Stakeholders for Online Course Registration System

Primary Stakeholders

1.Students

•Role: End users of the system.

•Interest: Students use the platform to search for courses, view availability, and register.

•Importance: High – The usability and accuracy of the system directly impact their academic scheduling.

2.Instructors

•Role: Providers of course information.

•Interest: Instructors need to view and manage their course rosters, schedules, and capacity.

•Importance: Medium – Instructors rely on the system for scheduling, but interact with fewer functions than students.

3.Academic Advisors

•Role: Assist students with course planning and academic progress.

•Interest: They need access to students’ schedules and course offerings to guide them effectively.

•Importance: Medium – Their work supports the student experience but doesn’t depend on every feature.

4.University Registrar’s Office

•Role: Oversees course offerings, registration deadlines, and academic records.

•Interest: Ensures the system maintains integrity and follows institutional policies.

•Importance: High – This stakeholder sets registration rules and needs accurate backend functionality.

5.IT Support Team

•Role: Maintains the system’s technical performance and resolves bugs.

•Interest: Needs access to backend data, issue logs, and system monitoring tools.

•Importance: Medium – They ensure continuous operation but don’t use the product in a traditional user sense.

6.University Administration

•Role: Decision-makers on budgeting and system-level features.

•Interest: Wants analytics on system usage, capacity planning, and performance indicators.

•Importance: Low to Medium – They are strategic users, not everyday users, but they influence long-term functionality.

**Dependency Considerations -**

**“Is this story foundational to other stories? Higher priority should be assigned to user stories that unblock subsequent work.”**

1. **User Authentication**  
    **• Dependent On: —**  
    **• Justification: Every secured action (viewing courses, registering, paying, profile edits, drops) requires users to be authenticated first.**
2. **Course Catalog Upload**  
    **• Dependent On: Authentication**  
    **• Justification: Without a populated catalog, students can’t browse or register for courses.**
3. **View Available Courses**  
    **• Dependent On: Course Catalog Upload, Authentication**  
    **• Justification: Students need an up-to-date catalog to make registration decisions.**
4. **Register for Courses**  
    **• Dependent On: View Available Courses, Authentication**  
    **• Justification: You can’t register for a course until you know what’s offered and you’re logged in.**
5. **Drop Registered Courses**  
    **• Dependent On: Register for Courses, Authentication**  
    **• Justification: Only courses already registered can be dropped.**
6. **Tuition Payment**  
    **• Dependent On: Register for Courses, Authentication**  
    **• Justification: Payment only makes sense once you know what you’ve registered for.**
7. **Email Confirmation**  
    **• Dependent On: Tuition Payment, Authentication**  
    **• Justification: Confirmations should only go out after successful payment and enrollment.**
8. **User Profile Management**  
    **• Dependent On: Authentication**  
    **• Justification: Profile edits can occur any time post-login and don’t block core registration flows.**
9. **Generate Enrollment Reports**  
    **• Dependent On: Course Catalog Upload, Register for Courses, Authentication**  
    **• Justification: Reporting needs both the catalog structure and registration data.**

**10. Password Reset**  
 **• Dependent On: Authentication**  
 **• Justification: The reset flow relies on the authentication framework being in place.**

Time Sensitivity for Course Registration System

**In our Course Registration System, we use time sensitivity to prioritize features based on their urgency and relevance to sprint goals and demo expectations.**

**Features that were required for our MVP and class sprint demo were labeled as high-priority due to their time-sensitive nature. Medium-priority features were important but not immediately required, while low-priority stories could be implemented in later sprints without affecting initial system functionality.**

**We followed the bell curve model as required, assigning:**

* **2-3 High-priority stories**
* **4-6 Medium-priority stories**
* **2-3 Low-priority stories**

**Below are examples from our backlog with their time sensitivity classifications:**

#### **High Time Sensitivity (Must be completed early):**

* **“As a student, I want to search and view available courses so that I can plan my schedule.”**  
   **→ Required for MVP and public demo.**
* **“As a student, I want to register for courses so I can secure a spot.”**  
   **→ Core functionality for validating enrollment and testing relationships.**
* **“As a student, I want to create an account with my student email so I can start registering for courses.”**
  + **This is the first step in using the platform. Without an account, students can’t do anything else. It’s the foundation of the system.**

#### **Medium Time Sensitivity (Important, but not urgent):**

* **“As an instructor, I want to view my course roster so I can track enrollments.”**  
   **→ Supports instructor view but not required in first sprint.**
* **“As an advisor, I want to view student schedules to assist with academic planning.”**  
   **→ Helps planning but not central to MVP.**
* **“As a student, I want to drop a course if my schedule changes.”**  
   **→ Needed eventually but can come after initial registration flow.**

#### **Low Time Sensitivity (Can be delayed):**

* **“As a student, I want to rate a course after completing it.”**  
   **→ Enhancement feature is not needed for initial functionality.**
* **“As a student, I want to hide full courses from search results so I don’t waste time trying to sign up for something I can’t take.”**

**→ This just makes the experience smoother and prevents frustration.**

Team Contribution Statement

We, the members of the team, affirm that all team members were actively involved in the completion of this assignment and contributed equally to the creation of the product backlog and supporting documentation.

All work was completed collaboratively, and each team member participated in the development of user stories, prioritization decisions, and the review process.

Team Members:

•Janelle Offeh

•Carolina Calderon

•Amin Adam

•Raj Patel