences.

1.4 Scope:

1. Message Analysis Capabilities: Users can upload screenshots of suspicious messages received via SMS, WhatsApp, or other messaging platforms. The system employs OCR technology to extract text and uses Random Forest machine learning models to classify content as "Safe" or "Malicious."

2. Transaction Fraud Detection: The platform accepts user input of transaction details including amount, type, account balances, and other relevant parameters through web forms. A trained machine learning pipeline with 8 key features analyzes these parameters to predict and flag potentially fraudulent payment requests.

3. Fraud Link Detection: Analyzes shared URLs using rule-based checks like domain structure, length, and suspicious extensions to detect phishing or malicious links, providing a Safe/Malicious verdict with brief reasoning.

5. Fraud Fighter Game: An engaging game that helps users learn fraud detection skills through fun, scenario-based quizzes with instant feedback to enhance digital awareness.

6. Digital Safety Library: A curated repository of educational articles, guides, and resources covering various aspects of digital safety—from recognizing phishing attempts and social engineering tactics to secure online banking practices and password management.

7. Web-Based Platform: The system is accessible through standard web browsers on desktop and mobile devices, requiring no specialized software installation or technical expertise.

8. User Feedback Mechanism: An integrated feedback channel allows users to report new scam patterns, suggest improvements, and contribute to the platform's continuous enhancement.

9. Unified Interface: A single, intuitive dashboard that consolidates all threat detection tools and educational resources, eliminating the need for multiple applications or services.

The platform is designed with scalability in mind to accommodate future enhancements including multi-language support for regional Indian languages and Hinglish, community-driven threat reporting, and native mobile applications for Android and iOS.

Chapter 2

Literature Review

The spread of internet-accessible devices has significantly transformed childhood development while simultaneously exposing young users to increasingly sophisticated digital threats. Sharma, Gupta, and Chen (2024) highlighted a persistent gap in centralized, intelligent tools that help parents understand and counteract such modern digital risks [1]. Traditional safety manuals quickly become obsolete and fail to leverage modern analytical technologies, whereas online threats now involve global accessibility, emotional manipulation, and advanced social engineering—making simple parental oversight insufficient. The key challenge lies in bridging the knowledge gap between digital-native children and their parents through creative and inclusive solutions that simplify complex cybersecurity concepts into actionable awareness, aligning with global efforts toward sustainable digital literacy [1].

Existing digital safety solutions are generally classified into educational resources, automated content filters, and fraud detection systems used in the banking sector. Educational and government resources, while informative, are mostly static and fail to provide real-time support when parents face suspicious content. Machine learning–based spam filters, as discussed by Sharma et al. (2024), perform automatic classification but remain “black box” systems that offer little interpretability or user learning [1]. Likewise, Patel and Krishnan (2023) examined machine learning algorithms for UPI transaction fraud detection and found that even advanced models such as Random Forest and XGBoost, though accurate, operate behind the scenes without enabling proactive verification by end-users [2]. Consequently, parents must rely on fragmented systems that neither integrate multilingual capability nor offer on-demand validation of suspicious links or messages.

The technical foundations of these systems include rule-based, Natural Language Processing (NLP), and Machine Learning (ML)–based techniques for anomaly and fraud detection. Sharma et al. (2024) emphasized the importance of deep-learning-based NLP models in detecting phishing and smishing attempts in messaging applications, showing that contextual text understanding significantly improves detection accuracy beyond keyword-based methods [1]. Patel and Krishnan (2023) demonstrated how machine learning algorithms, particularly ensemble methods, effectively identify fraudulent transaction patterns in Indian payment systems like UPI [2]. In parallel, Smith (2007) provided an in-depth overview of the Tesseract OCR Engine, which serves as a foundation for extracting textual information from images or screenshots, a crucial step in message-level fraud analysis [3].

Despite these technological advancements, major research gaps persist. Current systems make automated decisions without explaining their reasoning, reducing user trust and learning potential. Rule-based systems, while fast, lack adaptability, whereas machine learning models, though accurate, struggle to communicate results in user-friendly terms. This gap underlines the need for an integrated, transparent, and educational solution. ParentShield.AI addresses this by combining OCR (Smith, 2007), NLP (Sharma et al., 2024), and ML-based fraud detection (Patel & Krishnan, 2023) within a unified multilingual interface. It empowers parents with real-time verification tools, instant fraud predictions, and interactive learning modules—transforming passive awareness into active protection while promoting digital inclusion and responsible technology use [1][2][3].

Chapter 3

Proposed System

The proposed system, ParentShield.AI, is a web-based application designed to provide comprehensive digital threat protection through a centralized and user-friendly platform. It addresses the inefficiencies and limitations of existing solutions by integrating advanced machine learning capabilities with intuitive user interfaces, making sophisticated cybersecurity tools accessible to non-technical users.

ParentShield.AI offers a unified ecosystem that combines real-time threat analysis with educational resources. The system processes user-uploaded message screenshots through OCR and machine learning to detect phishing and scams, analyzes transaction details through an 8-feature ML pipeline to predict fraud, and provides a comprehensive library of safety articles for proactive learning. By consolidating these features into a single platform, ParentShield.AI eliminates the need for multiple disparate tools and provides users with immediate, actionable insights at the critical moment of decision-making.

The platform operates on a three-tier architecture consisting of a React.js frontend for user interaction, a Flask API backend for request orchestration, and specialized AI modules for threat analysis. This architecture ensures fast response times, scalability, and maintainability while providing clear, explainable verdicts that empower users to make informed decisions about their digital safety.

3.1 Features and Functionality:

1. Message Fraud Detection: Users upload screenshots of suspicious messages received via SMS, WhatsApp, or other platforms. The system extracts text using OCR technology and analyzes it with a Random Forest Classifier trained to detect phishing, scams, and fake offers. Results include a clear verdict (Safe/Malicious), confidence score, and explanation highlighting key threat indicators.

2. Transaction Fraud Prediction: Users input transaction details through a web form including amount, type, and account balance information. The system analyzes these 8 key

features using a machine learning pipeline to identify fraudulent patterns. Output provides a verdict (Legitimate/Fraud Flagged), risk level (Low/Medium/High), and fraud probability score with explanations of suspicious patterns.

3. Digital Safety Library: A curated collection of educational articles organized by categories including phishing recognition, UPI safety, social engineering, password security, safe browsing, and app permissions. Content is presented in simple, jargon-free language with practical guidance and real-world examples accessible through browsing or search functionality.

4. Real-Time Analysis: Fast processing ensures message analysis completes in 3-4 seconds and transaction analysis in under 1 second, providing immediate feedback when users need it most for critical decision-making.

5. Explainable Verdicts: Unlike black-box security systems, ParentShield.AI provides clear explanations for all classifications, highlighting specific indicators like urgency language, suspicious links, unusual amounts, or atypical patterns that contributed to the verdict.

6. User-Friendly Interface: Intuitive React-based interface with clear navigation, step-by-step guidance, and responsive design that works across desktop and mobile devices, requiring minimal technical knowledge to operate.

7. Feedback Mechanism: Integrated system for users to report classification accuracy, new scam patterns, and suggestions for improvement, enabling continuous platform enhancement and community-driven threat intelligence.

8. Fraud Link Detection: It scans and analyzes URLs shared by users to identify suspicious patterns such as phishing domains, misspelled URLs, or unusual extensions. It checks the structure, length, and domain reputation of each link to determine its safety and alerts the user if the link appears fraudulent.

9. Interactive Quiz: It is an educational module designed to make online safety learning fun and engaging. Through interactive quizzes and gamified scenarios, users can test and improve their fraud detection skills, receive instant feedback, and gain awareness of real-world digital threats.

10. Report Generator: It compiles a comprehensive analysis report based on the user's activity and AI analysis results. It provides detailed visualizations of daily usage patterns, highlighting potential risks, behavioral insights, and safety recommendations. The report helps users track their online behavior trends, understand detected threats, and download the summarized findings for reference or review.

Chapter 4

Requirements Analysis

The requirements analysis defines the functional and non-functional specifications necessary for ParentShield.AI to effectively serve its target users and achieve its objectives.

4.1 Functional Requirements

1. User Authentication and Profile Management:
Users can conveniently access the platform directly through web browsers without the need for mandatory account creation, allowing them to perform basic threat analyses quickly and effortlessly. For those who choose to register, secure authentication mechanisms are implemented to ensure the protection of user data and maintain complete privacy throughout their interactions with the platform.

2. Message Analysis Module:
The system allows users to upload images in common formats such as JPG, PNG, and JPEG, with a maximum size limit of 5MB. Once uploaded, the application extracts text from the image using Optical Character Recognition (OCR) with sufficient accuracy for further analysis. The extracted text is then preprocessed to remove noise, special characters, and formatting inconsistencies, ensuring clean and reliable input. Finally, the processed text is displayed to the user for verification before proceeding to the next stage of analysis.

3. Transaction Fraud Detection Module:
The system accepts transaction screenshots in formats such as JPG, PNG, or JPEG (up to 5MB) and extracts relevant text information using Optical Character Recognition (OCR). The extracted data is then processed through trained machine learning algorithms that analyze keywords and pattern-based datasets to predict potential fraud. Finally, the system generates a clear verdict with a fraud probability score and visually highlights suspicious elements within the message or screenshot, ensuring better user understanding and awareness.

4. Digital Safety Library:

The platform ensures that all content is presented in a readable and mobile-friendly format, enhanced with engaging visuals such as videos and infographics to make learning more interactive. Additionally, it tracks user engagement to identify which topics are most popular and to detect content gaps, enabling continuous improvement and personalization of the learning experience.

5. User Feedback System:

The platform also includes a feedback mechanism where users can share their experience and help improve the system. Parents can provide feedback on whether the AI's classification verdicts were accurate or not, allowing continuous refinement of the model's performance. Additionally, they can share suggestions for new features or improvements, ensuring that the platform evolves to better meet user needs and deliver a more reliable, user-friendly experience over time.

4.2 Non-Functional Requirements

Performance:

The platform is designed for high efficiency and responsiveness, ensuring a smooth user experience. Each message uploaded for analysis, including OCR processing, is completed within 5 seconds, providing users with quick and reliable results. Additionally, the overall platform response time for page loads remains under 2 seconds, allowing parents to navigate and access features seamlessly without delays.

