APTINOVA - AI-POWERED RECRUITMENT PLATFORM

Project Submitted to the
SRM University AP, Andhra Pradesh
for the partial fulfillment of the requirements to award the degree of

Bachelor of Technology
in
Computer Science & Engineering
School of Engineering & Sciences

submitted by

Ayon Sarkar(AP21110010286)

Sanjana Maini(AP21110010313)

Tanishk Yadav(AP21110010266)

Under the Guidance of

Dr. Ajay Bhardwaj

Department of Computer Science & Engineering
SRM University-AP
Neerukonda, Mangalgiri, Guntur
Andhra Pradesh - 522 240
May 2025

DECLARATION

I undersigned hereby declare that the project report **Aptinova - AI-Powered Recruitment Platform** submitted for partial fulfillment of the requirements for the award of degree of Bachelor of Technology in the Computer Science & Engineering, SRM University-AP, is a bonafide work done by me under supervision of Dr. Ajay Bhardwaj. This submission represents my ideas in my own words and where ideas or words of others have been included, I have adequately and accurately cited and referenced the original sources. I also declare that I have adhered to ethics of academic honesty and integrity and have not misrepresented or fabricated any data or idea or fact or source in my submission. I understand that any violation of the above will be a cause for disciplinary action by the institute and/or the University and can also evoke penal action from the sources which have thus not been properly cited or from whom proper permission has not been obtained. This report has not been previously formed the basis for the award of any degree of any other University.

Place	:	Date	: April 28, 2025
Name of student	: Ayon Sarkar	Signature	: .....
Name of student	: Sanjana Maini	Signature	: .....
Name of student	: Tanishk Yadav	Signature	: .....

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

SRM University-AP

Neerukonda, Mangalgiri, Guntur

Andhra Pradesh - 522 240

CERTIFICATE

This is to certify that the report entitled **Aptinova - AI-Powered Recruitment Platform** submitted by **Ayon Sarkar, Sanjana Maini, Tanishk Yadav** to the SRM University-AP in partial fulfillment of the requirements for the award of the Degree of Master of Technology in in is a bonafide record of the project work carried out under my/our guidance and supervision. This report in any form has not been submitted to any other University or Institute for any purpose.

Project Guide	Head of Department
Name : Dr. Ajay Bhardwaj	Name : Dr. Murali Krishna Enduri
Signature:	Signature: .....

ACKNOWLEDGMENT

I wish to record my indebtedness and thankfulness to all who helped me prepare this Project Report titled **Aptinova - AI-Powered Recruitment Platform** and present it satisfactorily.

I am especially thankful for my guide and supervisor Dr. Ajay Bhardwaj in the Department of Computer Science & Engineering for giving me valuable suggestions and critical inputs in the preparation of this report. I am also thankful to Dr. Murali Krishna Enduri, Head of Department of Computer Science & Engineering for encouragement.

My friends in my class have always been helpful and I am grateful to them for patiently listening to my presentations on my work related to the Project.

Ayon Sarkar, Sanjana Maini, Tanishk Yadav,
(Reg. No. AP21110010286, AP21110010313, AP21110010266)
B. Tech.
Department of Computer Science & Engineering
SRM University-AP

ABSTRACT

Aptinova is a full-stack web-based recruitment platform designed to streamline the hiring process for both HR professionals and job seekers. In today's fast-paced recruitment landscape, organizations face challenges in efficiently shortlisting the right candidates from a large pool of applicants. This often necessitates multiple rounds of tests and interviews to ensure optimal hiring decisions. Simultaneously, candidates struggle with managing applications across various platforms and tracking their progress. Aptinova addresses these challenges by providing a unified, intelligent platform offering a seamless user experience for both HR users and candidates. Key features include AI-assisted resume parsing, automated candidate screening based on job relevance, integrated online assessments with proctoring capabilities, interview scheduling, and comprehensive analytics for HR managers. The platform utilizes a modern tech stack including Node.js, Express, PostgreSQL, React (Next.js), and leverages AI models via Google Gemini for enhanced decision-making. Aptinova aims to make recruitment more efficient, objective, and user-friendly.

CONTENTS

LIST OF TABLES

LIST OF FIGURES

Chapter 1

INTRODUCTION TO THE PROJECT

Aptinova is a platform designed to make hiring process easier and seamless. It has two major user groups: Candidate and HR. The candidates can register, apply for jobs, upload their resumes, get a matching score and view their applications. While the HR can post jobs, add other HR's into their team, and view applicants.

1.1 OVERVIEW

Aptinova is an innovative, full-stack web application designed to revolutionize the recruitment landscape by leveraging Artificial Intelligence (AI) and modern web technologies. It serves as a comprehensive platform connecting job seekers (Candidates) with employers (HR professionals and HR Managers), streamlining the entire hiring lifecycle from job posting to onboarding. The platform addresses inefficiencies in traditional hiring, such as manual resume screening, application management overhead, and potential biases, by offering intelligent automation, objective assessments, and data-driven insights.

1.2 PROJECT GOALS AND OBJECTIVES

The primary goal of the Aptinova project is to create a seamless, efficient, and intelligent hiring ecosystem. Specific objectives include:

- **Enhance Efficiency:** Automate repetitive tasks like resume parsing and initial screening to reduce time-to-hire.
- **Improve Candidate Experience:** Provide candidates with a centralized platform to manage applications, receive timely updates, and understand job relevance through matching scores.
- **Empower HR Professionals:** Equip HR teams with tools for effective applicant tracking, assessment management, interview scheduling, and collaborative decision-making.
- **Reduce Bias:** Implement objective, skill-based assessments and AI-driven analysis to minimize unconscious bias in candidate selection.
- **Provide Data-Driven Insights:** Offer comprehensive analytics to HR Managers for optimizing recruitment strategies and understanding hiring trends.
- **Ensure Security and Integrity:** Implement robust security measures, including secure authentication (passkeys) and proctoring for online assessments.

1.3 SCOPE

The scope of the Aptinova platform encompasses the following core functionalities:

- **User Management:** Registration, login, profile management for Candidates, HR, and HR Managers. Role-based access control.
- **Job Management:** Creation, editing, publishing, and management of job postings by HR/HRM users.

- **Candidate Application:** Job search, filtering, application submission, and tracking application status.
- **AI Integration:** Resume parsing, job description analysis, personality trait analysis, AI-assisted scoring (including planned weighted average), and subjective answer grading.
- **Assessment Module:** Creation/selection of hiring tests (MCQ, Text, Code), test assignment, secure test-taking environment with proctoring, automated/AI grading.
- **Interview Scheduling:** Integration with Google Calendar for scheduling and sending invites.
- **Applicant Tracking System (ATS):** Kanban workflow view, status updates, applicant filtering, profile viewing.
- **Team & Organization Management (HRM):** Adding/removing HR team members, managing organization profile and settings.
- **Analytics & Reporting (HRM):** Dashboards displaying key hiring metrics, trends, and predictive insights.
- **Subscription Management (HRM/Candidate):** Handling different subscription tiers (Free, Pro, Startup, Enterprise) via Razorpay.

1.4 TARGET USERS

The platform is designed for three primary user groups:

- **Candidates:** Individuals seeking employment opportunities across various industries.

- **HR Professionals:** Recruiters and hiring team members responsible for managing specific job postings and interacting with applicants within an organization.
- **HR Managers (HRM):** Senior HR personnel or administrators responsible for overseeing the entire recruitment process within an organization, managing the HR team, setting up organization profiles, managing subscriptions, and accessing advanced analytics.

Table 1.1: User Roles and Key Permissions

Role	Key Capabilities	Primary Goal
Candidate	Register, Login (incl. Social, Passkey), Manage Profile (Details, Resume), Search/Filter Jobs, Apply for Jobs, Take Assessments, Track Applications	Find and apply for suitable job opportunities.
HR	Manage assigned Job Postings, Review Applicants for assigned jobs, Update Applicant Status (ATS Kanban), Assign Tests, Schedule Interviews, View Test Results, Submit Interview Feedback	Manage the operational hiring process for specific roles.
HRM	All HR capabilities + Manage entire Organization Profile & Settings, Manage HR Team members, View comprehensive Analytics & Reports, Manage Subscription Plans, Configure System Settings (e.g., Scoring Weights)	Oversee and optimize the entire recruitment strategy and platform usage for the organization.

Chapter 2

MOTIVATION

2.1 WHY IS APTINOVA IS THE NEED OF THE HOUR

The modern recruitment landscape presents significant challenges for both employers and job seekers. Traditional methods often suffer from inefficiencies, biases, and a disjointed user experience. Key problems motivating the development of Aptinova include:

- **Information Overload for Recruiters:** HR teams are frequently inundated with hundreds of resumes for a single job opening, making manual screening tedious, time-consuming, and prone to errors. Identifying the most suitable candidates becomes a significant bottleneck. Hence, Aptinova caters to the problem with a intuitive drag-and-drop for managing applicants as seen in Fig. 4.5

- **Candidate Application Fatigue:** Job seekers often struggle to find roles that genuinely align with their skills and aspirations. Applying blindly to numerous positions across different platforms is inefficient and discouraging, reducing the likelihood of success. Tracking application statuses across multiple sites adds further complexity.

- **Reliability of Existing Platforms:** Current job application platforms can be unreliable, lacking features for comprehensive assessment, efficient communication, or robust management.

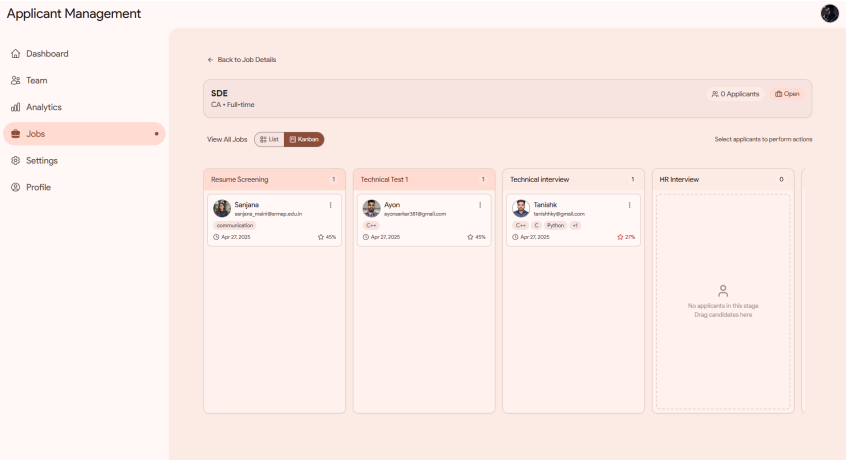

Figure 2.1: Application Management for HRs

- **Subjectivity and Bias:** Traditional screening can be susceptible to unconscious biases, potentially overlooking qualified candidates who don't fit a specific mold.

- **Lack of Integrated Workflow:** Managing different stages of hiring – screening, testing, interviews, feedback – often requires juggling multiple tools and platforms.

Aptinova was conceived to address these pain points by creating a unified, intelligent, and user-centric platform. The core motivation is to simplify and modernize recruitment, fostering efficient interaction between applicants and recruiters.

2.2 ADDRESSING REAL-WORLD PROBLEMS

The project stems from observing the real-time challenges faced during the hiring phase. By providing a solution that streamlines application management, offers skill-based matching, integrates assessments, and simplifies communication, Aptinova aims to provide tangible value. It serves

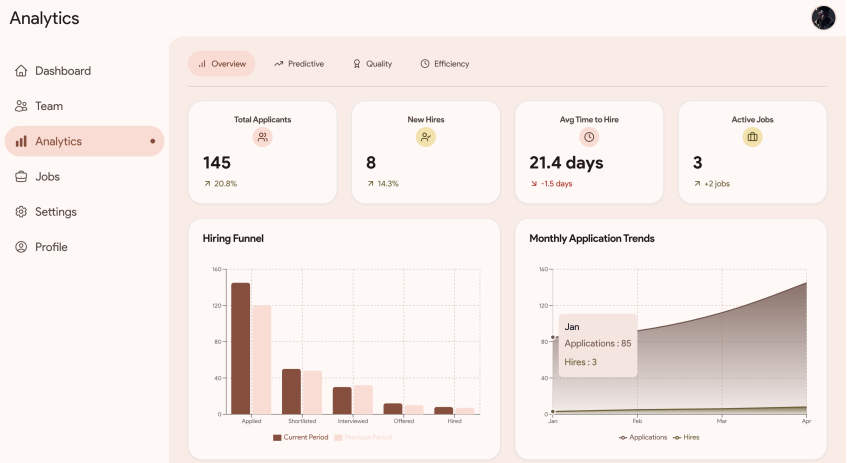

Figure 2.2: Active Analytics for HR Manager

as a platform for applying technical skills (like web development, AI integration, database management) to solve a practical, widespread problem.

