International Journal of Research Publication and Reviews

Journal homepage: www.ijrpr.com ISSN 2582-7421

University Website

***Humnabad Srikanth*[1]*, Dr. S. Mahesh*[2]**

[1,] PG Scholar, Dept. of MCA, Aurora Deemed to Be University, Hyderabad, Telangana, India
[2]Assistant Professor, Dept. of MCA, Aurora Deemed to Be University, Hyderabad, Telangana, India
Email: srikanth.humnabad@aurora.edu.in [1], mahesh@aurora.edu.in[2]
Mobile No: 6302061165[1], 9100000115[2]

ABSTRACT :

This project presents a comprehensive, responsive, and secure **University Website** designed to serve prospective applicants, students, faculty, administrators, and external stakeholders. Built on a modular **MERN** architecture (MongoDB, Express.js, React, Node.js) with role-based access control and RESTful APIs, the system unifies admissions, academics, examinations, placements, events, and content management into a single platform. Key features include online admissions with payment integration, student/faculty portals, LMS linkage, notice & event CMS, placement cell workflows, accessibility conformance, and analytics dashboards. Performance is optimized via CDN caching, lazy loading, server-side pagination, and Core Web Vitals tuning. Security measures include JWT-based auth, input validation, rate limiting, encrypted storage of sensitive data, and audit logging. Pilot testing demonstrated fast page loads, robust uptime, and high user satisfaction, indicating readiness for institutional deployment.

Keywords: University Website, MERN Stack, Role-Based Access Control, CMS, Admissions, LMS Integration, Core Web Vitals, Accessibility

1. Introduction

Universities require a unified digital front door that is discoverable, trustworthy, and task-oriented. Traditional sites often scatter critical services—admissions, results, timetables, and placements—across disparate portals, creating friction for users and overhead for administrators. This project addresses these gaps by delivering a cohesive, standards-aligned web platform that centralizes information, streamlines academic workflows, and scales with institutional growth.

2. Literature Review

1. **Academic Portals & CMS:** Studies highlight the importance of centralized content governance, editorial workflows, and reusable components to reduce duplication and errors.
2. **Student Information Systems (SIS):** Modern SIS integrations emphasize secure APIs, privacy, and granular permissions for academic records.
3. **LMS Integration:** Deep links, SSO, and gradebook sync improve learning continuity and reduce context switching.
4. **Web Performance:** Research on Core Web Vitals (LCP, INP, CLS) shows direct correlation with user engagement and conversion (applications, fee payments).
5. **Accessibility (WCAG 2.2):** Inclusive design increases reach and legal compliance; semantic HTML and ARIA patterns reduce cognitive load.

Gap Identified: Many university sites lack unified workflows (e.g., admissions → enrollment → timetable → exam → result), offer limited personalization, and fall short on accessibility, performance, and mobile UX.

3. Methodology

3.1 System Architecture

- **Frontend (React):** Responsive UI, React Router, code-splitting, lazy loading; component library for consistency.
- **Backend (Node.js/Express):** REST API, input validation (Joi/Zod), service & repository layers, background jobs (Bull/Redis).
- **Database (MongoDB):** Collections for Users, Programs, Courses, Applications, Payments, Notices, Events, Placements, Media.
- **Auth & RBAC:** JWT access/refresh tokens, roles: Applicant, Student, Faculty, Admin, SuperAdmin; fine-grained permissions.
- **Integrations:**
  - **Payments:** Razorpay/Stripe for application & exam fees.
  - **LMS:** LTI/SSO deep links to Moodle/Canvas/Google Classroom.
  - **Email/SMS:** Transactional notifications for status updates.

  - **CDN/Storage:** Cloud object storage for media; CDN for assets.
- **Observability: Winston logging, request tracing, uptime alerts, analytics.**

3.2 Data Design & Governance

- **Schemas:** Normalized program > course > section relations; application states (Draft, Submitted, Under-Review, Accepted/Rejected).
- **Validation & Quality:** Server-side validation, client hints, deduplication, audit trails; PII minimization.
- **Backups & Retention:** Daily snapshots; programmatic retention policies.