Reliability:

The ParentShield.AI platform is designed for maximum reliability and usability, ensuring a seamless experience for all users. It maintains a system uptime of 99% or higher, providing continuous and dependable access. The platform incorporates graceful error handling with clear and helpful user messages to minimize confusion during any technical issues. From a usability standpoint, it features an intuitive interface that requires minimal user training, uses clear and jargon-free language throughout, and follows a responsive design that works

smoothly across both desktop and mobile devices, making it accessible to users of all technical backgrounds.

Maintainability:

The system will include comprehensive developer documentation to ensure smooth understanding, integration, and maintenance of all modules. It will employ version control and testing frameworks to maintain code reliability, enable collaboration, and ensure continuous quality assurance throughout development. Additionally, the platform will incorporate mechanisms for model updates and retraining that allow AI models to be improved regularly without causing any disruption to ongoing services or user experience.

Chapter 5

Project Design

The design of ParentShieldAI focuses on creating a unified, user-friendly, and intelligent system that integrates multiple components to ensure seamless fraud detection and digital safety awareness. This chapter presents the structural and functional blueprint of the platform, highlighting how different modules—such as message and transaction analysis, user feedback, safety library, and interface design—interact cohesively within the system. The project design emphasizes scalability, usability, and security, ensuring that users can easily access fraud detection tools, analyze suspicious content, and gain valuable educational insights in real time.

5.1 Use Case Diagram

The use case diagram for ParentShield.AI illustrates the interactions between the primary actors and the system, capturing the core functionalities. The main actor is the End User, such as a parent or general user, who interacts directly with the platform's threat detection tools and educational resources.

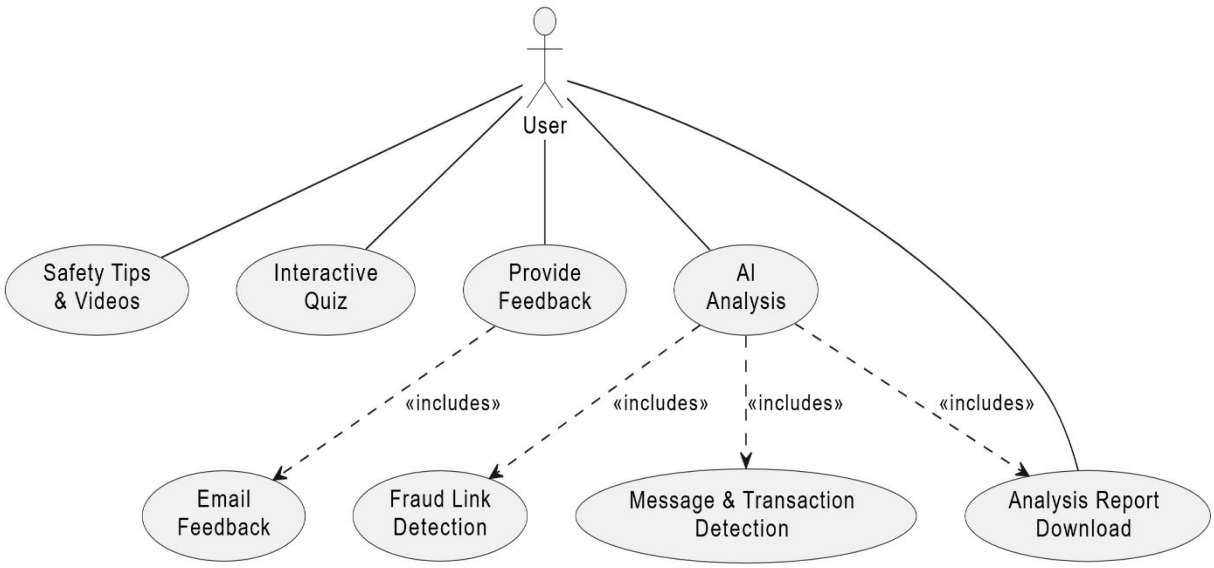

Fig 5.1 ParentShield.AI use case diagram

The user can perform several key actions. These include the ability to upload a message screenshot, which allows the user to submit an image of a suspicious message for analysis. The system then performs OCR extraction and classification, after which the user can view the message analysis result and its corresponding explanation.

Similarly, another primary use case enables the user to enter transaction details into a structured form for fraud prediction. After the system processes this input, the user can then view the transaction analysis result, which includes a fraud assessment and risk level.

Beyond threat detection, the user can also browse the digital safety library to explore educational articles by category, or search the library content for specific topics or threat types. Finally, the user can submit feedback to report on classification accuracy or suggest improvements. This feedback is then handled by the system, which will store the feedback for review and log the analysis events for performance monitoring.

5.2 DFD (Data Flow Diagram)

The Data Flow Diagram (DFD) illustrates how data moves through the ParentShield.AI platform, beginning from the initial user input and tracing its path to the final output. This diagram visually maps the transformations, processes, and storage points involved in handling the data.

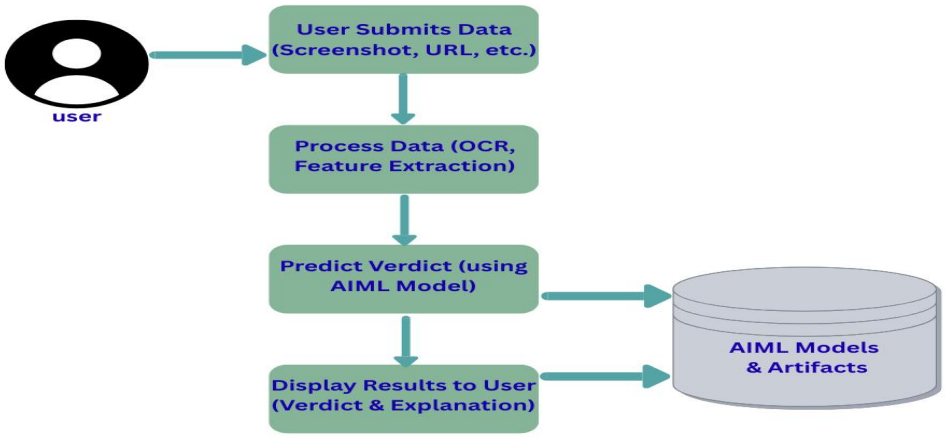

Fig 5.2: Data Flow Diagram

User Interaction: The User initiates an HTTP Request by interacting with the React.js Interface (Tier 1: Presentation Layer). This involves actions like uploading a message screenshot or submitting transaction data. The system then Displays the Results (the final verdict and explanation) back to the user in the interface.

Processes: The React.js Interface sends an API Call (e.g., /api/predict_message) to the Python Flask API (Tier 2: Application Layer). This API acts as the central controller.

For message analysis, the Flask API Sends the image data to the Pytesseract OCR Engine (Tier 3: Data & AI Layer) and Returns the extracted text. The Flask API then Sends this processed data (the text, or transaction features) to the Random Forest Models. The models analyze the data and Return a prediction (e.g., "Safe" or "Fraudulent") back to the Flask API, which formats the final API Response (JSON) and sends it to the React frontend.

Databases / Storage: The Random Forest Models in Tier 3 rely on Model Artifacts / Storage. This data store holds the pre-trained model files (like .pkl files), vectorizers, and scalers necessary to make an accurate prediction.

The main ParentShield.AI system is broken down into the following core sub-processes:

1. Message Analysis: This process is initiated when the user provides a message screenshot as an image file. This input first undergoes Image Preprocessing, where it is resized, enhanced, and converted for optimal analysis. Next, OCR Text Extraction is performed to pull the raw text from the image. This text is then passed to a Text Preprocessing stage for cleaning and tokenization. Following this, Feature Extraction (using TF-IDF vectorization) converts the clean text into a numerical format that the Random Forest classification model can understand. The model then analyzes these features to produce the final output, which consists of a clear Verdict (Safe/Malicious), a confidence score, and a simple explanation for the user. This process utilizes Data Stores for its model artifacts and to log the analysis results.

2. Transaction Fraud Detection: This process begins when the user inputs specific transaction details, which consist of eight key features. The data first goes through Data Validation to ensure it is in the correct format. It then undergoes Feature Engineering and

Normalization (using StandardScaler) to prepare it for the predictive model. The prepared features are then fed into a Fraud Prediction model (such as Random Forest or Logistic Regression) to assess the likelihood of fraud. This leads to a Risk Assessment, generating the final output: a Verdict (Legitimate/Fraud), an assigned risk level, and a probability score. This process also relies on Data Stores for its model artifacts and analysis logs.

3. Digital Safety Library: This process is activated by a User request, which can be the selection of a content category or a specific search query. The system's Query Processing function first interprets this request. Then, the Content Retrieval function fetches the relevant articles, guides, or quiz information from the platform's knowledge base. Finally, Content Formatting prepares the retrieved information to be displayed clearly to the user.

4. Feedback Management: This process handles input provided by the user in the form of feedback, which might include ratings on classification accuracy or written suggestions for improvement. This feedback then undergoes Feedback Validation to ensure it is legitimate and properly formatted before being stored and used by the development team for continuous platform improvement.

5. Fraud Link Detection: This process analyzes URLs shared by users to identify potential phishing or malicious links. It inspects the domain structure, URL length, presence of IP addresses, and suspicious extensions through rule-based analysis. Based on these checks, it provides a Verdict (Safe/Malicious) with reasoning, helping users stay safe from fraudulent websites.

6. Interactive Quiz: The Interactive Quiz enhances user awareness by presenting engaging, scenario-based questions related to online safety and digital behavior. It evaluates responses to offer instant feedback and personalized tips, promoting digital literacy and reinforcing safe online habits.

5.3 System Architecture

The System Architecture of ParentShield.AI illustrates a robust, multi-layered framework designed to ensure modularity, scalability, and maintainability. It adopts a three-tier architecture comprising the Presentation Layer (Frontend), the Application Layer (Backend), and the Data and AI Layer. Each tier serves a distinct yet interconnected role in processing user requests and delivering accurate, real-time threat analysis.