2.3 PLATFORM FOR INNOVATION AND SKILL DEVELOPMENT

Developing Aptinova provided an opportunity to:

- **Explore Technological Trends:** Research and implement modern technologies like AI (Gemini for parsing, grading, analysis), secure authentication (Passkeys/WebAuthn), cloud services (Cloudinary, Vercel), payment gateways (Razorpay), and web frameworks (Next.js, Express).
- **Enhance Core Skills:** Strengthen skills in full-stack development, database design, API integration, AI model utilization, UI/UX design, and project management.
- **Promote Problem-Solving:** Design and implement solutions for complex requirements like integrated testing, proctoring, weighted scor-

ing, and multi-user workflows.

- **Foster Future Ideas:** Serve as a foundation for potential future development or even startup ventures in the HR technology space.

2.4 BENEFITS OF THE APTINOVA APPROACH

The Aptinova platform offers several key benefits:

- **For Employers/HR:** Reduced time-to-hire, improved quality of hires through objective assessments, streamlined workflow, reduced screening bias, data-driven decision-making via analytics, better team collaboration.
- **For Candidates:** Centralized application management, improved job matching, clearer understanding of application status, fair skill-based evaluation, enhanced user experience.
- **Overall:** Creates a more efficient, transparent, and equitable hiring market.

Chapter 3

LITERATURE SURVEY

3.1 INTRODUCTION

This chapter reviews existing literature and platforms related to online recruitment, Applicant Tracking Systems (ATS), AI in hiring, online assessment tools, and security mechanisms like proctoring and passkeys. The purpose is to situate Aptinova within the current technological landscape and identify areas where it offers unique contributions or improvements.

3.2 EXISTING RECRUITMENT PLATFORMS

- **Overview:** Platforms like LinkedIn, Indeed, Glassdoor, Monster, etc., dominate the online job market.
- **Strengths:** Large user base, extensive job listings, established brand recognition.
- **Weaknesses:** Often act as job boards with limited integrated ATS features, varying levels of application tracking for candidates, potential for application spam, limited built-in assessment tools.
- **Aptinova's Relation:** Aptinova differentiates itself by integrating the job board functionality with a comprehensive ATS, AI-driven matching/screening, and built-in assessment/interview tools within a single platform.

3.3 APPLICANT TRACKING SYSTEMS (ATS)

- **Overview:** Software like Greenhouse, Lever, Workday Recruiting, Taleo focus on managing the hiring workflow for employers [21].
- **Strengths:** Workflow automation, candidate database management, reporting features, compliance tools.
- **Weaknesses:** Can be expensive, UI/UX varies, often lack sophisticated candidate-facing features or integrated AI assessment capabilities comparable to Aptinova's goals. Candidate experience can sometimes be poor ("black hole" effect).
- **Aptinova's Relation:** Aptinova incorporates core ATS functionalities but aims for a more modern UI/UX, deeper AI integration (parsing, scoring, analysis), and a stronger focus on the candidate experience alongside employer needs.

3.4 ARTIFICIAL INTELLIGENCE (AI) IN RECRUITMENT

- **Resume Parsing:** Many ATS systems use basic keyword matching or rule-based parsing. More advanced tools leverage NLP/ML for better data extraction.
- **Candidate Matching:** AI algorithms are used to match candidates to jobs based on skills, experience, and other factors, aiming to predict job success.
- **Chatbots:** Used for initial screening, answering candidate questions, and scheduling.

- **Assessment & Scoring:** AI can be used to generate test questions, grade coding challenges, analyze video interviews (sentiment, keywords), and score subjective answers.

- **Bias Mitigation:** Research focuses on developing fair AI algorithms to reduce human bias in screening.

- **Aptinova's Relation:** Aptinova heavily utilizes AI (specifically Google Gemini models[15]) for resume parsing, job description analysis, trait analysis, subjective answer grading, and aims to provide AI-assisted scoring, positioning it at the forefront of AI adoption in recruitment platforms.

3.5 ONLINE ASSESSMENT AND PROCTORING

- **Assessment Platforms:** Tools like HackerRank, Codility, TestGorilla offer specialized technical and cognitive assessments.

- **Proctoring Solutions:** Services like ProctorU, Examity provide remote invigilation using webcams, microphones, and screen monitoring to ensure test integrity. Methods include live proctoring, automated AI proctoring, and record-and-review.

- **Aptinova's Relation:** Aptinova integrates assessment capabilities directly into the hiring workflow, including coding challenges and potentially other types (MCQ, Text). It incorporates its own proctoring features (fullscreen, tab switch detection, basic webcam/mic monitoring based on useProctoring.js) rather than relying on third-party proctoring services, offering a more seamless experience.

3.6 SECURE AUTHENTICATION: PASSKEYS (WEBAUTHN)

- **Overview:** WebAuthn[11] is a standard enabling passwordless authentication using public-key cryptography, often via biometrics or hardware keys (FIDO2). Passkeys are platform-synchronized credentials based on this standard.
- **Benefits:** Phishing-resistant, more secure than passwords, improved user experience.
- **Adoption:** Increasingly adopted by major platforms (Google, Apple, Microsoft).
- **Aptinova's Relation:** The backend code includes dependencies (@simplewebauthn/server, @passwordless-id/webauthn) and routes (auth.js) explicitly implementing passkey registration and authentication, making Aptinova an early adopter of this modern security standard within the recruitment platform space.

3.7 LITERATURE GAPS AND APTINOVA'S CONTRIBUTION

While many platforms address parts of the recruitment process, few offer a deeply integrated, AI-first solution covering the entire lifecycle from job posting and candidate sourcing through advanced assessments (with proctoring) and interviews to analytics and onboarding, all while supporting modern security like passkeys. Aptinova aims to fill this gap by providing a unified, intelligent, and secure platform for both candidates and employers.

Chapter 4

DESIGN AND METHODOLOGY

4.1 SYSTEM ARCHITECTURE

Aptinova employs a modern, full-stack web application architecture designed for scalability, maintainability, and user experience. It follows a robust 3-Tier Client-Server Model, separating concerns into distinct layers: Presentation, Logic (Application), and Data. This architecture promotes modularity, allowing independent development, deployment, and scaling of each tier. Communication between the client (Presentation Tier) and the server (Logic Tier) occurs primarily via RESTful APIs over HTTPS.

4.1.1 3-Tier Client-Server Architecture

The 3-tier architecture is a well-established software architecture pattern that logically separates an application into three distinct physical or logical computing tiers:

- Presentation Tier (Client): This is the topmost level, responsible for interacting directly with the end-user. Its primary role is to display information to the user and collect input from them. It focuses on the User Interface (UI) and User Experience (UX). It typically runs on the user's device (e.g., web browser, mobile app).

- **Logic Tier (Application Server / Middle Tier):** This tier acts as the intermediary between the Presentation Tier and the Data Tier. It contains

the core business logic, processes user input received from the presentation layer, makes decisions, performs calculations, and orchestrates communication with the data tier and external services. This tier often hosts the web server and API endpoints.

- **Data Tier (Database Server):** This tier is responsible for storing, retrieving, and managing the application's data. It typically consists of a database management system (DBMS) and the data itself. The logic tier interacts with this tier to perform CRUD (Create, Read, Update, Delete) operations.

Benefits of 3-Tier Architecture:

- **Modularity & Maintainability:** Changes in one tier (e.g., UI redesign) have minimal impact on others, simplifying development and maintenance.
- **Scalability:** Each tier can be scaled independently based on specific load requirements (e.g., add more web servers without changing the database).
- **Flexibility & Reusability:** The logic tier can serve multiple presentation tiers (e.g., web and mobile clients). Business logic is centralized and reusable.
- **Security:** Isolating the data tier behind the logic tier enhances security, as the client does not directly access the database. Access control and validation are enforced in the logic tier.
- **Improved Development Workflow:** Different teams can work concurrently on different tiers (frontend, backend, database).

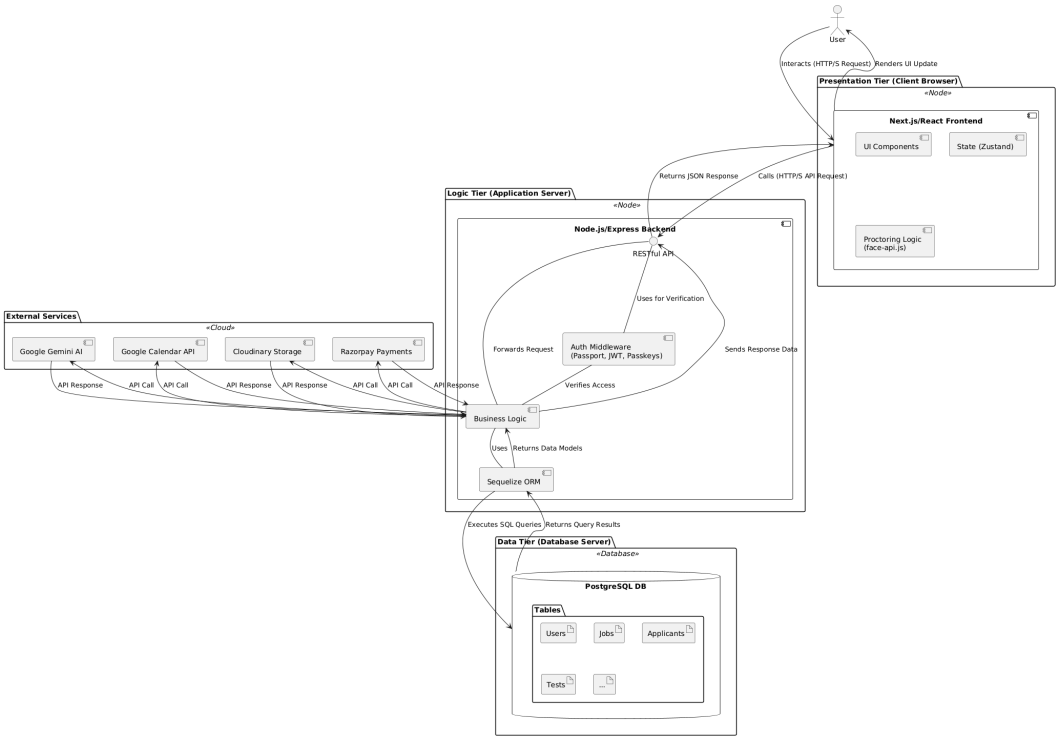

Figure 4.1: Aptinova's 3-tier architecture

4.1.2 Aptinova's Implementation of the 3-Tier Architecture

Aptinova meticulously implements the 3-tier architecture, leveraging specific technologies within each tier to achieve its goals:

1. **Presentation Tier (Client - Aptinova Frontend):**

   - **Description:** This tier is the user-facing web application that runs in the user's browser (Chrome, Firefox, etc.). It is responsible for rendering the entire user interface for Candidates, HR, and HR Managers.
   - **Responsibilities:**
     - Displaying job listings, application forms, candidate profiles, dashboards, analytics charts, test interfaces, etc.
     - Handling user interactions (button clicks, form submissions, drag-and-drop in Kanban).

- Performing client-side validation for forms to provide immediate feedback (`app/utils/validation.ts`).
- Managing client-side state (e.g., user authentication status, theme preferences, fetched data) using Zustand[7] (`app/store.ts`).
- Making asynchronous API calls (using Axios, likely wrapped in service functions like `services/jobService.js`) to the Logic Tier to fetch data or trigger actions.
- Rendering dynamic content based on API responses.
- Implementing client-side proctoring logic using libraries like `face-api.js` and Web APIs (`hooks/useProctoring.js`).
- Handling Material You dynamic theming (`components/ThemeInitializer.jsx`, `utils/colourgenerator.js`).

- **Technologies:** Next.js [1], React [2], Tailwind CSS[6], Zustand, Axios, Framer Motion, Recharts, Monaco Editor (@monaco-editor/react), @hello-pangea/dnd, face-api.js, TensorFlow.js.
- **Deployment:** Hosted on Vercel, leveraging its CDN for fast global delivery of static assets and serverless functions for Next.js features.

2. **Logic Tier (Application Server - Aptinova Backend):**

- **Description:** This tier is the central hub of Aptinova's operations, running as a Node.js application. It exposes RESTful API endpoints that the Presentation Tier consumes.
- **Responsibilities:**
  - Receiving and processing HTTP requests from the Presentation Tier.

