



Palestine Technical University (Kadoorie)

Faculty of Engineering and Technology

Department of Computer Systems Engineering



**Collabrium: INTEGRATED ENTERPRISE MANAGEMENT PLATFORM**

**Prepared By:**

|               |           |
|---------------|-----------|
| Abeer Mahmoud | 202110119 |
| Hanan Issa    | 202110057 |
| Nada Asad     | 202112529 |
| Raghad Hanon  | 202110023 |

**Supervised By:**

Dr. Thaer Sammar

A graduation project submitted in partial fulfillment of the requirements  
for the Bachelor's Degree in Computer Systems Engineering.

Tulkarm, Palestine,

June 2026

# ACKNOWLEDGMENT

In the name of Allah, the Most Gracious, the Most Merciful.

First and foremost, we express our deepest gratitude to Allah Almighty for granting us the strength, patience, and determination required to complete this project. His guidance and blessings supported us throughout every stage of this journey.

We would like to extend our sincere appreciation to our project supervisor, Dr. Thaer Sammar, for his continuous guidance, valuable advice, patience, and unwavering support throughout the development of this project.

We are also deeply grateful to all our lecturers and instructors, whose knowledge and dedication contributed greatly to our academic and professional development.

Special gratitude is extended to Eng. Doa'a Abd-alsamad, Software Engineer at ITG Offices in Nablus, for her valuable contribution to this project. She shared practical insights into professional software development, reviewed and tested several aspects of the system, and provided constructive feedback that helped improve its overall quality.

Finally, we express our heartfelt appreciation to every member of our team. Through dedication, collaboration, and mutual support, we were able to transform our ideas into a complete system and successfully bring this project to completion.

## الإهداء

إلى الوطن الذي علّمنا أن الأحلام الكبيرة تولد من بين التحديات، وأن الأمل شجرة لا تتحني ما دام في الأرض من يرويها حبًا وصبرًا.

إلى القدس، نبض القلوب، وسيدة الحكايات، والمدينة التي كلما ذكر اسمها ازداد فينا الانتماء رسوخًا.

إلى أمهاتنا وأبائنا،

إلى الأيادي التي ارتفعت بالدعاء قبل أن تمتد بالعطاء،

إلى الذين حملوا عنا أثقال الطريق، واحتملوا تعب الأيام ليصنعوا لنا مستقبلًا أجمل،

نهدي هذا الإنجاز، فما نحن إلا غرس من غراسهم الطيب.

إلى إخوتنا وأخواتنا،

الذين كانوا دفء الأيام وسندها، ورفقاء الرحلة في لحظات التعب والفرح.

إلى أساتذتنا ومعلمينا،

الذين أشعلوا في عقولنا نور المعرفة، وفتحوا لنا أبوابًا واسعة نحو العلم والطموح.

وإلى فريقنا،

إلى القلوب التي اجتمعت على حلم واحد، والعقول التي تشاركت الفكرة والجهد والمسؤولية،

إلى من جعلوا من ساعات العمل الطويلة ذكريات جميلة، ومن التحديات جسورًا نعبر بها نحو النجاح.

كنتم شركاء الطريق والإنجاز، وسيبقى هذا العمل شاهدًا على ما تصنعه الإرادة حين تجتمع مع التعاون والإخلاص.

نهديكم هذا الجهد المتواضع،

فهو ليس صفحات وكلمات فحسب، بل حكاية تعب وصبر وأمل، كُتبت سطورها يومًا بعد يوم حتى وصلت إلى هذه اللحظة.

والحمد لله الذي بنعمته تتم الصالحات.

# ABSTRACT

This project presents the design and implementation of Collabrium, an Integrated Enterprise Management Platform developed to address the challenges organizations face when relying on multiple disconnected systems for project management, communication, employee development, and organizational administration. The use of separate tools often leads to fragmented workflows, inefficient collaboration, duplicated information, and limited visibility into organizational activities. To overcome these challenges, Collabrium provides a centralized digital environment that integrates essential enterprise functions within a single platform. The system combines project management, learning and development, internal communication, user and tenant administration, analytics, and AI-assisted services to improve operational efficiency and support organizational growth.

The platform was implemented using a modular architecture composed of interconnected subsystems, including Tenant Management, User and Team Management, Project and Work Item Management, the Learning Hub, the Community Hub, and AI Assistance for intelligent recommendations, data insights, content generation, and reporting.

By integrating these capabilities into a unified ecosystem, Collabrium helps streamline enterprise workflows, enhance collaboration, support continuous learning, improve information flow, and enable more effective management of organizational resources.

## الملخص

يعرض هذا المشروع تصميم وتطوير منصة Collabrium، وهي منصة متكاملة لإدارة المؤسسات، تم تطويرها لمعالجة التحديات التي تواجهها المؤسسات عند الاعتماد على عدة أنظمة منفصلة لإدارة المشاريع، والتواصل، وتطوير الموظفين، والإدارة التنظيمية. إذ يؤدي استخدام الأدوات المتفرقة غالبًا إلى تجزئة سير العمل، وضعف التعاون، وتكرار المعلومات، ومحدودية الرؤية الشاملة للأنشطة المؤسسية.

ولمعالجة هذه التحديات، توفر Collabrium بيئة رقمية مركزية تدمج الوظائف المؤسسية الأساسية ضمن منصة واحدة. ويجمع النظام بين إدارة المشاريع، والتعلم والتطوير، والتواصل الداخلي، وإدارة المستخدمين والمؤسسات، والتحليلات، والخدمات المدعومة بالذكاء الاصطناعي، بهدف تحسين الكفاءة التشغيلية ودعم النمو المؤسسي.

تم تطوير المنصة باستخدام بنية معيارية تتكون من أنظمة فرعية مترابطة، تشمل إدارة المؤسسات، وإدارة المستخدمين والفرق، وإدارة المشاريع وعناصر العمل، ومنصة التعلم، ومنصة المجتمع، ونظام المساعدة الذكية الذي يوفر التوصيات، والرؤى المستندة إلى البيانات، وتوليد المحتوى، ومراجعة التقارير.

ومن خلال دمج هذه الإمكانيات ضمن بيئة موحدة، تسهم Collabrium في تبسيط سير العمل المؤسسي، وتعزيز التعاون، ودعم التعلم المستمر، وتحسين تدفق المعلومات، وتمكين إدارة الموارد التنظيمية بصورة أكثر فاعلية.

# Contents

|                                                        |           |
|--------------------------------------------------------|-----------|
| <b>ACKNOWLEDGMENT</b>                                  | <b>II</b> |
| الإهداء                                                | III       |
| <b>ABSTRACT</b>                                        | <b>IV</b> |
| الملخص                                                 | V         |
| Contents                                               | VI        |
| List of Tables                                         | IX        |
| <b>CHAPTER ONE</b>                                     | <b>1</b>  |
| <b>1.1 Overview</b>                                    | <b>2</b>  |
| <b>1.2 Problem Statement</b>                           | <b>2</b>  |
| <b>1.3 Objectives</b>                                  | <b>3</b>  |
| <b>1.4 Scope of Work</b>                               | <b>4</b>  |
| <b>1.5 Development Methodology</b>                     | <b>4</b>  |
| <b>CHAPTER TWO</b>                                     | <b>6</b>  |
| <b>2.1 Overview</b>                                    | <b>7</b>  |
| 2.2 Existing Enterprise and Collaboration Platforms    | 7         |
| 2.2.1 Jira                                             | 7         |
| 2.2.2 Azure DevOps                                     | 8         |
| 2.2.3 Notion                                           | 9         |
| 2.2.4 Trello                                           | 10        |
| <b>2.3 Comparison</b>                                  | <b>11</b> |
| <b>2.4 Discussion and Identified Gap</b>               | <b>13</b> |
| <b>2.5 Conclusion</b>                                  | <b>13</b> |
| <b>CHAPTER THREE</b>                                   | <b>14</b> |
| <b>3.1 Overview</b>                                    | <b>15</b> |
| <b>3.2 Technologies and Tools</b>                      | <b>15</b> |
| 3.2.1 Backend Technologies                             | 15        |
| 3.2.2 Frontend Technologies                            | 17        |
| 3.2.3 UI/UX Design                                     | 18        |
| 3.2.4 Quality Assurance & Testing Tools                | 18        |
| 3.2.5 Development Environments and Collaboration Tools | 19        |
| <b>CHAPTER FOUR</b>                                    | <b>20</b> |
| <b>4.1 Overview</b>                                    | <b>21</b> |
| 4.2 Functional Requirements                            | 21        |
| 4.2.1 User Authentication & Access Control Subsystem   | 21        |

|                                                     |    |
|-----------------------------------------------------|----|
| 4.2.2 Tenant, User, and Team Management Subsystem   | 22 |
| 4.2.3 Project Management Subsystem                  | 23 |
| 4.2.4 Work Item and Sprint Management Subsystem     | 23 |
| 4.2.5 Bug Tracking & Test Case Management Subsystem | 24 |
| 4.2.6 Learning Hub Subsystem                        | 24 |
| 4.2.7 Community Hub Subsystem                       | 25 |
| 4.2.8 AI Assistance Subsystem                       | 26 |
| 4.2.9 Analytics and Reporting Subsystem             | 26 |
| 4.2.10 Notification and Alert Subsystem             | 27 |
| 4.2.11 Platform Services Subsystem                  | 27 |
| 4.3 Non-Functional Requirements                     | 28 |
| 4.3.1 Security                                      | 28 |
| 4.3.2 Performance                                   | 29 |
| 4.3.3 Scalability                                   | 30 |
| 4.3.4 Usability                                     | 31 |
| 4.3.5 Reliability                                   | 32 |
| 4.3.6 Maintainability                               | 33 |
| 4.3.7 Interoperability                              | 34 |
| 4.3.8 Data Integrity                                | 34 |
| CHAPTER FIVE                                        | 36 |
| 5.1 System Architecture Overview                    | 37 |
| 5.2 Actors                                          | 38 |
| 5.3 Use Case Diagrams                               | 40 |
| 5.3.1 Authentication Subsystem                      | 40 |
| 5.3.2 Platform Services Subsystem                   | 41 |
| 5.3.3 Tenant Management Subsystem                   | 41 |
| 5.3.4 User Management Subsystem                     | 43 |
| 5.3.5 Team Management Subsystem                     | 45 |
| 5.3.6 Project Management Subsystem                  | 46 |
| 5.3.7 Work Item Core Management Subsystem           | 48 |
| 5.3.8 Sprint Management Subsystem                   | 51 |
| 5.3.9 Reports and Analytics Subsystem               | 52 |
| 5.3.10 Learning Hub Subsystem                       | 54 |
| 5.3.11 Community Hub Subsystem                      | 56 |
| 5.3.12 AI Assistant Management Subsystem            | 58 |
| 5.3.13 Notifications and Alerts Subsystem           | 61 |

|                                                                   |     |
|-------------------------------------------------------------------|-----|
| 5.4 Use Case Descriptions                                         | 62  |
| Table 5.4.1: Login Use Case Description                           | 62  |
| Table 5.4.2: View Tenant Profile Use Case Description             | 63  |
| Table 5.4.3: Register New User Use Case Description               | 64  |
| Table 5.4.4: Deactivate User Use Case Description                 | 65  |
| Table 5.4.5: Assign Course to Team Use Case Description           | 66  |
| 5.5 Domain Model Class Diagram                                    | 67  |
| 5.6 Domain Model Class Diagram                                    | 77  |
| <b>5.7 Activity Diagrams</b>                                      | 85  |
| 5.7.1 Authentication Subsystem Activity Diagrams                  | 85  |
| 5.7.2 Platform Services Subsystem Activity Diagrams               | 92  |
| 5.7.3 Tenant Management Subsystem Activity Diagrams               | 93  |
| 5.7.4 User Management Subsystem Activity Diagrams                 | 99  |
| 5.7.5 Team Management Subsystem Activity Diagrams                 | 103 |
| 5.7.6 Project Management Subsystem Activity Diagrams              | 107 |
| 5.7.7 Work Item Core Management Subsystem Activity Diagrams       | 115 |
| 5.7.8 Sprint Management Subsystem Activity Diagrams               | 122 |
| 5.7.9 Reports and Analytics Subsystem Activity Diagrams           | 127 |
| 5.7.10 Learning Hub Subsystem Activity Diagrams                   | 129 |
| 5.7.11 Community Hub Subsystem Activity Diagrams                  | 132 |
| 5.7.12 AI Assistant Management Subsystem Activity Diagrams        | 135 |
| 5.7.13 Notifications and Alerts Subsystem Activity Diagrams       | 140 |
| 5.8 System Sequence Diagrams                                      | 142 |
| 5.8.1 Authentication Subsystem System Sequence Diagrams           | 142 |
| 5.8.2 Platform Services Subsystem System Sequence Diagrams        | 147 |
| 5.8.3 Tenant Management Subsystem System Sequence Diagrams        | 148 |
| 5.8.4 User Management Subsystem System Sequence Diagram           | 153 |
| 5.8.5 Team Management Subsystem System Sequence Diagram           | 157 |
| 5.8.6 Project Management Subsystem System Sequence Diagram        | 161 |
| 5.8.7 Work Item Core Management Subsystem System Sequence Diagram | 171 |
| 5.8.8 Sprint Management Subsystem System Sequence Diagram         | 179 |
| 5.8.9 Reports and Analytics Subsystem Activity Diagrams           | 186 |
| 5.8.10 Learning hub Subsystem System Sequence Diagram             | 187 |
| 5.8.11 Community hub Subsystem System Sequence Diagram            | 203 |
| 5.8.12 AI Subsystem System Sequence Diagram                       | 210 |
| 5.8.13 Notification Subsystem System Sequence Diagram             | 213 |

|                                           |     |
|-------------------------------------------|-----|
| CHAPTER SIX                               | 216 |
| 6.1 Design System                         | 217 |
| 6.1.1 Visual Design and Color Scheme      | 217 |
| 6.2 System Interfaces                     | 218 |
| 6.2.1 Authentication Subsystem Interfaces | 218 |
| 6.2.2 Platform Services Interfaces        | 220 |
| 6.2.3 Tenant Interfaces                   | 221 |
| 6.2.4 Users Interfaces                    | 223 |
| 6.2.5 Teams Interfaces                    | 226 |
| 6.2.6 Projects Interfaces                 | 228 |
| 6.2.7 Analysis Interfaces                 | 237 |
| 6.2.8 Learning Hub Interfaces             | 239 |
| 6.2.9 Community Hub Interfaces            | 247 |
| CHAPTER SEVEN                             | 252 |
| 7.1 System Adoption and Evaluation        | 253 |
| 7.2 Marketing the System                  | 253 |
| 7.3 Subscription and Payment System       | 253 |

## List of Tables

|                                                                 |    |
|-----------------------------------------------------------------|----|
| Table 2.3. 1 Comparison Table .....                             | 12 |
| Table 5.4 1 Login Use Case Description.....                     | 62 |
| Table 5.4 2 View Tenant Profile Use Case Description.....       | 63 |
| Table 5.4 3 Register New User Use Case Description.....         | 64 |
| Table 5.4 4 Deactivate User Use Case Description.....           | 65 |
| Table 5.4 5 Assign Course to Team Use Case Description.....     | 66 |
| Figure 2.2. 1 Jira UI .....                                     | 8  |
| Figure 2.2. 2 Azure DevOps UI .....                             | 9  |
| Figure 2.2. 3 Notion UI .....                                   | 10 |
| Figure 2.2. 4 Trello UI .....                                   | 11 |
| Figure 5.3 1 Authentication Subsystem Use Case Diagram.....     | 40 |
| Figure 5.3 2 Platform Subsystem Use Case Diagram .....          | 41 |
| Figure 5.3 3 Tenant Management Subsystem Use Case Diagram ..... | 42 |
| Figure 5.3 4 User Management Subsystem Use Case Diagram.....    | 44 |

|                                                                                                |    |
|------------------------------------------------------------------------------------------------|----|
| Figure 5.3 5 Team Management Subsystem Use Case Diagram .....                                  | 45 |
| Figure 5.3 6 Project Management Subsystem Use Case Diagram .....                               | 47 |
| Figure 5.3 7 Work Item Core Management Subsystem Use Case Diagram .....                        | 49 |
| Figure 5.3 8 Sprint Management Subsystem Use Case Diagram.....                                 | 51 |
| Figure 5.3 9 Reports and Analytics - Tenant, Project and Sprint Reports Use Case Diagram ..... | 52 |
| Figure 5.3 10 Learning Hub Subsystem Use Case Diagram .....                                    | 55 |
| Figure 5.3 11 Community Hub Subsystem Use Case Diagram .....                                   | 57 |
| Figure 5.3 12 AI Assistance - Work Items Use Case Diagram .....                                | 59 |
| Figure 5.3 13 Notifications and Alerts Subsystem Use Case Diagram .....                        | 61 |
|                                                                                                |    |
| Figure 5.5 1 Collabrium Database ERD Overview .....                                            | 67 |
| Figure 5.5 2 Tenant, Team, Skills and Notification Management ERD .....                        | 68 |
| Figure 5.5 3 Identity and Access Management ERD .....                                          | 69 |
| Figure 5.5 4 Project, Sprint and History Management ERD.....                                   | 70 |
| Figure 5.5 5 Work Item Core and Relationship Management ERD.....                               | 71 |
| Figure 5.5 6 Work Item Collaboration and Supporting Artifacts ERD.....                         | 72 |
| Figure 5.5 7 Learning Courses and User Progress ERD .....                                      | 73 |
| Figure 5.5 8 Learning Roadmaps and Resources ERD .....                                         | 74 |
| Figure 5.5 9 Community Articles and Posts ERD.....                                             | 75 |
| Figure 5.5 10 Events and Announcements ERD.....                                                | 76 |
|                                                                                                |    |
| Figure 5.6 1 Collabrium Domain Model Class Diagram Overview.....                               | 77 |
| Figure 5.6 2 Tenant, Team, Skills and Notification Class Diagram.....                          | 78 |
| Figure 5.6 3 Identity and Access Management Class Diagram .....                                | 79 |
| Figure 5.6 4 Projects, Sprint and History Class Diagram .....                                  | 80 |
| Figure 5.6 5 Work Item Core and Relationship Class Diagram.....                                | 81 |
| Figure 5.6 6 Work Item Collaboration and Supporting Artifacts Class Diagram .....              | 82 |
| Figure 5.6 7 Learning Courses and User Progress Class Diagram.....                             | 82 |
| Figure 5.6 8 Learning Roadmaps and Resources Class Diagram .....                               | 83 |
| Figure 5.6 9 Community Articles and Posts Class Diagram.....                                   | 84 |
| Figure 5.6 10 Events and Announcements Class Diagram .....                                     | 84 |
|                                                                                                |    |
| Figure 5.7.1 1 Login Activity Diagram.....                                                     | 85 |
| Figure 5.7.1 2 Sign Out Activity Diagram .....                                                 | 85 |
| Figure 5.7.1 3 Confirm Email Address Activity Diagram .....                                    | 86 |
| Figure 5.7.1 4 Resend Confirmation Email Activity Diagram.....                                 | 86 |
| Figure 5.7.1 5 Reset Password Activity Diagram .....                                           | 87 |
| Figure 5.7.1 6 Request Password Reset Activity Diagram .....                                   | 87 |
| Figure 5.7.1 7 Change Password Activity Diagram.....                                           | 88 |
| Figure 5.7.1 8 Complete User Setup Activity Diagram .....                                      | 88 |
| Figure 5.7.1 9 Register Tenant Employee Activity Diagram.....                                  | 89 |
| Figure 5.7.1 10 Complete Tenant Setup Activity Diagram.....                                    | 90 |
| Figure 5.7.1 11 Register New Super Admin Activity Diagram .....                                | 91 |
|                                                                                                |    |
| Figure 5.7.2 1 Send Contact Message Activity Diagram .....                                     | 92 |

|                                                                              |     |
|------------------------------------------------------------------------------|-----|
| Figure 5.7.2 2 View Public Content Activity Diagram .....                    | 92  |
| Figure 5.7.3 1 View and Filter Current Tenant Users Activity Diagram .....   | 93  |
| Figure 5.7.3 2 View Current Tenant Profile Activity Diagram .....            | 93  |
| Figure 5.7.3 3 Deactivate Current Tenant Activity Diagram .....              | 94  |
| Figure 5.7.3 4 Upload or Update Tenant Logo Activity Diagram .....           | 94  |
| Figure 5.7.3 5 Update Current Tenant Profile Activity Diagram .....          | 95  |
| Figure 5.7.3 6 View Current Tenant with Admin Details Activity Diagram ..... | 95  |
| Figure 5.7.3 7 View Tenant Details and Analysis Activity Diagram .....       | 96  |
| Figure 5.7.3 8 View and Filter Tenant Users Activity Diagram .....           | 96  |
| Figure 5.7.3 9 Activate or Deactivate Tenant Activity Diagram .....          | 97  |
| Figure 5.7.3 10 Register Tenant with Tenant Admin Activity Diagram .....     | 97  |
| Figure 5.7.3 11 Browse and Filter Tenants Activity Diagram .....             | 98  |
| Figure 5.7.4 1 View User Profile Activity Diagram .....                      | 99  |
| Figure 5.7.4 2 Update My Profile Activity Diagram .....                      | 99  |
| Figure 5.7.4 3 Upload Profile Image Activity Diagram .....                   | 100 |
| Figure 5.7.4 4 View User Skills Activity Diagram .....                       | 100 |
| Figure 5.7.4 5 Manage My Skills Activity Diagram .....                       | 101 |
| Figure 5.7.4 6 View Online Users Activity Diagram .....                      | 101 |
| Figure 5.7.4 7 Update User Role Activity Diagram .....                       | 102 |
| Figure 5.7.5 1 Browse and Filter Teams Activity Diagram .....                | 103 |
| Figure 5.7.5 2 View Team Details Activity Diagram .....                      | 103 |
| Figure 5.7.5 3 View Candidate Team Members Activity Diagram .....            | 104 |
| Figure 5.7.5 4 Create Team Activity Diagram .....                            | 104 |
| Figure 5.7.5 5 Update Team Details Activity Diagram .....                    | 105 |
| Figure 5.7.5 6 Add Team Members Activity Diagram .....                       | 105 |
| Figure 5.7.5 7 Remove Team Members Activity Diagram .....                    | 106 |
| Figure 5.7.5 8 Activate or Deactivate Team Activity Diagram .....            | 106 |
| Figure 5.7.6 1 View My Projects Activity Diagram .....                       | 107 |
| Figure 5.7.6 2 View Project Details Activity Diagram .....                   | 107 |
| Figure 5.7.6 3 View Project Board Activity Diagram .....                     | 108 |
| Figure 5.7.6 4 View Project Backlog Activity Diagram .....                   | 108 |
| Figure 5.7.6 5 View Project Epics Activity Diagram .....                     | 109 |
| Figure 5.7.6 6 View Project User Stories Activity Diagram .....              | 110 |
| Figure 5.7.6 7 View Project History Activity Diagram .....                   | 110 |
| Figure 5.7.6 8 Browse All Projects Activity Diagram .....                    | 111 |
| Figure 5.7.6 9 Create Project Activity Diagram .....                         | 112 |
| Figure 5.7.6 10 Edit Project Activity Diagram .....                          | 113 |
| Figure 5.7.6 11 Delete Project Activity Diagram .....                        | 113 |
| Figure 5.7.6 12 Assign Team to Project Activity Diagram .....                | 114 |
| Figure 5.7.6 13 Unassign Team from Project Activity Diagram .....            | 114 |

|                                                                                        |     |
|----------------------------------------------------------------------------------------|-----|
| Figure 5.7.7 1 Browse and Filter Work Items Activity Diagram.....                      | 115 |
| Figure 5.7.7 2 Create Planning Work Item Activity Diagram .....                        | 115 |
| Figure 5.7.7 3 Update Planning Work Item Activity Diagram .....                        | 116 |
| Figure 5.7.7 4 Create Execution Work Item Activity Diagram .....                       | 117 |
| Figure 5.7.7 5 Update Execution Work Item Activity Diagram .....                       | 118 |
| Figure 5.7.7 6 Manage Work Item Comments Activity Diagram.....                         | 119 |
| Figure 5.7.7 7 Manage Work Item Attachments Activity Diagram .....                     | 119 |
| Figure 5.7.7 8 Manage Work Item Code Links Activity Diagram.....                       | 120 |
| Figure 5.7.7 9 Manage Work Item Relations Activity Diagram .....                       | 121 |
|                                                                                        |     |
| Figure 5.7.8 1 Browse and Filter Sprints .....                                         | 122 |
| Figure 5.7.8 2 Create Sprint Activity Diagram .....                                    | 123 |
| Figure 5.7.8 3 Update Sprint Activity Diagram .....                                    | 123 |
| Figure 5.7.8 4 Add Work Items to Sprint Activity Diagram.....                          | 124 |
| Figure 5.7.8 5 Start Sprint Activity Diagram .....                                     | 124 |
| Figure 5.7.8 6 Complete Sprint Activity Diagram .....                                  | 125 |
| Figure 5.7.8 7 Cancel Sprint Activity Diagram.....                                     | 125 |
| Figure 5.7.8 8 Delete Sprint Activity Diagram .....                                    | 126 |
|                                                                                        |     |
| Figure 5.7.9 1 View or Export Tenant, Project, and Sprint Report Activity Diagram..... | 127 |
| Figure 5.7.9 2 View or Export Report Activity Diagram.....                             | 128 |
|                                                                                        |     |
| Figure 5.7.10 1 Manage Courses Activity Diagram.....                                   | 129 |
| Figure 5.7.10 2 Assign Course Activity Diagram.....                                    | 129 |
| Figure 5.7.10 3 Manage Roadmap Structure Activity Diagram.....                         | 130 |
| Figure 5.7.10 4 AI Roadmap Generation Activity Diagram .....                           | 130 |
| Figure 5.7.10 5 Manage Articles Activity Diagram.....                                  | 131 |
| Figure 5.7.10 6 Manage Article Interaction Activity Diagram .....                      | 131 |
|                                                                                        |     |
| Figure 5.7.11 1 Manage Announcements Activity Diagram.....                             | 132 |
| Figure 5.7.11 2 Manage Events Activity Diagram.....                                    | 132 |
| Figure 5.7.11 3 Event Attendance Activity Diagram.....                                 | 133 |
| Figure 5.7.11 4 Manage Post Activity Diagram.....                                      | 133 |
| Figure 5.7.11 5 Manage Post Interaction Activity Diagram .....                         | 134 |
| Figure 5.7.11 6 View Posts Activity Diagram .....                                      | 134 |
|                                                                                        |     |
| Figure 5.7.12 1 Generate Planning Work Item Field Suggestions Activity Diagram .....   | 135 |
| Figure 5.7.12 2 Generate Execution Work Item Field Suggestions Activity Diagram.....   | 136 |
| Figure 5.7.12 3 AI Roadmap Generation Activity Diagram .....                           | 137 |
| Figure 5.7.12 4 AI Skill Gap and Learning Recommendation Activity Diagram.....         | 138 |
| Figure 5.7.12 5 Generate AI Report Insight Activity Diagram.....                       | 139 |
|                                                                                        |     |
| Figure 5.7.13 1 Browse and Filter My Notifications Activity Diagram .....              | 140 |

|                                                                                     |     |
|-------------------------------------------------------------------------------------|-----|
| Figure 5.7.13 2 View Unread Notifications Count Activity Diagram .....              | 140 |
| Figure 5.7.13 3 Receive Real - Time Notifications Activity Diagram.....             | 140 |
| Figure 5.7.13 4 Receive Unread Count Updates Activity Diagram .....                 | 141 |
| Figure 5.7.13 5 Mark Notification Read Activity Diagram .....                       | 141 |
| Figure 5.7.13 6 Mark All Notifications as Read Activity Diagram .....               | 141 |
|                                                                                     |     |
| Figure 5.8.1 1 Login System Sequence Diagram.....                                   | 142 |
| Figure 5.8.1 2 Confirm Email Address Sequence Diagram .....                         | 143 |
| Figure 5.8.1 3 Resend Confirmation Email Sequence Diagram .....                     | 143 |
| Figure 5.8.1 4 Change Password System Sequence Diagram .....                        | 144 |
| Figure 5.8.1 5 Reset Password System Sequence Diagram .....                         | 144 |
| Figure 5.8.1 6 Complete Tenant Setup System Sequence Diagram.....                   | 145 |
| Figure 5.8.1 7 Register New Super Admin Sequence Diagram.....                       | 146 |
| Figure 5.8.1 8 Register Tenant Employee Sequence Diagram.....                       | 146 |
|                                                                                     |     |
| Figure 5.8.2 1 Send Contact Massage System Sequence Diagram .....                   | 147 |
| Figure 5.8.2 2 View Public Content Sequence Diagram.....                            | 147 |
|                                                                                     |     |
| Figure 5.8.3 1 View and Filter Current Tenant User System Sequence Diagram.....     | 148 |
| Figure 5.8.3 2 View Current Tenant Profile System Sequence Diagram.....             | 148 |
| Figure 5.8.3 3 Deactivate Current Tenant Activity System Sequence Diagram .....     | 149 |
| Figure 5.8.3 4 Upload or Update Tenant Logo System Sequence Diagram .....           | 149 |
| Figure 5.8.3 5 Update Current Tenant Profile System Sequence Diagram.....           | 150 |
| Figure 5.8.3 6 View Current Tenant with Admin Details System Sequence Diagram ..... | 150 |
| Figure 5.8.3 7 View Tenant Details and Analysis System Sequence Diagram.....        | 151 |
| Figure 5.8.3 8 Browse and Filter Tenants System Sequence Diagram .....              | 151 |
| Figure 5.8.3 9 Activate or Deactivate System Sequence Diagram.....                  | 152 |
| Figure 5.8.3 10 Register Tenant with Tenant Admin System Sequence Diagram.....      | 152 |
|                                                                                     |     |
| Figure 5.8.4 1 View User Profile System Sequence Diagram .....                      | 153 |
| Figure 5.8.4 2 Update My Profile System Sequence Diagram .....                      | 153 |
| Figure 5.8.4 3 Upload Profile Image System Sequence Diagram.....                    | 154 |
| Figure 5.8.4 4 View User Skill System Sequence Diagram.....                         | 154 |
| Figure 5.8.4 5 Manage My Skill System Sequence Diagram .....                        | 155 |
| Figure 5.8.4 6 View Online Users System Sequence Diagram.....                       | 156 |
| Figure 5.8.4 7 Update User Role System Sequence Diagram.....                        | 156 |
|                                                                                     |     |
| Figure 5.8.5 1 Browse and Filter Teams System Sequence Diagram .....                | 157 |
| Figure 5.8.5 2 View Team Details System Sequence Diagram .....                      | 157 |
| Figure 5.8.5 3 View Candidate Team Member System Sequence Diagram .....             | 158 |
| Figure 5.8.5 4 Create Team System Sequence Diagram.....                             | 158 |
| Figure 5.8.5 5 Update Team Details System Sequence Diagram.....                     | 159 |
| Figure 5.8.5 6 Add Team Members System Sequence Diagram .....                       | 159 |
| Figure 5.8.5 7 Remove Team Member System Sequence Diagram.....                      | 160 |

|                                                                                        |     |
|----------------------------------------------------------------------------------------|-----|
| Figure 5.8.5 8 Activate or Deactivate Team System Sequence Diagram .....               | 160 |
| Figure 5.8.6 1 View My Projects System Sequence Diagram .....                          | 161 |
| Figure 5.8.6 2 View Project Detail System Sequence Diagram.....                        | 161 |
| Figure 5.8.6 3 View Project Board System Sequence Diagram.....                         | 162 |
| Figure 5.8.6 4 View Project Backlog System Sequence Diagram .....                      | 163 |
| Figure 5.8.6 5 View Project Epic System Sequence Diagram .....                         | 164 |
| Figure 5.8.6 6 View Project User Stories System Sequence Diagram .....                 | 165 |
| Figure 5.8.6 7 View Project History System Sequence Diagram.....                       | 166 |
| Figure 5.8.6 8 Browse All Projects History System Sequence Diagram .....               | 167 |
| Figure 5.8.6 9 Edit Project Details System Sequence Diagram .....                      | 168 |
| Figure 5.8.6 10 Delete Selected Project System Sequence Diagram .....                  | 169 |
| Figure 5.8.6 11 Select A Team for A Project System Sequence Diagram.....               | 169 |
| Figure 5.8.6 12 Remove the Assigned Team from A Project System Sequence Diagram .....  | 170 |
| Figure 5.8.7 1 Browse and Filter Work Items System Sequence Diagram.....               | 171 |
| Figure 5.8.7 2 Create Planning Work Item System Sequence Diagram .....                 | 172 |
| Figure 5.8.7 3 Update Planning Work Item System Sequence Diagram .....                 | 172 |
| Figure 5.8.7 4 Create Execution Work Item System Sequence Diagram .....                | 173 |
| Figure 5.8.7 5 Update Execution Work Item System Sequence Diagram .....                | 174 |
| Figure 5.8.7 6 Manage Work item Attachments System Sequence Diagram .....              | 175 |
| Figure 5.8.7 7 Manage Work item Comment System Sequence Diagram.....                   | 176 |
| Figure 5.8.7 8 Manage Work item Code Links System Sequence Diagram.....                | 177 |
| Figure 5.8.7 9 Manage Work item Relations System Sequence Diagram .....                | 178 |
| Figure 5.8.9 1 View or Export Tenant, Project, and Sprint Report Sequence Diagram..... | 186 |
| Figure 5.8.9 2 View or Export Report Sequence Diagram .....                            | 186 |
| Figure 5.8.10 1 Create Course System Sequence Diagram.....                             | 187 |
| Figure 5.8.10 2 Get Courses System Sequence Diagram.....                               | 187 |
| Figure 5.8.10 3 Get Course by id System Sequence Diagram .....                         | 188 |
| Figure 5.8.10 4 Upload Course Trumbull System Sequence Diagram .....                   | 188 |
| Figure 5.8.10 5 Update Course System Sequence Diagram.....                             | 189 |
| Figure 5.8.10 6 Delete Course System Sequence Diagram.....                             | 189 |
| Figure 5.8.10 7 Assign Course to User Sequence Diagram .....                           | 190 |
| Figure 5.8.10 8 Assign Course to Team System Sequence Diagram .....                    | 190 |
| Figure 5.8.10 9 Start Course System Sequence Diagram.....                              | 191 |
| Figure 5.8.10 10 Complete Course System Sequence Diagram.....                          | 191 |
| Figure 5.8.10 11 Save Course System Sequence Diagram .....                             | 192 |
| Figure 5.8.10 12 Un Save Course System Sequence Diagram .....                          | 192 |
| Figure 5.8.10 13 Rate Course System Sequence Diagram.....                              | 193 |
| Figure 5.8.10 14 Manage Roadmap System Sequence Diagram .....                          | 194 |
| Figure 5.8.10 15 Manage Roadmap Structure System Sequence Diagram.....                 | 194 |
| Figure 5.8.10 16 Manage Subtopics and Resource System Sequence Diagram .....           | 195 |
| Figure 5.8.10 17 Manage Roadmap Tags Sequence Diagram .....                            | 195 |

|                                                                                          |     |
|------------------------------------------------------------------------------------------|-----|
| Figure 5.8.10 18 Get Roadmaps System Sequence Diagram.....                               | 196 |
| Figure 5.8.10 19 Get Roadmap Tag System Sequence Diagram .....                           | 196 |
| Figure 5.8.10 20 Create Article System Sequence Diagram .....                            | 197 |
| Figure 5.8.10 21 Update Article System Sequence Diagram.....                             | 197 |
| Figure 5.8.10 22 Delete Article System Sequence Diagram .....                            | 198 |
| Figure 5.8.10 23 Get Articles System Sequence Diagram .....                              | 198 |
| Figure 5.8.10 24 Get Article Details System Sequence Diagram .....                       | 199 |
| Figure 5.8.10 25 Upload Article Cover Page System Sequence Diagram .....                 | 199 |
| Figure 5.8.10 26 Add Article Comment System Sequence Diagram.....                        | 200 |
| Figure 5.8.10 27 Get Article Comments System Sequence Diagram .....                      | 200 |
| Figure 5.8.10 28 Delete Article Comments System Sequence Diagram.....                    | 201 |
| Figure 5.8.10 29 Like Article System Sequence Diagram .....                              | 201 |
| Figure 5.8.10 30 Unlike Article System Sequence Diagram.....                             | 202 |
|                                                                                          |     |
| Figure 5.8.11 1 Manage Post System Sequence Diagram.....                                 | 203 |
| Figure 5.8.11 2 Manage Post Tag System Sequence Diagram.....                             | 203 |
| Figure 5.8.11 3 Upload Post Images Tag System Sequence Diagram .....                     | 204 |
| Figure 5.8.11 4 Get Posts System Sequence Diagram .....                                  | 204 |
| Figure 5.8.11 5 Get Post Details System Sequence Diagram.....                            | 205 |
| Figure 5.8.11 6 Like Post System Sequence Diagram .....                                  | 205 |
| Figure 5.8.11 7 Unlike Post System Sequence Diagram.....                                 | 206 |
| Figure 5.8.11 8 Manage Event System Sequence Diagram .....                               | 206 |
| Figure 5.8.11 9 Attend Event System Sequence Diagram .....                               | 207 |
| Figure 5.8.11 10 Get Events System Sequence Diagram.....                                 | 207 |
| Figure 5.8.11 11 Manage Announcement System Sequence Diagram .....                       | 208 |
| Figure 5.8.11 12 Manage Announcement Audience System Sequence Diagram .....              | 208 |
| Figure 5.8.11 13 Get Announcement System Sequence Diagram .....                          | 209 |
|                                                                                          |     |
| Figure 5.8.12 1 Generate AI Report System Sequence Diagram .....                         | 210 |
| Figure 5.8.12 2 Generate Execution Work Item Field System Sequence Diagram .....         | 211 |
| Figure 5.8.12 3 Generate Planning Work Item Field System Sequence Diagram .....          | 211 |
| Figure 5.8.12 4 Generate Roadmap Draft Using AI Field System Sequence Diagram .....      | 212 |
| Figure 5.8.12 5 Generate Roadmap Suggestions Using AI Field System Sequence Diagram..... | 212 |
|                                                                                          |     |
| Figure 5.8.13 1 Browse and Filter My Notifications System Sequence Diagram.....          | 213 |
| Figure 5.8.13 2 View Unread Notifications Count System Sequence Diagram .....            | 213 |
| Figure 5.8.13 3 Mark Notification as Read System Sequence Diagram .....                  | 214 |
| Figure 5.8.13 4 Mark All Notification As Read System Sequence Diagram.....               | 214 |
| Figure 5.8.13 5 View Unread Count Notifications Update System Sequence Diagram .....     | 215 |
| Figure 5.8.13 6 Receive Real-Time Notifications System Sequence Diagram.....             | 215 |
|                                                                                          |     |
| Figure 6.2.1 1 Login Interface.....                                                      | 218 |
| Figure 6.2.1 2 Forgot Password Interface .....                                           | 218 |
| Figure 6.2.1 3 Email Confirm Message Interface .....                                     | 219 |
| Figure 6.2.1 4 Complete Tenant Setup Interface.....                                      | 219 |

|                                                                              |     |
|------------------------------------------------------------------------------|-----|
| Figure 6.2.1 5 Reset Password Interface .....                                | 220 |
| Figure 6.2.2 1 Send Contact Interface .....                                  | 220 |
| Figure 6.2.2 2 Notifications Dialog .....                                    | 220 |
| Figure 6.2.2 3 View Online Users Dialog .....                                | 221 |
| Figure 6.2.3 1 Super Admin- Tenants Table .....                              | 221 |
| Figure 6.2.3 2 Super Admin- Add Tenant Dialog .....                          | 222 |
| Figure 6.2.3 3 Super Admin- Tenant Details Drawer .....                      | 222 |
| Figure 6.2.3 4 Super Admin- Tenant Profile Interface .....                   | 222 |
| Figure 6.2.3 5 Super Admin- Activate & Deactivate Tenant Dialog .....        | 223 |
| Figure 6.2.4 1 Super Admin- Add New Super Admin Dialog .....                 | 223 |
| Figure 6.2.4 2 Tenant Manager- Add New Tenant Member Dialog .....            | 223 |
| Figure 6.2.4 3 Tenant Users Table .....                                      | 224 |
| Figure 6.2.4 4 Change User Role Dialog .....                                 | 224 |
| Figure 6.2.4 5 User Details Drawer .....                                     | 224 |
| Figure 6.2.4 6 User Profile Interface .....                                  | 225 |
| Figure 6.2.4 7 Edit Profile Interface .....                                  | 225 |
| Figure 6.2.4 8 View User Skills .....                                        | 225 |
| Figure 6.2.5 1 Tenant Main Dashboard .....                                   | 221 |
| Figure 6.2.5 2 Teams Board .....                                             | 226 |
| Figure 6.2.5 3 Tenant Manager- Create & Edit Team Dialog .....               | 226 |
| Figure 6.2.5 4 Team Details Page .....                                       | 226 |
| Figure 6.2.5 5 Tenant Manager- Add Team Member Dialog .....                  | 227 |
| Figure 6.2.5 6 Tenant Manager- Remove Team Member Dialog .....               | 227 |
| Figure 6.2.5 7 Tenant Manager- Activate & Deactivate Team Dialog .....       | 227 |
| Figure 6.2.6 1 User Projects Board - Card View .....                         | 228 |
| Figure 6.2.6 2 User Projects Board- Table View .....                         | 228 |
| Figure 6.2.6 3 Tenant Manager- Create Project Dialog .....                   | 228 |
| Figure 6.2.6 4 Tenant Manager- Confirm Reassign Team To Project .....        | 229 |
| Figure 6.2.6 5 Project History Dialog .....                                  | 229 |
| Figure 6.2.6 6 Project Backlog Board .....                                   | 229 |
| Figure 6.2.6 7 Project Epics & Related Work Items Board .....                | 230 |
| Figure 6.2.6 8 Project Story Board .....                                     | 230 |
| Figure 6.2.6 9 View Work Item Details Drawer .....                           | 230 |
| Figure 6.2.6 10 Assign Work Item Dialog .....                                | 231 |
| Figure 6.2.6 11 Product Team Manager- Create Planning Work Item Dialog ..... | 231 |
| Figure 6.2.6 12 Product Team Member- Create Execution Work Item Dialog ..... | 231 |
| Figure 6.2.6 13 Manage Work Item Comments Interface .....                    | 232 |
| Figure 6.2.6 14 Manage Work Item Attachments Interface .....                 | 232 |
| Figure 6.2.6 15 Manage Code Links Interface .....                            | 233 |

|                                                                              |     |
|------------------------------------------------------------------------------|-----|
| Figure 6.2.6 16 Manage Work Item Relations Dialog .....                      | 233 |
| Figure 6.2.6 17 Sprints Interface .....                                      | 233 |
| Figure 6.2.6 18 Sprint Details Interface .....                               | 234 |
| Figure 6.2.6 19 Start Sprint Dialog .....                                    | 234 |
| Figure 6.2.6 20 Edit Sprint Dialog .....                                     | 234 |
| Figure 6.2.6 21 Complete Sprint Dialog .....                                 | 235 |
| Figure 6.2.6 22 Cancel Sprint Dialog .....                                   | 235 |
| Figure 6.2.6 23 Delete Sprint Dialog .....                                   | 236 |
|                                                                              |     |
| Figure 6.2.7 1 Create Work Item Using AI .....                               | 232 |
| Figure 6.2.7 2 Tenant Information Report .....                               | 237 |
| Figure 6.2.7 3 Sprint Progress Report .....                                  | 237 |
| Figure 6.2.7 4 Developer Productivity Report .....                           | 237 |
| Figure 6.2.7 5 Project Health Report .....                                   | 238 |
| Figure 6.2.7 6 Teams Productivity Report .....                               | 238 |
| Figure 6.2.7 7 AI Generated Report .....                                     | 238 |
|                                                                              |     |
| Figure 6.2.8 1 Tenant Member- Roadmap Page .....                             | 239 |
| Figure 6.2.8 2 Tenant Manager- Create Roadmap Dialog .....                   | 239 |
| Figure 6.2.8 3 Tenant Manager- Create Roadmap Using AI Dialog .....          | 239 |
| Figure 6.2.8 4 Tenant Manager- Create Roadmap Node Using AI Dialog .....     | 240 |
| Figure 6.2.8 5 Tenant Manager- Create Roadmap Subtopic Using AI Dialog ..... | 240 |
| Figure 6.2.8 6 Tenant Manager- Create Roadmap Resource Using AI Dialog ..... | 240 |
| Figure 6.2.8 7 View Roadmap Node Details Drawer .....                        | 241 |
| Figure 6.2.8 8 Roadmap View Page .....                                       | 241 |
| Figure 6.2.8 9 Roadmap Builder Page .....                                    | 241 |
| Figure 6.2.8 10 Manage Roadmap Tags Dialog .....                             | 242 |
| Figure 6.2.8 11 Courses Page .....                                           | 242 |
| Figure 6.2.8 12 Tenant Manager- Create & Edit Course Dialog .....            | 242 |
| Figure 6.2.8 13 Tenant Manager- Assign Course To Team/ Member Dialog .....   | 243 |
| Figure 6.2.8 14 Tenant Manager- Change Course Thumbnail Dialog .....         | 243 |
| Figure 6.2.8 15 Manage Roadmap Tags Dialog .....                             | 243 |
| Figure 6.2.8 16 Submit Course Rating Interface .....                         | 244 |
| Figure 6.2.8 17 Tenant Manager- Assignments Page .....                       | 244 |
| Figure 6.2.8 18 Tenant Member- Assign to Me Page .....                       | 244 |
| Figure 6.2.8 19 Tenant Member- Saved Courses Interface .....                 | 245 |
| Figure 6.2.8 20 Tenant Member- My Courses Interface .....                    | 245 |
| Figure 6.2.8 21 Tenant Manager- Articles Page .....                          | 245 |
| Figure 6.2.8 22 Tenant Member- Articles Page .....                           | 246 |
| Figure 6.2.8 23 Tenant Manager- Create Article Page .....                    | 246 |
| Figure 6.2.8 24 Detailed Article Page .....                                  | 246 |
| Figure 6.2.8 25 Tenant Manager- Edit Article Interface .....                 | 247 |
| Figure 6.2.8 26 Article Comments Section .....                               | 247 |
|                                                                              |     |
| Figure 6.2.9 1 Community Feed Interface .....                                | 247 |

|                                                                          |     |
|--------------------------------------------------------------------------|-----|
| Figure 6.2.9 2 Create & Edit Post Dialog .....                           | 248 |
| Figure 6.2.9 3 Post Details Dialog .....                                 | 248 |
| Figure 6.2.9 4 Events Interface .....                                    | 248 |
| Figure 6.2.9 5 Create & Edit Event Dialog .....                          | 249 |
| Figure 6.2.9 6 Change Event Banner Dialog.....                           | 249 |
| Figure 6.2.9 7 Cancel Event Dialog .....                                 | 249 |
| Figure 6.2.9 8 Detailed Event Drawer.....                                | 250 |
| Figure 6.2.9 9 Tenant Member- Announcements Interface .....              | 250 |
| Figure 6.2.9 10 Announcement Details Dialog.....                         | 250 |
| Figure 6.2.9 11 Tenant Manager- Announcements Dashboard Card View .....  | 251 |
| Figure 6.2.9 12 Tenant Manager- Announcements Dashboard Table View ..... | 251 |
| Figure 6.2.9 13 Create & Edit Announcement Dialog .....                  | 251 |

## CHAPTER ONE

### **INTRODUCTION**

This chapter introduces the Collabrium platform and presents the background of the project, the problem it addresses, its main objectives, scope of work, and the development methodology followed throughout its implementation.

### 1.1 Overview

In today's digital environment, organizations depend on various systems to manage projects, track tasks, monitor performance, support employee development, and facilitate internal communication. Although each system serves a specific purpose, using multiple disconnected tools often results in fragmented workflows, duplicated information, reduced efficiency, and communication gaps between teams and departments.

To address these challenges, this project presents Collabrium, an Integrated Enterprise Management Platform designed to centralize essential organizational operations within a unified digital environment. The platform combines tenant, user, and team management with agile project and work item management, analytics, learning activities, community interaction, and notification services.

Collabrium also incorporates AI-assisted capabilities that provide personalized recommendations, intelligent insights, content generation, and reporting support. By integrating these functions into one web-based platform, the system aims to improve collaboration, streamline workflows, support continuous learning, and enhance coordination across organizational departments.

### 1.2 Problem Statement

Modern organizations often rely on multiple digital tools for project tracking, task management, communication, employee development, and performance monitoring. Although these tools support specific functions, they frequently operate as separate systems, resulting in fragmented workflows, duplicated information, inconsistent data, and limited visibility across departments.

This fragmentation makes it more difficult for managers to monitor progress, coordinate teams, maintain accountability, and access a unified view of organizational activities. Employees may also experience delays, communication gaps, and difficulty locating relevant information or learning resources when these services are distributed across disconnected platforms.

To address this problem, Collabrium was designed and implemented as an Integrated Enterprise Management Platform that centralizes essential organizational functions within a single environment. The platform aims to improve workflow coordination, information accessibility, collaboration, and operational visibility across teams and departments.

## 1.3 Objectives

The main objective of this project is to design and implement Collabrium, an integrated enterprise management platform that centralizes organizational operations and improves collaboration, visibility, and efficiency across different departments.

The specific objectives are:

- To provide a unified digital environment that reduces reliance on fragmented systems and disconnected tools.
- To implement a multi-tenant architecture that enables multiple organizations to use the platform while maintaining secure data isolation.
- To provide role-based access control that manages user permissions according to assigned roles and responsibilities.
- To support agile project management through projects, sprints, backlogs, work items, boards, and progress tracking.
- To facilitate user, team, and organizational management within each tenant.
- To support employee learning and development through courses, roadmaps, articles, skills, assignments, and progress tracking.
- To enhance internal collaboration through community posts, discussions, announcements, events, and real-time notifications.
- To integrate AI-assisted capabilities that provide recommendations, suggestions, analytical insights, content generation, and reporting support.

- To provide dashboards and reports that support monitoring and data-driven decision-making.
- To develop a modular and maintainable system that can support future expansion and additional enterprise services.

## **1.4 Scope of Work**

The scope of Collabrium covers the main organizational activities performed by authorized employees and administrators within organizations that adopt the platform. Each organization operates as an independent tenant with isolated users, teams, projects, and data, while access to system features is controlled according to assigned roles and permissions.

The platform includes tenant, user, and team management; agile project and work item management; sprint and backlog tracking; learning and skill development; community interaction; notifications; analytics; reporting; and AI-assisted services. Users can perform different operations according to their responsibilities, including managing projects and work items, accessing learning content, participating in community activities, and reviewing relevant reports and insights.

By integrating these functions within a unified digital environment, Collabrium supports centralized information management, structured collaboration, improved workflow visibility, and more efficient coordination across organizational teams and departments.

## **1.5 Development Methodology**

The project was developed using the Agile software development methodology, which follows an iterative and incremental approach. The development process was divided into two-week sprints, with each sprint focusing on a defined set of system features and improvements.

The project began with requirements analysis and system planning, followed by UI/UX design, system modeling, backend and frontend implementation, integration, and testing. At the end of each

sprint, the completed work was reviewed, tested, and refined based on feedback and newly identified requirements.

This approach supported continuous development, reduced implementation risks, improved adaptability to changes, and enabled steady progress toward the completion of the system.

## CHAPTER TWO

### **Literature Review**

This chapter reviews existing project management, collaboration, communication, and enterprise productivity platforms that are related to Collabrium. It examines their main capabilities, strengths, and limitations, and compares them with the proposed platform to identify the functional gaps addressed by this project.

## 2.1 Overview

This chapter reviews several widely used platforms related to project management, collaboration, communication, and knowledge sharing, including Jira, Azure DevOps, Notion, and Trello. Each platform is examined in terms of its main features, strengths, limitations, and suitability for organizational use.

A comparative analysis is then presented to identify the functional gaps that remain when organizations rely on separate specialized tools. This analysis provides the basis for positioning Collabrium as an integrated platform that combines project management, learning, communication, analytics, and AI-assisted capabilities within a unified enterprise environment.

## 2.2 Existing Enterprise and Collaboration Platforms

This section reviews several widely used platforms that support project management, software development, collaboration, communication, and knowledge sharing. Each platform is examined based on its core capabilities, strengths, limitations, and relevance to the functional scope of Collabrium.

### 2.2.1 Jira

Jira is a project and work management platform developed by Atlassian. It is widely used by software development teams to plan, organize, and track work, particularly within Agile methodologies such as Scrum and Kanban. The platform provides configurable boards, workflows, backlogs, sprints, dashboards, and reporting capabilities that help teams monitor project progress and manage development activities.

Jira supports different work item types, including Epics, User Stories, Tasks, and Bugs. Teams can prioritize backlog items, assign work to members, estimate effort, plan sprints, and track work

through customizable workflow stages. It also supports integrations and marketplace applications that extend its capabilities across development, testing, documentation, and collaboration processes. Despite its strong project management capabilities, Jira primarily focuses on planning and tracking work. Broader enterprise functions, such as employee learning management, organizational community engagement, integrated skill development, and tenant-level administration, generally depend on additional products, applications, or integrations. In comparison, Collabrium combines project management with learning, community, analytics, user administration, and AI-assisted services within a unified enterprise platform. [1]

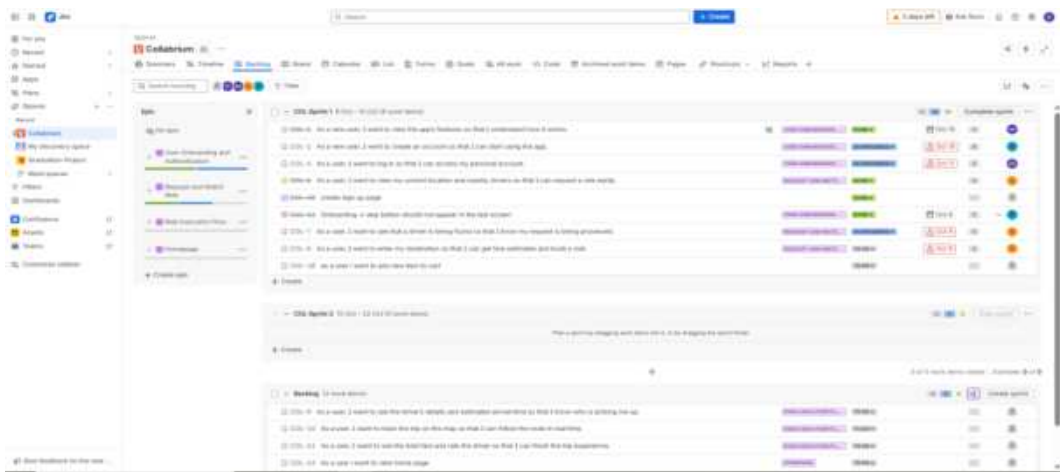


Figure 2.2. 1 Jira UI

### 2.2.2 Azure DevOps

Azure DevOps is a collection of software development services provided by Microsoft to support project planning, source code management, testing, and deployment. Its main services include Azure Boards, Azure Repos, Azure Pipelines, Azure Test Plans, and Azure Artifacts.

Azure Boards supports Agile workflows through configurable boards, backlogs, sprints, dashboards, and hierarchical work items such as Epics, Features, User Stories, Tasks, and Bugs.

Although Azure DevOps provides comprehensive support for the software development lifecycle, its primary focus remains on technical development and delivery processes. Collabrium extends beyond this scope by integrating project management with organizational learning, community interaction, analytics, administration, and AI-assisted services within a unified platform. [2]

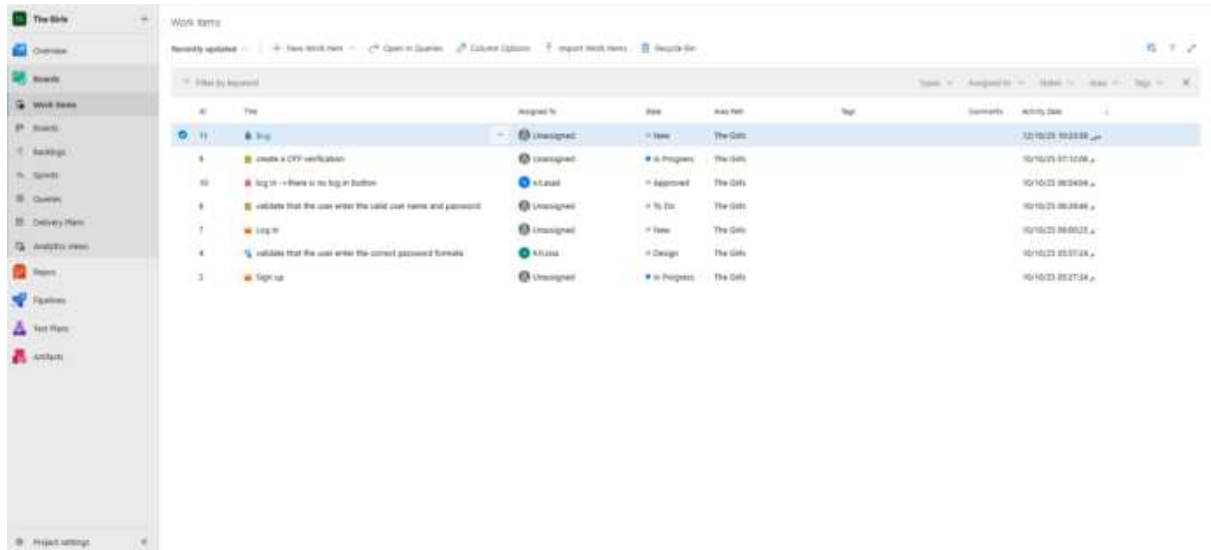


Figure 2.2. 2 Azure DevOps UI

### 2.2.3 Notion

Notion is a flexible workspace platform that combines documentation, wikis, databases, project tracking, and team collaboration. It enables users to organize information, create customized pages, manage projects and tasks, and visualize work through views such as boards, calendars, and timelines.

The platform also provides AI-assisted capabilities for content generation, summarization, information retrieval, and workflow automation. Its flexible structure allows teams to design workflows according to their needs.

However, Notion is primarily a customizable productivity and knowledge-management workspace rather than a specialized enterprise management system. Collabrium provides more structured support for multi-tenant administration, role-based project workflows, employee learning, skills analysis, community interaction, and organizational analytics within a unified platform. [3]

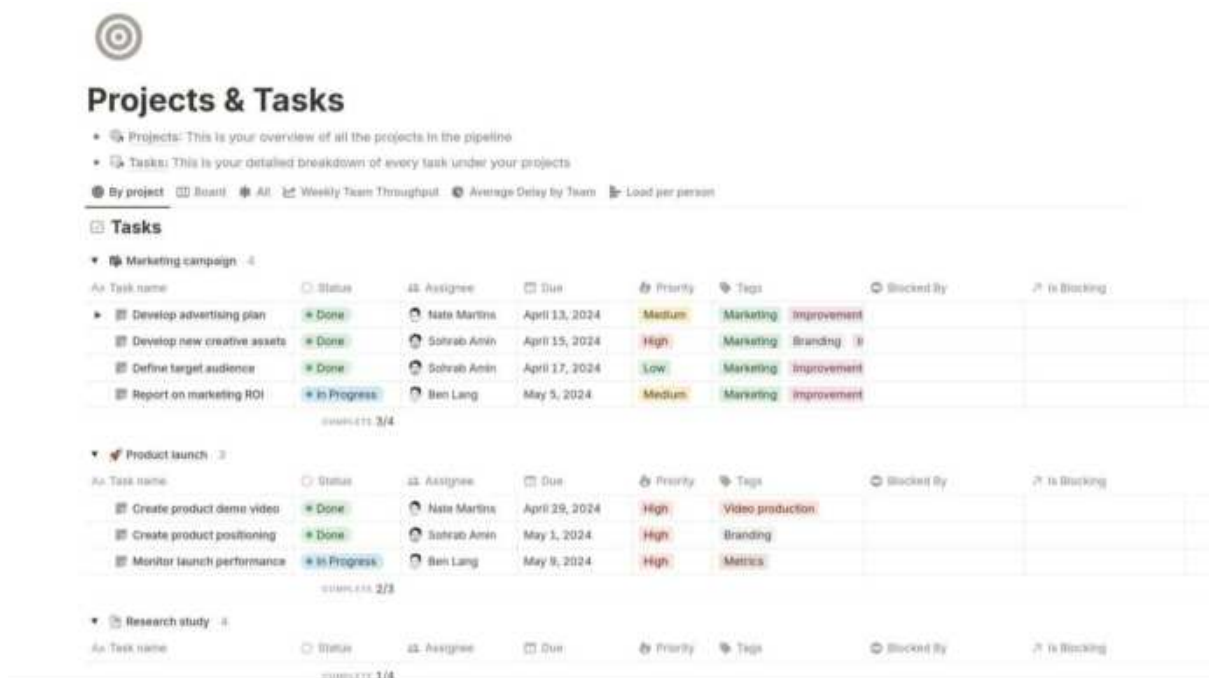


Figure 2.2. 3 Notion UI

## 2.2.4 Trello

Trello is a visual project management platform developed by Atlassian. It organizes work through boards, lists, and cards, allowing teams to represent tasks and move them between workflow stages such as “To Do,” “In Progress,” and “Done.”

The platform supports task assignment, due dates, checklists, custom fields, workflow automation, and integrations through Power-Ups. Its simple visual structure makes it suitable for individuals and teams seeking an accessible way to organize and monitor work.

However, Trello mainly focuses on visual task and workflow management. Collabrium provides a broader enterprise environment by integrating structured agile project management with tenant administration, employee learning, community interaction, analytics, and AI-assisted services. [4]

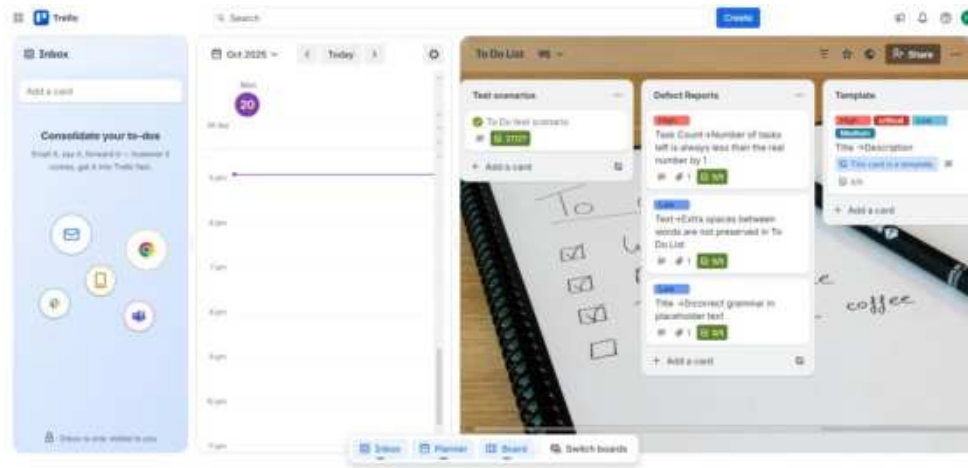


Figure 2.2. 4 Trello UI

## 2.3 Comparison

Table 2.1 compares the reviewed platforms according to their primary functions and their support for project management, communication, learning, administration, analytics, and AI-assisted capabilities. The comparison highlights how each platform specializes in particular organizational activities, while Collabrium integrates these functions within a unified enterprise environment.

| Feature Category               | Jira                        | Azure DevOps                       | Notion                             | Trello                                                   | Collabrium (Proposed)                                                         |
|--------------------------------|-----------------------------|------------------------------------|------------------------------------|----------------------------------------------------------|-------------------------------------------------------------------------------|
| Primary Focus                  | Project and work management | Software development lifecycle     | Knowledge and workspace management | Visual task management                                   | Integrated enterprise management                                              |
| Agile Project Management       | Advanced                    | Advanced                           | General and customizable           | Basic to moderate                                        | Integrated                                                                    |
| Work Item Tracking             | Detailed and configurable   | Detailed and lifecycle-oriented    | Customizable databases             | Cards and lists                                          | Hierarchical and centralized                                                  |
| Communication and Community    | Comments and integrations   | Basic collaboration                | Pages and team collaboration       | Basic reporting                                          | Posts, comments, events, announcements, and notifications                     |
| Learning and Skill Development | Not a primary function      | Not a primary function             | Knowledge content support          | Basic reporting                                          | Integrated Learning Hub                                                       |
| Multi-Tenant Administration    | Not a primary function      | Organization-based administration  | Workspace-based                    | Basic reporting                                          | Integrated tenant isolation and administration                                |
| Analytics and Reporting        | Project-focused             | Development-focused                | General workspace views            | Basic reporting                                          | Project, team, user, and organizational analytics                             |
| AI-Assisted Capabilities       | Available                   | Available across selected services | Available                          | Available through platform capabilities and integrations | Integrated recommendations, suggestions, skills analysis, and report insights |
| Overall Scope                  | Project management          | Development lifecycle management   | Flexible productivity workspace    | Visual workflow management                               | Unified management, learning, community, analytics, and AI assistance         |

Table 2.3. 1 Comparison Table

## 2.4 Discussion and Identified Gap

---

The reviewed platforms provide strong capabilities within their primary areas of specialization. Jira and Azure DevOps offer advanced support for project and software lifecycle management, Notion provides flexible knowledge and workspace organization, Trello simplifies visual task tracking.

However, organizations that require project management, employee learning, internal community interaction, analytics, and administrative control may still need to rely on several separate platforms.

This separation can increase workflow fragmentation, duplicate organizational information, and limit the ability to obtain a unified view of users, teams, projects, and learning activities.

Collabrium addresses this gap by integrating these functions within a multi-tenant enterprise platform. It combines structured project and work item management with learning and skill development, community engagement, user and team administration, analytics, reporting, and AI-assisted capabilities. This integration provides a centralized environment that supports coordination, information accessibility, and organizational decision-making.

## 2.5 Conclusion

---

This chapter reviewed several widely used platforms for project management, software development, knowledge organization, task tracking, and workplace communication. The analysis showed that each platform provides strong capabilities within a specific area; however, organizations may still need to combine multiple tools to manage projects, learning, communication, analytics, and administrative activities.

Collabrium addresses this limitation by integrating these functions within a unified multi-tenant platform. By combining project and work item management, learning and skill development, community interaction, analytics, administration, and AI-assisted services, the platform provides a more centralized approach to managing organizational operations.

## CHAPTER THREE

### **DEVELOPMENT METHODOLOGY AND TECHNOLOGIES**

This chapter presents the development methodology followed throughout the project and describes the main technologies, frameworks, tools, and development environments used to design and implement the Collabrium platform.

### **3.1 Overview**

Collabrium was developed using the Agile software development methodology, which follows an iterative and incremental approach. The project was divided into two-week sprints, with each sprint focusing on a defined set of requirements, system features, and improvements.

The development process included requirements analysis, system modeling, UI/UX design, backend and frontend implementation, integration, and testing. At the end of each sprint, the completed work was reviewed and refined based on testing results, team discussions, and newly identified requirements.

This methodology supported continuous development, improved adaptability to changes, reduced implementation risks, and enabled steady progress throughout the project lifecycle.

### **3.2 Technologies and Tools**

Collabrium was implemented using a set of modern technologies, frameworks, libraries, and development tools selected to support scalability, maintainability, security, real-time communication, and an efficient user experience. These technologies were organized according to their role in backend development, frontend development, user interface design, testing, documentation, and collaborative development.

#### **3.2.1 Backend Technologies**

The backend of Collabrium was developed using C# and .NET 8 with ASP.NET Core Web API. The system follows Clean Architecture to separate the domain, application, infrastructure, and presentation layers, improving maintainability, testability, and modularity. [8] [9]

The main backend technologies and libraries include:

- **SQL Server:** Used as the primary relational database for storing organizational, project, user, learning, and community data.
- **Entity Framework Core:** Used for object-relational mapping, database access, and schema management through Code-First migrations.
- **MediatR and CQRS:** Used to separate commands and queries and organize application operations into independent request handlers.
- **ASP.NET Core Identity and JWT:** Used for user management, authentication, token-based access, and role- and policy-based authorization. [10]
- **FluentValidation:** Used to validate incoming requests before executing business operations.
- **AutoMapper:** Used to map domain and persistence models to Data Transfer Objects.
- **SignalR:** Used to provide real-time notifications and live system updates.
- **Cloudinary:** Used for storing and managing uploaded media and attachments.
- **QuestPDF:** Used to generate downloadable reports in PDF format.
- **SMTP Email Service:** Used to send account, verification, password recovery, and system-related emails.
- **Serilog:** Used for structured logging and monitoring application errors and events.
- **Swagger/OpenAPI:** Used to document and test the available API endpoints.
- **Ollama with Gemma 3:** Used to run the AI model locally and provide recommendations, suggestions, analytical insights, and report-related assistance.

### 3.2.2 Frontend Technologies

The frontend of Collabrium was developed using React 19, TypeScript, and Vite. React was used to build reusable and interactive user interface components, while TypeScript improved code reliability through static type checking. Vite provided a fast development environment and an optimized production build process. [11]

The main frontend technologies and libraries include:

- **Material UI and Tailwind CSS:** Used to create responsive, consistent, and reusable interface components and layouts. [12]
- **TanStack Query:** Used to retrieve, cache, synchronize, and update server-side data efficiently.
- **React Custom Hooks:** Used to encapsulate reusable interface logic and reduce code duplication across components.
- **React Router:** Used to manage navigation and route protection across the platform.
- **Axios:** Used to communicate with backend API endpoints and handle HTTP requests and responses.
- **React Hook Form and Zod:** Used to manage form state and validate user input.
- **Microsoft SignalR Client:** Used to receive real-time notifications and live system updates from the backend.
- **DnD Kit:** Used to implement drag-and-drop interactions within work item boards and backlog interfaces.
- **Recharts and ECharts:** Used to present analytics, reports, and performance data through charts and visual dashboards.
- **React Toastify:** Used to display success, warning, and error notifications to users.
- **React Dropzone and React Cropper:** Used to support file uploading and image selection or cropping.
- **Date-fns:** Used to format, compare, and manage date and time values.

- **ESLint:** Used to enforce consistent coding standards and identify potential frontend code issues.

### 3.2.3 UI/UX Design

The user interface and user experience of Collabrium were designed using Figma to define the platform's structure, navigation, visual identity, and interaction patterns before implementation.

The main UI/UX design tools and practices include:

- **Design System:** Used to standardize colors, typography, spacing, icons, and component behavior across the platform.
- **User-Centered Navigation:** Designed to provide consistent and accessible navigation across different modules and user roles.
- **Blender and Figma:** Used, where required, to create and prepare graphical assets for the platform.

### 3.2.4 Quality Assurance & Testing Tools

Several tools and testing practices were used throughout the development process to verify system functionality, identify implementation issues, and maintain the quality of both backend and frontend components.

The main quality assurance and testing tools include:

- **Postman:** Used to test API endpoints, request parameters, authentication flows, and response results.
- **Swagger/OpenAPI:** Used to inspect, document, and interactively test backend API endpoints. [13]
- **Chrome DevTools:** Used to debug frontend components, inspect network requests, monitor browser errors, and evaluate responsive layouts.

- **SQL Server Management Studio:** Used to inspect database records, verify relationships, and validate the results of database operations.
- **Manual Functional Testing:** Used to validate system workflows, role permissions, form behavior, navigation, and integration between frontend and backend components.

### 3.2.5 Development Environments and Collaboration Tools

Several development environments and collaboration tools were used to support implementation, version control, task organization, and coordination among team members.

