**EduConnect Milestone 2**

Kyler Andrews, Dorien Parris, Fayolah Pierre, Deyandra Burke  
Florida Agricultural and Mechanical University  
CIS4301: Information Systems Design & Development

Dr. Xavier Caddle  
March 3, 2025

Table of Contents

[Team Meeting 3](#_Toc191846017)

[WBS…………………………………………………………………………………4](#_Toc191846018)

[Gantt](#_Toc191846019) Chart…………………………………………………………………………………………………………..5

Network Diagram…………………………………………………………………………………………………………..6

***EduConnect – Detailed Requirements Document…………………………….7***

***Use Cases……………………………………………………………………….12***

Research On Other Systems……………………………………………………………..21

[API……………………………………………………………………………….22](#_Toc191846021)

|  |  |
| --- | --- |
| Team Meeting | Date: 2/25/2025Time: 2:30Location: Zoom |

|  |  |  |  |
| --- | --- | --- | --- |
| Meeting called by: | Fayolah Pierre | Type of meeting: | Brief Zoom meeting |
| Facilitator: | Kyler Andrews | Note taker: | Kyler Andrews |
| Timekeeper: | Fayolah Pierre |  |  |

|  |  |
| --- | --- |
| Attendees: | Kyler Andrews |
| Please read: | Milestone 2 Requirments |
| Please bring: | n/a |

# Minutes

|  |  |  |  |
| --- | --- | --- | --- |
| Agenda item: | Milestone 2 | Presenter: | Fayolah Pierre |

#### Discussion:

This was just something brief to create deadlines for the team contract to make sure everyone had enough time to read and agree to the terms.

#### Conclusions:

Everyone was informed when it was sent out to their emails so that they could read it and suggest changes if they wanted.

| Action items | Person responsible | Deadline |
| --- | --- | --- |
| * 1&2 * 3&4 | Fayolah P.  Dorien P. | 3/2/2025  3/2/2025 |
| * 5&6 | Deyandra B. | 3/2/2025 |
| * 7&8 | Kyler A. | 3/2/2025 |

|  |  |  |  |
| --- | --- | --- | --- |
| Agenda item: | Enter agenda item here | Presenter: | Enter presenter here |

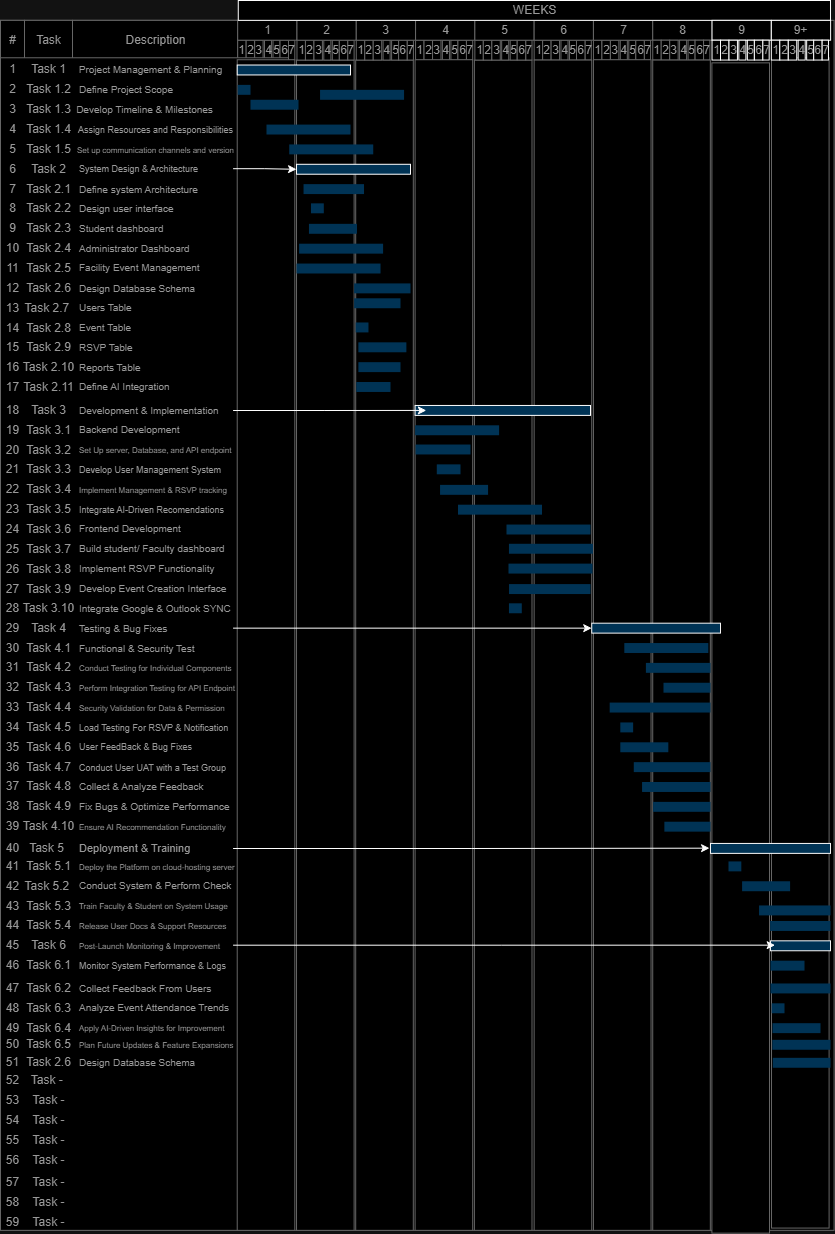
#### Discussion:

