Ecole Polytechnique De Sousse

Spécialité

Ingénieur en Génie Logiciel

Réalisé par

Bedoui Moncef Aziz

GreenGate Web App

Soutenu le 09/01/2026 devant le Jury composé de :

Mr Gueddes Abdelwaheb Professeur

Projet de Mern APP fait à

(5ème Année GL1)

Table des matières

Introduction **1**

1 The Backend Engine : Node.js, Express, and AI Integration **3**
1.1 Project Directory Structure . . . 3
1.2 System Architecture : The MVC Pattern . . . 3
1.3 Security and Session Management . . . 4
1.4 Analytics Engine and BI Data Modeling . . . 4
1.4.1 Tracking and Event Logging . . . 4
1.4.2 BI Data Structure . . . 4
1.5 AI Controller and Intelligence Logic . . . 5
1.5.1 Testing and Validation . . . 5

2 The User Experience : Discovery and Engagement **7**
2.1 Frontend Design with React 19 and Vite . . . 7
2.2 The Knowledge Hub : Blogs and AI Insights . . . 7
2.3 The Opportunities Page : External Career Leads . . . 8
2.4 The Fellowship Portal : Internal Application Workflow . . . 8

3 The Admin Dashboard : Management and Decision Making **10**
3.1 Administrative Interface Overview . . . 10
3.2 Content Management and SEO Optimization . . . 10
3.3 Fellowship Adjudication Pipeline . . . 11
3.4 Business Intelligence (BI) and Data Visualization . . . 11
3.5 Security and Role-Based Access . . . 12

Conclusion **13**

Introduction

Project Overview

In an era defined by environmental transition, the gap between ecological awareness and professional implementation remains a significant barrier. **GreenGate** is designed as a specialized ecosystem to bridge this gap, acting as a centralized hub where sustainability content meets career advancement. The platform serves a dual purpose : it educates users through a high-quality blog repository and empowers them through a curated directory of external opportunities and exclusive internal fellowship programs.

Problem Statement

Aspiring environmental professionals often struggle with information fragmentation. Relevant insights are scattered across various media, and career opportunities—specifically fellowships—lack a centralized management system that allows for seamless application and real-time status tracking. GreenGate addresses these challenges by consolidating these resources into a single, cohesive digital environment.

Technological Foundation

To ensure scalability, security, and a modern user experience, GreenGate is built upon the **MERN Stack**, a robust collection of JavaScript-based technologies :

- **MongoDB :** A NoSQL database used for flexible data storage of blogs and fellowship applications.
- **Express.js & Node.js :** A high-performance backend architecture capable of handling complex business logic and secure API routing.
- **React.js :** Used for building two distinct, highly responsive frontends (User and Admin) that provide a seamless Single Page Application (SPA) experience.

Intelligence Integration

The platform distinguishes itself through the integration of the **Google Gemini 2.5 SDK**. Unlike traditional search features, GreenGate utilizes a generative AI agent capable of analyzing "featured" content. This allows users to engage in a contextual dialogue with the platform's database, receiving summarized insights and personalized recommendations derived directly from the hosted sustainability blogs.

Objectives and Scope

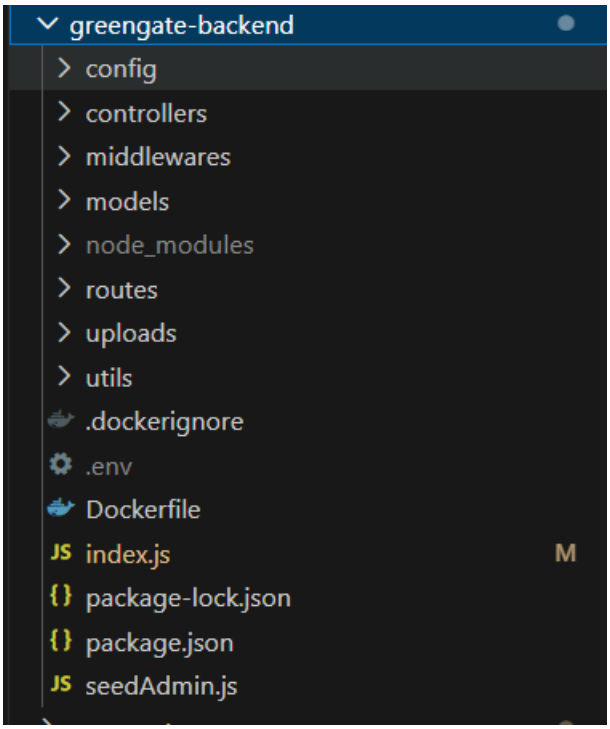
This report details the technical implementation of GreenGate, focusing on the development of the administrative decision-making pipeline, the user engagement interface, and the AI-driven backend controllers that power the ecosystem's intelligence.