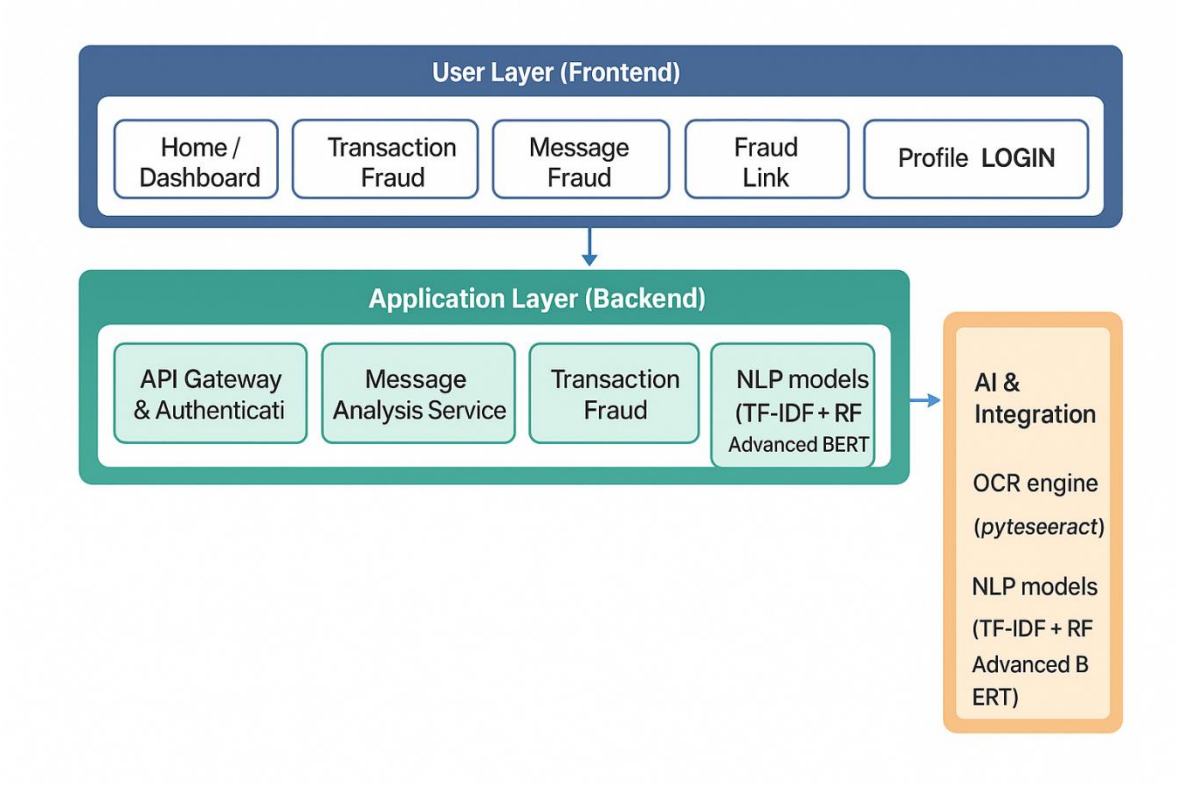

Fig 5.3: System Architecture

1. Presentation Layer (Frontend)

The Presentation Layer provides an intuitive and responsive web-based interface developed using React.js. This layer is responsible for rendering all user-facing components, enabling seamless access to core features such as the Message Upload interface, the Transaction Form, and the Digital Safety Library. It is designed to handle all user interactions, manage client-side navigation, and display the final analysis results from the backend in a clear, visually formatted, and easy-to-understand manner.

2. Application Layer (Backend)

The Application Layer, powered by Python Flask, serves as the core logic controller that manages all API calls and processes user requests. It defines all API endpoints (such as POST /api/predict_message and POST /api/predict_transaction) and orchestrates the backend services. When a request is received, this layer validates the input, routes it to the appropriate analysis service, and formats the final prediction into a JSON response. It is also responsible for handling all error conditions and implementing security measures like input validation.

3. Data and AI Layer

The Data and AI Layer is responsible for all intelligent processing and data management. This layer houses the core AI services, including the Pytesseract OCR Engine for text extraction from images, the Pillow library for image preprocessing, and the trained Random Forest machine learning models used for both message classification and transaction prediction. This layer also manages the Model Artifacts (such as trained pickle files, vectorizers, and scalers) and handles the temporary storage of uploaded images, which are deleted after analysis to ensure user privacy.

The overall architecture flow begins when a user interacts with the React frontend. This triggers an API request to the Flask backend, which validates the request and routes it to the appropriate analysis service. The service then calls upon the AI models in the Data and AI Layer, receives a prediction, and formats the result.

This final result is returned to the frontend and displayed clearly to the user, completing the end-to-end process.

5.4 Implementation

The implementation section describes the technical approach to building ParentShieldAI, including development methodology, key implementation decisions, and deployment strategy.

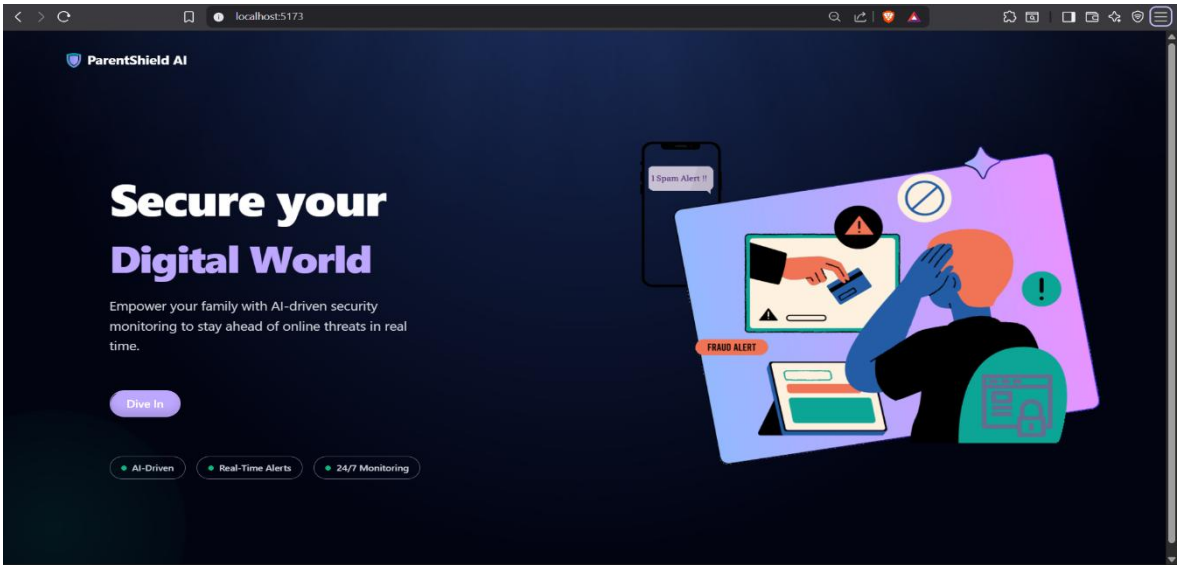

Fig 5.4.1: Application Welcome Page

The Fig 5.4.1 illustrates the application's Welcome Page, which serves as the primary landing screen and the initial point of contact for all unauthenticated users. The design is intentionally focused, centered on immediately communicating the project's core value proposition through the clear, prominent headline: "Secure your Digital World."

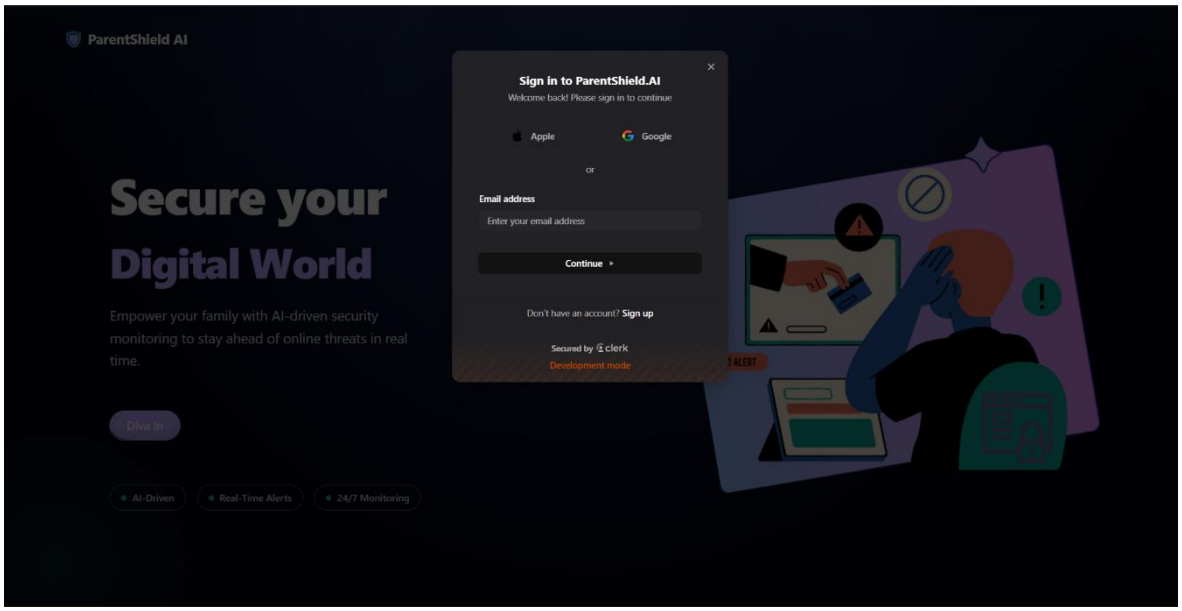

Fig 5.4.2: User Sign-In/Sign-Up Modal

The Fig 5.4.2 modal confirms the use of secure, external authentication services (Clerk), streamlining the user experience while ensuring robust security for managing user access.