- Authenticating and authorizing users (using Passport.js, JWT[9], Passkeys middleware - `middleware/auth.js, routes/auth.js`).
- Implementing core business logic:
  - Managing job postings (CRUD operations - `routes/jobRoutes.js`).
  - Handling candidate applications and status updates (`routes/applicantRoutes.js`).
  - Processing user profile creation and updates (`routes/candidateRoutes.js`, `routes/hrRoutes.js`, `routes/hrmroutes.js`).
  - Managing hiring tests (creation, starting/ending tests, scoring - `routes/hiringTestRoutes.js`).
  - Executing submitted code in a sandboxed environment (`routes/codeExecutionRoutes.js`, `utils/codeExecution/codeRunner.js`).
  - Orchestrating interview scheduling (`routes/interviewsRoutes.js`).
  - Generating analytics data (`routes/hrmroutes.js`).
  - Handling subscription payments (`routes/paymentRoutes.js`).
- Interacting with the Data Tier (PostgreSQL) via the Sequelize ORM[5] to fetch and persist data (`models/*.js`).
- Integrating with external services:
  - Google Gemini for AI tasks [15](parsing, analysis, grading - `routes/parserRoutes.js`, `routes/hrRequestRoutes.js`, `routes/validateSubjectiveRoutes.js`).
  - Google Calendar API[16] for scheduling (`routes/interviewsRoutes.js`).

        * Cloudinary[17] for file storage (`routes/get-started.js`).
        * Razorpay for payments (`routes/paymentRoutes.js`).
        * Nodemailer for sending emails (`utils/emailService.js`).
    - Formatting data and sending responses (typically JSON) back to the Presentation Tier.
- **Technologies:** Node.js[3], Express.js[4], Sequelize, PostgreSQL (pg driver), Passport.js[8], JWT (jsonwebtoken), bcrypt, @simplewebauthn/server[1 Axios, Nodemailer, Cloudinary SDK, Razorpay SDK[18], Google APIs (googleapis, @google/generative-ai).
- **Deployment:** Hosted on Vercel as serverless functions (vercel.json, api/index.js).

3. **Data Tier (Database Server):**
    - **Description:** This tier comprises the PostgreSQL relational database that persistently stores all application data.
    - **Responsibilities:**
        - Storing structured data for users (Candidates, HR, HRMs), Organizations, Jobs, Applicants, Tests, Interviews, Passkeys, Subscriptions, etc. (defined in `models/*.js`).
        - Ensuring data integrity through constraints, relationships (defined via Sequelize associations), and transactions.
        - Providing efficient data retrieval and updates based on queries initiated by the Logic Tier (via Sequelize).
        - Handling data backups and recovery (managed by the database hosting provider or administrator).
    - **Technologies:** PostgreSQL, Sequelize (as the ORM interacting with the DB).

- **Deployment:** Likely hosted on a cloud database service (e.g., Vercel Postgres, AWS RDS, Google Cloud SQL, Supabase) or a self-managed server, accessible only by the Logic Tier.

4.1.3 Interactions and Benefits in Aptinova

The clear separation between these tiers in Aptinova provides significant advantages:

- **Frontend-Backend Decoupling:** The Next.js frontend communicates with the Node.js backend exclusively through well-defined REST APIs. This allows the frontend team to focus on UI/UX improvements using React/Tailwind/Material You without needing deep knowledge of the backend implementation details, and vice-versa.

- **Scalability:** If Aptinova experiences high user traffic, the Vercel frontend infrastructure can scale automatically. Similarly, the backend serverless functions can scale horizontally. If database load becomes a bottleneck, the PostgreSQL instance can be scaled vertically or horizontally (e.g., read replicas) independently of the other tiers.

- **Technology Flexibility:** The frontend could theoretically be rewritten using a different framework (e.g., Vue, Angular) without impacting the backend, as long as it consumes the same API. Likewise, parts of the backend logic could be refactored or even replaced without requiring immediate frontend changes.

- **Enhanced Security:** User browsers never connect directly to the PostgreSQL database. All database interactions are mediated by the backend (Logic Tier), which enforces authentication, authorization

(`middleware/auth.js`), and input validation, reducing the attack surface. Sensitive credentials (API keys, database connection strings) reside only in the backend environment.

- **Centralized Business Logic:** All core recruitment processes, AI integrations, and data manipulations are handled consistently within the Node.js backend, ensuring uniformity regardless of how the user interacts with the frontend.

By adopting the 3-tier architecture, Aptinova establishes a robust, scalable, and maintainable foundation capable of supporting its complex features and future growth.

4.2 DATABASE DESIGN

The database schema, managed via Sequelize models, is designed to support the platform's core functionalities. Key entities include:

- **Organizations:** Stores company details, including profile, contact info, subscription status, and subdomain.
- **Users (Implicitly represented by Candidates, HRs, HRManagers):** Store user credentials, authentication tokens (Google), status, and potentially basic profile info.
- **Candidates:** Detailed candidate profiles including personal info, experience, education, skills, documents (resume URL), social links, subscription status, etc.
- **HRs:** HR team member profiles linked to an Organization.
- **HRManagers:** HR Manager profiles linked to an Organization, with potentially higher privileges.

- **Jobs:** Job posting details including title, description, requirements, location, salary, deadlines, status, associated Organization/HR, and the defined hiringProcess.
- **Applicants:** Represents a candidate's application to a specific job, linking Candidates and Jobs. Stores application status, score, test results (start/end time, warnings), reference to assigned HiringTest, and the applicant-specific instance of the hiringProcess.
- **HiringTests:** Stores test definitions, including name, description, duration, passing score, questions (JSONB format), linked to a Job and Organization.
- **Interviews:** Details about scheduled interviews, linking Job, Applicant, Candidate, Organization. Includes summary, description, interviewers, date/time, meeting link (Google Meet), feedback, score, status.
- **Passkeys:** Stores `WebAuthn`[11] credential information for passwordless login.
- **VerificationCodes, InvalidTokens, PasskeyChallenges, WebAuthnSessions:** Support authentication and security processes.
- **SubscriptionHistory:** Tracks changes in subscription plans.
- **Entity-Relationship Diagram (ERD):** The Entity Relationship diagram for Aptinova's database can be seen in Figure 4.2

4.3 UI/UX DESIGN

The frontend is designed with user experience as a priority, utilizing:

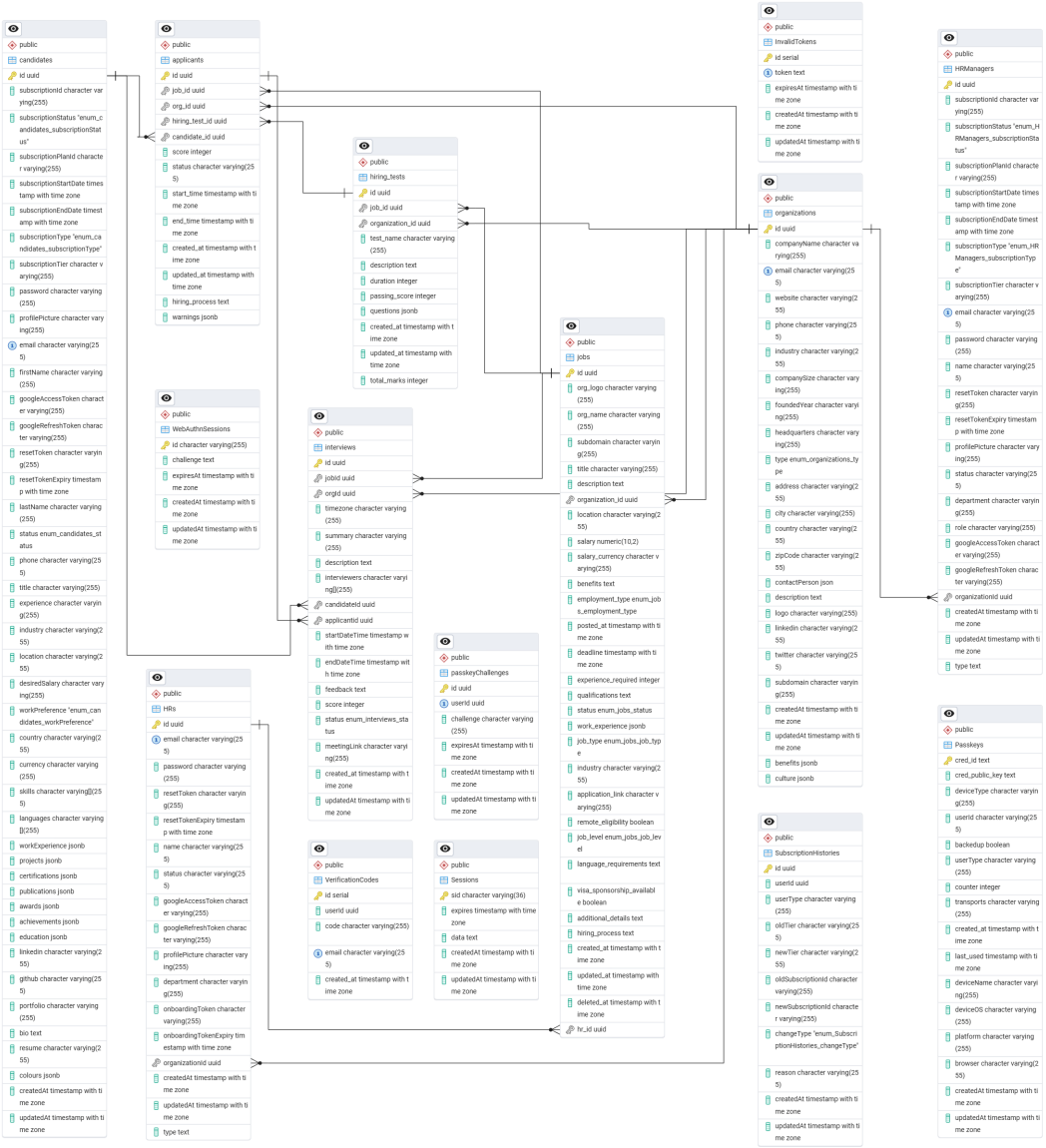

Figure 4.2: Entity Relationship Diagram

- **Framework:** Next.js with React for building interactive and performant user interfaces.

- **Styling:** Tailwind CSS[6] for a utility-first approach, enabling rapid development and consistent styling.

- **Theming:** Material You [19] dynamic theming allows personalization based on user preferences or content (e.g., profile picture), providing a modern and engaging aesthetic. The implementation uses CSS variables defined in `globals.css` and likely managed via the Zustand store and ThemeInitializer.

- **Responsiveness:** Designed to work seamlessly across different screen sizes (desktop, mobile, tablets) using Tailwind's responsive utilities and potentially adaptive component structures (e.g., sidebar vs. bottom nav).

- **Component Library:** Likely uses Shadcn/ui or a similar library for prebuilt, accessible components, augmented with custom components.

- **Animation:** Framer Motion is used for subtle animations and transitions, enhancing user engagement (e.g., page transitions, button interactions, modal popups).

- **UI Mockups/Screenshots:** Aptinova's intuitive UI and focus on performing automated and impenetrable tests avoiding any kind of malpractive can be seen in the Fig. 4.3 showing Dashboard for Candidate, Fig. 4.4 showing Testing view(Candidate's view) depicting screenshot detection and warnings at the corner, Fig. 4.5 depicting HR's Application Management View and Fig. 4.6 showing Applicant's Progress Window(HR's View).