Chapitre 1

The Backend Engine : Node.js, Express, and AI Integration

1.1 Project Directory Structure

The backend of GreenGate is built with a modular approach to ensure scalability and maintainability. By separating the codebase into specialized folders, we ensure that the logic for data schemas, API routes, and business rules remains organized. This structure allows multiple developers to work on different features (such as AI or Analytics) without code conflicts.



1.2 System Architecture : The MVC Pattern

To facilitate seamless communication between the MongoDB database and the two frontend applications (User and Admin), the backend utilizes a **RESTful API** architecture based on the **Model-View-Controller (MVC)** pattern :

— **Routes :** These act as the entry points for all incoming HTTP requests. They define the API endpoints (e.g., `POST /api/ai/chat`) and map them to specific functions.
— **Controllers :** This layer contains the core business logic. It processes the incoming data, communicates with the AI SDK or the database, and prepares the final JSON response for the frontend.
— **Models :** Built with **Mongoose 9.0.2**, these define the schemas for our data. They ensure that every Blog, Opportunity, and Fellowship application follows a strict structure before being saved to MongoDB.

1.3 Security and Session Management

Security is a foundational requirement for GreenGate. The backend implements two primary security measures :

— **Bcryptjs :** A library used to hash user passwords before storage, ensuring that even in the event of a database leak, user credentials remains protected.
— **JWT (JSON Web Tokens) :** Used to manage secure sessions. Once a user or admin logs in, they receive a signed token which is then used to authorize subsequent requests to protected routes.

1.4 Analytics Engine and BI Data Modeling

A unique feature of the GreenGate backend is its role as a data collection engine for **Business Intelligence (BI)**. We have implemented a specialized tracking logic to capture engagement metrics that are later visualized in Power BI.

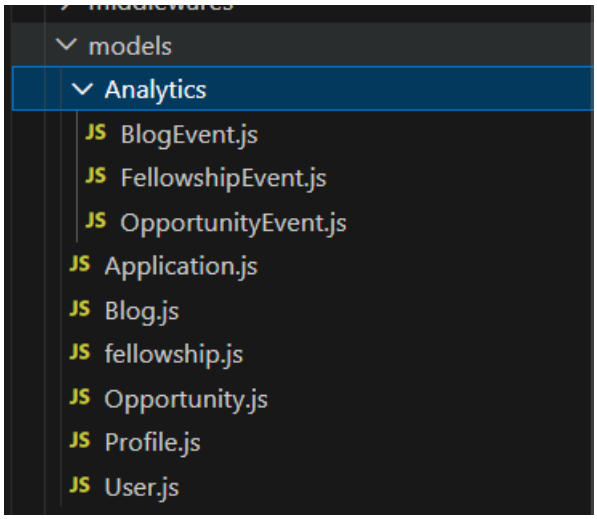
1.4.1 Tracking and Event Logging

Within our controllers, we have integrated "event triggers" that update analytics fields in real-time :

— **Content Engagement :** We track *Views* for both Blogs and Opportunities to determine which topics are most popular.
— **Conversion Metrics :** We log every *Application* for Fellowships. This allows us to calculate the ratio between the number of people who saw an opportunity and those who actually applied.