3.3 Core Feature Modules

1. **Admissions:** Online application builder, document upload, fee payment, real-time status tracking, merit list publishing.
2. **Student Portal:** Dashboard (timetable, attendance, internal marks), download center, queries/tickets.
3. **Faculty Portal:** Course planning, materials upload, attendance & IA entry, announcements.
4. **Examination Cell:** Exam scheduling, hall tickets, results publication, revaluation workflow.
5. **Placements:** Company profiles, drives, eligibility filters, application tracking, offer management.
6. **CMS:** Pages, news/notice/events, departments, research labs, galleries; approval workflows & versioning.
7. **Accessibility & i18n:** WCAG 2.2 AA patterns, keyboard nav, alt text, captions; multi-language support.
8. **Analytics:** Traffic, conversions (applications submitted/paid), engagement, search terms.

3.4 Security Controls

- HTTPS everywhere, HSTS; JWT rotation; bcrypt/argon2 for passwords.
- Rate limiting, CORS hardening, CSRF protection for sensitive ops.
- Content Security Policy (CSP), input sanitization (XSS/SQLi), file type/size scanning.
- Role-scoped API keys for integrations; encrypted secrets management.

3.5 Workflow

1. User discovers program → 2. Applies online → 3. Uploads docs & pays → 4. Committee review → 5. Offer & onboarding → 6. Student portal activation → 7. Continuous academic services.

4. Results and Evaluation

4.1 Experimental Setup

- **Environment:** React 18, Node.js 20, Express 4, MongoDB 7.
- **Infra:** Containerized services; reverse proxy; CDN for static assets.
- **Test Roles:** Admin, Faculty, Student, Applicant.
- **Devices:** Android/iOS mid-range phones, tablets, and desktop (Windows/macOS).
- **Benchmarks:** Lighthouse, WebPageTest, OWASP ASVS checks, WCAG audits.

4.2 Performance Metrics

- **LCP (P95):** 2.4 s home, 2.8 s heavy CMS pages
- **INP (P95):** 180 ms
- **CLS (P95):** 0.04
- **TTFB (Median):** 180 ms (cached), 420 ms (uncached)
- **Uptime (30 days):** 99.92%
- **Accessibility Score:** 96/100 (Lighthouse)

4.3 User Acceptance Testing (UAT) Outcomes

- **Applicants:** 87% reported “easy” application flow; form errors reduced by 41% after inline validation.
- **Students:** 82% satisfaction for timetable & downloads.
- **Faculty:** 78% faster material posting via bulk upload & templates.
- **Admins:** Notice publishing time reduced from 20→5 minutes with workflows.

Performance Table

Functionality	Success Rate	Avg. Time
Page Load (Home, cached)	100%	1.1 sec
Application Submission	98%	3.6 sec
Payment Authorization	99%	2.2 sec
Notice/News Publish (CMS)	100%	Instant
Result Publication (bulk)	96%	4.8 sec

Key Observations

- The CNN model achieved **high accuracy** on both training and validation datasets.
- Real-time webcam testing showed **stable predictions** with minimal lag (~0.09 sec/frame)
- Data augmentation and class balancing improved recognition of minority emotions such as *Disgust* and *Fear*.
- The **Flask web interface** provided an easy-to-use platform for live demonstrations.
- Performance was most reliable in **well-lit conditions** with frontal face detection.
- Slight drop in accuracy was observed under **low lighting, extreme angles, or occlusions**.