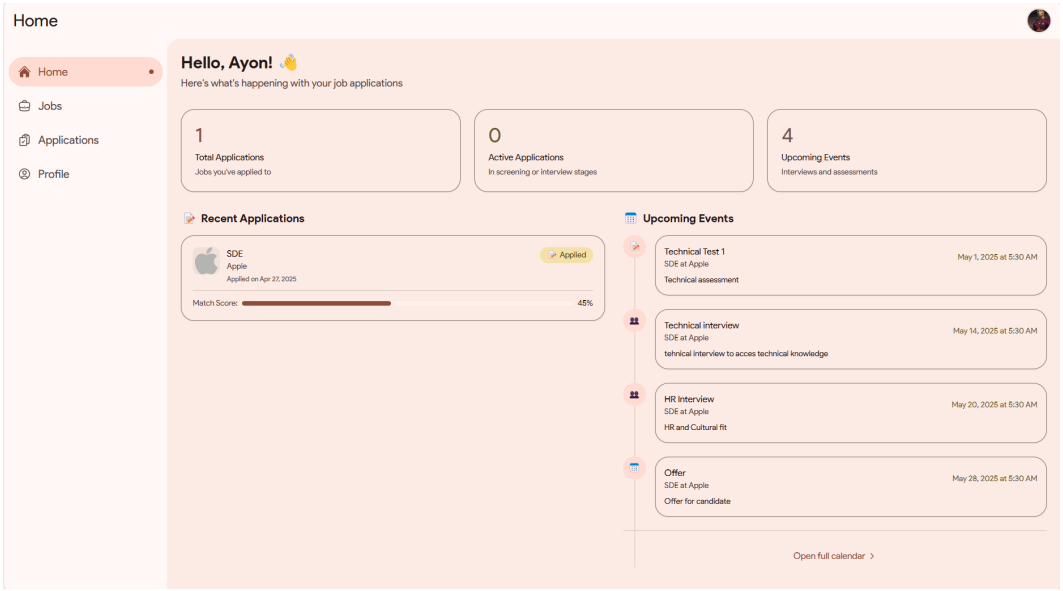

Figure 4.3: Dashboard for Candidate

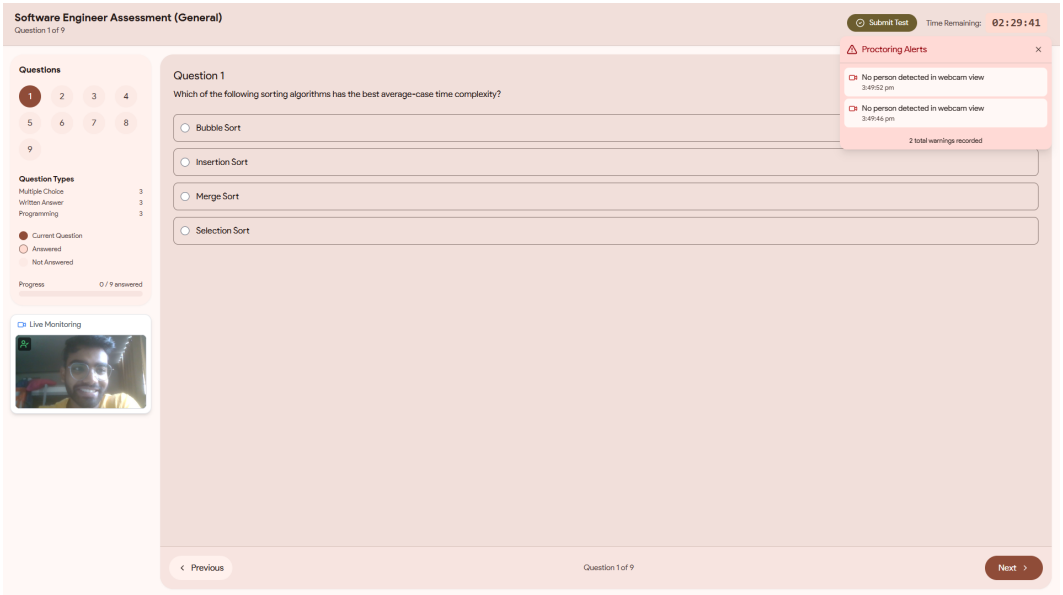

Figure 4.4: Testing view(Candidate's view)

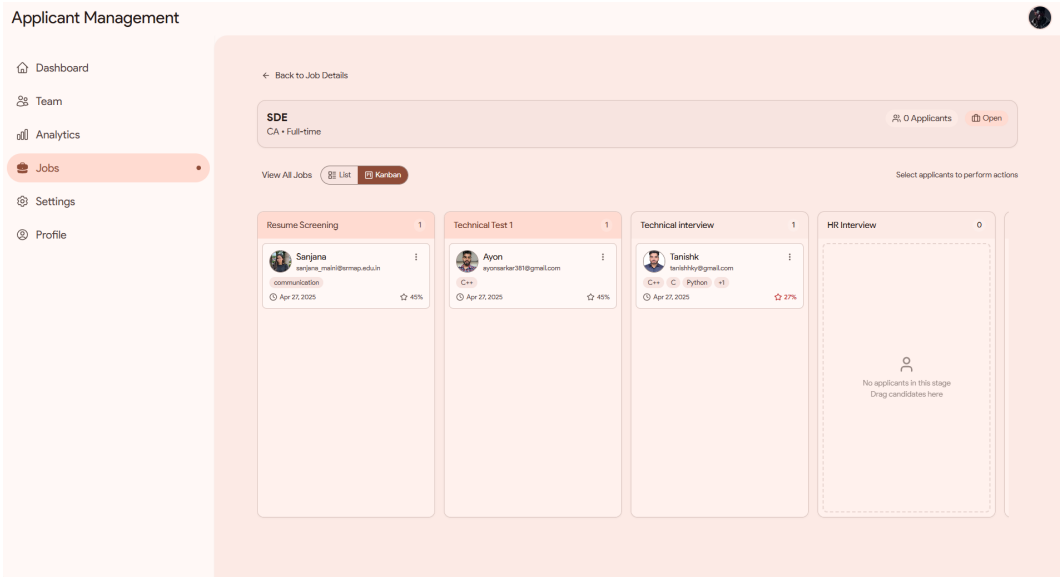

Figure 4.5: HR's Application Management View

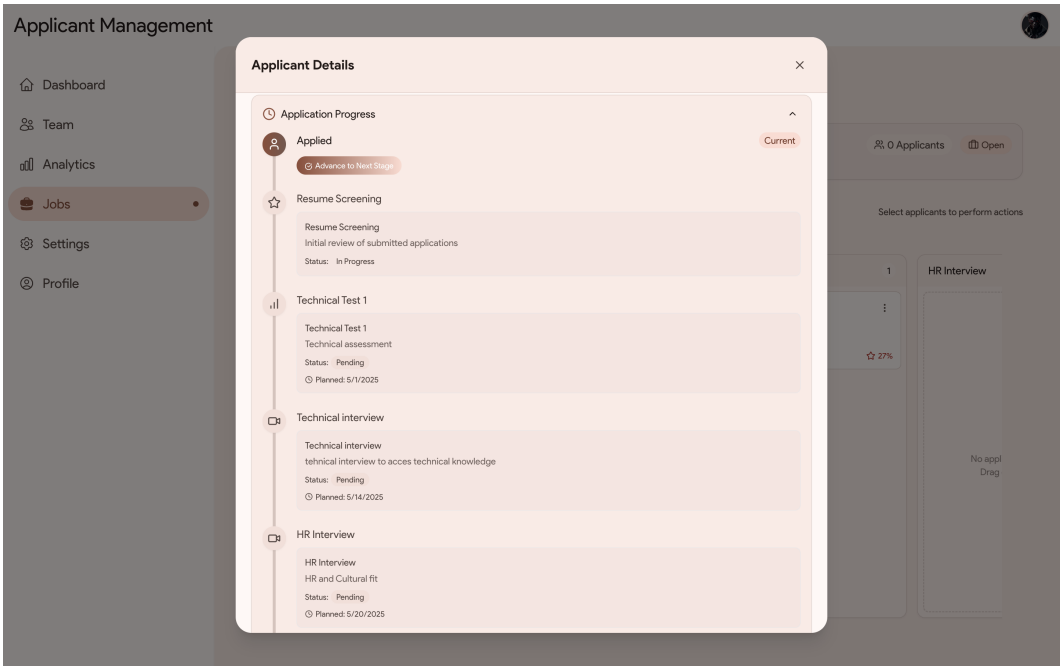

Figure 4.6: Applicant's Progress Window(HR's View)

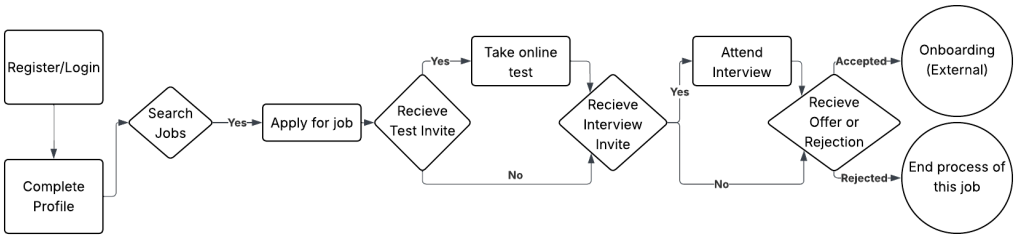

Figure 4.7: Candidate Workflow

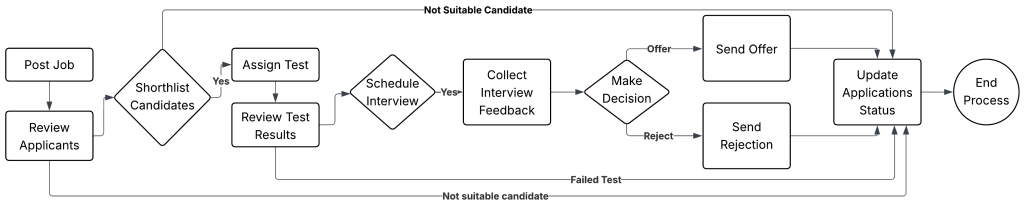

Figure 4.8: HR Workflow

4.4 WORKFLOW DESIGN

- **Candidate Workflow:** Candidate workflow is illustrated in Fig. 4.7
- **HR Workflow:** HR workflow is illustrated in Fig. 4.8
- **HRM Workflow:** HR Manager workflow is illustrated in Fig. 4.9
- **Data Flow Diagrams (DFDs):** DFD for processing job application and scheduling interviews can be seen in Fig. 4.10 and Fig. 4.11 shows HR's and Applicant's interactions with Assessment module. Fig. 4.14 shows user's interaction with the authentication module and fig. 4.15 shows

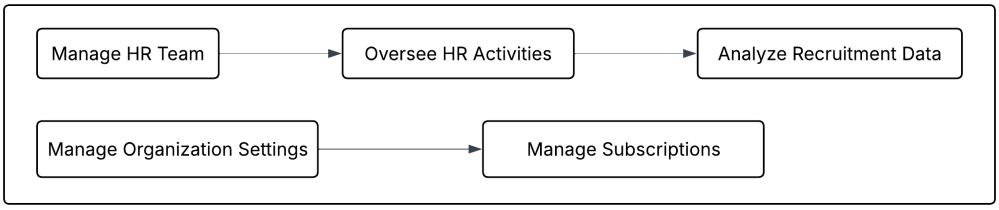

Figure 4.9: HR Manager Workflow

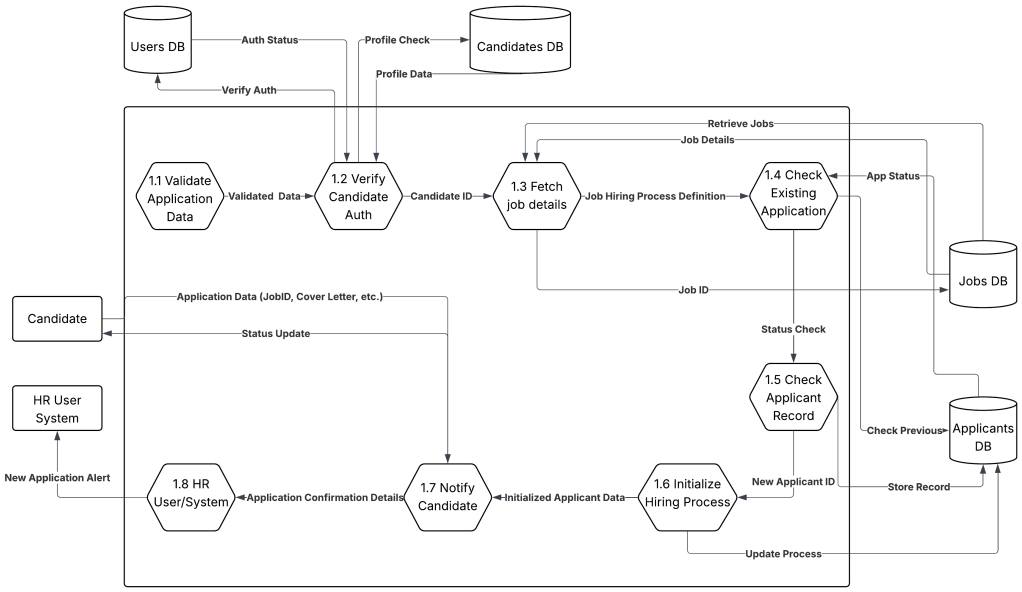

Figure 4.10: Data Flow Diagram(DFD) for Processing of Job Application

applicant's interaction with job application module. Fig. 4.16 shows HR Manager's interactions with the system for various functions.

