

Project Documentation – HITA Website (MERN) Harar

Innovation & Technology Agency September 2025

1 Vision

The **Harar Innovation & Technology Agency (HITA)** website drives **innovation, digital transformation, and startup support** in Harar. It connects **citizens, startups, and agency staff** through services, vacancies, programs, events/news, and success stories, powered by a **fully admin-controlled CMS**.

2 Goals

- Promote **innovation, research, and digital literacy**.
- Provide a **centralized portal** for directorates, services, programs, and updates.
- Enable **transparent vacancy and job application management**.
- Empower admins with **full control over all pages**.
- Ensure **scalable, multilingual, and responsive design**.

3 User Roles & Stories

3.1 Roles

- **Visitor**: Browse content, services, programs, events/news, and vacancies.
- **User**: Register/login, apply for vacancies, track applications.
- **Admin**: Full control over content, users, vacancies, programs, events/news, directorates, and applicants.
- **Directorate Staff**: Manage content for their directorate pages only.

3.2 Stories

- As a **visitor**, I can view services, programs, events/news, and vacancies.
- As a **user**, I can apply for vacancies and receive email updates.
- As an **admin**, I can manage all pages, media, and users.
- As a **directorate staff**, I can update only my department's section.

4 Admin Control System

The admin dashboard is a **comprehensive CMS** for managing website content, media, and users.

4.1 Features

- **Page & Content Management**:
 - Edit **Hero, About, Services, Directorates, Programs, Vacancies, Innovations, Events, and News**.
 - Upload and optimize images/PDFs via **Cloudinary**.
 - Support multilingual text (i18n).
- **Vacancy & Applicant Management**:

- Create, edit, delete **vacancies**.
- Review **applicants**, filter by status, update progress.
- **User Management:**
 - Manage **registered users** and roles (Admin, Staff, User).
 - Assign **directorate responsibilities** to staff.
- **Directorates & Services:**
 - Staff update their **department content**.
 - Admins approve or override edits.
- **Programs, Events, & News:**
 - Publish **programs, announcements, or success stories**.
 - Attach images/PDFs via **Cloudinary**.
- **System Settings:**
 - Manage **supported languages** (i18n).
 - Update **branding** (logos, footer, theme).
 - Configure **email templates** for notifications.

5 System Workflow

1. **Visitors** browse services, vacancies, programs, events/news, and innovations.
2. **Users** register/login and apply for vacancies.
3. **Admins** manage programs, events/news, vacancies, directorates, applicants, and all pages via the CMS.
4. **Directorate staff** update their department's content.
5. **Emails** sent via Mailtrap (dev) or SMTP (production).
6. **Images/media** uploaded and optimized via Cloudinary.

6 Architecture

1	[React Frontend]	<-->	[Express Backend]	<-->	[MongoDB Atlas]
2	(UI)		(REST API)		(Data Storage)

- **Frontend:** React 19, TailwindCSS, Framer Motion, i18n
- **Backend:** Node.js, Express.js, JWT Auth
- **Database:** MongoDB Atlas with Mongoose
- **Email:** Nodemailer + Mailtrap (development)
- **Image/Media:** Cloudinary (upload and optimization)
- **Deployment:** Vercel (frontend), Render/Heroku (backend)

7 Email System (Mailtrap)

7.1 Why Mailtrap?

Safely test emails (signup, password reset, application updates) **without sending to real users**.

7.2 Setup

1. Create an account at [Mailtrap](#).
2. Get SMTP credentials.
3. Add to /server/.env:

```
1 MAILTRAP_HOST=smtp.mailtrap.io
2 MAILTRAP_PORT=2525
3 MAILTRAP_USER=your_mailtrap_username
4 MAILTRAP_PASS=your_mailtrap_password
5 MAIL_FROM=no-reply@hita.gov
```

4. Preview emails in Mailtrap's inbox.

8 Project Timeline (2 Months)

8.1 Month 1 – Foundation

Setup frontend (React, Tailwind, Framer Motion)

Setup backend (Express, MongoDB Atlas)

JWT authentication

Mailtrap integration

Initial Agile backlog

8.2 Month 2 – Features

Hero section with animated cards (i18n)

Services, directorates, programs, events/news pages

Vacancy/job application system

Admin dashboard (CMS for all pages, programs, events/news, directorates, applicants)

Navbar with dropdowns

Responsive design (mobile-first)

CI/CD pipelines (GitHub Actions, Vercel, Render)