This was just something brief to create deadlines for the team’s assignment to make sure everyone had enough time to read and agree to the terms.

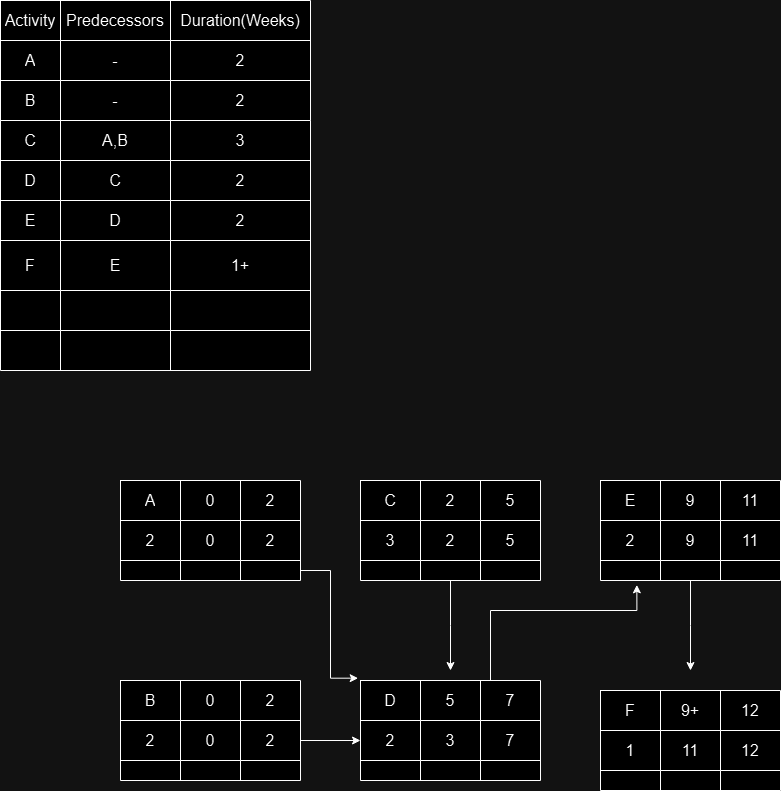
A diagram of a company

Description automatically generated

**Gantt Chart**



**Network diagram**



***EduConnect – Detailed Requirements Document***

**1. Project Overview**

**1.1 Purpose**

EduConnect is an educational event management application designed to improve communication and organization of academic and extracurricular events. The platform provides students, faculty, and administrators with real-time event notifications, RSVP functionality, and AI-driven recommendations to enhance engagement.

**1.2 Problem Statement**

Many students miss out on important academic and extracurricular events due to ineffective communication and organization. EduConnect addresses this issue by centralizing event information, offering personalized alerts, and integrating with existing scheduling tools.