The main tools include:

- **Visual Studio:** Used for backend development, debugging, and managing the ASP.NET Core solution.
- **Visual Studio Code:** Used primarily for frontend development with React and TypeScript.
- **Git:** Used for tracking source code changes and managing different development branches.
- **GitHub:** Used as the shared source code repository and for coordinating code integration among team members.
- **Figma:** Used to collaborate on interface designs, prototypes, and reusable design components.

## CHAPTER FOUR

# SYSTEM REQUIREMENTS

This chapter presents the functional and non-functional requirements of the Collabrium platform. The functional requirements describe the services and operations provided by each subsystem, while the non-functional requirements define the quality attributes related to security, performance, scalability, usability, reliability, maintainability, interoperability, and data integrity.

### 4.1 Overview

The system requirements define the expected behavior, capabilities, and quality standards of the Collabrium platform. They were identified based on the needs of organizations, administrators, managers, and employees who interact with the system through different roles and permissions.

The requirements are divided into functional and non-functional categories. Functional requirements describe the operations supported by each subsystem, while non-functional requirements specify the technical and quality constraints that guide the system's design, implementation, and operation.

### 4.2 Functional Requirements

The functional requirements define the operations, services, and interactions supported by the Collabrium platform. They are organized according to the main system subsystems, including authentication and access control, tenant and user management, project and work item management, learning, community interaction, notifications, analytics, reporting, and AI-assisted services.

#### 4.2.1 User Authentication & Access Control Subsystem

- The system shall allow registered users to securely sign in using their email address and password.
- The system shall authenticate users using JSON Web Tokens (JWT) and refresh tokens.
- The system shall support email confirmation, forgotten-password recovery, password reset, and password change operations.
- The system shall require newly registered users to change their temporary password during their first access to the platform.

- The system shall restrict access to suspended, deactivated, or unauthorized user accounts.
- The system shall enforce role- and policy-based authorization according to each user's assigned responsibilities.
- The system shall ensure that users can access only the data and operations associated with their tenant.
- The system shall allow authorized administrators to create, activate, deactivate, and manage user accounts.
- The system shall protect secured API endpoints and real-time connections from unauthenticated access.

#### 4.2.2 Tenant, User, and Team Management Subsystem

- The system shall allow the Super Admin to register, view, update, activate, and deactivate tenants.
- The system shall maintain logical data isolation between tenants to prevent unauthorized cross-organization access.
- The system shall allow authorized tenant administrators to view and update their organization's information.
- The system shall allow authorized users to register, view, update, activate, suspend, and deactivate employee accounts within their tenant.
- The system shall allow administrators to assign and update user roles according to organizational responsibilities.
- The system shall provide searchable and filterable dashboards for managing users and tenants.
- The system shall allow authorized users to create, view, update, activate, and deactivate teams.
- The system shall support assigning team leaders and managing team members.

- The system shall ensure that users, teams, and related records remain associated with the correct tenant.
- The system shall notify affected users when important account, role, or team-related changes occur.

#### 4.2.3 Project Management Subsystem

- The system shall allow authorized users to create and manage projects within their tenant.
- The system shall allow users to define project information, including the project name, key, description, start date, and end date.
- The system shall allow authorized users to view and update project details.
- The system shall support assigning a project to an appropriate organizational team.
- The system shall restrict project access according to the user's role, tenant, and project permissions.
- The system shall provide searchable and filterable project views to help users locate and manage projects.
- The system shall allow authorized users to monitor project status and overall progress.
- The system shall record important project changes to maintain a traceable project history.
- The system shall notify relevant users when a project is assigned or when significant project information changes.
- The system shall provide access to project-related dashboards, analytics, and reports for authorized users.

#### 4.2.4 Work Item and Sprint Management Subsystem

- The system shall support multiple work item types, including Epics, User Stories, Tasks, Bugs, and Test Cases.
- The system shall allow authorized users to create, view, update, delete, and prioritize work items within a project.
- The system shall support hierarchical relationships between Epics, User Stories, and their related child work items.

- The system shall allow users to link related work items and maintain traceability between them.
- The system shall support assigning work items to eligible project members.
- The system shall allow users to update work item status, priority, requirements, acceptance criteria, and other relevant details.
- The system shall provide backlog, story board, task board, and sprint views to support Agile workflows.
- The system shall allow authorized users to create, schedule, start, update, complete, and monitor sprints.
- The system shall support managing sprint scope by adding or removing eligible work items.
- The system shall allow users to add comments, code links and attachments to work items.
- The system shall record work item changes and activities to provide a traceable history.
- The system shall notify relevant users when work items are assigned or when important details are changed.
- The system shall provide sprint and work item analytics to support progress monitoring and reporting.

#### 4.2.5 Bug Tracking & Test Case Management Subsystem

- The system allows users to report bugs with priority, and reproduction steps.
- The system allows assigning bugs to responsible team members.
- The system allows tracking bug status.
- The system supports linking bugs to user stories, tasks, and test cases.
- The system allows managing test cases and linking them to system requirements.

#### 4.2.6 Learning Hub Subsystem

- The system shall provide learning content through Courses, Roadmaps, and Articles.
- The system shall allow authorized users to create, view, update, publish, and manage learning materials.

- The system shall support organizing learning content using skills, categories, and tags.
- The system shall allow users to browse, search, filter, and access available learning materials.
- The system shall allow authorized managers to assign courses to individual users or teams.
- The system shall track user progress, completion status, and learning activity.
- The system shall allow users to save learning materials for later access.
- The system shall support course ratings and feedback.
- The system shall allow users to interact with articles through likes and comments.
- The system shall provide personalized learning recommendations based on user skills, identified skill gaps, and assigned responsibilities.
- The system shall allow users and authorized managers to review learning progress and assignment details.

#### 4.2.7 Community Hub Subsystem

- The system shall allow users to create, view, update, and delete community posts according to their permissions.
- The system shall support organizing posts using categories and tags.
- The system shall allow users to search and filter community content.
- The system shall allow users to interact with posts through comments and likes.
- The system shall allow users to view organizational announcements.
- The system shall allow authorized administrators to create, update, publish, and manage announcements.
- The system shall allow users to view scheduled organizational events and their details.
- The system shall allow users to register for or cancel their attendance at available events.
- The system shall allow authorized administrators to create, update, and manage events.
- The system shall provide notifications for relevant community activities, announcements, and events.

#### 4.2.8 AI Assistance Subsystem

- The system shall provide AI-assisted recommendations for learning materials based on user skills and identified skill gaps.
- The system shall support AI-assisted suggestions when creating work items.
- The system shall generate analytical insights from project, sprint, team, user, and organizational report data.
- The system shall support employee skill analysis and provide role-related matching suggestions.
- The system shall recommend relevant learning resources while authorized users create or develop learning roadmaps.
- The system shall allow users to review AI-generated outputs before applying or relying on them.

#### 4.2.9 Analytics and Reporting Subsystem

- The system shall provide dashboards for monitoring project, sprint, team, and user performance.
- The system shall calculate and display key metrics related to project progress, work item distribution, velocity, cycle time, productivity, and completion rates.
- The system shall provide reports for blocked, overdue, and completed work items.
- The system shall support analysis of bug distribution according to severity and status.
- The system shall provide Learning Hub reports covering courses, roadmaps, articles, learning assignments, user progress, completion rates, and ratings.
- The system shall provide Community Hub reports covering posts, comments, announcements, events, attendance, and user engagement activities.
- The system shall allow authorized users to filter analytics and reports by project, sprint, team, user, content type, and time period.
- The system shall present analytical data through charts, summaries, and detailed report views.

- The system shall allow authorized users to generate and download reports in PDF format.
- The system shall support AI-generated insights based on the selected report data.
- The system shall restrict access to analytics and reports according to tenant, role, and assigned permissions.

#### 4.2.10 Notification and Alert Subsystem

- The system shall provide real-time notifications using SignalR.
- The system shall notify users about relevant project, sprint, work item, team, role, learning, and community activities.
- The system shall support notifications related to account creation, role changes, and important administrative actions.
- The system shall provide email notifications for selected account and system events, including email confirmation and password recovery.
- The system shall allow users to view their notifications and distinguish between read and unread items.
- The system shall allow users to mark individual notifications or all notifications as read.
- The system shall provide direct navigation from a notification to the related system page or resource.
- The system shall ensure that notifications are delivered only to authorized users within the correct tenant.

#### 4.2.11 Platform Services Subsystem

- The system shall allow users to view and update their personal profiles.
- The system shall allow authorized users to view the profiles of other users within their tenant.
- The system shall display relevant professional, organizational, role and skill within user profiles.
- The system shall allow users to submit contact, inquiry, or support requests through the platform.

- The system shall provide shared navigation and access to platform-wide services according to each user's role and permissions.
- The system shall ensure that profile information and platform services remain restricted to authorized users within the appropriate tenant.

## 4.3 Non-Functional Requirements

The non-functional requirements define the quality attributes and technical constraints that guide the design, implementation, and operation of Collabrium. These requirements address security, performance, scalability, usability, reliability, maintainability, interoperability, and data integrity to ensure that the platform operates efficiently, securely, and consistently.

### 4.3.1 Security

The system shall protect organizational and user data from unauthorized access and ensure that users can perform only the operations permitted by their assigned roles, policies, and tenant membership.

Implementation Measures:

- ASP.NET Core Identity is used for secure user, password, and role management.
- Passwords are stored using secure hashing mechanisms rather than plain text.
- JWT access tokens and refresh tokens are used to authenticate API requests and maintain secure user sessions.
- Role- and policy-based authorization is applied to protect restricted operations and resources.
- Tenant validation and data filtering are enforced to prevent unauthorized access across organizations.
- Secured API endpoints and SignalR connections require authenticated access.
- HTTPS/TLS is used to protect data transmitted between the client and the server.
- Request validation is applied before executing business operations.

- Sensitive internal entities are not exposed directly through API responses; Data Transfer Objects are used instead.
- Structured logging is used to record application errors and unauthorized access attempts without intentionally exposing sensitive credentials.

#### 4.3.2 Performance

The system shall process user requests efficiently and maintain responsive interaction while handling concurrent operations across its different modules.

Implementation Measures:

- ASP.NET Core Web API and asynchronous programming are used to process requests without unnecessary blocking.
- Cancellation tokens are passed through asynchronous operations to stop processing requests that are no longer required.
- Entity Framework Core queries are optimized using techniques such as AsNoTracking for read-only operations.
- Query projections and read models are used to retrieve only the data required by each operation instead of loading complete entities.
- Unnecessary related data and database relationships are excluded from queries to reduce processing and data transfer.
- Pagination and cursor-based retrieval are applied when loading large collections.
- Maximum page-size limits are applied to prevent excessive records from being returned in a single request.
- Database queries are filtered and sorted at the server level to reduce unnecessary client-side processing.
- TanStack Query is used on the frontend to cache server data and minimize repeated API requests.

- SignalR is used for real-time notifications and updates, reducing the need for frequent client-side polling.
- React and Vite are used to support efficient interface rendering and optimized production builds.
- Analytics and reports are filtered according to selected data and time periods to avoid processing unnecessary data.
- Cloudinary is used to store and deliver media files efficiently, reducing the load placed on the application server.

#### 4.3.3 Scalability

The system shall support future growth in the number of tenants, users, teams, projects, and system modules without requiring major changes to its core architecture.

Implementation Measures:

- Clean Architecture is used to separate the domain, application, infrastructure, and presentation layers, allowing each layer to evolve independently.
- The system is divided into modular subsystems, including tenant management, user management, project management, learning, community, notifications, analytics, and AI assistance.
- Multi-tenant architecture enables multiple organizations to use the platform while maintaining logically isolated data and operations.
- Dependency injection and interface-based design are used to reduce coupling and simplify the replacement or extension of services.
- CQRS and MediatR are used to organize commands and queries into independent handlers, making new operations easier to add and maintain.
- Stateless API authentication using JWT supports future deployment across multiple application instances.
- Pagination and filtered data retrieval reduce the impact of increasing data volumes.

- Reusable frontend components and modular feature organization support the addition of new interfaces and system functions.
- External services, such as media storage, email delivery, and AI processing, are integrated through abstractions that allow future replacement or expansion.
- Command-side repositories and query-side read models are separated, allowing write and read operations to evolve independently as system requirements and data volumes increase.
- Query read models use direct projections to return only the required DTO fields instead of retrieving complete entities and unnecessary related data.
- Domain models are separated from persistence entities, keeping business rules independent from database structures and allowing the persistence implementation to evolve without affecting the core domain logic.

#### 4.3.4 Usability

The system shall provide a clear, consistent, and user-friendly interface that enables users with different roles and technical backgrounds to access and perform their tasks efficiently.

Implementation Measures:

- A consistent design system is applied across the platform, including standardized colors, typography, spacing, icons, and interface components.
- Navigation menus and available actions are adapted according to the user's role and permissions.
- Responsive layouts are used to maintain usability across different screen sizes and devices.
- Forms provide clear labels, validation messages, and feedback to help users' complete operations correctly.
- Success, warning, and error messages are displayed to inform users about the results of their actions.

- Search, filtering, sorting, and pagination features are provided to simplify access to large collections of data.
- Dashboards, charts, boards, and summary cards are used to present complex information in an understandable visual format.
- Reusable interaction patterns are maintained across projects, work items, learning, community, and administration modules.
- Descriptive labels, readable content, and accessibility-related attributes are used to improve interface clarity and compatibility with assistive technologies.

#### 4.3.5 Reliability

The system shall maintain consistent operation, preserve data accuracy, and handle failures without exposing incomplete or inconsistent results to users.

Implementation Measures:

- SQL Server constraints, primary keys, foreign keys, and defined relationships are used to preserve database consistency.
- Entity Framework Core migrations are used to manage database schema changes in a controlled and traceable manner.
- The Unit of Work pattern is used to commit related database changes as a single operation.
- Input validation is performed before business operations are executed, reducing invalid or incomplete data.
- Domain rules are applied to prevent invalid state transitions and inconsistent business operations.
- A global exception-handling mechanism is used to process unexpected errors and return consistent API responses.
- Structured logging is used to record application failures and support troubleshooting.
- Asynchronous operations use cancellation tokens to stop abandoned requests and reduce unnecessary processing.

- User-facing feedback is provided when an operation fails, allowing users to understand the result and retry when appropriate.

#### 4.3.6 Maintainability

The system shall be organized in a modular and understandable manner that supports debugging, modification, and the addition of new features without introducing unnecessary changes across unrelated components.

Implementation Measures:

- Clean Architecture is used to separate the Domain, Application, Infrastructure, and Presentation layers, ensuring that each layer has a clearly defined responsibility.
- Domain models encapsulate business rules, behavior, and valid state transitions independently of database implementation details.
- Persistence entities are maintained within the infrastructure layer to represent database fields, keys, constraints, and relationships.
- Mapping components are used to transfer data between domain models and persistence entities, preventing database concerns from affecting the core business logic.
- Command-side repositories are used to load and persist domain models during write operations.
- Dedicated read models are used for query operations and return purpose-specific DTOs containing only the data required by each use case.
- CQRS and MediatR are used to organize commands and queries into independent handlers, reducing coupling between application operations.
- Dependency injection and interface-based abstractions are used to simplify service replacement, testing, and future extension.
- FluentValidation is used to separate request validation rules from business operation handlers.
- Reusable frontend components, custom hooks, and feature-based organization reduce code duplication and simplify interface maintenance.

- Structured logging and centralized exception handling support error diagnosis and troubleshooting.
- Swagger/OpenAPI documentation is maintained to clarify backend endpoints, request structures, and response contracts.
- Git and GitHub are used to track code changes, manage branches, and coordinate integration among team members.

#### 4.3.7 Interoperability

The system shall support reliable communication between its internal components and external services through standardized interfaces and data formats.

Implementation Measures:

- RESTful API endpoints are used to enable communication between the frontend and backend components.
- JSON is used as the primary data exchange format for API requests and responses.
- Swagger/OpenAPI is used to document API contracts and facilitate integration and testing.
- SignalR provides a standardized real-time communication channel between the server and connected clients.
- External services, including Cloudinary, SMTP email delivery, and Ollama-based AI processing, are integrated through dedicated service abstractions.
- Interface-based integration reduces direct dependency on specific third-party providers and supports future replacement or extension.
- Data Transfer Objects are used to define clear contracts between system layers and client applications.
- Validation and consistent response structures are applied to improve reliability when exchanging data between components.

#### 4.3.8 Data Integrity

The system shall ensure that stored and processed data remains accurate, valid, consistent, and associated with the correct organizational context throughout its lifecycle.

#### Implementation Measures:

- FluentValidation is used to validate incoming requests before they reach the core business logic.
- Domain models enforce business rules and prevent invalid state changes or unsupported operations.
- SQL Server primary keys, foreign keys, required fields, unique constraints, and defined relationships are used to preserve referential integrity.
- The Unit of Work pattern is used to commit related database operations consistently within a single transaction.
- Tenant identifiers are validated and applied to ensure that records remain associated with the correct organization.
- Entity Framework Core mappings define database relationships, constraints, and field requirements in a centralized manner.
- Command handlers verify the existence and validity of related entities before creating or updating records.
- Controlled status transitions are applied to projects, sprints, work items, and other state-based entities.
- Date ranges and dependent values are validated to prevent logically inconsistent records.
- Project and work item history records are maintained to provide traceability for important changes.
- Data Transfer Objects restrict the data accepted from or returned to clients, reducing unintended modifications to internal entities.

## CHAPTER FIVE

### System Design and Architecture

This chapter presents the design and architecture of the Collabrium platform. It describes the system actors, architectural structure, major subsystems, and the interactions between users and system functions. It also presents the use case diagrams, use case descriptions, domain model, activity diagrams, and system sequence diagrams used to represent the system's behavior and internal organization.

## 5.1 System Architecture Overview

Collabrium was designed as a modular, multi-tenant web platform composed of a React-based frontend, an ASP.NET Core Web API backend, a SQL Server database, and several supporting services for real-time communication, media storage, email delivery, report generation, and AI-assisted processing. Each organization operates as an independent tenant, and tenant-related validation is applied to maintain logical isolation between organizational data.

The backend follows Clean Architecture and is divided into Domain, Application, Infrastructure, and Presentation layers. The Domain layer contains business models, rules, and valid state transitions, while the Application layer defines use cases through commands, queries, handlers, interfaces, and DTOs. The infrastructure layer implements database access, persistence entities, repositories, read models, external integrations, and supporting services. The Presentation layer exposes secured RESTful API endpoints and SignalR hubs to the frontend.

The system applies CQRS to separate write and read responsibilities. Command-side repositories load and persist domain models during business operations, while query-side read models retrieve only the data required by each use case and return purpose-specific DTOs. Authentication is implemented using ASP.NET Core Identity and JWT, while role-, policy-, project-, and tenant-based authorization control access to system resources. SignalR supports real-time notifications, and the AI Assistance subsystem communicates with a locally hosted model through Ollama to generate recommendations, suggestions, and analytical insights.

## 5.2 Actors

The actors of Collabrium represent the different user roles that interact with the platform. Each actor is granted access to specific system functions according to assigned roles, authorization policies, tenant membership, team association, and project responsibilities.

1. User:

A general authenticated actor who can access shared platform services according to assigned permissions. Users can manage their profiles, view notifications, access authorized learning and community content, and use other platform features available to their roles.

2. Super Admin:

The highest administrative actor at the platform level. The Super Admin is responsible for registering, viewing, updating, activating, and deactivating tenants, monitoring platform-level information, and managing global tenant-related operations.

3. Tenant Admin:

An administrative actor responsible for managing a specific organization. The Tenant Admin manages tenant information, users, roles, teams, learning content, announcements, events, and other tenant-level services according to granted permissions.

4. Product Team Manager:

An actor responsible for overseeing product teams and monitoring project, team, and employee performance. This actor can review dashboards, analytics, reports, learning assignments, and project progress to support coordination and decision-making.

5. Product Owner:

An actor responsible for defining product priorities and managing the product backlog. The Product Owner can prioritize work items, manage sprint planning activities, and ensure that project work remains aligned with business goals.

6. Team Lead:

An actor responsible for coordinating the daily work of a project team. The Team Lead

assigns work items, monitors sprint progress, manages team execution, reviews completed work, and supports team members during project delivery.

7. Business Analyst:

An actor responsible for analyzing and documenting business requirements. The Business Analyst defines Epics and User Stories, refines backlog items, documents acceptance criteria, and ensures that work items reflect identified business needs.

8. Developer:

A product team member responsible for implementing assigned work items. Developers can view and update their tasks, manage work item progress, add comments and attachments, collaborate with team members, and access relevant project and learning resources.

9. Quality Assurance Engineer:

A product team member responsible for validating system quality. The QA Engineer manages test-related work items, reports and tracks bugs, documents testing information, and verifies that implemented features satisfy defined requirements and acceptance criteria.

10. Human Resources User:

An actor responsible for monitoring employee-related information, skills, learning progress, and performance indicators. The HR user can access authorized workforce analytics, reports, skill-related information, and employee development data.

11. IT User:

An actor responsible for selected technical and account-management operations within the organization. Depending on assigned permissions, the IT user can register employee accounts, support access-related activities, and assist with technical user-management processes.

## 5.3 Use Case Diagrams

Use case diagrams provide a visual representation of the interactions between users (Actors) and the sub systems. These diagrams help in understanding the functional requirements and how different users will interact with Collabrium system.

### 5.3.1 Authentication Subsystem

The **Authentication Subsystem** is responsible for managing secure access to the platform for all types of users. It ensures that users can **log in** safely, verify their identity, and recover access when needed through features like **login**, **email confirmation**, and **password reset**. It also enforces security policies such as **forced password changes** and allows users to update their credentials. Additionally, it supports tenant-specific onboarding by enabling the Tenant Admin to **complete the tenant setup** process securely.

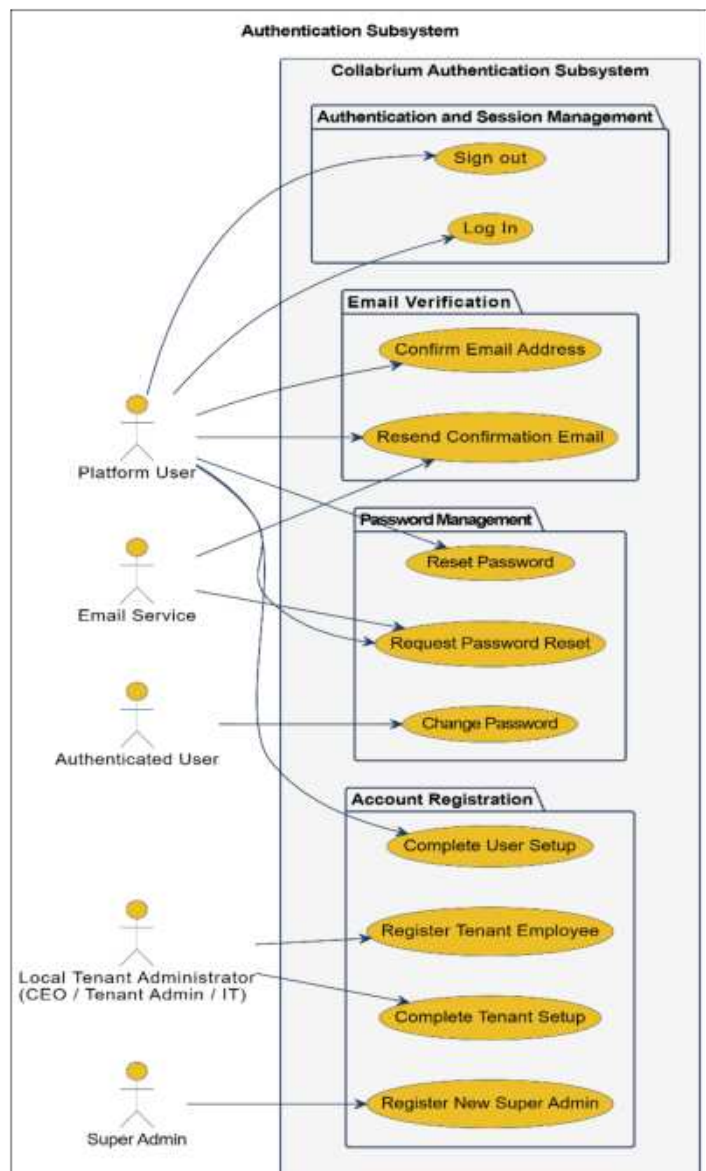


Figure 5.3 1 Authentication Subsystem Use Case Diagram

### 5.3.2 Platform Services Subsystem

The **Platform Services Subsystem** provides essential platform-wide services that enhance user experience and interaction across the system. It allows users to **send contact Message**, **view public content**.

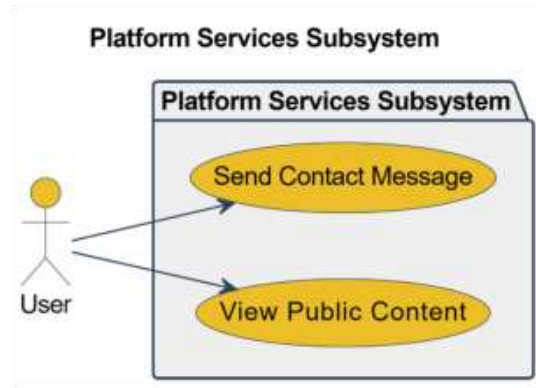


*Figure 5.3 2 Platform Subsystem Use Case Diagram*

### 5.3.3 Tenant Management Subsystem

The **Tenant Management Subsystem** is responsible for managing tenant information and administration across the platform. It allows tenant members to view their current tenant profile and access tenant user information through filtering and browsing features.

For authorized managers, including CEOs, Tenant Administrators, and IT personnel, the subsystem provides tools to view and update tenant details, manage tenant branding by uploading or updating logos, and oversee tenant-related information. Tenant Administrators can also deactivate their current tenant when required.

At the platform level, Super Administrators can register new tenants, browse and analyze tenant information, manage tenant users, and control tenant availability through activation or deactivation functions. The subsystem also integrates with the Media Storage Service to support tenant logo management and related media operations.

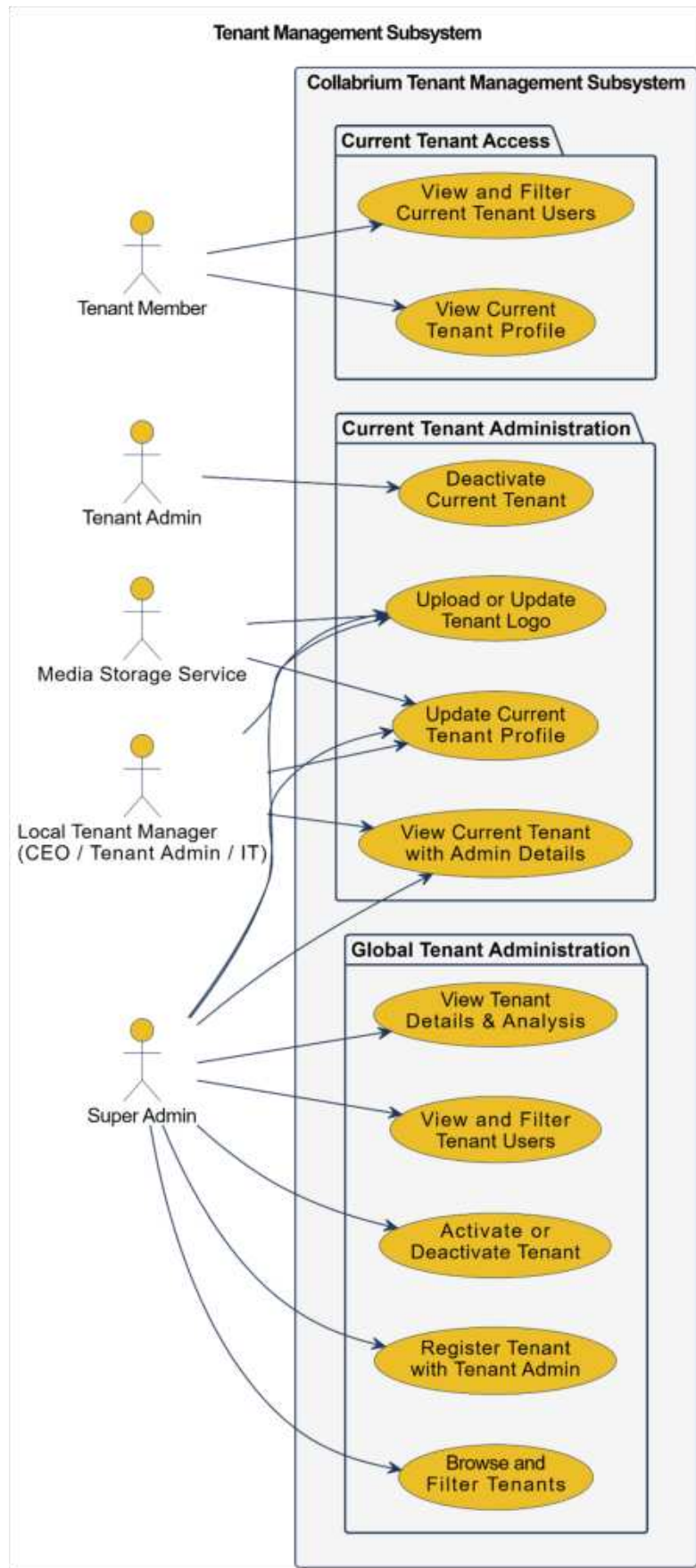


Figure 5.3 3 Tenant Management Subsystem Use Case Diagram

#### 5.3.4 User Management Subsystem

The **User Management Subsystem** is responsible for managing user information, skills, and role-related activities within the platform. It enables tenant members to access and maintain their personal profiles by viewing profile details, updating profile information, and uploading profile images. The subsystem also supports skills management by allowing users to view their own skills, view the skills of other users, and manage their own skills.

In addition, the subsystem provides visibility into user activity through the ability to view online users. Administrative control is supported through role management features, where Tenant Administrators can update user roles to ensure that users have the appropriate permissions and responsibilities within the organization. Together, these capabilities help maintain accurate user information, encourage skill visibility, and support effective access control across the platform.

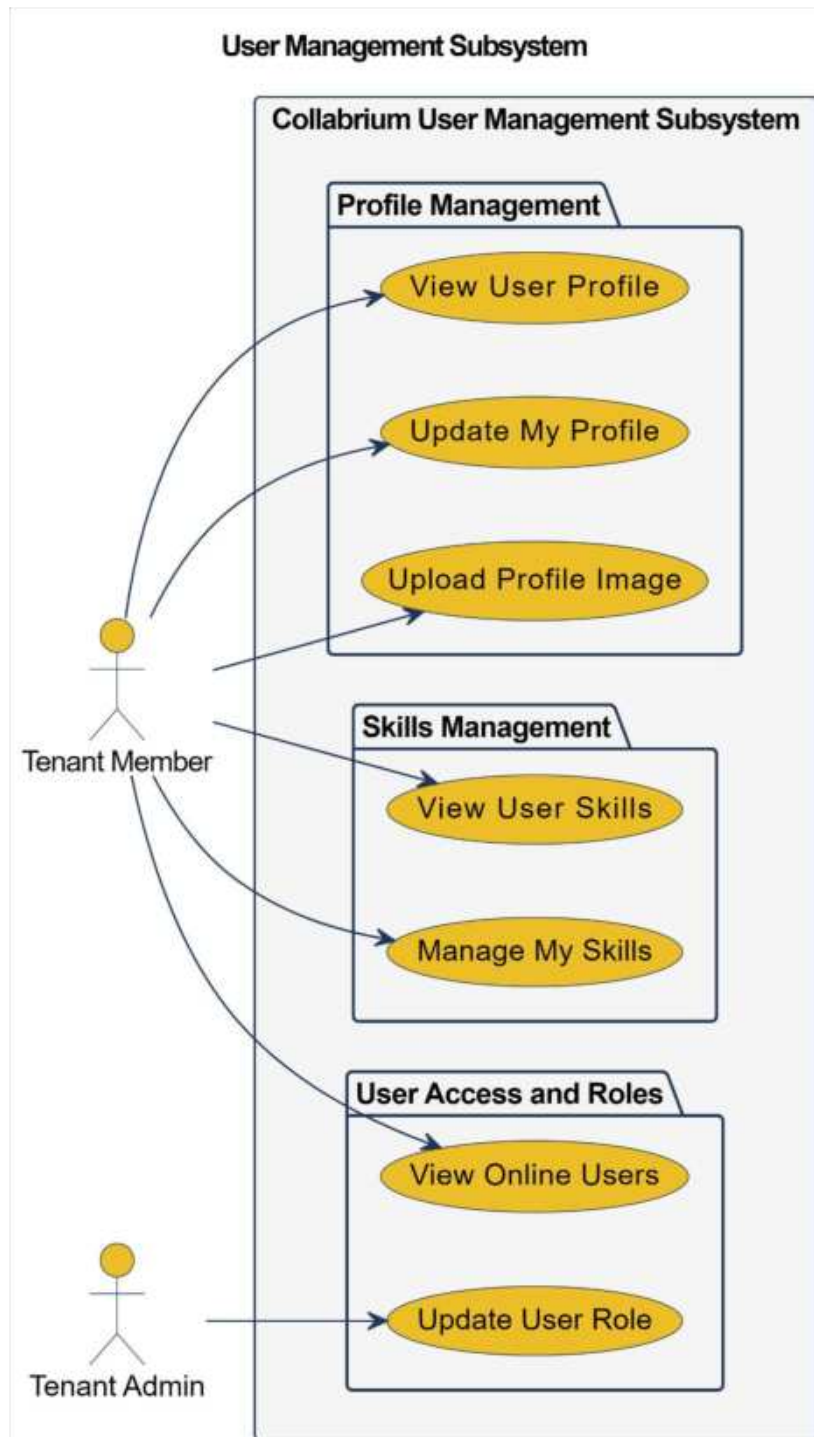


Figure 5.3 4 User Management Subsystem Use Case Diagram

### 5.3.5 Team Management Subsystem

The **Team Management Subsystem** is responsible for organizing and managing teams within the platform. It enables tenant members to browse available teams, apply filters to locate specific teams, view detailed team information, and explore candidate team members. These features provide users with better visibility into team structures and available resources across the organization.

The subsystem also supports team administration activities for authorized team managers, including Tenant Administrators, Product Owners, Team Leads, and Business Analysts.

Authorized users can create new teams, update team information, add or remove team members, and control team availability by activating or deactivating teams. Together, these capabilities help maintain well-structured teams, support collaboration, and ensure efficient management of team membership and organization.

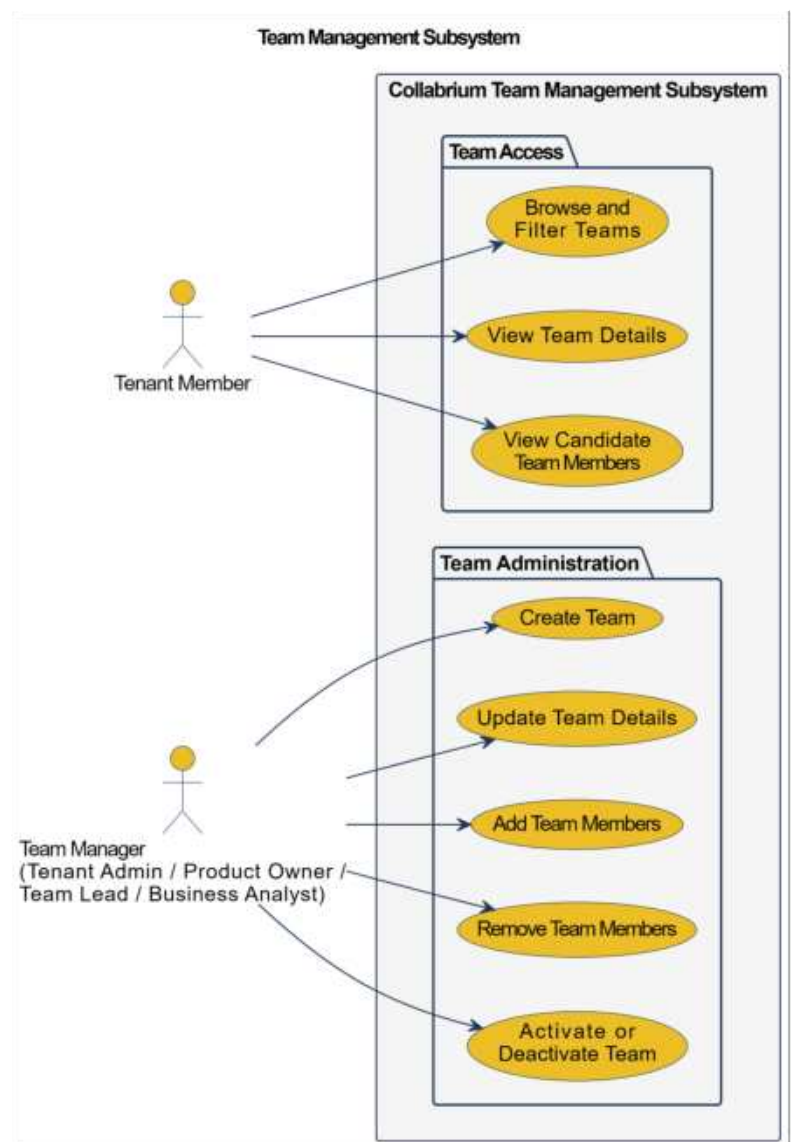


Figure 5.3 5 Team Management Subsystem Use Case Diagram

### 5.3.6 Project Management Subsystem

The Project Management Subsystem is responsible for organizing, tracking, and managing projects throughout their lifecycle. It allows tenant members to access their assigned projects and view important project information, including project details, boards, backlogs, epics, user stories, and project history. These features provide team members with a clear view of project progress and ongoing activities.

The subsystem also supports project oversight by enabling high-level managers, such as CEOs and Tenant Administrators, to browse and filter projects across the organization. For project managers, including Product Owners, Team Leads, Business Analysts, and Tenant Administrators, the subsystem provides administrative capabilities such as creating, updating, and deleting projects, as well as assigning or unassigning teams. Together, these functions support effective project planning, collaboration, and project governance across the platform.

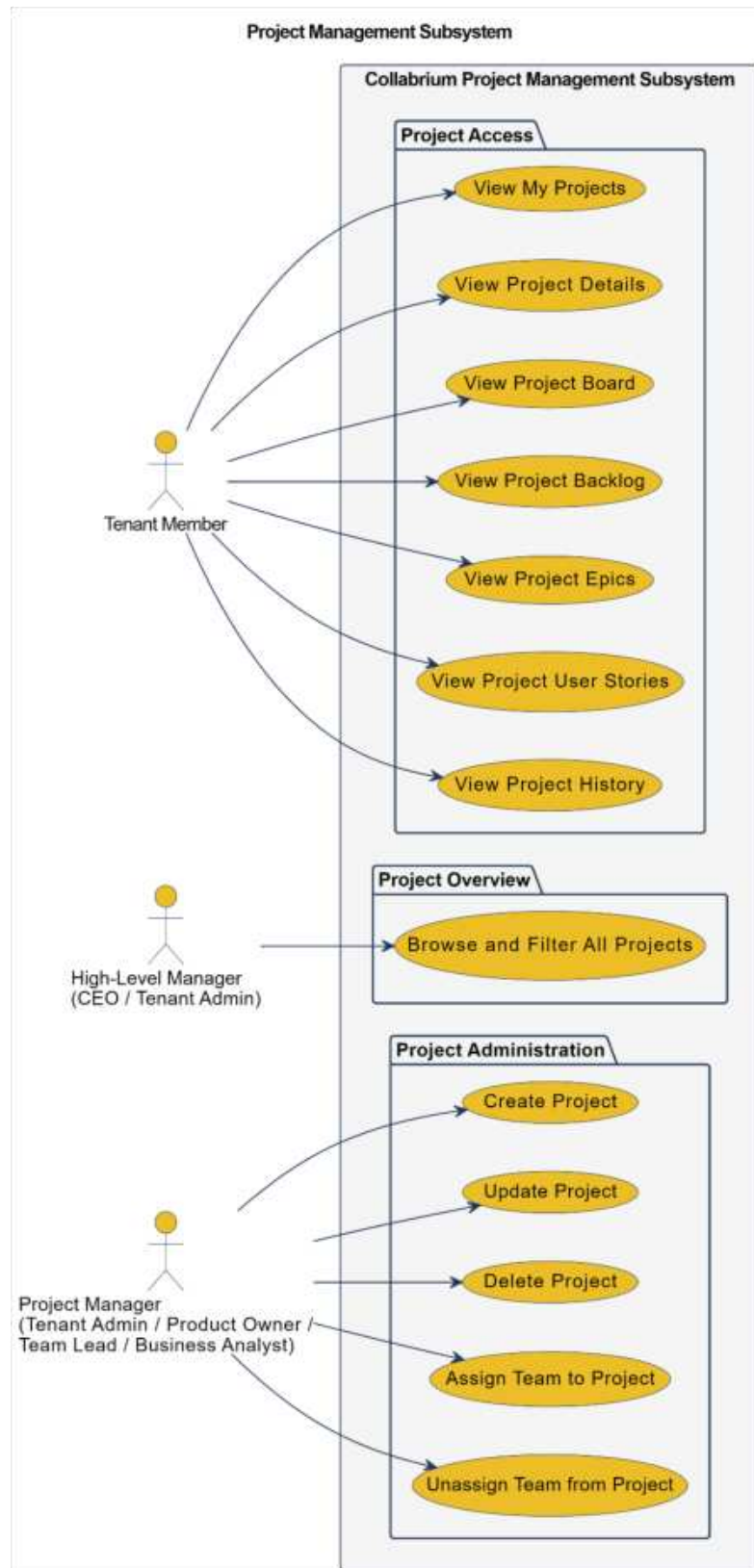


Figure 5.3 6 Project Management Subsystem Use Case Diagram

### 5.3.7 Work Item Core Management Subsystem

The Work Item Core Management Subsystem is responsible for managing work items throughout their lifecycle and supporting collaboration within projects. It allows project members to browse and view work items, update their status, manage assignments, and delete work items when permitted.

The subsystem supports planning activities through the creation and management of epics and user stories, while also enabling execution-focused contributors to create and update tasks, bugs, and test cases.

In addition, it provides collaboration and traceability features by allowing users to manage comments, attachments, code links, and relationships between work items. These capabilities help teams organize project requirements, track progress, maintain traceability, and improve collaboration across the development process.

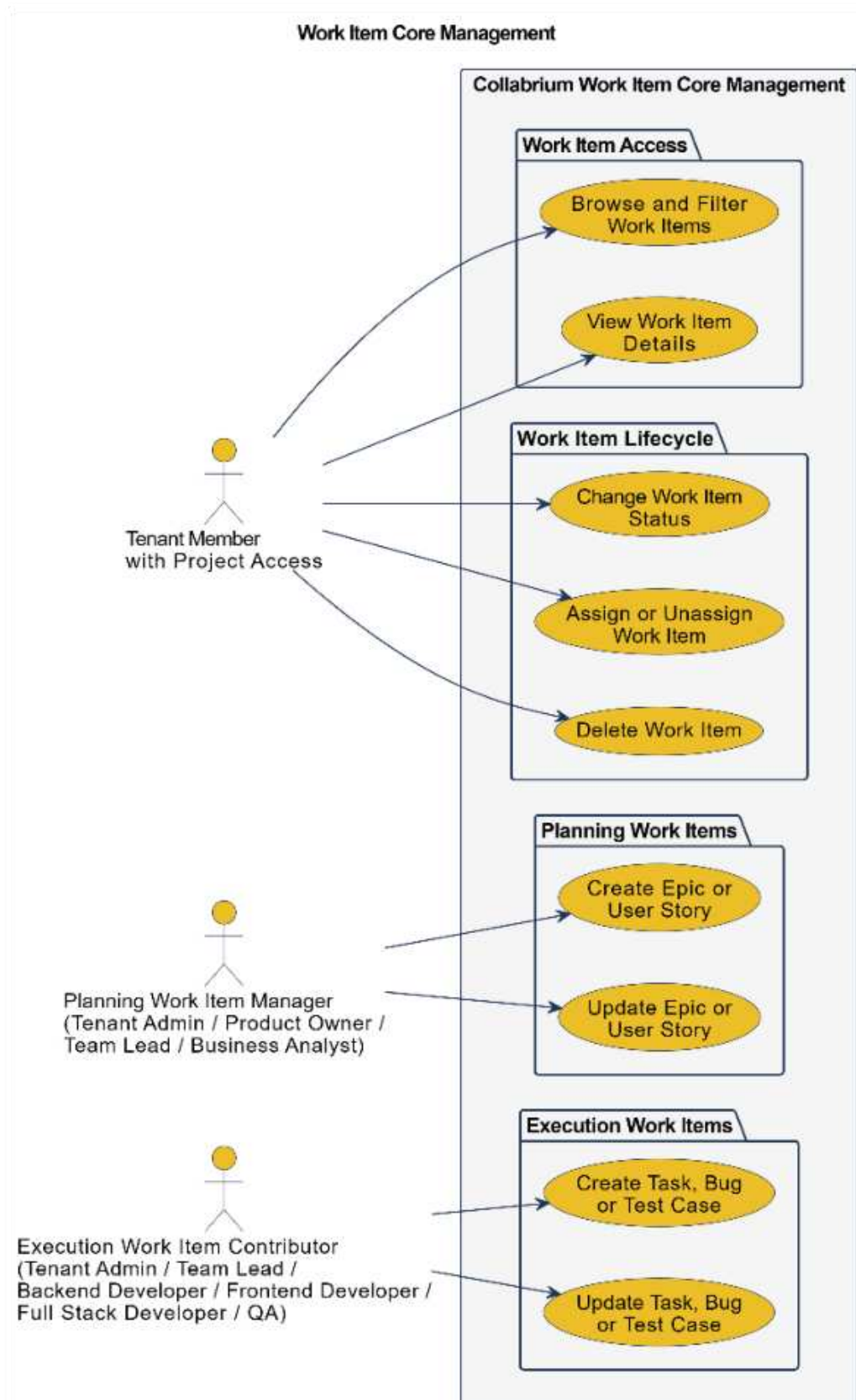
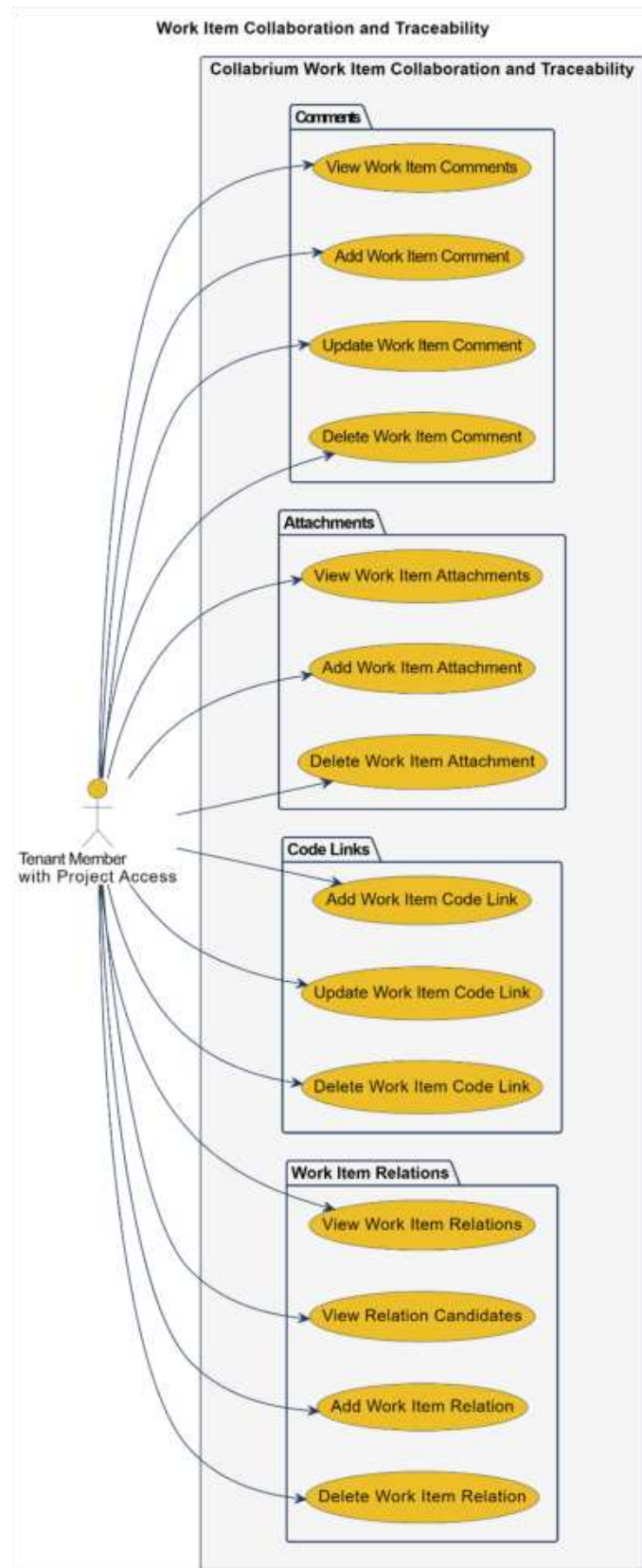


Figure 5.3 7 Work Item Core Management Subsystem Use Case Diagram



*Work Item Collaboration and Traceability Use Case Diagram*

### 5.3.8 Sprint Management Subsystem

The **Sprint Management Subsystem** is responsible for planning, organizing, and controlling sprints within the project lifecycle. It enables project members to browse and view sprint information, including sprint details and the work items assigned to each sprint, along with their progress.

For sprint planning activities, the subsystem allows sprint managers, such as Product Owners, Team Leads, Business Analysts, and Tenant Administrators, to create new sprints, update sprint details and schedules, and assign work items to specific sprints. It also supports sprint lifecycle control by enabling managers to start, complete, cancel, or delete sprints when necessary. These features ensure structured sprint execution, better tracking of progress, and efficient delivery of project work in an iterative manner.

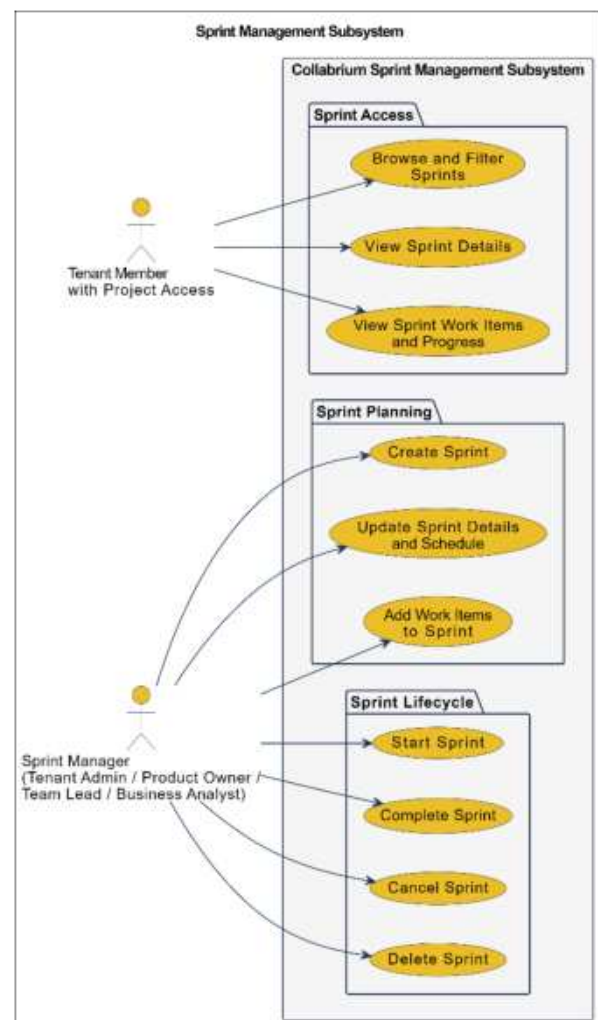


Figure 5.3 8 Sprint Management Subsystem Use Case Diagram

### 5.3.9 Reports and Analytics Subsystem

The **Reports and Analytics Subsystem** provides comprehensive reporting and analytical capabilities across tenants, projects, sprints, and work items to support performance evaluation and decision-making. It enables different levels of users to access structured insights based on their role and scope of responsibility.

At the tenant and project level, the subsystem allows high-level managers and authorized users to view or export reports related to tenant information, project progress, sprint summaries, cycle time, and velocity. These reports help monitor overall delivery performance and assess how

efficiently projects and teams are progressing.

At the productivity and work item level, tenant members can access reports that evaluate developer and team productivity, track bug severity distribution, identify blocked or overdue work items, and assess overall project health. Together, these analytics features transform raw project data into actionable insights that support continuous improvement and better decision-making across the platform.

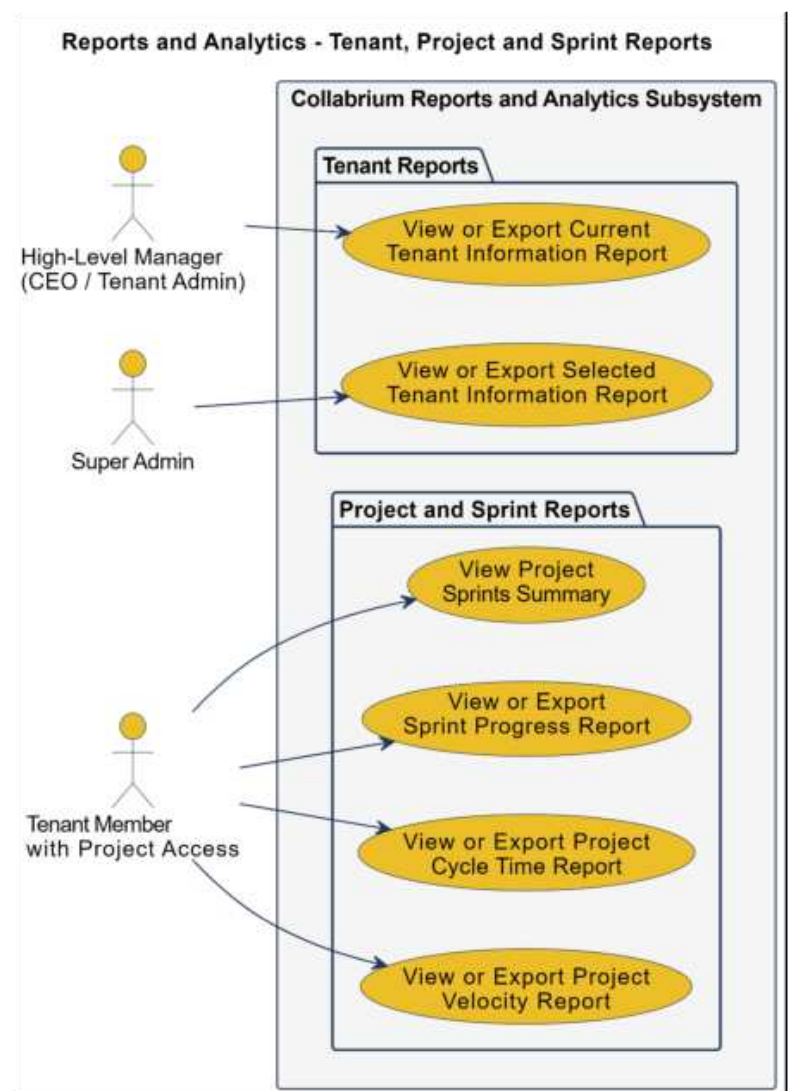
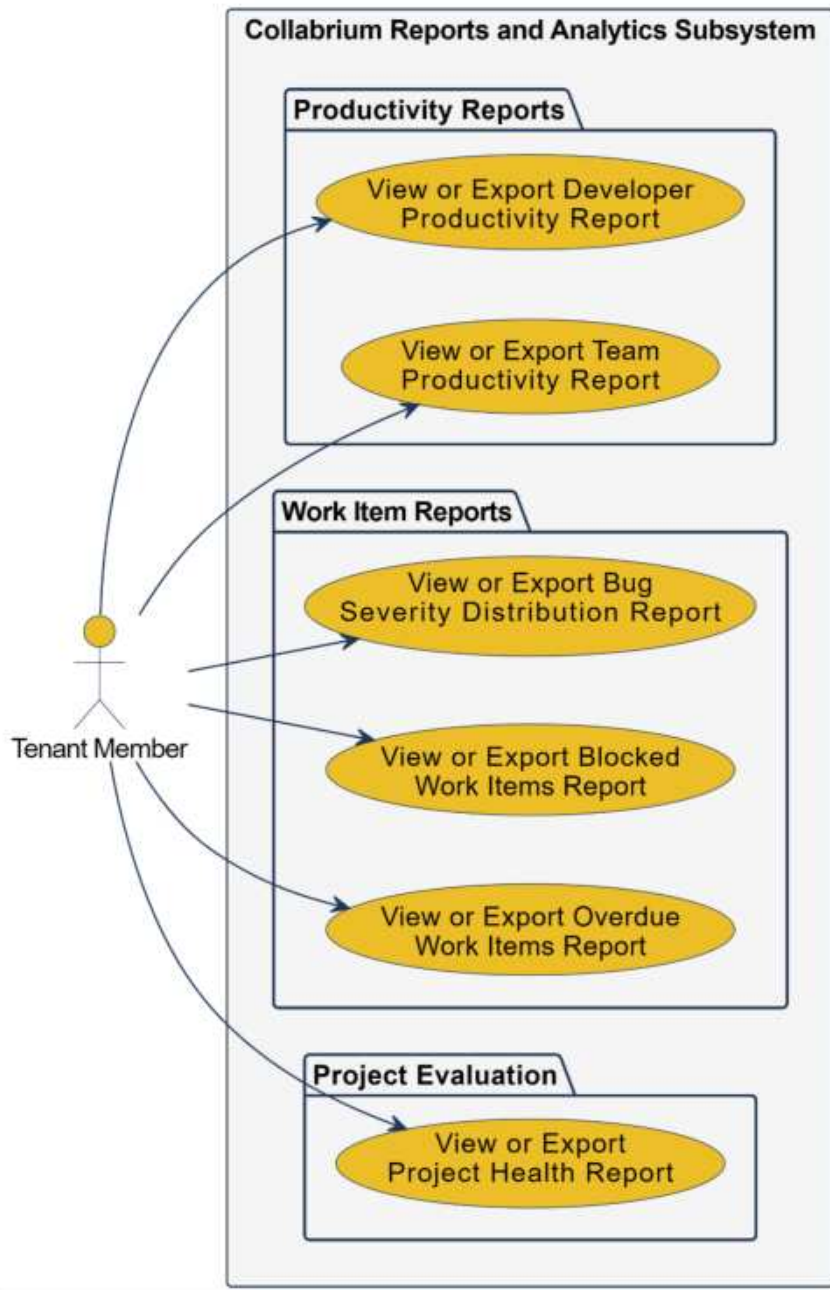


Figure 5.3 9 Reports and Analytics - Tenant, Project and Sprint Reports Use Case Diagram

## Reports and Analytics - Productivity and Work Item Reports



*Reports and Analytics - Tenant, Project and Sprint Reports Use Case Diagram*

### 5.3.10 Learning Hub Subsystem

The **Learning Hub Subsystem** is responsible for managing learning content, user learning activities, and knowledge distribution across the platform. It provides tenant members with the ability to browse courses, view course details, save or start courses, complete learning activities, and rate courses based on their experience. Users can also track their learning progress through personal dashboards that include assigned courses, saved courses, enrolled courses, and overall learning summaries.

In addition, the subsystem allows users to explore published learning roadmaps, including their topics, subtopics, and related resources, as well as interact with published articles through likes and comments. This creates an interactive learning environment that combines structured learning paths with community engagement.

On the management side, learning content managers such as Product Owners, Team Leads, Business Analysts, and Tenant Administrators can create and manage courses, organize course metadata, and review course statistics and ratings. They are also responsible for managing learning roadmaps, including publishing, hiding, structuring content, and organizing topics and resources. The subsystem further supports AI-based assistance for generating roadmap drafts and content suggestions.

Finally, the system includes course assignment management features that allow managers to assign or unassign courses to individual users or teams, as well as track assignment statistics and monitor learning distribution across the organization. Together, these capabilities ensure a structured, scalable, and engaging learning experience for all users.

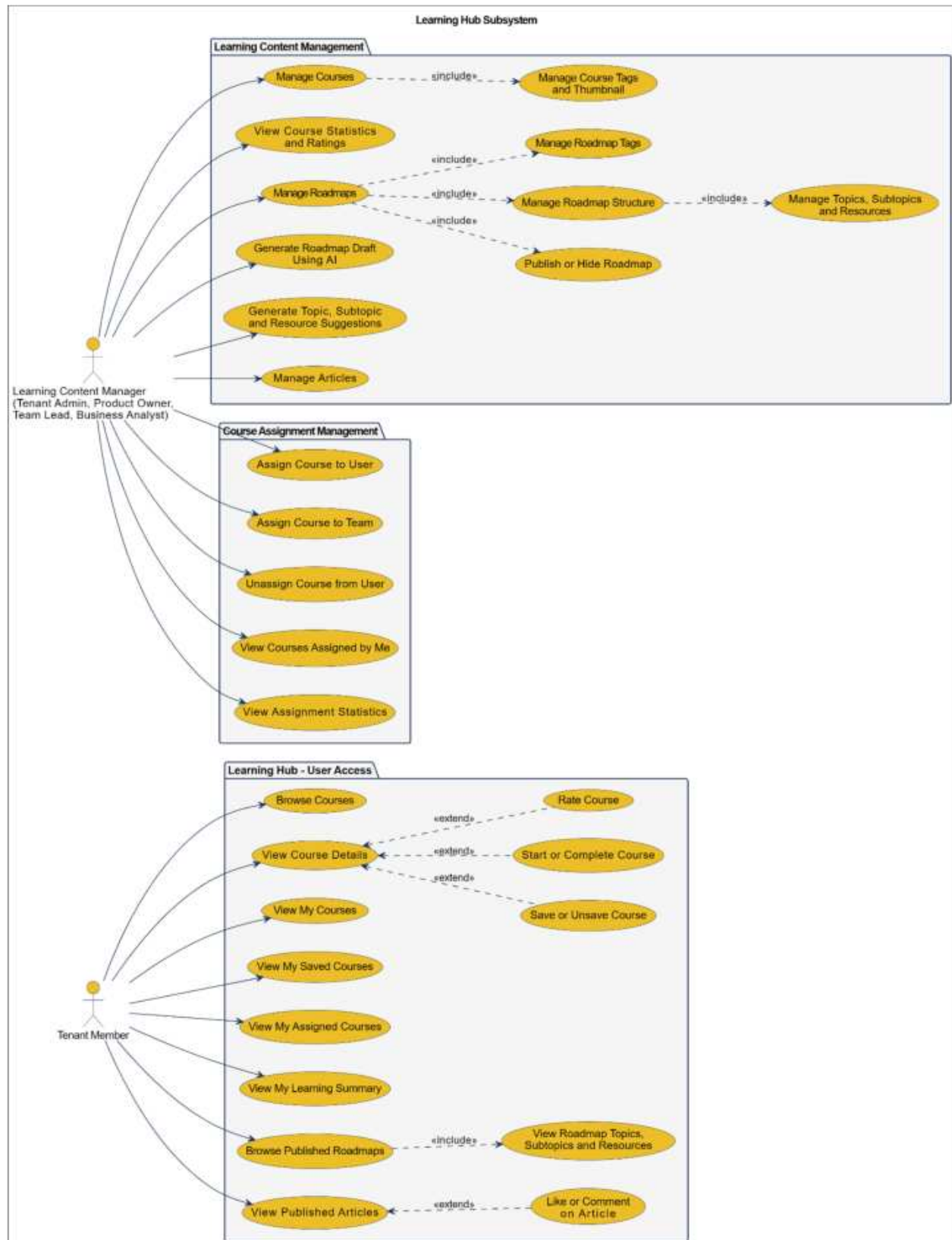


Figure 5.3 10 Learning Hub Subsystem Use Case Diagram

### 5.3.11 Community Hub Subsystem

The Community Hub Subsystem is designed to facilitate communication, engagement, and interaction between users within the platform. It provides tenant members with access to announcements, events, and community posts, allowing them to stay informed and actively participate in community activities.

Users can view announcements and explore their details, browse events along with event calendars, and manage their attendance by joining or canceling participation. The subsystem also supports community posts where users can browse content, view post details, and interact through likes, comments, and viewing engagement metrics such as likes count.

On the management side, community content managers, including Tenant Administrators, Product Owners, Team Leads, and Business Analysts, are responsible for creating and managing announcements and events, including defining target audiences and uploading event banners. They also manage posts by pinning or unpinning important content to ensure visibility and relevance. Overall, the subsystem enhances communication flow and strengthens engagement across the community.

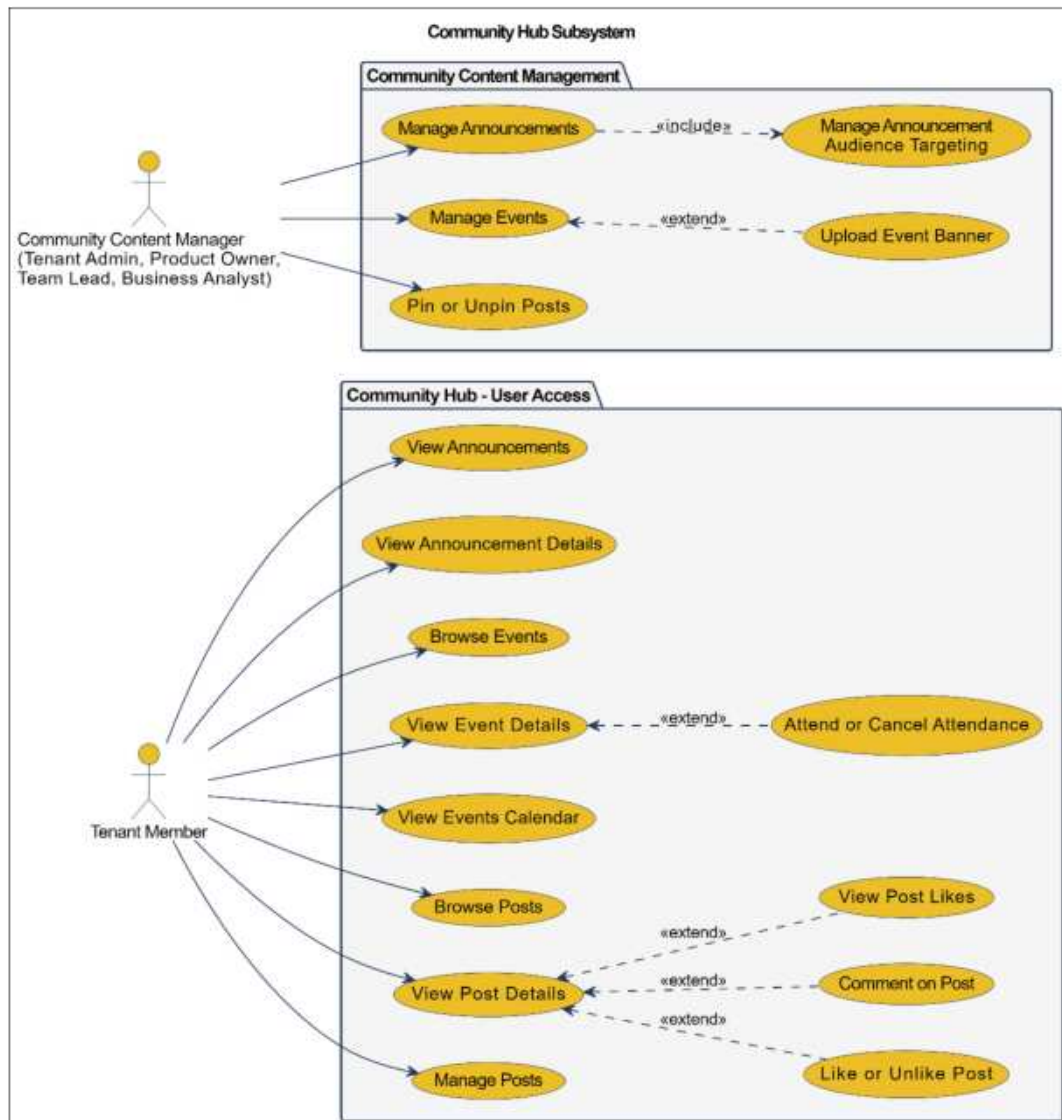


Figure 5.3 11 Community Hub Subsystem Use Case Diagram

### 5.3.12 AI Assistant Management Subsystem

The **AI Assistance Subsystem** enhances productivity and decision-making across the platform by providing intelligent suggestions and analytical insights tailored to different user roles and system areas.

For work item management, the subsystem supports planning and execution activities by generating AI-based suggestions. Planning managers, such as Product Owners and Team Leads, can receive AI-generated suggestions for epics and user stories, while execution contributors, including developers and QA engineers, can generate suggestions for tasks, bugs, and test cases. This helps users define clearer and more structured work items with reduced effort.

In the learning and skills domain, the system assists tenant members by analyzing skill gaps and helping learning content managers create structured learning roadmaps. It can generate full roadmap drafts, topics, subtopics, and resource suggestions to support skill development and continuous learning.

For reporting and analytics, the subsystem provides AI-generated insights for different levels of users. High-level managers can access tenant-level insights, while project members receive insights related to sprint progress, cycle time, and velocity. Additionally, tenant members can explore insights on productivity, bug distribution, blocked and overdue work items, and overall project health. These capabilities transform static reports into actionable intelligence, enabling smarter decisions across all organizational levels.