Cloudinary media upload integration

9 Current Backlog

Vacancy filtering & search (supports ?page=&limit=&status=)

Directorate content editor for staff

Analytics dashboard for admins

Real-time notifications (email/SMS)

Expand **i18n** translations

10 Roadmap

- **Data analytics** for agency performance
- **Collaboration tools** for startups & directorates
- **Real-time notifications**
- Integration with **national innovation networks**

11 Developer Guide

11.1 Prerequisites

- **Node.js** ≥ 18
- **npm** or **yarn**
- **MongoDB Atlas** account
- **Git**
- **Cloudinary** account

11.2 Clone Repository

```
1 git clone https://github.com/your-org/hita.git
2 cd hita
```

11.3 Environment Variables

Backend (/server/ .env):

```
1 PORT=5000
2 MONGO_URI=mongodb+srv://<username>:<password>@cluster.mongodb.net/hita
3 JWT_SECRET=supersecretkey
4 MAILTRAP_HOST=smtp.mailtrap.io
5 MAILTRAP_PORT=2525
6 MAILTRAP_USER=your_mailtrap_username
7 MAILTRAP_PASS=your_mailtrap_password
8 MAIL_FROM=no-reply@hita.gov
9 CLOUDINARY_CLOUD_NAME=your_cloud_name
10 CLOUDINARY_API_KEY=your_api_key
11 CLOUDINARY_API_SECRET=your_api_secret
```

Frontend (/client/ .env):

```
1 VITE_API_URL=http://localhost:5000/api
```

11.4 Install Dependencies

```
1 # Frontend
2 cd client
3 npm install
4
5 # Backend
6 cd ../server
7 npm install
```

11.5 Run the App

```
1 # Backend
2 cd server
3 npm run dev
4
5 # Frontend
6 cd client
7 npm run dev
```

- **Frontend:** `http://localhost:5173`
- **Backend:** `http://localhost:5000/api`

11.6 Run Tests

```
1 npm run test
```

Note: APIs include error handling (400, 401, 404) with JSON responses.

12 Project Structure

```
1 /client
2   /src
3     /components    # Reusable UI components
4     /pages         # Page-level views
5     /store         # State management (Zustand/hooks)
6     /i18n          # Multilingual config
7   /server
8     /routes        # API endpoints
9     /models        # Mongoose schemas
10    /controllers    # Business logic
11    /middleware     # Auth/validation
12    /config         # DB, SMTP, Cloudinary setup
```

13 Agile Workflow

- **Sprint Length:** 2 weeks
- **Scrum Board:** GitHub Projects / Trello / Jira
- **Meetings:** Weekly planning & retrospective
- **Definition of Done:**
 - Code **peer-reviewed** and merged
 - Tests **passing**
 - Deployed to **staging**
 - Docs **updated**

14 Contributing

1. Fork and clone the repo.
2. Create a branch: `git checkout -b feature/xyz`
3. Commit: `git commit -m "Add feature xyz"`
4. Push: `git push origin feature/xyz`
5. Open a **Pull Request**.

15 License

MIT © 2025 Harar Innovation & Technology Agency

16 API Reference

All endpoints use **JSON** payloads. JWT required for protected routes (Authorization: Bearer <token>).

hitaBlue			
/api/auth/register	POST	Register a new user	No
/api/auth/login	POST	User login, returns JWT	No
/api/users	GET	List all users	Admin
/api/users/:id	PUT	Update user role/details	Admin
/api/vacancies	GET	List all vacancies	No
/api/vacancies	POST	Create a vacancy	Admin
/api/vacancies/:id	PUT	Update a vacancy	Admin
/api/vacancies/:id	DELETE	Delete a vacancy	Admin
/api/applicants	GET	List all applicants	Admin
/api/applicants/:id	PUT	Update applicant status	Admin
/api/programs	GET	List all programs	No
/api/programs	POST	Create a program	Admin
/api/programs/:id	PUT	Update a program	Admin
/api/events	GET	List all events	No
/api/events	POST	Create an event	Admin
/api/news	GET	List all news	No
/api/news	POST	Create a news article	Admin
/api/directors	GET	List all directorates	No
/api/directors/:id	PUT	Update directorate content	Admin/Staff

Notes:

- **Auth:** Use Authorization: Bearer <token> header for protected routes.
- **Error Handling:** Returns JSON with status (e.g., 400, 401) and message.
- **Cloudinary:** Media uploads are handled via /api/upload (POST, Admin/Staff).