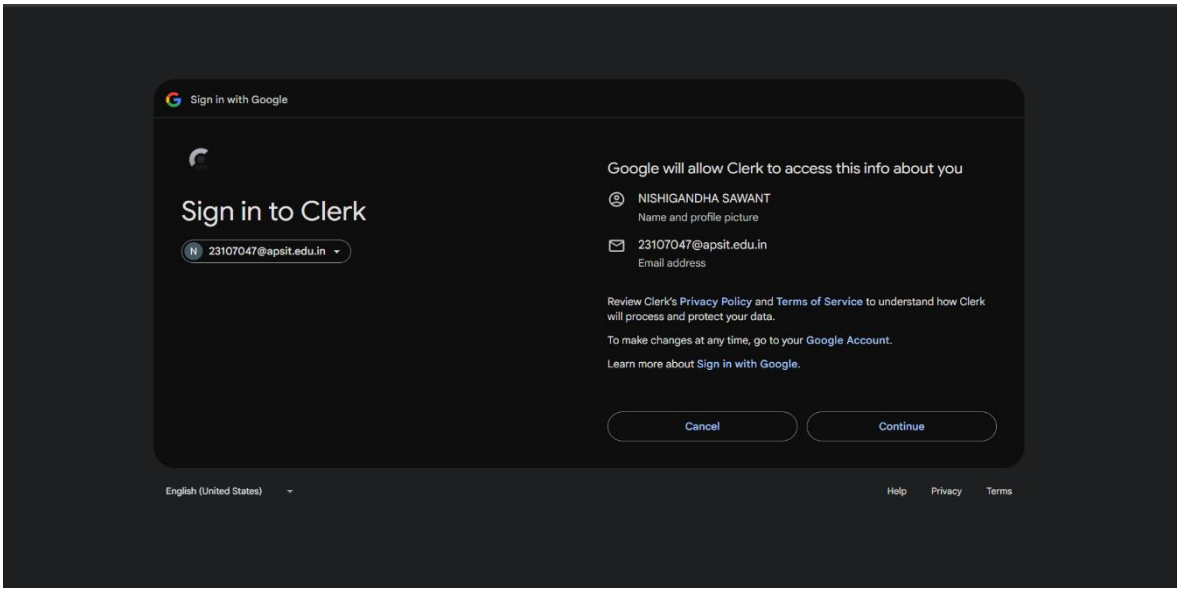

Fig 5.4.3: Third-Party Authentication Authorization (Google)

The Fig 5.4.3 shows the OAuth authorization screen, requesting user permission for Clerk to access their Google profile data.

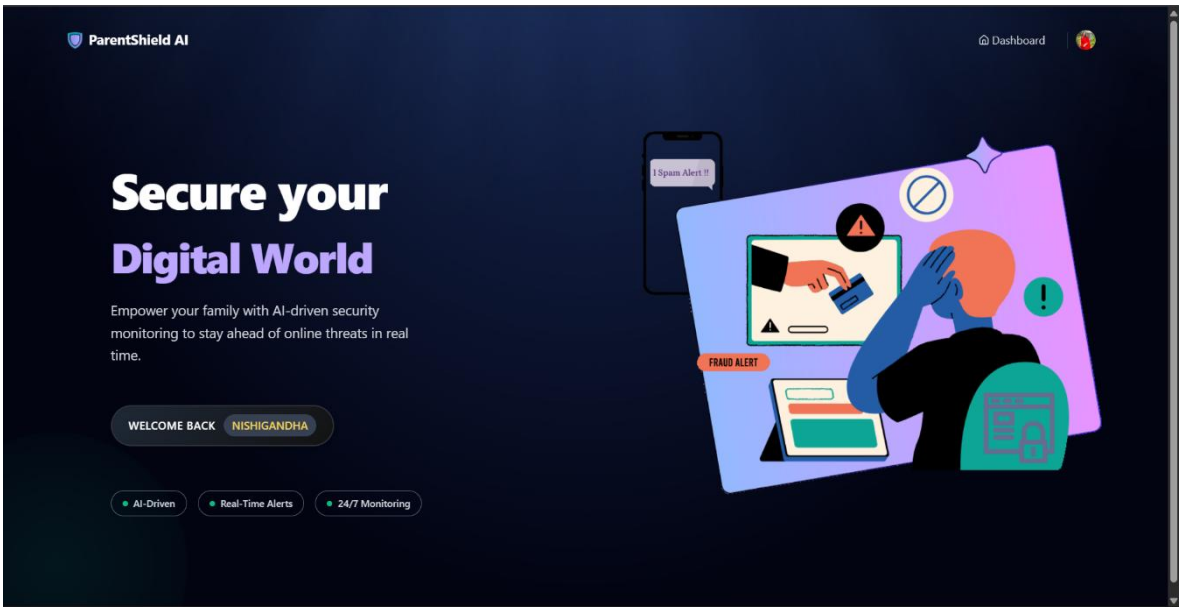

Fig 5.4.4: Authenticated Landing Page

The Fig 5.4.4 shows that after successful sign-in, the welcome screen updates to greet the user by name, confirming their authenticated status before navigation.

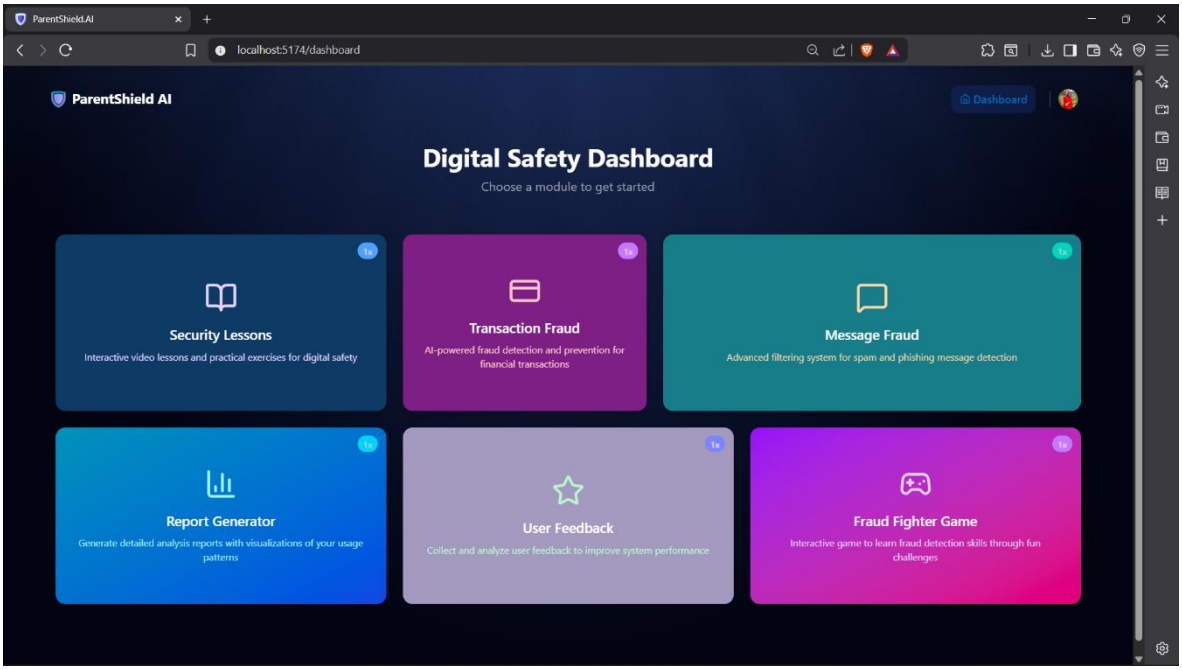

Fig 5.4.5: Central Analytics Dashboard

The Fig 5.4.5 is the primary user dashboard, providing a high-level analytics overview of key metrics, designed for monitoring the application's overall performance.

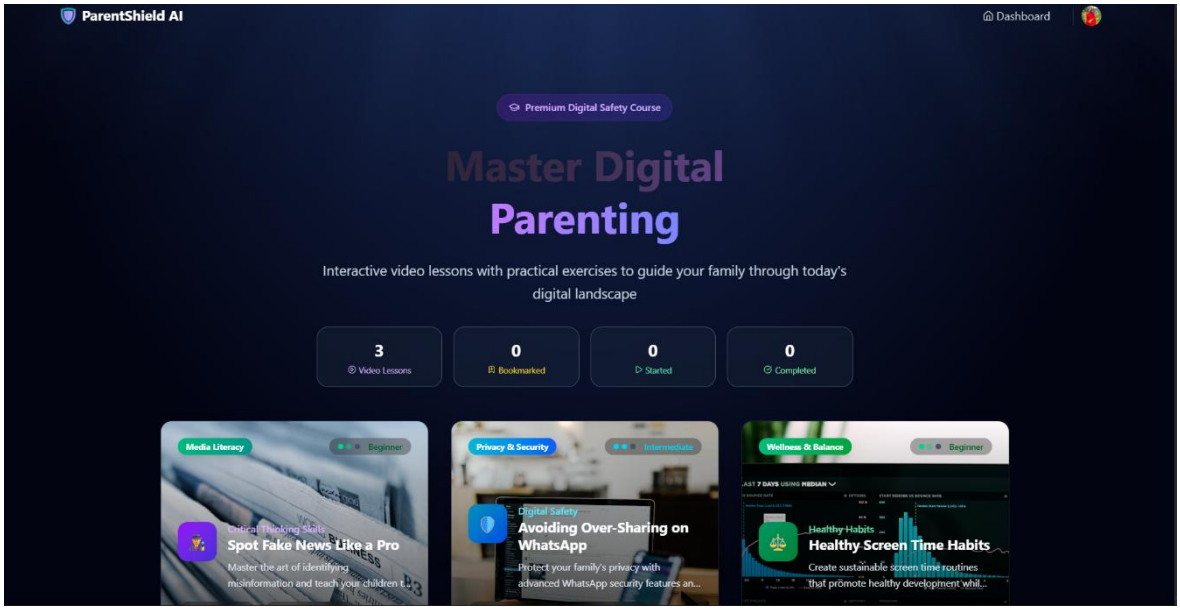

Fig 5.4.6: Digital Awareness Tips Module

The Fig 5.4.6 shows the 'Digital Awareness Tips' module provides curated lessons on essential safety topics, allowing users to read, track progress, and bookmark relevant expert guidance.