**1.3 Objectives**

* Provide a **centralized** platform for event management.
* Enable **real-time notifications** and RSVP tracking.
* Offer **personalized event recommendations** through AI.
* Integrate with **Google Calendar, Outlook, and Apple Calendar**.
* Provide **detailed analytics** for administrators.

**2. System Requirements**

**2.1 Functional Requirements**

1. **User Registration & Authentication**
   1. Role-based access control (Administrators, Students, Faculty, Report Generators).
   2. Secure login using email/password and OAuth authentication (Google, Microsoft).
   3. Profile setup with customizable preferences.
2. **Event Management**
   1. Create, edit, and delete events.
   2. Categorize events (academic, social, sports, clubs, etc.).
   3. RSVP tracking and attendance management.
3. **Notifications & Alerts**
   1. Real-time push notifications, SMS, and email reminders.
   2. AI-based event recommendations.
4. **RSVP & Attendance Tracking**
   1. Users can RSVP and update attendance status.
   2. Attendance trends stored for reporting and analysis.
5. **Calendar Integration**
   1. Synchronization with Google Calendar, Outlook, and Apple Calendar.
6. **Reports & Analytics**
   1. Engagement reports for administrators.
   2. AI-driven attendance forecasting.

**2.2 Non-Functional Requirements**

* **Performance**: The system must support at least 10,000 concurrent users.
* **Scalability**: Cloud-based infrastructure with horizontal scaling capabilities.
* **Security**: End-to-end encryption and secure API access.
* **Compliance**: Adherence to **FERPA** and **GDPR** for data protection.

**3. Technical Requirements**

**3.1 Technology Stack**

|  |  |
| --- | --- |
| **Component** | **Technology Choice** |
| **Frontend** | React (Web), Flutter (Mobile) |
| **Backend** | Node.js with Express |
| **Database** | PostgreSQL / MongoDB |
| **Hosting** | AWS / Firebase |
| **Authentication** | Firebase Auth / OAuth |
| **AI Engine** | Python (TensorFlow for event recommendations) |
| **Push Notifications** | Firebase Cloud Messaging (FCM) |

**4. Database Design**

|  |  |
| --- | --- |
| **Table Name** | **Description** |
| **Users** | Stores user details (ID, name, email, role, preferences). |
| **Events** | Stores event details (ID, name, date, time, location, category). |
| **RSVP** | Tracks user attendance for events. |
| **Reports** | Stores engagement and analytics data. |

**5. Development Roadmap**

**5.1 Phase 1: Planning (1 Month)**

* Define project scope and technical specifications.
* Create UI/UX wireframes.

**5.2 Phase 2: Development (3-4 Months)**

* Develop core functionalities (event creation, RSVP, notifications).
* Implement AI-driven recommendations.
* Build database structure and APIs.

**5.3 Phase 3: Testing & Deployment (2 Months)**

* Conduct unit and integration testing.
* Launch **beta version** in selected schools.
* Optimize performance and security.

**5.4 Phase 4: Marketing & Launch (1 Month)**

* Promote the app via school partnerships.
* Deploy mobile and web apps.
* Gather feedback for future improvements.

**6. Startup Requirements**

**6.1 Team & Roles**

* **Project Manager** – Oversees development and deployment.
* **Backend Developer** – Handles database and API logic.
* **Frontend Developer** – Builds the UI/UX.
* **AI Engineer** – Implements event recommendations.
* **Marketing Team** – Manages outreach and promotions.

**6.2 Budget & Funding**

|  |  |
| --- | --- |
| **Expense Category** | **Estimated Cost** |
| **Development** | $50,000 (Team salaries, tools) |
| **Hosting & Infrastructure** | $10,000 (AWS, Firebase) |
| **Marketing & Outreach** | $5,000 (Ads, promotions) |
| **Legal & Compliance** | $5,000 (Privacy policies, licensing) |
| **Total Estimated Budget** | **$70,000** |

**7. Conclusion**

EduConnect aims to enhance school event management by providing a **user-friendly**, **AI-powered**, and **efficient** platform for students, faculty, and administrators. With a solid development and marketing strategy, this app can significantly improve event communication and participation in schools.