- **Use Case Diagrams:** Fig. 4.12 depicts the UCD (Use Case Diagram) for HR's Application Tracking Module, fig. 4.13

4.5 METHODOLOGY

The project adopts a component-based architecture (React/Next.js) for the frontend and a modular, service-oriented approach (Express routes per resource) for the backend. Key methodological choices include:

- **API-First Design:** Backend provides RESTful APIs consumed by the frontend, allowing potential future development of other clients (e.g., mobile app) [20].
- **ORM Usage:** Sequelize abstracts database interactions, improving developer productivity and database portability.

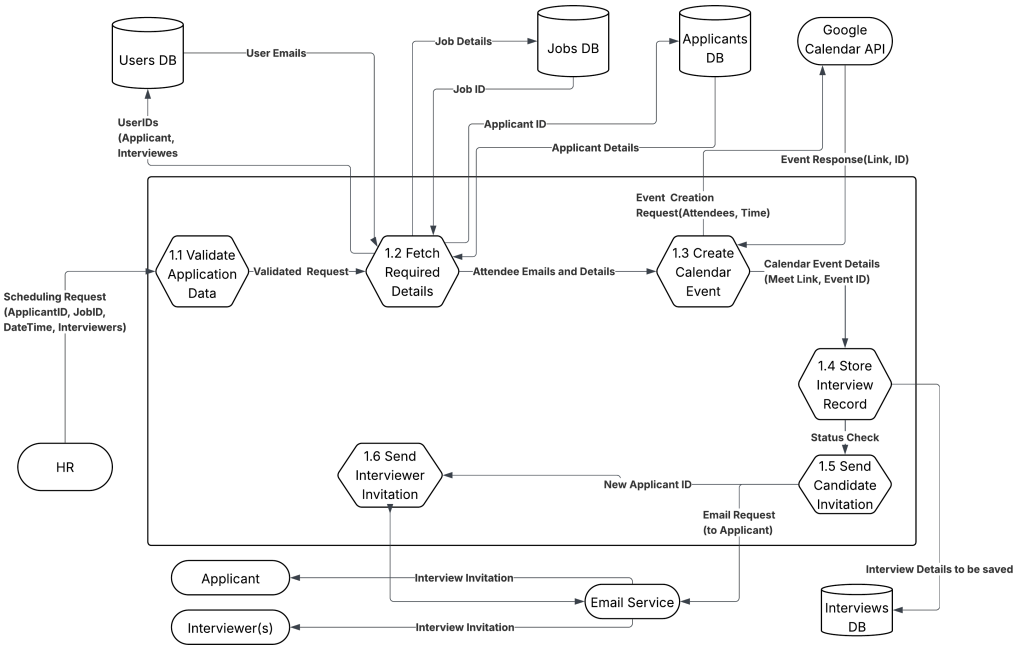

Figure 4.11: Data Flow Diagram(DFD) for Interview Scheduling

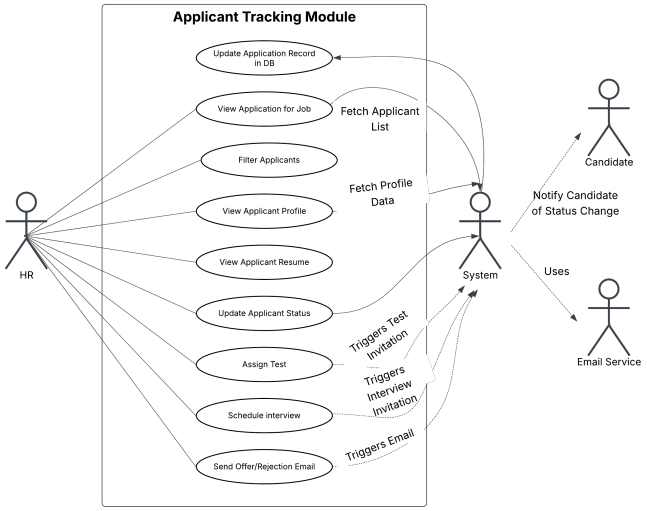

Figure 4.12: Use Case Diagram for Application Tracking Module

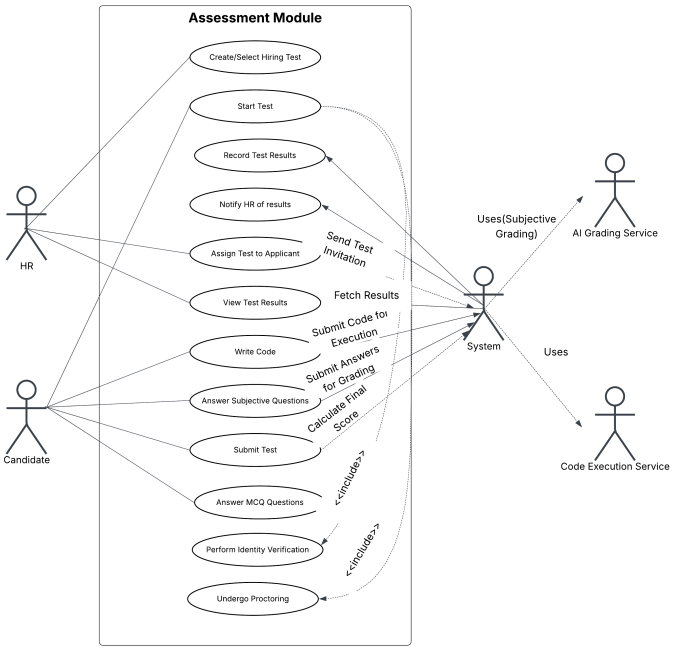

Figure 4.13: Use Case Diagram for Assessment Module

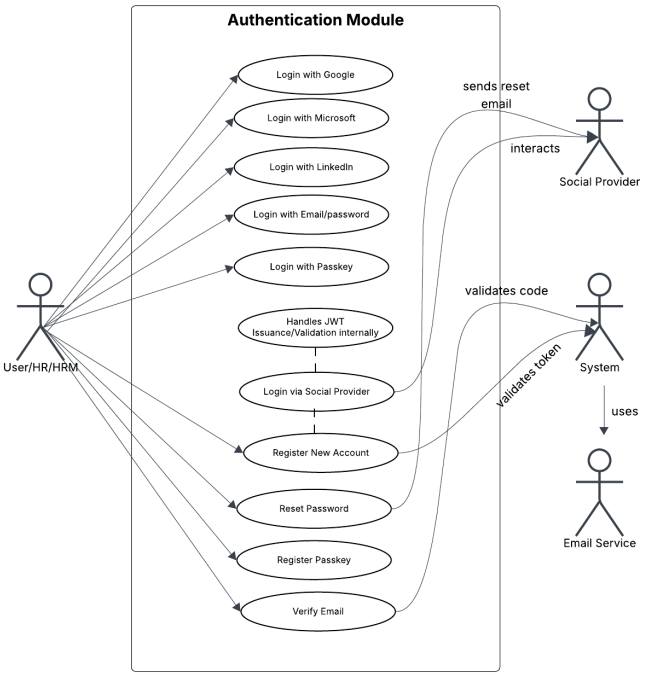

Figure 4.14: Use Case Diagram for Authentication Module

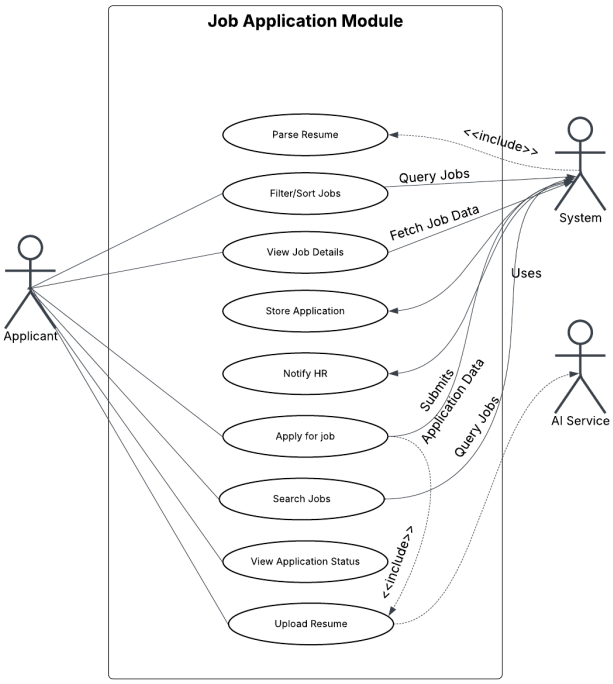

Figure 4.15: Use Case Diagram for Job Application Module

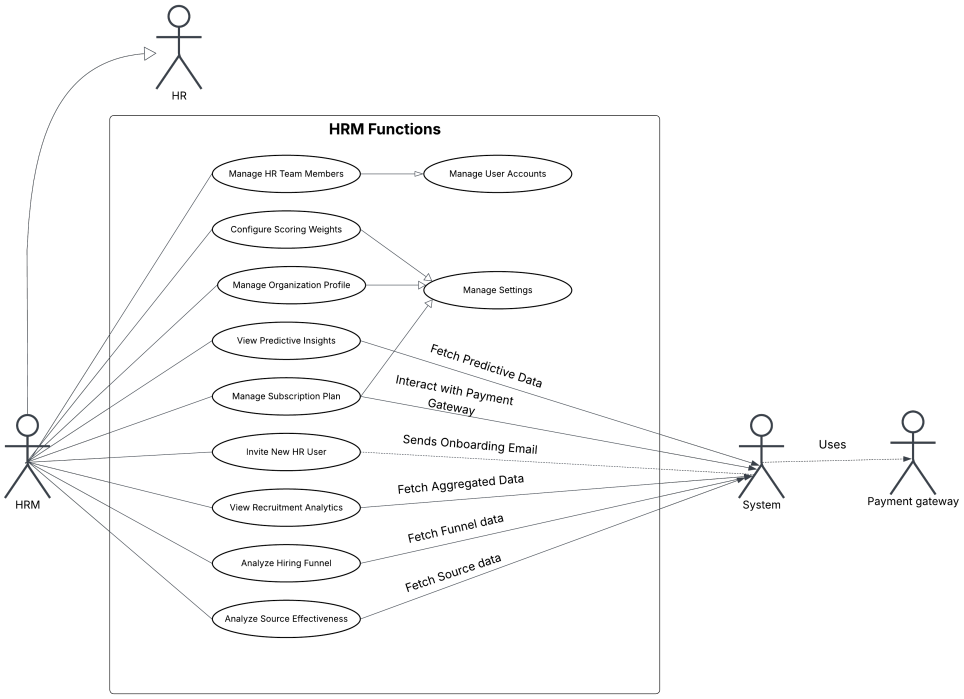

Figure 4.16: Use Case Diagram for HR Manager Functions

- **AI Integration:** Leveraging external AI services (Google Gemini) for specialized tasks like NLP and grading, rather than building complex models in-house.
- **Security Focus:** Incorporating modern authentication (Passkeys) and assessment integrity measures (Proctoring).
- **State Management:** Using Zustand for efficient global state management in the frontend.
- **Dynamic Theming:** Implementing Material You for a personalized user interface.

Chapter 5

IMPLEMENTATION

5.1 USER AUTHENTICATION AND AUTHORIZATION

- **Registration & Login:**Implemented in `routes/auth.js`. Supports email/password (using bcrypt for hashing), Google, Microsoft, and LinkedIn OAuth (using Passport.js[8] strategies configured in the `config/passport.js`). Requires email verification via codes which are sent using Nodemailer (`utils/sendVerificationCode.js`) and stored in `VerificationCodes` model.

- **Session Management:** Uses JSON Web Tokens (JWT) for stateless authentication (`utils/tokens.js, middleware/auth.js`)[9]. Access tokens have a shorter lifespan (e.g., 24h), while refresh tokens (stored in secure `HttpOnly` cookies) have a longer lifespan (e.g., 100d) for session persistence. A `/refresh-token` endpoint exists to issue new access tokens. Invalidated tokens are tracked (`models/invalidToken.js`).

- **Password Management:** Includes forgot password and reset password functionality, likely using secure, time-limited tokens sent via email (`utils/sendPasswordResetEmail.js`).

- **Passkey (WebAuthn) Authentication:** Implements `FIDO2/WebAuthn` standards for passwordless login. Uses @simplewebauthn/server library[11].