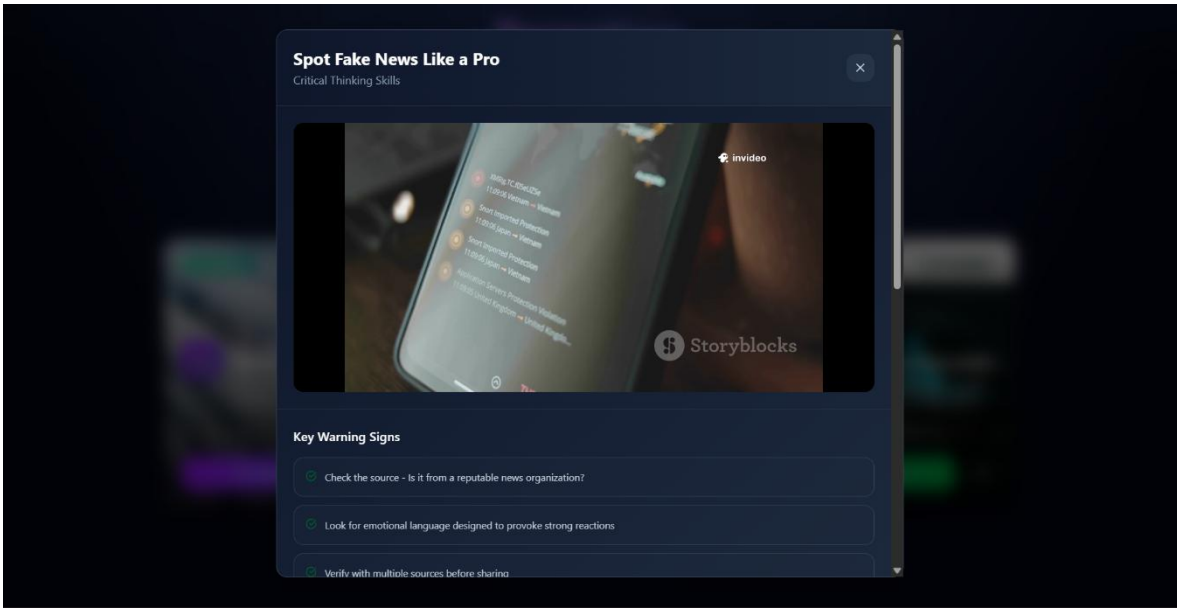

Fig 5.4.7: Detailed Security Lesson Modal

Fig 5.4.7 provides a modal view detailing a security lesson, such as 'Spot Fake News Like a Pro.' The view features an embedded instructional video and is followed by actionable, bulleted Key Warning Signs for critical analysis.

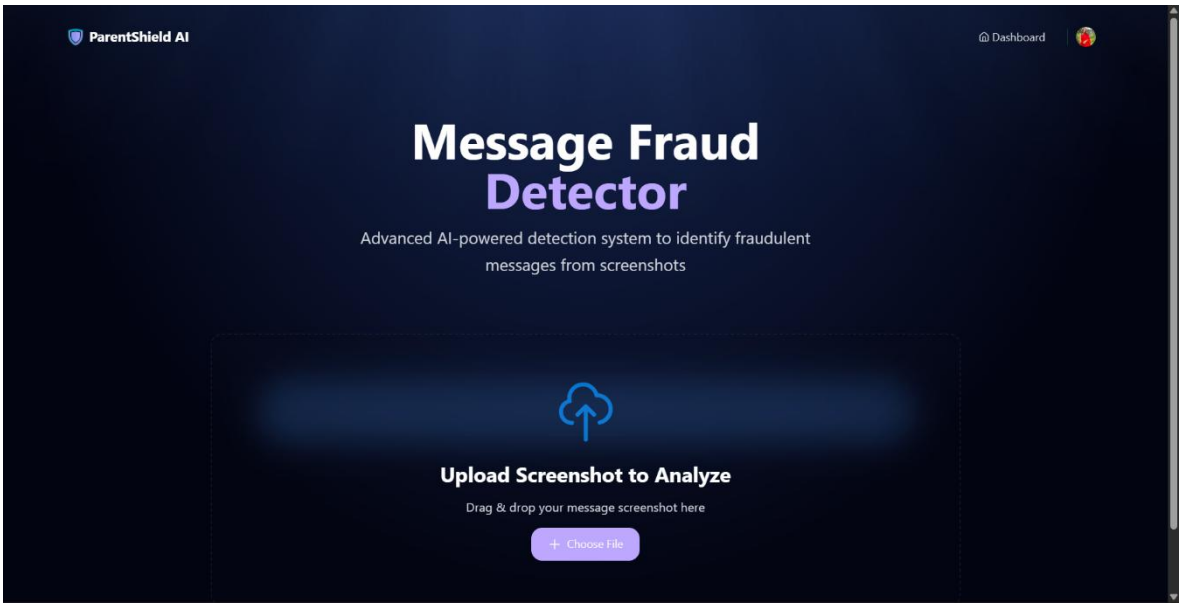

Fig 5.4.8: Message Fraud Detector Tool

The Fig 5.4.8 shows the Message Fraud Detector interface uses AI to analyze a screenshot of a text message, enabling users to proactively check messages for phishing or spam risks before engaging.

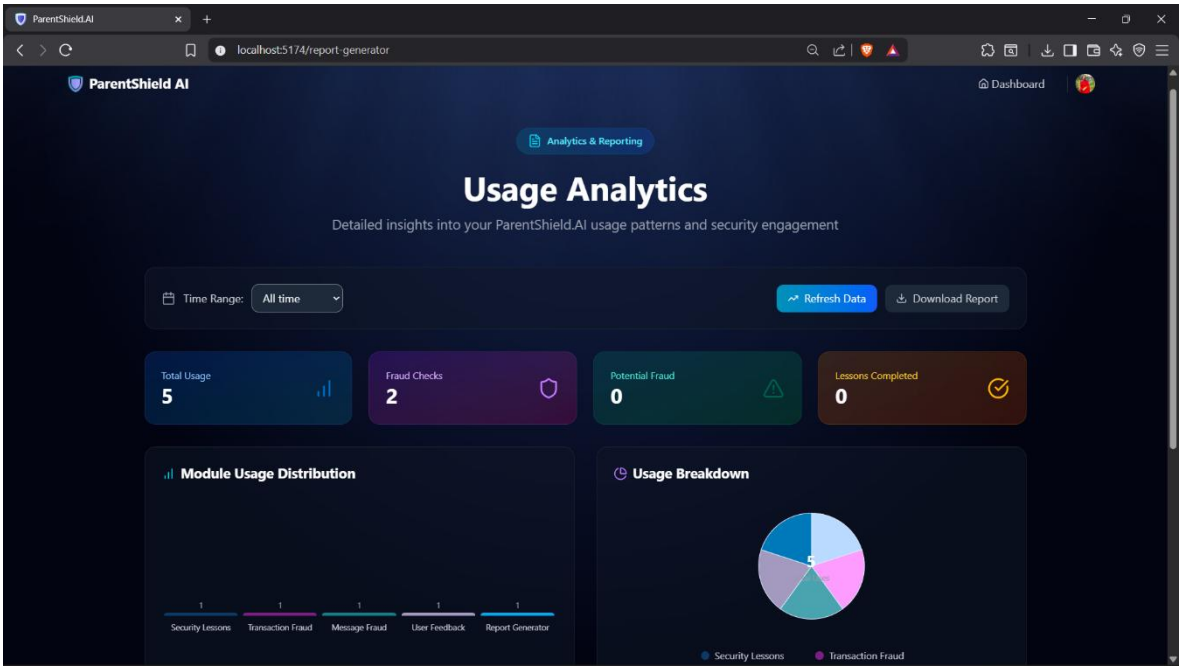

Fig 5.4.9: Usage Analytics Dashboard

This Fig 5.4.10 displays the Usage Analytics dashboard, providing insights into user engagement and security metrics like Total Usage, Fraud Checks, and Module Usage Distribution.

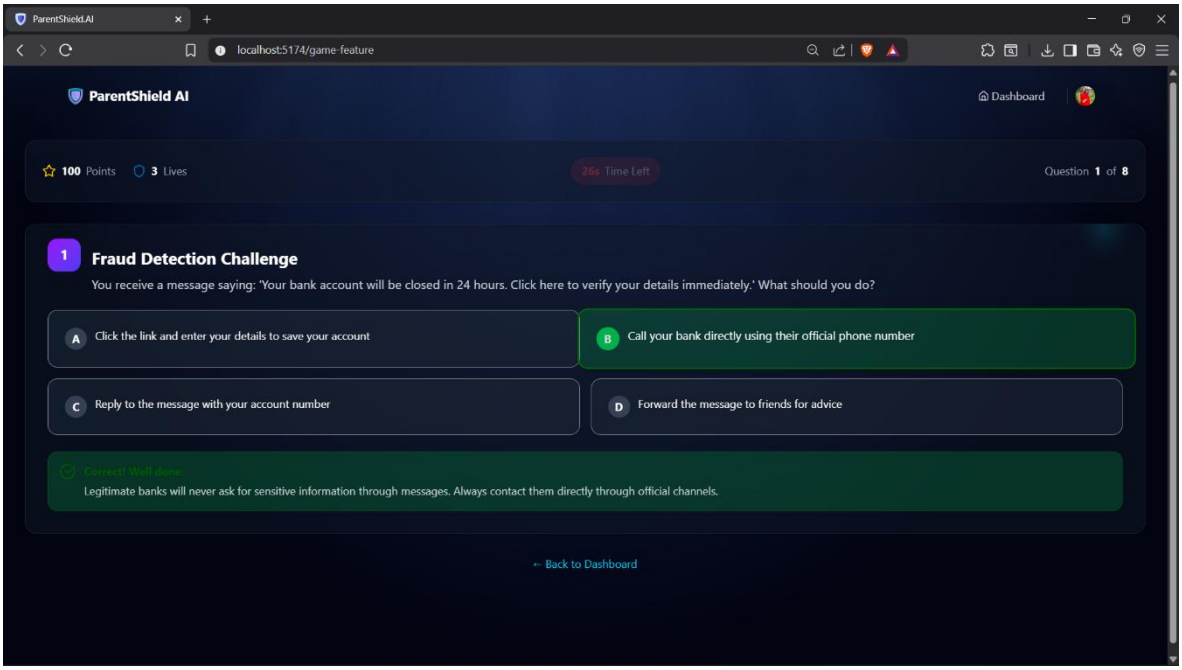

Fig 5.4.10: Interactive Learning Quiz (Game Feature)

This Fig 5.4.11 shows the "Fraud Detection Challenge" quiz, an interactive game feature designed to test and improve a user's ability to identify phishing and fraud by presenting them with real-world scenarios.

