

PUK360: Project Overview

PUK360 is a cloud-hosted, web-based application designed as a centralized digital hub for all **North-West University (NWU)** events. Developed by NWU students, including Project Manager **Zai Hansa**, the platform aims to unify fragmented campus communication and improve student engagement.

Quick Links

- **Live Application:** [Access PUK360](#)
- **Video Demonstration:** [Watch the Demo](#)
- **Frontend Repository:** [GitHub - React/Tailwind](#)
- **Backend Repository:** [GitHub - Node.js/Express](#)

Key Features

- **Centralized Event Feed:** A real-time listing of all upcoming NWU events, including faculty, SRC, residence, and society activities.
- **Smart Discovery:** Users can search and filter events by category, campus, or specific dates.
- **Interactive Event Details:** Includes detailed descriptions, venue information, organizer contacts, and downloadable posters.
- **Feedback System:** Students can leave star ratings and written reviews after attending events.
- **Mobile-Optimized:** A fully responsive design ensures seamless access across desktop and mobile devices without an app installation.

User Roles & Permissions

The system utilizes secure, role-based access control tied to official NWU credentials:

Role	Capabilities
Students	Browse feed, RSVP to listings, manage profiles, and submit reviews.
Hosts	Create and manage event listings (edit/duplicate/delete) and track engagement.
Admins	Oversee the ecosystem, moderate events, review host applications, and view analytics.

✂ Technical Architecture

The platform is built on a modern, scalable tech stack hosted on **Amazon Web Services (AWS)**:

- **Frontend:** React and Tailwind CSS for a fast, modular, role-aware user interface.
- **Backend:** Node.js and Express.js, using JWT (JSON Web Tokens) for secure, stateless authentication.
- **Database:** Live SQL Server/Azure SQL database managed via the Sequelize ORM.
- **Storage & API:** Event posters are stored in AWS S3 using presigned URLs, with backend documentation provided via Swagger/OpenAPI.

Would you like me to help you draft a formal README file for the GitHub repositories based on this information?