  - **Registration:** `POST /auth/passkey/register/options` generates registration challenges; POST /auth/passkey/register/verify verifies the authenticator response and saves the credential (`models/passkey.js`). Stores device info.
  - **Authentication:** POST `/auth/passkey/authenticate/options` generates authentication challenges (supports conditional mediation for auto-trigger); `POST /auth/passkey/authenticate/verify` verifies the authenticator response and logs the user in, issuing JWTs. Uses WebAuthnSession model to manage challenge state.

- **Authorization:** Middleware (`middleware/auth.js`) checks JWT validity (`authenticateJWT`) and restricts endpoint access based on user type (`authorizeUserType`, `authorizeUserTypes`), querying respective user models (Candidate, HR, HRManager) to confirm role.

5.2 CANDIDATE PROFILE AND JOB APPLICATION

- **Profile Management:** Candidates complete their profile during onboarding process (`routes/get-started.js`) or update it later (`routes/candidateRoutes.js`). Includes detailed sections for personal information, experience, education, skills, languages, certifications, projects, awards, achievements, social links, bio, resume upload (Cloudinary), and profile picture upload (Cloudinary). Uses complex validation within the Candidate model. Material You theme colors can be generated from profile pictures (`utils/colourgenerator.js`).

- **Job Discovery:** Candidates can view all jobs (`/jobs/all` endpoint in `routes/jobRoutes.js`) with filtering (search, location, type, etc.) and sorting. Applied jobs are excluded from this view for logged-in

candidates.

- **Application:** Applying to a job (`POST /jobs/:id/apply`) creates an Applicant record, linking the Candidate and Job. Initializes the applicant's `hiringProcess` based on the job's defined process, setting the first step (likely Shortlist) to 'In Progress'.

- **Application Tracking:** Candidates can view their applications (`GET /jobs/applications`) and their status progression via the Applicant model's status and `hiringProcess fields`. Sensitive data like `scores/comments` are filtered out from the `hiringProcess` view for candidates (`routes/applicantRoutes.js`). This is visualized by the component `ApplicantProgressBar.jsx`.

5.3 JOB POSTING AND MANAGEMENT (HR/HRM)

- **Job Creation:** HR/HRMs use a multi-step form (likely to be frontend `app/orgs/hrm/jobs/create/page.jsx`) to create job postings. The backend (`POST /jobs in routes/jobRoutes.js`) saves job details to the Job model, including title, description (Markdown support via react-markdown)[2], requirements (the qualifications field, likely text/Markdown), benefits, salary, location, type, level, deadline, etc. Crucially, it allows defining a custom `hiringProcess` as a JSON array of steps (type, name, description).

- **AI Job Description Analysis:** An endpoint (`/hrrequest/analyze` in `routes/hrRequestRoutes.js`) uses Gemini AI [15] to analyze a provided job description, extracting technical skills, soft skills, potential beneficial skills, minimum experience, and red flags, returning a struc-

tured JSON response. This assists HR in refining job posts or understanding requirements.

- **Job Management:** HR/HRMs can view all jobs for their organization (`GET /jobs`), view details of a specific job (`GET /jobs/:id`), update jobs (`PUT /jobs/:id`), and potentially delete or change status (e.g., to 'Closed', 'Filled' via `PUT /jobs/:id/status`).

5.4 APPLICANT TRACKING SYSTEM (ATS)

- **Applicant Viewing:** HR/HRMs can view applicants for a specific job (`GET /applicants/byjob/:jobId`). The frontend likely offers list and Kanban views (`app/orgs/hrm/jobs/[jobid]/applicants/page.jsx`).
- **Workflow Management:** The dragging and dropping of applicants between stages defined in the job's `hiringProcess` is allowed in Kanban view (`components/HiringWorkflow.jsx`).
- **Status Updates:** Moving an applicant triggers status updates (`PUT /applicants/:id/status`). Some stage transitions (Test, Interview, Offer, Reject) might involve modals for additional actions (assigning tests, scheduling interviews, adding notes) which is handled by `handleInitiateMove` in the frontend. The applicant's `hiringProcess` JSON field is updated to reflect the current stage and status.
- **Applicant Details:** HR/HRMs can view detailed applicant profiles (`GET /applicants/:id/profile`), including resume, skills, experience, and potentially scores/feedback from completed hiring process steps. A modal (`components/ApplicantDetailsModal.jsx`) displays this information.

5.5 AI-POWERED RESUME PARSING

- **Implementation:** `routes/parserRoutes.js` defines the `/parseResume` endpoint.
- **Process:**
    1. Accepts raw file buffer (PDF or DOCX) via POST request. Middleware (express.raw) handles body parsing.
    2. Extracts text using pdf-parse for PDF or mammoth for DOCX.
    3. Sends extracted text to Google Gemini (gemini-1.5-pro-latest) with a detailed prompt (`extractionPrompt`) requesting structured JSON output for Skills, Experience, Education, Certifications, Projects, Summary, Languages, and Other sections.
    4. Parses the Gemini response (handling potential markdown formatting) into a JSON object.
    5. Maps the extracted JSON data (`structuredData`) to the backend's database schema (Candidate model) using the function `convertCandidateJsonToSchema`. This involves:
        - Parsing dates (`parseFlexibleDate`).
        - Validating required fields for nested structures like projects or education.
        - Parsing the 'Other' text field to potentially extract achievements (`parseAchievements`).
        - Handling missing fields gracefully.
    6. Returns the mapped JSON (`mappedData`) conforming to the database schema.

- **Benefits:** Automates data entry, provides structured candidate data quickly, reduces manual screening time.

5.6 HIRING TESTS & ASSESSMENTS

- **Test Creation:** HR/HRMs can create custom tests (`POST /hiring-tests`) via a form (`components/HiringTestForm.jsx`) defining name, description, duration, passing score, and questions (MCQ, Text, Code). Questions are stored as JSONB in the HiringTest model. They can also select from ready-made tests (`GET /hiring-tests/ready-made`) defined in `config/tests.js`.

- **Test Assignment:** When an applicant is moved to a 'Test' stage, the frontend triggers a modal (`handleInitiateMove` calls the `setShowHiringTestModal`). Upon the submission of the form (`handleCreateHiringTest`), the chosen test (either newly created or ready-made) is assigned to the applicant(s) by updating the `hiringTestId` field in the Applicant model and sending an email invite for the test (`utils/emailTemplates/testInvitation.js, utils/emailService.js`). The applicant's status is updated.

- **Test Taking Interface:** (`app/tests/[testid]/page.jsx`):
  - Loads test metadata first, then starts the test (POST /hiring-tests/:id/start), which returns the questions (without correct answers) and records the start time in the Applicant model.
  - Displays questions one by one or via navigation (`components/QuestionDisplay.jsx`, `components/QuestionNavigation.jsx`).
  - Includes a timer (`hooks/useTestTimer.js`).

  - Supports code questions with a Monaco editor (`components/tests/MonacoCodeEditor.jsx`).
  - Includes online status checks (`hooks/useOnlineStatus.js`).

- **Code Execution:** Code submissions (`components/tests/QuestionDisplay.jsx` uses `runTestCases`) are sent to the backend (`POST /code/execute`). `utils/codeExecution/codeRunner.js` routes the code to the appropriate sandbox (`utils/codeExecution/sandboxEval.js`) based on language (Python, JS supported). Runs against predefined test cases, returns results including pass/fail status and marks.

- **Subjective Grading:** Endpoint `/validate-subjective (routes/validateSubjectiveRoutes.js)` uses Gemini AI (gemini-1.5-flash-latest) to score text-based answers against a model answer, considering correctness and coverage.

- **Test Submission:** Answers are sent to `/hiring-tests/:id/end`, which calculates the final score (summing points for correct MCQ answers, potentially adding AI-graded scores later), records the end time, warnings, and final score in the Applicant model.

- **Proctoring:** Implemented client-side (`hooks/useProctoring.js`, `components/tests/IdentityVerification.jsx`, `components/tests/LiveMonitoring.jsx`):
  - Identity Verification: Requires selfie capture and potentially liveness checks before starting the test. Uses face-api.js.
  - Monitoring: Fullscreen enforcement, tab switch detection, copy/paste/shortcut prevention, microphone checks (mute, noise/voice via

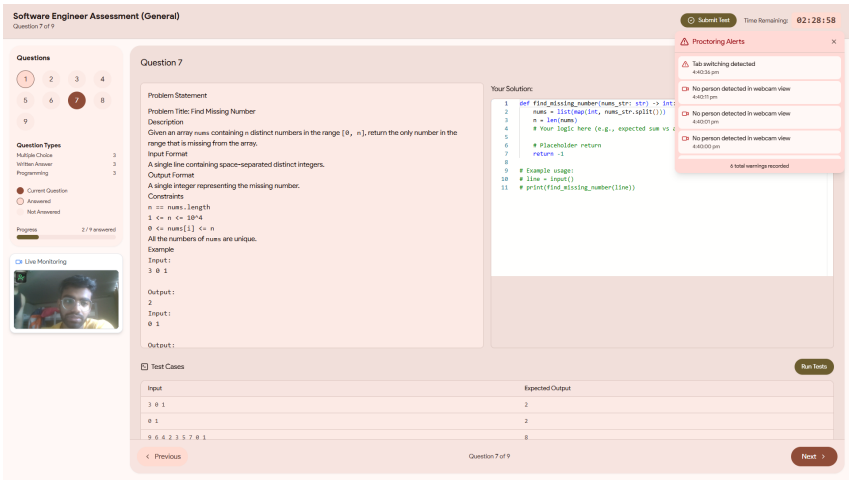

Figure 5.1: Coding UI with active proctoring and warnings

Web Audio API), webcam monitoring (face detection using `face-api.js/@tensorflow-models/face-detection`[14]-checks for no face, multiple faces).

– Warnings: Warnings are logged (`POST /hiring-tests/:id/warning`) and displayed (`components/ProctoringWarning.jsx`). Exceeding a threshold can trigger test termination (`shouldTerminateTest flag`).

- **Benefits:** Standardized and objective skill evaluation, reduced bias, flexibility of remote testing, enhanced integrity through proctoring.

5.7 INTERVIEW SCHEDULING AND FEEDBACK

- **Scheduling:** When an applicant is moved to an 'Interview' stage, a modal (`components/InterviewForm.jsx`) is triggered (`handleInitiateMove` calls `setShowInterviewModal`). HR fills in details (date, time, duration, notes, interviewers).

- **Google Calendar Integration:** The backend (`POST /interviews/schedule`

Table 5.1: Proctoring Features Summary

Feature	**Description**	**Purpose**
Fullscreen Enforcement	Requires the test to be taken in fullscreen mode.	Prevent access to other apps/tabs
Tab Switch Detection	Detects if the user navigates away from the test tab/window.	Prevent looking up answers
Copy/Paste Prevention	Disables copy/paste functionality during the test.	Prevent plagiarism/external aids
Shortcut Key Blocking	Disables common system shortcuts (e.g., Alt+Tab, Ctrl+C/V).	Prevent cheating methods
Microphone Monitoring	Detects if microphone is muted, excessive noise, or voices.	Ensure test environment integrity
Webcam Monitoring	Uses `face-api.js` for: - Identity Verification: Compares initial selfie with webcam feed. - Face Detection: Checks for no face, or multiple faces.	Ensure test taker integrity Verify correct person Ensure only candidate present
Warning System	Logs proctoring violations and displays warnings to the candidate.	Deter cheating, inform HR
Potential Termination	Test may be automatically terminated if warning threshold is exceeded.	Enforce test rules

`in routes/interviewsRoutes.js`) uses the Google Calendar API[16] (googleapis) with OAuth 2.0 to create a calendar event with Google Meet link. Requires `REFRESH_TOKEN` configured on the backend.

- **Email Invitations:** After scheduling, emails are sent via Nodemailer (`utils/emailService.js`) to both the candidate and the interviewers, containing event details and the Google Meet link.

- **Feedback Collection:** Interviewers access a feedback form (likely via a link, potentially `/orgs/feedback/[interviewId]/page.jsx`) where they can submit qualitative feedback and a quantitative score (0-100). The backend (POST `/feedback/:id`) saves this to the Interview model.

5.8 ANALYTICS AND REPORTING (HRM)

- **Backend Logic:** `routes/hrmroutes.js` contains endpoints (`/dashboard`, `/analytics`, `/analytics/predictive`, `/analytics/candidate-quality`) for fetching aggregated and calculated data. Uses Sequelize[5] with raw SQL queries and aggregations (COUNT, AVG, SUM, PERCENTILE_CONT, EXTRACT, CASE) for complex metrics.

- Key Metrics:

  - Job Performance: Applicant count per job, applications per day, shortlist conversion rate.