**Use Case 1: Register for an Event**

* **Primary Actor:** Student
* **Goal:** The student successfully registers for an event on EduConnect.
* **Preconditions:**
  + The student has an active account.
  + The student is logged into the system.
  + The event is open for registration.
* **Trigger:** The student selects an event they want to attend.
* **Main Success Scenario:**
  1. The student logs into EduConnect.
  2. The student navigates to the "Upcoming Events" section.
  3. The student selects an event from the list.
  4. The system displays the event details, including date, time, location, and available slots.
  5. The student clicks the "Register" button.
  6. The system verifies event availability and confirms the registration.
  7. The system updates the RSVP list and sends a confirmation email to the student.
  8. The student receives a calendar invite and a registration success notification.
* **Alternative Flows:**
  + **3a:** If the event is full, the system offers a "Join Waitlist" option.
  + **6a:** If the registration process fails due to a system error, the student is notified and prompted to retry later.
  + **7a:** If the student wants to cancel their registration, they can do so from their profile under "My Events."

**Use Case 2: Create an Event**

* **Primary Actor:** Event Organizer (Admin/Faculty/Campus Organization)
* **Goal:** The event organizer creates and publishes an event.
* **Preconditions:**
  + The event organizer is logged in with the necessary permissions.
  + The required event details are ready.
* **Trigger:** The organizer initiates the "Create Event" process.
* **Main Success Scenario:**
  1. The event organizer logs into EduConnect.
  2. The organizer navigates to the "Event Management" section.
  3. The organizer clicks on "Create New Event."
  4. The system prompts the organizer to enter event details:
     + Title
     + Description
     + Date & Time
     + Location (Physical/Virtual)
     + RSVP options (e.g., open registration, invitation-only)
     + Event capacity limit
  5. The organizer uploads images, flyers, or promotional materials.
  6. The system provides an event preview for review.
  7. The organizer clicks "Publish," and the event is added to the system.
  8. The system sends notifications to relevant students and adds the event to the public event list.
* **Alternative Flows:**
  + **4a:** If required fields are missing, the system prompts the organizer to fill in the missing information.
  + **7a:** If the event conflicts with another major event, the system suggests an alternative time slot.
  + **8a:** If the organizer wants to save the event as a draft, they can do so and publish it later.

**Use Case 3: Send Event Reminder**

* **Primary Actor:** EduConnect System
* **Goal:** The system automatically sends event reminders to registered students.
* **Preconditions:**
  + The event is scheduled.
  + Students have registered for the event.
  + Notification settings allow event reminders.
* **Trigger:** The system checks for upcoming events.
* **Main Success Scenario:**
  1. The system identifies all registered students for an upcoming event.
  2. 24 hours before the event, the system sends a reminder via:
     + Email
     + Push notification (if enabled)
  3. One hour before the event, a final reminder is sent.
  4. If the event is virtual, the system includes the meeting link in the reminder.
  5. The system tracks email open rates and notification delivery status.
* **Alternative Flows:**
  + **2a:** If a student has opted out of notifications, no reminders are sent.
  + **3a:** If the system fails to send the email, it retries after 15 minutes.
  + **4a:** If the event is canceled or rescheduled, the system sends a cancellation notice with updated details.

**Use Case 4: Cancel Event Registration**

* **Primary Actor:** Student
* **Goal:** The student cancels their registration for an event.
* **Preconditions:**
  + The student is logged into EduConnect.
  + The student is already registered for the event.
* **Trigger:** The student decides to cancel their RSVP.
* **Main Success Scenario:**
  1. The student logs into EduConnect.
  2. The student navigates to "My Events."
  3. The student selects the event they want to cancel.
  4. The system displays event details with a "Cancel Registration" button.
  5. The student clicks "Cancel Registration."
  6. The system asks for confirmation before proceeding.
  7. The student confirms, and the system removes their name from the RSVP list.
  8. The system sends a cancellation confirmation email.
* **Alternative Flows:**
  + **3a:** If the event has a cancellation deadline, the system prevents cancellation and notifies the student.
  + **7a:** If the system fails to process the cancellation due to a network issue, it prompts the student to retry later.

