

Class Modeling

Zachary Handel

View Expired List of Courses for School-(VECS)

- **Actor:** Admin
- **Goal:** **View** a list of expired courses that were previously **offered** by the school.
- **Preconditions:**
 1. There must be an admin account registered in the system.
 2. The admin must be logged in.
- **Scenario:**
 1. The admin **accesses** the course management section of the system.
 2. The admin **selects** the "view courses" section from the page.
 3. The system **displays** a list of all courses at the school.
 4. The admin with **filter** the status column to only be able to **see** the classes with a status of expired.
 5. The admin has successfully **viewed** the expired list of courses for the school.

View Courses by Teacher-(VCT)

- Actor: Admin
- Goal: **View** a list of courses taught by a specific teacher.
- Preconditions:
 1. There must be an admin account registered in the system.
 2. The admin must be logged in.
- Scenario:
 1. The admin **accesses** the course management section of the system.
 2. The admin **selects** the "view courses" section from the page.
 3. The system **displays** a list of all courses at the school.
 4. The admin will **filter** the teacher column to only be able to **see** the classes with the teacher of their choosing.
 5. The admin has successfully **viewed** all courses taught by a specific teacher.

Purge Expired Courses-(PEC)

- Actor: Admin
- Goal: To **purge** expired courses from the list of expired courses.
- Preconditions:
 1. There must be preexisting expired courses for the school.
- Scenario:
 1. The user **accesses** the "building management" tab.
 2. The user **selects** the "manage expired courses" section from the Page.
 3. The user is presented with a list of expired courses.
 4. The user **clicks** one of the expired courses which **highlights** it and enables the "Remove" button.
 5. The user **clicks** the "Remove" button.
 6. The user is **prompted** with a dialog that allows them to **confirm** if they want to **purge** the expired courses from the list of expired courses with the options "Cancel" and "Remove".
 7. The user **clicks** the "**Remove**" button.
 8. The dialog **closes** and the user is presented with an updated list of expired courses without the removed course.
 9. The user has **purged** an expired course from the list of expired courses.

Purge Expired Attendance Records-(VACS)

- Actor: Admin
- Goal: To **purge** an expired attendance record from the list of attendance records.
- Preconditions:
 1. There must be preexisting expired attendance records for a course.
- Scenario:
 1. The user **accesses** the "building management" tab.
 2. The user **selects** the "manage expired courses" section from the Page.
 3. The user is presented with a list of expired courses with a button next to each course that says "View Attendance Records"
 4. The user **clicks** "View Attendance Records" next to the desired course they want to **remove** a record for.
 5. The user is **redirected** to a list of expired attendance records for the selected course.
 6. The user **clicks** on one of the expired attendance records which **highlights** it and **enables** the "Remove" button.
 7. The user **clicks** the "Remove" button.

8. The user is **prompted** with a dialog that **allows them to confirm** if they want to **purge** the expired attendance record from the list of expired attendance records with the options "Cancel" and "Remove".
9. The user **clicks** the "Remove" button.
10. The dialog **closes** and the user is **presented** with an updated list of expired attendance records without the removed attendance Record.
11. The user has **purged** an expired attendance record from the list of expired attendance records.

Create Faculty Account-(CFA)

- Actor: Admin
- Goal: To **create** a faculty account for a school.
- Preconditions:
 1. An admin account must be **registered** on the platform.
 2. The desired faculty member must have an existing e-mail account.
- Scenario:
 1. The user **accesses** the "building management" tab.
 2. The user is **redirected** to the "building management" view.
 3. The user **selects** the "manage faculty" tab.
 4. The user is **redirected** to the "manage faculty" tab which **displays** a list of current faculty members as well as a button that says "Add Faculty".
 5. The user **selects** the "Invite Faculty" button.
 6. The user is **redirected** to a form that **asks for the details of the faculty member they want to add**. The user **inputs** the details which includes a required e-mail address.
 7. The user **clicks** "Send Invite" which **sends** an email to the desired faculty member.
 8. The faculty member will follow the FRA use case.
 9. The user is **redirected** to the "manage faculty" view.
 10. Once the faculty member follows the FRA use case, the faculty account is **added** to the list of faculty accounts.
 11. The faculty account is **created** for the school.

Delete Faculty Account-(DFA)

- Actor: Admin
- Goal: To **delete** a faculty account from a **school**.
- Preconditions:
 1. There must be a preexisting faculty account in the list of faculty accounts.
- Scenario:
 1. The user **accesses** the "building management" tab.
 2. The user is **redirected** to the "building management" view.
 3. The user **selects** the "manage faculty" tab.
 4. The user is **redirected** to the "manage faculty" tab which **displays** a list of current faculty members as well as a button that says "Delete Account".
 5. The user **clicks** the desired faculty member account to **delete** which **highlights** the account and **enables** the "Delete Account" Button.
 6. The user **clicks** the "Delete Account" button.
 7. The user is **prompted** with a dialog that allows them to **confirm** if they want to **delete** the faculty account from the list of faculty accounts with the options "Cancel" and "Delete".
 8. The user **clicks** "Delete".
 9. The dialog **closes** and the user is **presented** with an updated list of faculty members without the deleted faculty account.
 10. The user has **deleted** a faculty account from the school.

Edit Faculty Account-(EFA)

- Actor: Admin
- Goal: To **edit** a faculty member's account information
- Preconditions:
 1. There must be a preexisting faculty account in the list of faculty accounts.
- Scenario:
 1. The user **accesses** the "building management" tab.
 2. The user is **redirected** to the "building management" view.
 3. The user **selects** the "manage faculty" tab.
 4. The user is **redirected** to the "manage faculty" tab which **displays** a list of current faculty members as well as a button that says "Edit".
 5. The user **clicks** the desired faculty member account to **edit** which **highlights** the account and **enables** the "Edit" button.

6. The user **clicks** the "Edit" button.
7. The user is **redirected** to a view that **displays** all editable data for the faculty member.
8. The user **edits** the **desired data** on the **view** for the faculty member.
9. The user **clicks** "Save".
10. The user is **redirected** back to the "manage faculty" view.
11. The user has **edited** the faculty member's account.