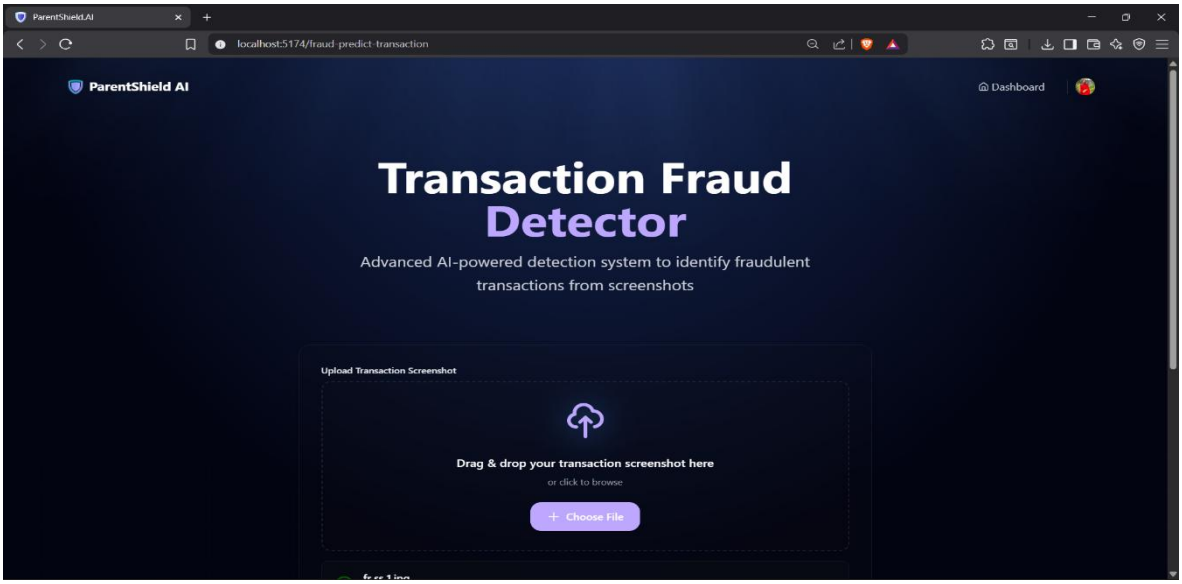

Fig 5.4.11: Transaction Fraud Detector Interface

This Fig 5.4.9 shows the Transaction Fraud Detector page, located at the /fraud-predict-transaction endpoint. It provides a clean interface for users to upload a transaction screenshot using either drag-and-drop or the "Choose File" button, initiating the AI-powered fraud analysis.The module validates 8 transaction features, organizes data into Pandas DataFrame, normalizes using StandardScaler, and applies Random Forest/Logistic Regression. Output includes fraud probability (0-1) converted to risk levels (Low: <0.3, Medium: 0.3-0.7, High: >0.7) with explanations of suspicious patterns

Chapter 6

Technical Specification

The development of ParentShieldAI was guided by a carefully selected technology stack aimed at achieving accuracy, speed, and a smooth user experience. Since the platform deals with real-time message and transaction analysis, every component—from frontend tools to backend frameworks—was chosen to ensure reliability, scalability, and seamless integration. The architecture combines modern web technologies with powerful machine learning libraries, ensuring the system remains both user-friendly and technically robust.

6.1 Frontend Technologies

The frontend of ParentShield.AI was developed using React.js**,** HTML5**,** CSS3**,** and modern JavaScript (ES6+)**.** React's component-based structure allows for reusable and dynamic UI elements, improving both performance and maintainability. Combined with responsive layouts built using CSS Grid and Flexbox, the interface adapts smoothly across devices. This ensures that users—from beginners to experienced internet users—can easily navigate and interact with the platform.

6.2 Backend Technologies

Python 3.x: High-level, interpreted programming language chosen for its extensive libraries for machine learning, data processing, and web development. Python's readable syntax and rich ecosystem make it ideal for rapid development and integration of AI components.

Flask: Lightweight WSGI web application framework for Python. Flask provides minimal overhead and maximum flexibility, making it perfect for building RESTful APIs. Its simplicity allows easy integration with machine learning libraries while maintaining clean, maintainable code structure.

6.3 Machine Learning and Data Processing

For machine learning, the platform uses Scikit-learn, supported by Pandas and NumPy for data handling. These tools power the message and transaction fraud detection models by

managing data preprocessing, feature extraction, and model evaluation. The Random Forest algorithm was selected for its high accuracy and interpretability. This combination ensures quick, reliable predictions while maintaining transparency in how results are generated.

6.4 Image Processing and OCR

The system uses Pillow (PIL) for image preprocessing tasks such as resizing, conversion, and enhancement. For text extraction, Pytesseract, a Python wrapper for Google's Tesseract OCR engine, was employed. Together, these tools accurately extract textual data from message screenshots, allowing the model to analyze and detect potential fraud in real-world communication formats.

6.5 Development and Deployment Tools

1. Git: Version control system for tracking code changes, managing branches, and facilitating collaborative development.

2. npm/yarn: Package managers for managing JavaScript dependencies and build tools for the React frontend.

3. pip: Package installer for Python libraries and dependencies.

4. Gunicorn: Python WSGI HTTP server for running Flask applications in production environments.

5. Nginx: High-performance web server and reverse proxy for serving static files and forwarding API requests to the Flask backend.

Chapter 7

Project Scheduling

The project's 14-week phased plan ensured smooth progress from research to deployment by setting clear goals for efficient development, integration, and testing**.**

Sr No.	Group Members	Duration	Task Performed
1	Nishigandha, Nidhi, Abhishek	1st–2nd Week of July	Defined the project scope, objectives, and key user requirements for *ParentShield.AI*. Conducted an in-depth literature review on fraud detection methods, identified suitable datasets,
2	Nidhi, Abhishek	3rd–4th Week of July	Collected and preprocessed datasets (Message Fraud & PaySim). Performed data cleaning, EDA, and train-test split.
3	Nishigandha, Abhishek	1st Week of August	Developed and tuned TF-IDF + Random Forest for message fraud; compared models for transaction fraud.
4	Nidhi, Nishigandha	3rd Week of August	Built Flask-based backend APIs with OCR and fraud detection modules. Added *Digital Safety Library* and feedback features.
5	Abhishek, Nidhi	1st Week of September	Designed React frontend with dashboard, upload interface, and result pages. Integrated APIs and optimized UI.
6	Nishigandha, Abhishek	3rd Week of September	Integrated all modules, conducted system testing, debugged, and optimized performance.

Fig 7.1: Gantt Chart of ParentShield.AI

Chapter 8

Results

This chapter highlights the key outcomes from the design and implementation of ParentShield.AI, focusing on system performance, analytical results, and user testing. The platform integrates two core modules — Message Analysis and Transaction Analysis — which were assessed for accuracy, efficiency, and overall usability. The Message Analysis module demonstrated an average response time of 3–4 seconds from image upload to final verdict, ensuring real-time analysis for users. It achieved strong detection accuracy across various categories, including phishing messages, fake job offers, and lottery scams, while also correctly identifying legitimate messages with high precision. Although some false positives were observed in urgent legitimate messages and occasional false negatives in newly emerging scams, the overall performance proved robust and reliable for real-world application. Similarly, the Transaction Analysis module processed user inputs in less than a second, maintaining high effectiveness in detecting unauthorized transfers, multiple small suspicious transactions, and potential account takeover patterns, confirming the model's stability and responsiveness.

To assess accessibility and user satisfaction, *ParentShield.AI* underwent user testing with participants aged 40–65. The results showed high task completion rates, with most users easily navigating core features such as message uploads, transaction analysis, and understanding verdict explanations. Feedback from participants was overwhelmingly positive, praising the intuitive design, fast response, and clear explanations. Suggestions for improvement included the addition of Hindi language support, more detailed insights into detected threats, and a dedicated mobile version. Overall, the system achieved its goal of combining high-performance fraud detection with user-friendly interaction, reinforcing ParentShield.AI as a trustworthy and efficient platform that empowers parents to stay safe and informed in the digital environment.

8.3 Comparison With Existing Solutions:

Existing digital safety solutions such as Truecaller, bank fraud systems, and general educational websites each address only a small part of the online safety problem. Truecaller provides limited message verification by flagging suspicious calls and texts, but it lacks comprehensive analysis or educational value for users. Banking systems, on the other hand, employ strong AI-driven fraud detection for financial transactions, yet these tools are restricted to their own customers and operate entirely in the background—offering no real-time or user-facing verification. Fraud awareness websites and educational portals contribute to awareness through articles and tutorials, but they remain passive, generic, and non-interactive, providing no on-demand protection.

In contrast, ParentShieldAI integrates all these aspects into a single, unified platform. It allows parents to perform on-demand analysis of suspicious messages, links, and screenshots while also providing transaction verification, clear AI-generated explanations, and interactive learning resources. Unlike existing solutions, which either work silently or educate superficially, ParentShieldAI actively empowers users through real-time fraud detection and engaging educational tools—bridging the gap between awareness and action in a way that is simple, transparent, and accessible for all users.

Key Differentiator: ParentShield.AI uniquely combines on-demand verification tools with educational resources in a single, user-friendly platform designed specifically for less tech-savvy users.