1.4.2 BI Data Structure

This data is structured using timestamps and relational IDs, making it "Power BI ready." This allows the administration to monitor growth trends, such as the monthly increase in fellowship demand or the success rate of various sustainable categories.



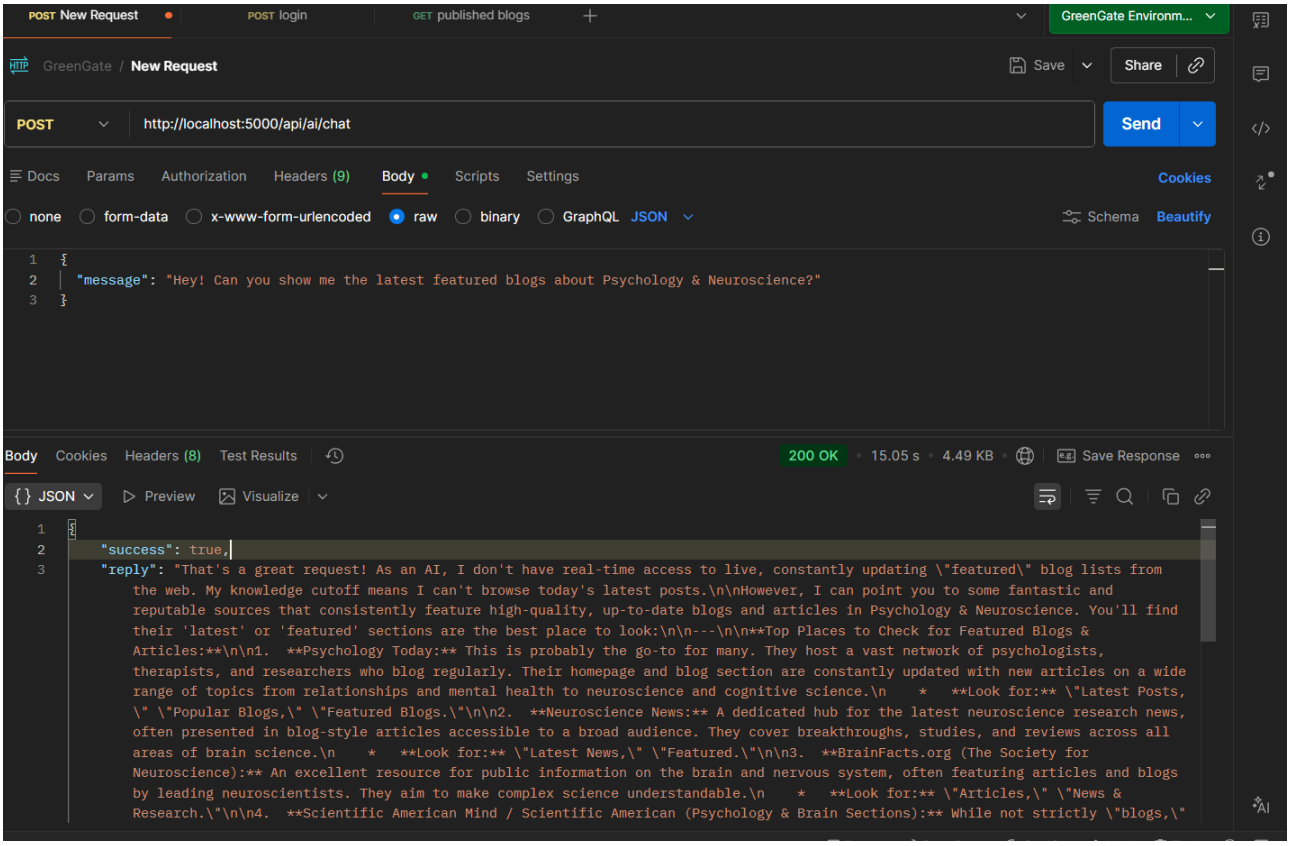
1.5 AI Controller and Intelligence Logic

The "intelligence" of the platform is powered by the `@google/genai` SDK. We developed an AI Controller that functions as a **Database Agent**. Unlike standard chatbots that rely on generic knowledge, this agent is fed real-time data from our **Blogs** collection.
When a user triggers the "Featured Insights" function, the controller :