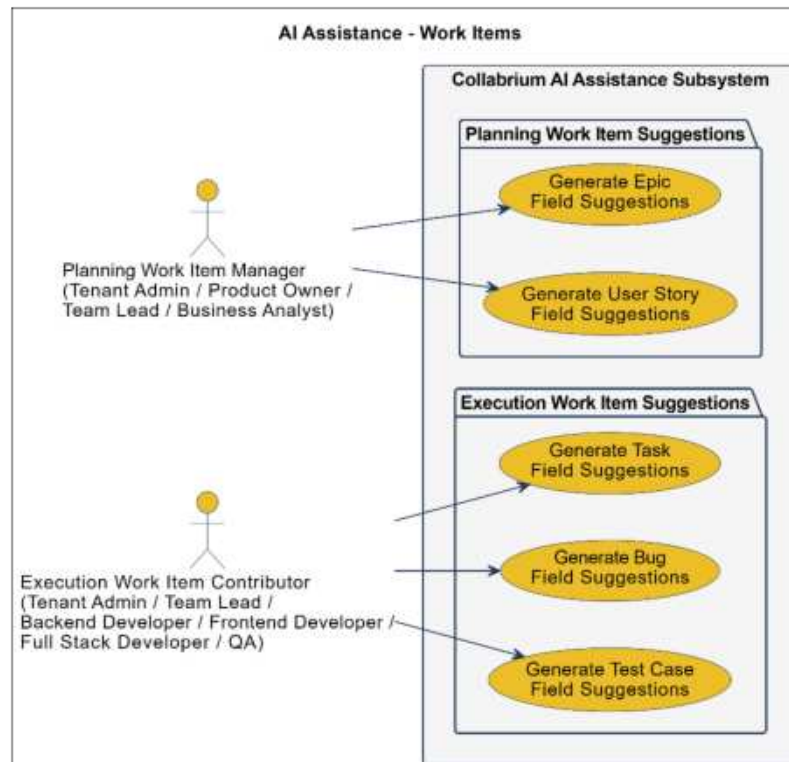
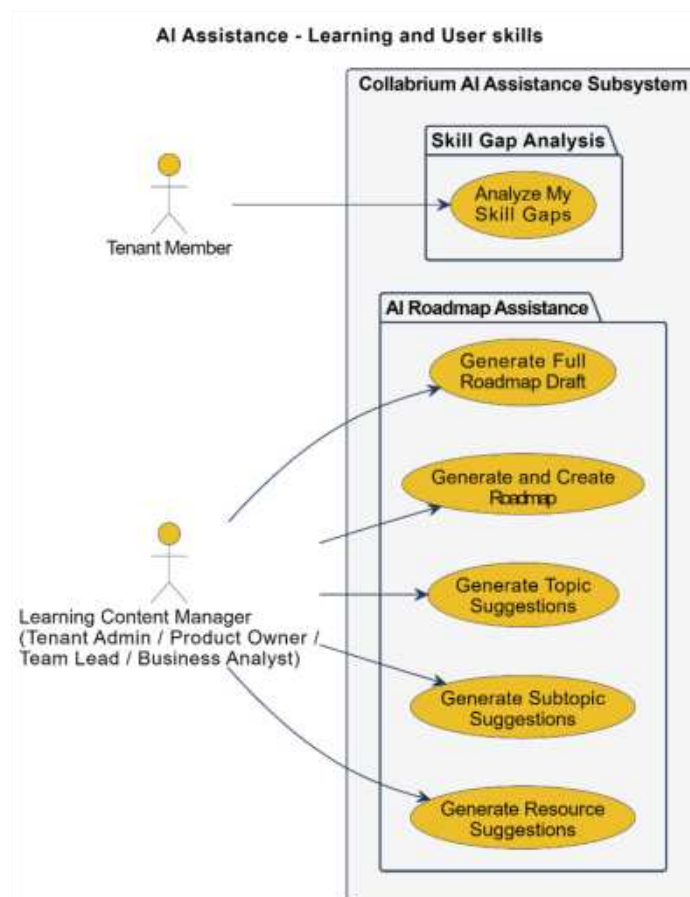
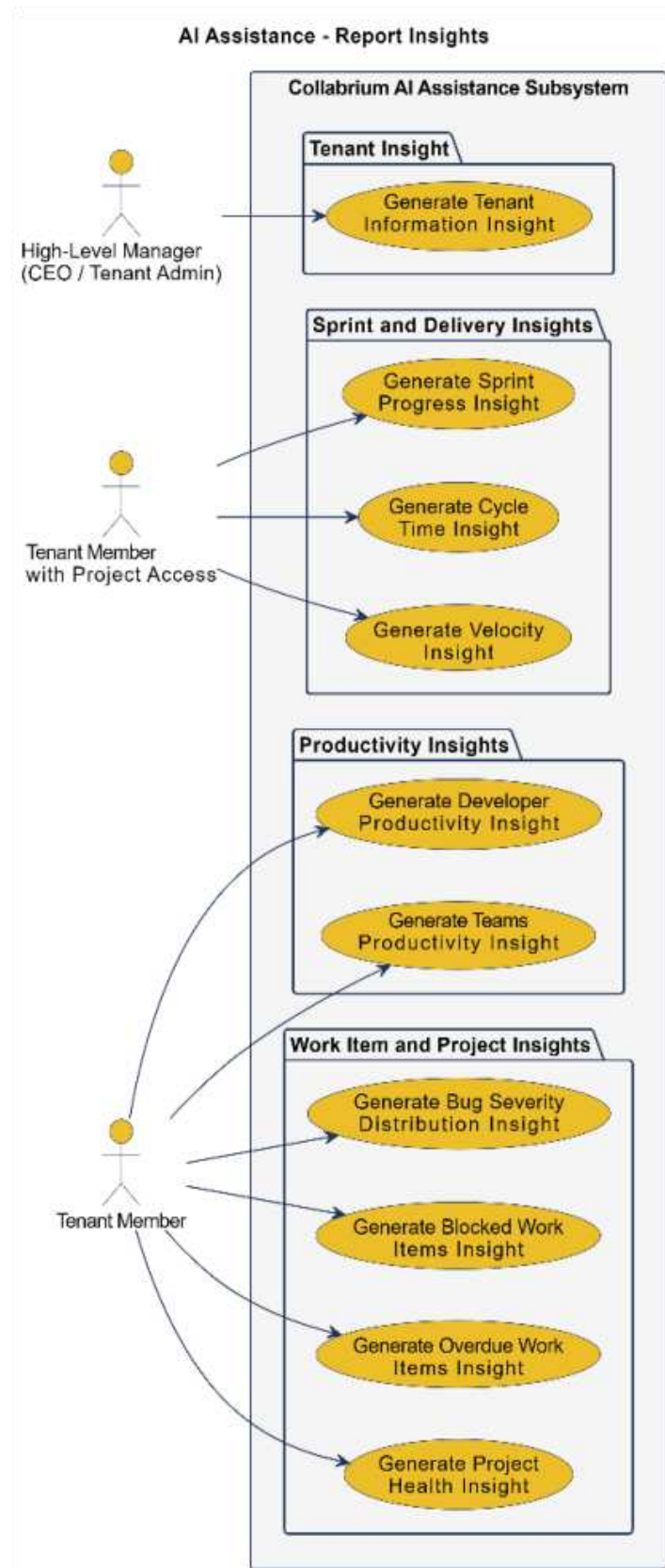


Figure 5.3 12 AI Assistance - Work Items Use Case Diagram



AI Assistance – Learning and User Skills Use Case Diagram



*AI Assistance - Report Insights Use Case Diagram*

### 5.3.13 Notifications and Alerts Subsystem

The **Notifications and Alerts Subsystem** is responsible for keeping users continuously informed about important activities and updates within the platform. It enables tenant members to browse and filter their notifications, as well as view the number of unread notifications to stay aware of pending updates. The subsystem also provides notification management features that allow users to mark individual notifications as read or mark all notifications at once, helping them maintain an organized notification center. In addition, it supports real-time alerts by delivering instant notifications and updating unread counts dynamically as new events occur. Overall, this subsystem ensures timely communication between the system and users while maintaining clarity and control over incoming alerts.

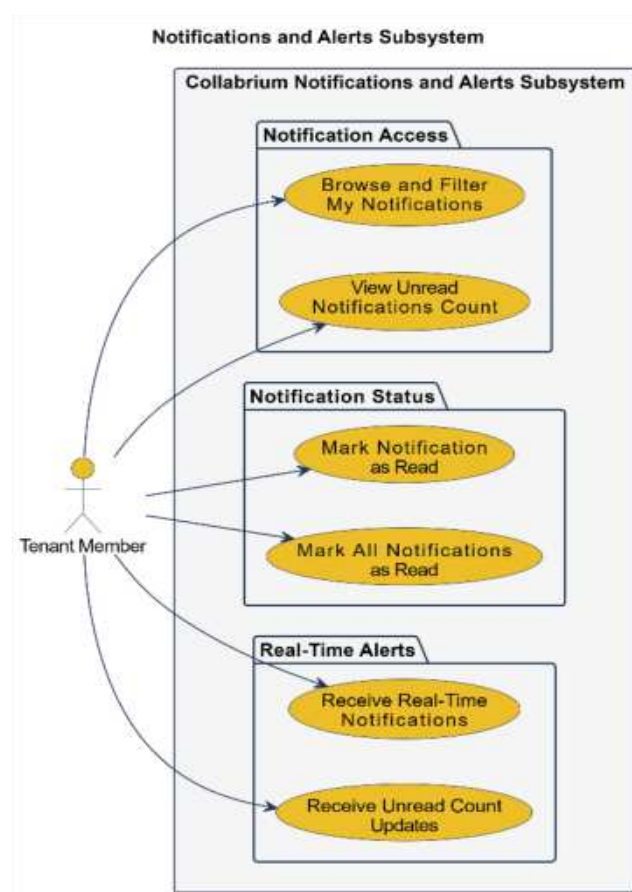


Figure 5.3 13 Notifications and Alerts Subsystem Use Case Diagram

## 5.4 Use Case Descriptions

---

Table 5.4.1: Login Use Case Description

|                       |                                                                                                                                                                                                                                                                                                                       |
|-----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Use Case name:        | Login                                                                                                                                                                                                                                                                                                                 |
| Actors                | User                                                                                                                                                                                                                                                                                                                  |
| Subsystem             | Authentication                                                                                                                                                                                                                                                                                                        |
| Goal                  | allow the user to sign in to his account in the system.                                                                                                                                                                                                                                                               |
| Preconditions:        | <ul style="list-style-type: none"><li>- User must have a registered account.</li><li>- User must have a confirmed email.</li><li>- User must have completed the mandatory first-time password change (Force Change Password) after registration.</li><li>- Credentials (email + password) must be provided.</li></ul> |
| Triggering Event:     | <ul style="list-style-type: none"><li>- The user opens the login page.</li><li>- The user submits their email and password.</li></ul>                                                                                                                                                                                 |
| Scenario:             | <p>The User attempts to authenticate into the system by entering valid login credentials (email and password).</p> <p>The system validates the request, checks the credentials against the Identity database, and either grants access or rejects the attempt.</p>                                                    |
| Exception conditions: | <ul style="list-style-type: none"><li>- Invalid email or password.</li><li>- Email not confirmed.</li><li>- User didn't change password (first-time login).</li><li>- Account suspended or deactivated.</li><li>- Database or identity service unavailable.</li></ul>                                                 |

Table 5.4.1 Login Use Case Description

Table 5.4.2: View Tenant Profile Use Case Description

|                       |                                                                                                                                                                                                                                                                                                                                  |
|-----------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Use Case name:        | View Tenant Details and Analysis                                                                                                                                                                                                                                                                                                 |
| Actors                | Super Admin                                                                                                                                                                                                                                                                                                                      |
| Subsystem             | Tenant Management                                                                                                                                                                                                                                                                                                                |
| Goal                  | Allow the Super Admin to view detailed information and analysis about a specific tenant in the system.                                                                                                                                                                                                                           |
| Preconditions:        | <ul style="list-style-type: none"> <li>- The Super Admin must be authenticated.</li> <li>- The tenant record to be retrieved must exist in the database.</li> <li>- A valid TenantId must be provided.</li> </ul>                                                                                                                |
| Triggering Event:     | <ul style="list-style-type: none"> <li>- The Super Admin selects a tenant from the Tenants Dashboard.</li> <li>- The Super Admin performs a direct request to view a tenant profile.</li> </ul>                                                                                                                                  |
| Scenario:             | <p>The Super Admin submits a request to view a specific tenant profile.</p> <p>The system validates the request, checks authorization policy, retrieves tenant data from the database, and returns the profile details.</p> <p>If the request is invalid or the actor is unauthorized, the system returns an error response.</p> |
| Exception conditions: | <ul style="list-style-type: none"> <li>- Tenant does not exist in the database.</li> <li>- Database unavailable / server error.</li> </ul>                                                                                                                                                                                       |

Table 5.4.2 View Tenant Profile Use Case Description

Table 5.4.3: Register New User Use Case Description

|                       |                                                                                                                                                                                                                                                           |
|-----------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Use Case name:        | Register Tenant Employee                                                                                                                                                                                                                                  |
| Actors                | Tenant Admin                                                                                                                                                                                                                                              |
| Subsystem             | Authentication                                                                                                                                                                                                                                            |
| Goal                  | Allow an authorized IT user to create a new user account within the tenant by submitting the required user details, triggering identity creation, and sending a confirmation email.                                                                       |
| Preconditions:        | <ul style="list-style-type: none"> <li>- IT user must be logged in.</li> <li>- Required user data (payload) must be provided.</li> <li>- SQL Identity database must be available</li> </ul>                                                               |
| Triggering Event:     | <ul style="list-style-type: none"> <li>- The IT user initiates "Create New User Account" from the system UI.</li> <li>- IT submits the Create User form.</li> </ul>                                                                                       |
| Scenario:             | <p>The IT user fills and submits a form to create a new user.</p> <p>The system validates the request, checks authorization, processes the creation command, stores the user in the Identity database, and sends the user an email confirmation link.</p> |
| Exception conditions: | <ul style="list-style-type: none"> <li>- Invalid or missing payload fields</li> <li>- Actor does not satisfy IsIT authorization policy</li> <li>- Identity creation failed</li> <li>- Email service unavailable</li> <li>- SQL Server issues</li> </ul>   |

Table 5.4 3 Register New User Use Case Description

Table 5.4.4: Deactivate User Use Case Description

|                       |                                                                                                                                                                                                   |
|-----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Use Case name:        | Deactivate User                                                                                                                                                                                   |
| Actors                | Tenant Admin                                                                                                                                                                                      |
| Subsystem             | User Management                                                                                                                                                                                   |
| Goal                  | Allow a tenant admin to deactivate a User (tenant employee) account temporarily, preventing the user from logging into the system.                                                                |
| Preconditions:        | <ul style="list-style-type: none"> <li>- Tenant Admin must be logged in.</li> <li>- Target user must exist.</li> </ul>                                                                            |
| Triggering Event:     | Tenant Admin initiates the "Deactivate User" action from the system UI.                                                                                                                           |
| Scenario:             | <p>The Tenant Admin requests deactivation for a user.</p> <p>The system checks authorization, marks the user account as inactive in the database, and returns a confirmation of deactivation.</p> |
| Exception conditions: | <ul style="list-style-type: none"> <li>- Actor does not satisfy deactivate authorization policy.</li> <li>- User not found.</li> <li>- Database update fails.</li> </ul>                          |

Table 5.4 4 Deactivate User Use Case Description

Table 5.4.5: Assign Course to Team Use Case Description

|                       |                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    |
|-----------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Use Case name:        | Assign Course to Team                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                              |
| Actors                | Learning Content Manager                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           |
| Subsystem             | Learning Hub System                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                |
| Goal                  | Allow a Learning Content Manager to assign a course to an entire team by submitting the team ID and course ID, triggering assignment in the system database, and making the course available to all team members.                                                                                                                                                                                                                                                                                                                                                  |
| Preconditions:        | <ul style="list-style-type: none"> <li>- Learning Content Manager must be logged in.</li> <li>- Team and Course IDs must be valid.</li> <li>- SQL database must be available.</li> </ul>                                                                                                                                                                                                                                                                                                                                                                           |
| Triggering Event:     | Learning Content Manager initiates the "Assign Course to Team" action from the Assign Course Dialog.                                                                                                                                                                                                                                                                                                                                                                                                                                                               |
| Scenario:             | The Learning Content Manager selects the desired course and initiates the assignment action, which opens an assignment dialog. Inside the dialog, the system displays all available teams for selection. The manager chooses the target team and submits the request. The system then validates the operation, verifies that the user has the required Product Team Manager role, processes the assignment, stores the relationship in the SQL database, and returns a success response to the frontend confirming that the course has been successfully assigned. |
| Exception conditions: | <ul style="list-style-type: none"> <li>- Invalid or missing team/ course IDs.</li> <li>- Actor does not satisfy Learning Content Manager authorization policy.</li> <li>- Database insertion fails.</li> </ul>                                                                                                                                                                                                                                                                                                                                                     |