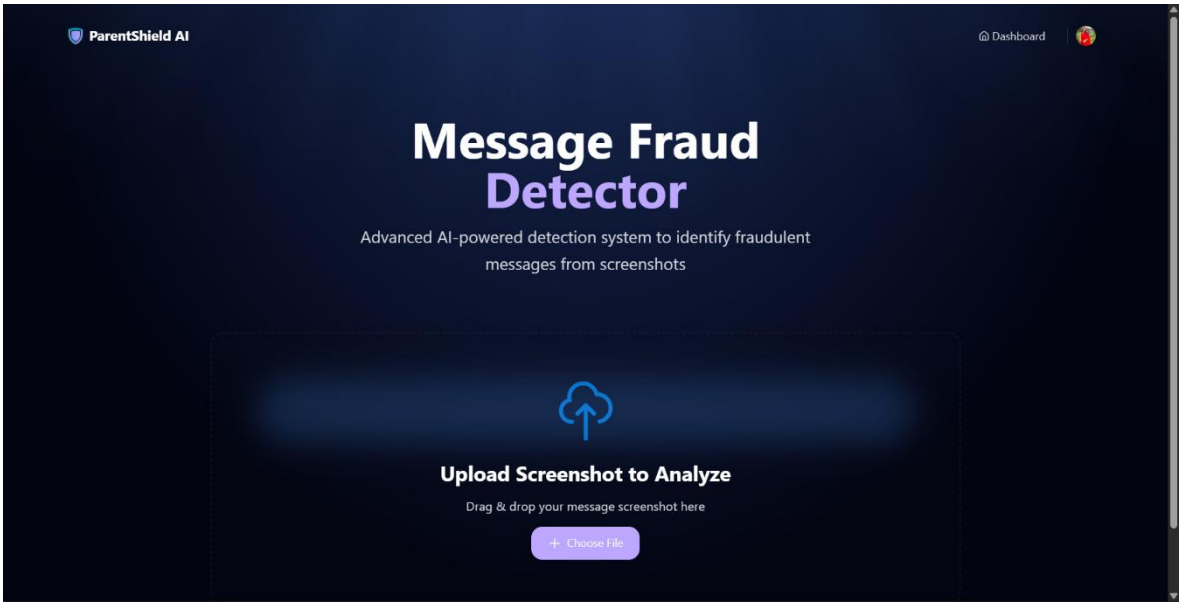

Fig 8.4.1: Message Fraud Detector Tool

The Fig 8.4.1 shows Message Fraud Detector interface uses AI to analyze a screenshot of a text message, enabling users to proactively check messages for phishing or spam risks before engaging.

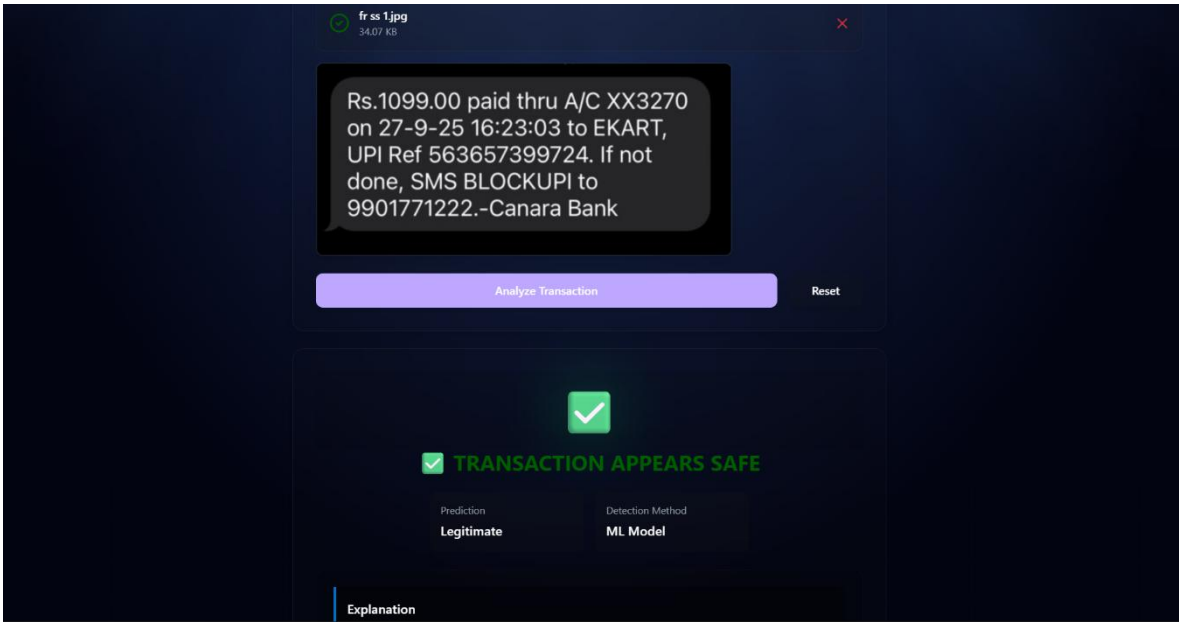

Fig 8.4.2: Message Fraud Detector Tool

This Fig 8.4.2 showcases the primary user interface for the **Message Fraud Detector**, a cornerstone, proactive security feature of the ParentShield.ai platform. This tool is engineered to empower parents and users who may be uncertain about the legitimacy of

text messages they receive, particularly those involving financial transactions or urgent requests.

The workflow, as depicted, is intentionally simple. A user begins by uploading a screenshot of the suspicious message. In the backend, the system immediately employs an Optical Character Recognition (OCR) engine to accurately extract the raw text from the image. This extracted text is then passed to our trained Machine Learning (ML) model.

The model analyzes the content for a variety of indicators associated with phishing, spam, and fraud, including suspicious linguistic patterns, a false sense of urgency, and the structure of any embedded URLs or links. The system is also designed with multilingual capabilities to parse and understand messages in English, Hindi, and Marathi.

As the figure demonstrates, the user has uploaded a screenshot of a legitimate bank transaction alert. After the user clicks "Analyze Transaction," the ML model correctly processes the content and returns a clear, unambiguous verdict: "TRANSACTION APPEARS SAFE." The interface further provides transparency by noting the Prediction as "Legitimate" and the Detection Method as the "ML Model."

By providing this immediate, on-demand analysis, the Message Fraud Detector enables users to proactively verify the safety of a message *before* they engage with it, click a dangerous link, or reply with sensitive information.

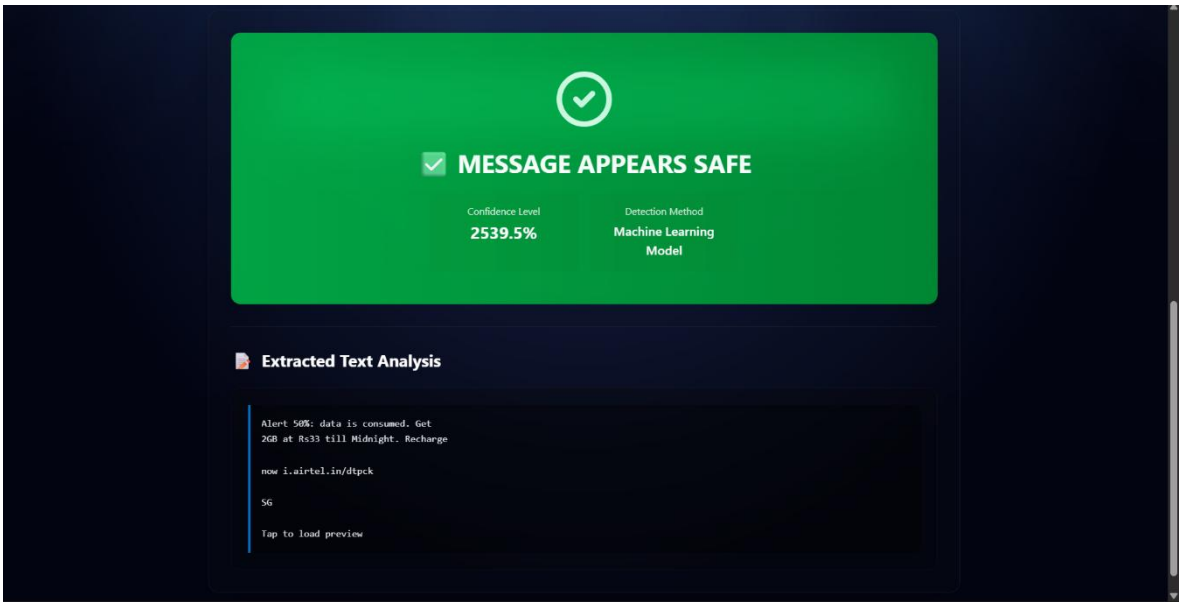

Fig 8.4.3: Message Fraud Detection Result (Safe)

Fig 8.4.3 displays a successful analysis result from the Message Fraud Detector. The interface clearly communicates a verdict of "MESSAGE APPEARS SAFE" within a distinct, green-colored banner, providing immediate visual confirmation for the user.

To build user trust, the system provides supporting details, including the Confidence Level associated with the prediction and the Detection Method used, which is confirmed as the "Machine Learning Model".

Furthermore, the **"Extracted Text Analysis"** section is shown below the verdict. This feature is critical as it displays the exact text that the **OCR engine extracted** from the user's screenshot. This transparency allows the user to verify the input data that the model used to arrive at its "Safe" conclusion.

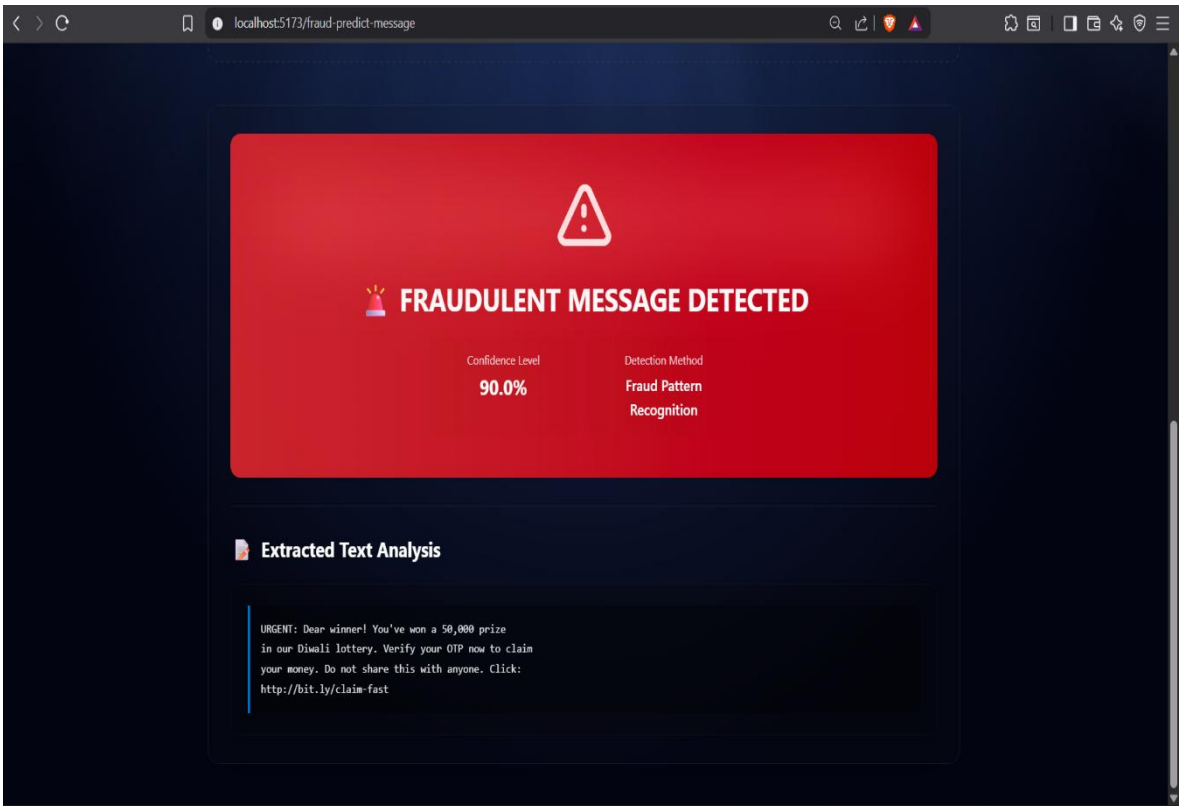

Figure 8.4.5: Fraudulent Message Detection Output

Fig 8.4.5 demonstrates the system's critical alert mechanism when a high-risk threat is successfully identified. The interface immediately captures the user's attention by displaying a prominent, bright red banner with the clear warning: "FRAUDULENT MESSAGE DETECTED."