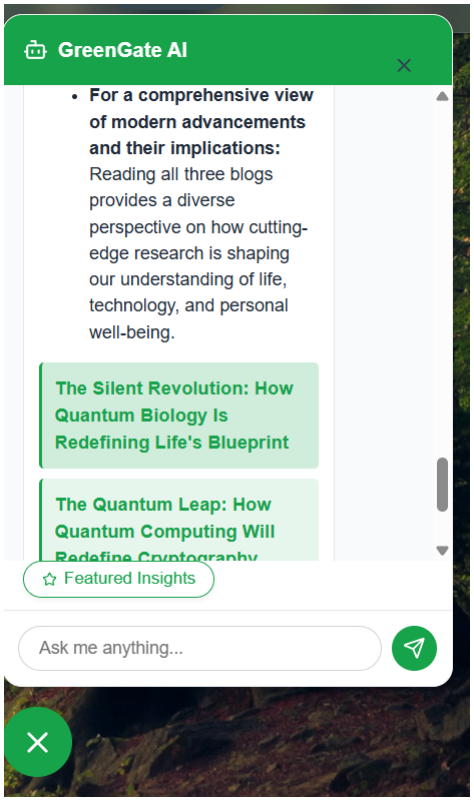
1. Queries the database for all blogs marked as "featured."
2. Aggregates their titles, categories, and excerpts into a structured prompt.
3. Sends this context to the Gemini 2.5 model to generate a report including key takeaways and reading recommendations.

1.5.1 Testing and Validation

The AI logic was first validated using **Postman** to ensure the JSON responses were accurate and that the AI didn't "hallucinate" information outside our database.



Following backend verification, the feature was integrated into the frontend chat interface, providing a formatted, easy-to-read report directly to the user.



Chapitre 2

The User Experience : Discovery and Engagement

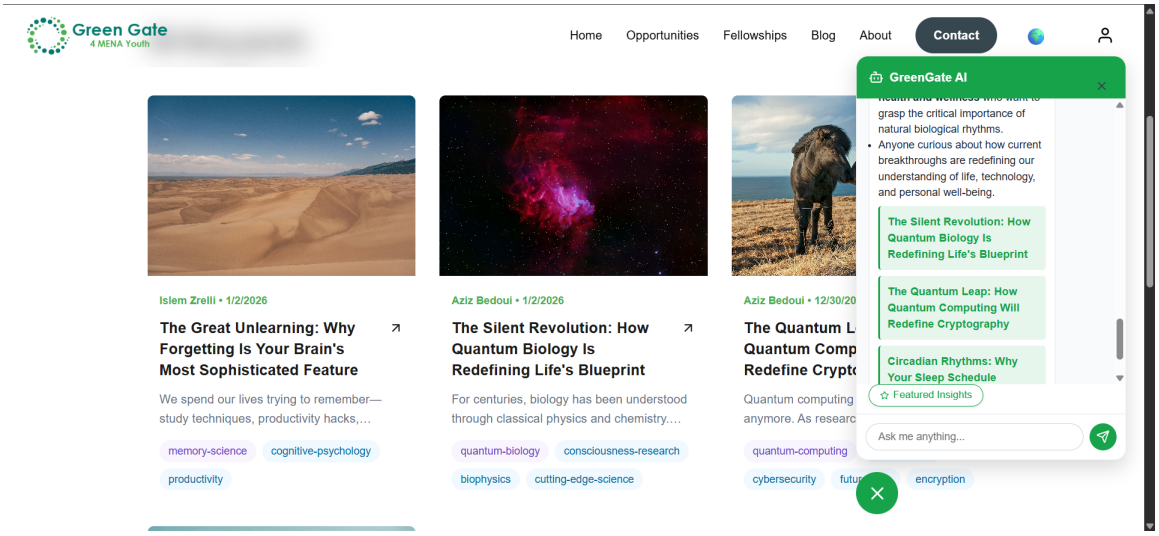
2.1 Frontend Design with React 19 and Vite

The user-facing application, `greengate-user`, is built using the latest **React 19** framework. To ensure a fast and modern development workflow, we utilized **Vite** as the build tool, providing near-instant Hot Module Replacement (HMR). The visual identity is established through **Material UI (MUI)**, which offers a clean, professional aesthetic, while **Framer Motion** is employed to handle sophisticated page transitions and interactive animations.