**Use Case 5: View Event Analytics**

* **Primary Actor:** Event Organizer
* **Goal:** The event organizer views engagement and attendance analytics for past events.
* **Preconditions:**
  + The organizer is logged in with event management permissions.
  + At least one event has already occurred.
* **Trigger:** The organizer selects an event to review analytics.
* **Main Success Scenario:**
  1. The organizer logs into EduConnect.
  2. The organizer navigates to "Event Management" and selects "View Analytics."
  3. The system retrieves and displays event insights, including:
     + Total number of RSVPs
     + Actual attendance rate
     + No-show percentage
     + Feedback ratings (if applicable)
  4. The organizer exports analytics as a report.
* **Alternative Flows:**
  + **3a:** If no data is available, the system notifies the organizer.
  + **4a:** If the export fails due to a technical issue, the system provides an error message and a retry option.

**Use Case 6: Send Event Feedback Survey**

* **Primary Actor:** EduConnect System
* **Goal:** The system automatically sends event feedback surveys to attendees.
* **Preconditions:**
  + The event has concluded.
  + The organizer has enabled feedback collection.
* **Trigger:** The system detects that the event has ended.
* **Main Success Scenario:**
  1. The system identifies all attendees.
  2. Within 24 hours after the event, the system sends a feedback request via email and in-app notifications.
  3. The student clicks the survey link and submits feedback.
  4. The system stores responses and generates analytics for the event organizer.
* **Alternative Flows:**
  + **2a:** If the attendee has disabled feedback notifications, no survey is sent.
  + **3a:** If the student does not respond, the system sends a one-time follow-up reminder.

**Use Case 7: Manage User Roles**

* **Primary Actor:** System Administrator
* **Goal:** The administrator assigns or revokes permissions for users.
* **Preconditions:**
  + The administrator is logged in with system management rights.
  + The user being modified has an active account.
* **Trigger:** The administrator needs to update a user's role.
* **Main Success Scenario:**
  1. The administrator logs into EduConnect.
  2. The administrator navigates to "User Management."
  3. The system displays a list of users with current roles.
  4. The administrator selects a user and modifies their role (e.g., Student → Event Organizer).
  5. The system saves changes and updates access rights immediately.
  6. The user receives an email notification about their new role.
* **Alternative Flows:**
  + **4a:** If the role assignment conflicts with security policies, the system denies the request.
  + **6a:** If email notifications fail, the system logs an error and suggests manual contact.

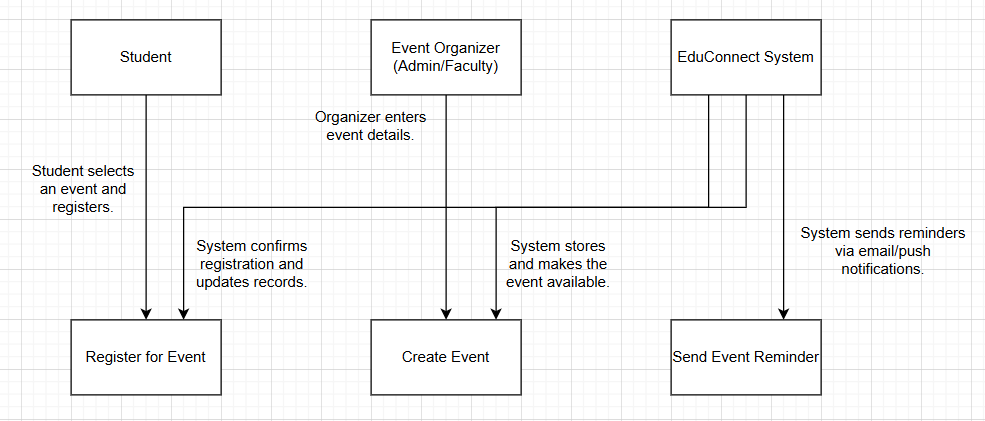
**Use Case 8: Sync with External Calendars**