Table 5.4 5 Assign Course to Team Use Case Description

# 5.5 Domain Model Class Diagram

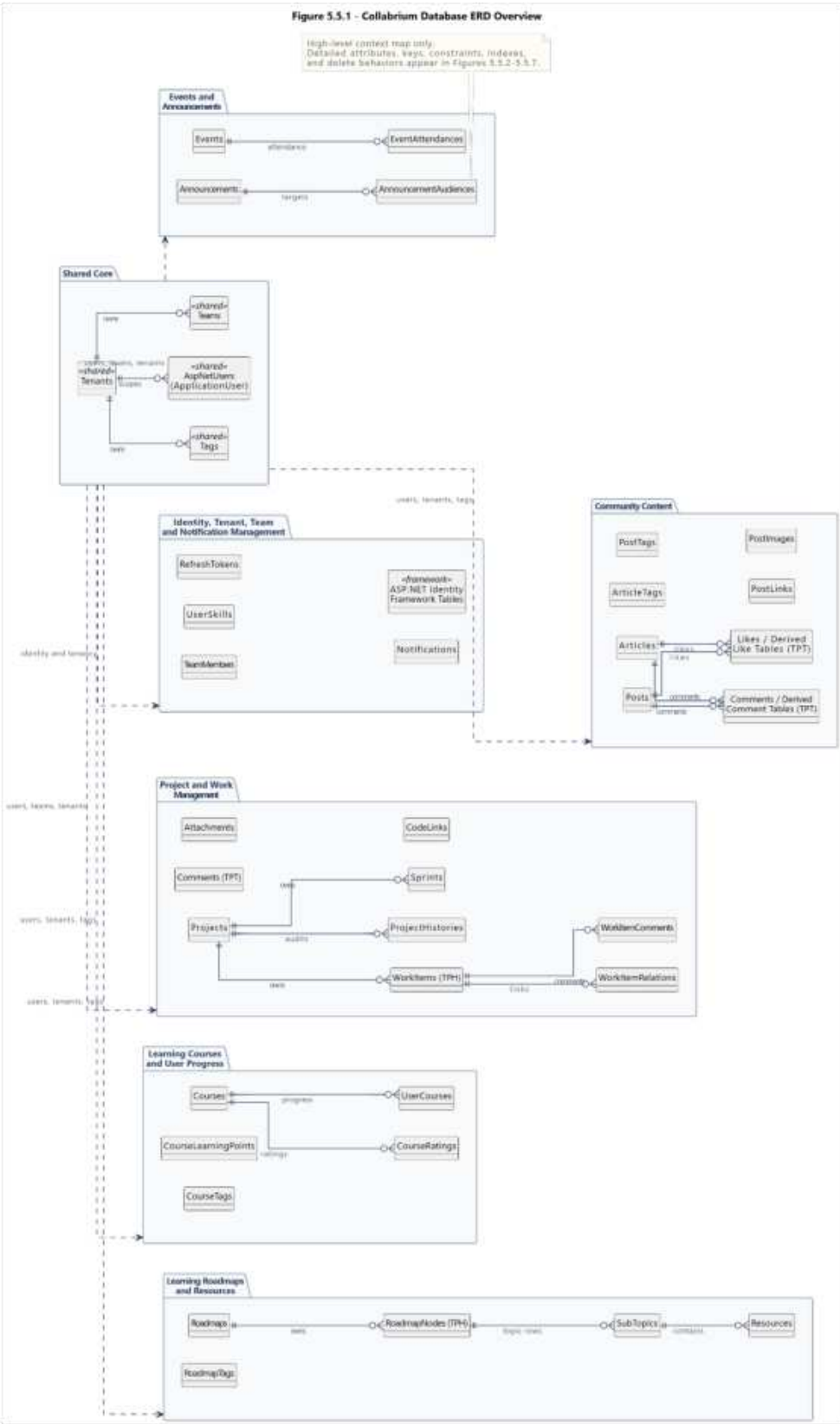


Figure 5.5.1 Collabrium Database ERD Overview

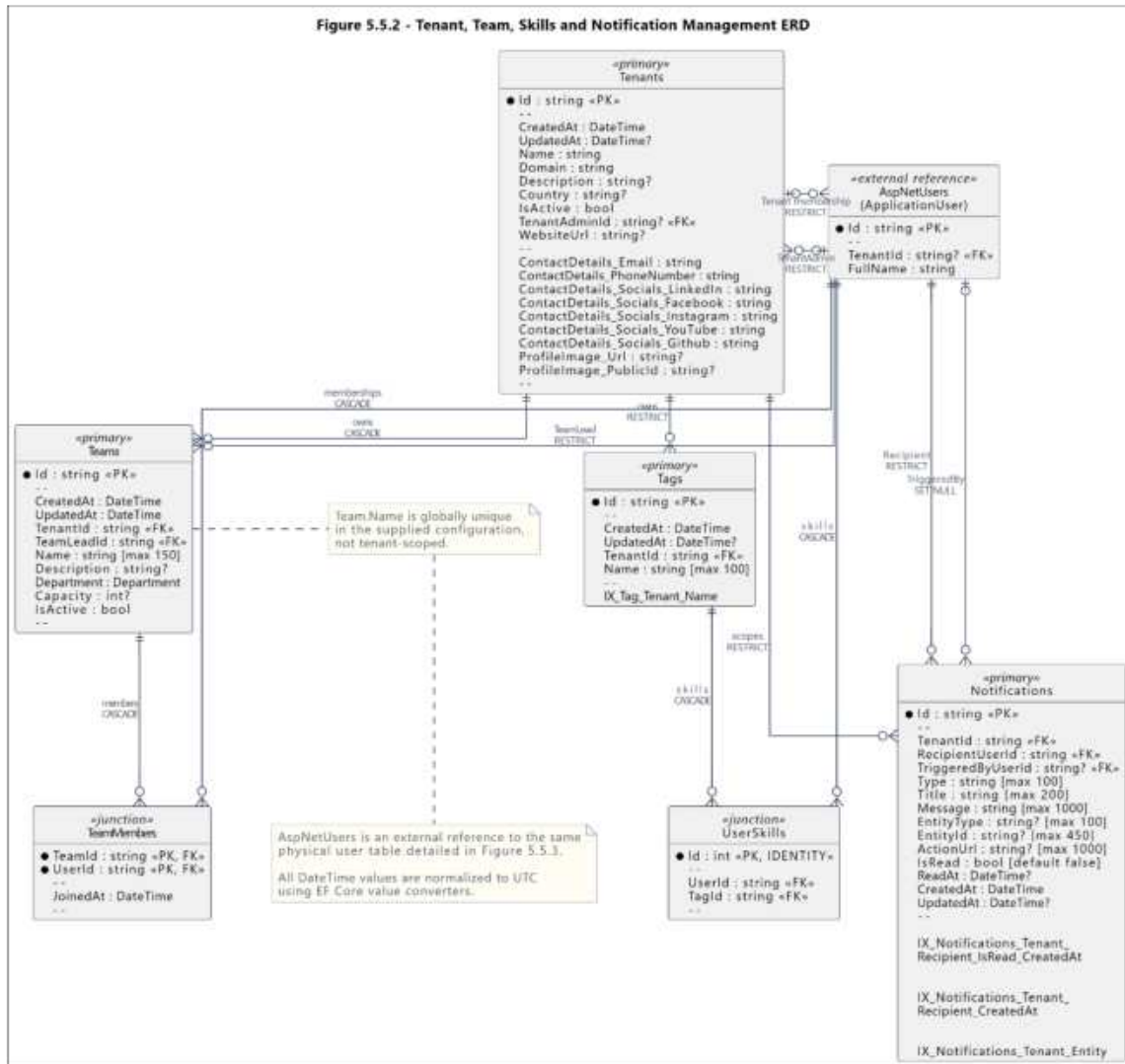


Figure 5.5.2 Tenant, Team, Skills and Notification Management ERD

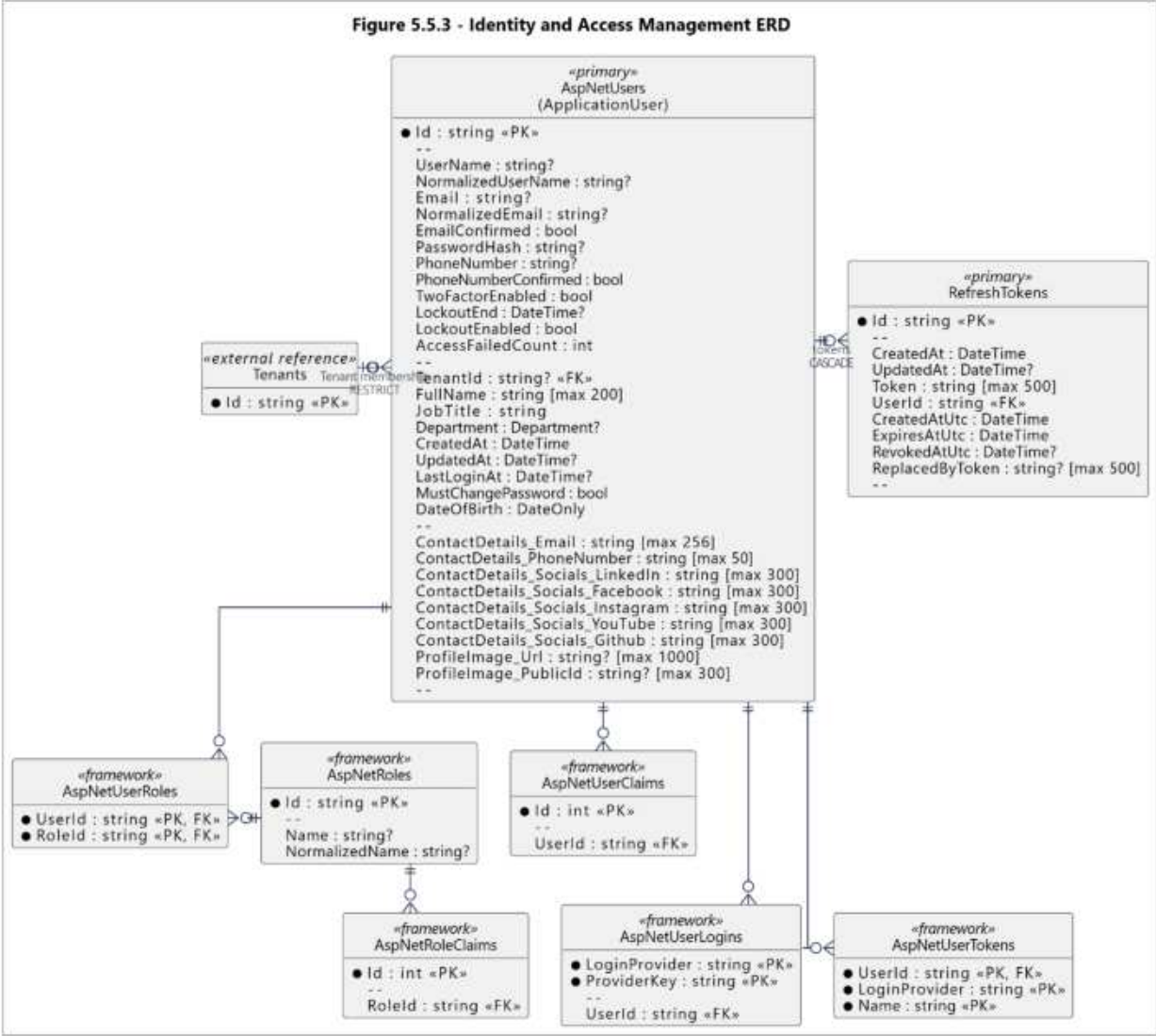


Figure 5.5 3 Identity and Access Management ERD

**Figure 5.5.4 - Project, Sprint and History Management ERD**

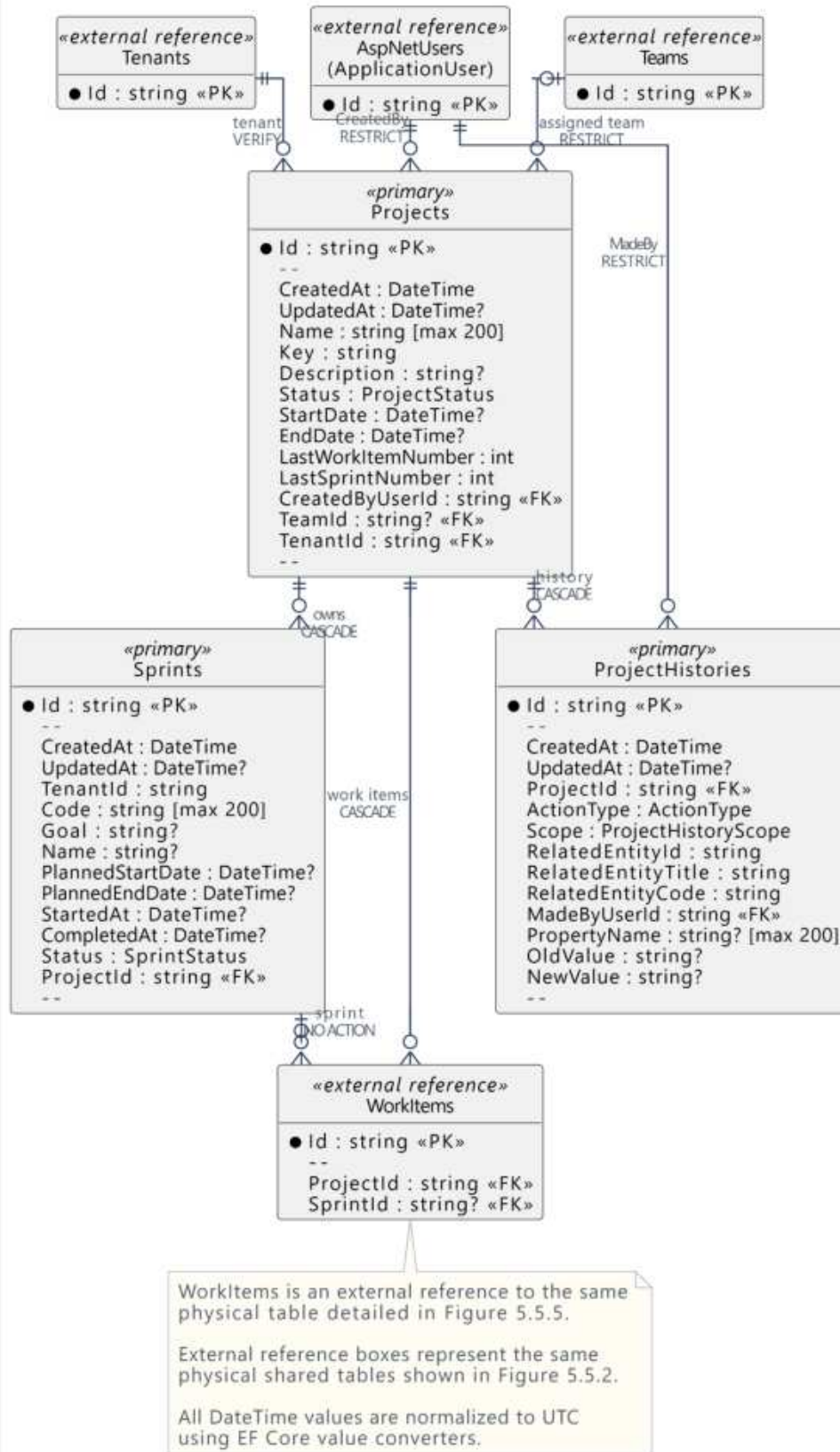


*Figure 5.5 4 Project, Sprint and History Management ERD*

Figure 5.5.5 - Work Item Core and Relationship Management ERD

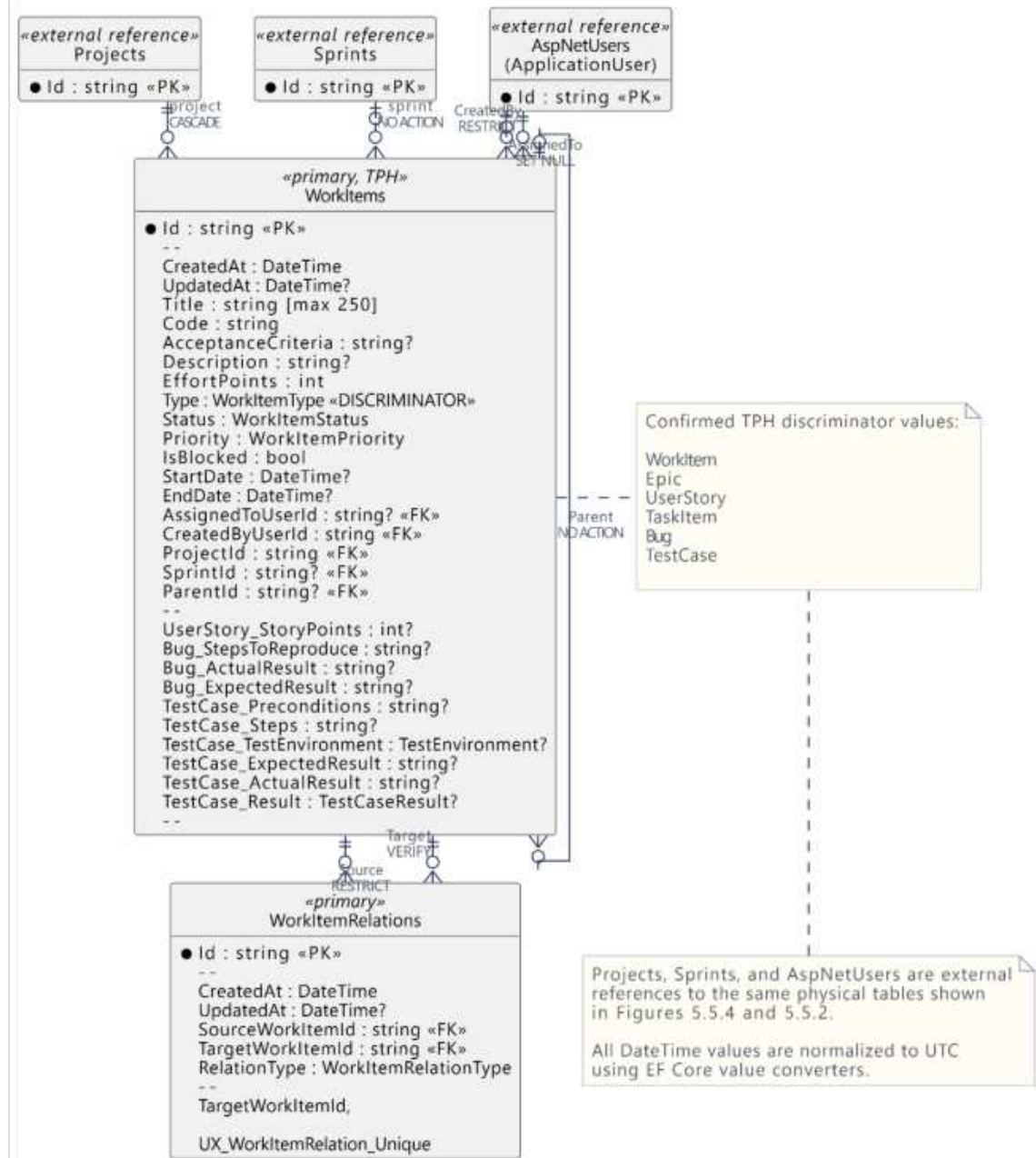


Figure 5.5.5 Work Item Core and Relationship Management ERD

Figure 5.5.6 - Work Item Collaboration and Supporting Artifacts ERD

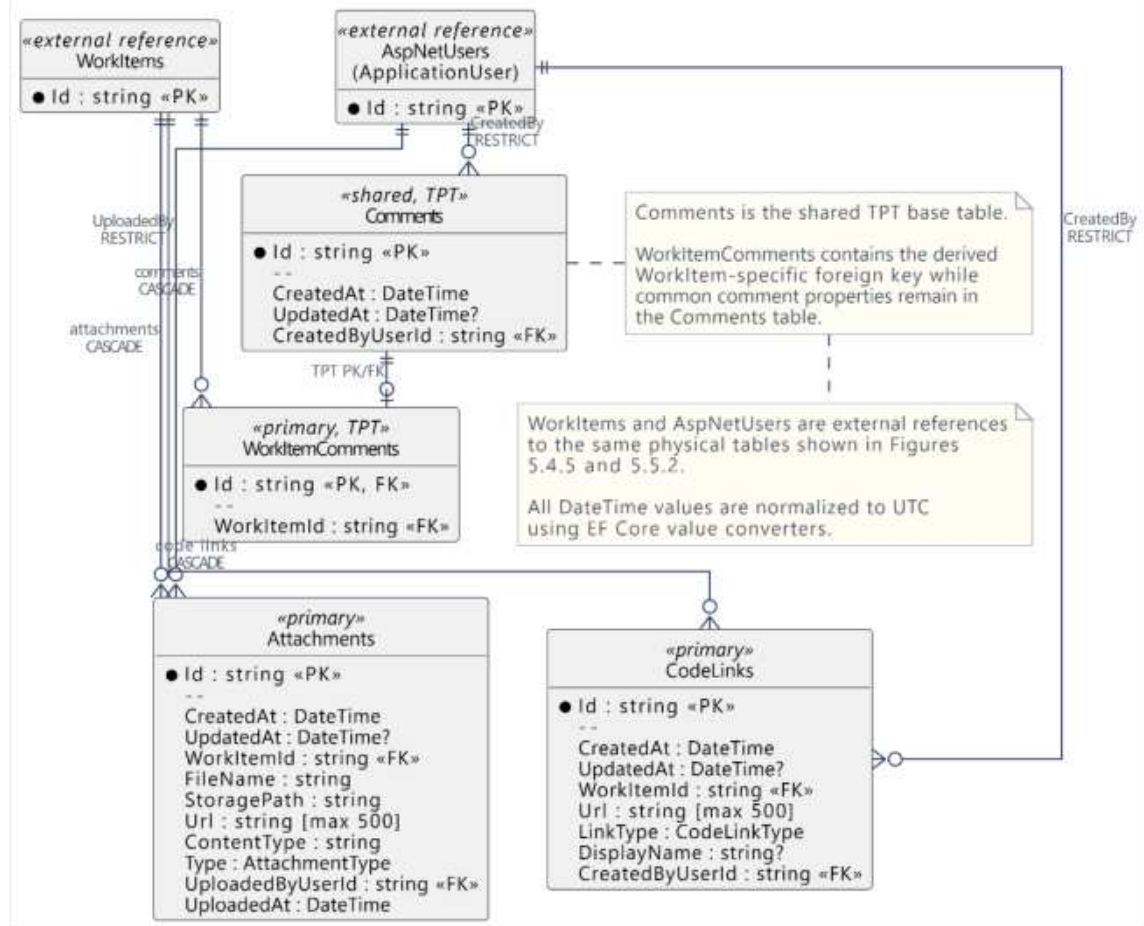


Figure 5.5.6 Work Item Collaboration and Supporting Artifacts ERD

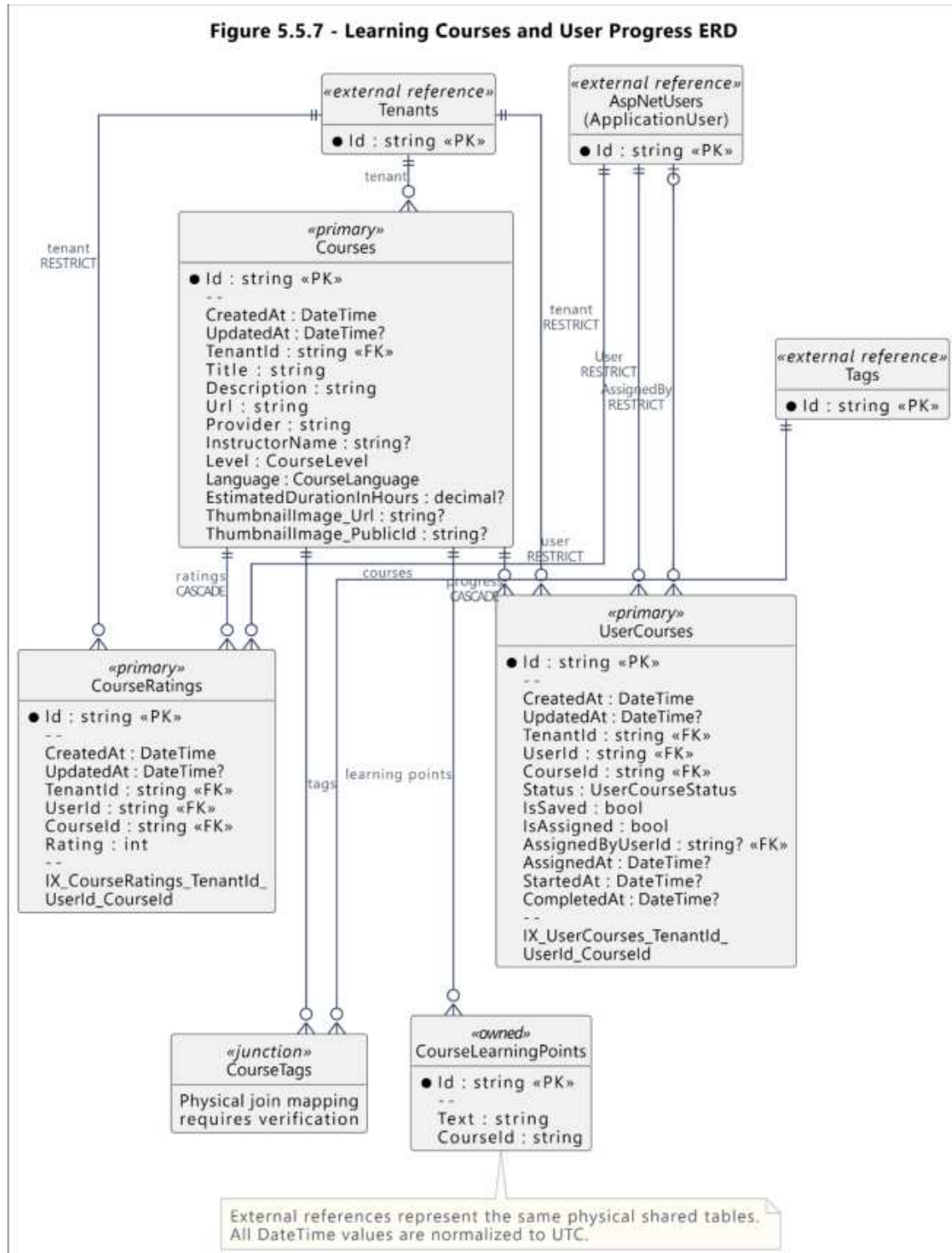


Figure 5.5.7 Learning Courses and User Progress ERD

Figure 5.5.8 - Learning Roadmaps and Resources ERD

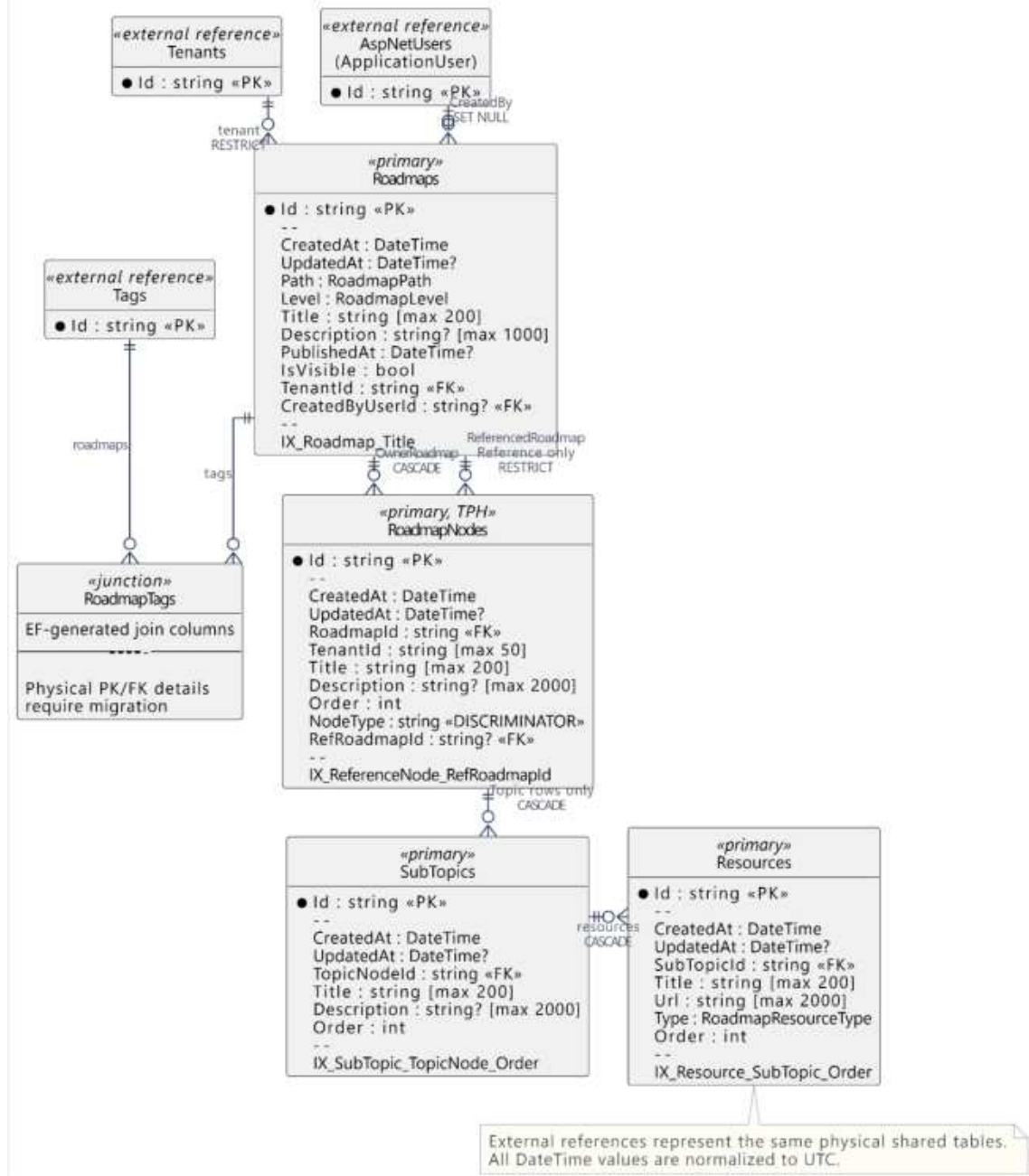


Figure 5.5.8 Learning Roadmaps and Resources ERD

Figure 5.5.9 - Community Articles and Posts ERD

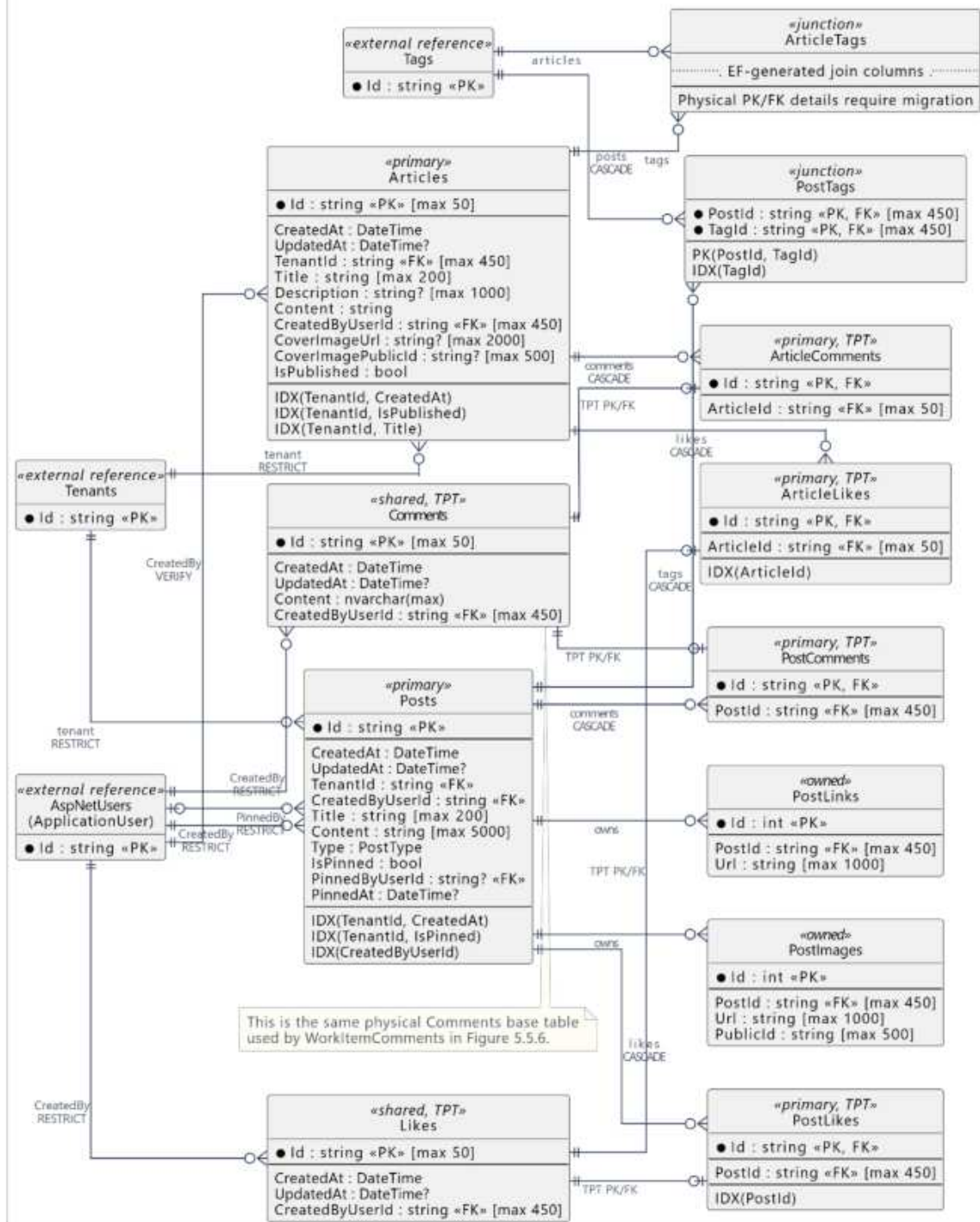


Figure 5.5.9 Community Articles and Posts ERD

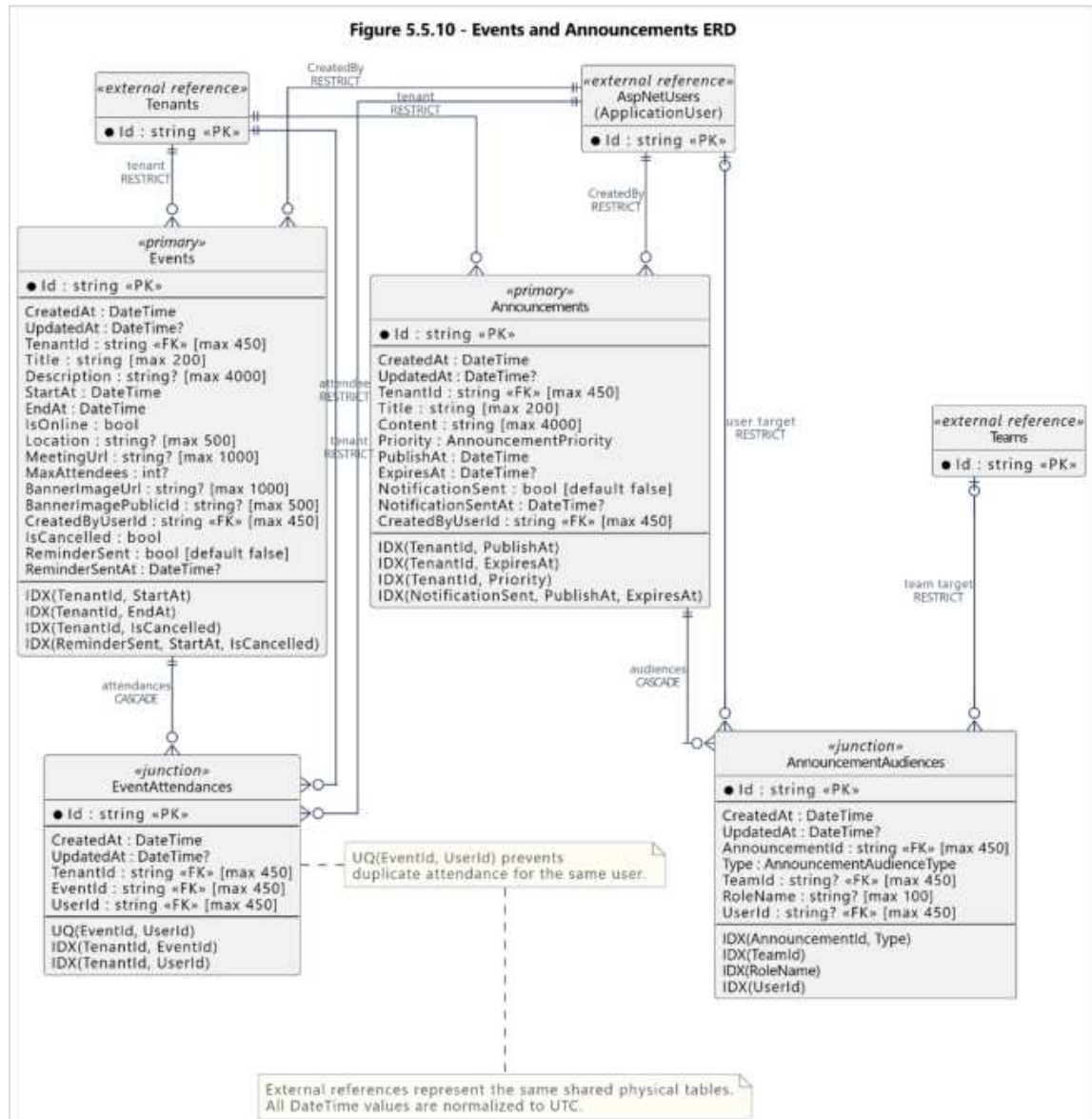


Figure 5.5 10 Events and Announcements ERD

## 5.6 Domain Model Class Diagram

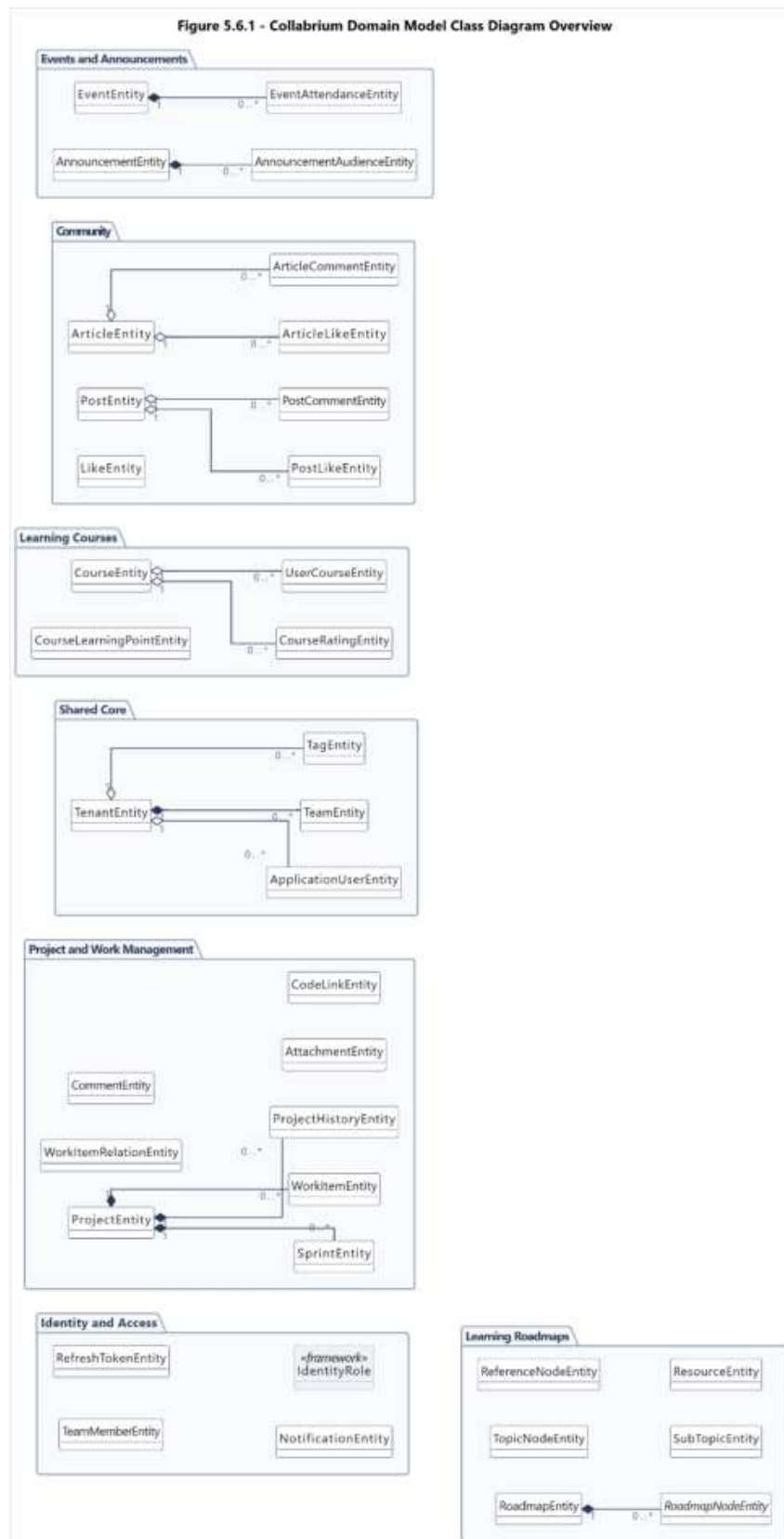


Figure 5.6.1 Collabrium Domain Model Class Diagram Overview

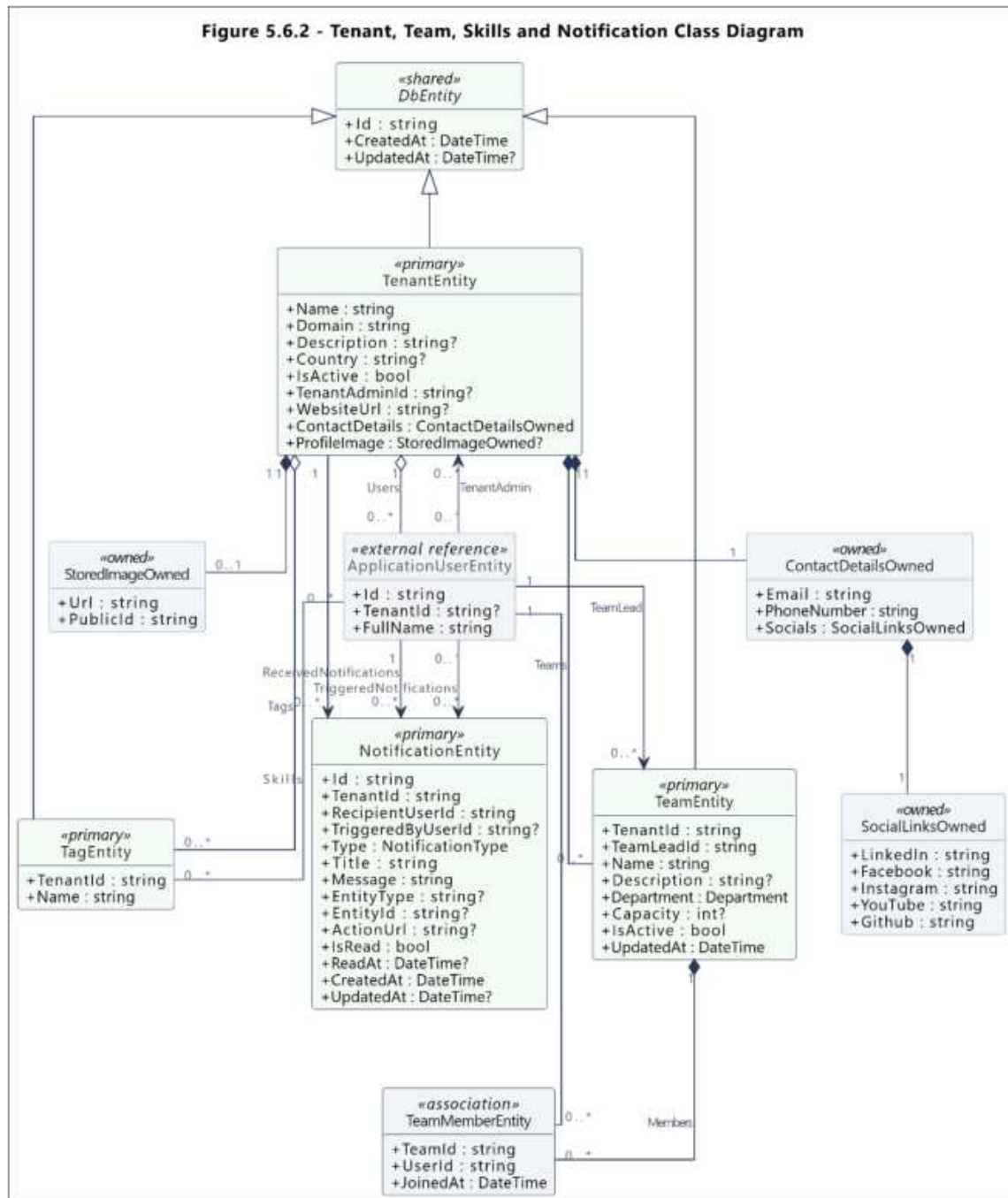


Figure 5.6 2 Tenant, Team, Skills and Notification Class Diagram

**Figure 5.6.3**  
**Identity and Access Management Class Diagram**

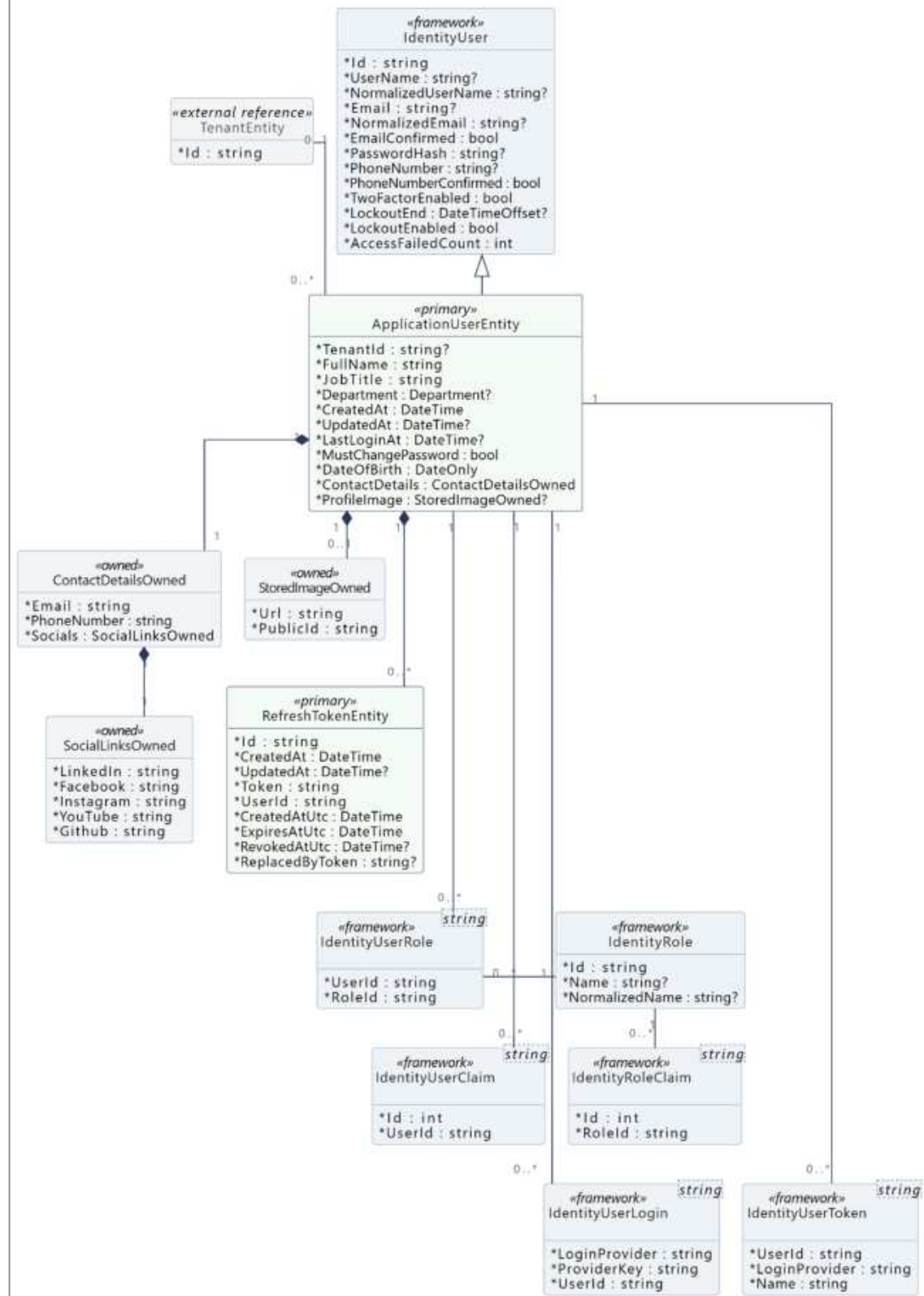


Figure 5.6.3 Identity and Access Management Class Diagram

**Figure 5.6.4 - Project, Sprint and History Class Diagram**

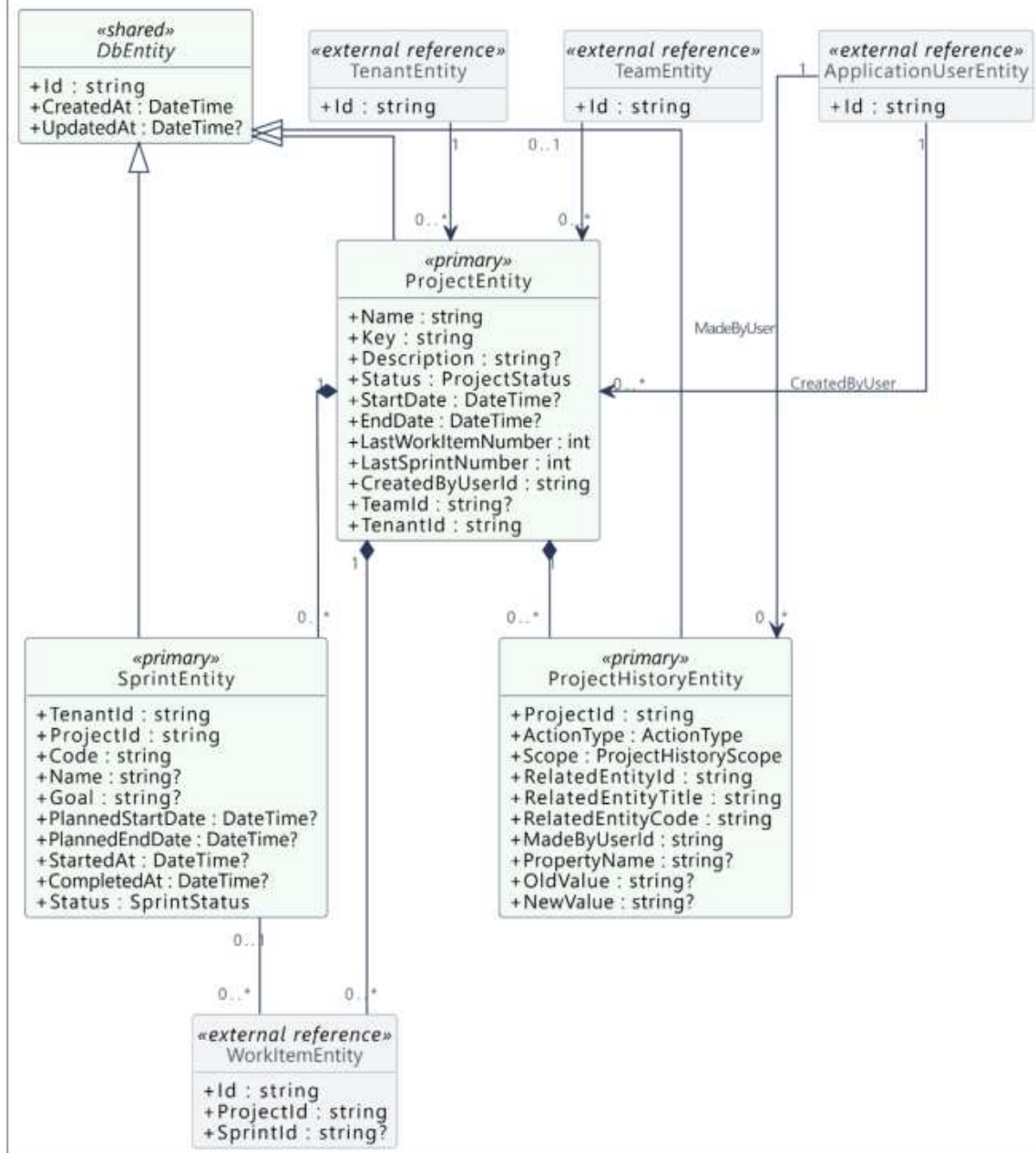


Figure 5.6.4 Projects, Sprint and History Class Diagram

**Figure 5.6.5 - Work Item Core and Relationship Class Diagram**

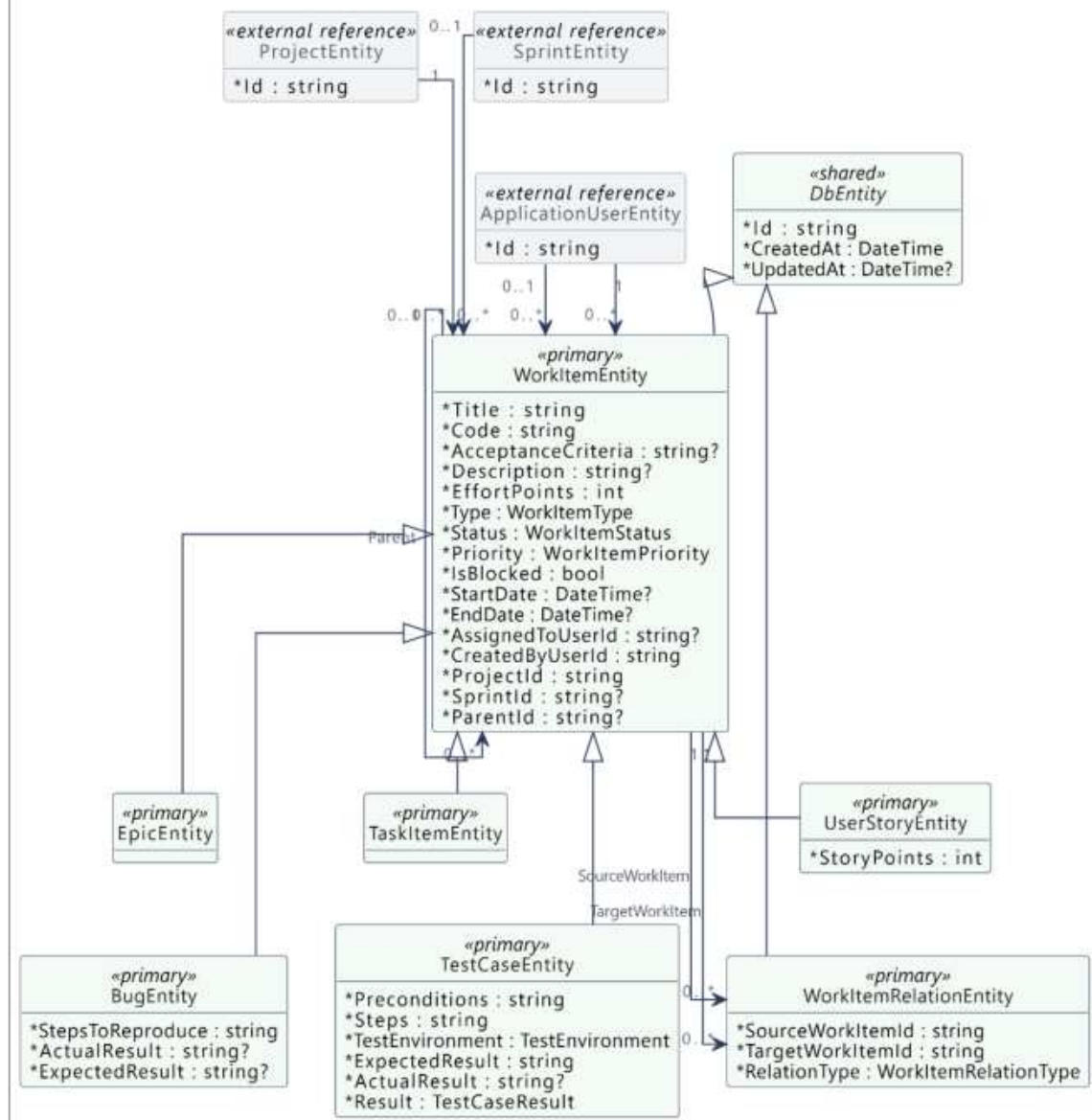


*Figure 5.6.5 Work Item Core and Relationship Class Diagram*

**Figure 5.6.6 - Work Item Collaboration and Supporting Artifacts Class Diagram**

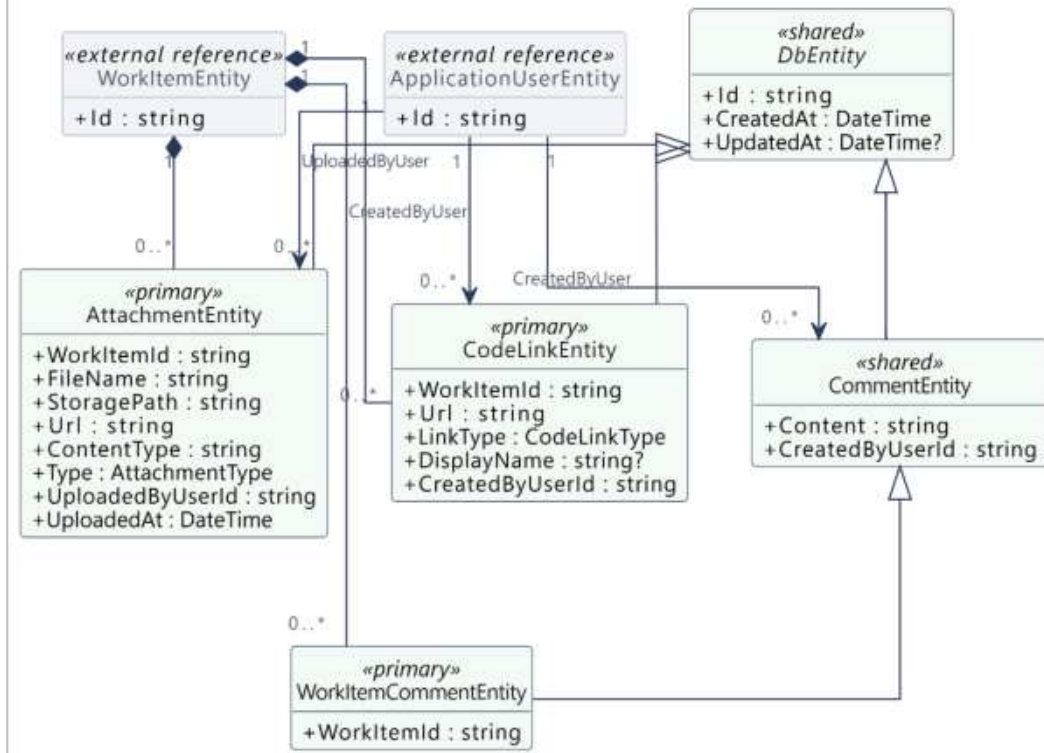


Figure 5.6.6 Work Item Collaboration and Supporting Artifacts Class Diagram

**Figure 5.6.7 - Learning Courses and User Progress Class Diagram**

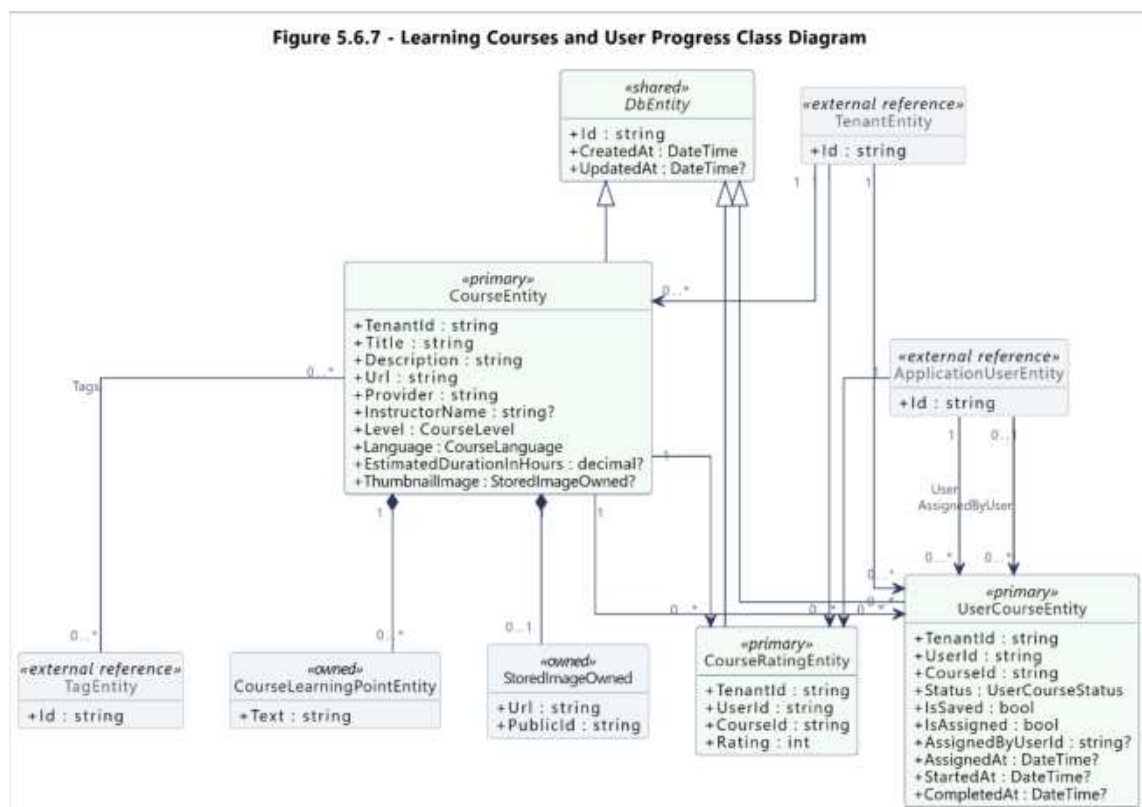


Figure 5.6.7 Learning Courses and User Progress Class Diagram

**Figure 5.6.8 - Learning Roadmaps and Resources Class Diagram**

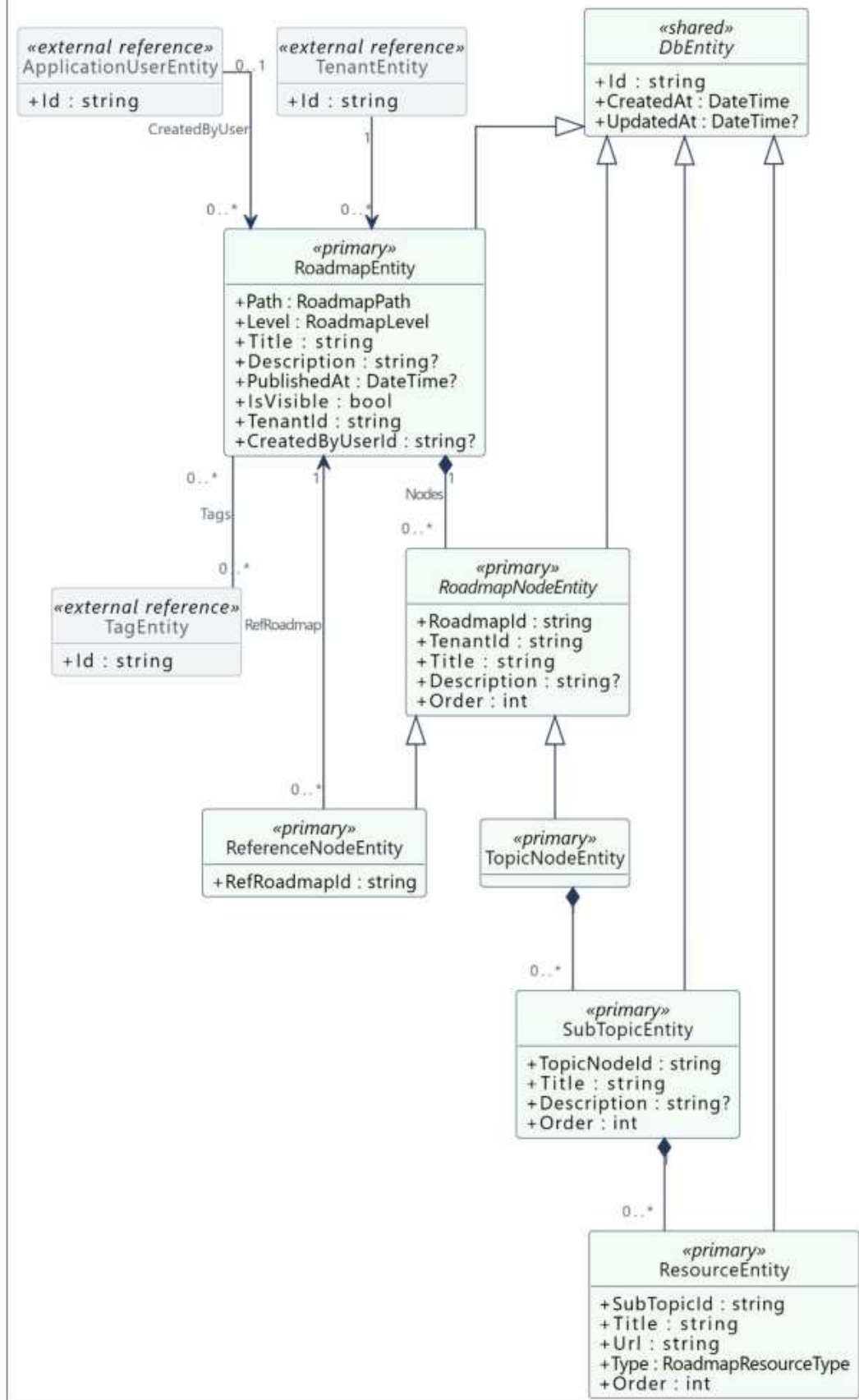


Figure 5.6.8 Learning Roadmaps and Resources Class Diagram

Figure 5.6.9 - Community Articles and Posts Class Diagram

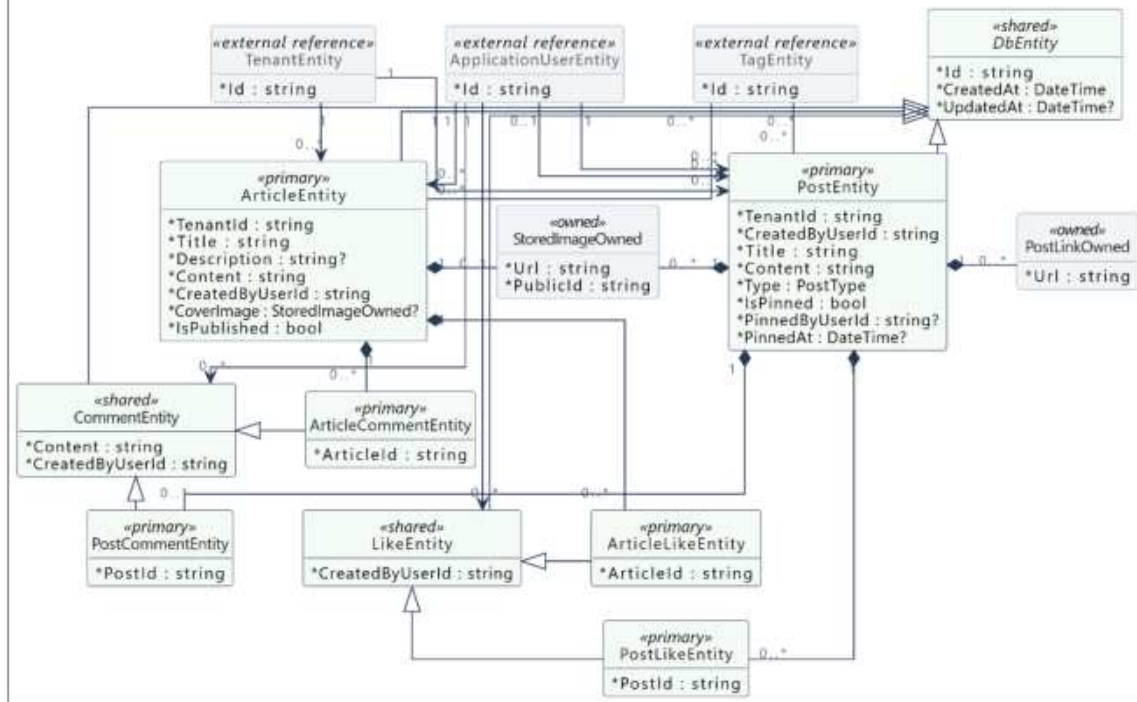


Figure 5.6.9 Community Articles and Posts Class Diagram

Figure 5.6.10 - Events and Announcements Class Diagram

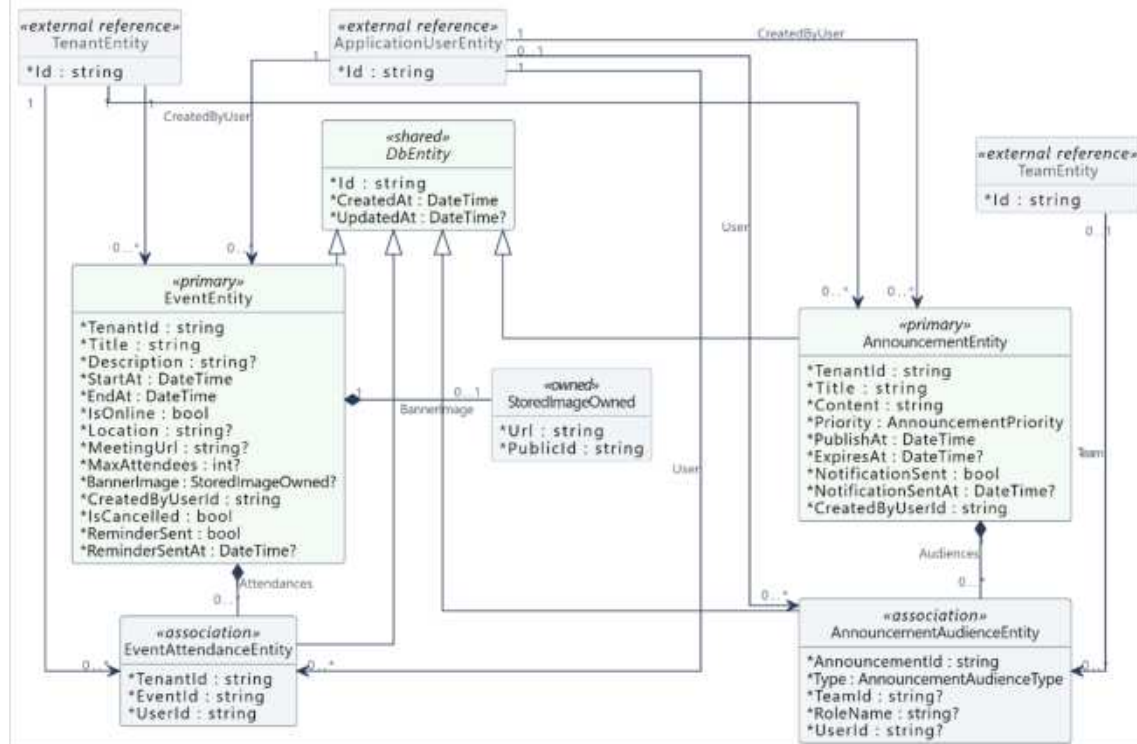


Figure 5.6.10 Events and Announcements Class Diagram

# 5.7 Activity Diagrams

## 5.7.1 Authentication Subsystem Activity Diagrams

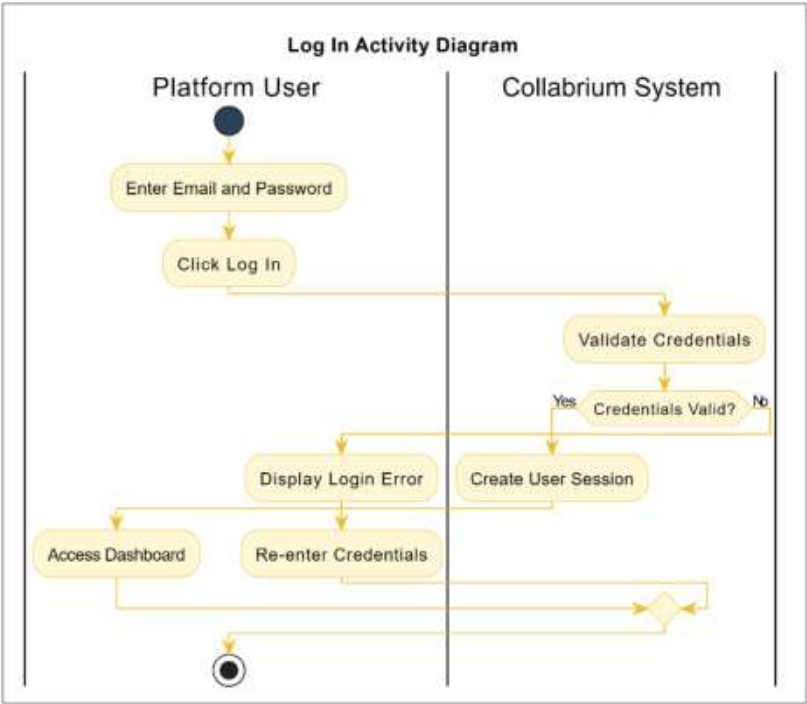


Figure 5.7.1 1 Login Activity Diagram

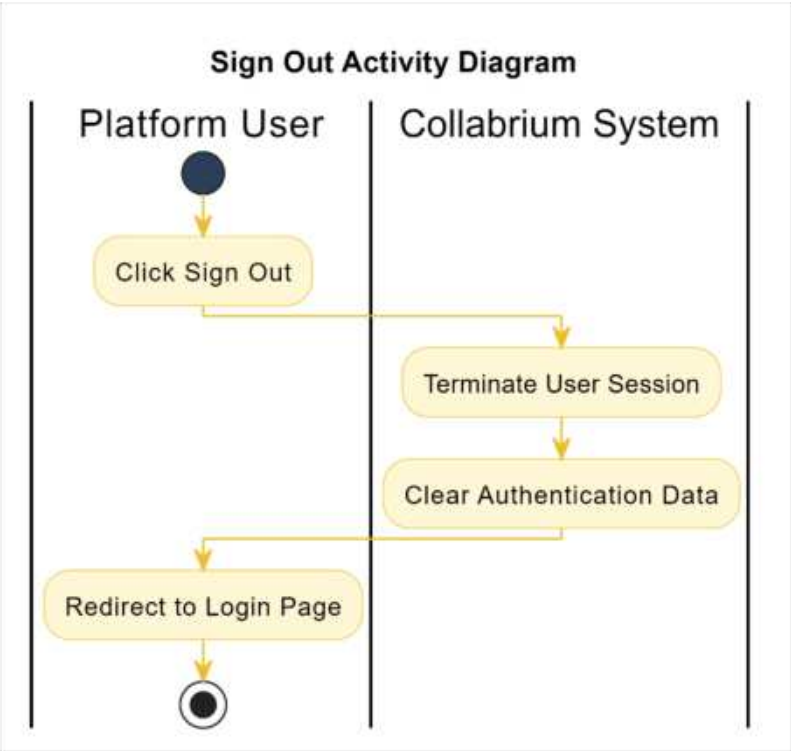


Figure 5.7.1 2 Sign Out Activity Diagram

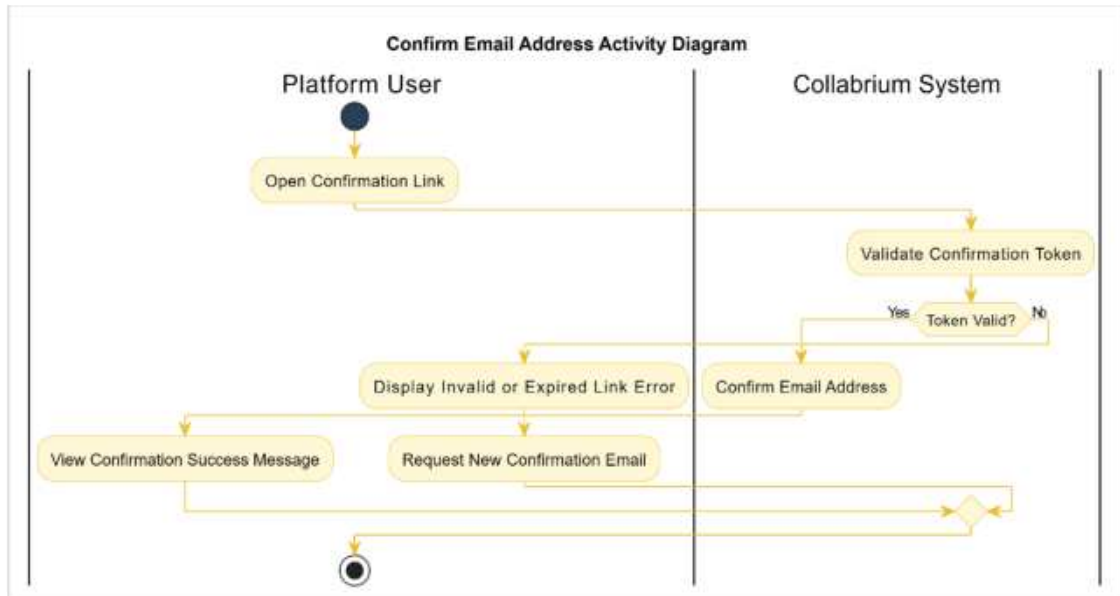


Figure 5.7.1 3 Confirm Email Address Activity Diagram

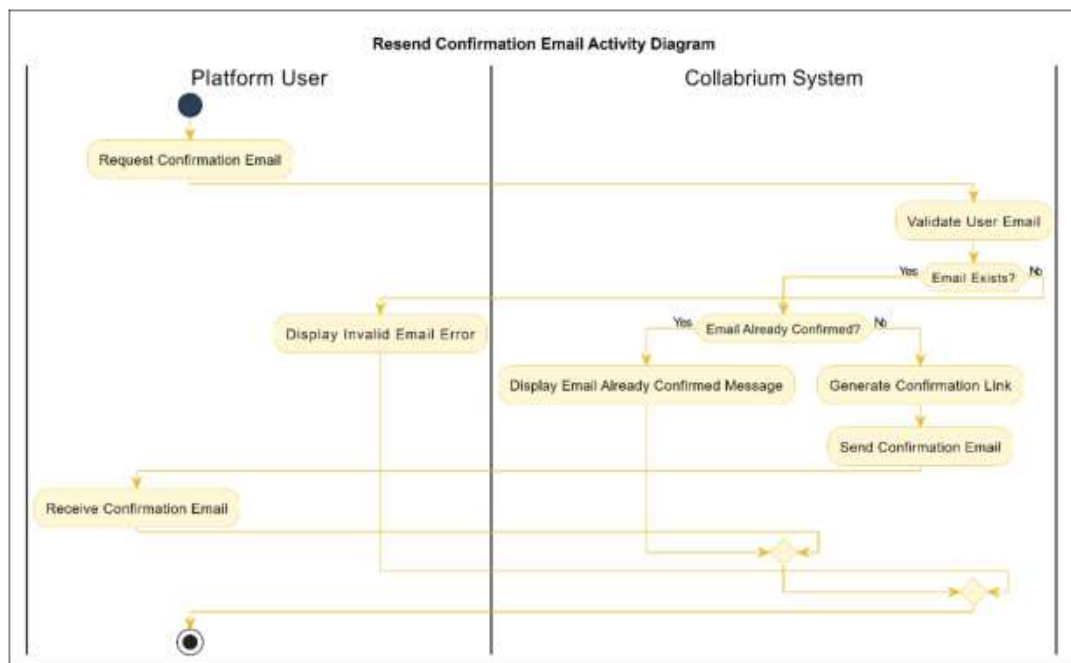


Figure 5.7.1 4 Resend Confirmation Email Activity Diagram

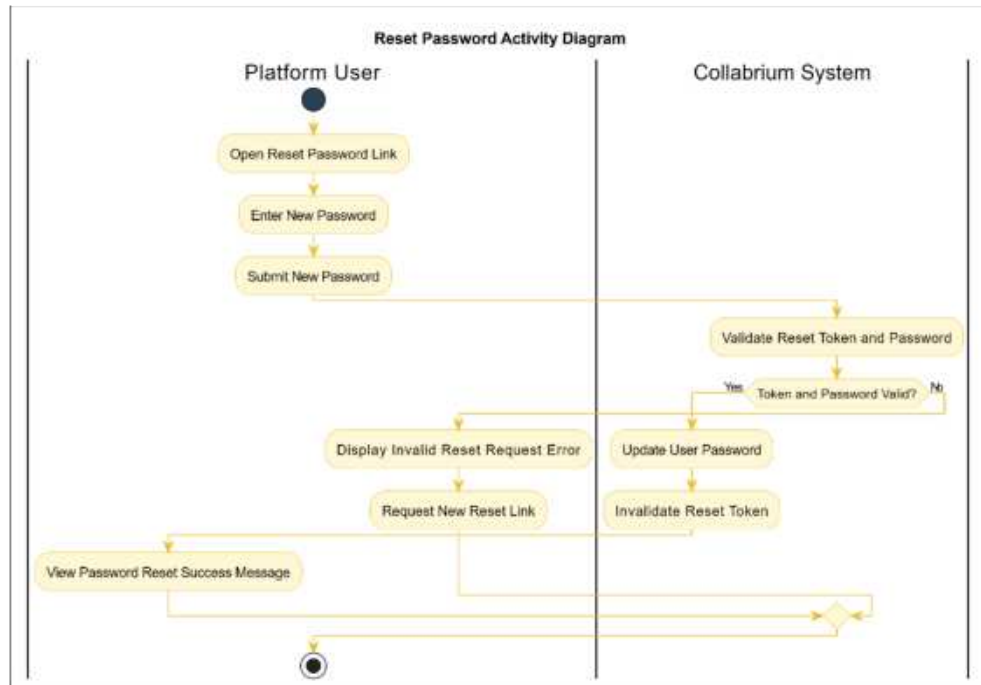