The Detection Method is listed as "Fraud Pattern Recognition," signifying that the text matched known linguistic patterns and structural red flags associated with scams.

The "Extracted Text Analysis" section below validates this finding, showing the OCR-captured text which contains classic scam elements: an unsolicited prize ("won a 50,000 prize"), a false sense of urgency ("claim your OTP now"), and a suspicious shortened URL, all designed to phish for user information.

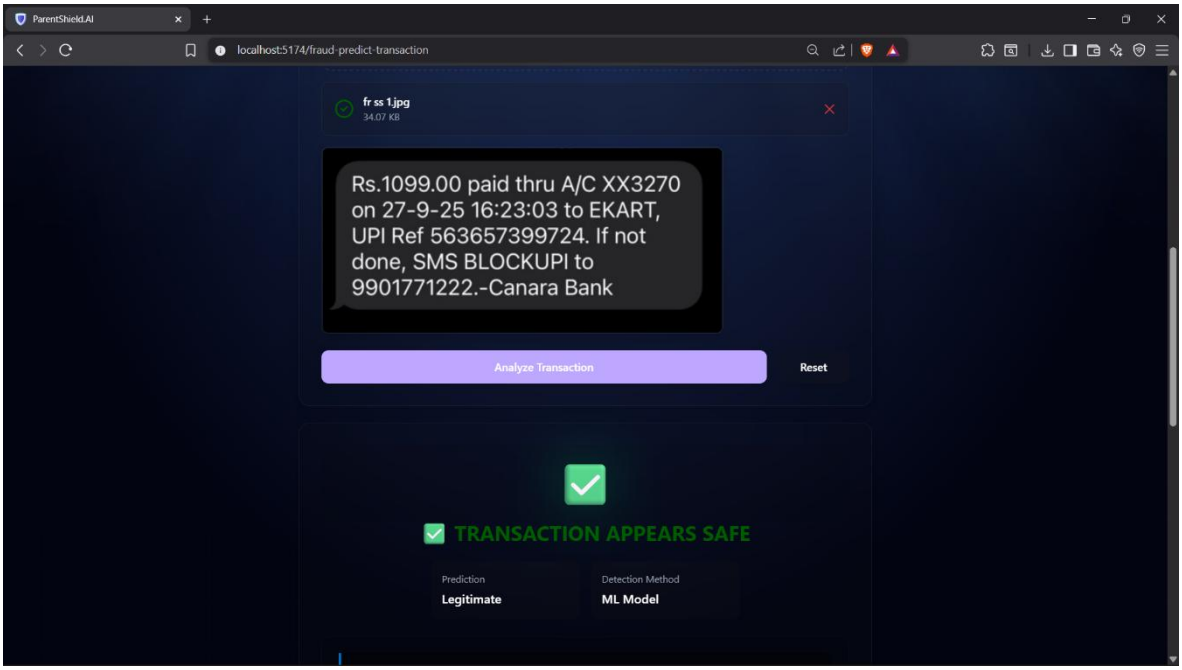

Fig 8.4.6: Legitimate Transaction Analysis Output

This Fig 8.4.6 demonstrates the system's output when analyzing a safe transaction screenshot. After the ML model returns a "TRANSACTION APPEARS SAFE" verdict, the interface provides a comprehensive breakdown. This includes a detailed Explanation of the verdict, a "Transaction Details" card showing key information (Amount, Type, Transaction ID) extracted by the AI, the raw "Extracted Text Analysis" captured by the OCR engine.

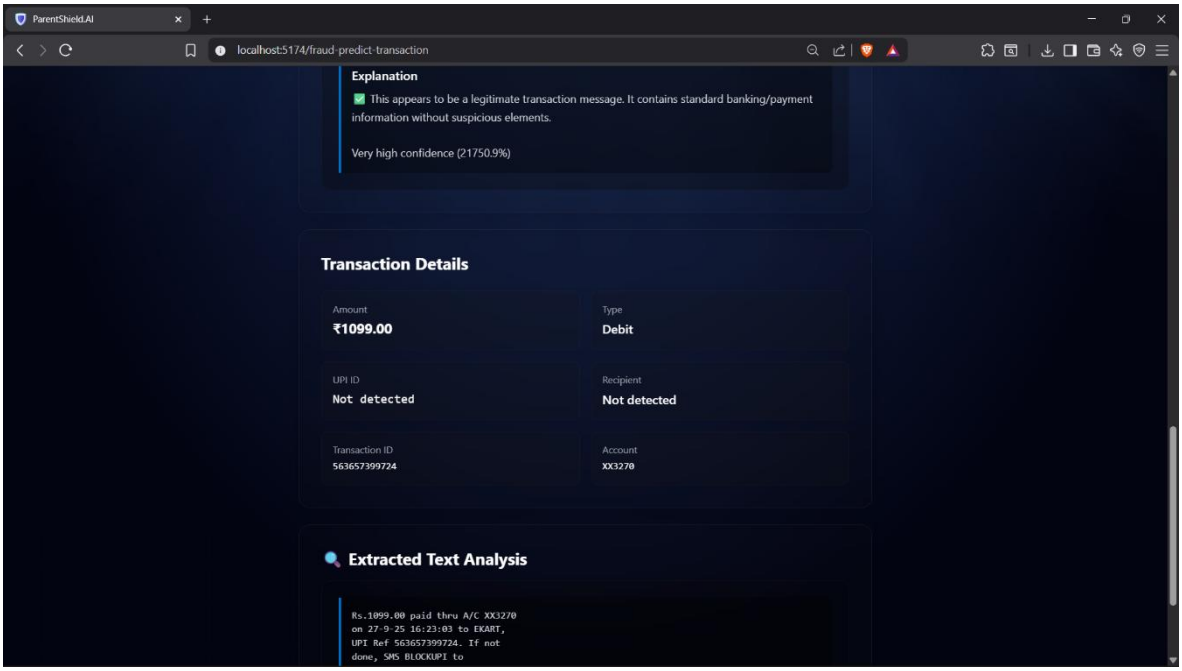

Fig 8.4.7: Continuation of output for transaction fraud detection

Chapter 9

Conclusion

ParentShield.AI bridges a vital gap in India's digital safety landscape by offering an intelligent, easy-to-use, and educational platform designed especially for parents and less tech-savvy users. By combining real-time fraud detection for messages and transactions with an interactive Digital Safety Library, it empowers individuals to make safer online choices. The system's technical backbone, which effectively applies a Random Forest Classifier for message fraud detection and an ML model for transaction prediction, successfully demonstrates the capability of AI in identifying and mitigating online threats. Its React-based frontend and Flask backend ensure a fast, seamless user experience, while OCR integration enables accurate text extraction from real-world message screenshots. Together, these innovations make ParentShield.AI not only a protective tool but also an educational resource promoting digital awareness and confidence.

Beyond its technical success, ParentShield.AI aligns with the United Nations Sustainable Development Goals 4, 9, and 16—advancing education, innovation, and security. It promotes digital inclusion by helping reduce online fraud, build trust in digital payments, and support vulnerable users in navigating technology safely. This project reflects how empathetic design and explainable AI can create lasting social impact. While the current system fulfills its goals effectively, continued updates and adaptive improvements will be key as fraud tactics evolve. Ultimately, ParentShield.AI sets the foundation for a safer, more informed, and inclusive digital future for all.

Chapter 10

Future Scope

This chapter outlines the strategic vision for ParentShield.AI, building upon the current platform's foundation. It begins with a transparent assessment of the project's present limitations, followed by a detailed, multi-phased roadmap for future development. This roadmap aims to address current challenges, expand capabilities, and solidify the platform's role as a comprehensive digital safety ecosystem.

ParentShield.AI can be enhanced with advanced AI models like BERT and LSTM for contextual nuance, real-time API integration with messaging apps like WhatsApp, and a community-driven threat intelligence hub. Adding native mobile applications, expanded multilingual support for Hinglish and regional languages, and improved OCR for low-quality images will further improve accuracy, accessibility, and user trust.

Dedicated Mobile Application The first step forward would be the development of a dedicated mobile application for Android and iOS. A mobile-first approach would allow users to receive real-time scanning alerts, get instant push notifications for threats, and access the platform's features on the go, greatly expanding accessibility and user engagement.

References

[1] Sharma, A., Gupta, R., & Chen, L., "Advancements in NLP for Real-Time Phishing and Smishing Detection in Messaging Apps", IEEE Transactions on Information Forensics and Security, Vol. 19, 2024

[2] Patel, S. K., & Krishnan, M., "A Comparative Analysis of Machine Learning Algorithms for UPI Transaction Fraud Detection", International Journal of Computer Science and Engineering, Vol. 11, 2023.

[3] Smith, R., "An Overview of the Tesseract OCR Engine", Proceedings of the Ninth International Conference on Document Analysis and Recognition (ICDAR), 2007