2.2 The Knowledge Hub : Blogs and AI Insights

The Blog module is the primary educational pillar of GreenGate. It is designed to transform passive reading into an interactive experience.

- **Dynamic Rendering :** Articles are fetched from the backend and rendered using `react-markdown`, allowing for rich text formatting including headers, lists, and code blocks.
- **AI Chat Assistant :** A floating chat bubble is integrated globally. It allows users to instantly generate summaries of "featured" blogs. This component uses a specialized React hook to communicate with the Gemini AI controller, displaying the results in a real-time conversational interface.

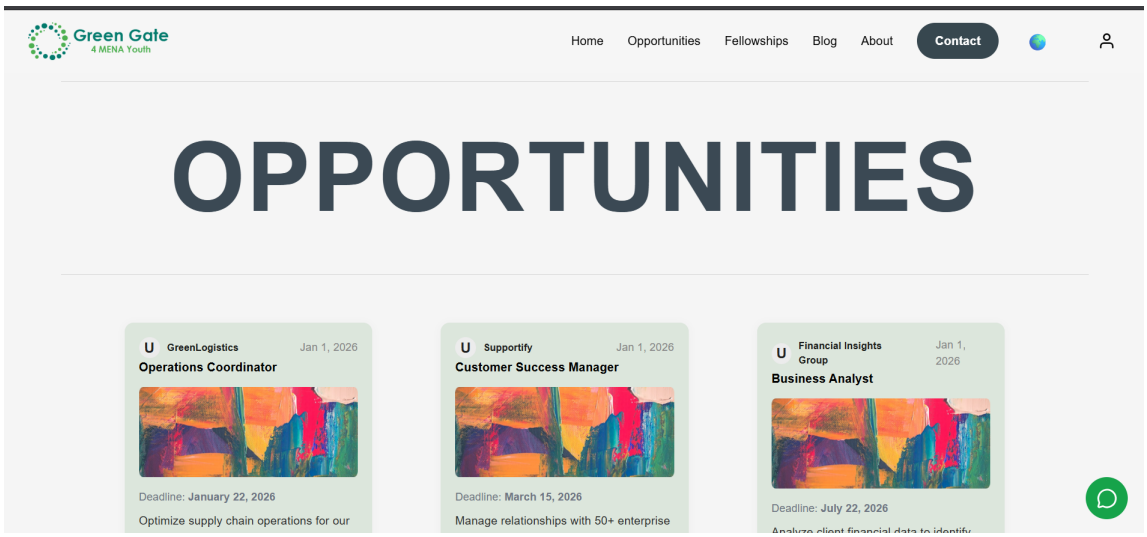


2.3 The Opportunities Page : External Career Leads

This page acts as a curated directory for external environmental jobs and internships.

— **Filtering and Search :** Users can filter opportunities by category (e.g., Renewable Energy, Conservation) to find roles that match their interests.

— **Redirection Logic :** Since these are external roles, the frontend handles safe redirection to the source websites, while the backend simultaneously logs the "click-through" event for our BI analytics.