Figure 5.7.1 5 Reset Password Activity Diagram

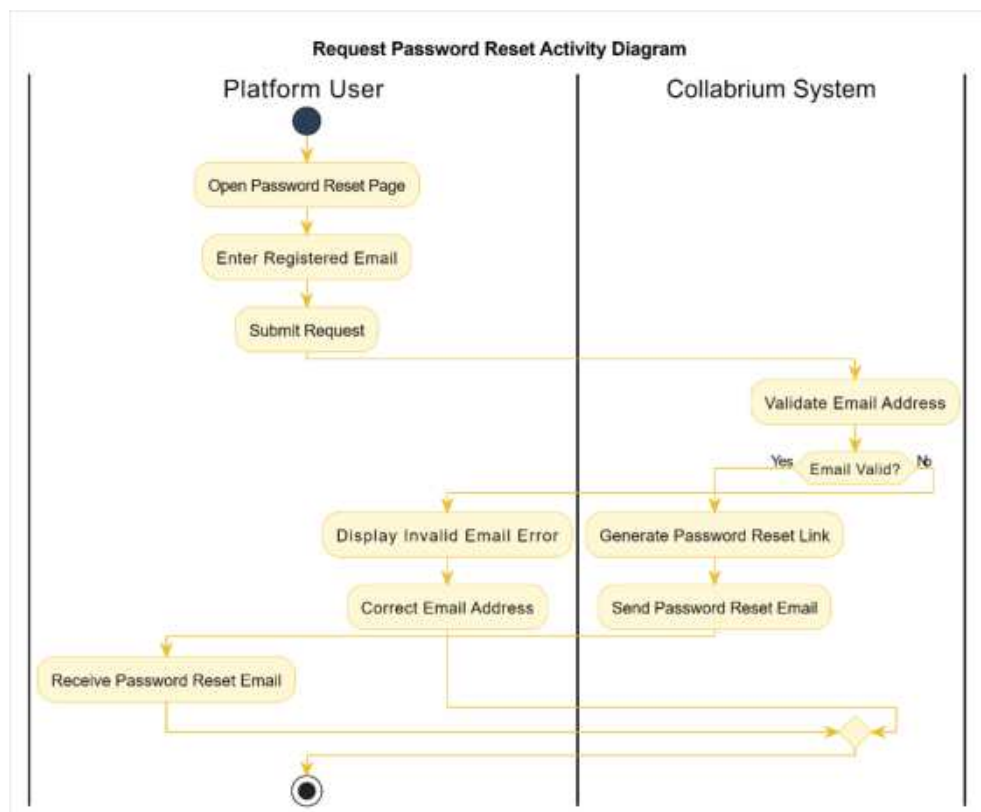


Figure 5.7.1 6 Request Password Reset Activity Diagram

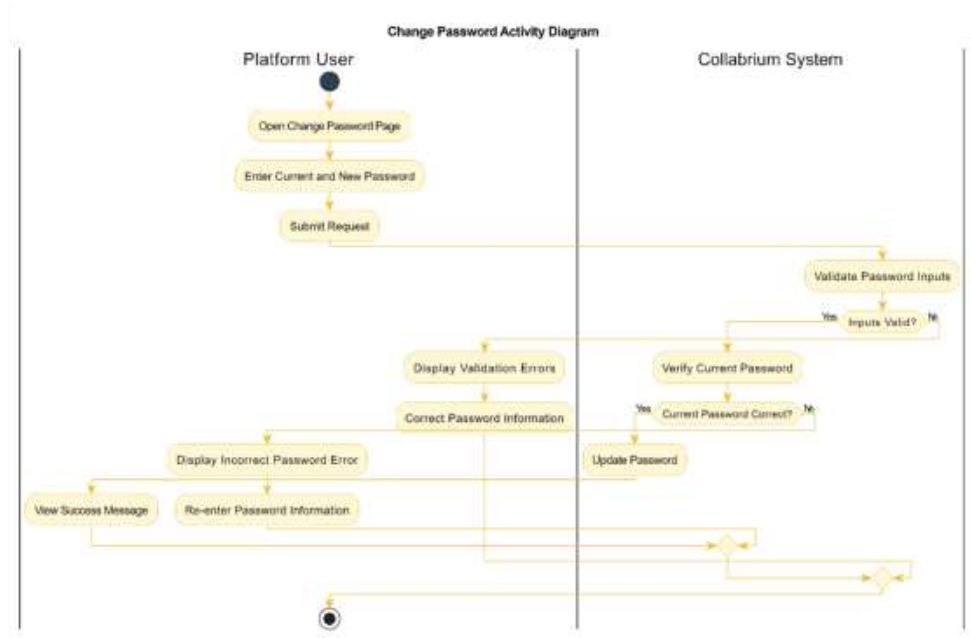


Figure 5.7.1 7 Change Password Activity Diagram

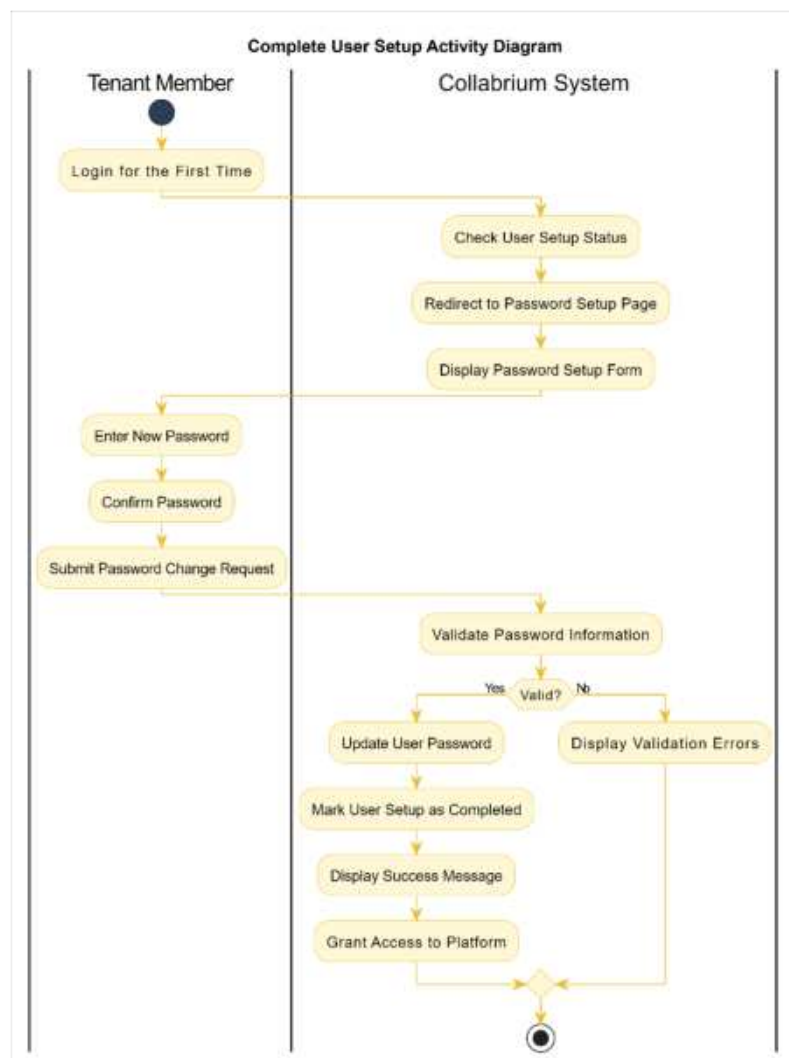


Figure 5.7.1 8 Complete User Setup Activity Diagram

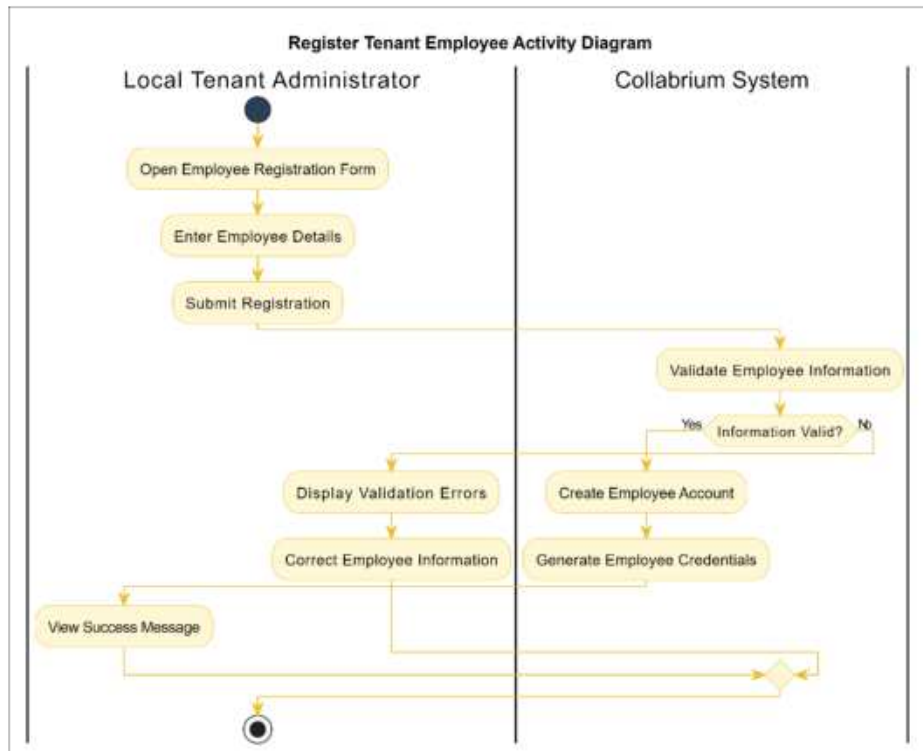


*Figure 5.7.1 9 Register Tenant Employee Activity Diagram*

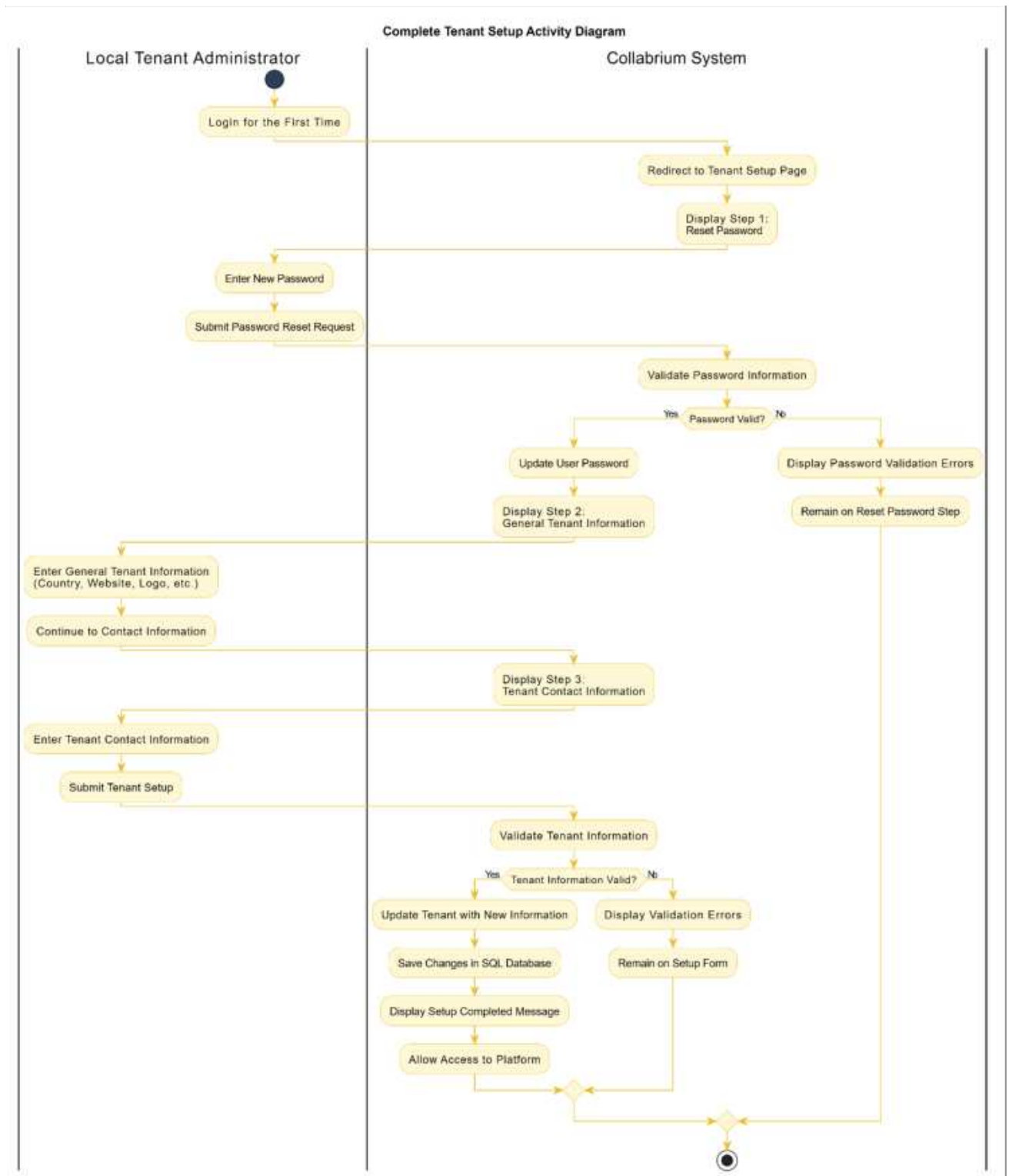


Figure 5.7.1 10 Complete Tenant Setup Activity Diagram

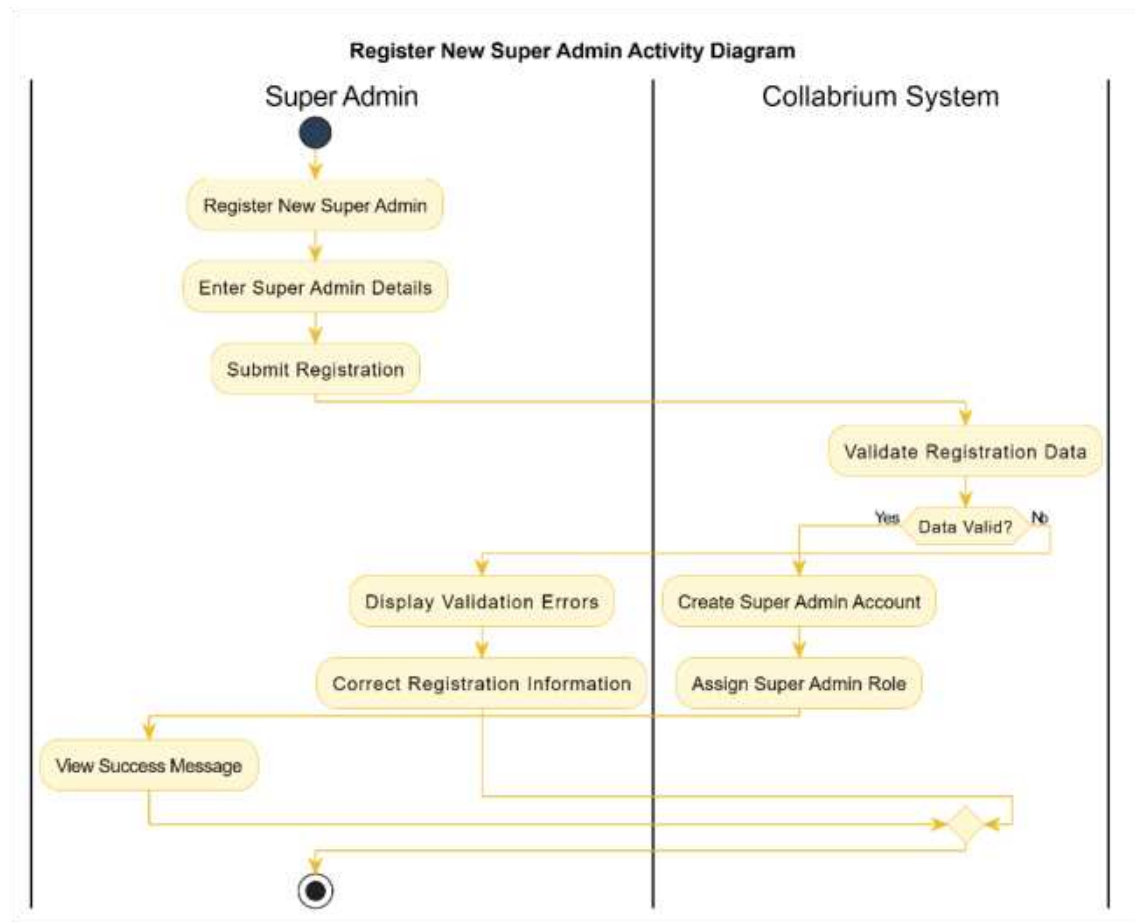


*Figure 5.7.1 11 Register New Super Admin Activity Diagram*

### 5.7.2 Platform Services Subsystem Activity Diagrams

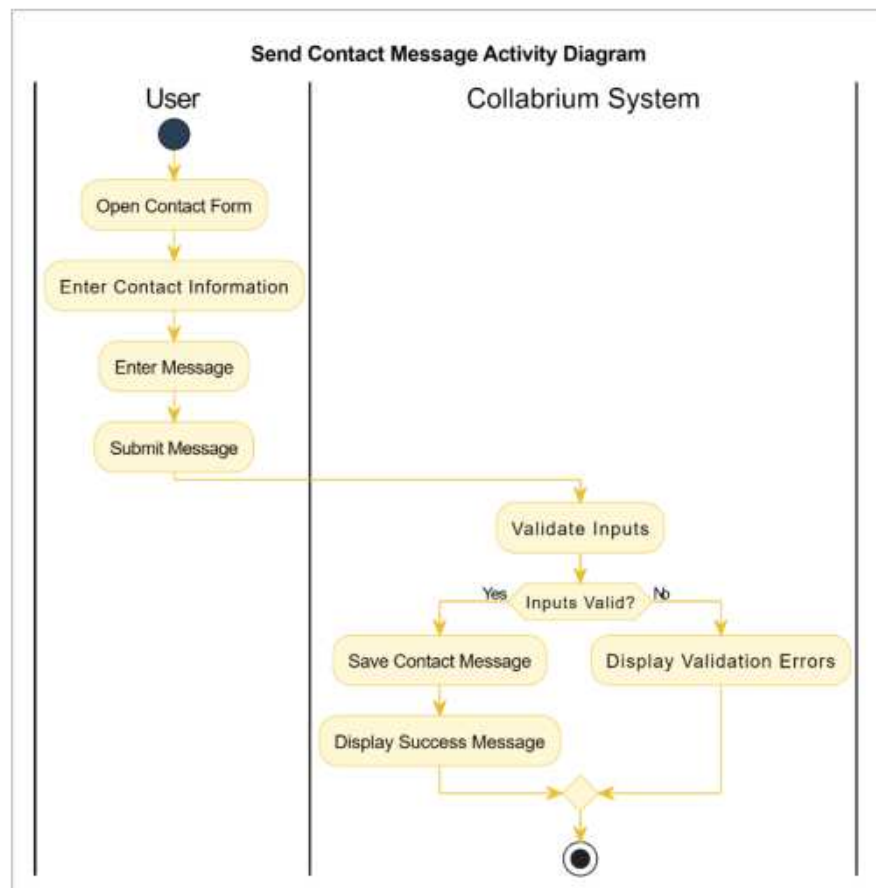


Figure 5.7.2 1 Send Contact Message Activity Diagram

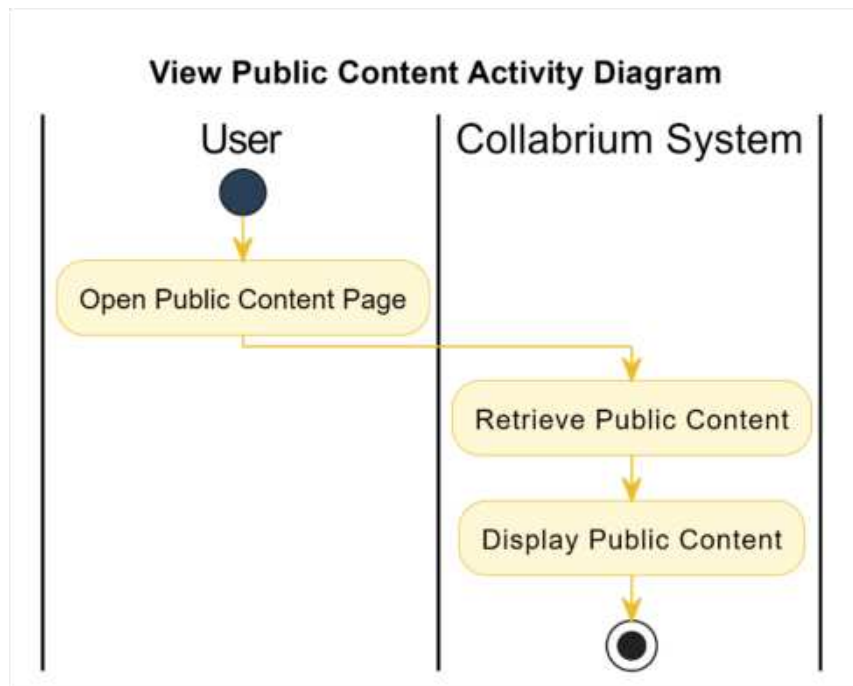


Figure 5.7.2 2 View Public Content Activity Diagram

### 5.7.3 Tenant Management Subsystem Activity Diagrams

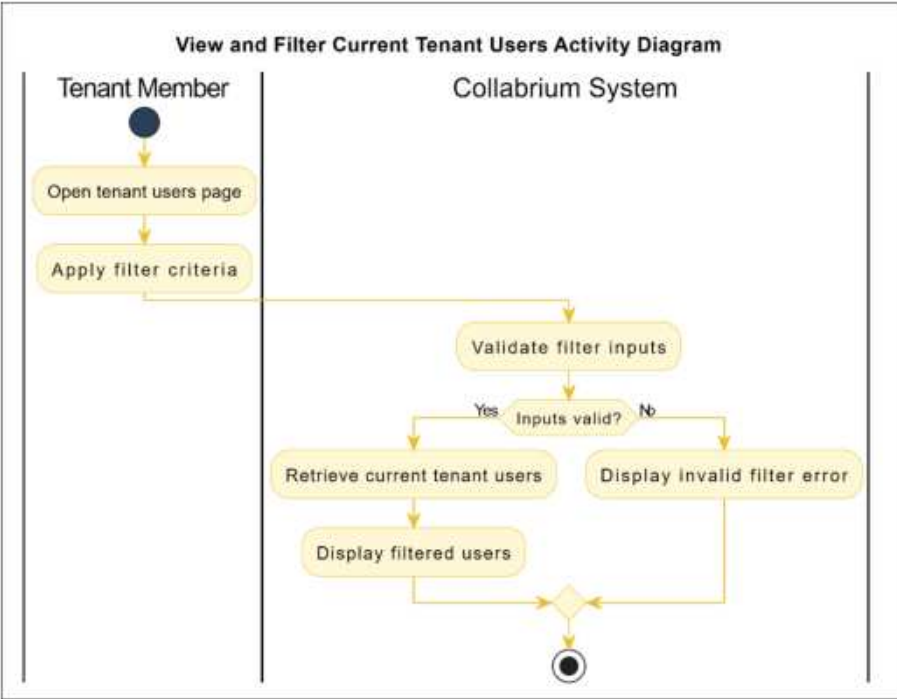


Figure 5.7.3 1 View and Filter Current Tenant Users Activity Diagram

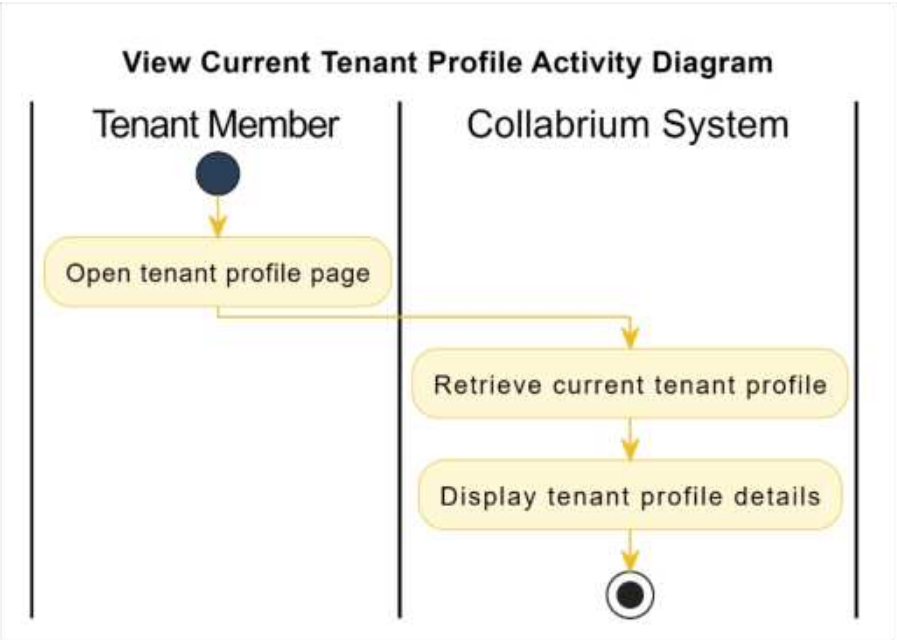


Figure 5.7.3 2 View Current Tenant Profile Activity Diagram

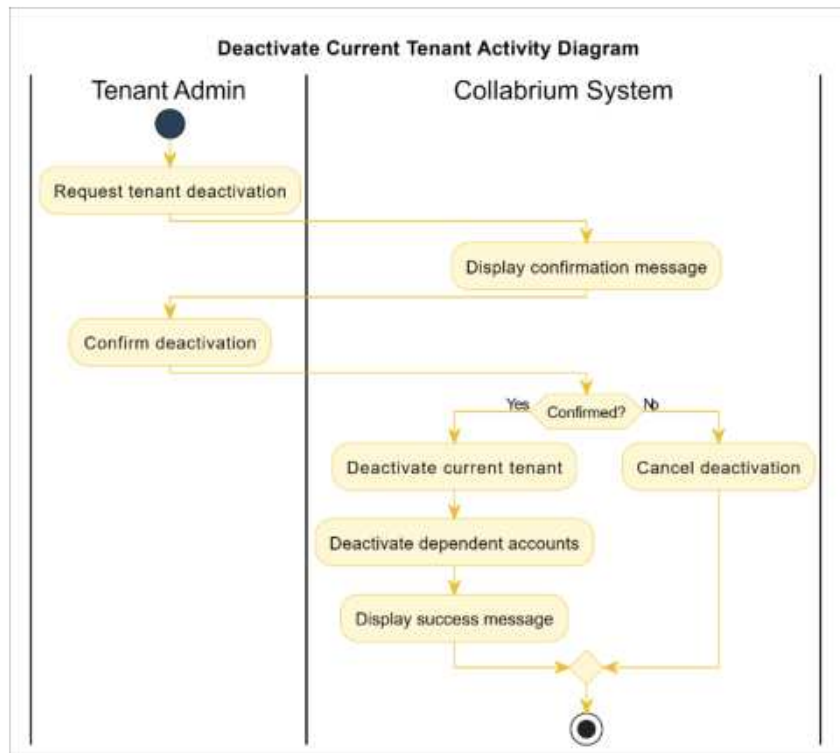


Figure 5.7.3 3 Deactivate Current Tenant Activity Diagram

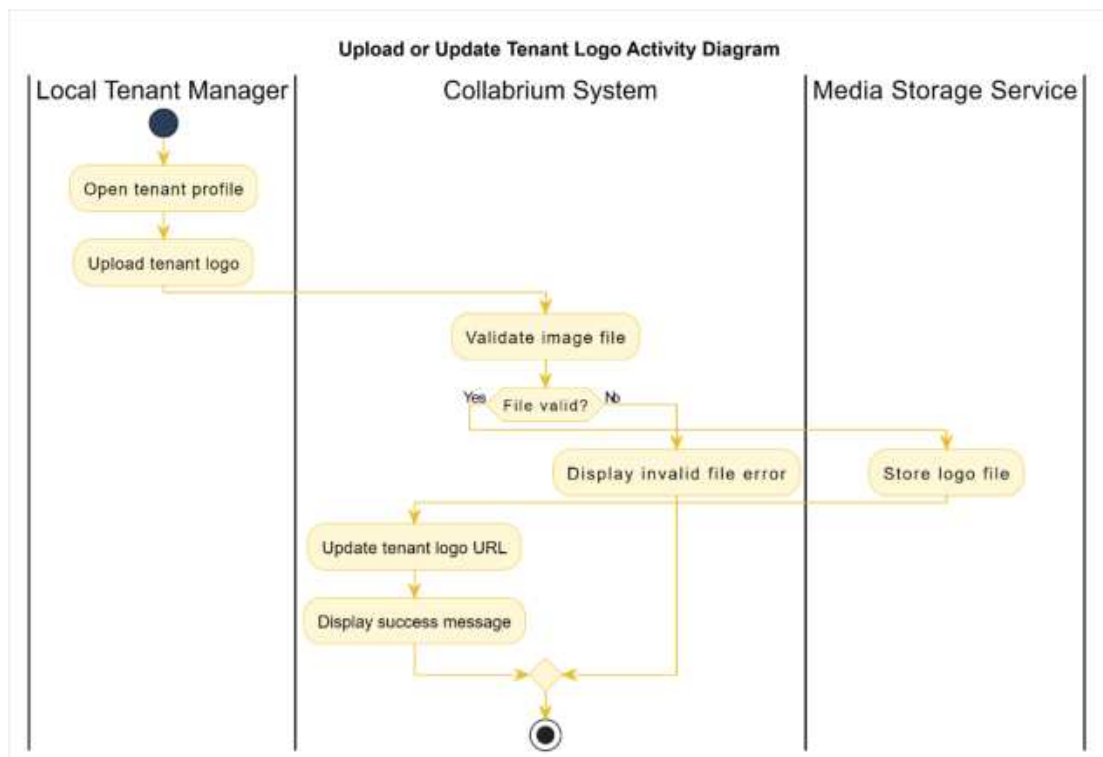


Figure 5.7.3 4 Upload or Update Tenant Logo Activity Diagram

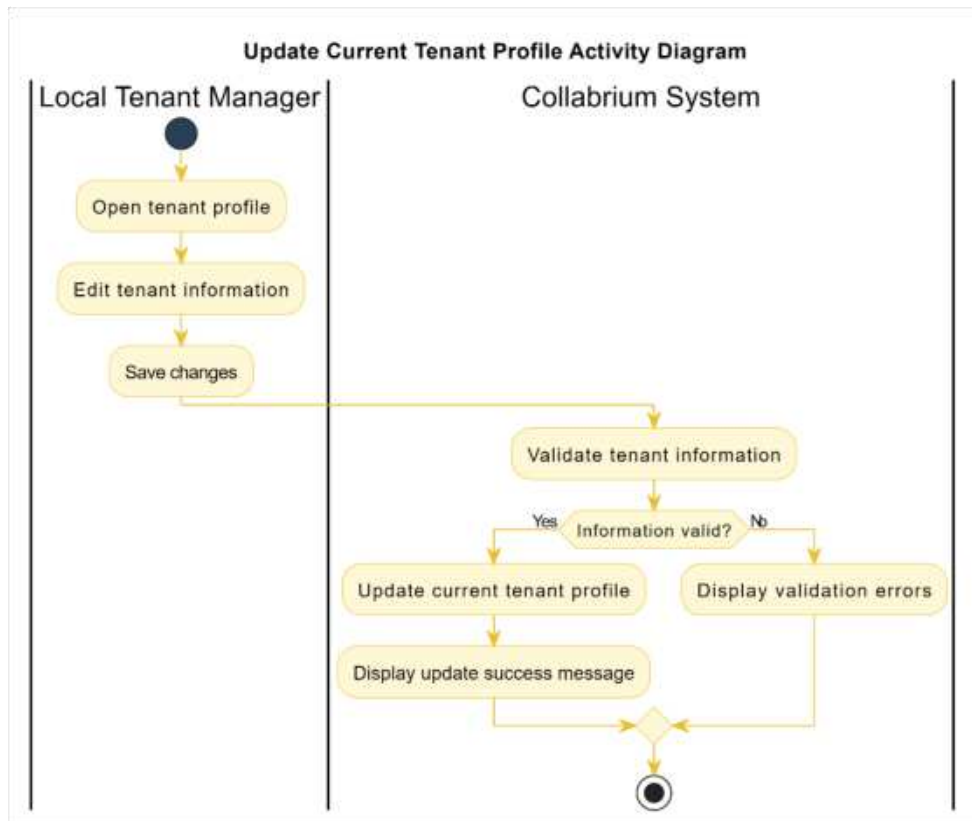


Figure 5.7.3 5 Update Current Tenant Profile Activity Diagram

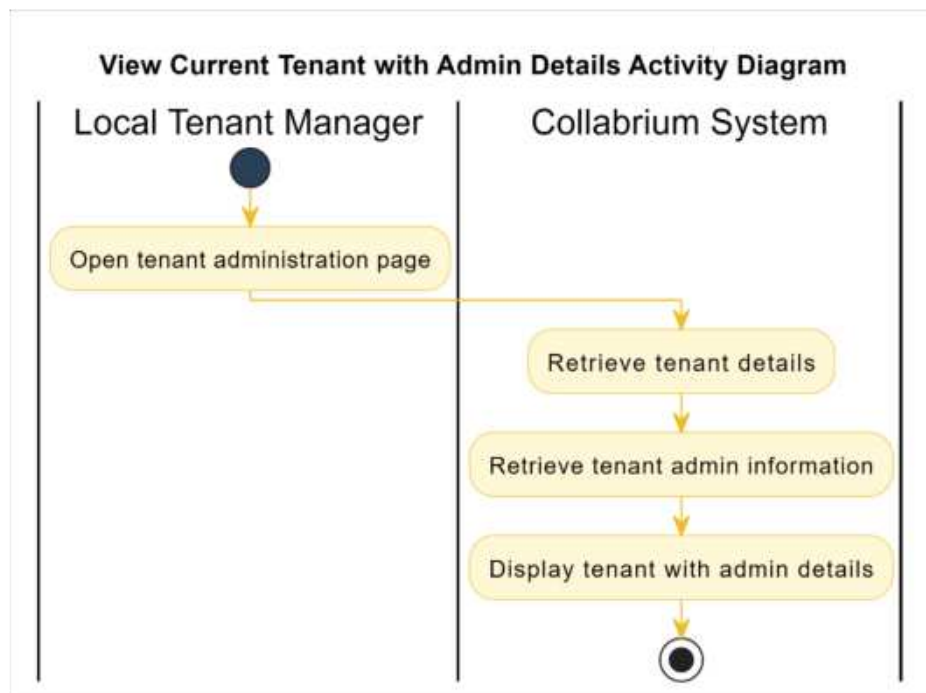


Figure 5.7.3 6 View Current Tenant with Admin Details Activity Diagram

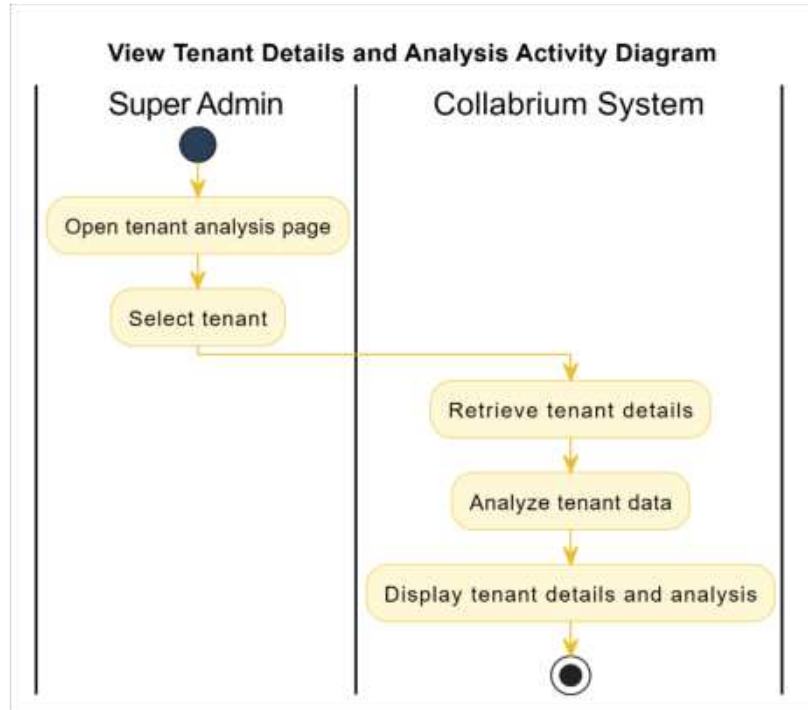


Figure 5.7.3 7 View Tenant Details and Analysis Activity Diagram

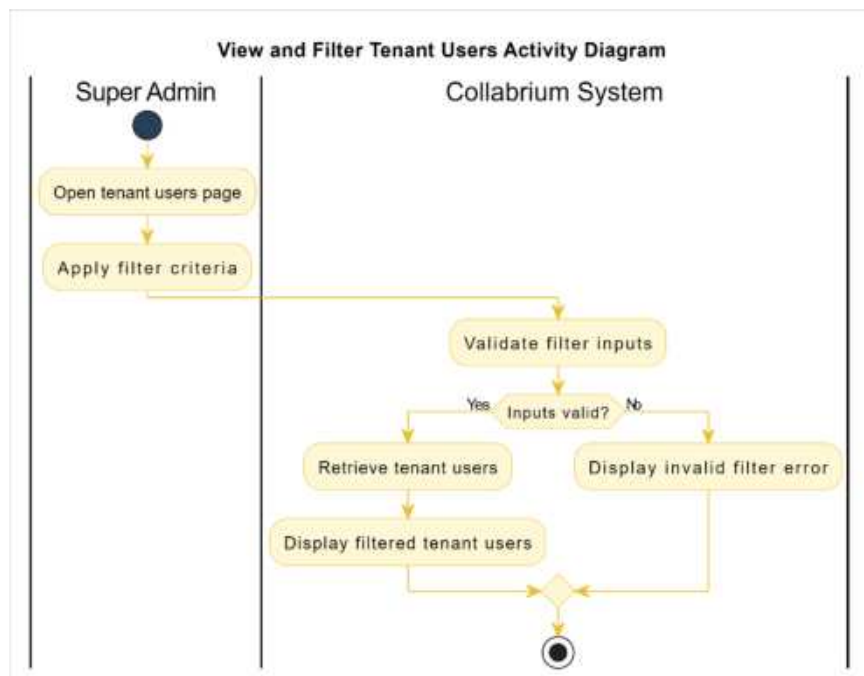


Figure 5.7.3 8 View and Filter Tenant Users Activity Diagram

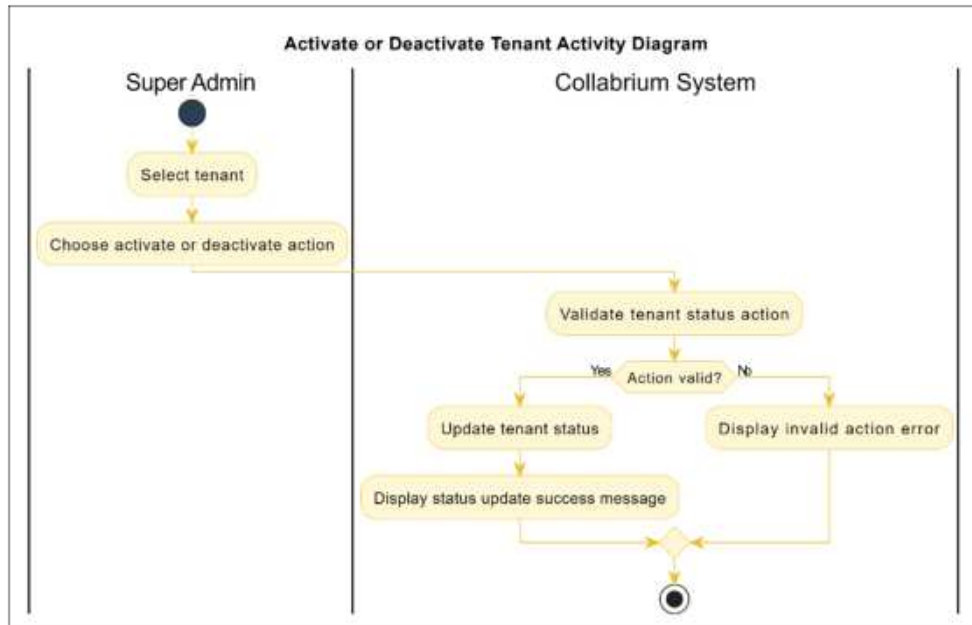


Figure 5.7.3 9 Activate or Deactivate Tenant Activity Diagram

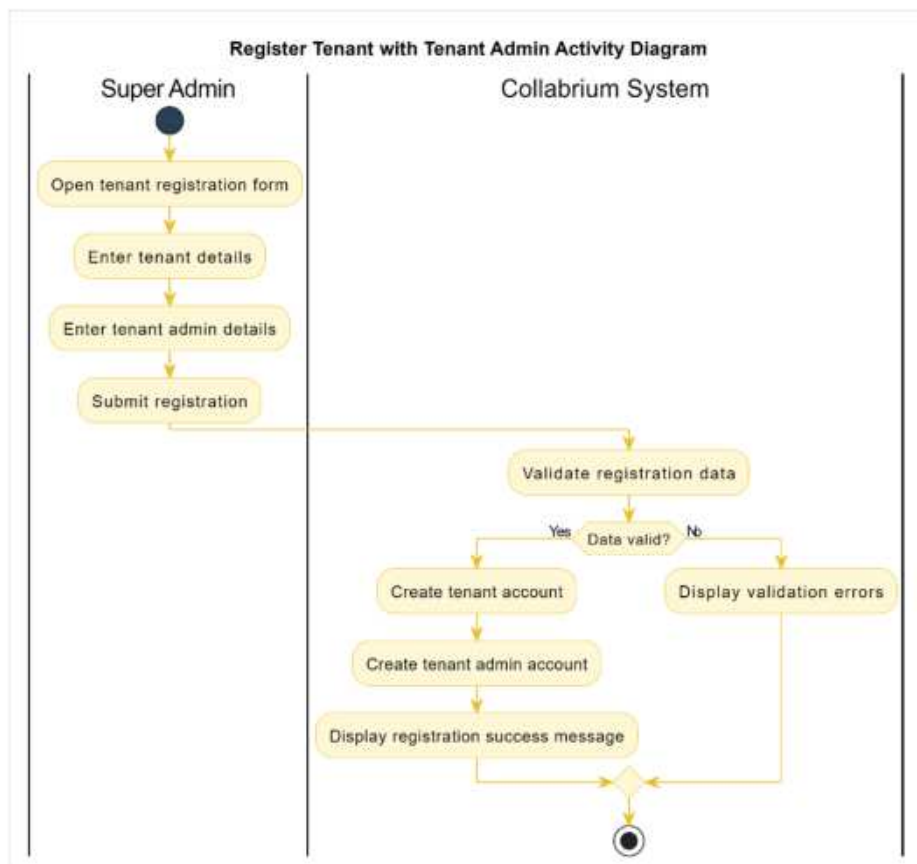


Figure 5.7.3 10 Register Tenant with Tenant Admin Activity Diagram

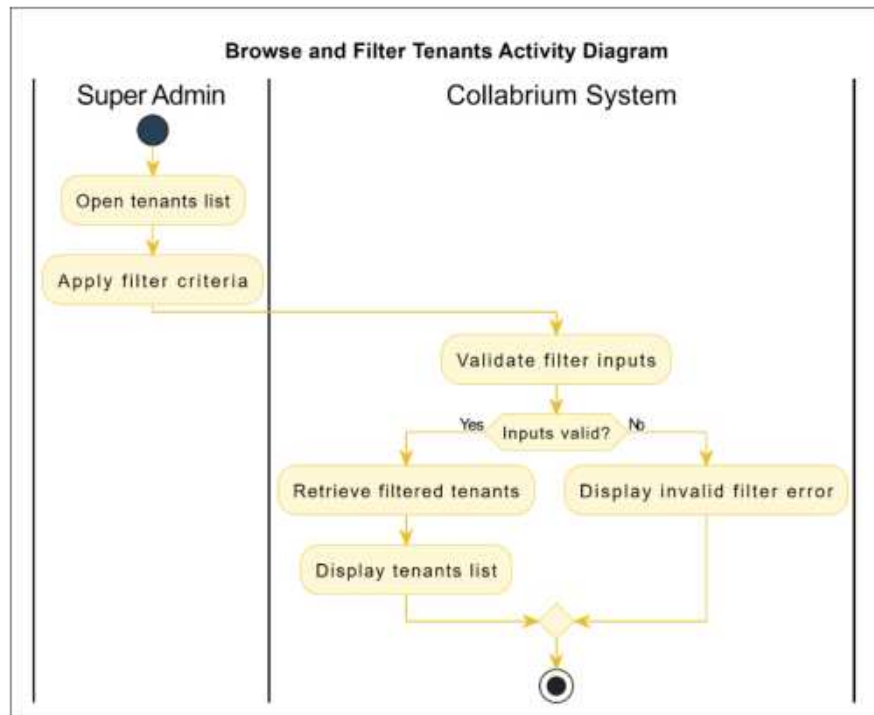


*Figure 5.7.3 11 Browse and Filter Tenants Activity Diagram*

## 5.7.4 User Management Subsystem Activity Diagrams

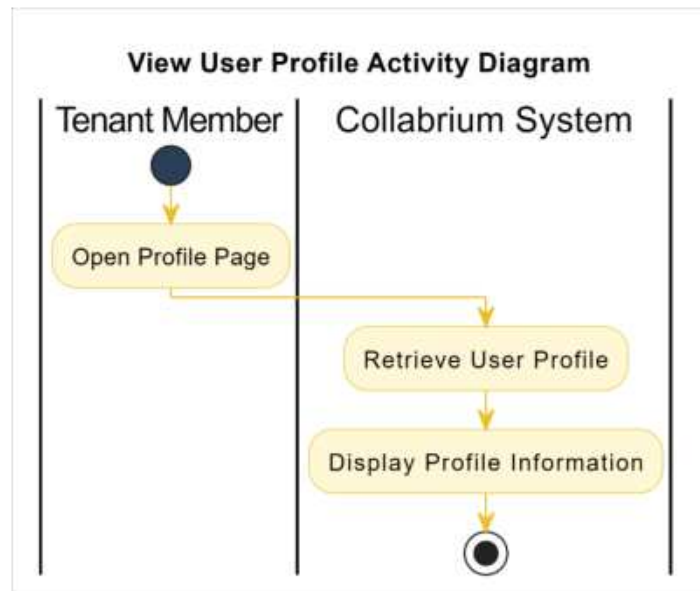


Figure 5.7.4 1 View User Profile Activity Diagram

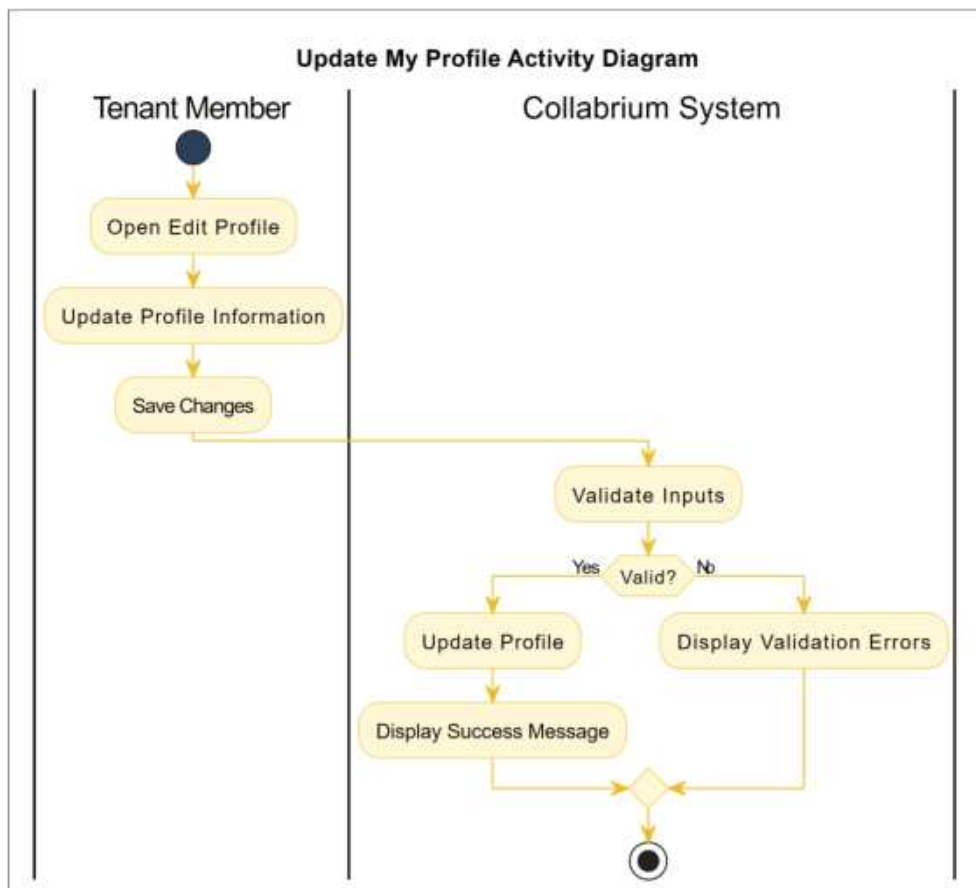


Figure 5.7.4 2 Update My Profile Activity Diagram

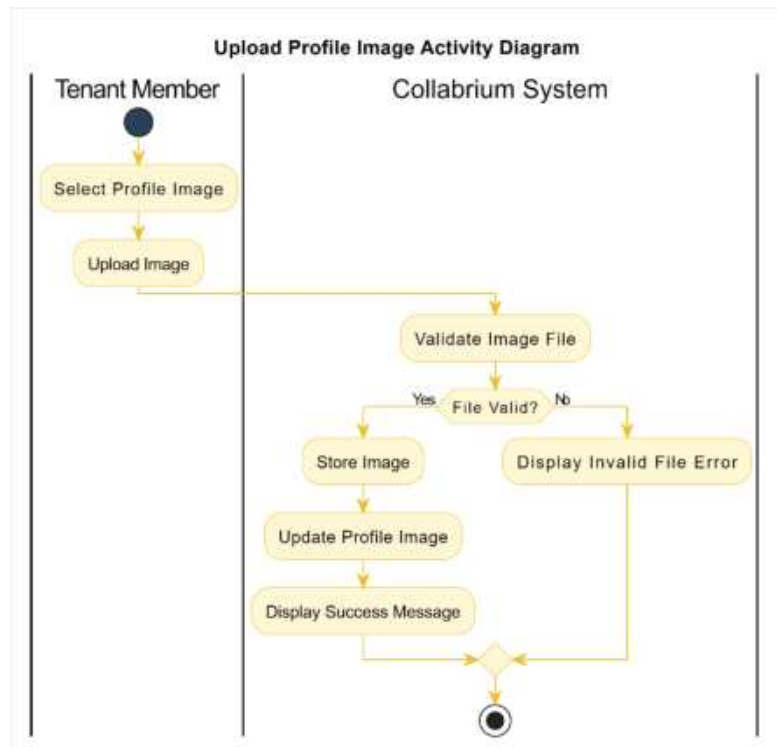


*Figure 5.7.4 3 Upload Profile Image Activity Diagram*

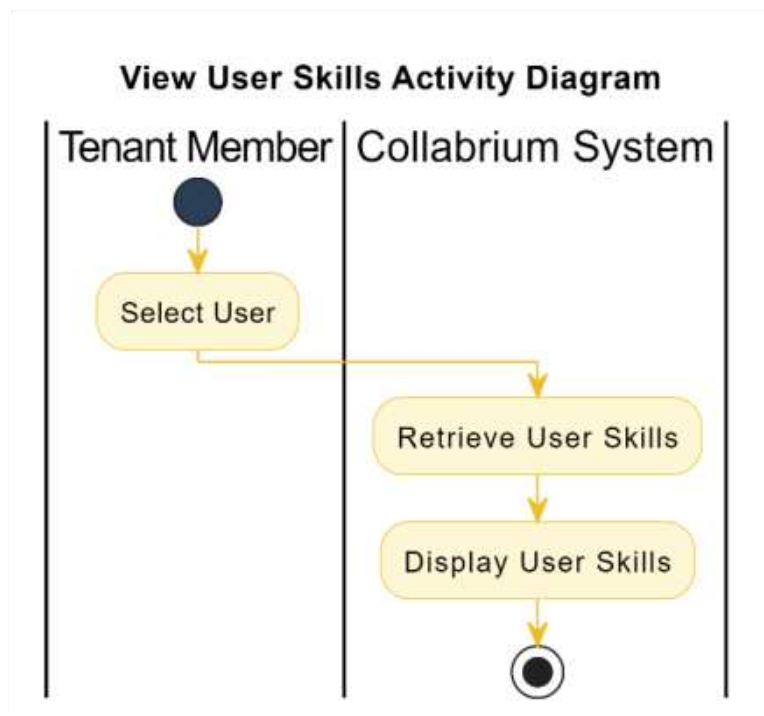


*Figure 5.7.4 4 View User Skills Activity Diagram*

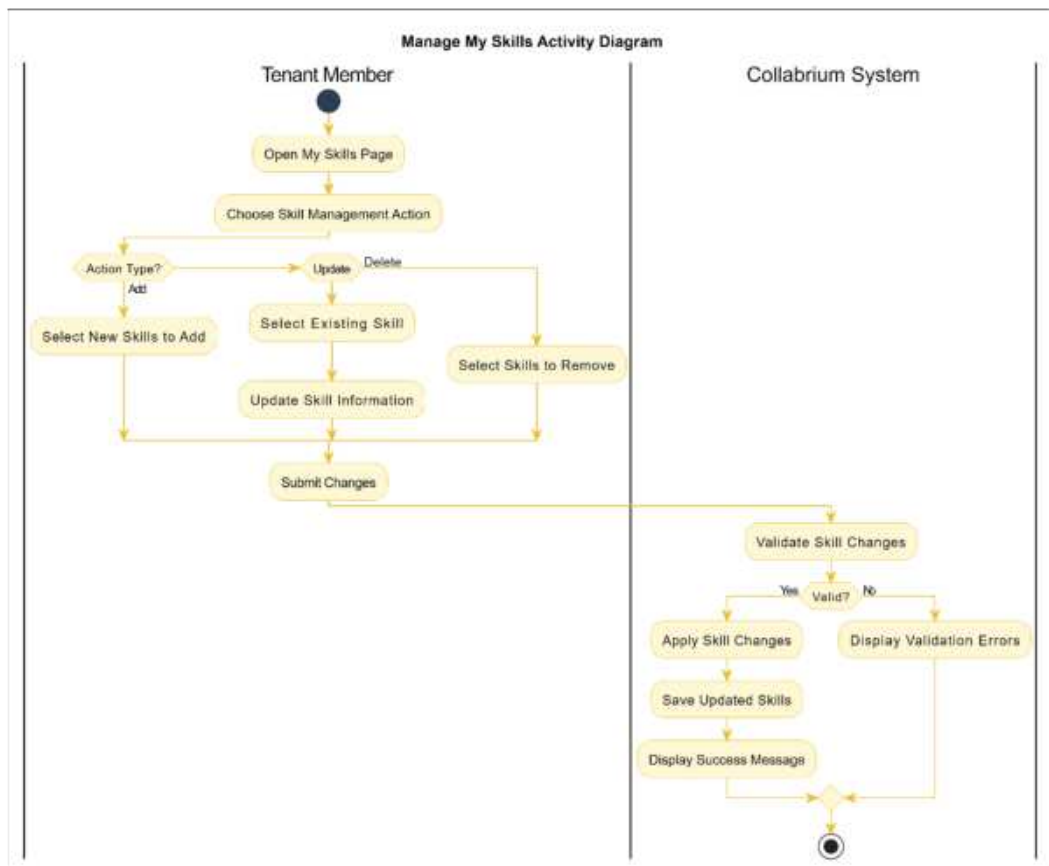


Figure 5.7.4 5 Manage My Skills Activity Diagram

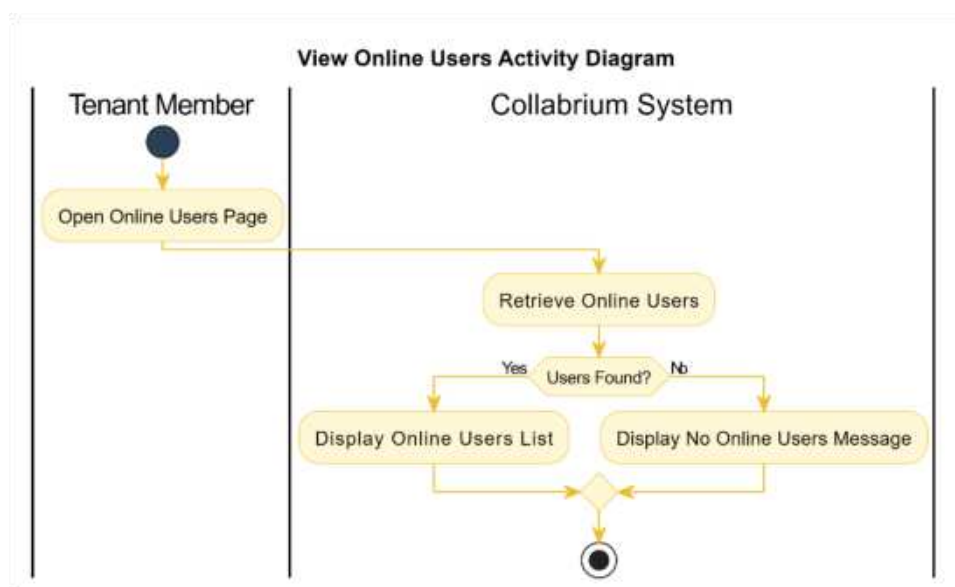


Figure 5.7.4 6 View Online Users Activity Diagram

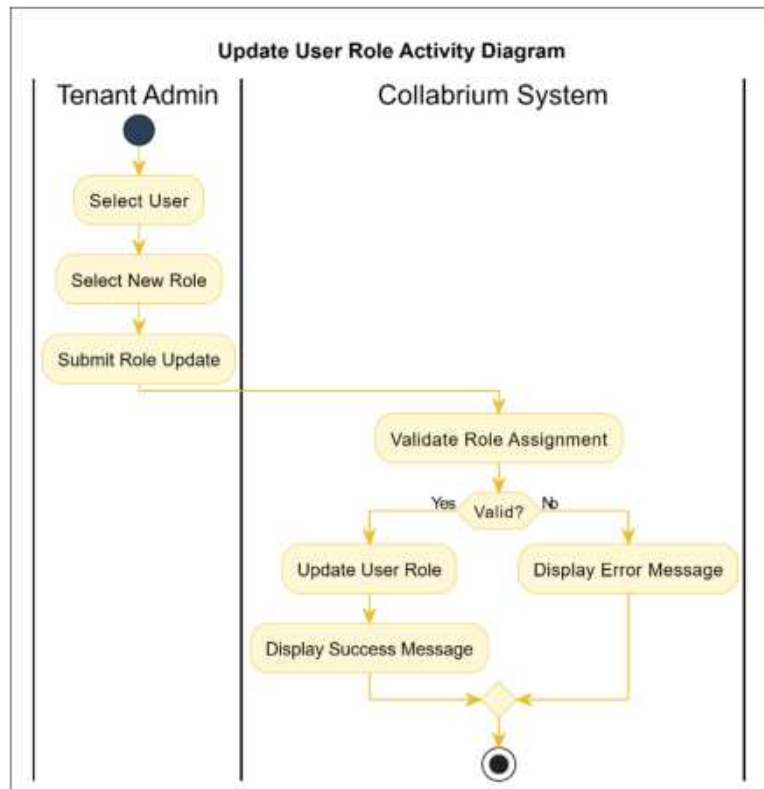


*Figure 5.7.4 7 Update User Role Activity Diagram*

### 5.7.5 Team Management Subsystem Activity Diagrams

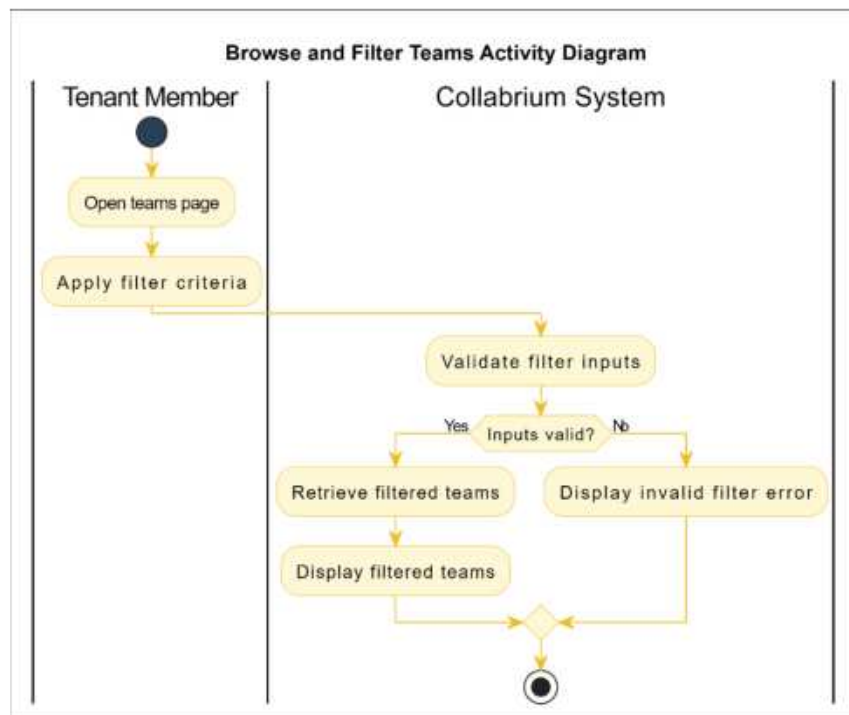


Figure 5.7.5 1 Browse and Filter Teams Activity Diagram

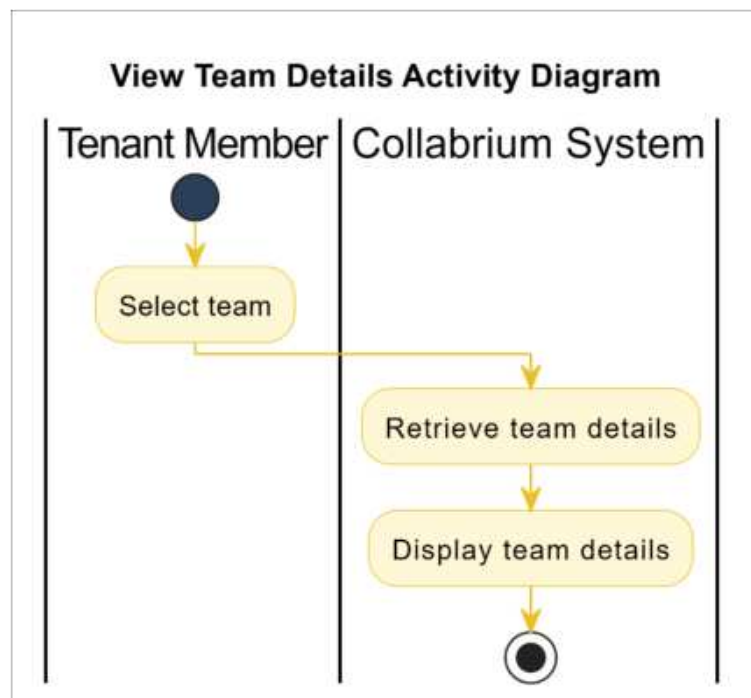


Figure 5.7.5 2 View Team Details Activity Diagram

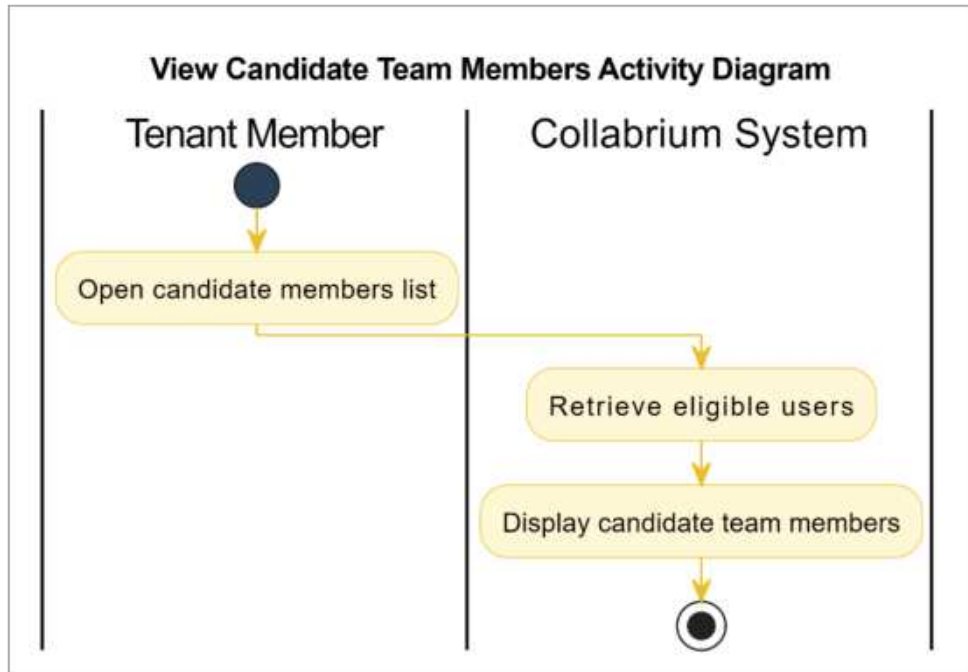


*Figure 5.7.5 3 View Candidate Team Members Activity Diagram*

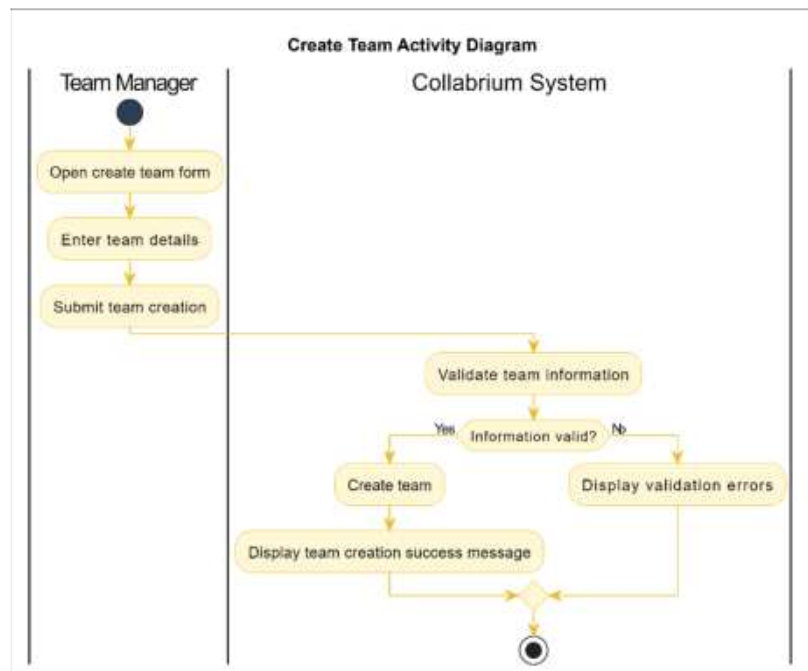


*Figure 5.7.5 4 Create Team Activity Diagram*

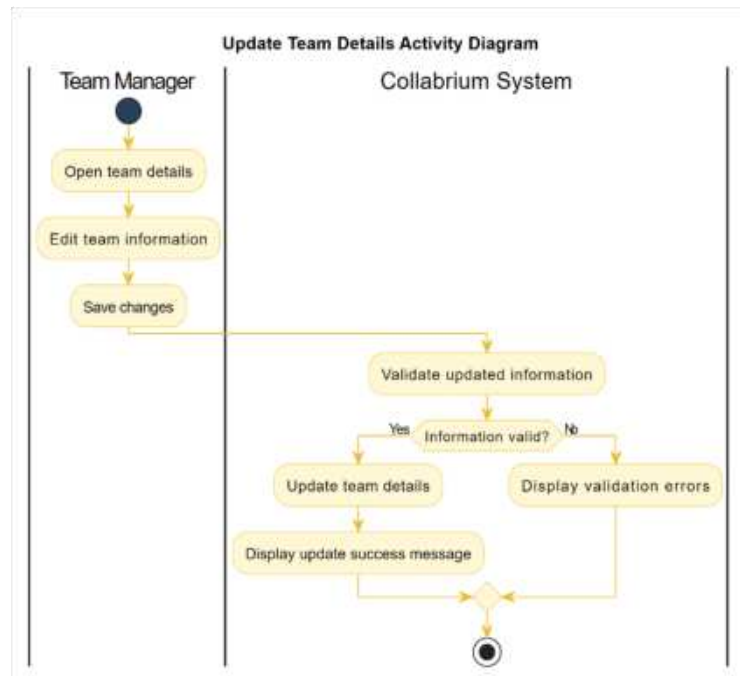


Figure 5.7.5 5 Update Team Details Activity Diagram

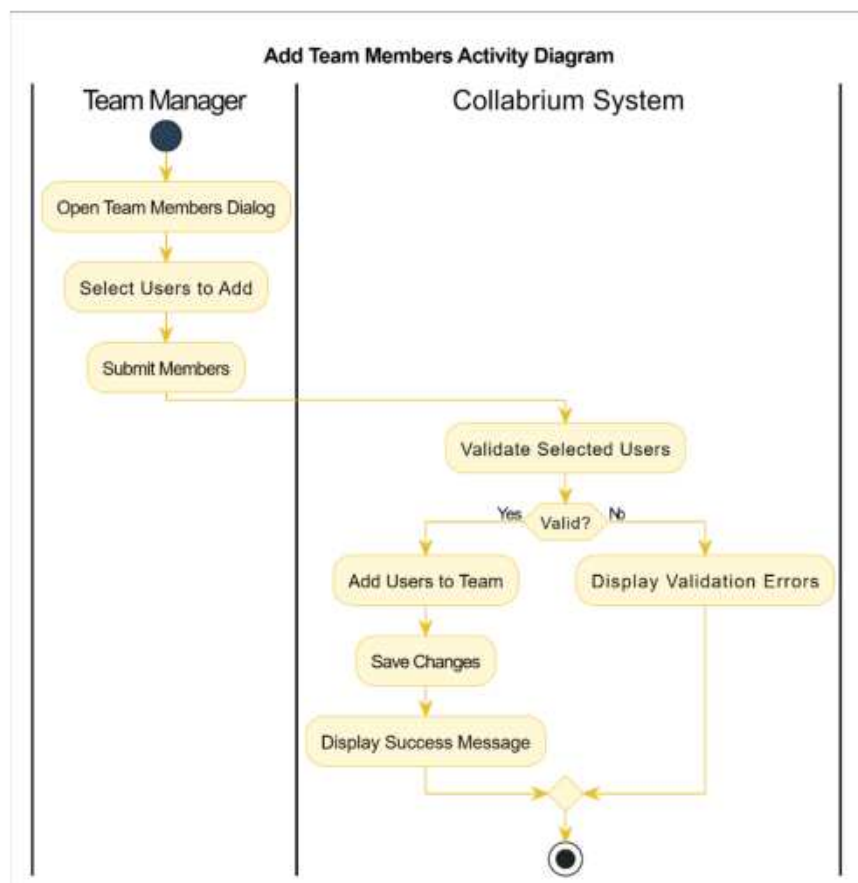


Figure 5.7.5 6 Add Team Members Activity Diagram