  - Hiring Funnel: Counts and conversion rates for each stage (Applied, Shortlisted, Interviewed, Offered, Hired). Includes trend comparison (current vs. previous period).

  - Source Effectiveness: Applicant count, hired count, and conversion rate per candidate source (LinkedIn, Indeed, etc.).

  - Time-to-Hire: Average days from application to hire, broken down by job type/level.

  - Test Completion Rates: Tracks completion status (Completed/Pending) for assigned tests, with trend analysis.

  - Monthly Trends: Tracks applicant volume and hires month-over-month.

  - Predictive: Time-to-fill predictions, seasonal hiring patterns, hiring forecast, pipeline bottleneck identification.

  - Candidate Quality: Test score distribution/averages by job, source quality index (combining scores and conversion), top-performing

skills analysis, interviewer effectiveness analysis (avg score, pass rate).

- **Frontend Display:** (`app/orgs/hrm/analytics/page.jsx`): Uses Recharts library to display data visually through Bar charts, Line charts, Area charts, and Pie charts. Presents key metrics in cards. Offers tabs for different analytic views (Overview, Predictive, Quality).
- **Benefits:** Enables data-driven recruitment strategy, identifies bottlenecks, measures source/interviewer effectiveness, helps forecast hiring needs.

Table 5.2: AI Features Overview

Feature	**AI Model Used**	**Purpose**	**Interacting Role(s)**
Resume Parsing	`gemini-1.5-pro`	Extract structured data (Skills, Experience, Edu.) from resumes	System / HR (viewing)
Job Desc. Analysis	`gemini-1.5-pro`	Extract Skills, Experience, Red Flags from job descriptions	HR/HRM (job creation)
Subjective Grading	`gemini-1.5-flash`	Score text-based answers in assessments against a model answer	System / HR (viewing)
Personality Traits	`gemini-2.5-pro`	Analyze text (potentially bio/summary) for personality traits	System / HR (potentially viewing)
AI Matching/Scoring	`gemini-1.5-flash`	Score candidate suitability for a job	Candidate / HR
*Note: Specific model names based on current version of Aptinova. May change.			

Chapter 6

HARDWARE/ SOFTWARE TOOLS USED

6.1 SOFTWARE TOOLS & TECHNOLOGIES

- **Backend:**
  - Runtime: Node.js (v18)[3]
  - Framework: Express.js[4]
  - Database: PostgreSQL
  - ORM: Sequelize[5]
  - Authentication: Passport.js, JSON Web Token (JWT), bcrypt, @simplewebauthn/server
  - API Client: Axios
  - Email: Nodemailer
  - File Storage: Cloudinary
  - Payments: Razorpay SDK
  - AI: Google Gemini API (@google/generative-ai)[15]
  - Calendar: Google Calendar API (googleapis) [16]
  - Code Execution: Python (python-shell), Node.js vm module
  - Deployment: Vercel
- **Frontend:**
  - Framework: Next.js (v15+)[1]

  - Language: JavaScript (React JSX), TypeScript
  - UI Library: React (v19+)[2]
  - Styling: Tailwind CSS[6], PostCSS
  - State Management: Zustand[7]
  - Animation: Framer Motion
  - Charting: Recharts
  - Rich Text/Markdown: @uiw/react-md-editor, react-markdown
  - Code Editor: Monaco Editor (@monaco-editor/react)
  - Drag & Drop: @hello-pangea/dnd
  - WebAuthn Client: @simplewebauthn/browser
  - Proctoring/Face Detection: face-api.js[13], @tensorflow/tfjs, @tensorflow-models/face-detection, @tensorflow-models/blazeface
  - Particles: @tsparticles/react, @tsparticles/slim

- **Development & Tools:**
  - Version Control: Git
  - Package Managers: npm
  - Code Editor: VS Code
  - Process Manager: Nodemon (for development)
  - Linting/Formatting: ESLint, Prettier
  - Database GUI: (e.g., pgAdmin, DBeaver)

Table 6.1: Technology Stack Summary

Tier / Category	**Technology / Service**	**Purpose**
Frontend	React (Next.js)[2]	UI Framework, SSR, Routing
	Tailwind CSS[6]	Utility-First CSS Styling
	Material You (via CSS vars)[19]	Dynamic Theming
	Zustand[7]	Client-side State Management
	Axios	HTTP Client (API Communication)
	face-api.js / TensorFlow.js[14]	Client-side Face Detection (Proctoring)
	Monaco Editor	Code Editor for Assessments
	Recharts	Data Visualization (Analytics Charts)
	Framer Motion	UI Animations
Backend	Node.js[3]	Server-side JavaScript Runtime
	Express.js[4]	Web Application Framework (API Handling)
	Sequelize[5]	ORM (Object-Relational Mapper) for Database Interaction
	Passport.js[8]	Authentication Strategies (Local, OAuth)
	JWT (jsonwebtoken)[9]	Session Management (Access/Refresh Tokens)
	bcrypt[10]	Password Hashing
	WebAuthn (`@simplewebauthn`)	Passkey[11] (Passwordless) Authentication
	Nodemailer	Email Sending (Notifications, Invites)

Continued on next page

Table 6.1 – continued from previous page

Tier / Category	Technology / Service	Purpose
	`python-shell`	Running Python scripts (Potentially for code execution)
Database	PostgreSQL	Relational Database Management System
Ext. Services	Google Gemini AI[15]	Resume Parsing, Job Analysis, Subjective Grading, Traits
	Google Calendar API	Interview Scheduling
	Google OAuth	User Authentication
	Microsoft OAuth	User Authentication
	LinkedIn OAuth	User Authentication
	Cloudinary[17]	Cloud-based Image/File Storage (Resumes, Logos)
	Razorpay[18]	Payment Gateway (Subscription Handling)
Deployment	Vercel	Hosting Platform (Frontend & Backend Serverless Functions)

6.2 HARDWARE

- **Development:** Standard developer laptops/desktops (Mac, Windows, Linux).
- **Deployment:** Vercel's cloud infrastructure (servers, CDN). PostgreSQL database server (cloud-hosted or self-hosted).
- **User Requirements:** Web browser (Chrome, Firefox, Safari, Edge rec-

ommended), Webcam and Microphone (for proctored tests and potentially video interviews).

Chapter 7

RESULTS & DISCUSSION

7.1 FUNCTIONALITY ACHIEVED

The Aptinova platform successfully implements a wide range of features constituting a modern, AI-enhanced recruitment system:

- Robust user authentication supporting multiple methods including passwordless passkeys.
- Separate, role-based interfaces and dashboards for Candidates, HR, and HR Managers.
- Comprehensive job posting and management capabilities for HR/HRM.
- Efficient candidate application tracking and workflow management via a Kanban board and status updates.
- AI-powered automation for resume parsing and job description analysis, significantly reducing manual effort.
- Integrated assessment module allowing creation and assignment of custom or ready-made tests (MCQ, Text, Code).
- Secure online test-taking environment with multi-faceted proctoring features (fullscreen, tab switching, basic camera/mic monitoring).
- Automated code grading and AI-assisted subjective answer grading.

- Seamless interview scheduling via Google Calendar integration and automated email invitations.
- Advanced analytics dashboard for HRMs providing insights into job performance, funnel metrics, source effectiveness, time-to-hire, and predictive trends.
- Subscription management system for different user tiers.
- Team management features for HRMs.
- Personalized UI through Material You dynamic theming.

7.2 DISCUSSION OF KEY FEATURES AND BENEFITS

- **AI Integration:** The use of Google Gemini[15] for parsing, analysis, and grading is a key differentiator.
  Benefit: Reduces manual workload, improves consistency, provides deeper insights (e.g., skills, traits), and potentially reduces bias compared to purely manual review.
- **Integrated Workflow:** Combining job posting, ATS, assessments, and interviews into one platform streamlines the entire process.
  Benefit: Improves efficiency for HR, reduces context switching, provides a cohesive experience for candidates.
- **Proctoring:** The built-in proctoring features enhance the credibility of remote assessments.
  Benefit: Increases confidence in assessment results, deters cheating, maintains fairness.

Tanishk Yadav

yadav_tanishk@srmap.edu.in | tanishkyadav.me | +91-9999095439

EDUCATION

BTech, Computer Science & Engineering, SRM University-AP (100% scholarship) **Sep 2021 – Present**
CGPA: 8.27
Senior Secondary Education, PCM, DAV Public School, Sector-49, Gurgaon **Apr 2019 – Jun 2021**
Score: 93%
Secondary Education, DAV Public School, Sector-49, Gurgaon **Apr 2017 – May 2019**
Score: 92.4%

PROFESSIONAL EXPERIENCE

Founder and Content Creator, ScienceWaale **Apr 2021 – Present**
- Pioneered and executed dynamic video content, delivering expert instruction in Quantitative Aptitude, Logical Reasoning, and educational content for exams, resulting in organic subscribers and viewership through 20+ videos.
- Collaborated with a team of educators to research and curate compelling videos, amassing over 1800+ subscribers and 200,000+ gross views.

Online Artificial Intelligence & Machine Learning Associate Tutor, ULearn - Edu **Sep 2024 – Jan 2025**
- Teaching and mentoring a batch of 50 undergraduate students from different disciplines in the field of AI & ML through virtual classes for 120 hours.
- Providing personalized guidance and support for students' understanding of AI/ML concepts, fostering academic growth and practical skill development, and helping them with 500+ coding problems and their research projects.

Intern, India Urban Data Exchange (IUDX) **Jun 2023 – Aug 2023**
- Applied expertise in cryptography, web development, and data visualization to engineer a portal for managing urban data, ensuring the creation of secure and legally compliant digitally signed artifacts.
- It is currently utilized by Public Sector Companies across 50 cities and 15+ industries.

LEADERSHIP EXPERIENCE

Head of Hospitality Committee and Co-Head of Sponsorship Team, ALOHA'23 **Aug 2023 – Oct 2023**
- Organizing the Annual Freshers' Event with a budget of Rs. 2M, including guest management, hospitality, and collaborating with local businesses for sponsorship and promotional offers for the 8000+ attendees from the students.

Public Relations Wing Convener, Student Council **May 2023 – Oct 2023**
- Organized diverse PR initiatives, addressed student concerns related to discipline, food serviced in mess, etc., and championing promotional efforts for events attended by 8000+ students.

Head of Registrations Committee, INFINITUS'23 (SRM University – AP) **Feb 2023 – Apr 2023**
- Managing the national-level fest budgeted at Rs. 8.3M and with 5000+ attendees. Collaborating with the university's financial body to optimize fees, create indents, and maintain a healthy trade-off between expenses and experience.

Public Relations Wing Member, Student Council **May 2022 – May 2023**
- Fostered strong inter-wing relationships within the Student Council to enhance and optimize event outcomes. Led and managed three event committees, maximizing individual productivity to successfully execute over 30 events.

RESEARCH & PROJECTS

Semantic Communication: Enhancing Spectrum Efficiency for Multimedia Applications–A review
- A research on the booming field of Semantic Communication with proposed architecture and its possible implications in various sectors such as healthcare, education, industries 4.0, reduce redundancy and network traffic.

Analysis of Sector-Wise S&P500 firms and sector-wise clustering for cohesive movement
- Classified publicly perceived monopolistic and oligopolistic players across S&P500 companies in their respective sectors and used multiple clustering parameters to address correlation between correlated sectors and its clusters.

Stock Price Prediction using LSTM and Technical Indicators
- Developed an advanced stock price forecasting system using Random Forest and LSTM models, integrating technical indicators (RSI, MACD, Bollinger Bands) to enhance predictive accuracy and trading insights.

Forex Trading Algorithm
- Developed an algorithm for real-time forex trading that utilizes ML models to predict exchange rates and identify arbitrage opportunities using current currency conversion rates with a depth of 5 currency trades.

Super Market Accounting System
- A full-fledged online portal for a supermarket accounting for goods and can work for 3 entities (Store Manager, Cashier, Accountant), it also displays net profits, sales, etc. with a modern UI and optimizing Database API calls.

SKILLS & LANGUAGES

C, C++, Java, Python, SQL, MATLAB, HTML, CSS, JavaScript, R, Prolog, Data Analysis, Web Development, Algorithm Implementation, Cryptography, Machine Learning (LSTM, XGBoost, RFC, etc.)