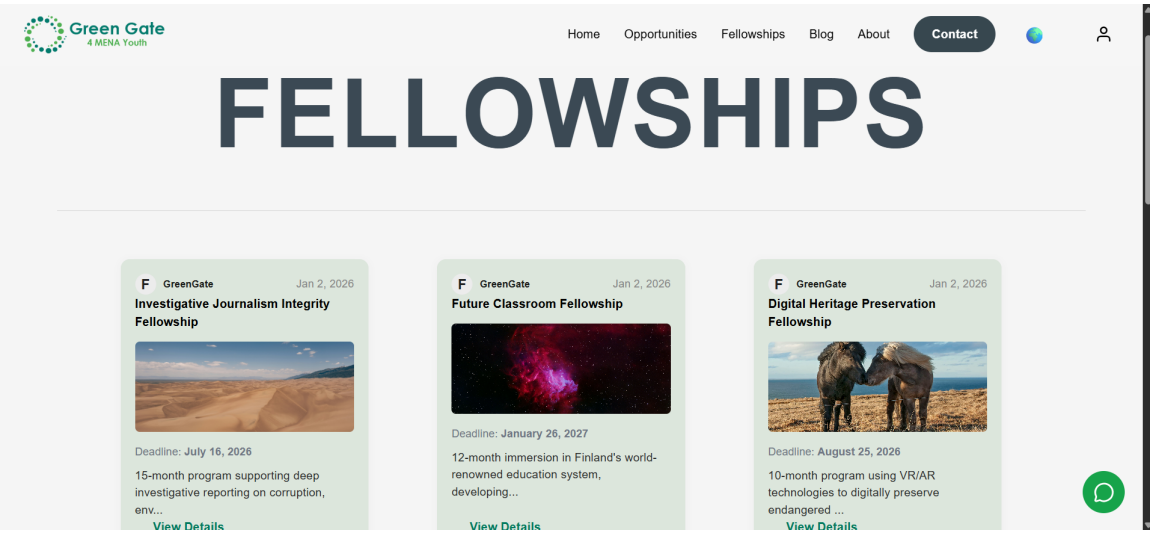


2.4 The Fellowship Portal : Internal Application Workflow

The Fellowship module is the most interactive part of the user platform. Unlike external opportunities, fellowships are managed entirely within GreenGate.

— **Application Form :** A multi-step form allows users to submit their personal details and professional background directly to our database.

— **Status Tracking Dashboard :** Once an application is submitted, users can visit their personal dashboard to track the progress. The UI dynamically reflects the status set by the admin : **Pending** (Yellow), **Accepted** (Green), or **Refused** (Red).



Chapitre 3

The Admin Dashboard : Management and Decision Making

3.1 Administrative Interface Overview

The `greengate-admin` application is a high-security portal built with **React 19** and **Material UI**. It serves as the central control plane for the GreenGate ecosystem, allowing administrators to moderate content, manage users, and analyze platform performance through data-driven dashboards.

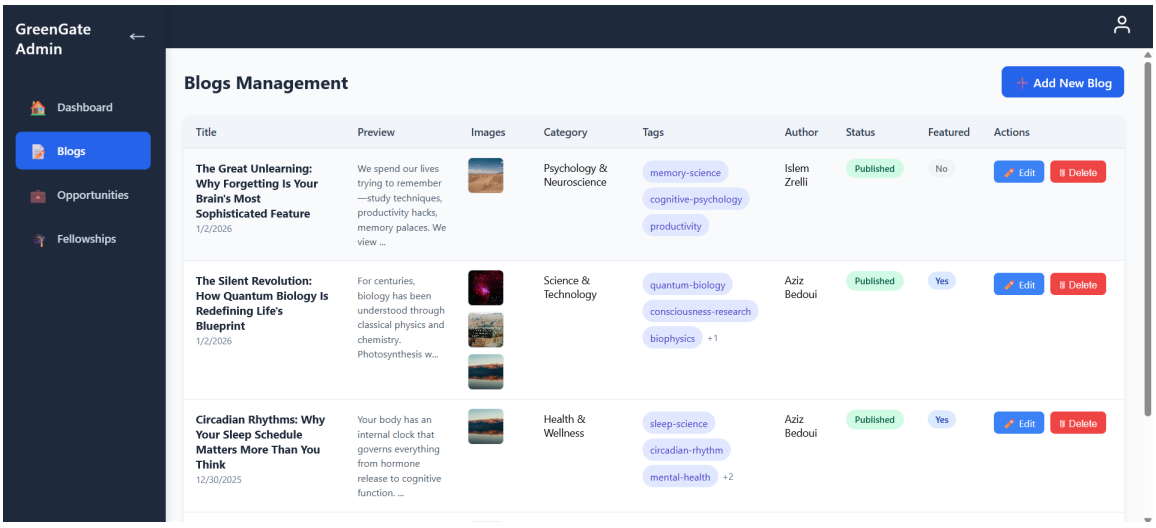
3.2 Content Management and SEO Optimization