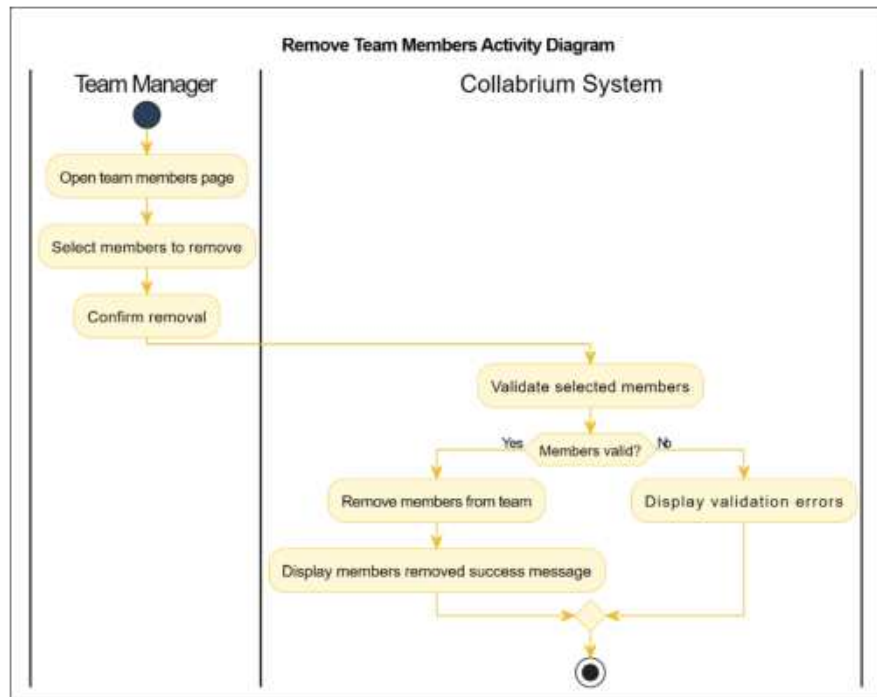


Figure 5.7.5 7 Remove Team Members Activity Diagram

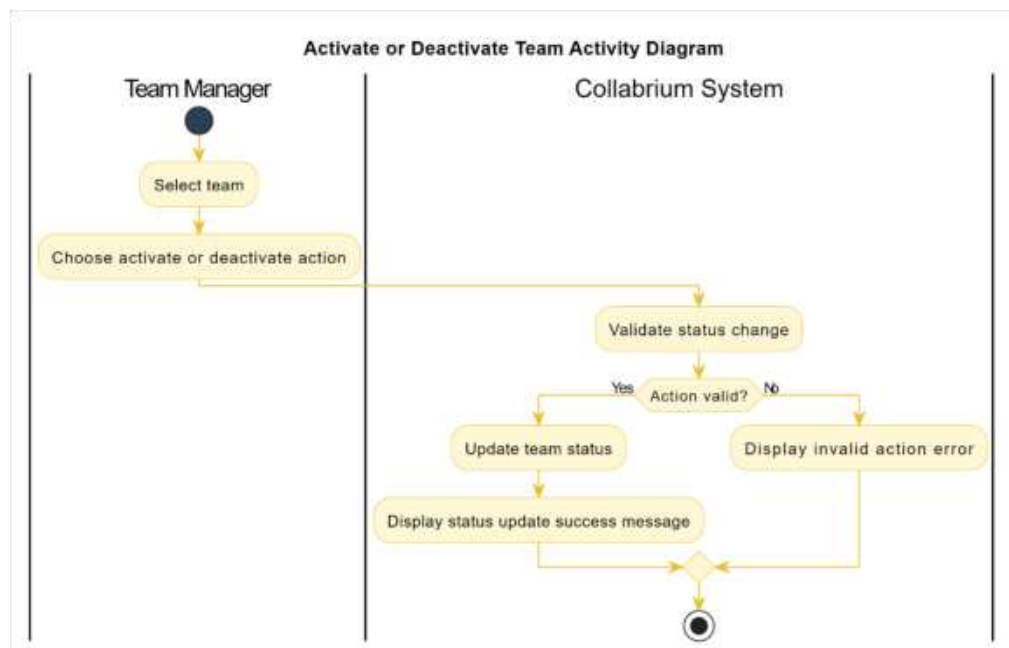


Figure 5.7.5 8 Activate or Deactivate Team Activity Diagram

## 5.7.6 Project Management Subsystem Activity Diagrams

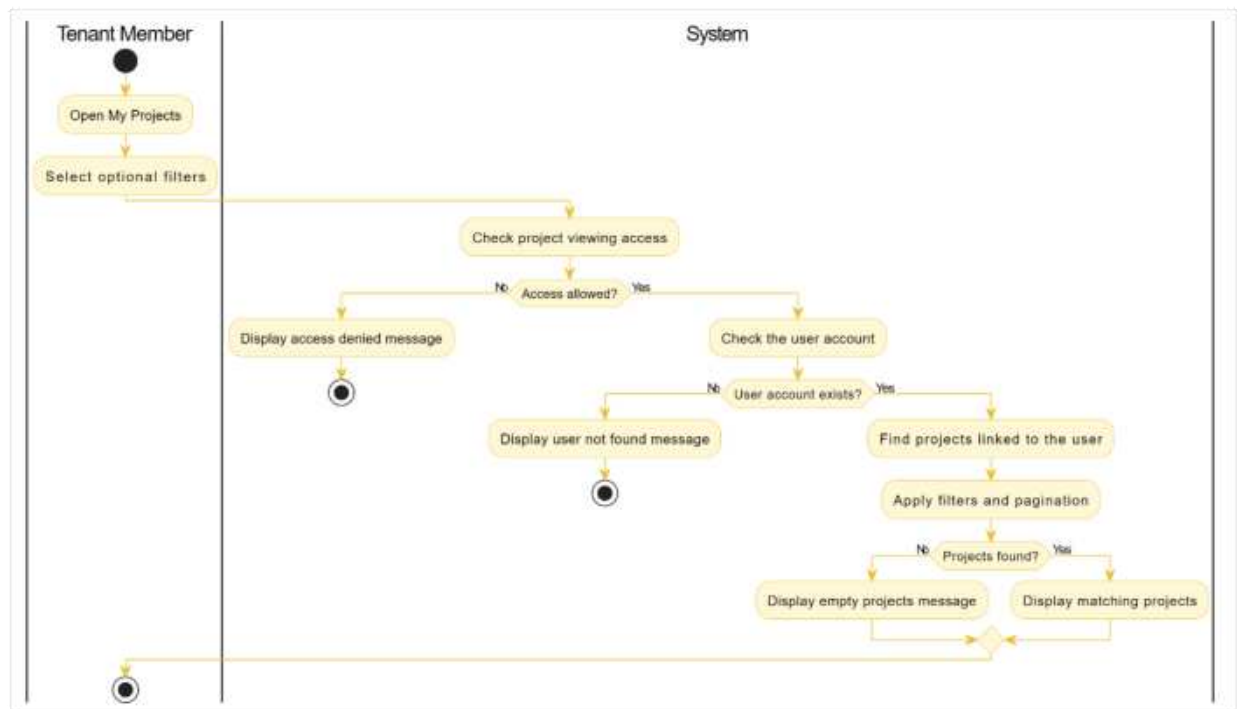


Figure 5.7.6 1 View My Projects Activity Diagram

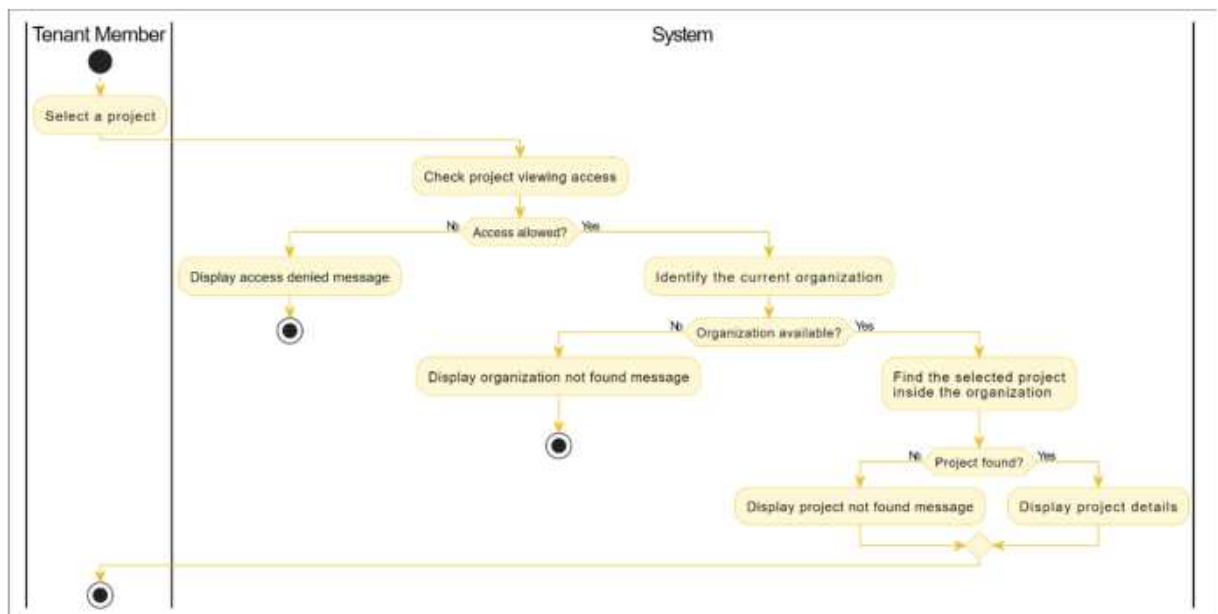


Figure 5.7.6 2 View Project Details Activity Diagram

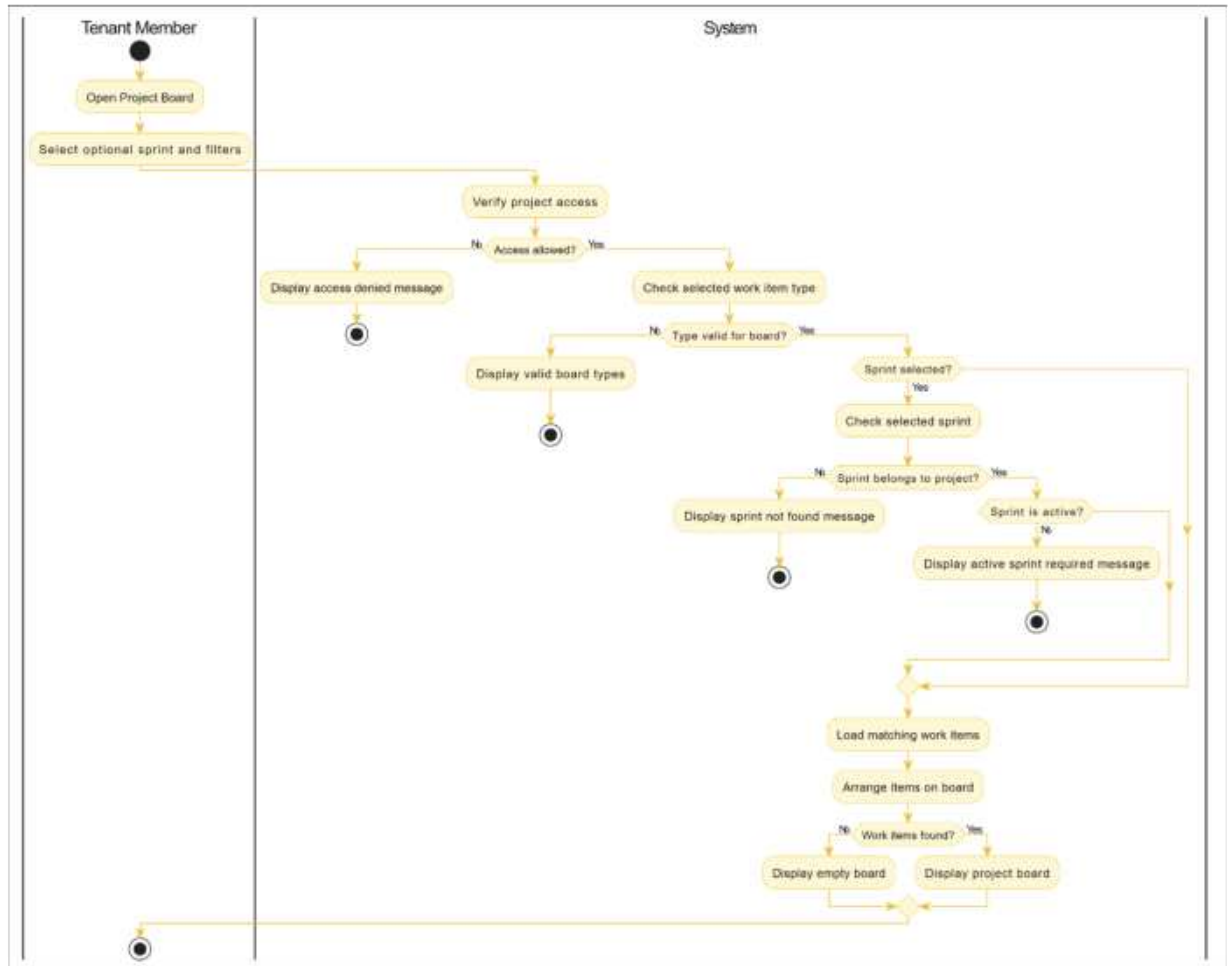


Figure 5.7.6 3 View Project Board Activity Diagram

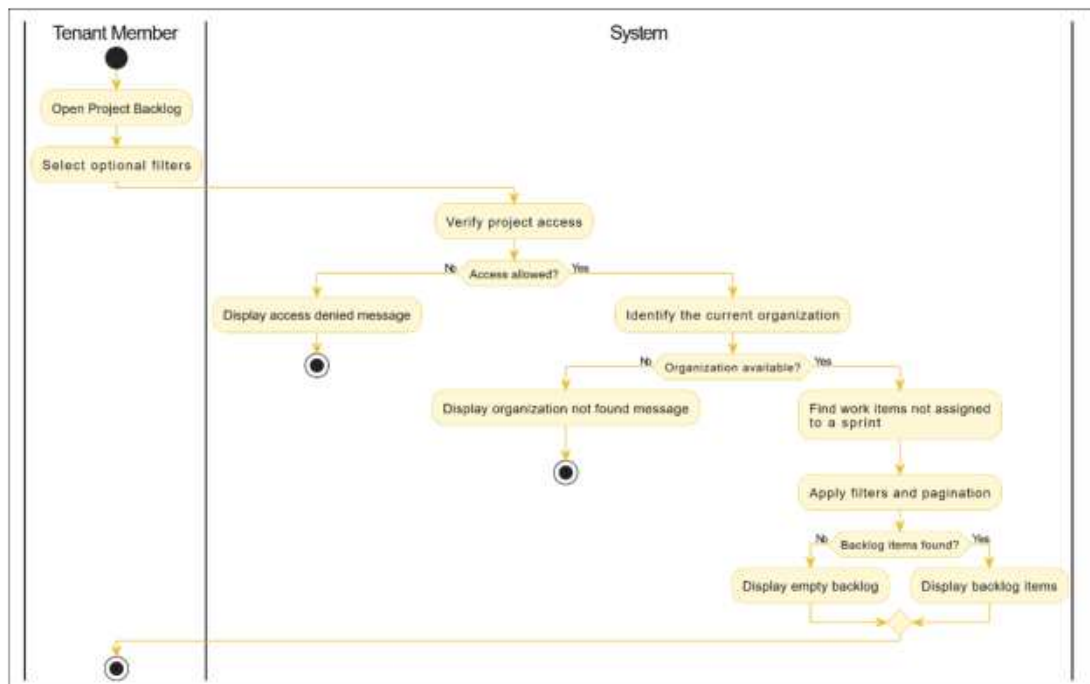


Figure 5.7.6 4 View Project Backlog Activity Diagram

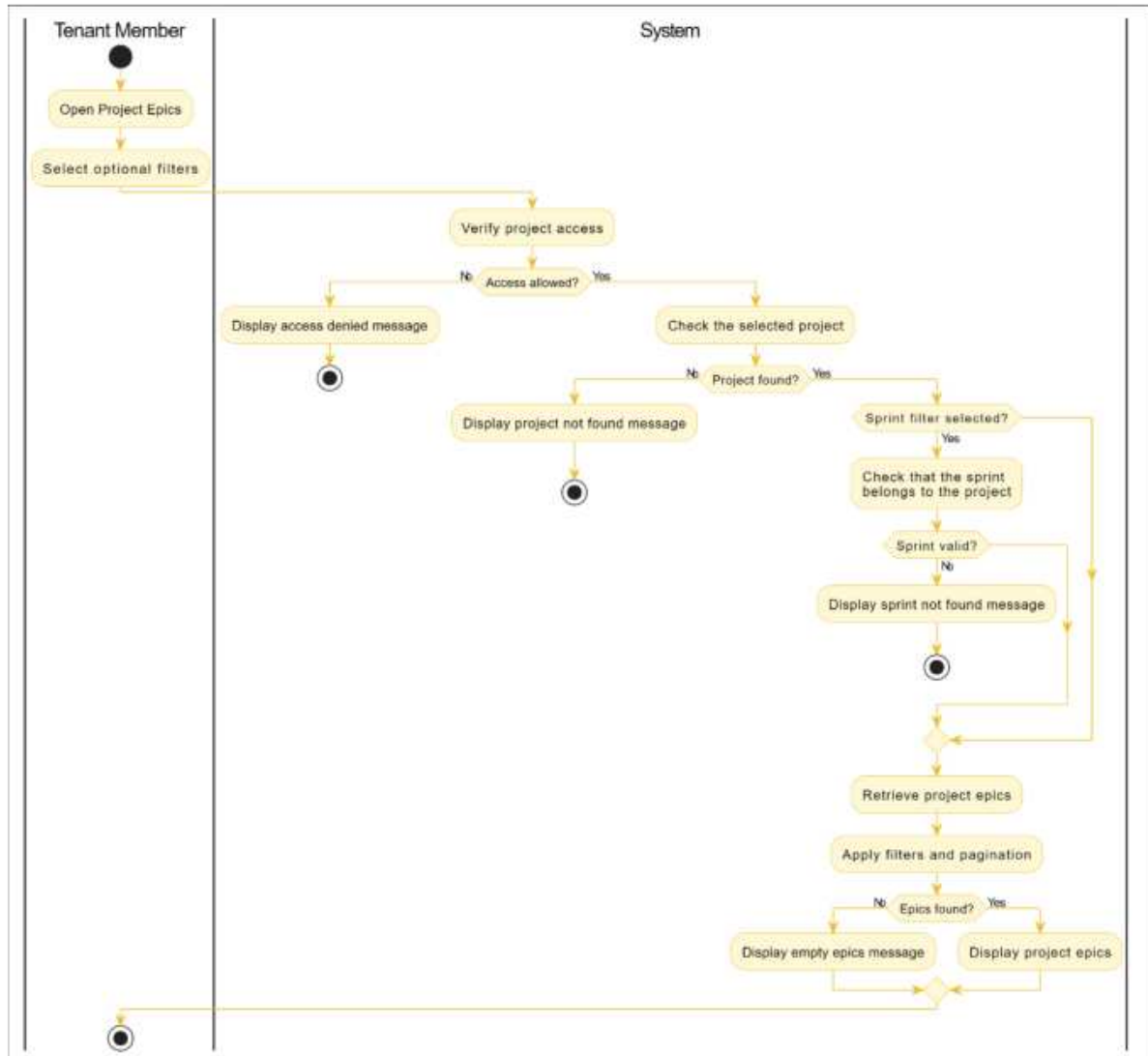


Figure 5.7.6 5 View Project Epics Activity Diagram

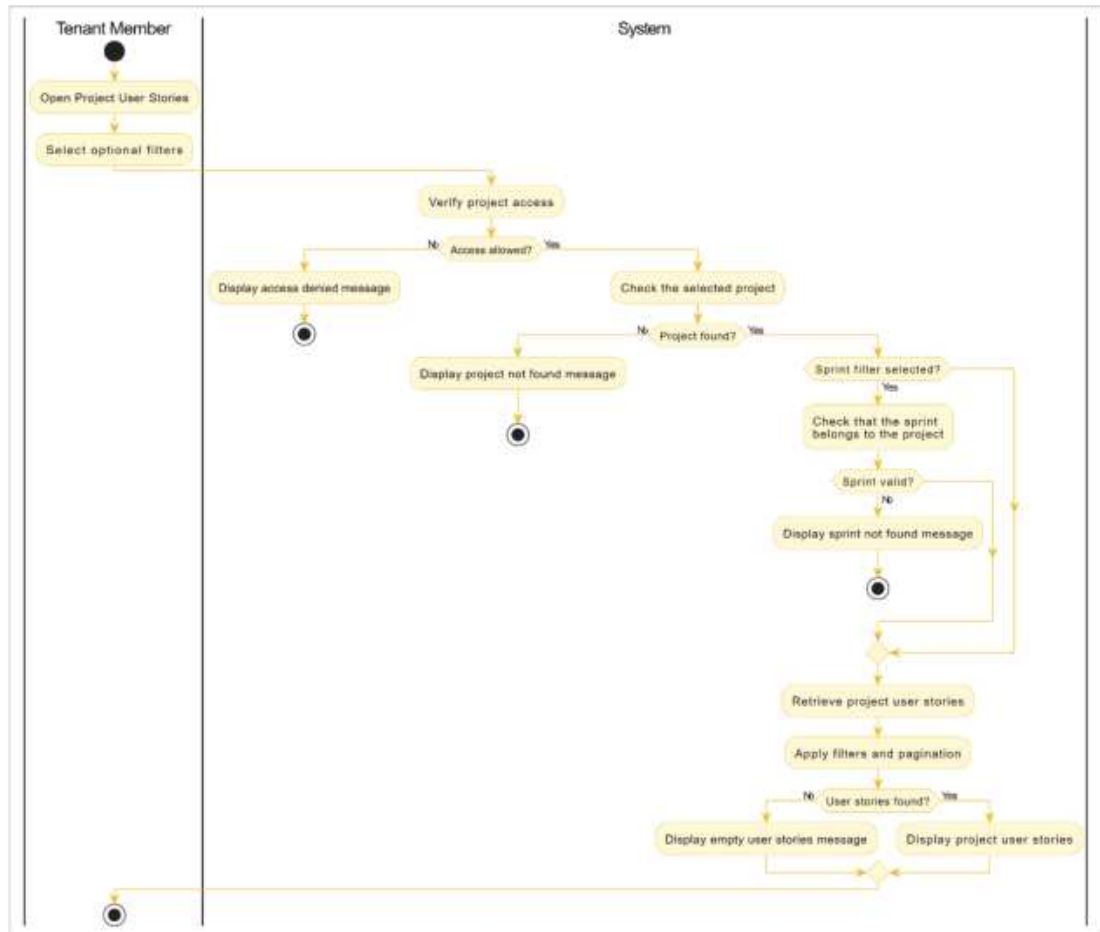


Figure 5.7.6 6 View Project User Stories Activity Diagram

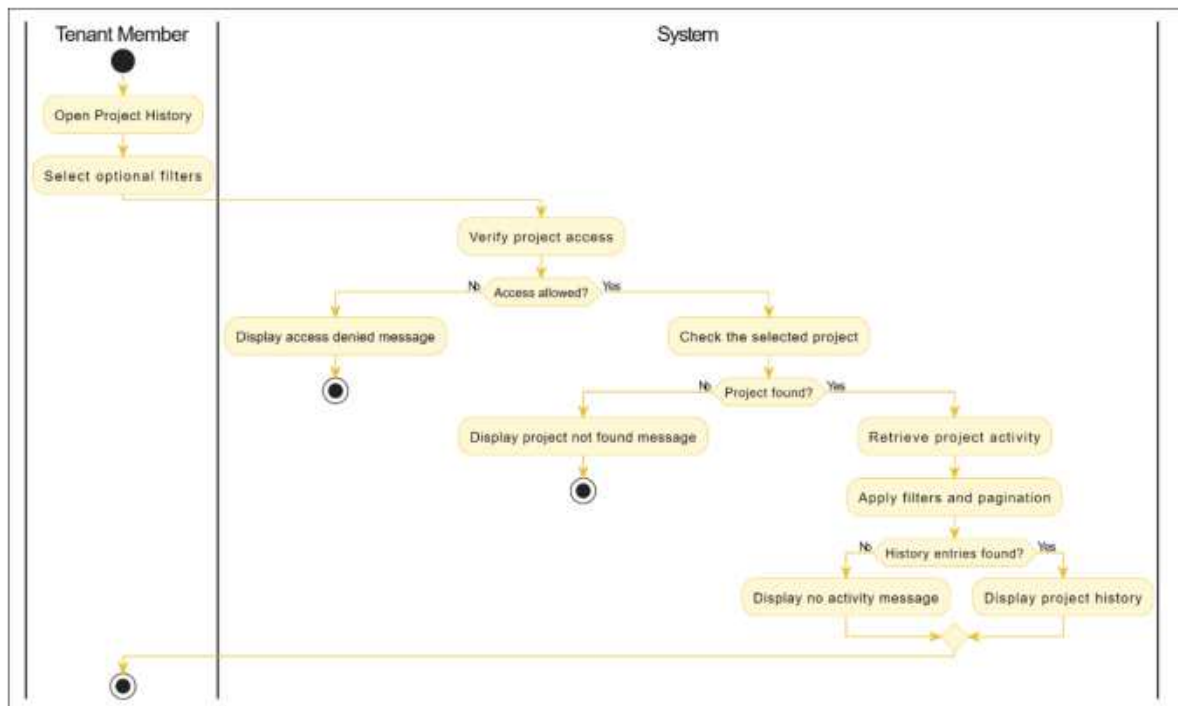


Figure 5.7.6 7 View Project History Activity Diagram

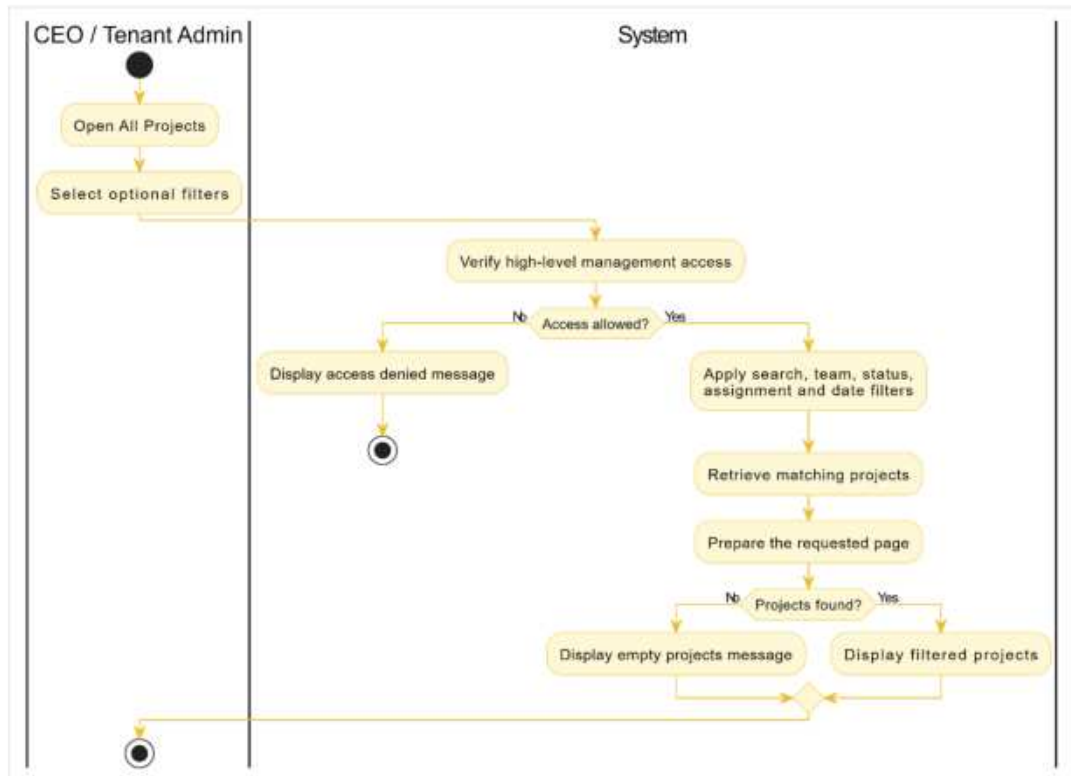


Figure 5.7.6 8 Browse All Projects Activity Diagram

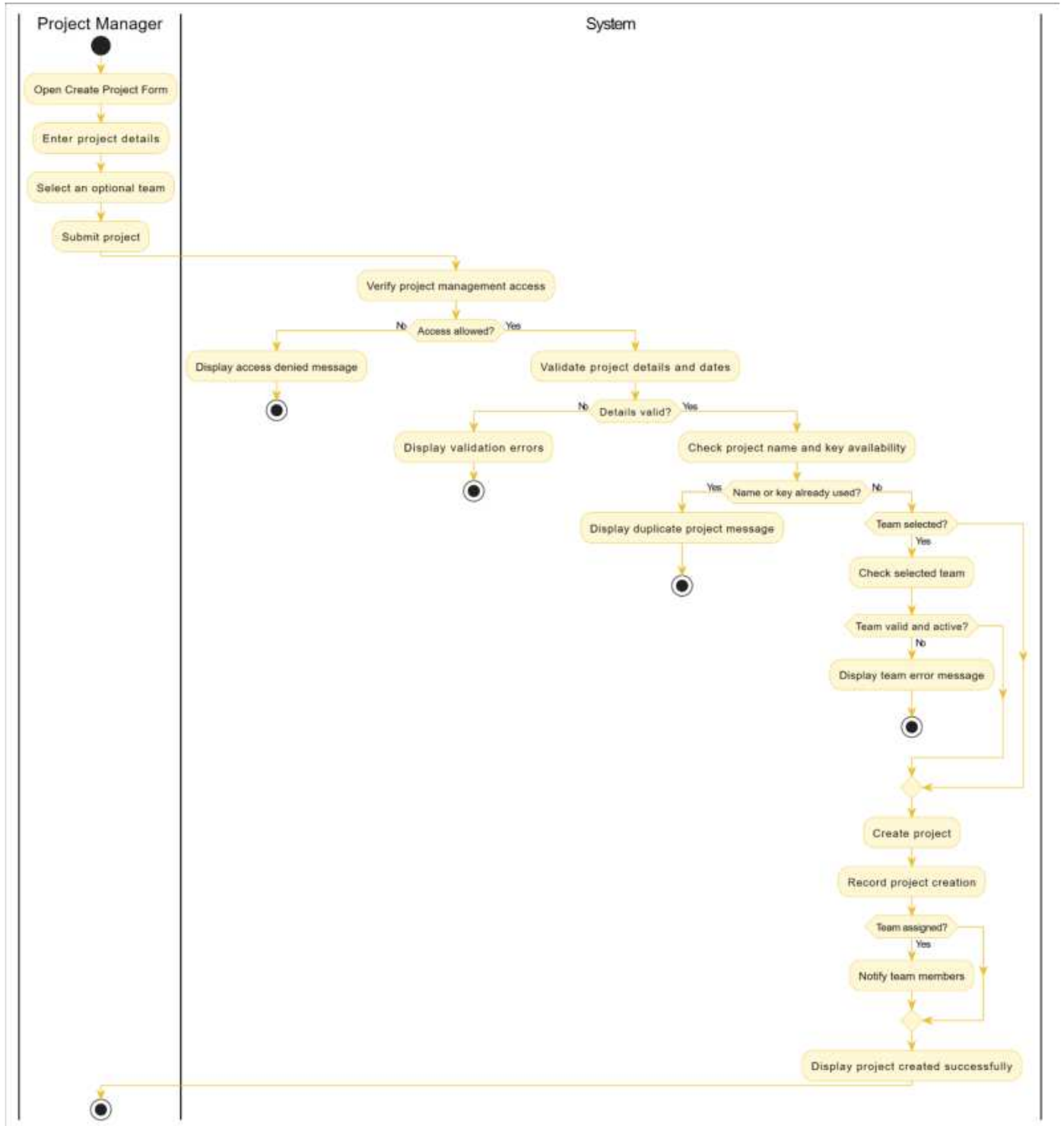


Figure 5.7.6 9 Create Project Activity Diagram

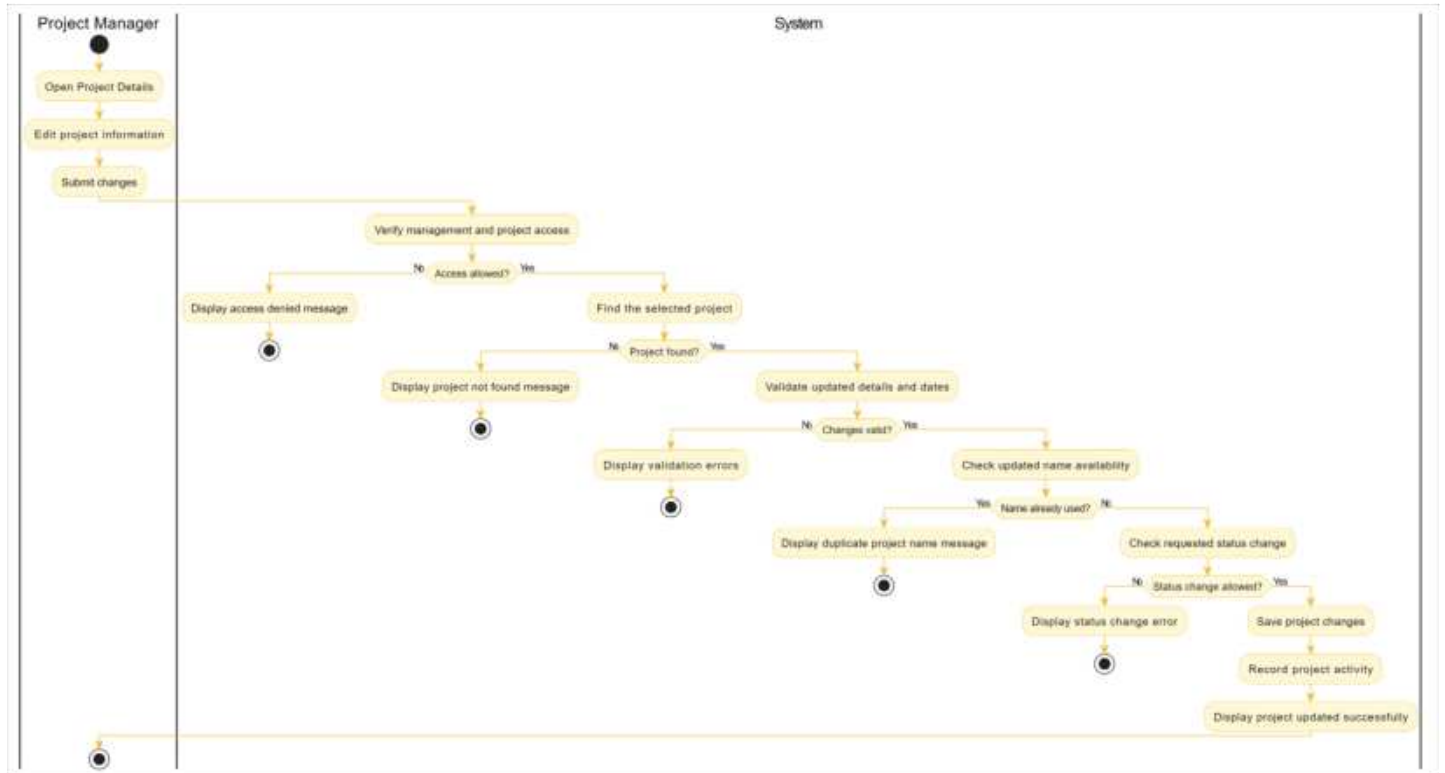


Figure 5.7.6 10 Edit Project Activity Diagram

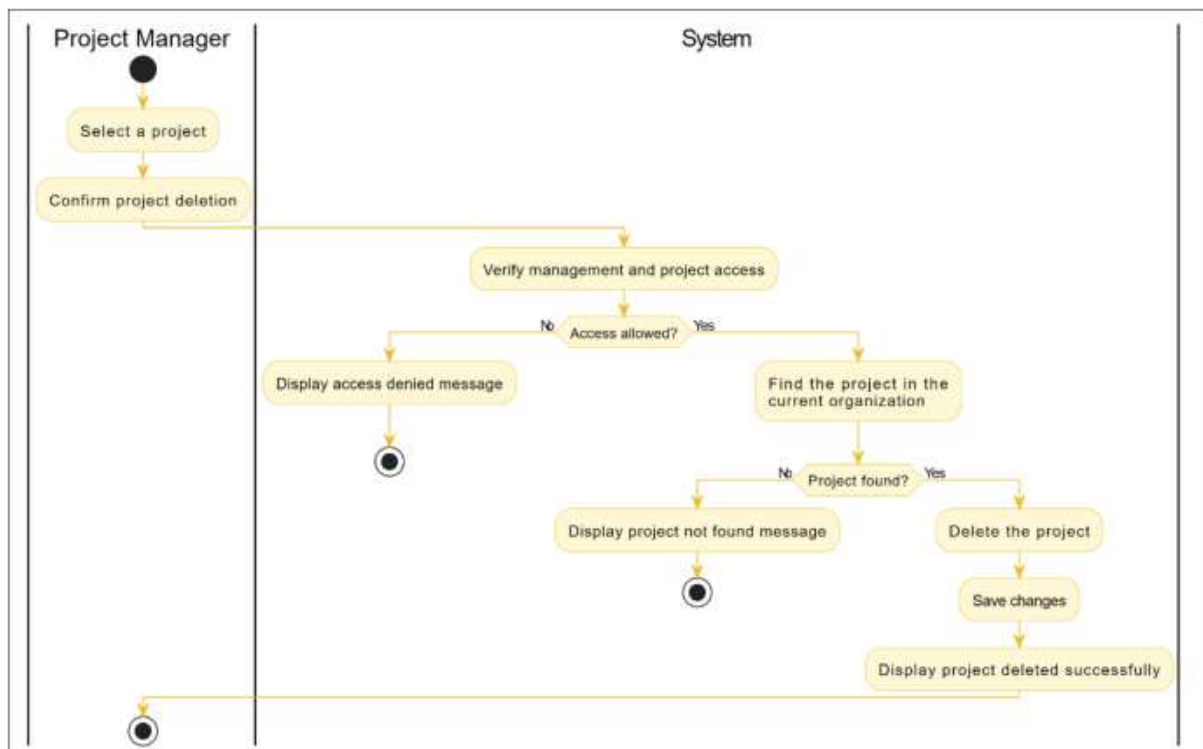


Figure 5.7.6 11 Delete Project Activity Diagram

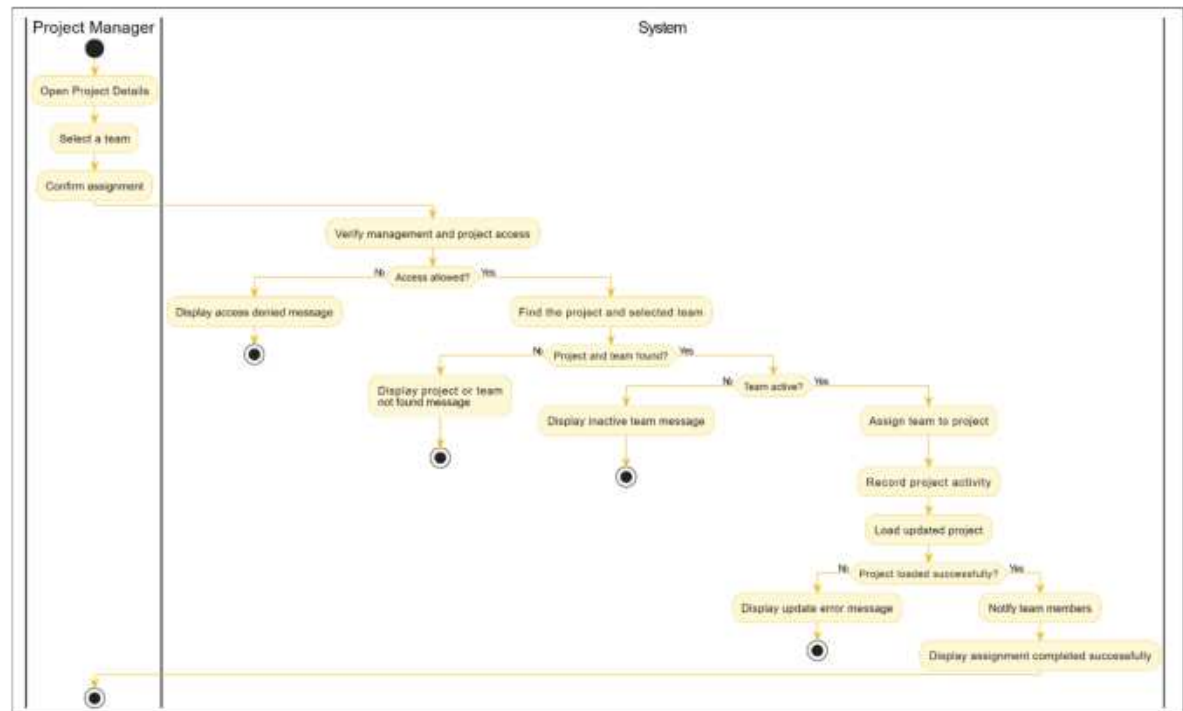


Figure 5.7.6 12 Assign Team to Project Activity Diagram

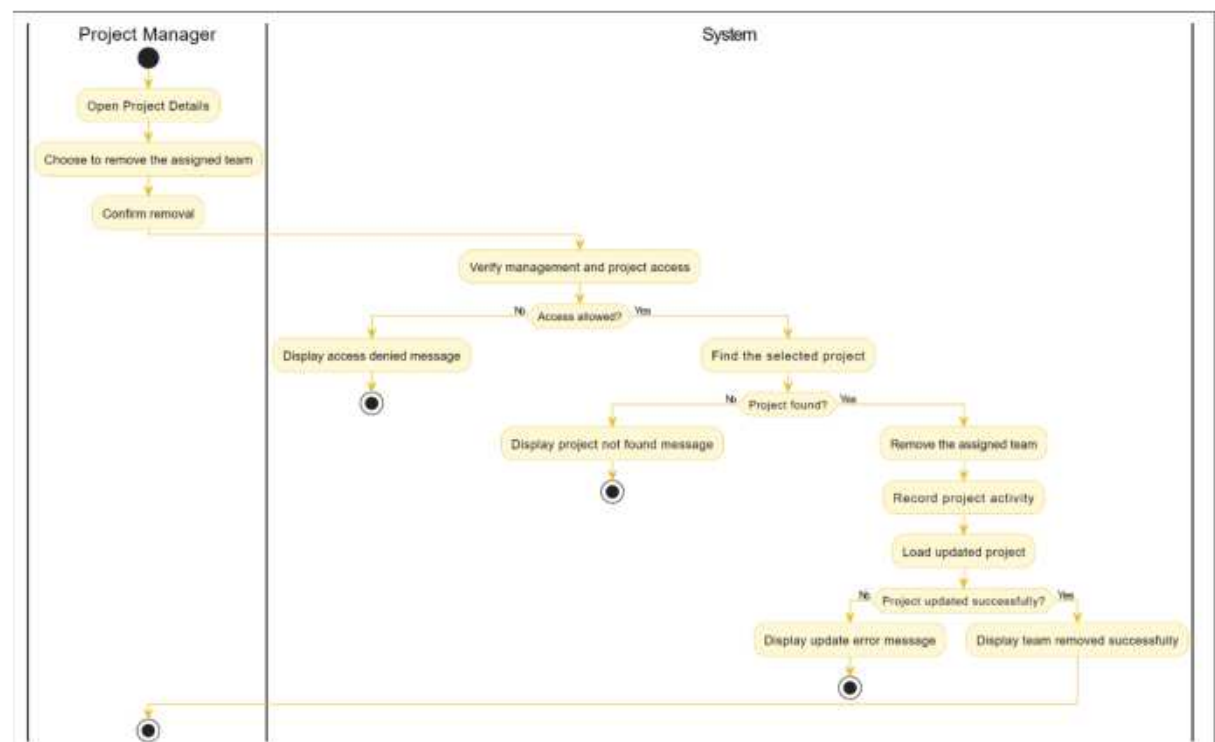


Figure 5.7.6 13 Unassign Team from Project Activity Diagram

### 5.7.7 Work Item Core Management Subsystem Activity Diagrams

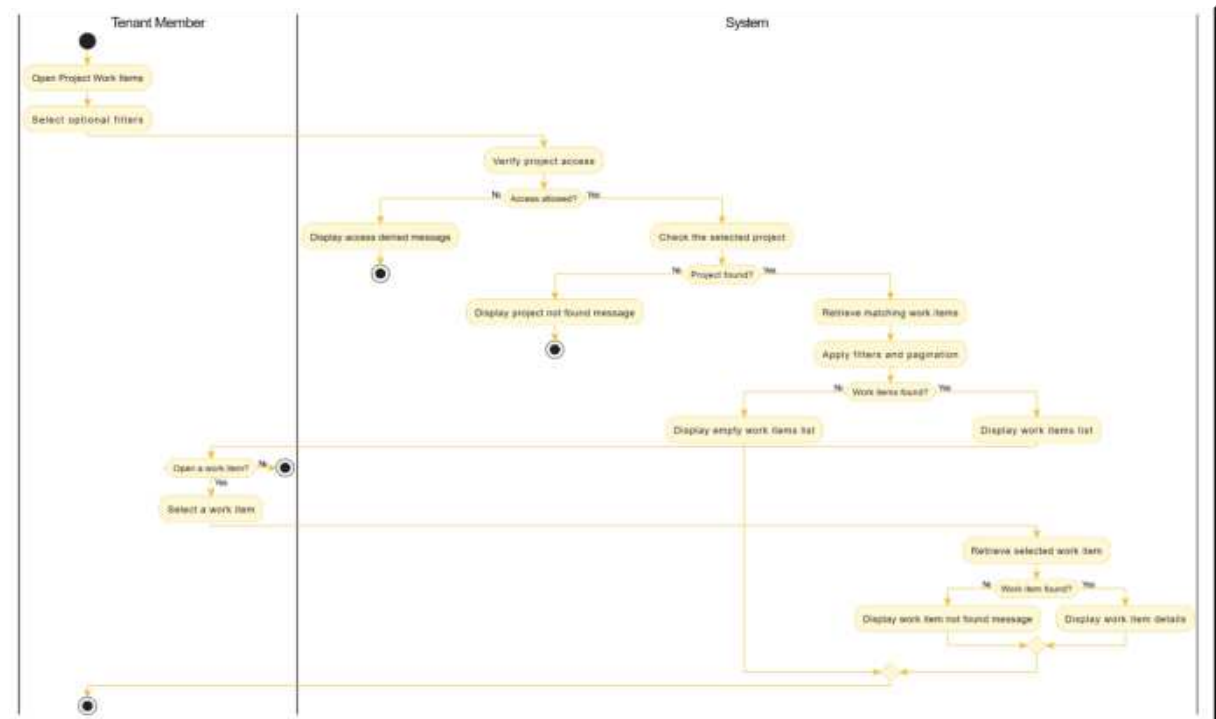


Figure 5.7.7 1 Browse and Filter Work Items Activity Diagram

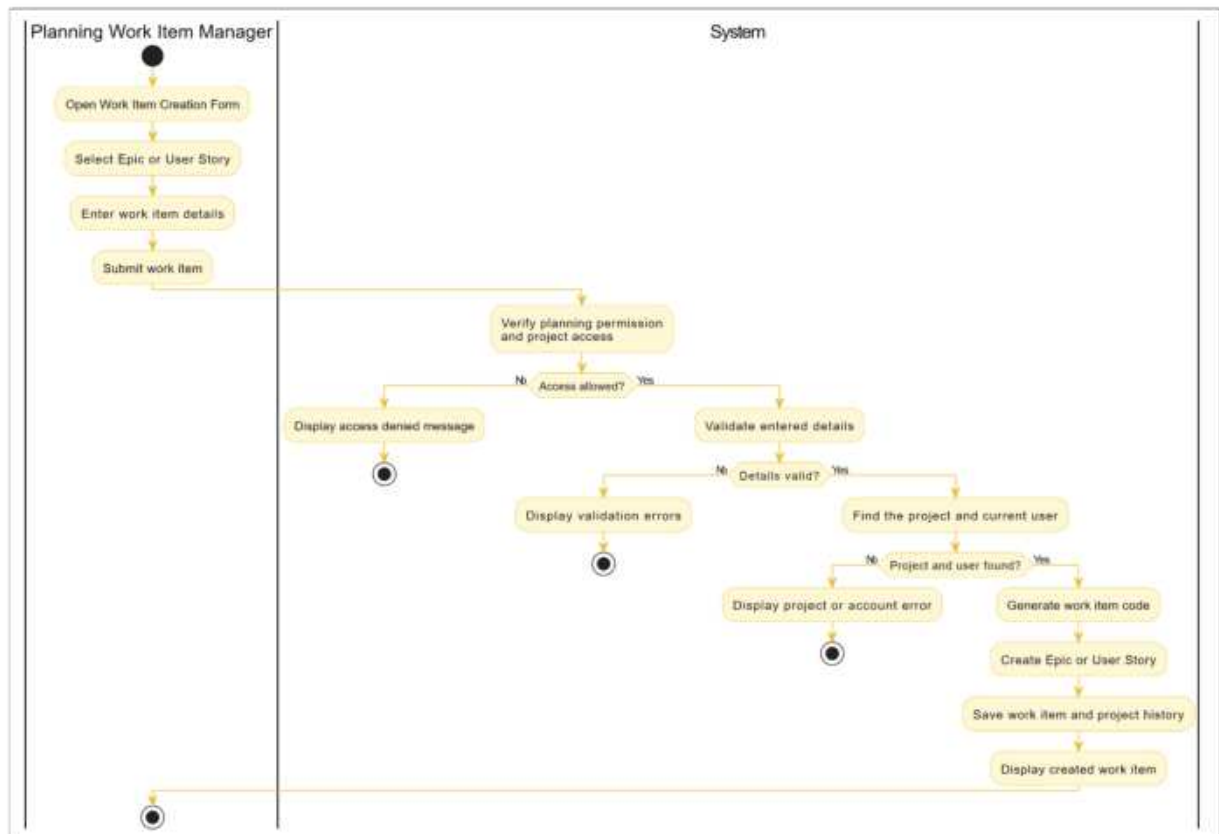


Figure 5.7.7 2 Create Planning Work Item Activity Diagram

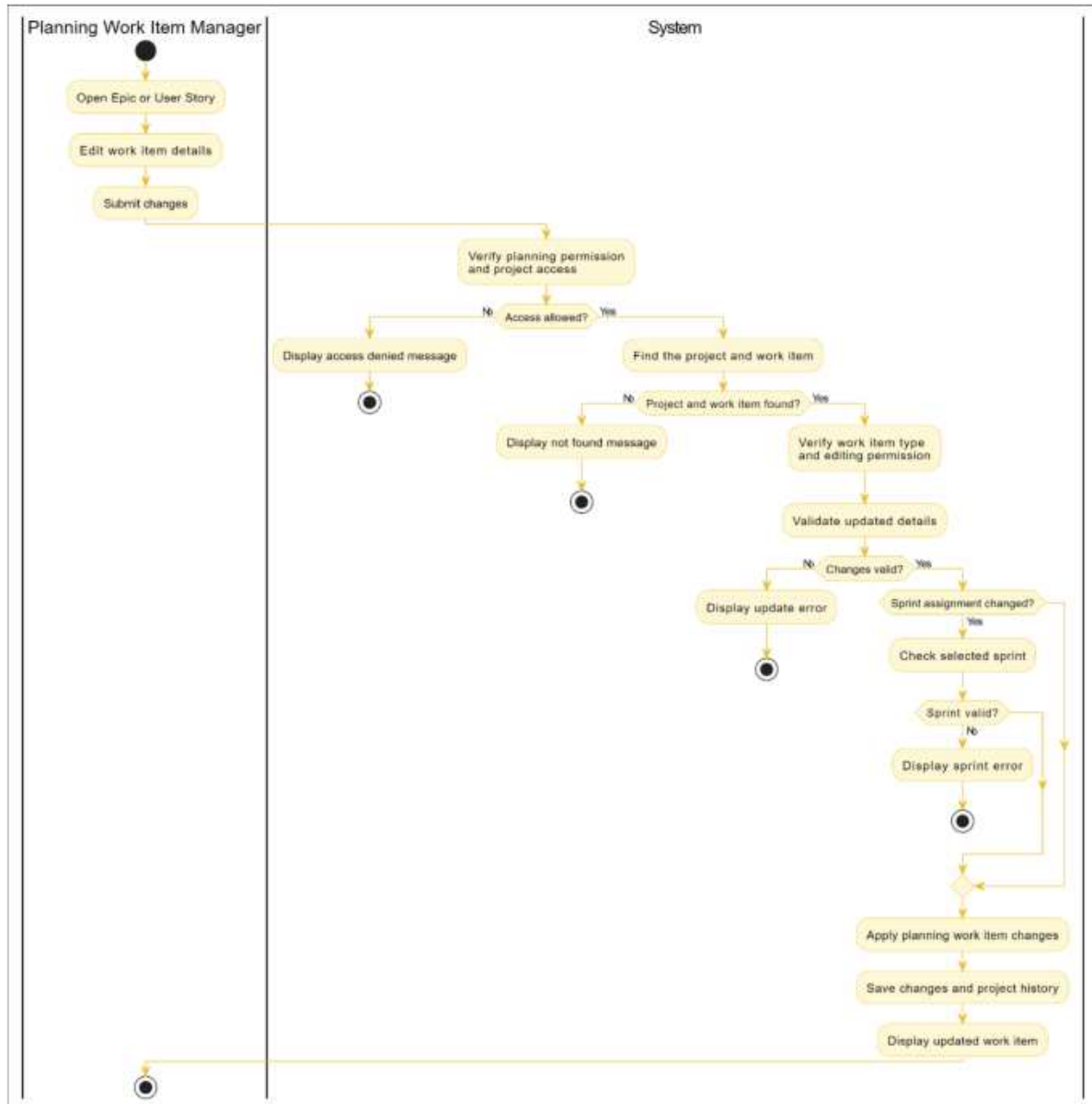


Figure 5.7.7 3 Update Planning Work Item Activity Diagram

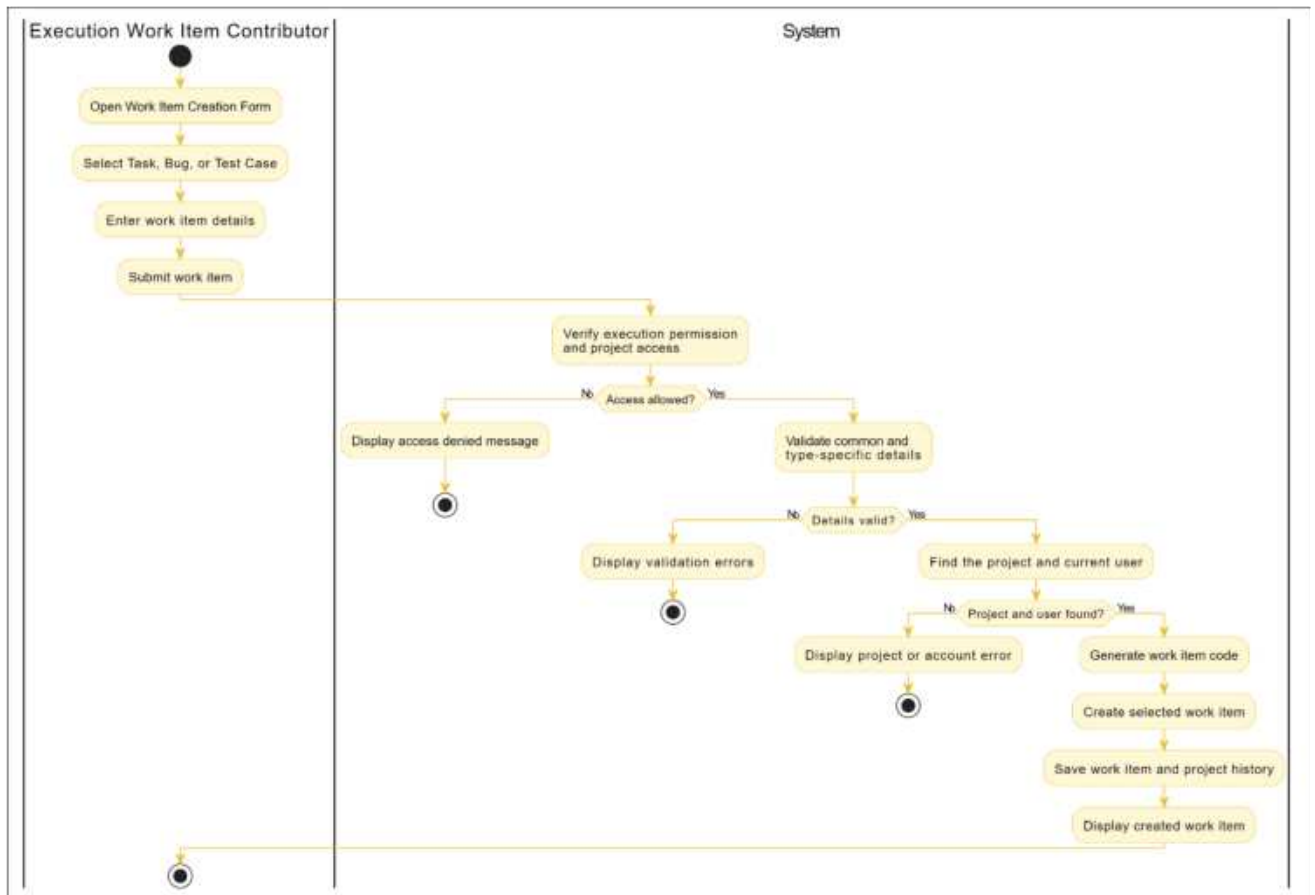


Figure 5.7.7 4 Create Execution Work Item Activity Diagram

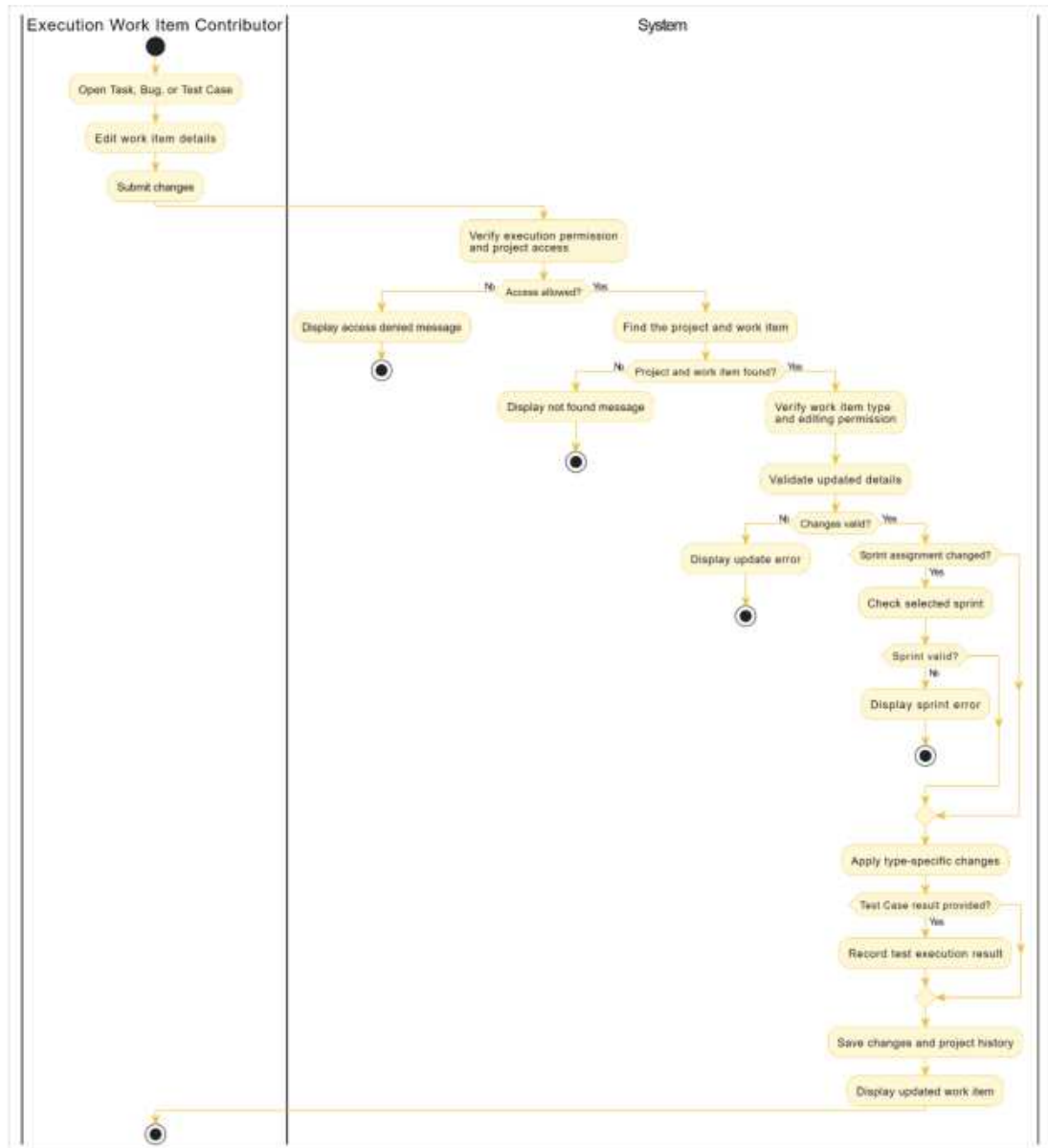


Figure 5.7.7 5 Update Execution Work Item Activity Diagram

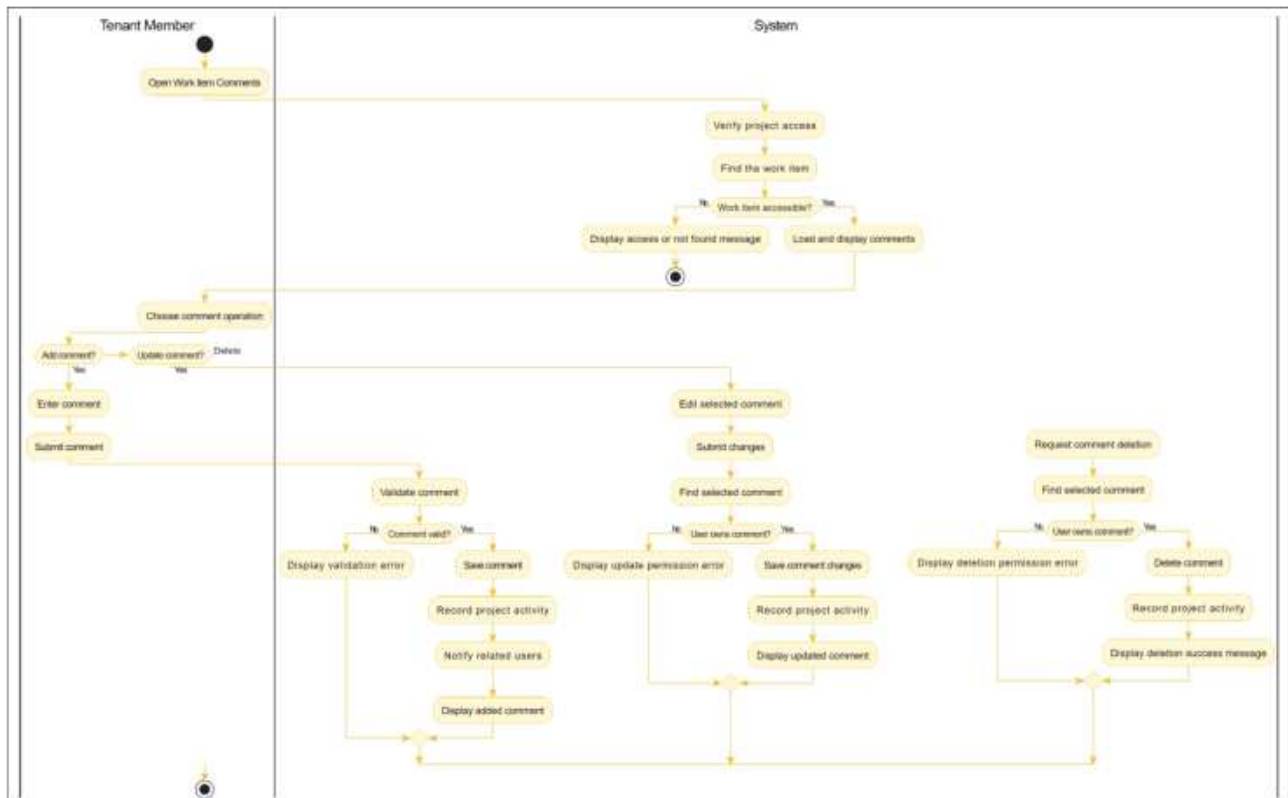


Figure 5.7.7 6 Manage Work Item Comments Activity Diagram

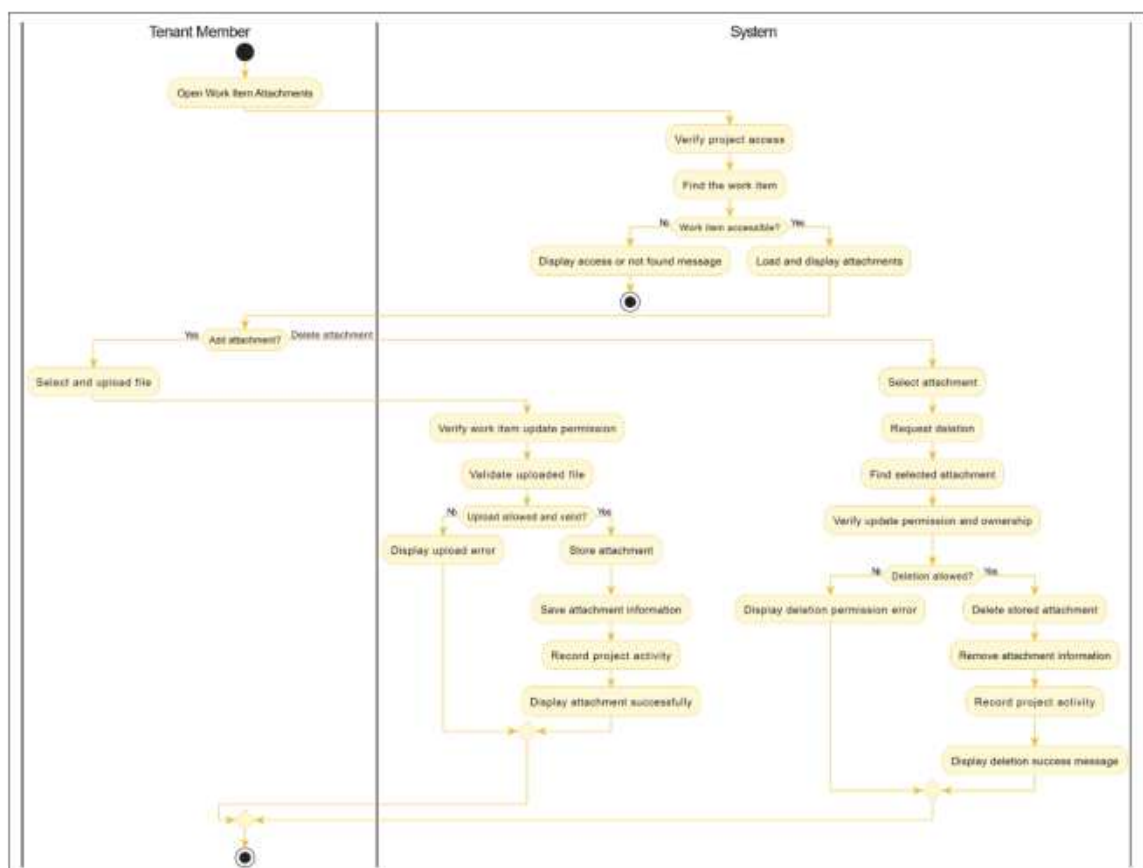


Figure 5.7.7 7 Manage Work Item Attachments Activity Diagram

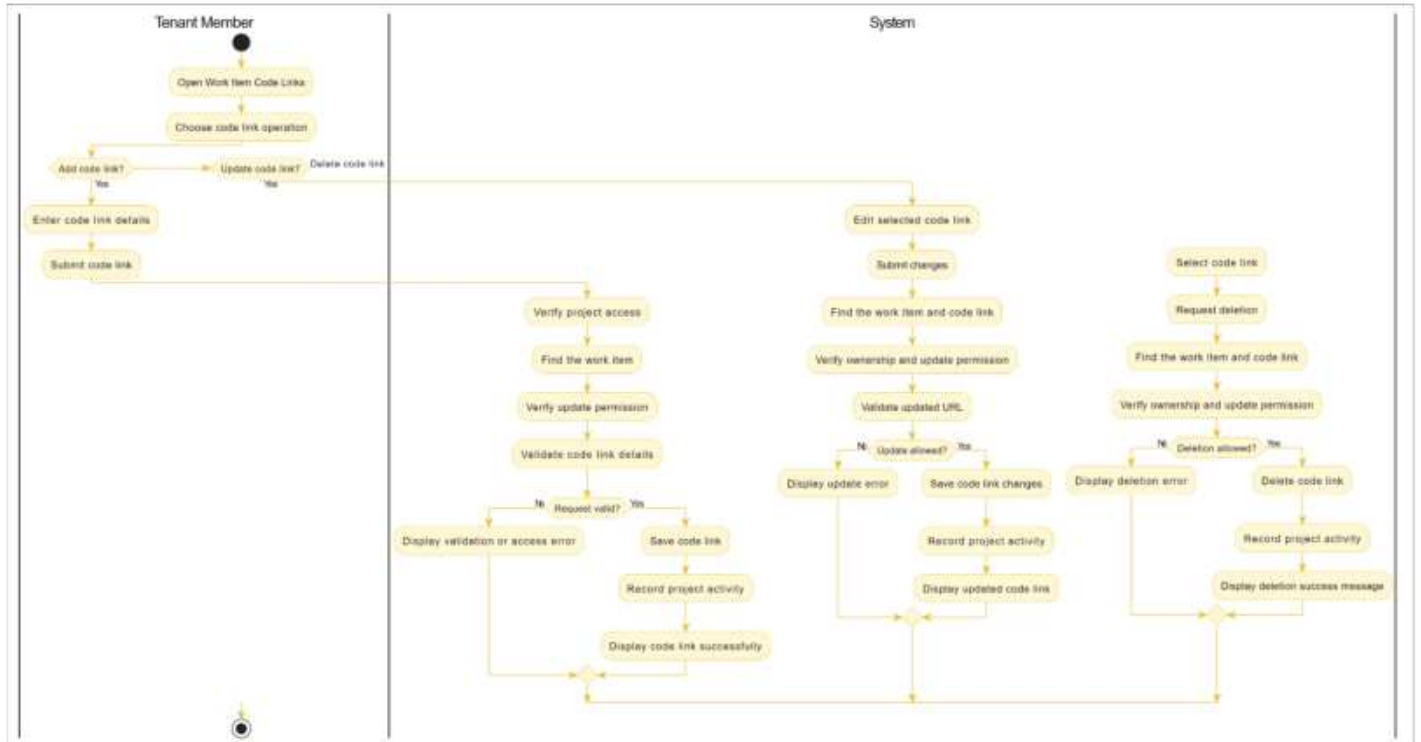


Figure 5.7.7 8 Manage Work Item Code Links Activity Diagram

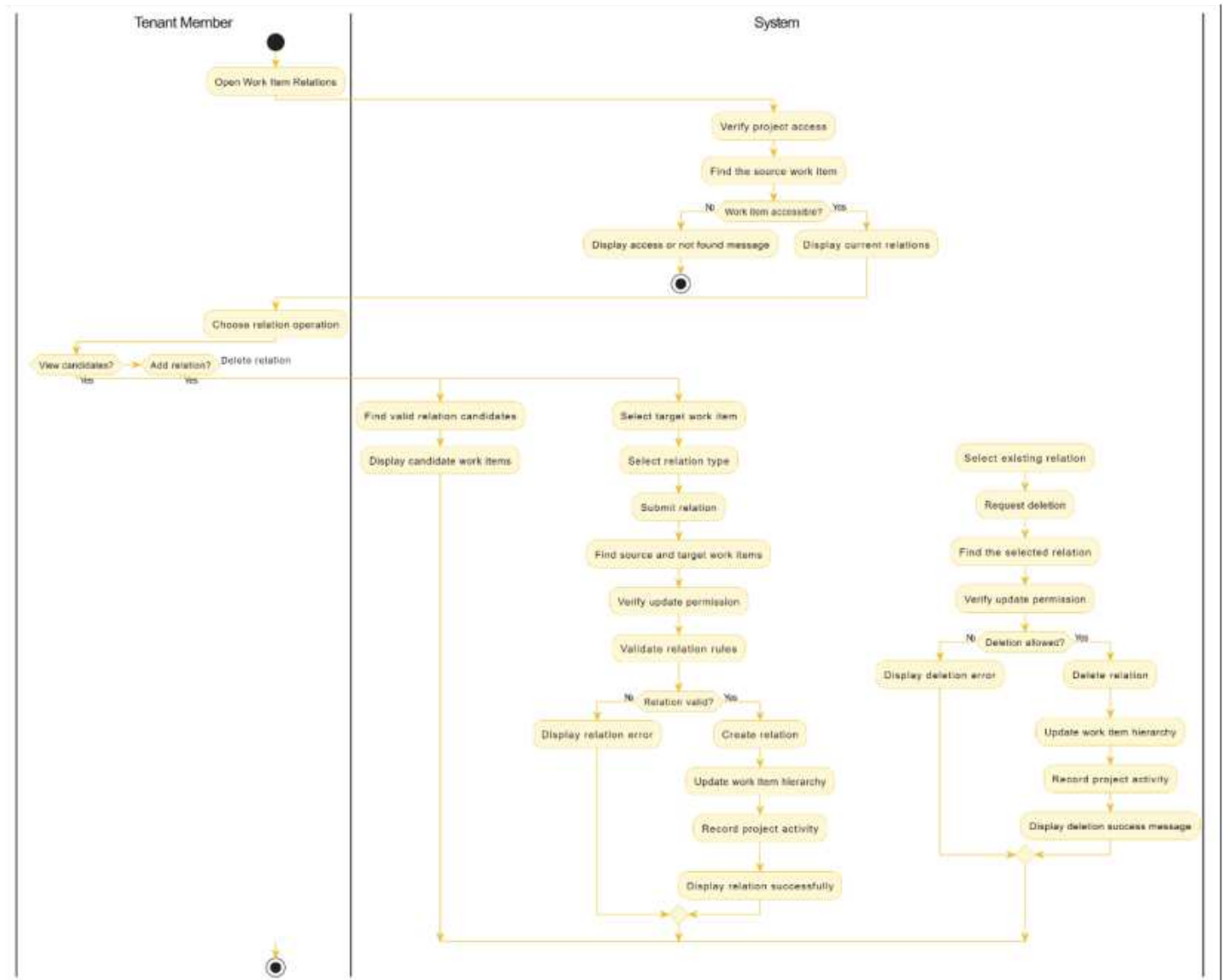


Figure 5.7.7 9 Manage Work Item Relations Activity Diagram

## 5.7.8 Sprint Management Subsystem Activity Diagrams

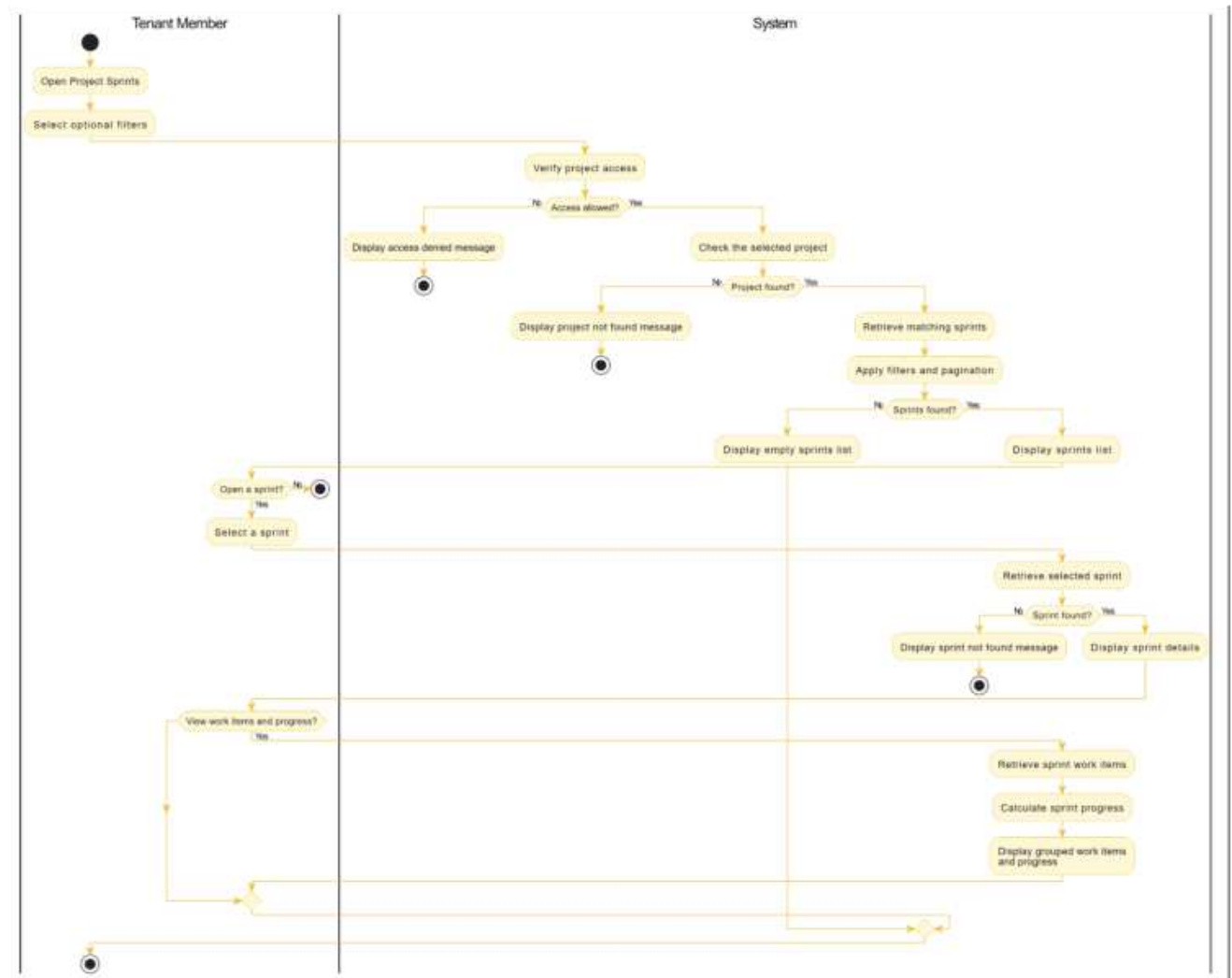


Figure 5.7.8 1 Browse and Filter Sprints

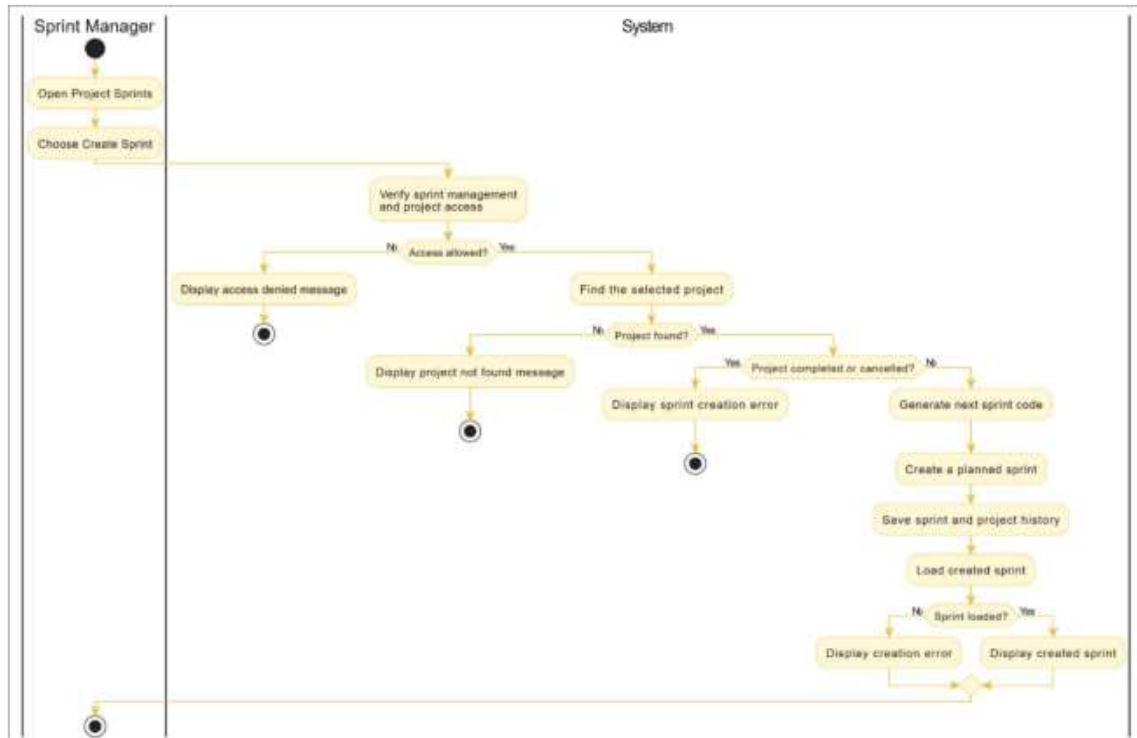


Figure 5.7.8 2 Create Sprint Activity Diagram

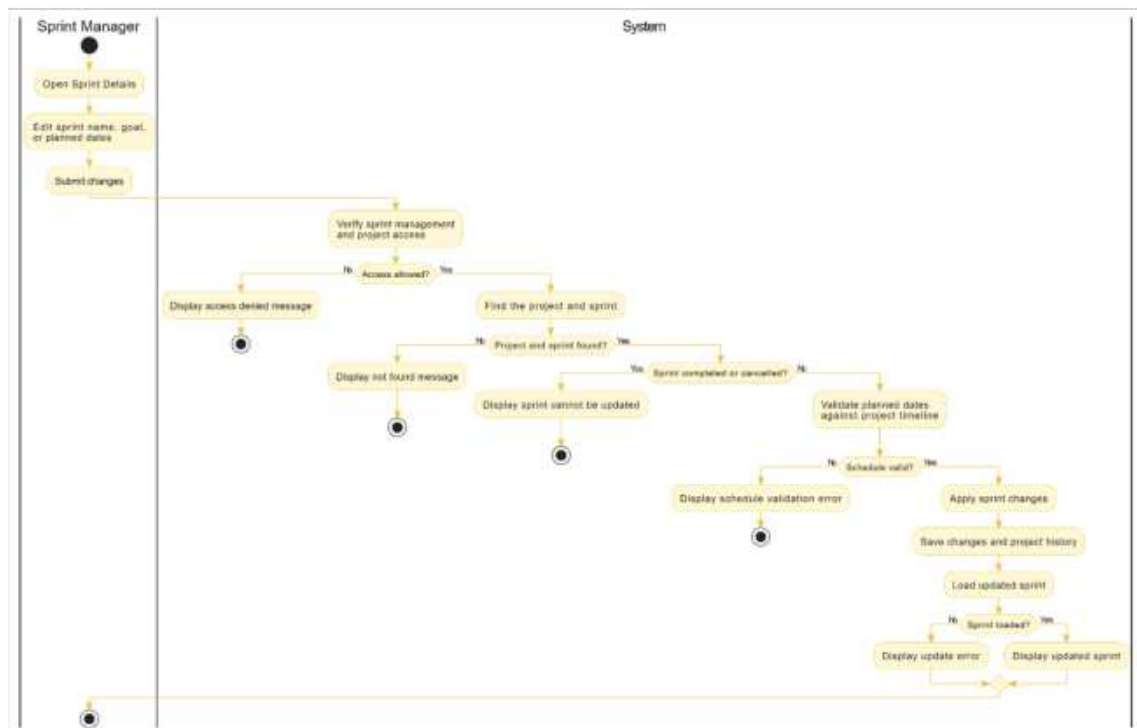


Figure 5.7.8 3 Update Sprint Activity Diagram

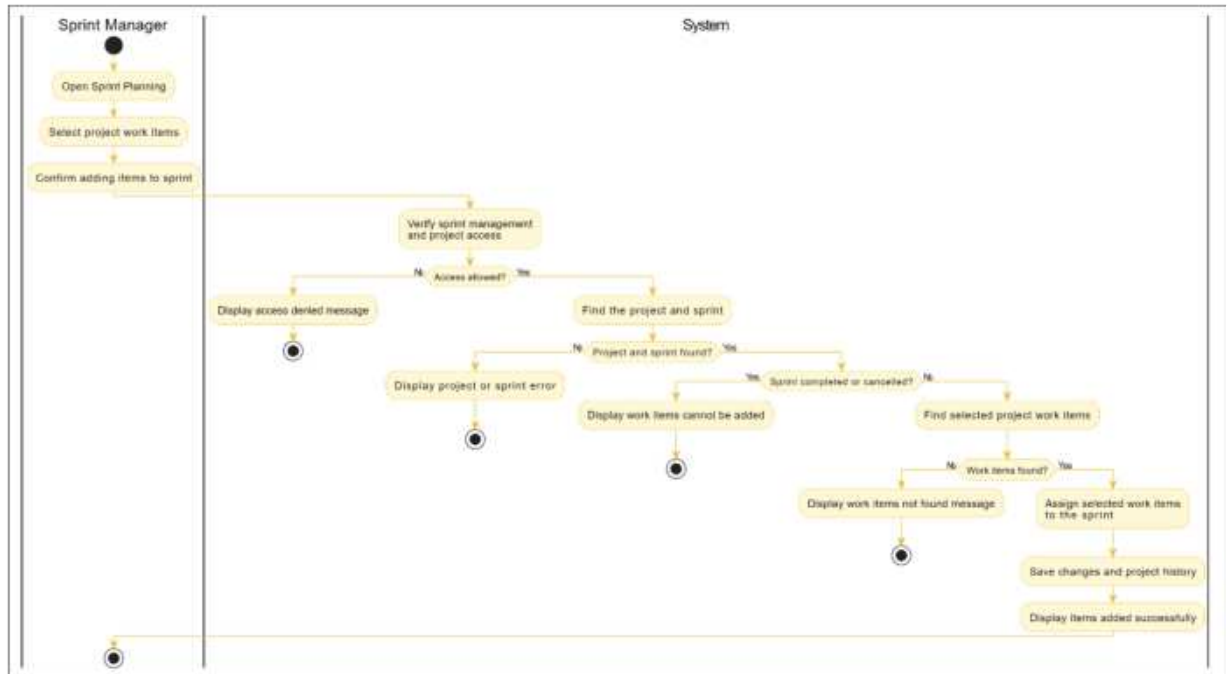


Figure 5.7.8 4 Add Work Items to Sprint Activity Diagram

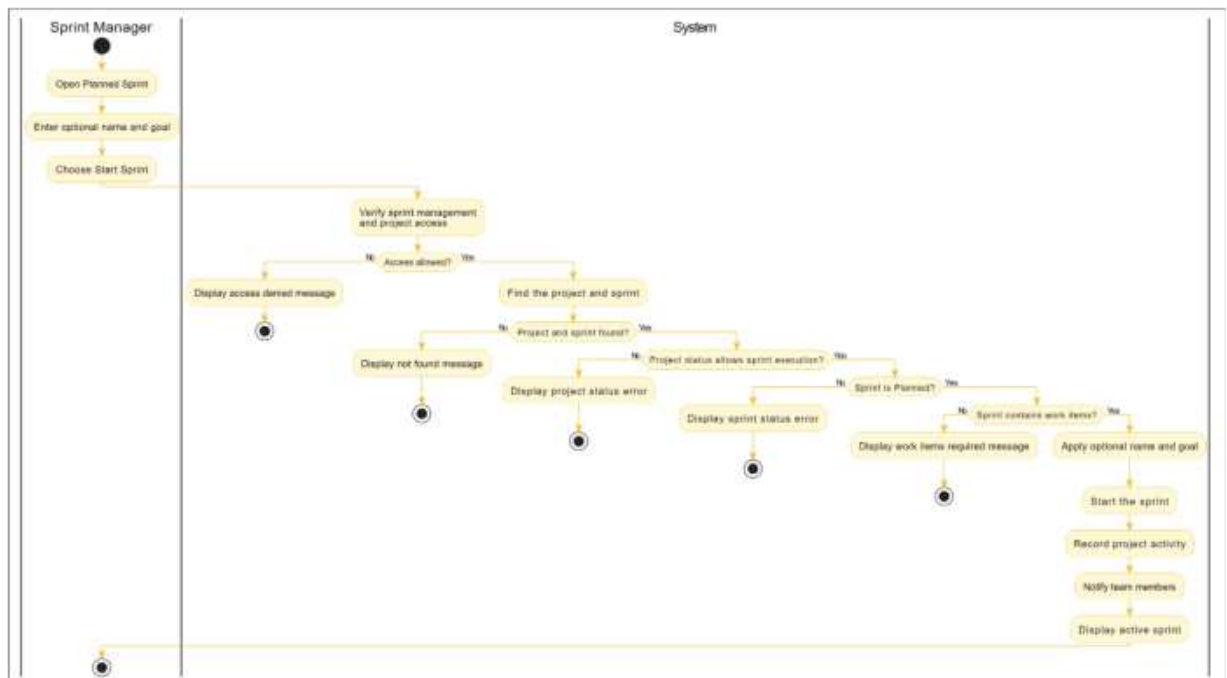


Figure 5.7.8 5 Start Sprint Activity Diagram

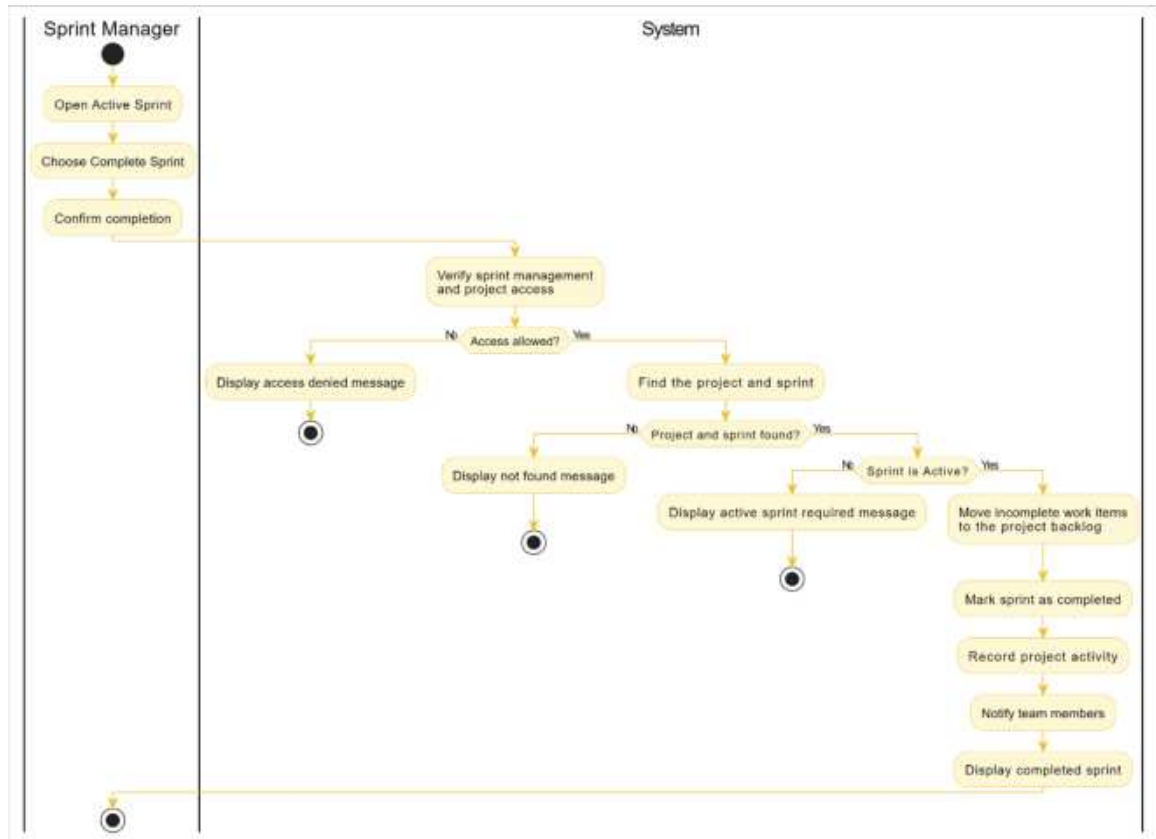


Figure 5.7.8 6 Complete Sprint Activity Diagram

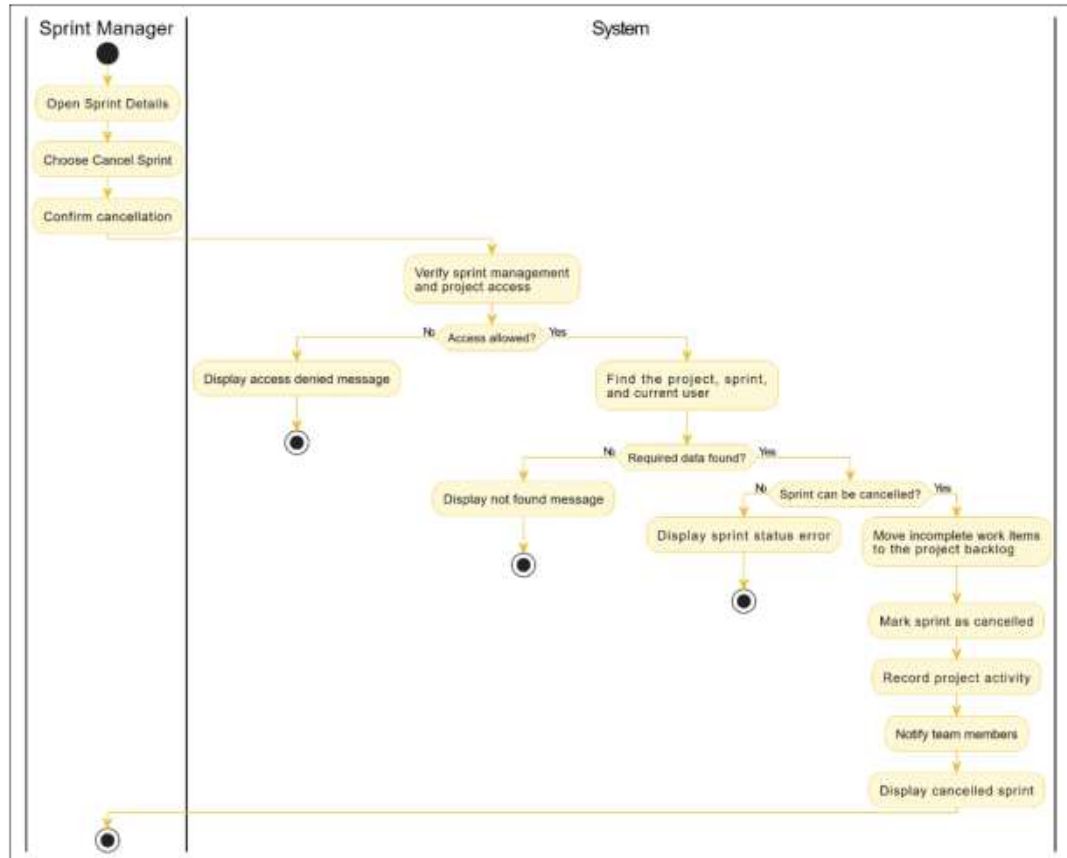


Figure 5.7.8 7 Cancel Sprint Activity Diagram

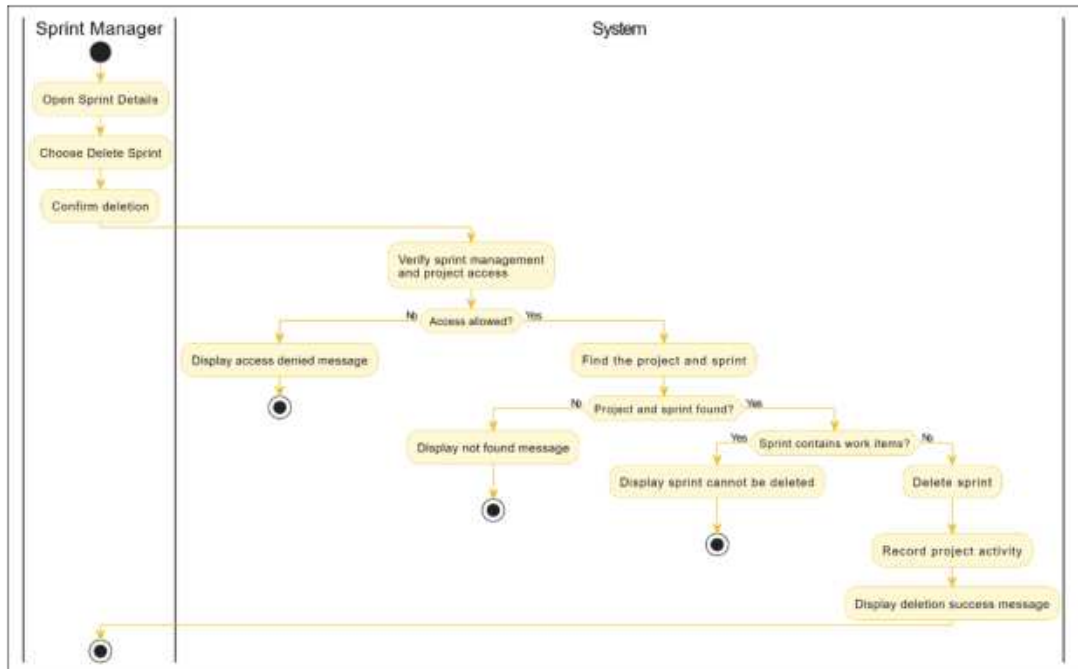


Figure 5.7.8 8 Delete Sprint Activity Diagram

### 5.7.9 Reports and Analytics Subsystem Activity Diagrams

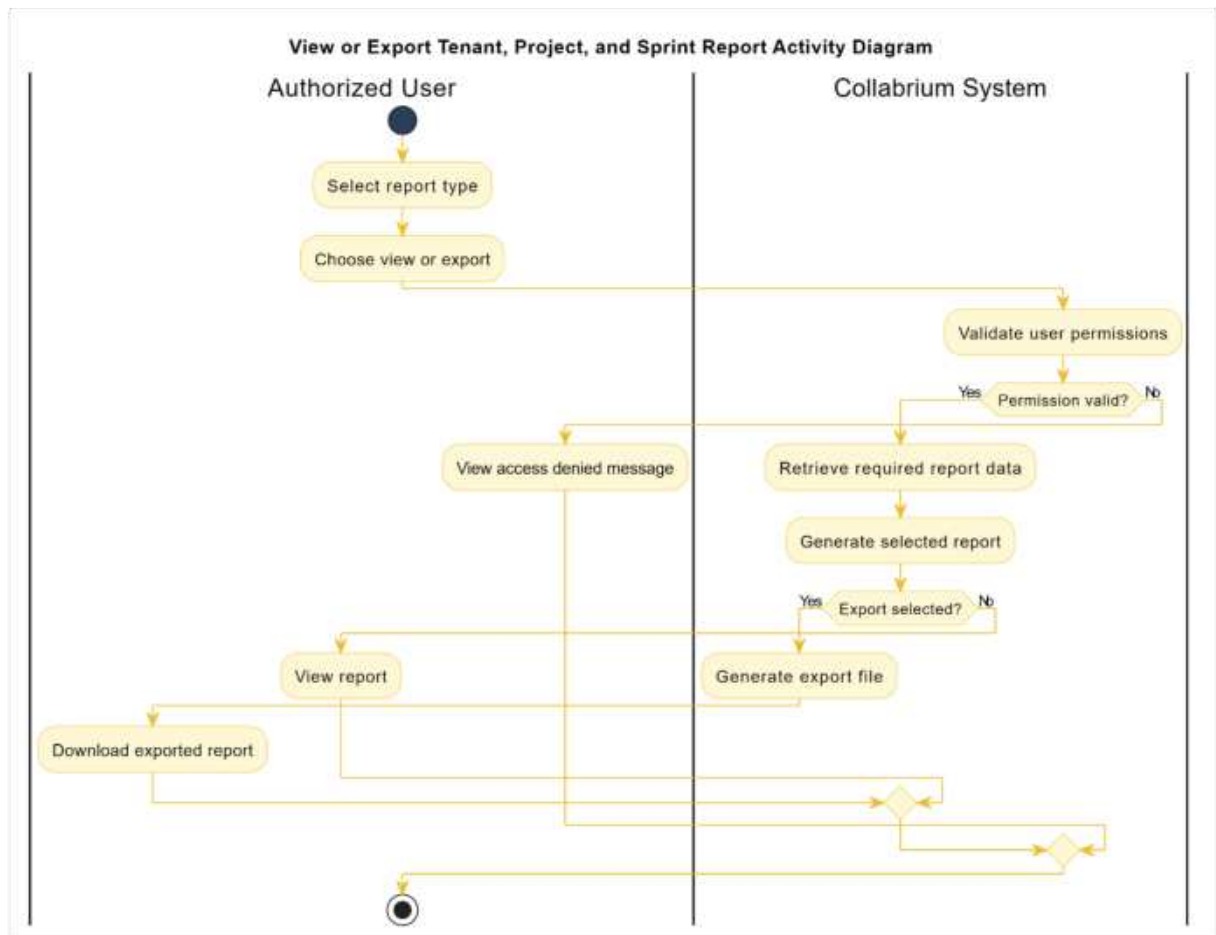


Figure 5.7.9 1 View or Export Tenant, Project, and Sprint Report Activity Diagram