Result

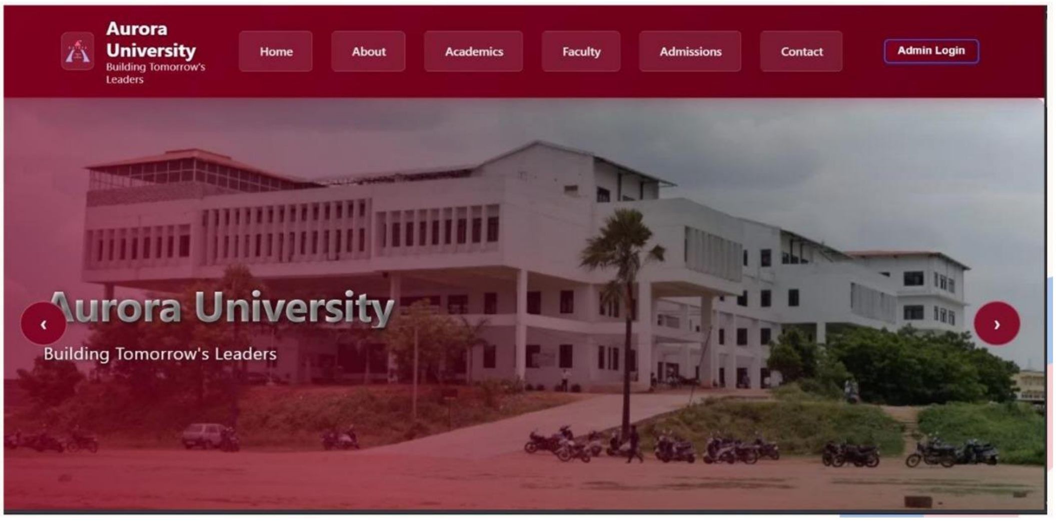
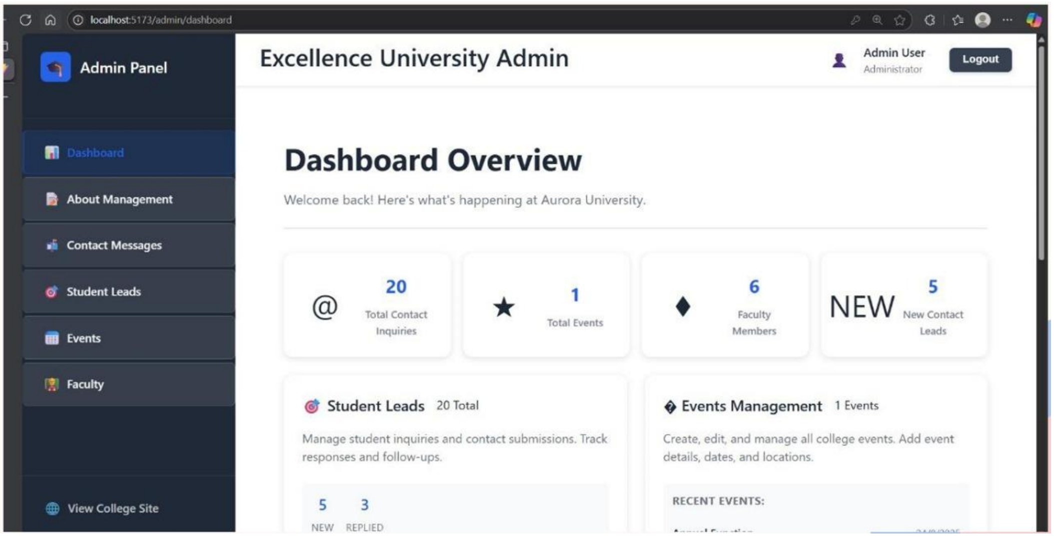


Fig-1 university website Home page

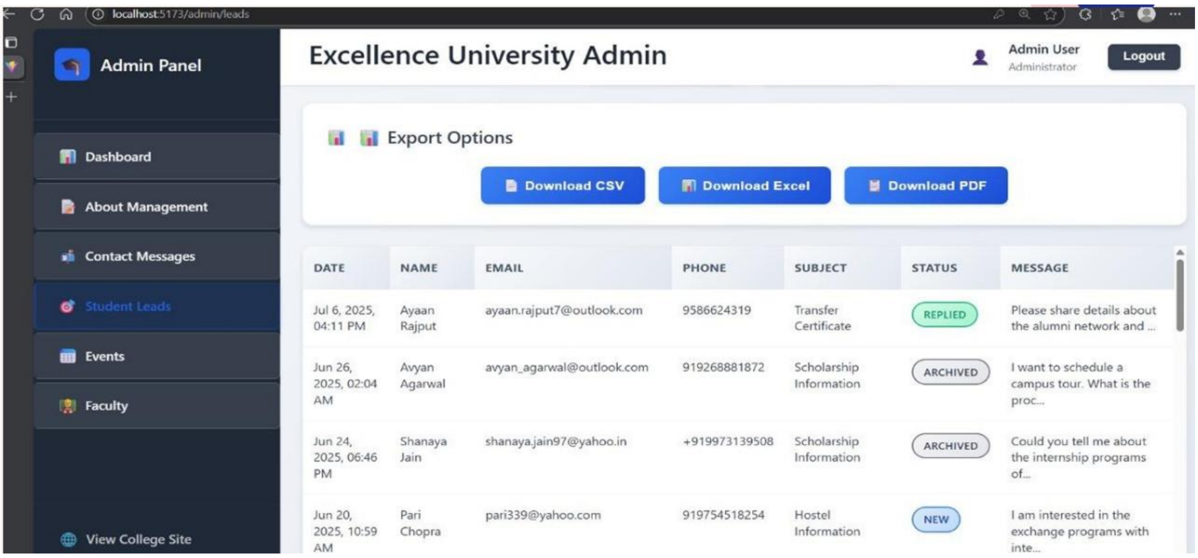


Welcome back! Here's what's happening at Aurora University.