Figure 7.1: Sample Resume

```
{
  "Skills": ["C","C++","Java","Python","SQL","NoSQL","HTML","CSS","JavaScript","R","Prolog","Data Analysis","Web Development",
  "Algorithm Implementation","Cryptography","Machine Learning","AI","LSTM","XGBoost","Random Forest","Data Visualization",
  "Digital Signatures","Quantitative Aptitude","Logical Reasoning","Stock Prediction","Forex Trading","Technical Indicators
  (SMA, EMA, Bollinger Bands)","Clustering","Database Optimization","API","Real-time Data Processing","Mentoring","Content
  Creation","Public Relations","Event Management","Sponsorship Management","Budget Management","Team Collaboration"
  ],
  "Experience": [
    {
      "Company": "2LEARN EDU", "Position": "Online Artificial Intelligence & Machine Learning Associate Tutor","StartDate":
      "Sep 2023","EndDate": "Present","DurationMonths": null,"Location": null,"Description": "TEACHING AND MENTORING A BATCH
      OF 50+ UNDERGRADUATE STUDENTS FROM DIFFERENT DISCIPLINES IN THE FIELD OF AI/ML THROUGH VIRTUAL CLASSES FOR 6-8 HOURS.
      \nPROVIDING PERSONALIZED GUIDANCE AND SUPPORT FOR STUDENTS' UNDERSTANDING OF AI/ML CONCEPTS, FOSTERING ACADEMIC GROWTH
      AND PRACTICAL SKILL DEVELOPMENT, AND HELPING THEM WITH 100+ CODING PROBLEMS AND THEIR RESEARCH PROJECTS."
    },
    {
      "Company": "SCIENCE WAALE","Position": "Founder and Content Creator","StartDate": "Sep 2021","EndDate": "Present",
      "DurationMonths": null,"Location": null,"Description": "PIONEERED AND EXECUTED DYNAMIC VIDEO CONTENT, DELIVERING
      EXPERT INSTRUCTION IN QUANTITATIVE APTITUDE, LOGICAL REASONING, AND EDUCATIONAL CONTENT FOR EXAMS, RESULTING IN
      ORGANIC SUBSCRIBERS AND VIEWERSHIP THROUGH 80+ VIDEOS.\nCOLLABORATED WITH A TEAM OF EDUCATORS TO RESEARCH AND CURATE
      COMPELLING VIDEOS, AMASSING OVER 6200+ SUBSCRIBERS AND 800,000+ GROSS VIEWS."
    },
    {
      "Company": "INDIAN URBAN DATA EXCHANGE (IUDX)", "Position": "Intern", "StartDate": "Jun 2023", "EndDate": "Aug 2023",
      "DurationMonths": 3, "Location": null, "Description": "APPLIED EXPERTISE IN CRYPTOGRAPHY, WEB DEVELOPMENT, AND DATA
      VISUALIZATION TO ENGINEER A PORTAL FOR MANAGING URBAN DATA, ENSURING THE CREATION OF SECURE AND LEGALLY COMPLIANT
      DIGITALLY SIGNED ARTIFACTS.\nIT IS CURRENTLY UTILIZED BY PUBLIC SECTOR COMPANIES ACROSS 10+ CITIES AND 60+ INDUSTRIES."
    },
    {
      "Company": "GEEKSFORGEEKS NETWORK", "Position": "Campus Ambassador", "StartDate": "Nov 2021", "EndDate": "Apr 2023",
      "DurationMonths": 18, "Location": null, "Description": "CRAFTED COMPELLING CONTENT, PROPOSED STRATEGIC INITIATIVES TO
      CAPTIVATE NEW USERS, AND ORCHESTRATED SEAMLESS COLLABORATION WITH TEAM MEMBERS AND AUTHORITIES, RESULTING IN A
      REMARKABLE 15% GROWTH WITHIN JUST 8 MONTHS.\nRECOGNIZED AS THE BEST PERFORMER DURING THE INAUGURAL MONTH IN THE ENTIRE
      ZONE AMONGST 10+ AMBASSADORS."
    },
    {
      "Company": "AARAMBH - SRM University AP", "Position": "Head of Hospitality Committee and Co-Head of Sponsorship Team",
      "StartDate": "Aug 2023", "EndDate": "Oct 2023", "DurationMonths": 3, "Location": null, "Description": "ORGANIZING THE
      ANNUAL FRESHERS' EVENT WITH A BUDGET OF RS. 8L, INCLUDING GUEST MANAGEMENT, HOSPITALITY, AND COLLABORATING WITH LOCAL
      BUSINESSES FOR SPONSORSHIP AND PROMOTIONAL OFFERS FOR THE 2000+ ATTENDEES FROM THE STUDENTS."
    },
```


Figure 7.2: Partial snapshot of generated JSON

- **Passkey Authentication:** Implementing modern, phishing-resistant authentication.

  Benefit: Enhances security for all users, improves login experience (passwordless option).

- **Analytics:** The comprehensive analytics suite provides actionable insights for HRMs.

  Benefit: Enables data-driven decision-making, optimization of recruitment strategies, identification of bottlenecks and high-performing sources/interviewers as in Fig. 2.2.

- **Material You Theming:** Offers a modern, personalized UI.

  Benefit: Improves user engagement and satisfaction, aligns with modern design trends.

7.3 WEIGHTED AVERAGE SCORING (PLANNED FEATURE)

- **Concept:** The final score for an applicant on a specific job application will be calculated as a weighted average of scores obtained from different stages of the hiringProcess defined for that job.
- **Components:** Potential score components include:
  - AI Resume Match Score (if implemented based on initial job description analysis)
  - Assessment/Test Scores (from HiringTest results via Applicant.score or linked test steps)
  - Interview Scores (from Interview.score linked via applicant/job)
  - Subjective Answer Scores (from AI grading)
- **Weighting Configuration:** HR Managers will have the capability (likely within the Job creation/editing interface or a dedicated settings area) to assign weights (percentages) to each relevant stage or score component (e.g., Test Score: 40%, Technical Interview: 30%, HR Interview: 20%, Resume Match: 10%). These weights will be specific to each job posting, allowing customization based on role requirements.
- **Calculation:** The backend (likely within applicantRoutes.js or a dedicated scoring service) will calculate the final weighted score when requested or possibly upon completion of all scored stages.
- **Display:** The final weighted score will be displayed prominently on the applicant's profile card within the ATS/Kanban view and potentially in the applicant details modal for HR/HRM users.

- **Benefit:** Provides a single, customizable metric for comparing candidates based on criteria deemed most important for the specific role, aiding objective decision-making.
- **Implementation Note:** Requires adding a mechanism for HRMs to set weights per job and backend logic to calculate and store/display the final weighted score.

7.4 DISCUSSION OF RESULTS

The implemented features demonstrate a robust and feature-rich platform. The successful integration of various modules (ATS, AI, Assessments, Interviews, Analytics) showcases the potential for significant improvements in recruitment efficiency and effectiveness. The AI components, particularly resume parsing and job analysis, promise substantial time savings for HR. The integrated assessment and proctoring system provide a credible way to evaluate candidates remotely. The comprehensive analytics offer valuable strategic insights previously unavailable in many fragmented systems. The planned weighted scoring adds another layer of objective evaluation.

Chapter 8

CONCLUSION

8.1 SUMMARY OF WORK

The Aptinova project successfully developed a full-stack, AI-enhanced recruitment platform addressing key challenges in modern hiring. It provides distinct interfaces and tailored functionalities for Candidates, HR Professionals, and HR Managers. Key achievements include the implementation of AI-driven resume parsing and job analysis, an integrated assessment module with coding challenges and proctoring, Google Calendar integration for interview scheduling, a comprehensive ATS with Kanban workflow, advanced analytics for HRMs, and support for modern security standards like passkeys. The platform architecture leverages Node.js, Express, PostgreSQL, Sequelize, React, Next.js, and integrates several external APIs, including Google Gemini and Cloudinary.

8.2 CONTRIBUTIONS

Aptinova's main contributions lie in: **Unified Platform:** Integrating job board, ATS, assessment, interview scheduling, and analytics into a single system.

- **AI-First Approach:** Utilizing AI extensively for automation (parsing), analysis (job descriptions, traits), and evaluation (subjective grading, scoring assistance).

- **Enhanced Security:** Implementing passkey authentication and built-in proctoring measures.
- **Modern User Experience:** Employing Material You dynamic theming and a responsive design. **Data-Driven Recruitment:** Providing powerful analytics and predictive insights for strategic HR decision-making.

8.3 LIMITATIONS

- **AI Model Reliance:** Performance is dependent on the accuracy and limitations of the underlying Gemini AI models. Bias in AI models is a potential concern requiring ongoing monitoring.
- **Proctoring Scope:** Current proctoring is primarily automated and may not be as foolproof as live human proctoring for high-stakes assessments.
- **Scalability Testing:** While designed with scalability in mind, real-world performance under heavy load has not been tested.
- **Feature Completeness:** Some features (like the final weighted average score calculation) are planned but not yet fully implemented in the analyzed codebase. AI matching score mentioned in introduction needs clarification on implementation status.

8.4 FUTURE WORK

- **Implement Weighted Average Scoring:** Complete the backend logic and frontend UI for configuring and displaying weighted average scores.

- **AI Candidate Matching:** Develop and integrate an AI model to proactively match candidates to suitable jobs based on their profiles and job requirements.
- **Mobile Application:** Develop native mobile apps (iOS/Android) for an enhanced mobile experience.
- **Deeper Integrations:** Integrate with more HRIS systems and background check providers.
- **Advanced Proctoring:** Explore more sophisticated AI proctoring techniques (e.g., gaze tracking, object detection) or integration with third-party proctoring services.
- **AI Bias Auditing:** Implement regular audits and mitigation strategies for potential bias in AI components.
- **Gamification:** Introduce gamified elements for candidate assessments or profile completion.
- **Enhanced Collaboration Tools:** Add more features for internal HR team communication and feedback sharing.

REFERENCES

[1] Next.js Documentation. Vercel. Accessed April 27, 2025. https://nextjs.org/docs

[2] React Documentation. Meta Platforms, Inc. Accessed April 27, 2025. https://react.dev/

[3] Node.js Documentation. OpenJS Foundation. Accessed April 27, 2025. https://nodejs.org/en/docs/

[4] Express.js - Fast, unopinionated, minimalist web framework for Node.js. OpenJS Foundation and contributors. Accessed April 27, 2025. https://expressjs.com/

[5] Sequelize ORM Documentation. Sequelize Contributors. Accessed April 27, 2025. https://sequelize.org/

[6] Tailwind CSS Documentation. Tailwind Labs Inc. Accessed April 27, 2025. https://tailwindcss.com/docs

[7] Zustand - State Management. Poimandres. Accessed April 27, 2025. https://github.com/pmndrs/zustand

[8] Passport.js Documentation. Jared Hanson. Accessed April 27, 2025. https://www.passportjs.org/

[9] JSON Web Tokens. Auth0. Accessed April 27, 2025. https://jwt.io/

[10] bcrypt - npm. npmjs.com. Accessed April 27, 2025. https://www.npmjs.com/package/bcrypt

[11] Web Authentication: An API for accessing Public Key Credentials Level 3. W3C Recommendation. World Wide Web Consortium (W3C). Accessed April 27, 2025. https://www.w3.org/TR/webauthn-3/

[12] simple-webauthn Documentation. Matthew Miller. Accessed April 27, 2025. https://simplewebauthn.dev/

[13] face-api.js - JavaScript API for Face Recognition. Vincent Mühler. Accessed April 27, 2025. https://github.com/justadudewhohacks/face-api.js

[14] TensorFlow.js Documentation. Google. Accessed April 27, 2025. https://www.tensorflow.org/js/

[15] Google AI Gemini Documentation. Google Cloud. Accessed April 27, 2025. https://cloud.google.com/gemini/docs

[16] Google Calendar API Documentation. Google for Developers. Accessed April 27, 2025. https://developers.google.com/calendar/api/guides/overview

[17] Cloudinary Documentation. Cloudinary. Accessed April 27, 2025. https://cloudinary.com/documentation

[18] Razorpay Payment Gateway Documentation. Razorpay. Accessed April 27, 2025. https://razorpay.com/docs/payment-gateway/

[19] Material Design 3 Guidelines. Google. Accessed April 27, 2025. https://m3.material.io/

[20] Fielding, R. T. (2000). Architectural Styles and the Design of Network-based Software Architectures. Doctoral dissertation, University of Cali-

fornia, Irvine. https://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm

[21] What is an Applicant Tracking System (ATS)? SAP Insights. Accessed April 27, 2025. https://www.sap.com/india/products/hcm/recruiting-software/what-is-an-applicant-tracking-system.html

[22] How AI is Transforming the Recruitment Process. Gi Group India. Accessed April 27, 2025. https://in.gigroup.com/how-ai-is-transforming-the-recruitment-process/