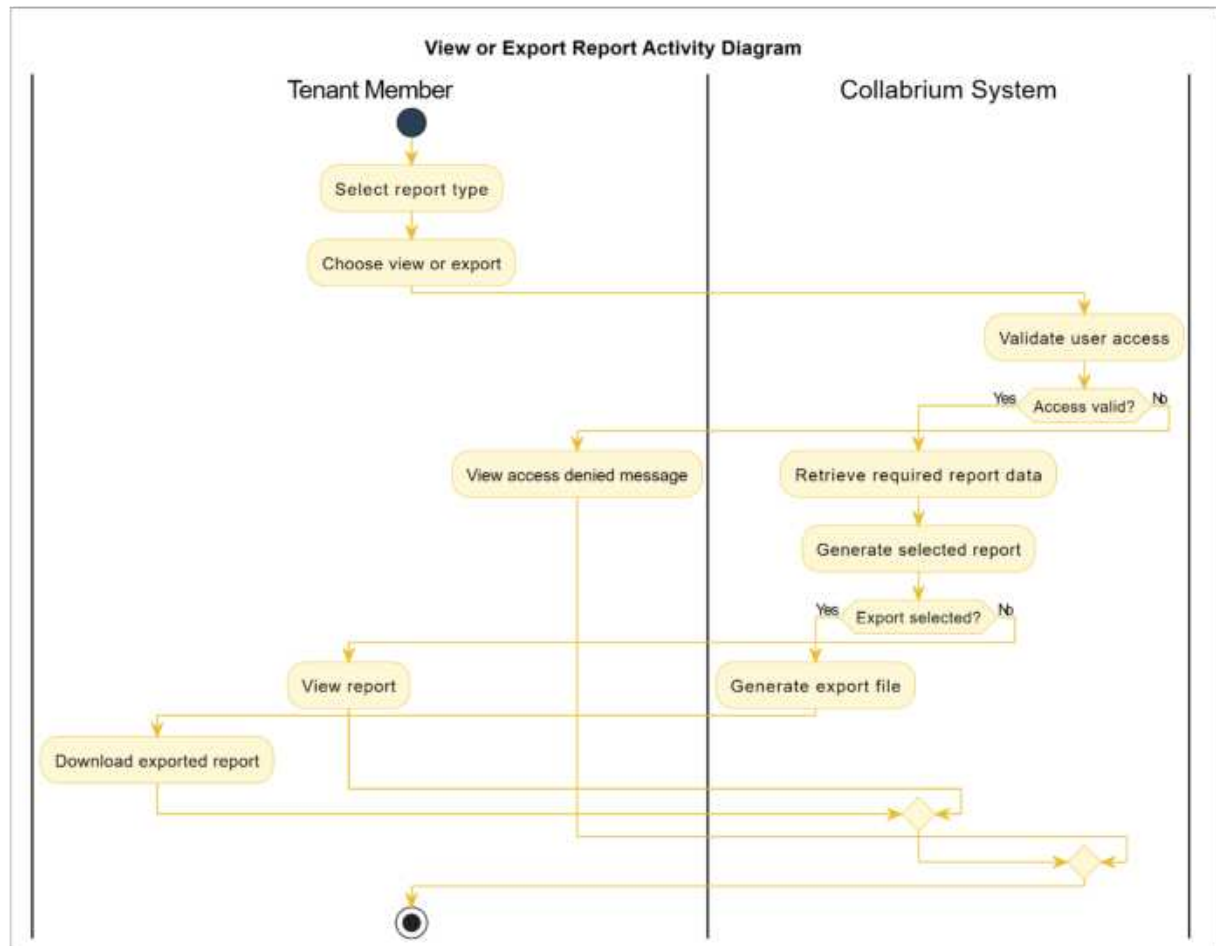


*Figure 5.7.9 2 View or Export Report Activity Diagram*

## 5.7.10 Learning Hub Subsystem Activity Diagrams

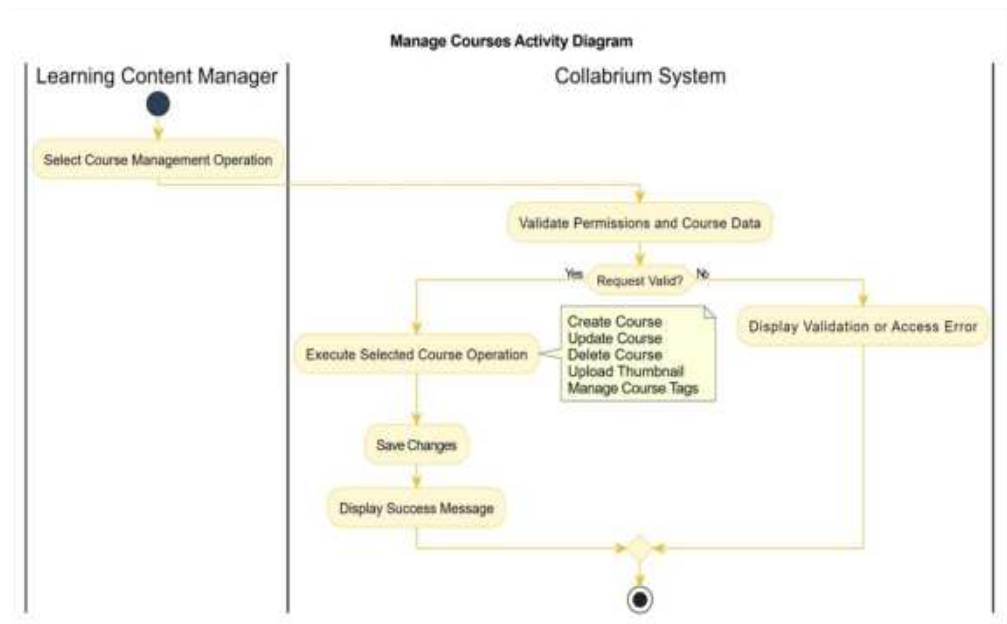


Figure 5.7.10 1 Manage Courses Activity Diagram

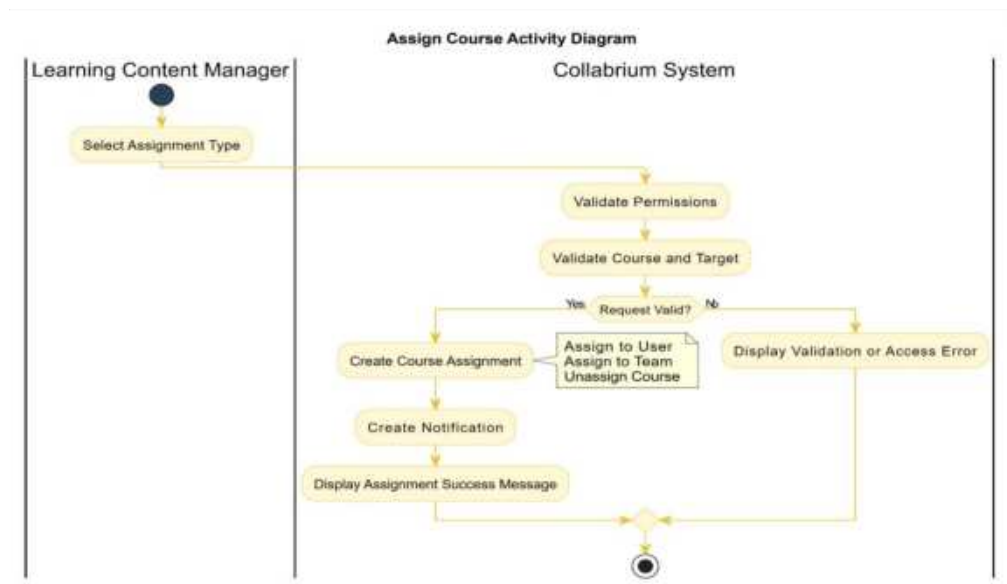


Figure 5.7.10 2 Assign Course Activity Diagram

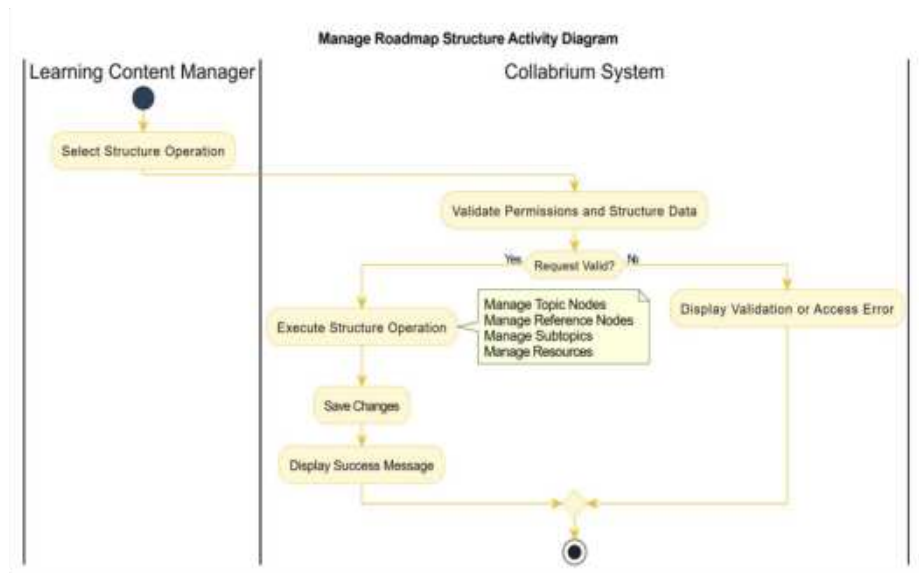


Figure 5.7.10 3 Manage Roadmap Structure Activity Diagram

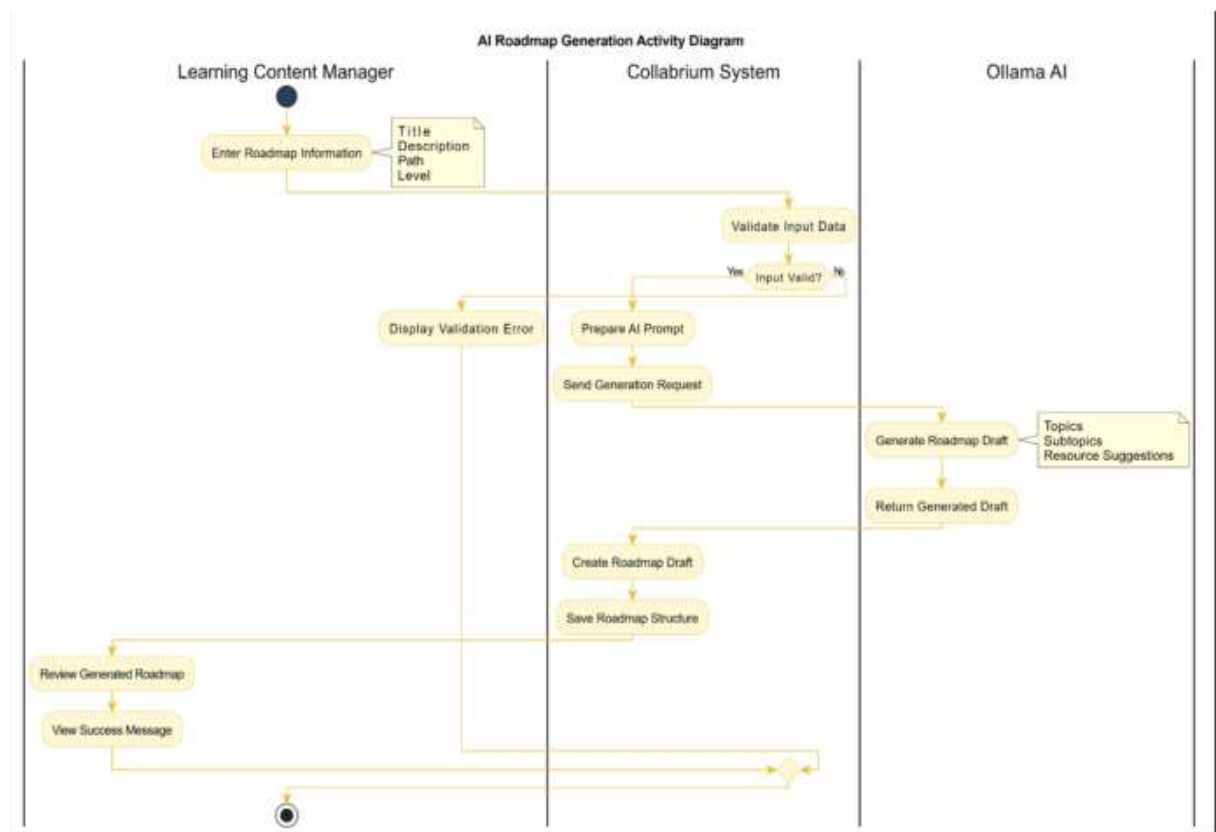


Figure 5.7.10 4 AI Roadmap Generation Activity Diagram

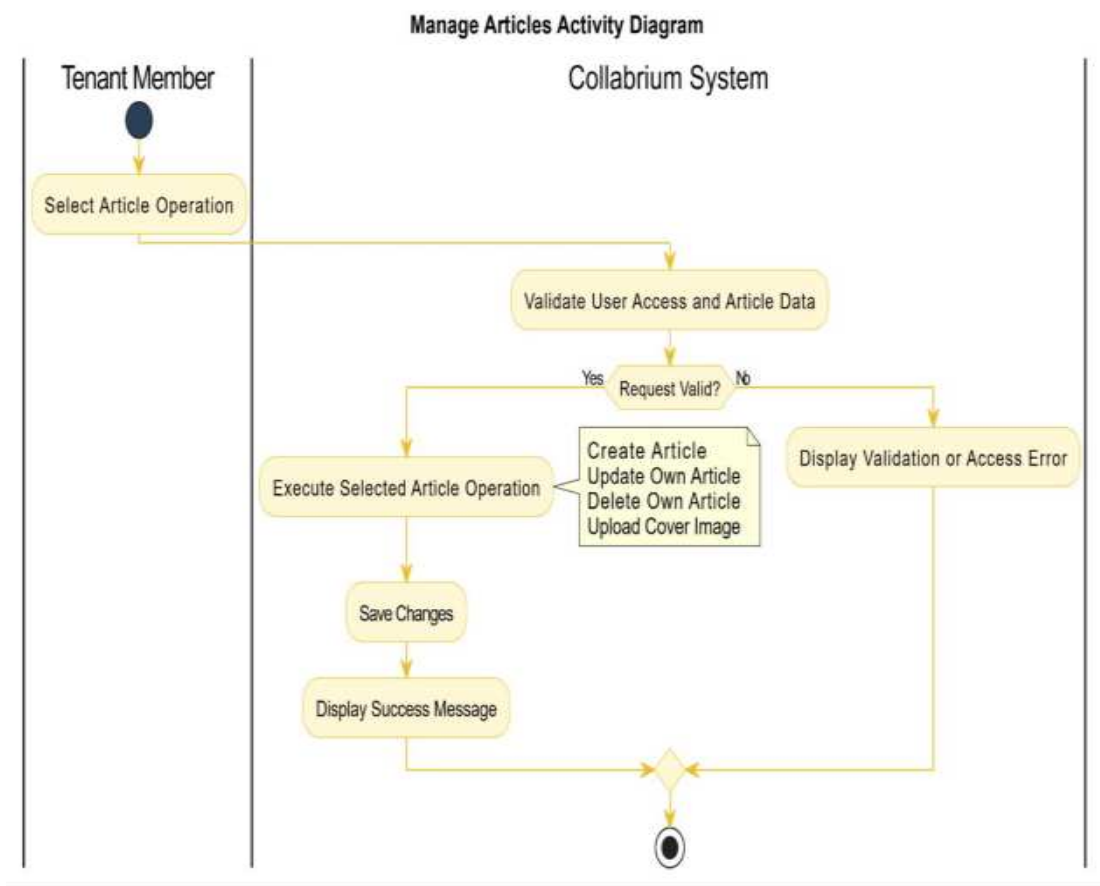


Figure 5.7.10 5 Manage Articles Activity Diagram

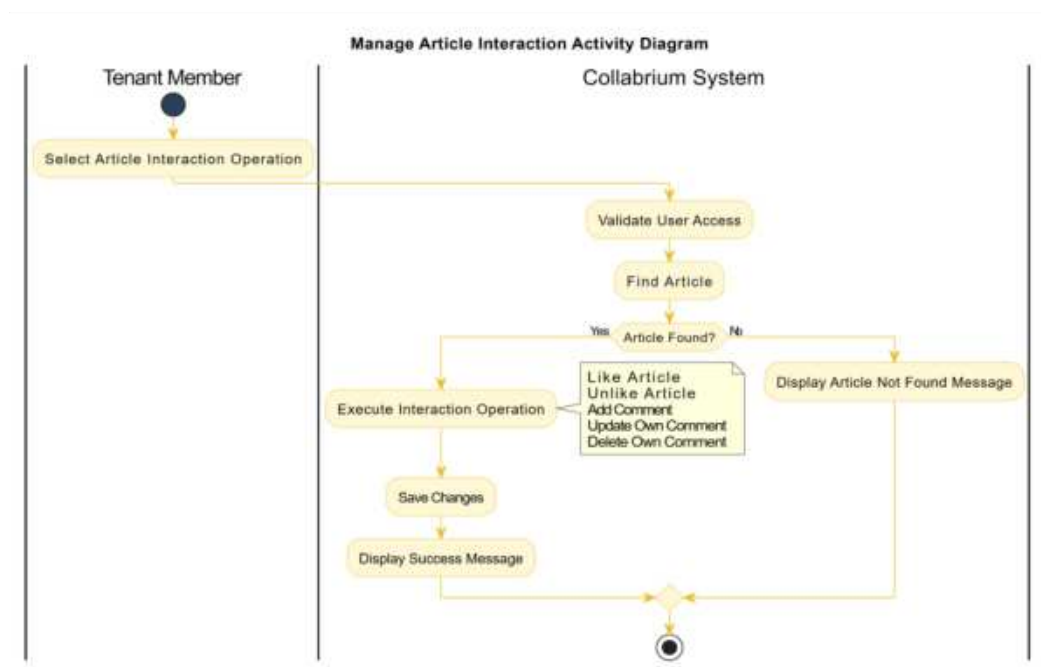


Figure 5.7.10 6 Manage Article Interaction Activity Diagram

### 5.7.11 Community Hub Subsystem Activity Diagrams

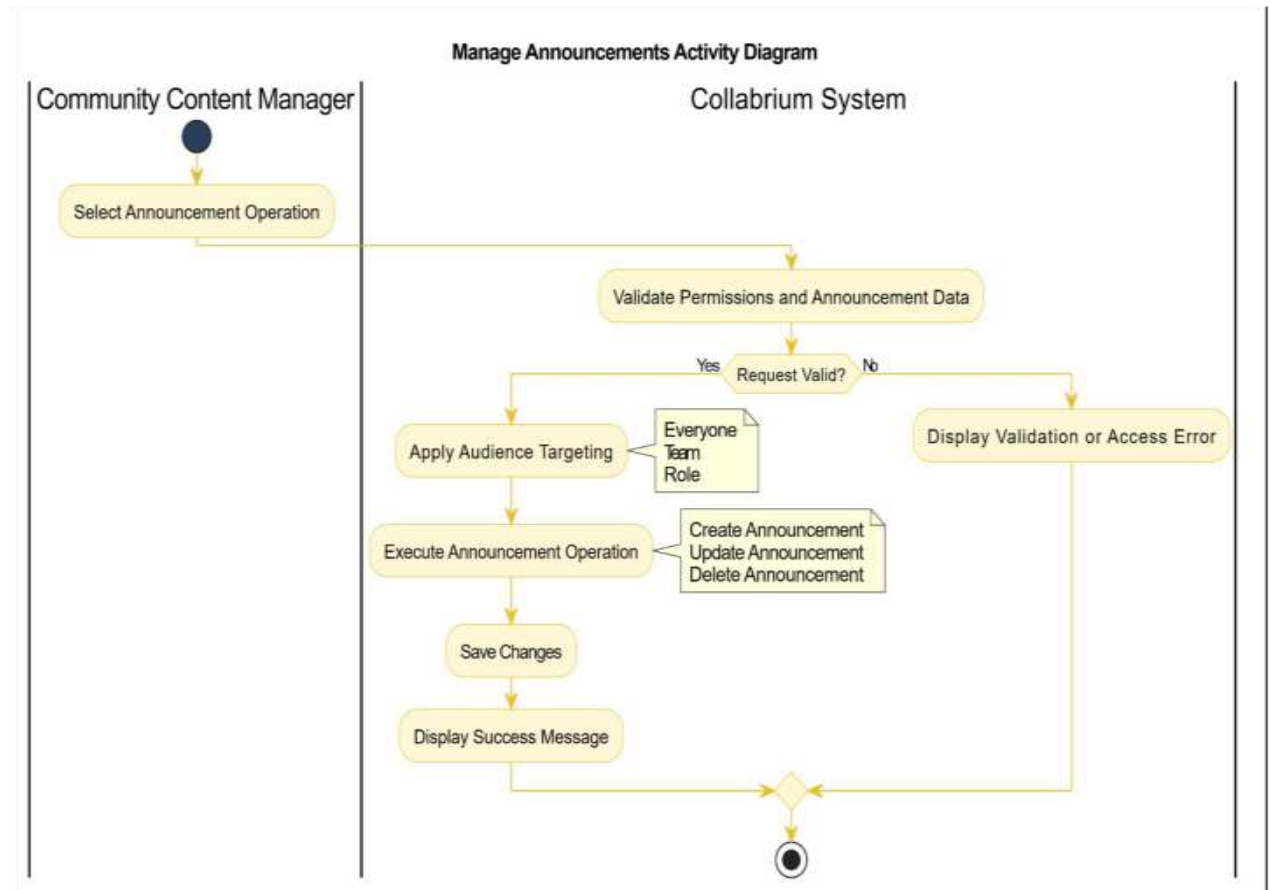


Figure 5.7.11 1 Manage Announcements Activity Diagram

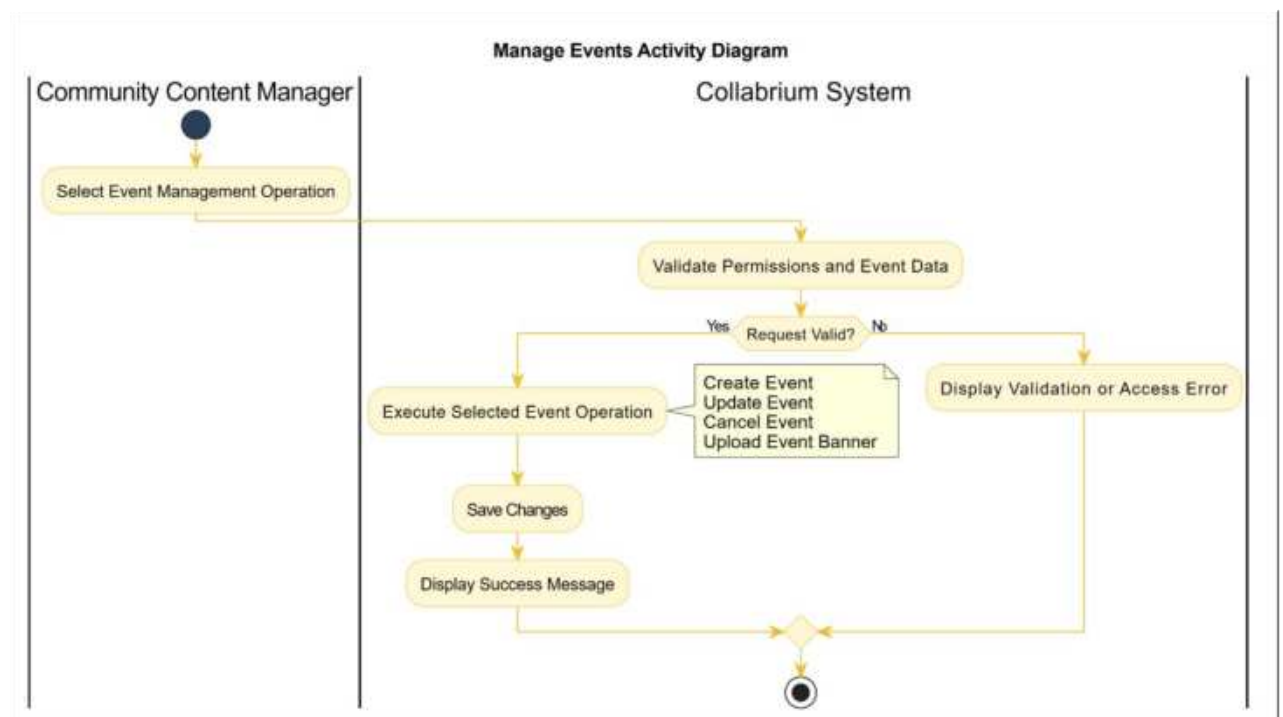


Figure 5.7.11 2 Manage Events Activity Diagram

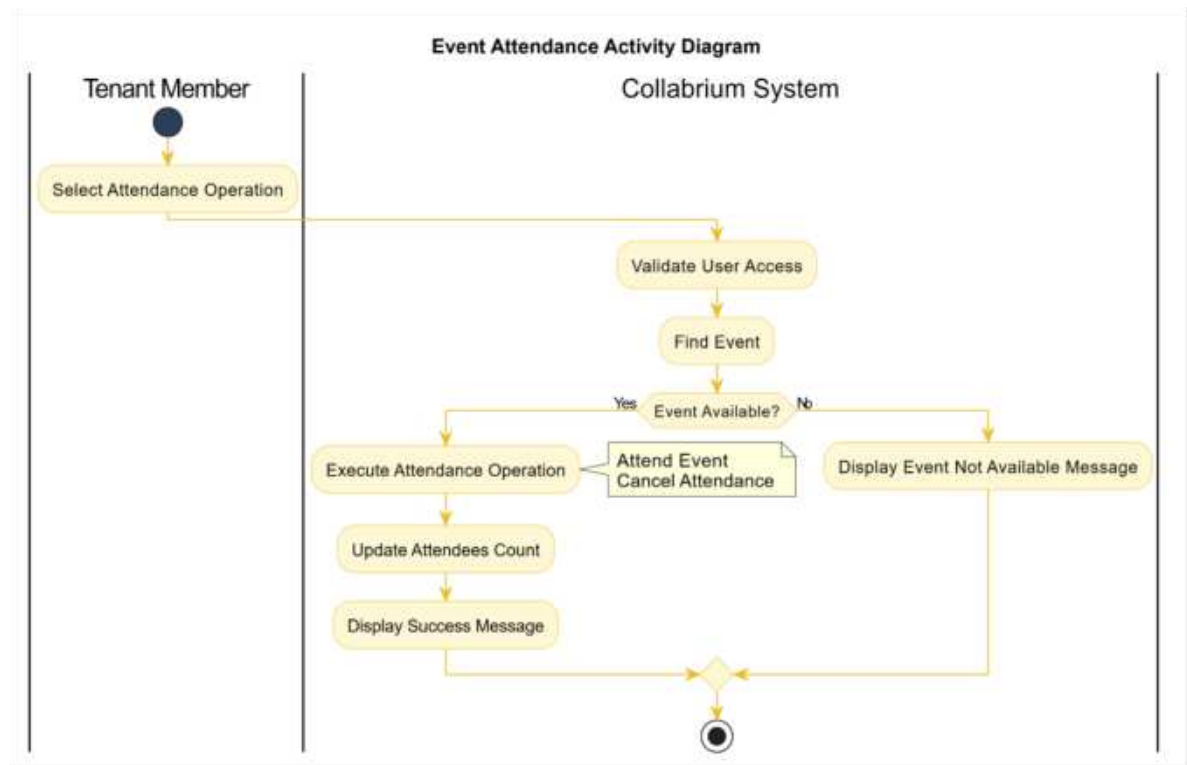


Figure 5.7.11 3 Event Attendance Activity Diagram

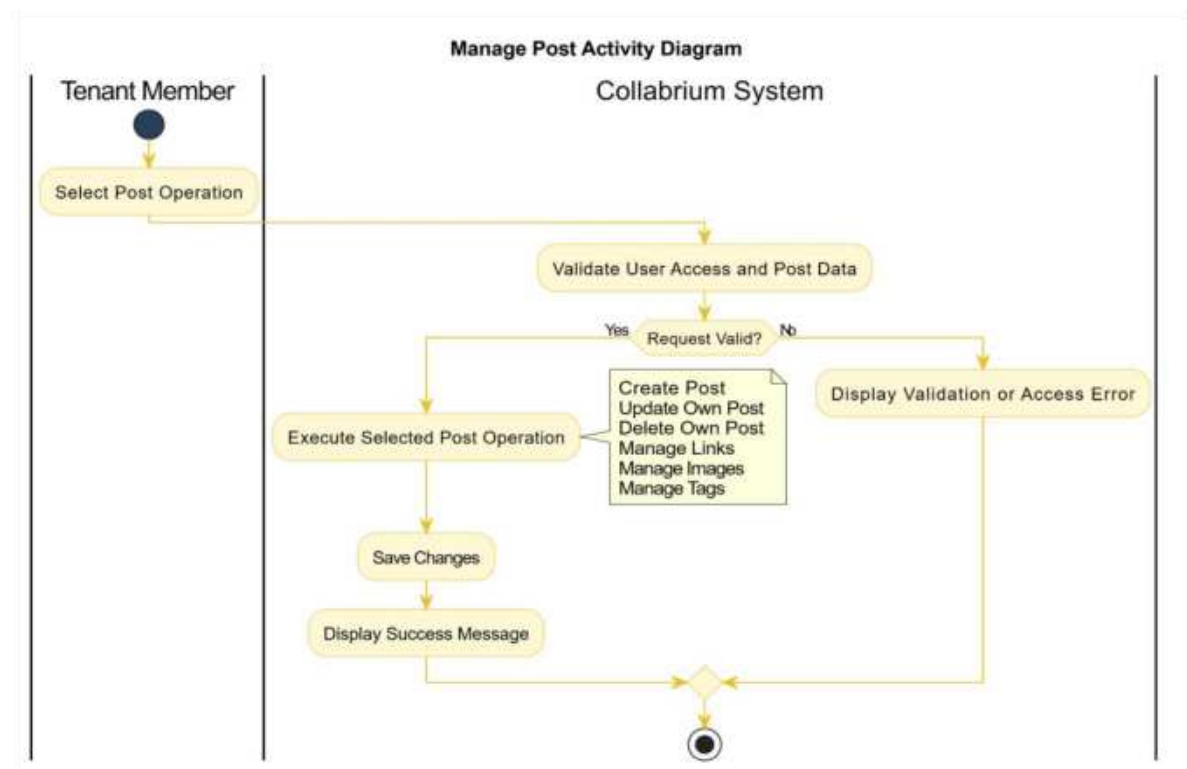


Figure 5.7.11 4 Manage Post Activity Diagram

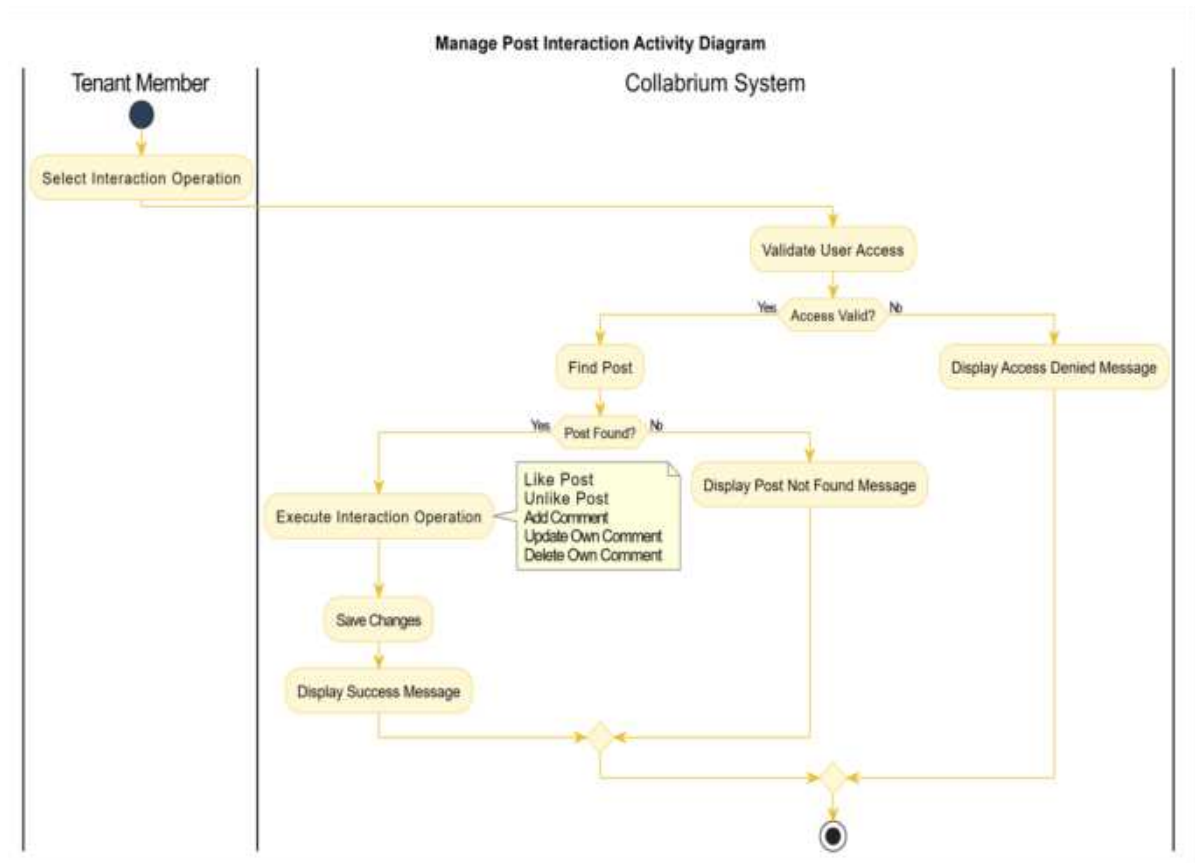


Figure 5.7.11 5 Manage Post Interaction Activity Diagram

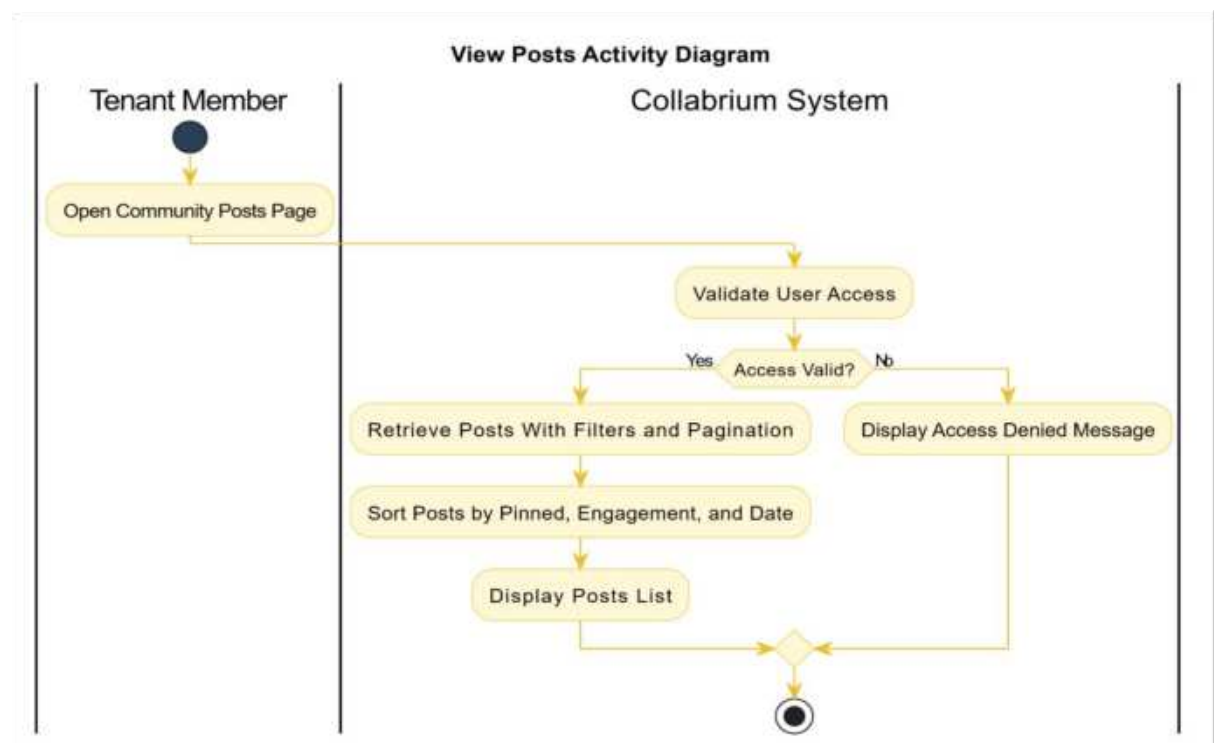


Figure 5.7.11 6 View Posts Activity Diagram

5.7.12 AI Assistant Management Subsystem Activity Diagrams

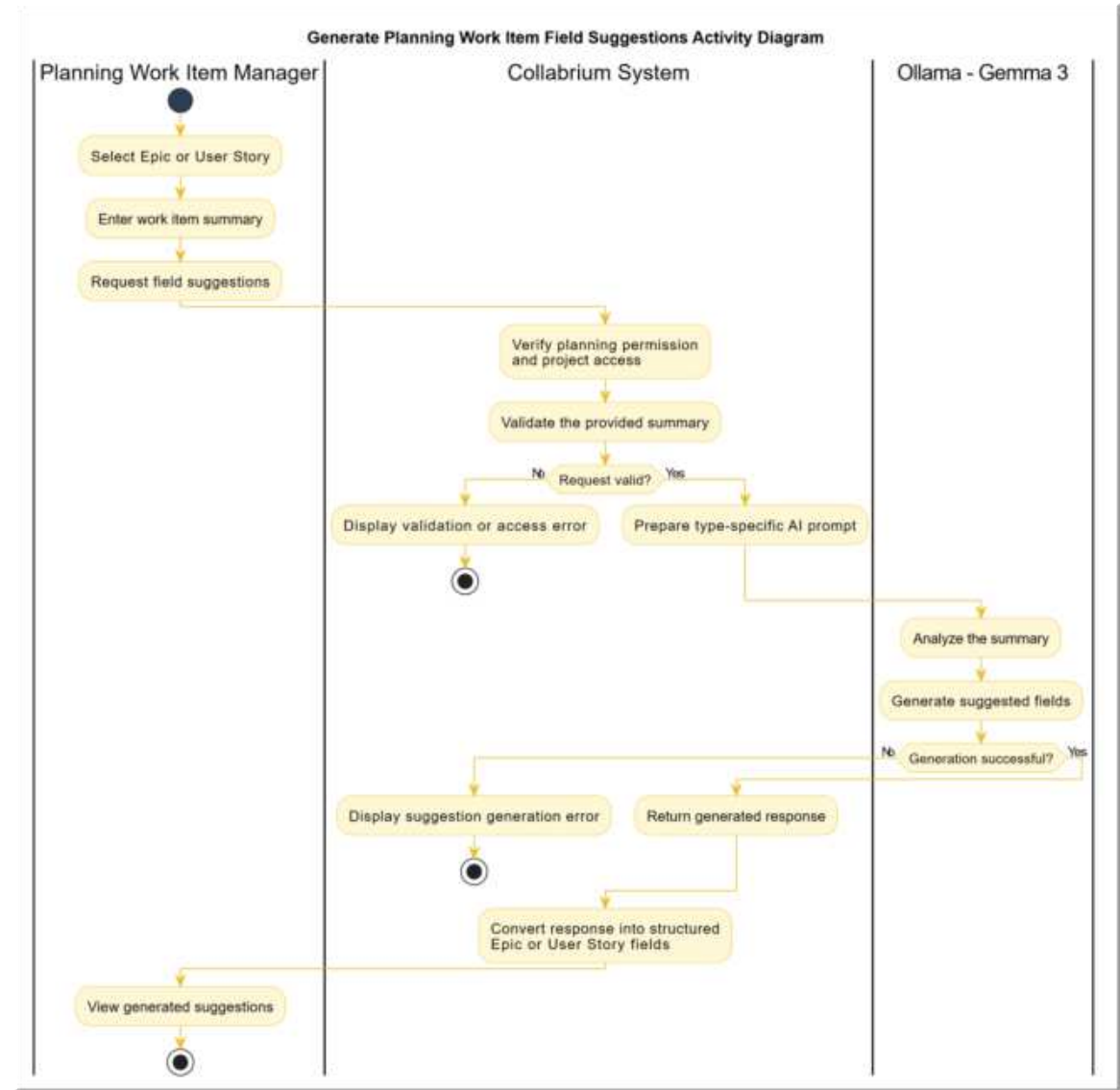


Figure 5.7.12 1 Generate Planning Work Item Field Suggestions Activity Diagram

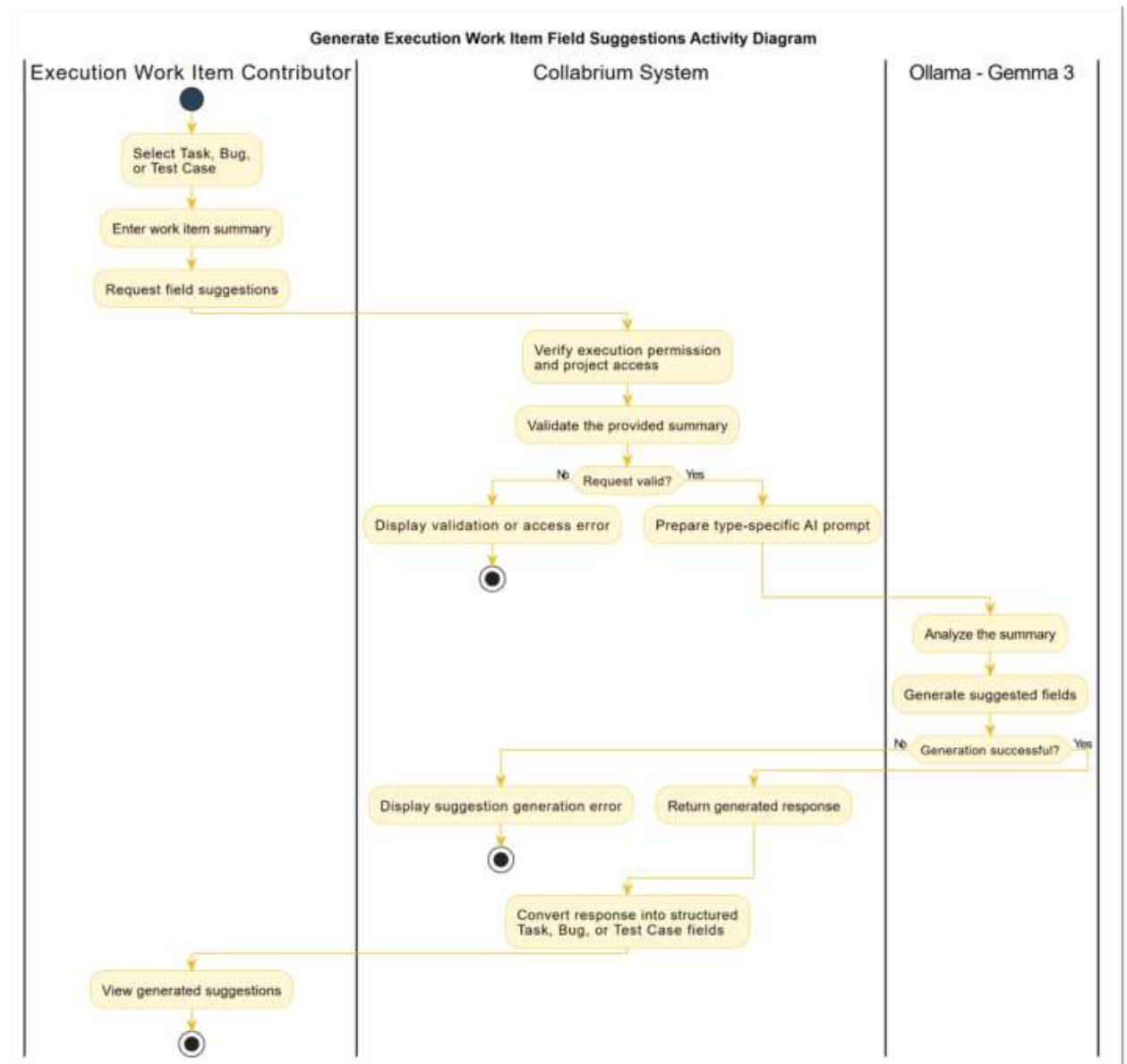


Figure 5.7.12 2 Generate Execution Work Item Field Suggestions Activity Diagram

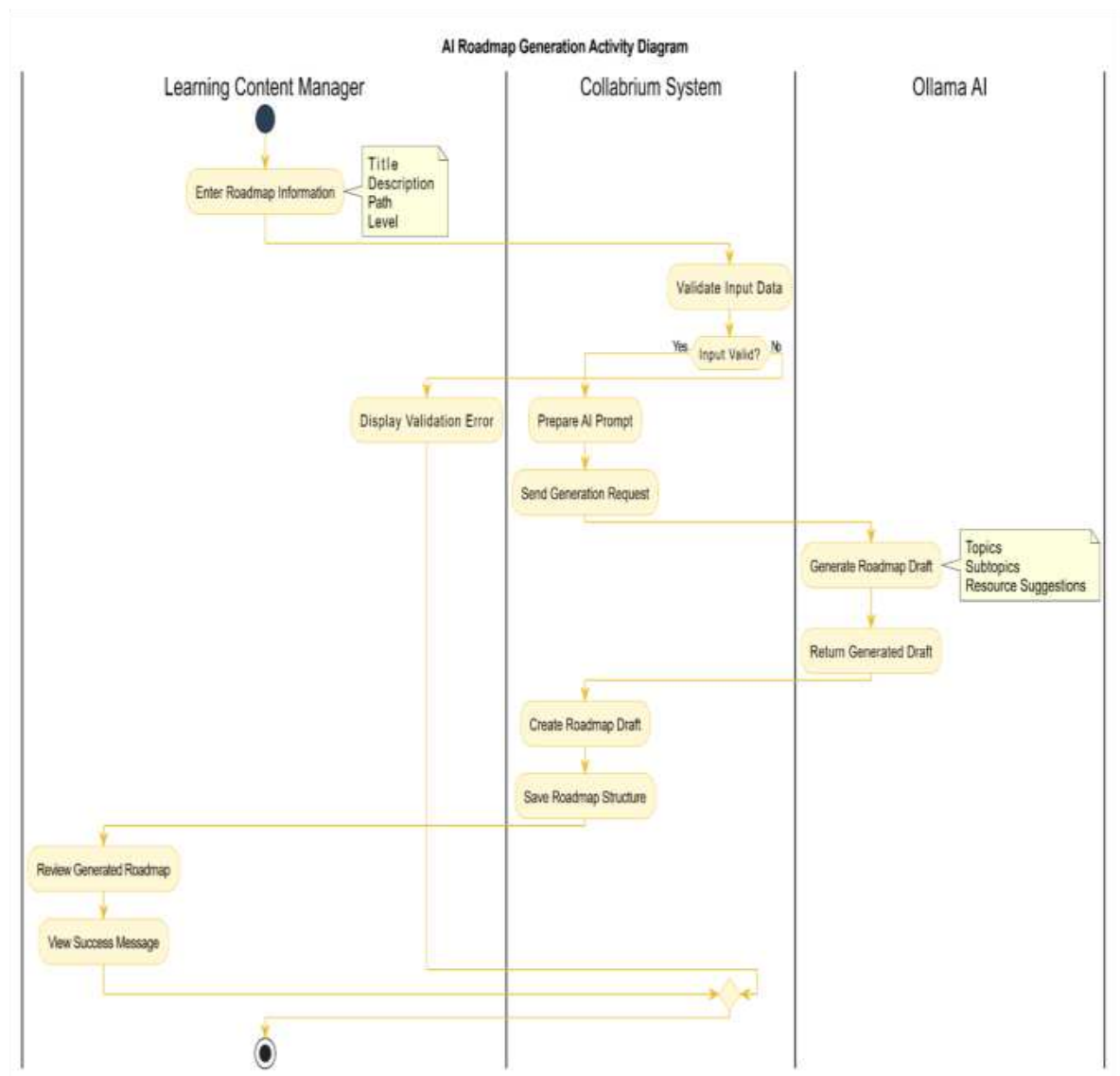


Figure 5.7.12 3 AI Roadmap Generation Activity Diagram

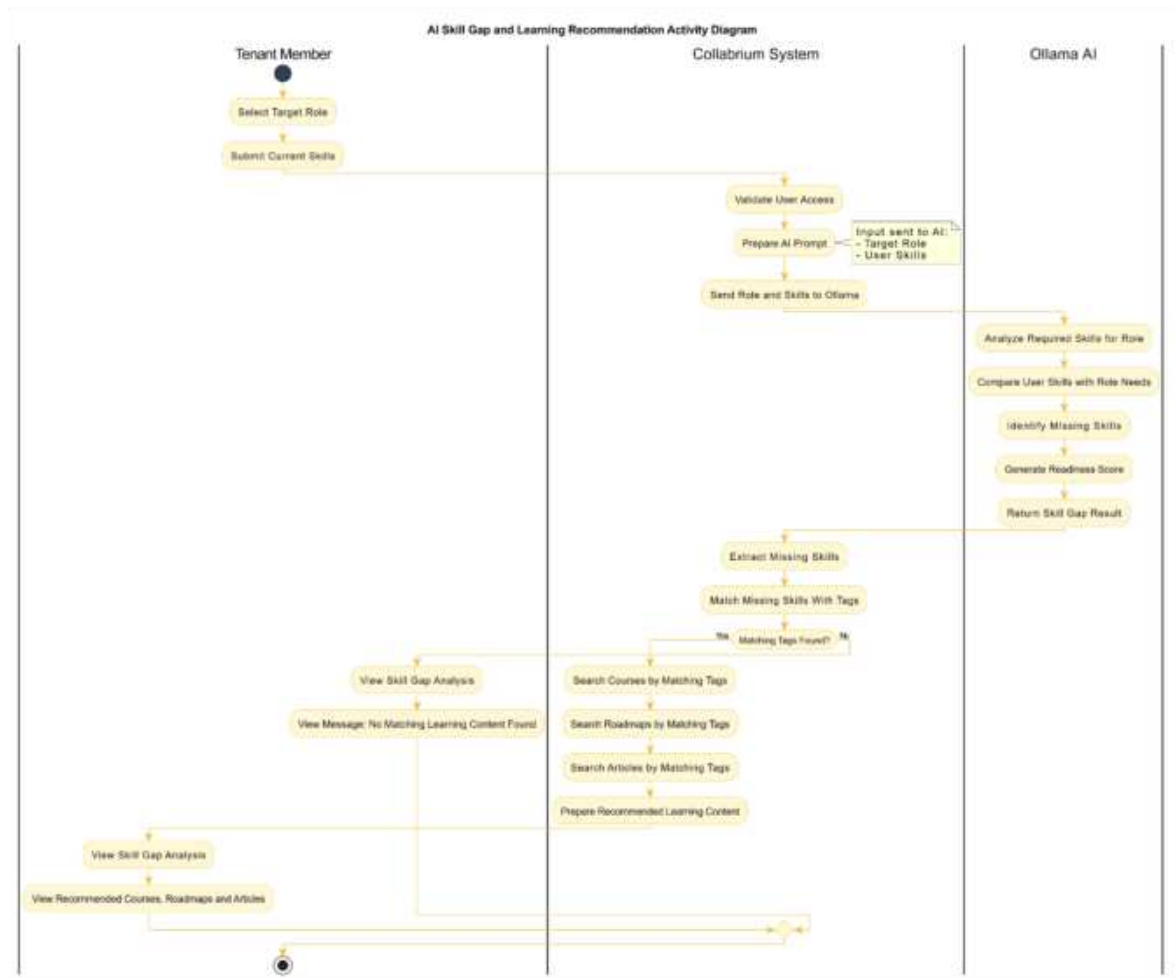


Figure 5.7.12 4 AI Skill Gap and Learning Recommendation Activity Diagram

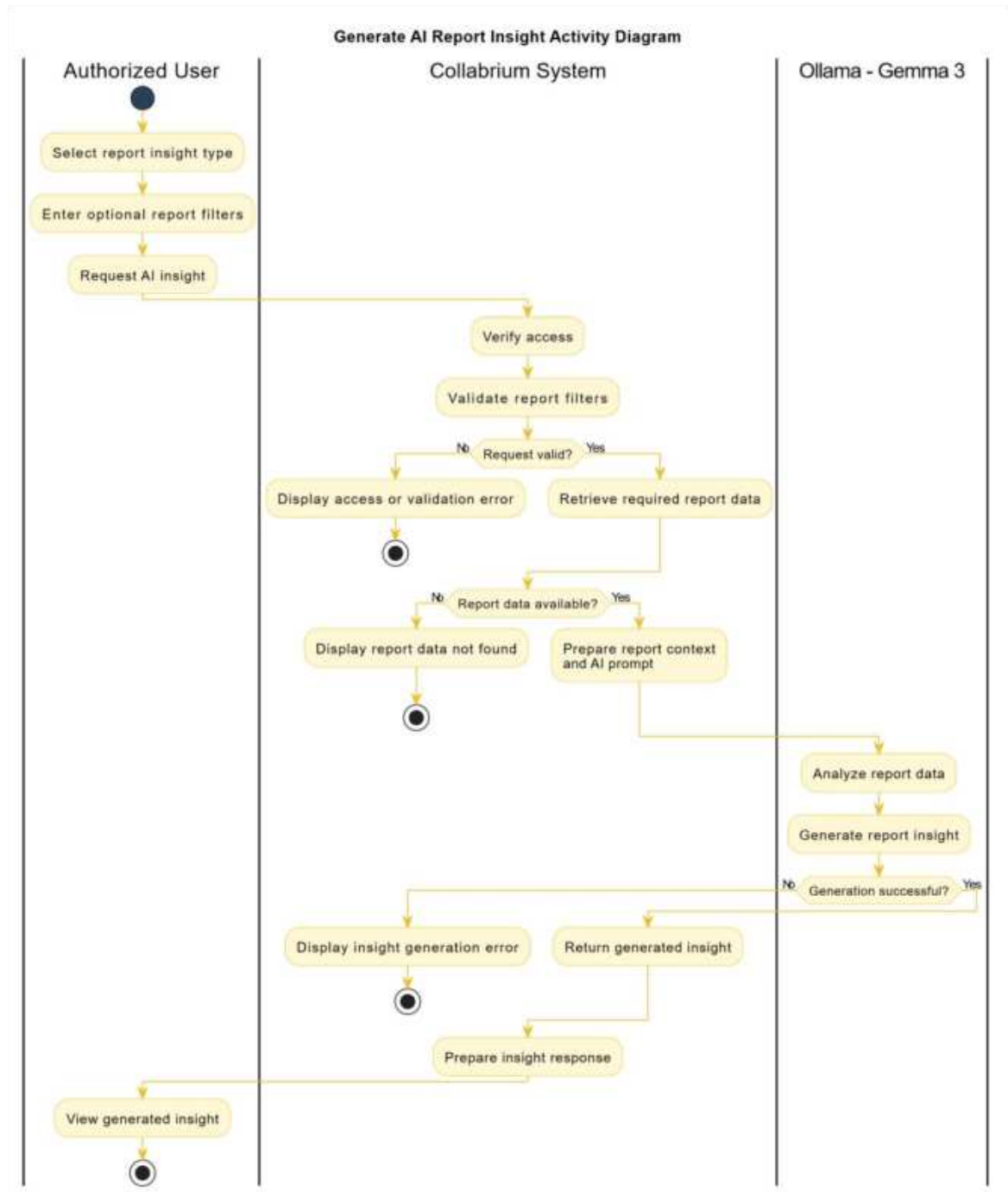


Figure 5.7.12 5 Generate AI Report Insight Activity Diagram

### 5.7.13 Notifications and Alerts Subsystem Activity Diagrams

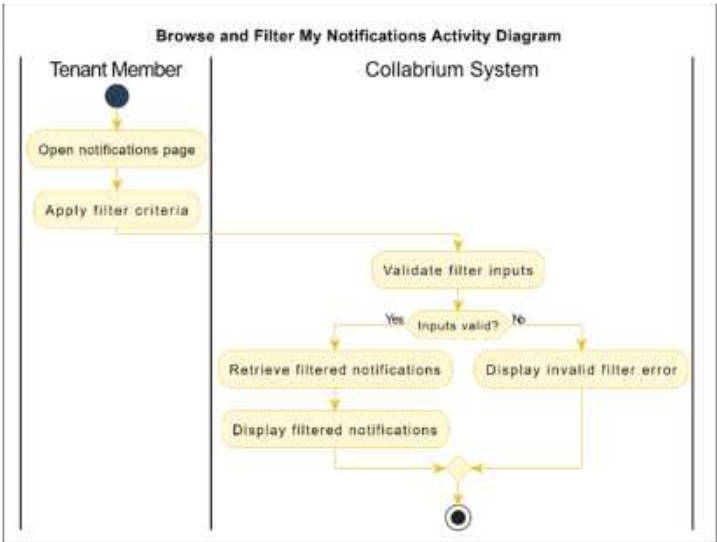


Figure 5.7.13 1 Browse and Filter My Notifications Activity Diagram

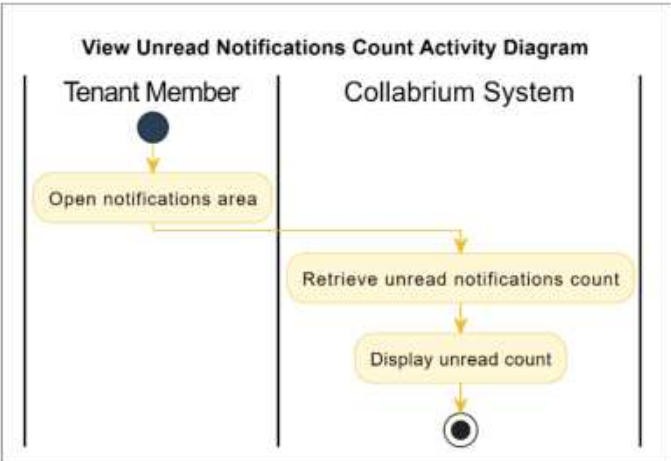


Figure 5.7.13 2 View Unread Notifications Count Activity Diagram

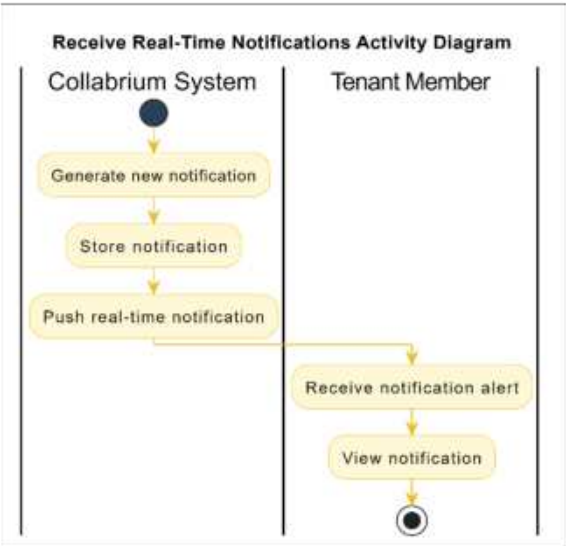


Figure 5.7.13 3 Receive Real - Time Notifications Activity Diagram

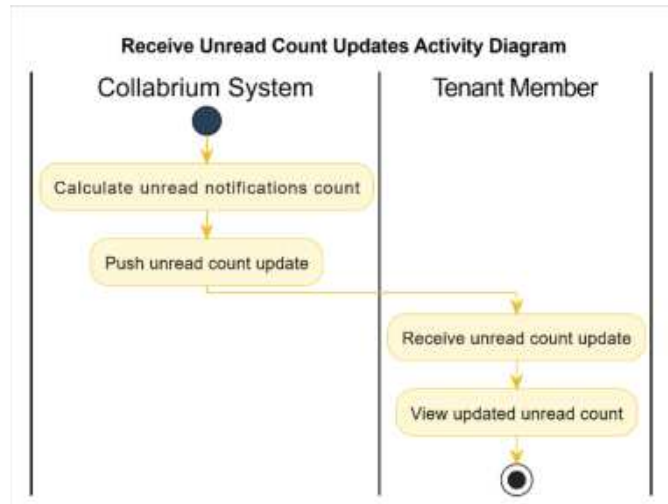


Figure 5.7.13 4 Receive Unread Count Updates Activity Diagram

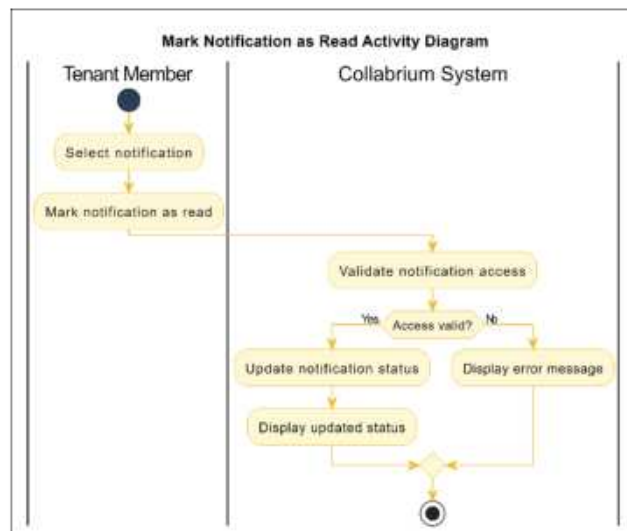


Figure 5.7.13 5 Mark Notification Read Activity Diagram

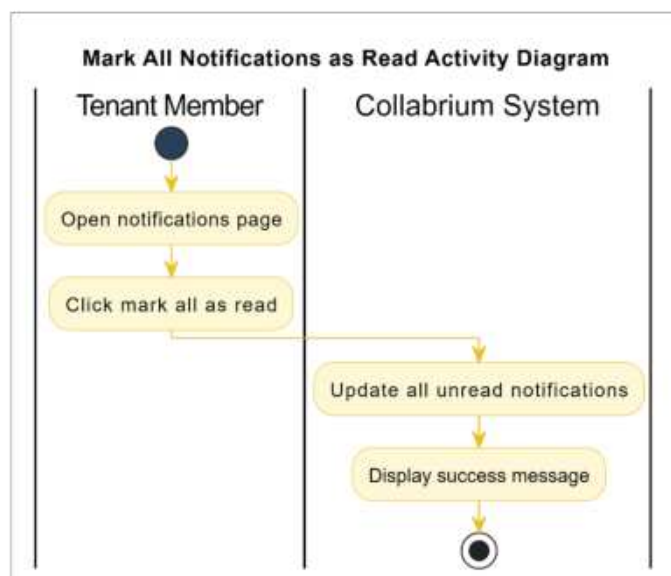


Figure 5.7.13 6 Mark All Notifications as Read Activity Diagram

## 5.8 System Sequence Diagrams

### 5.8.1 Authentication Subsystem System Sequence Diagrams

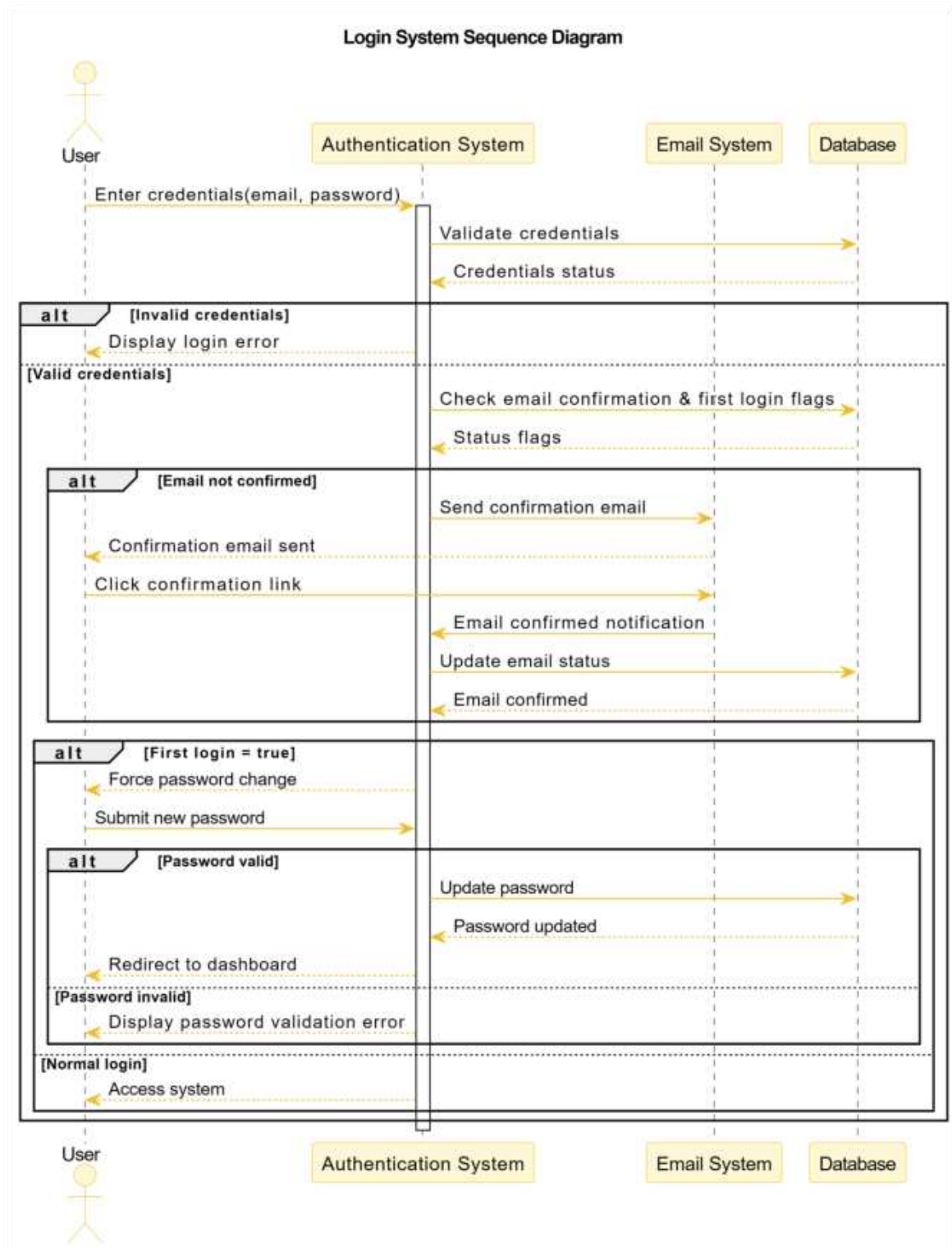


Figure 5.8.1 1 Login System Sequence Diagram

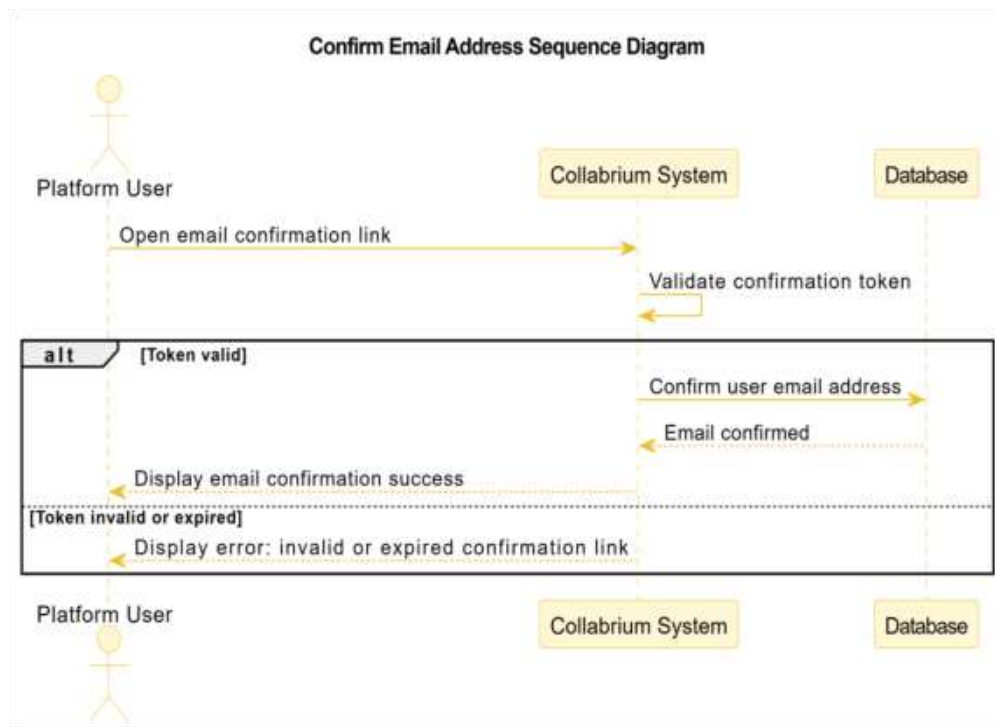


Figure 5.8.1 2 Confirm Email Address Sequence Diagram

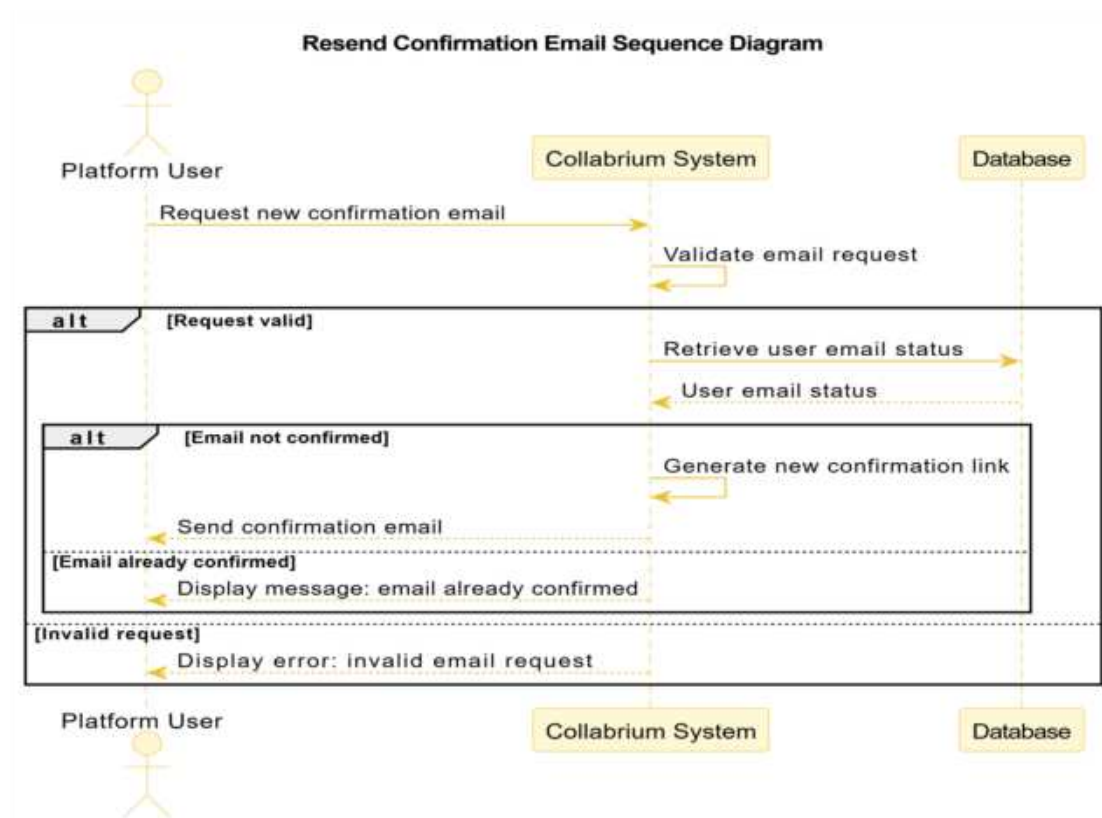


Figure 5.8.1 3 Resend Confirmation Email Sequence Diagram

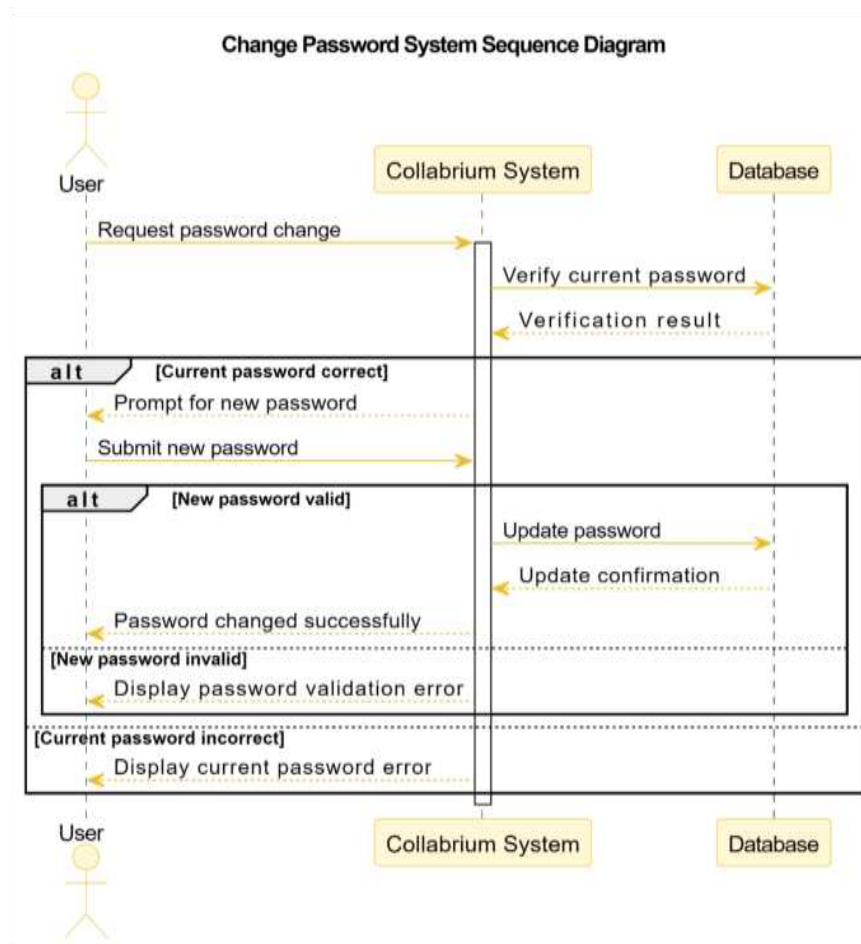


Figure 5.8.1 4 Change Password System Sequence Diagram

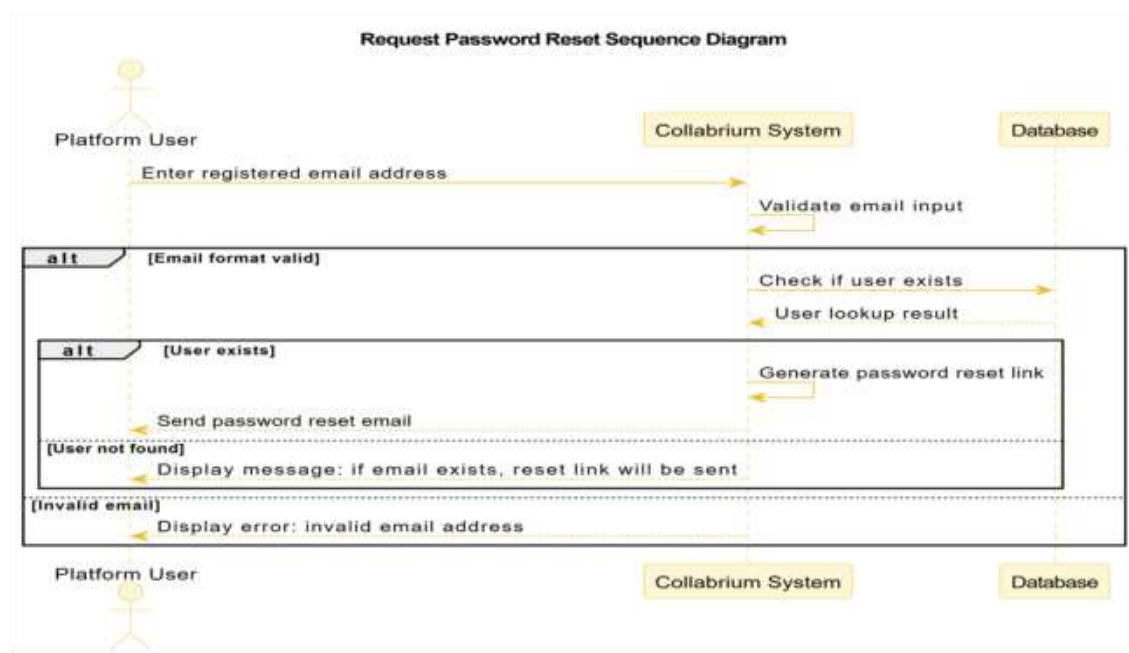


Figure 5.8.1 5 Reset Password System Sequence Diagram

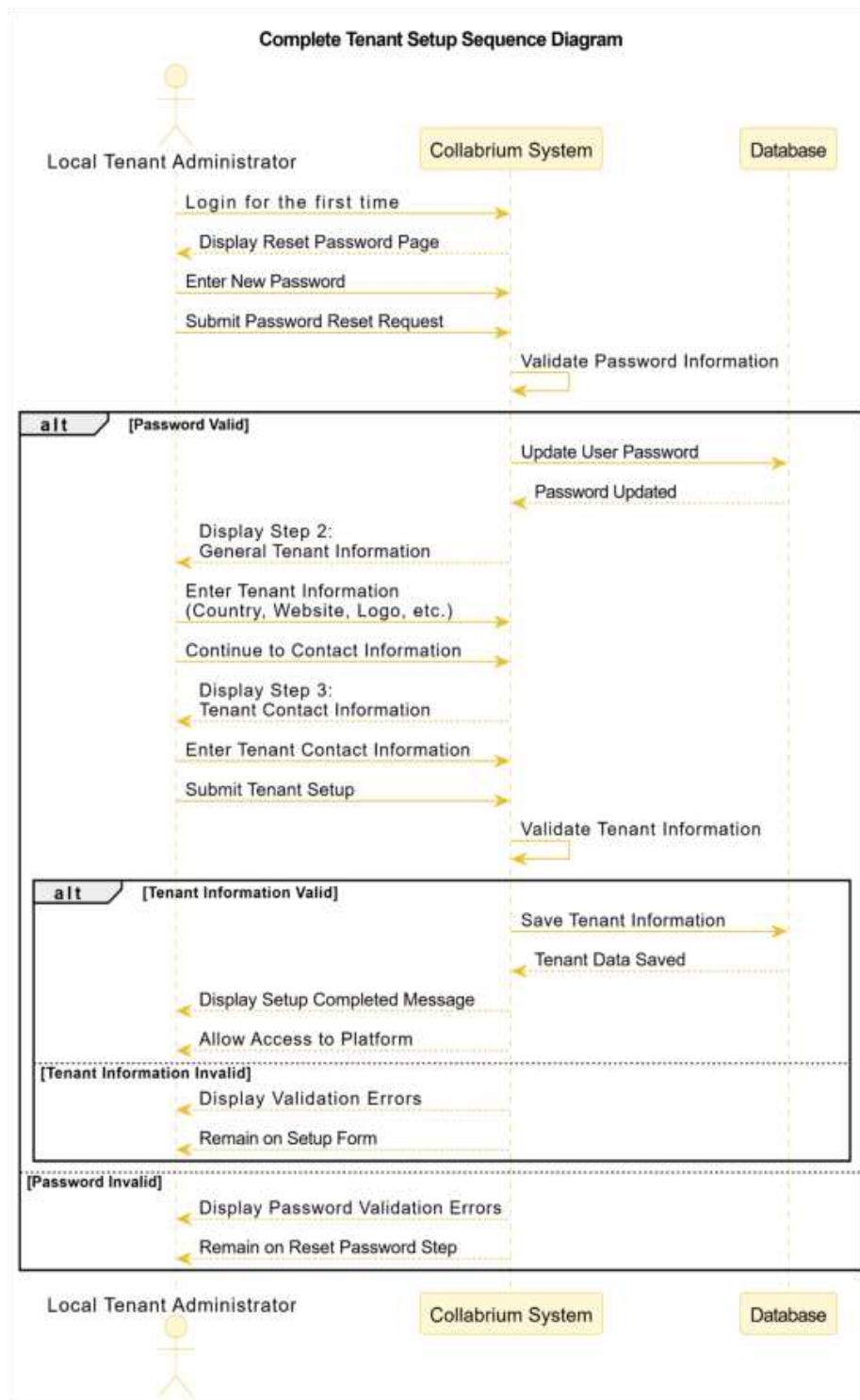


Figure 5.8.1 6 Complete Tenant Setup System Sequence Diagram

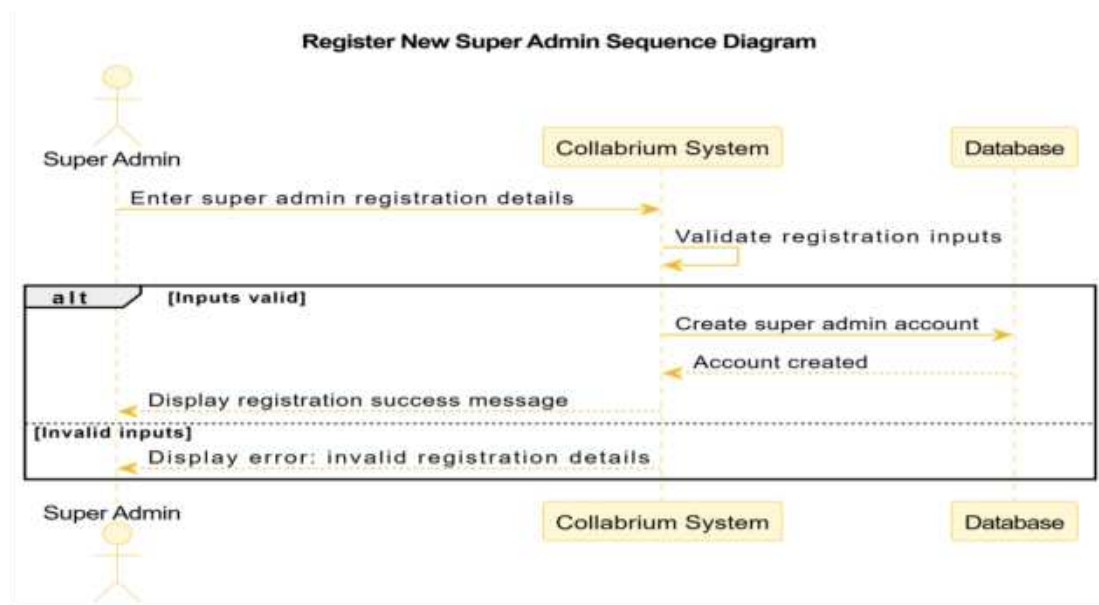


Figure 5.8.1 7 Register New Super Admin Sequence Diagram

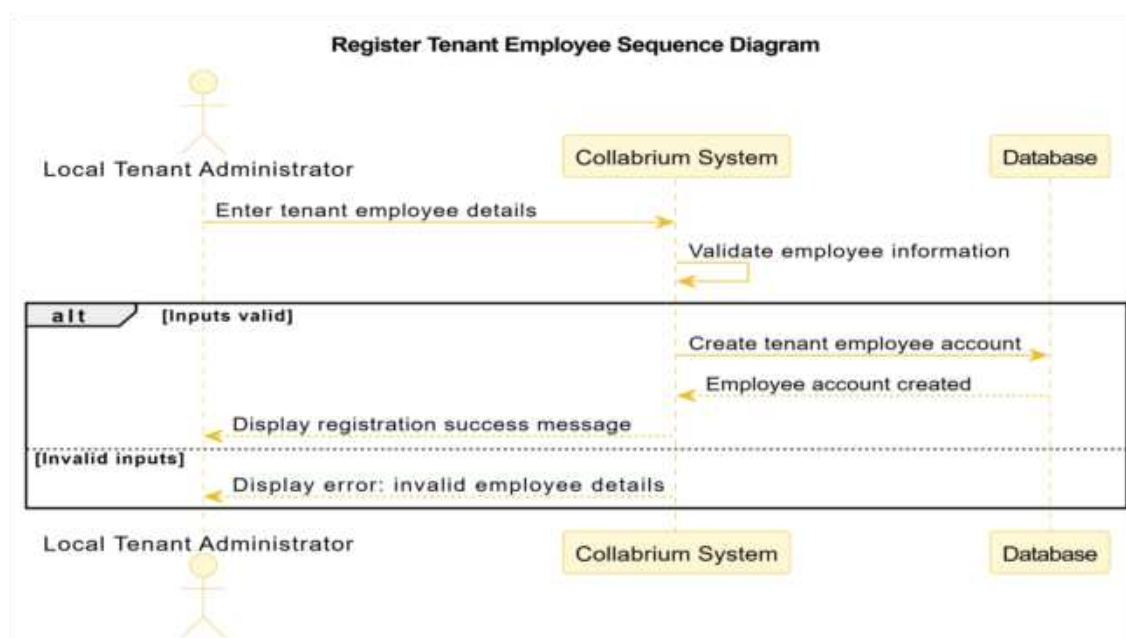


Figure 5.8.1 8 Register Tenant Employee Sequence Diagram

## 5.8.2 Platform Services Subsystem System Sequence Diagrams

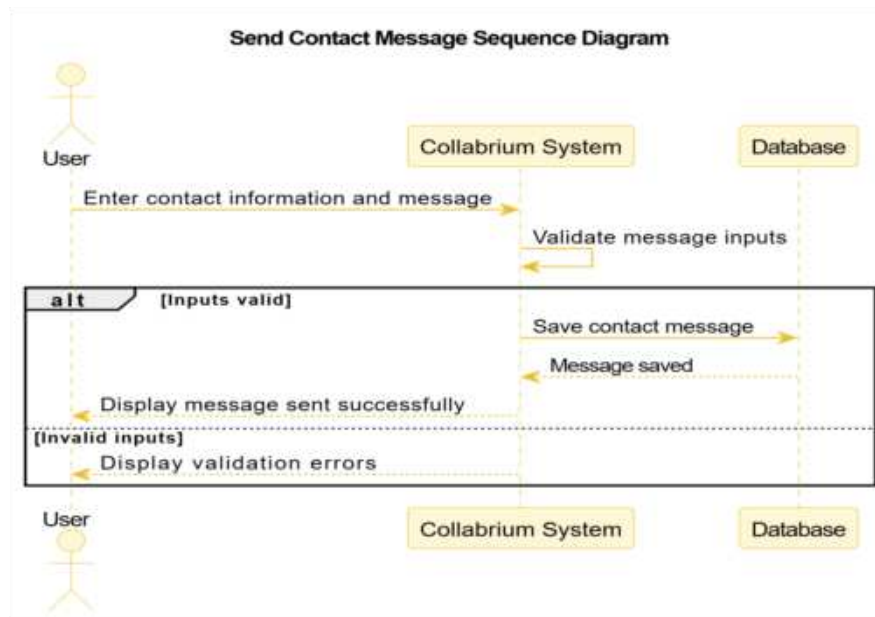