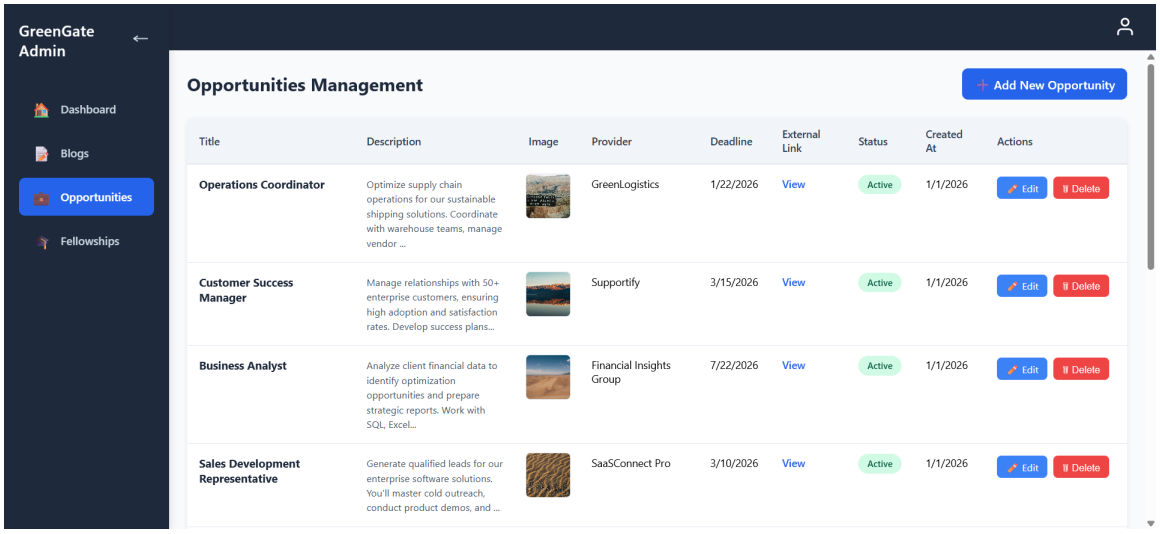
The administrative workspace provides a streamlined interface for the Knowledge Hub and Opportunity Directory.

— **Blogs Management :** Admins have full CRUD control. The interface allows for the toggling of "Featured" status, which directly determines the context used by the Gemini AI agent for generating user insights.

— **Opportunity Curation :** Admins manage the list of external career leads, ensuring all links and descriptions remain relevant to the sustainability sector.

— **SEO Control :** The panel includes fields for meta-descriptions and automated slug generation to maximize search engine visibility.

