BlueServe

A Centralized Platform for Ateneo's Service Hour Ecosystem

Introduction

Every year, hundreds of Atenean scholars are required to fulfill a set number of service hours to maintain their scholarships. These service hours are typically rendered by volunteering for university offices, student organizations, or official events. Despite the significance of these service hours, the current system managing these affairs is inefficient and scattered.

For over a decade, a website built by scholars in 2012 served as the primary avenue for logging and discovering service hour opportunities. Unfortunately, the platform was shut down at the end of last school year due to ___, leaving students without a centralized system to manage and track their hours. In its absence, scholars have reverted to relying on Facebook groups and posts – an approach that is cluttered, unreliable, and inadequate for long-term data tracking, validation, and uniformity.

Problem Statement

There is currently no unified platform that:

- Allows students to easily discover and filter service hour opportunities
- Enables organizations and offices to formally request and confirm volunteers
- Helps the Office of Admission and Aid (OAA) monitor and maintain a database of student service hour records

Objectives

This project aims to address these challenges by building a comprehensive and user-friendly Django-based web application that supports the full lifecycle of service hours, from opportunity discovery and signup to hour fulfillment and tracking. The system is tailored to the needs of three main stakeholders:

1. students,
2. organizations and university offices,
3. and the Office of Admission and Aid (OAA).

Deployment and Setup

To run the project locally:

1. Choose a location to clone the repository to
   cd "___"
2. Clone the GitHub Repository
   git clone https://github.com/lmmtolin/004_itmgt25_project.git

3. Navigate to servicehourstracker
   cd 004_itmgt25_project/servicehourstracker
4. Install QR Code
   pip install qrcode[pil]
5. Run the Python Server
   python manage.py runserver
6. Open development server in a browser

Key Features

1. OAA Dashboard
   a. **Description**
      Enables the OAA to track student student service hours, process volunteer requests from organizations/offices, and maintain student records.
   b. **Features**:
      i. Approve/deny volunteer requests
      ii. Track service hours
      iii. Access and maintain a database of scholars with their requirements
      iv. Add, remove, or update student information
   c. Key Data
      i. Service Hours: Total, pending, required per semester, penalties, excess/deficit
      ii. Student Information: Student ID, service hour status
2. Ateneo Office/Organization Dashboard
   a. **Description**
      Allows university offices and organizations to request volunteers and confirm attendance using QR codes
   b. Features
      i. Submit detailed volunteer requests for upcoming events
      ii. Confirm attendance by scanning student-generated QR codes
3. Volunteer/Student Dashboard
   a. **Description**
      Allows students to discover, filter, and manage service hour opportunities aligned with their schedules and track their records.
   b. **Features**:
      i. View summary of service hours
      ii. Sign up for events
      iii. Upload class schedules
      iv. Generate and retrieve QR codes for event check-ins
4. Other Features:
   a. Penalties for no-shows or incomplete service hours
   b. Retrieve class schedules directly from AISIS (to be confirmed)