* **Primary Actor:** Student
* **Goal:** The student syncs their EduConnect events with Google Calendar or Outlook.
* **Preconditions:**
  + The student has an active Google or Outlook account.
  + The student has registered for at least one event.
* **Trigger:** The student opts to sync their events.
* **Main Success Scenario:**
  1. The student logs into EduConnect.
  2. The student navigates to "Settings" and selects "Sync Calendar."
  3. The system prompts the student to choose a calendar service (Google or Outlook).
  4. The student grants permission for EduConnect to access their calendar.
  5. The system syncs upcoming events to the student's selected calendar.
  6. The student receives a confirmation message.
* **Alternative Flows:**
  + **4a:** If permissions are denied, the system notifies the student and cancels the sync.
  + **5a:** If an event update fails to sync, the system retries automatically.

**Use Case 9: Report an Event Issue**

* **Primary Actor:** Student
* **Goal:** The student reports an issue related to an event (e.g., incorrect information, inappropriate content).
* **Preconditions:**
  + The student is logged into EduConnect.
  + The event is active or upcoming.
* **Trigger:** The student notices an issue with an event listing.
* **Main Success Scenario:**
  1. The student navigates to the event page.
  2. The student clicks the "Report Issue" button.
  3. The system prompts the student to select an issue type (e.g., incorrect date, misleading details).
  4. The student submits the report.
  5. The system sends the report to the event organizer and system administrators.
  6. The student receives a confirmation that their report has been submitted.
* **Alternative Flows:**
  + **4a:** If the report submission fails, the student is prompted to try again later.
  + **5a:** If the event has already been canceled, the system notifies the student and closes the report.

**Use Case Diagram**



#### Research On Other Systems

#### 1. Eventbrite

**Functionality:**

* Online event registration and ticketing system.
* Users can create, promote, and administer events.
* Provides analytics for event performance.

**Design Features:**

* Clean and easy-to-use event management dashboard.
* Mobile app for on-the-go event management.
* Custom registration forms to capture attendance information.

**2. Microsoft Teams (Event Feature)**

**Functionality:**

* Event hosting via live sessions and webinars.
* Integrates with Microsoft Calendar and Outlook.
* Enables real-time collaboration and attendee involvement.

**Design Features:**

* Strong virtual event capabilities, including screen sharing and chat options.
* Integrated with Office 365 tools.
* Automated follow-up emails and feedback surveys.

**3. Google Calendar (Event Scheduling and Notification)**

**Functionality:**

* Enables users to create and share events.
* Sends out reminders and notifications.
* Integrates with third-party apps to synchronize events.

**Design Features:**

* Minimalist, intuitive UI for scheduling.
* Auto-sync across devices.
* Smart suggestions for meeting times based on availability.

#### 

#### 1. Google Calendar API

**Purpose:**

* Sync events with users’ Google Calendars.
* Allow users to add, update, and remove events automatically.

**Capabilities:**

* OAuth-based authentication for secure access.
* Event creation with RSVP tracking.
* Time zone handling and reminders.

#### 2. Outlook Calendar API

**Purpose:**

* Sync EduConnect events with Microsoft Outlook Calendar.
* Notify users of event changes.

**Capabilities:**

* User authentication via Microsoft Graph API.
* Read and write access to calendar events.
* Support for recurring event synchronization.

#### 3. Twilio API (SMS Notifications)

**Purpose:**

* Send event reminders via SMS.
* Provide two-factor authentication for account security.

**Capabilities:**

* Send bulk or personalized messages.
* Track message delivery and open rates.

#### 4. Mailgun / SendGrid API (Email Notifications)

**Purpose:**

* Send event registration confirmations and reminders.
* Automate feedback survey distribution.

**Capabilities:**

* High deliverability with tracking and analytics.
* Support for HTML-based emails.
* Batch processing for large-scale email campaigns.

#### 5. Google Maps API

**Purpose:**

* Display event locations on an interactive map.
* Provide directions to event venues.

**Capabilities:**

* Geolocation services and real-time traffic updates.
* Street view and location search integration.

#### 6. Stripe API (Payment Processing for Paid Events)

**Purpose:**

* Handle event ticket payments securely.
* Provide refunds and manage financial transactions.

**Capabilities:**

* Secure transaction processing with fraud detection.
* Multi-currency support.
* Detailed reporting and analytics.