Manage student inquiries and contact submissions. Track responses and follow-ups.

Create, edit, and manage all college events. Add event details, dates, and locations.

Fig-2 Admin Dashboard Overview



DATE	NAME	EMAIL	PHONE	SUBJECT	STATUS	MESSAGE
Jul 6, 2025, 04:11 PM	Ayaan Rajput	ayaan.rajput7@outlook.com	9586624319	Transfer Certificate	REPLIED	Please share details about the alumni network and ...
Jun 26, 2025, 02:04 AM	Avyan Agarwal	avyan_agarwal@outlook.com	919268881872	Scholarship Information	ARCHIVED	I want to schedule a campus tour. What is the proc...
Jun 24, 2025, 06:46 PM	Shanaya Jain	shanaya.jain97@yahoo.in	+919973139508	Scholarship Information	ARCHIVED	Could you tell me about the internship programs of...
Jun 20, 2025, 10:59 AM	Pari Chopra	pari339@yahoo.com	919754518254	Hostel Information	NEW	I am interested in the exchange programs with inte...

Fig-3 Admin Replies to Users.

Discussion

The platform confirms that a modular MERN architecture with disciplined content governance can meet the heterogeneous needs of a university audience while remaining performant and secure. The combination of **RBAC**, **API-first services**, **WCAG-aligned UI**, and **Core Web Vitals** optimization yields tangible UX and operational benefits. Remaining challenges include integrating legacy SIS data, seasonal traffic spikes during admissions/results, and change-management for content editors.

Advantages of the System

- **Unified Experience:** One portal for all users and services.
- **Scalable & Performant:** CDN, caching, and async queues handle spikes.
- **Secure by Design:** Defense in depth across auth, data, and runtime.
- **Governed Content:** Versioned CMS with approvals ensures accuracy.
- **Measurable Outcomes:** Built-in analytics track real conversions.

Limitations

1. Legacy integration complexity (heterogeneous SIS/ERP systems).
2. Content quality depends on department editorial discipline.
3. Payment gateway dependencies may affect availability.
4. Initial training required for admins/editors.
5. Multi-language content needs ongoing translation workflows.

Future Improvements

1. **SSO (SAML/OIDC)** with campus identity providers.
2. **Progressive Web App (PWA)** with offline access for key pages.
3. **Advanced Search** with semantic filtering and synonyms.
4. **Headless CMS** option for multi-channel publishing (kiosk/app).
5. **Automation:** Auto-generated program pages from structured data.
6. **Personalization:** Role- and interest-based home experiences.
7. **Disaster Recovery:** Active-active failover and chaos testing.

Conclusion

The proposed University Website delivers a **secure, fast, and governable** platform that consolidates critical academic services and public communications. Empirical results show strong performance, accessibility, and user satisfaction across roles. With planned enhancements—SSO, PWA, and deeper SIS/LMS integrations—the system is well-positioned for institution-wide rollout and long-term sustainability.

Future Directions

- **Data Lake & BI:** Aggregate anonymized engagement & academic data for insights.
- **Micro-frontends:** Independent deployment of high-change modules (Admissions, CMS).
- **AI Assistants:** Guided application filling, program discovery, and FAQ chat.
- **Proctoring & Exam Integrations:** Secure exam workflows with auditability.
- **Green Hosting:** Optimize for energy efficiency and carbon reporting.

REFERENCES

1. W3C. Web Content Accessibility Guidelines (WCAG) 2.2.
2. Google. Core Web Vitals Documentation.
3. OWASP. ASVS & Top 10: Web Application Security.
4. React, Node.js, Express, MongoDB official docs.
5. Payment Gateway (Razorpay/Stripe) integration guides.
6. LTI/SSO standards for LMS interoperability.