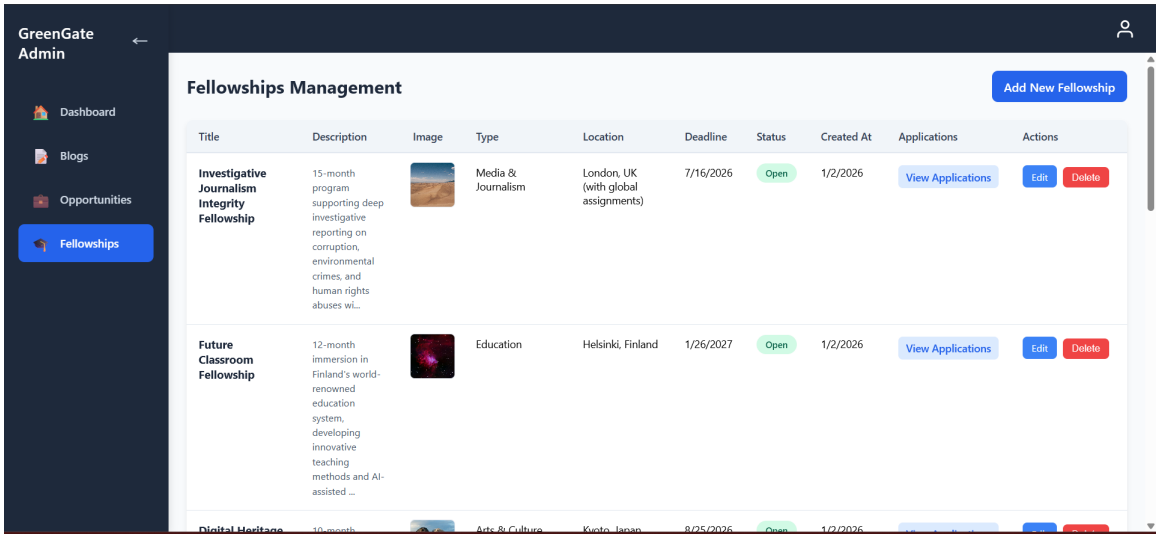




3.3 Fellowship Adjudication Pipeline

The Fellowship Review Portal is a specialized module designed for high-volume application processing. It implements a critical state-management workflow :

- **Profile Evaluation :** Admins can access and review the full professional profile of each applicant.
- **Decision Logic :** The system features a simple but powerful decision interface. Admins can click **Accept** or **Refuse**, which triggers an immediate update to the MongoDB status field and synchronizes the change with the User Frontend.



3.4 Business Intelligence (BI) and Data Visualization

A core innovation of the GreenGate admin panel is the integration of advanced analytics. By utilizing the data logged by our backend analytics engine, the dashboard provides a macroscopic view of the platform's health.

- **Metric Tracking :** The dashboard visualizes key metrics such as total blog views, popular categories, and fellowship application trends.

— **Power BI Integration :** Using the `powerbi-client-react` library, real-time reports are embedded directly into the admin interface, allowing for data-driven strategy adjustments.

3.5 Security and Role-Based Access

To safeguard sensitive applicant data, the admin dashboard is protected by :

— **Role Verification :** The backend middleware ensures that only users with the "admin" role can access the `/api/admin` endpoints.

— **Secure Session Persistence :** Administrative sessions are managed via JWT, ensuring that access is revoked automatically after a period of inactivity.

Conclusion et Perspectives

Summary of Achievements

GreenGate successfully demonstrates the potential of the MERN stack to create a high-impact, content-driven platform tailored for the environmental sector. By synergizing traditional content management with a sophisticated internal fellowship application system and Google Gemini AI integration, the project provides a robust solution for professional ecological advancement. The decoupled architecture ensures high performance and scalability, while the clear separation between the User and Admin frontends facilitates secure, specialized workflows for both the community and platform moderators.

Future Perspectives and Evolutions

While the current version of GreenGate establishes a strong foundation, several strategic evolutions are planned to further enhance the ecosystem's value :

- **AI-Driven Application Automation :** A primary perspective involves expanding the Gemini AI integration to assist users in the application process. Future iterations will include an "AI Application Assistant" that can analyze a user's profile and CV to automatically draft personalized motivation letters and pre-fill fellowship application forms, significantly reducing the friction of the application journey.
- **Monetization and Secure Payments :** To ensure the long-term sustainability of the platform, a payment gateway integration (such as Stripe or PayPal) is envisioned. This will allow for the implementation of application fees for premium fellowships or "Featured Application" tiers, providing a revenue stream to support further platform development while maintaining the integrity of the selection process.
- **Predictive Analytics for Career Coaching :** Utilizing the existing BI data, future updates will focus on predictive modeling to suggest fellowships to users based on their previous application history, interests, and success rates, creating a truly personalized career coaching experience.

In conclusion, GreenGate is not merely a static portal but a dynamic foundation. By evolving toward automated intelligence and integrated financial systems, it aims to become the definitive digital infrastructure for the global sustainability workforce.