Figure 5.8.2 1 Send Contact Message System Sequence Diagram

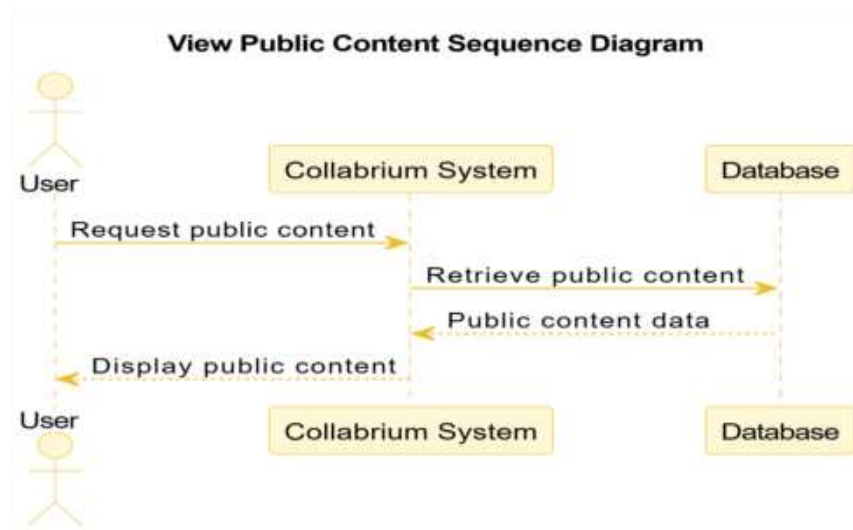


Figure 5.8.2 2 View Public Content Sequence Diagram

### 5.8.3 Tenant Management Subsystem System Sequence Diagrams

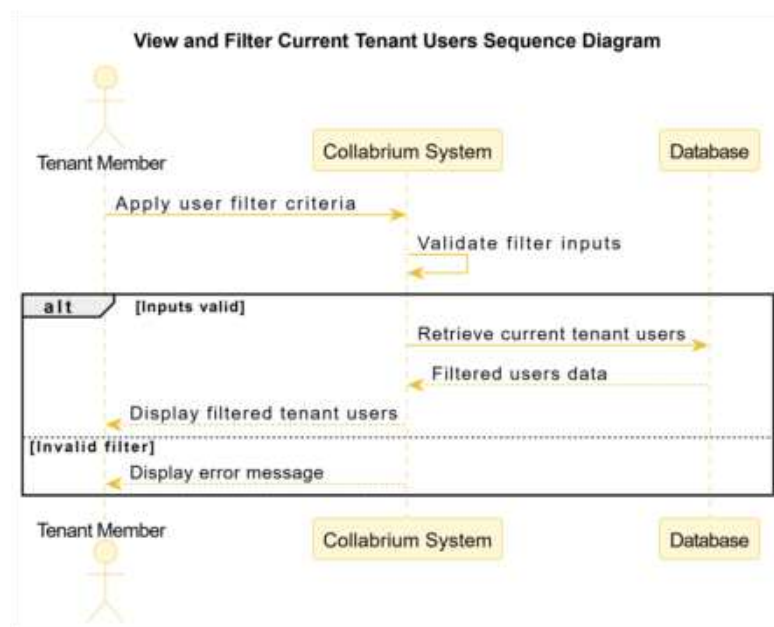


Figure 5.8.3 1 View and Filter Current Tenant User System Sequence Diagram

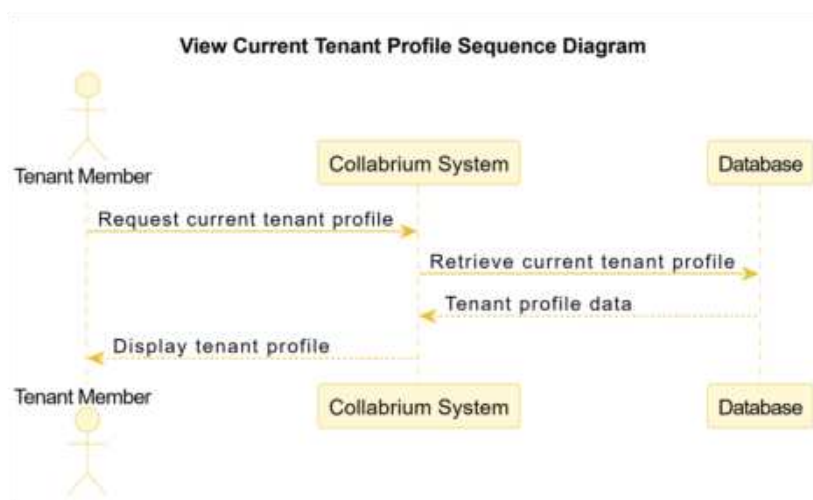


Figure 5.8.3 2 View Current Tenant Profile System Sequence Diagram

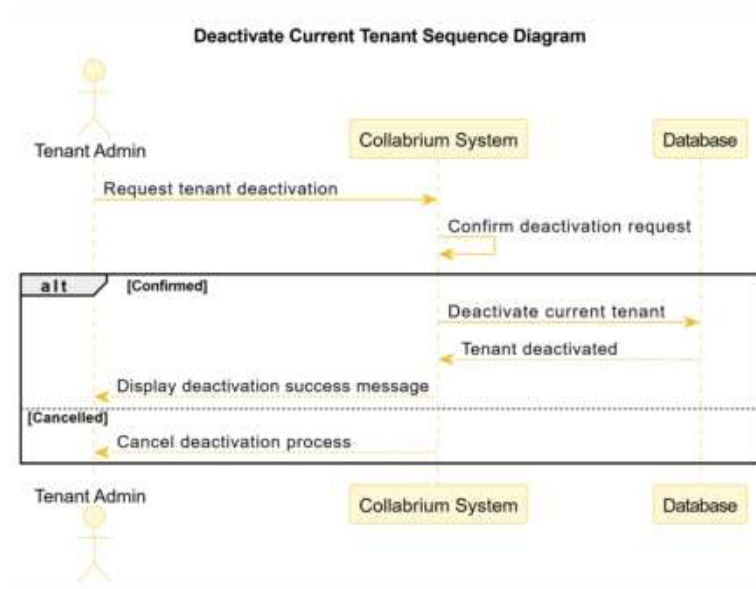


Figure 5.8.3 3 Deactivate Current Tenant Activity System Sequence Diagram

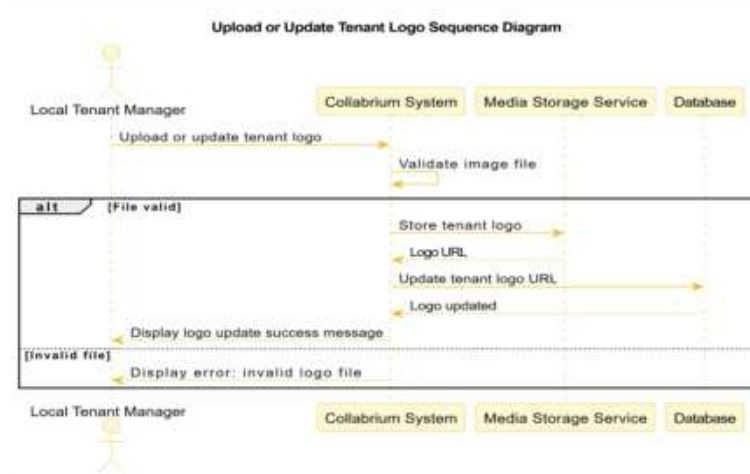


Figure 5.8.3 4 Upload or Update Tenant Logo System Sequence Diagram

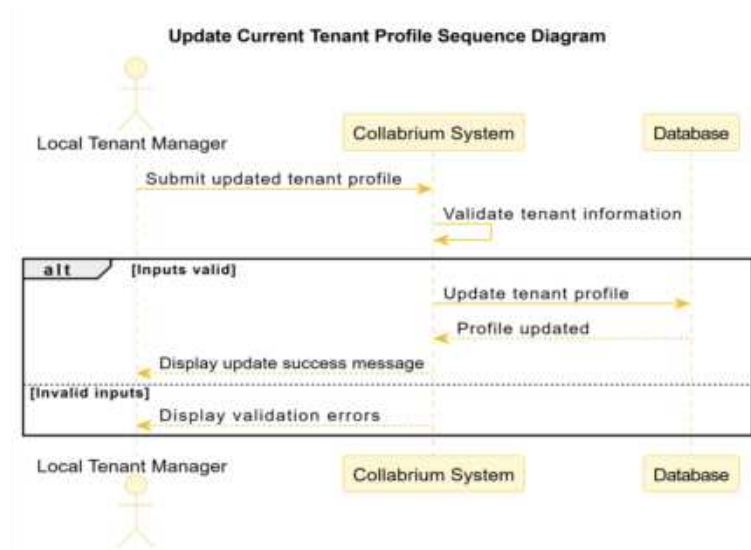


Figure 5.8.3 5 Update Current Tenant Profile System Sequence Diagram

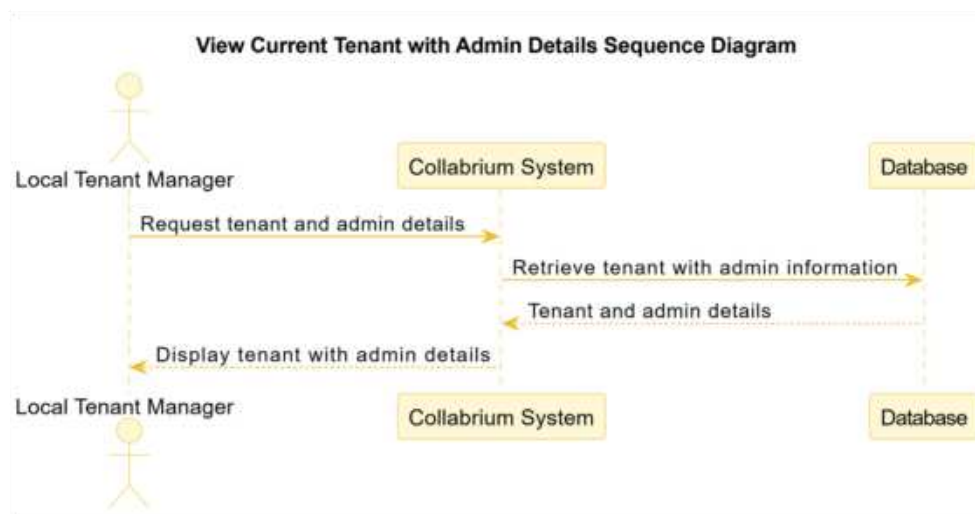


Figure 5.8.3 6 View Current Tenant with Admin Details System Sequence Diagram

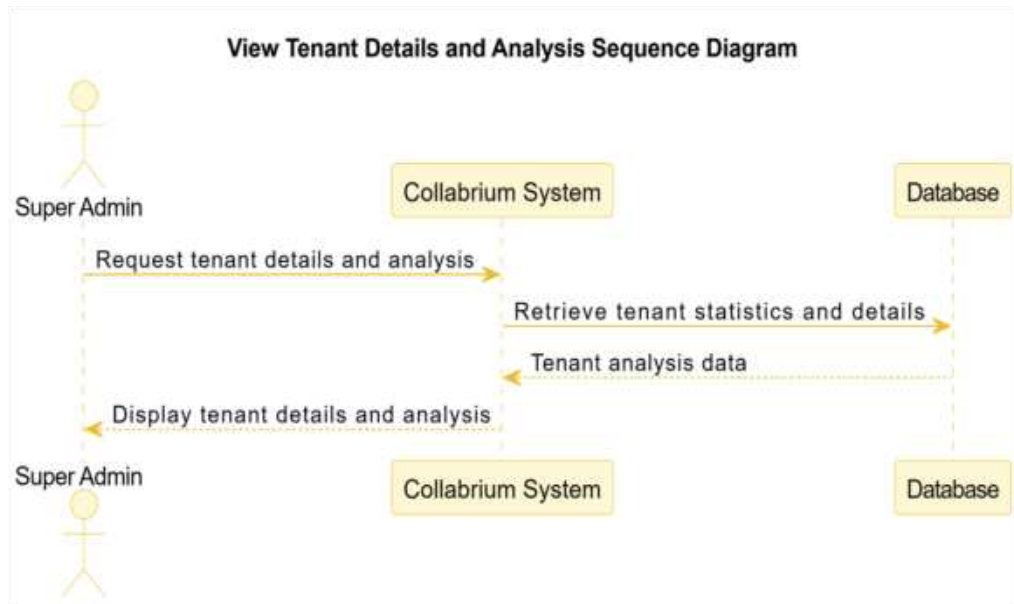


Figure 5.8.3 7 View Tenant Details and Analysis System Sequence Diagram

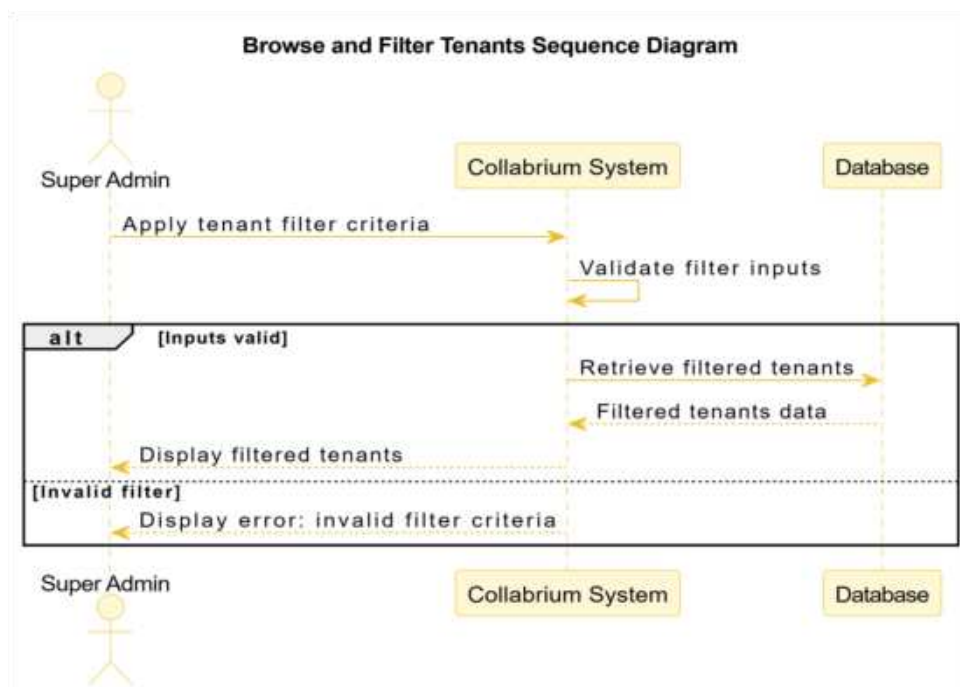


Figure 5.8.3 8 Browse and Filter Tenants System Sequence Diagram

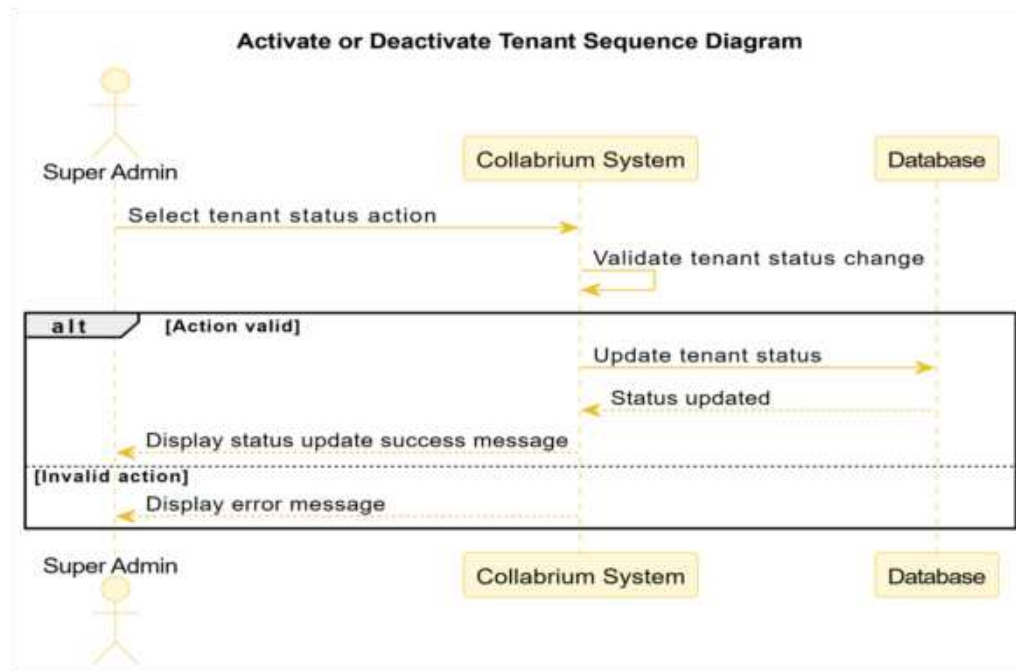


Figure 5.8.3 9 Activate or Deactivate System Sequence Diagram

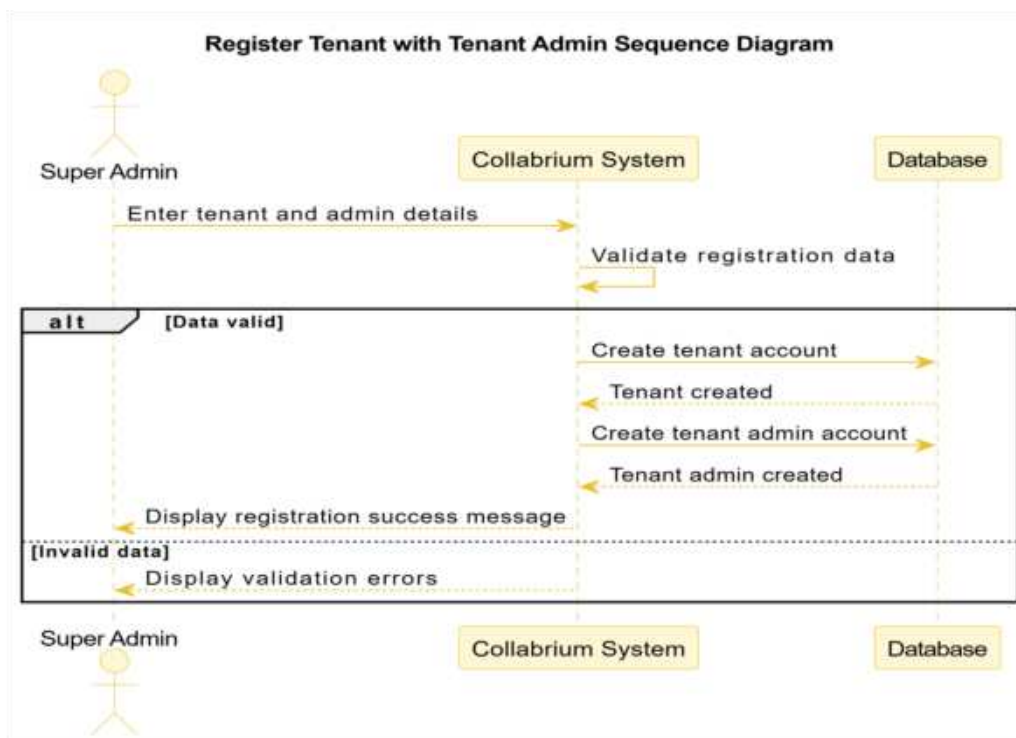


Figure 5.8.3 10 Register Tenant with Tenant Admin System Sequence Diagram

5.8.4 User Management Subsystem System Sequence Diagram

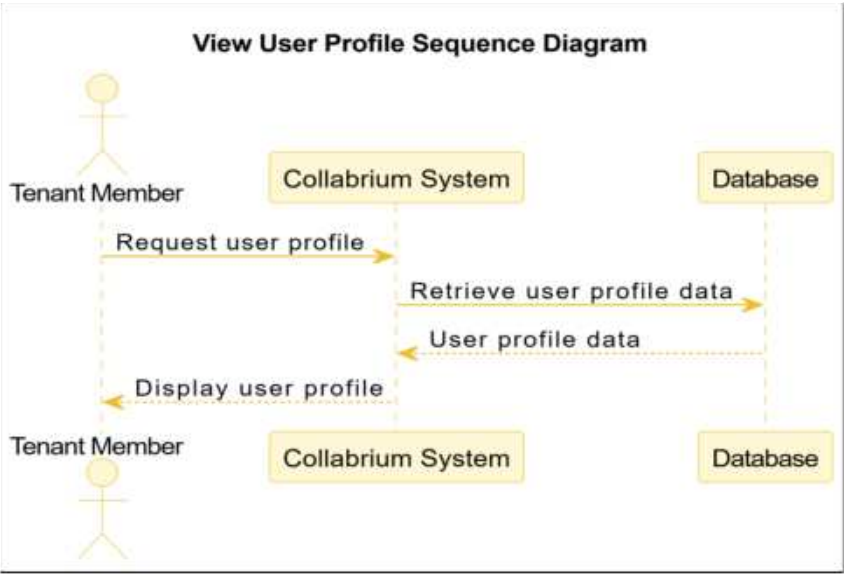


Figure 5.8.4 1 View User Profile System Sequence Diagram

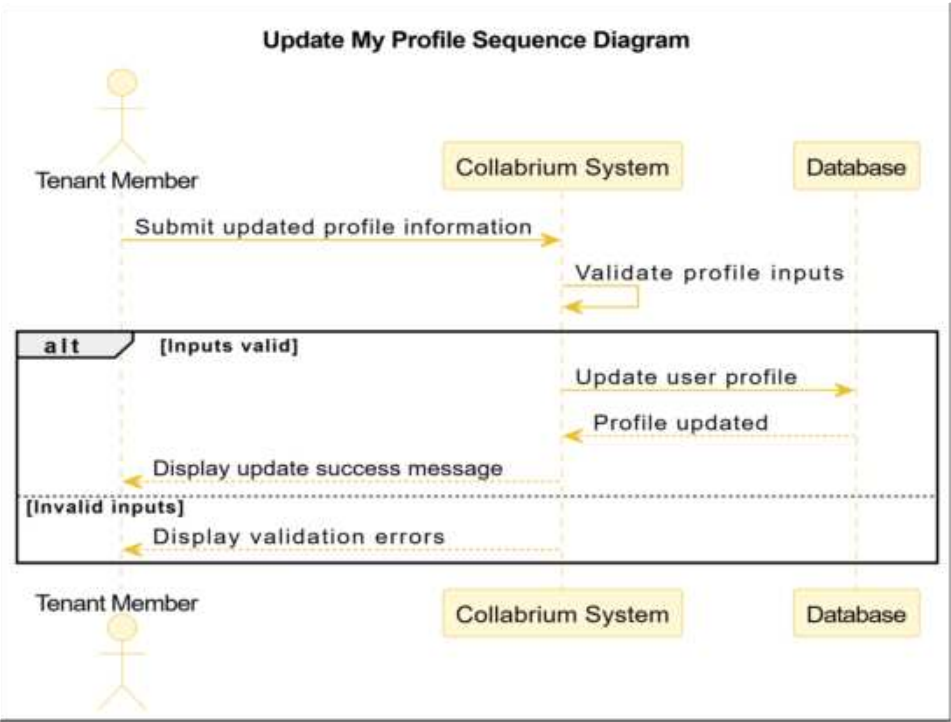


Figure 5.8.4 2 Update My Profile System Sequence Diagram

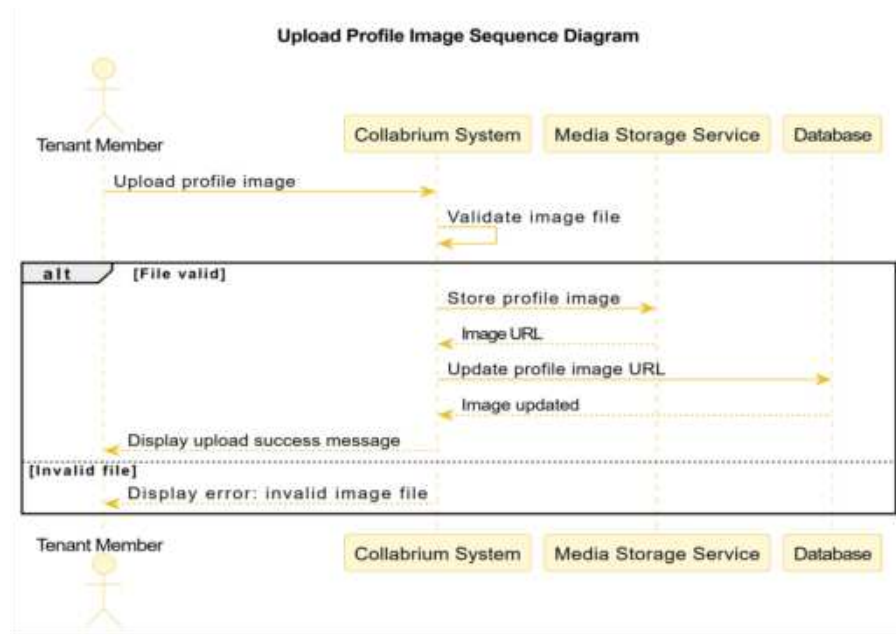


Figure 5.8.4 3 Upload Profile Image System Sequence Diagram

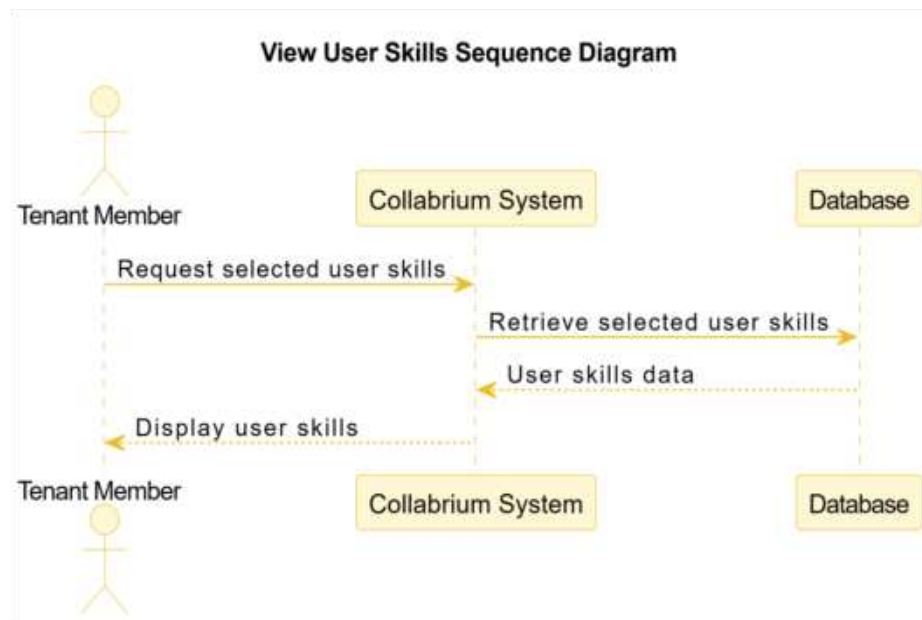


Figure 5.8.4 4 View User Skill System Sequence Diagram

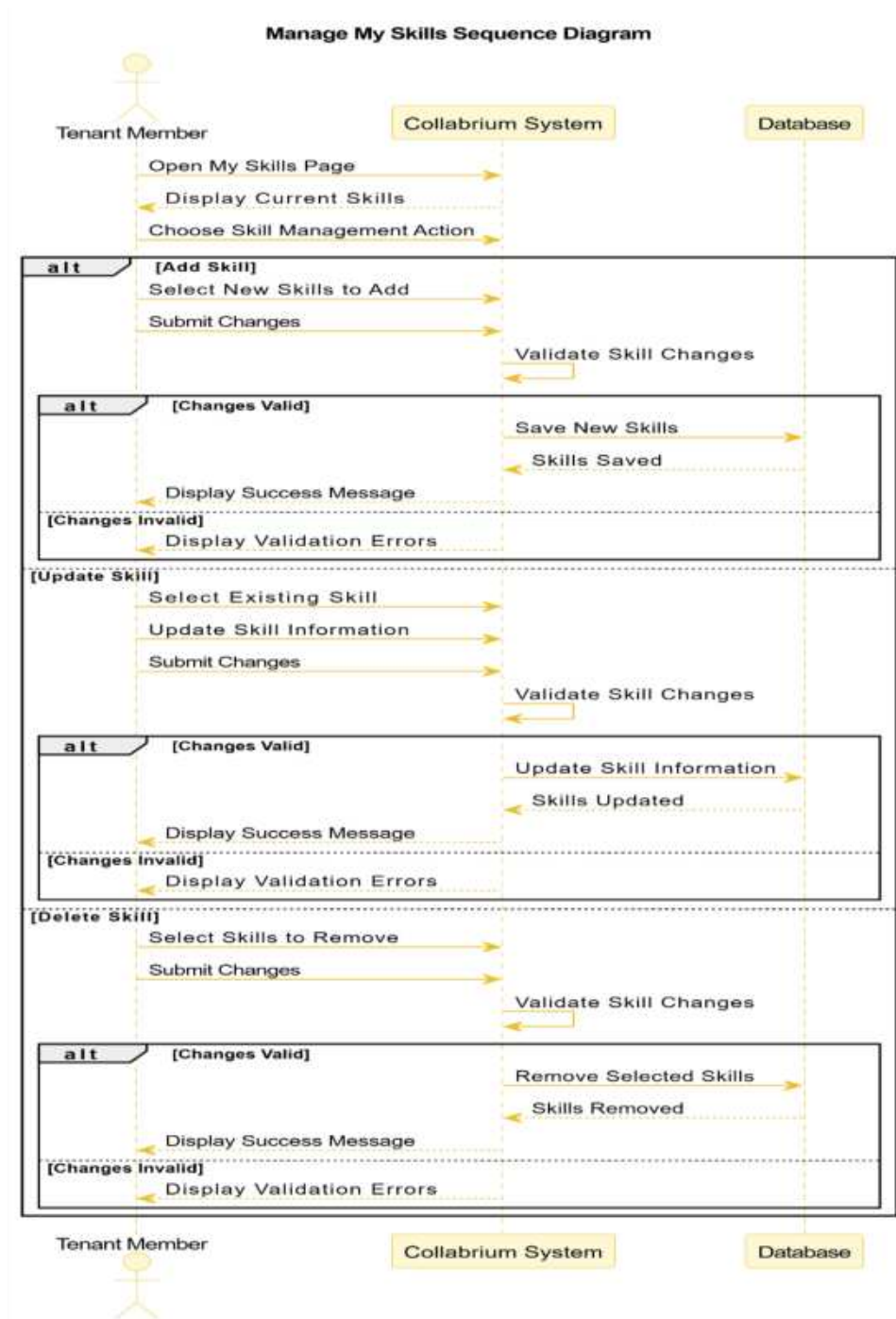


Figure 5.8.4 5 Manage My Skill System Sequence Diagram

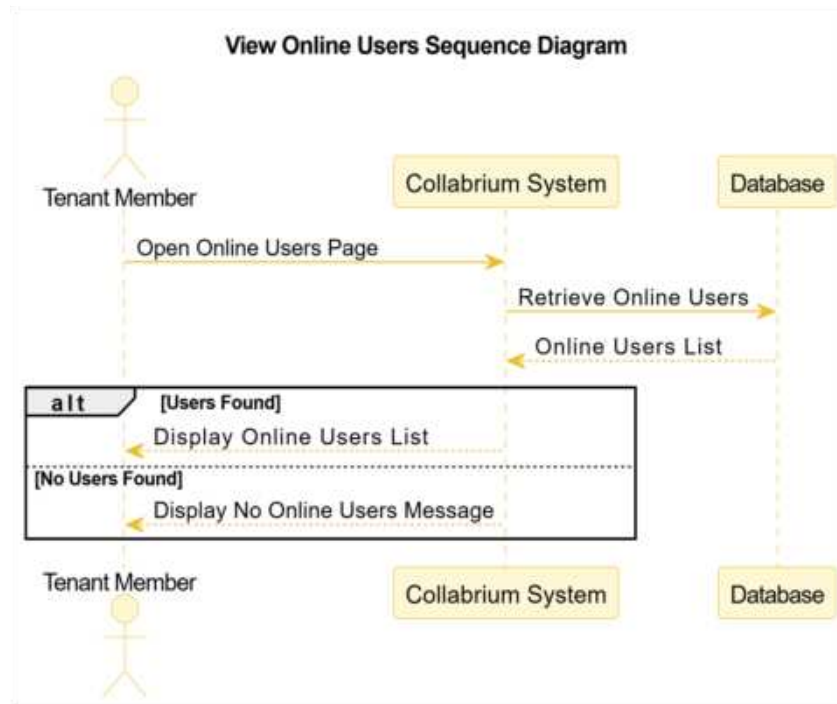


Figure 5.8.4 6 View Online Users System Sequence Diagram

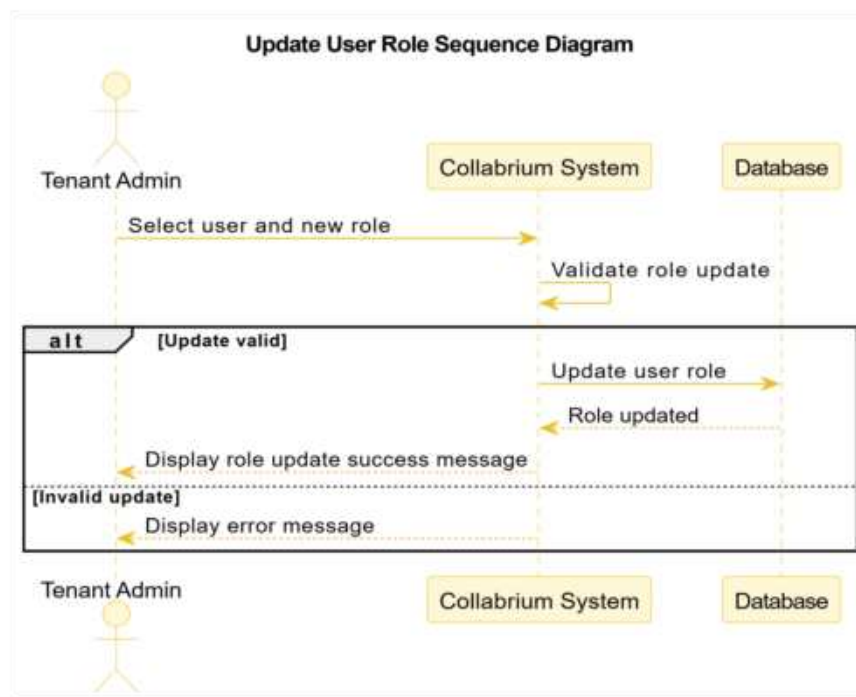


Figure 5.8.4 7 Update User Role System Sequence Diagram

5.8.5 Team Management Subsystem System Sequence Diagram

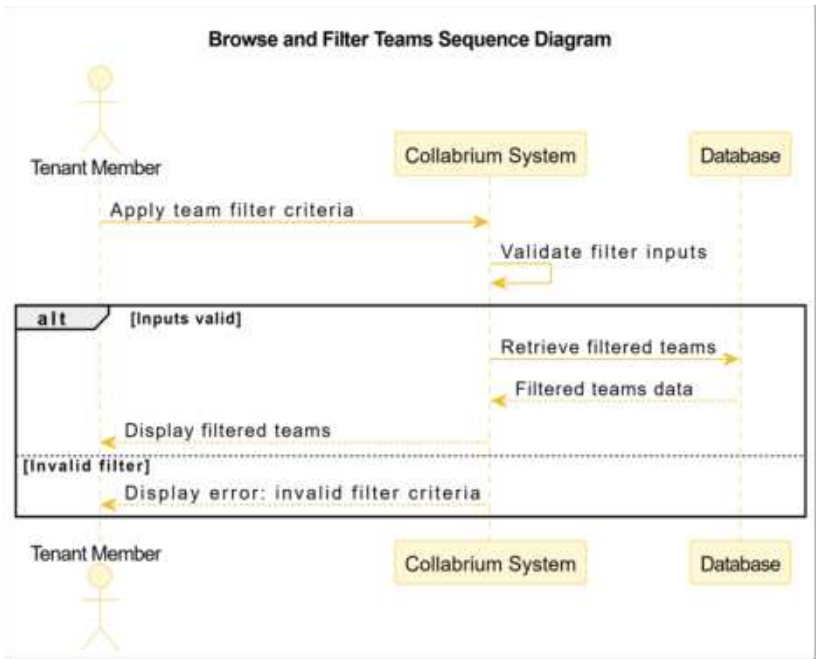


Figure 5.8.5 1 Browse and Filter Teams System Sequence Diagram

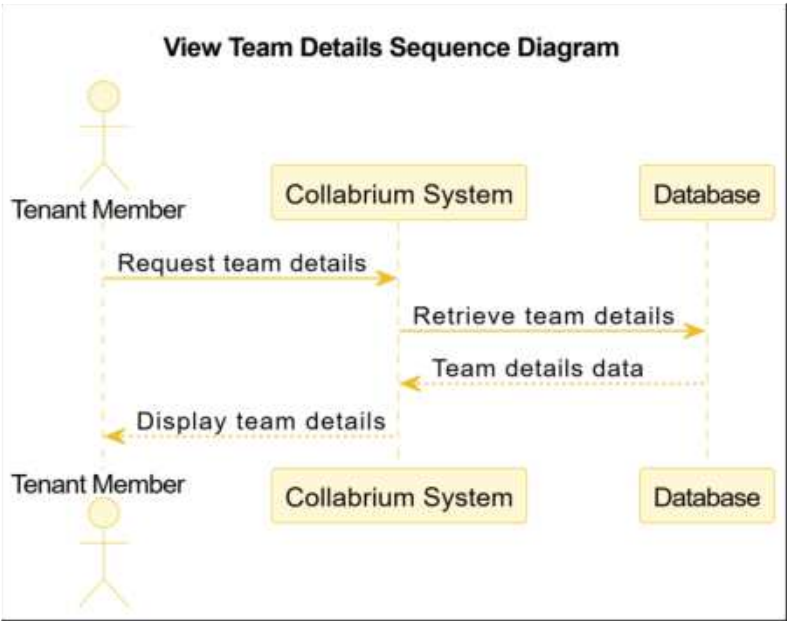


Figure 5.8.5 2 View Team Details System Sequence Diagram

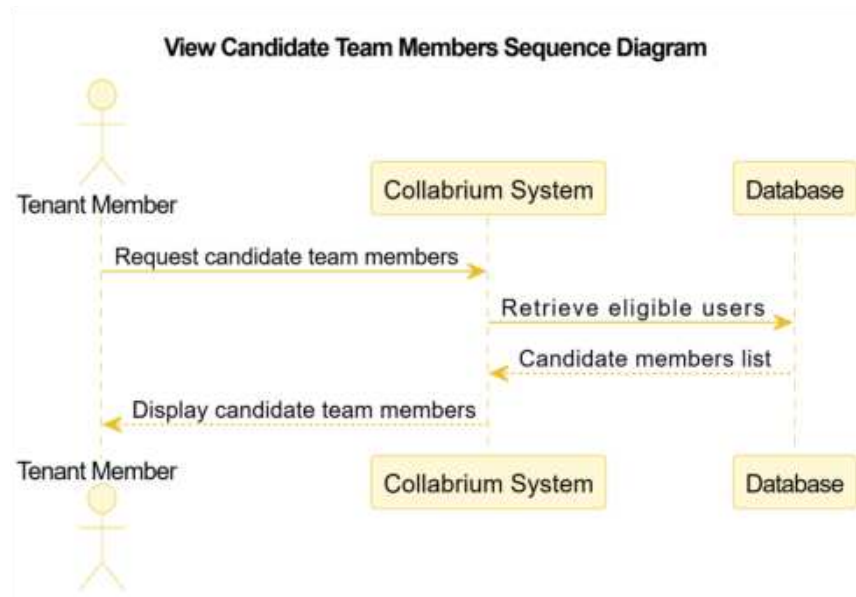


Figure 5.8.5 3 View Candidate Team Member System Sequence Diagram

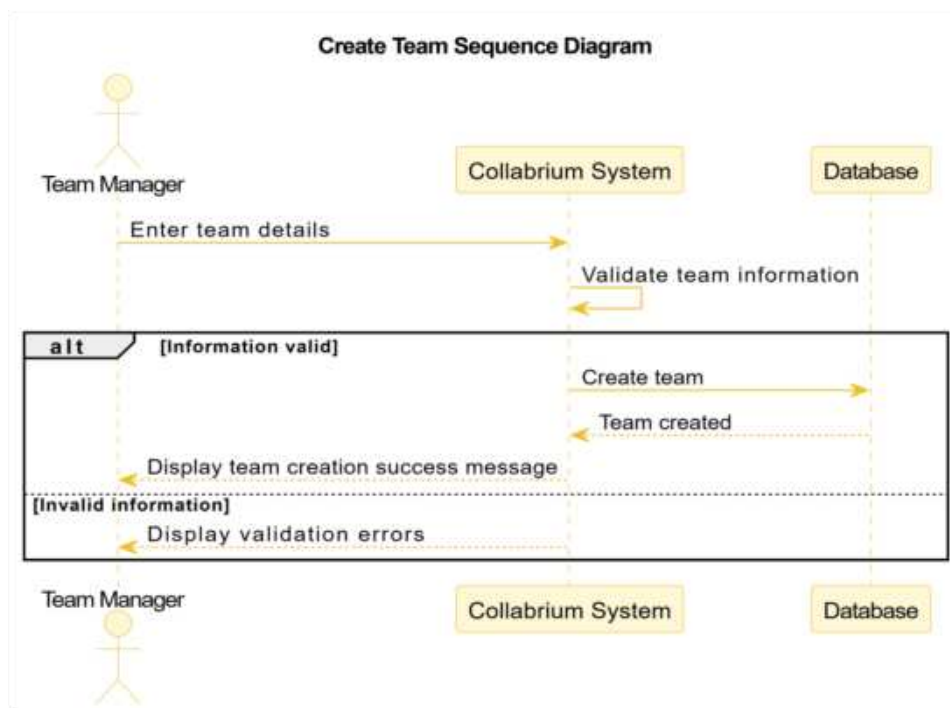


Figure 5.8.5 4 Create Team System Sequence Diagram

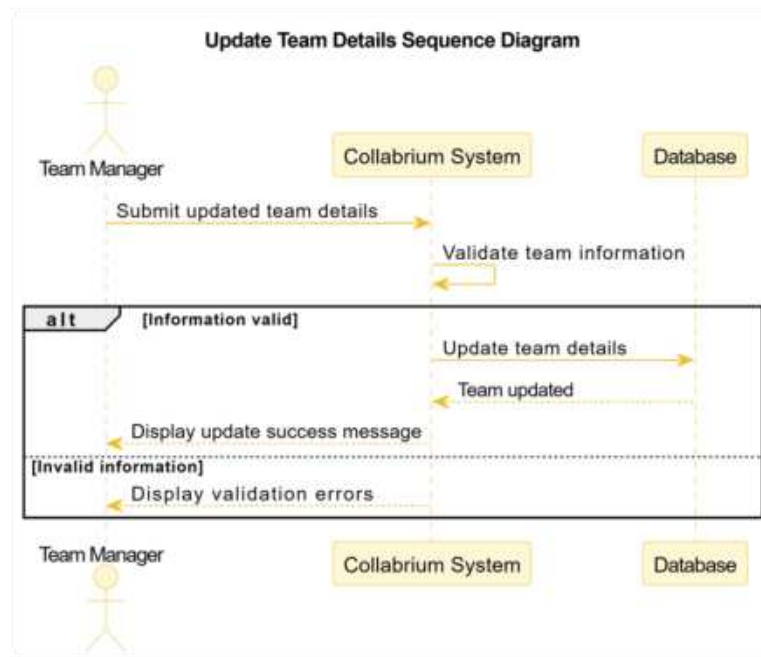


Figure 5.8.5 5 Update Team Details System Sequence Diagram

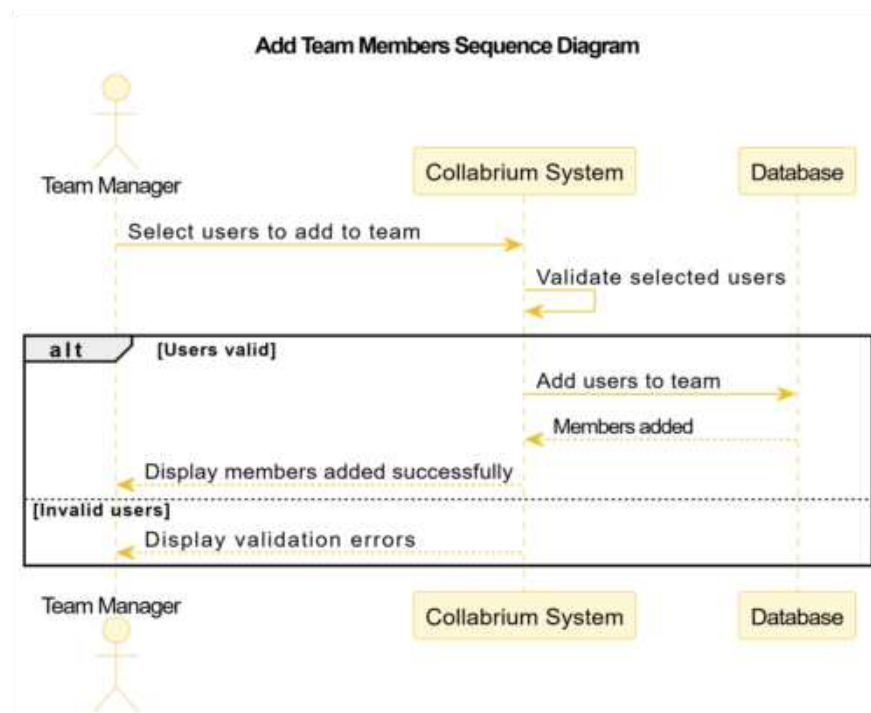


Figure 5.8.5 6 Add Team Members System Sequence Diagram

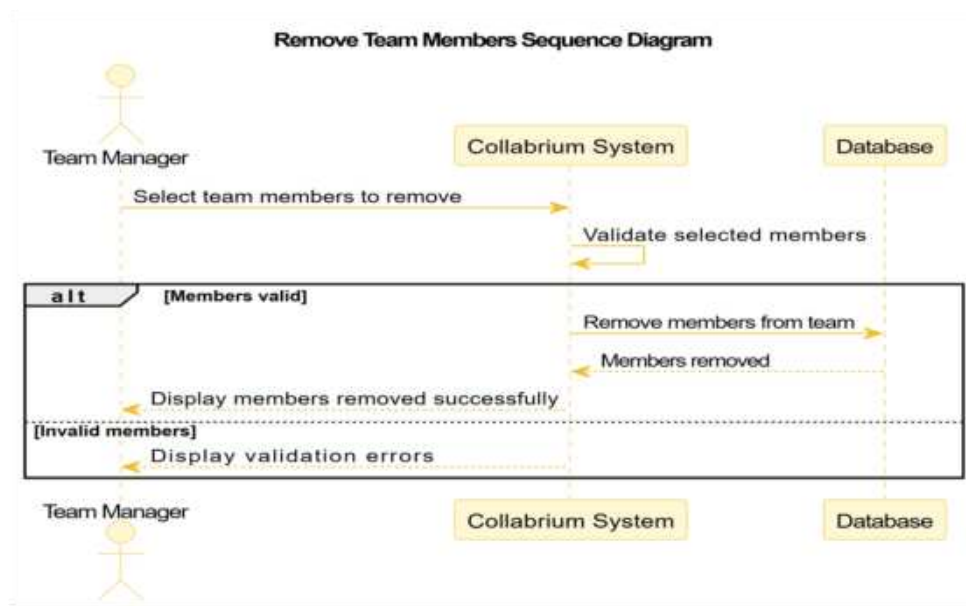


Figure 5.8.5 7 Remove Team Member System Sequence Diagram

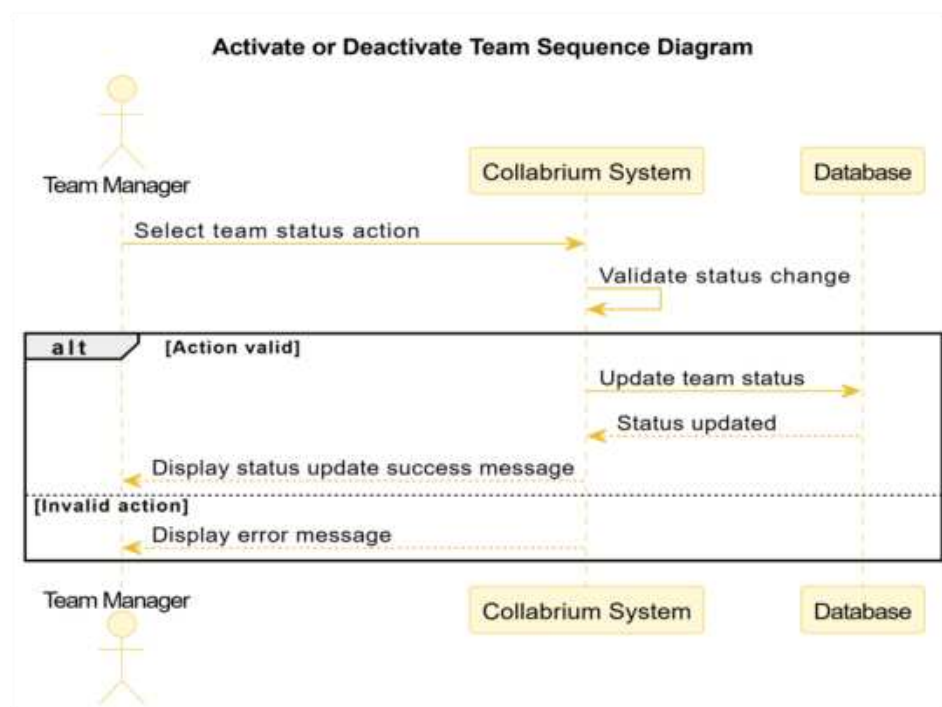


Figure 5.8.5 8 Activate or Deactivate Team System Sequence Diagram

## 5.8.6 Project Management Subsystem System Sequence Diagram

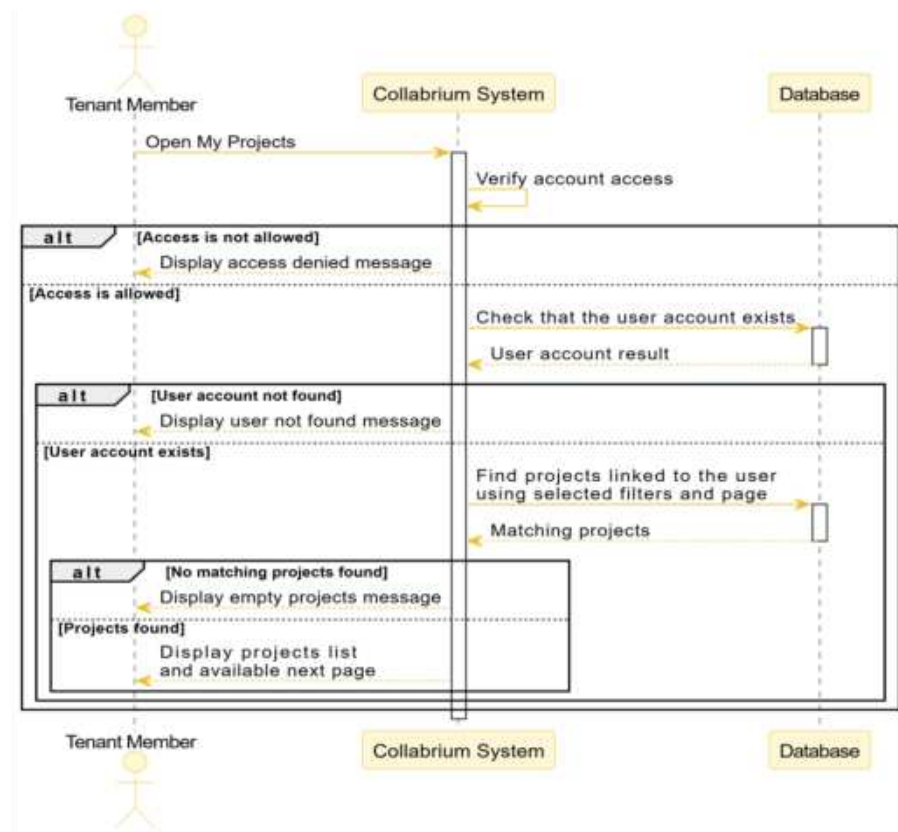


Figure 5.8.6 1 View My Projects System Sequence Diagram

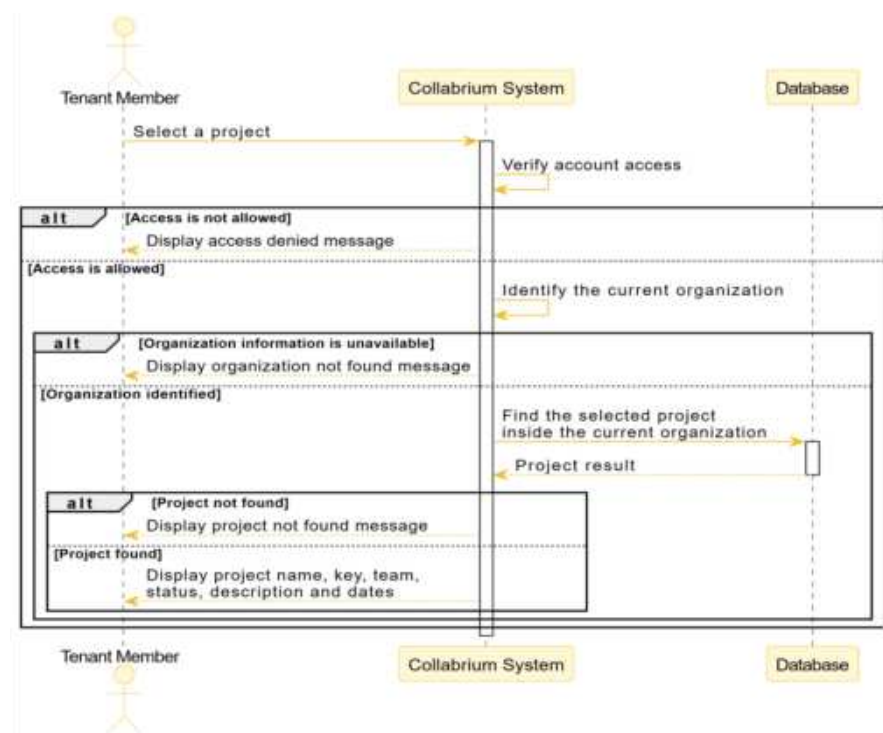


Figure 5.8.6 2 View Project Detail System Sequence Diagram

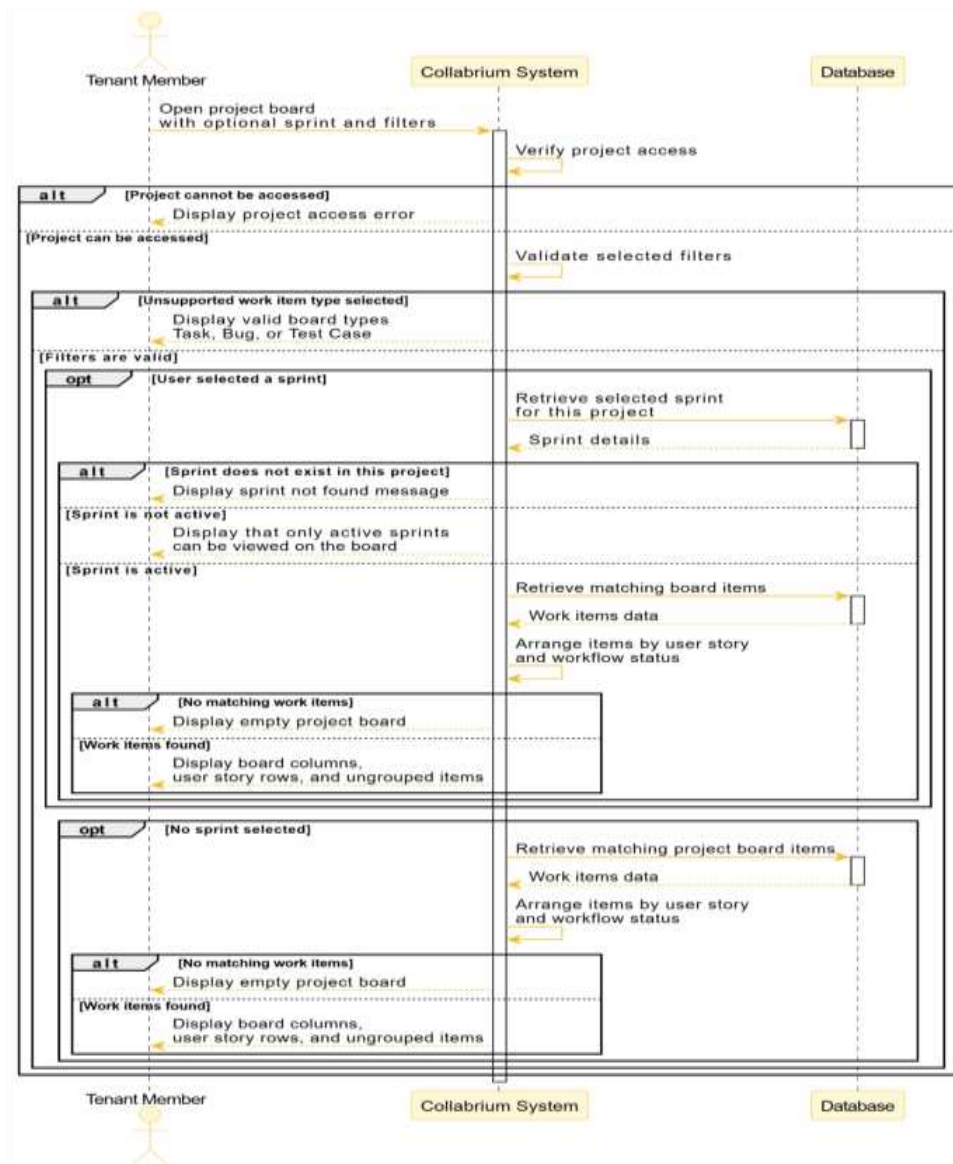


Figure 5.8.6 3 View Project Board System Sequence Diagram

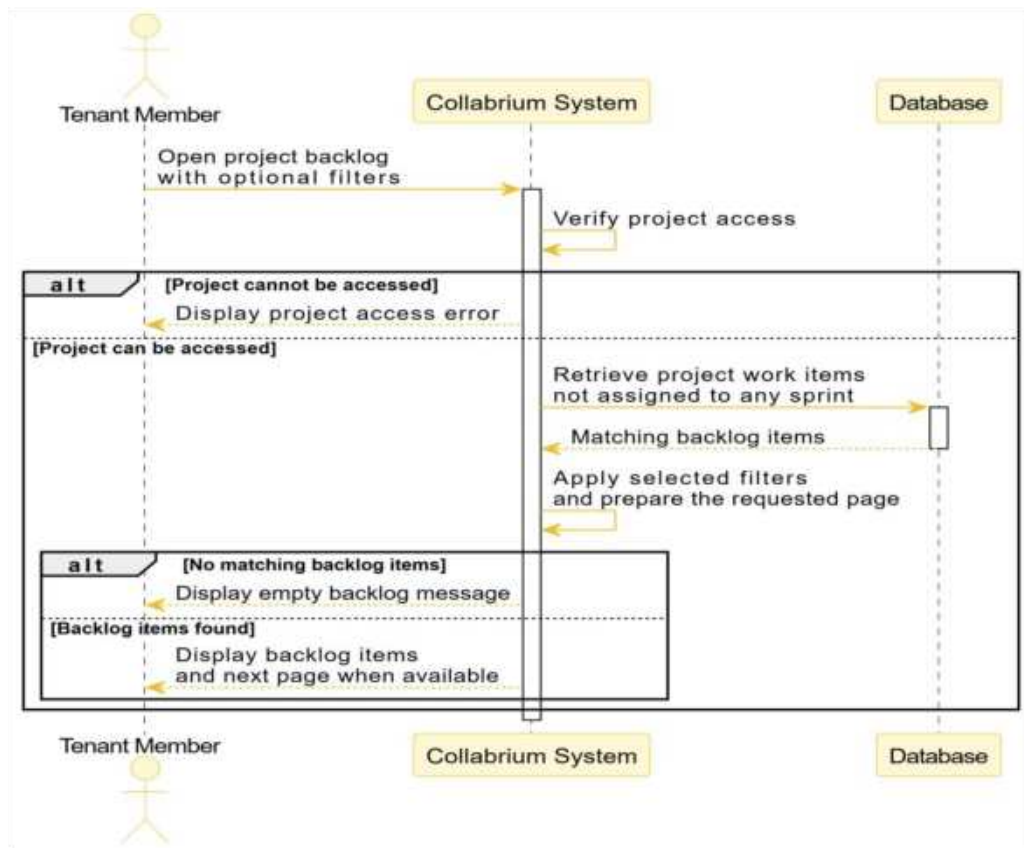


Figure 5.8.6 4 View Project Backlog System Sequence Diagram

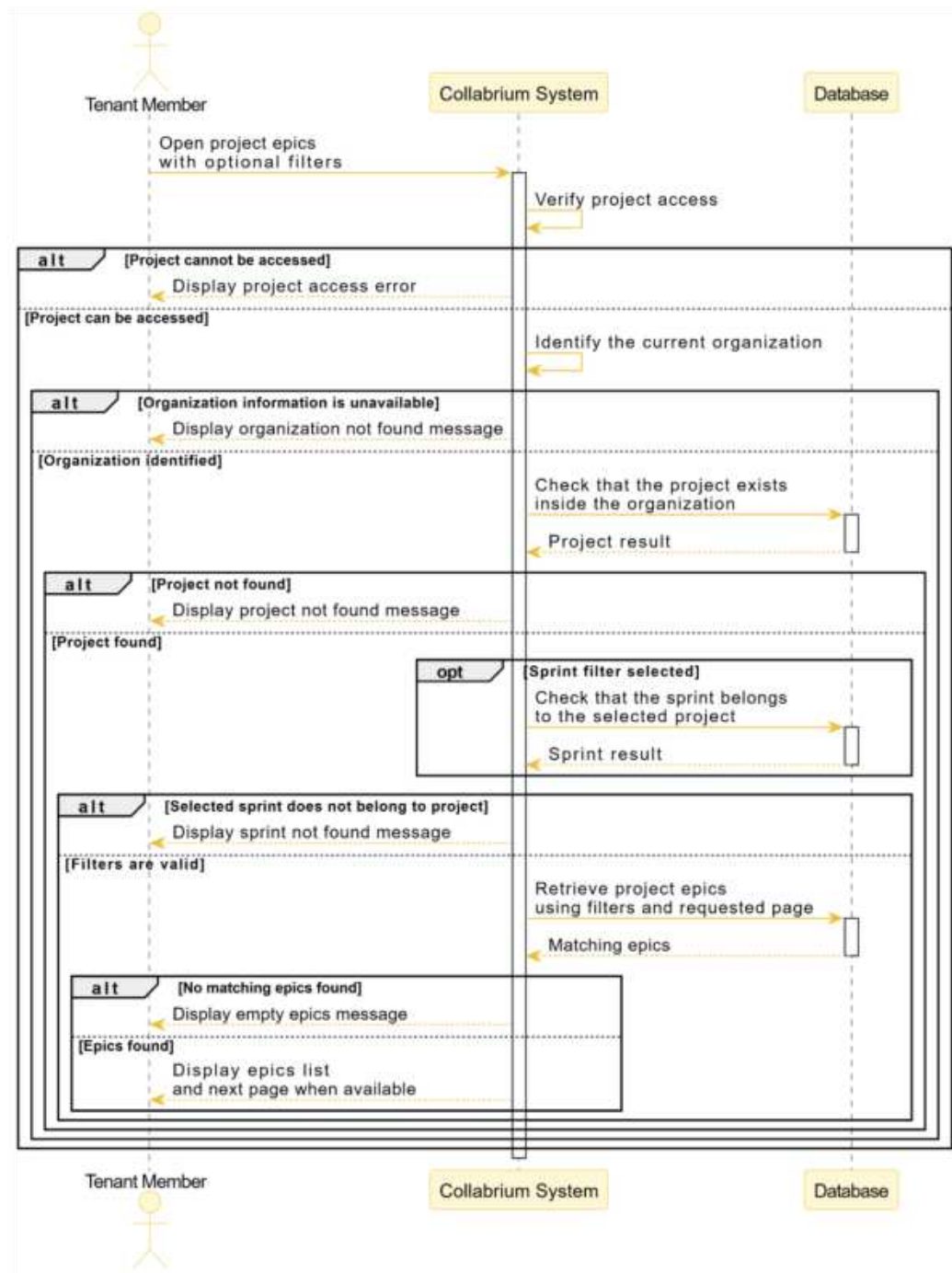


Figure 5.8.6 5 View Project Epic System Sequence Diagram

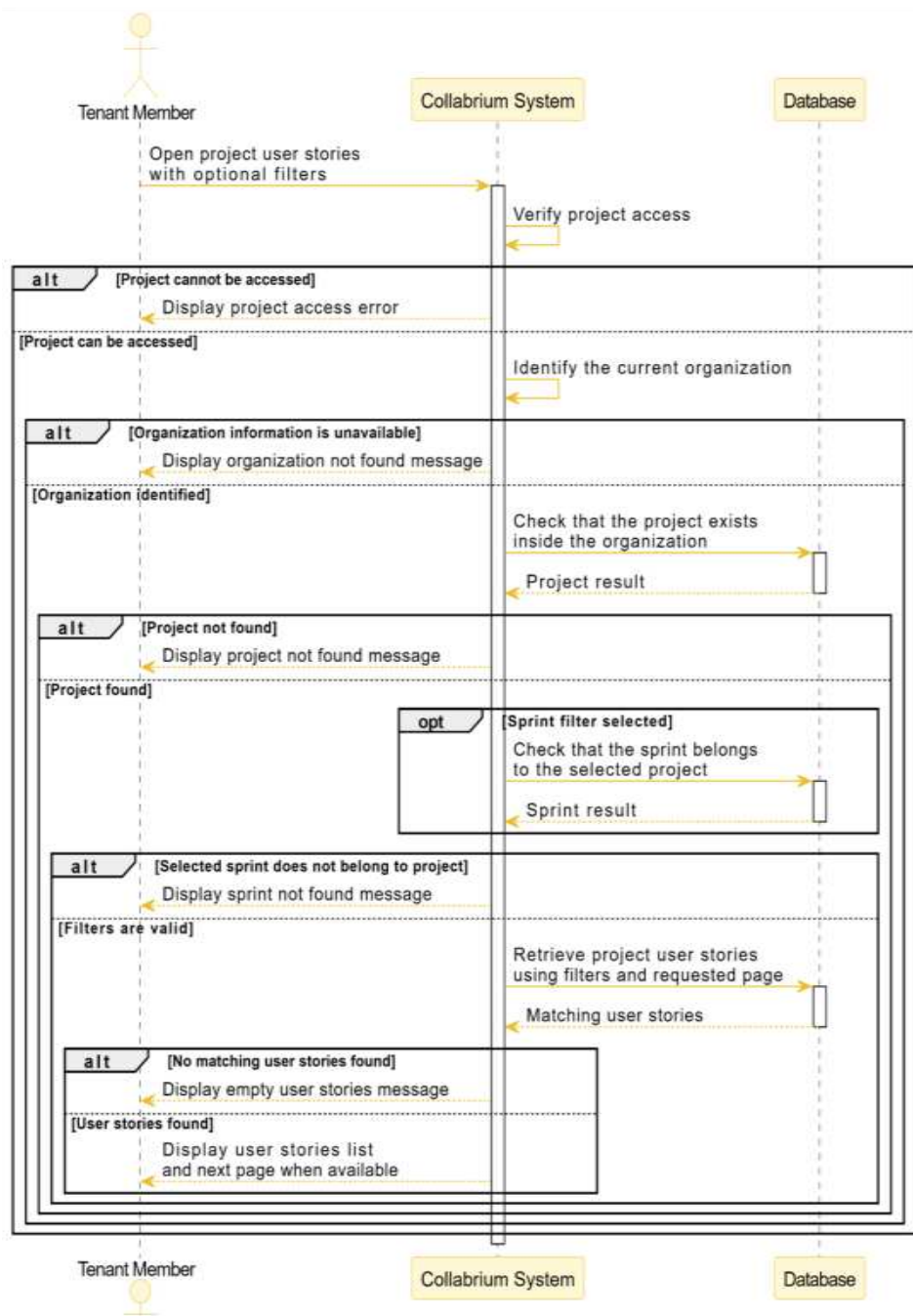


Figure 5.8.6 6 View Project User Stories System Sequence Diagram

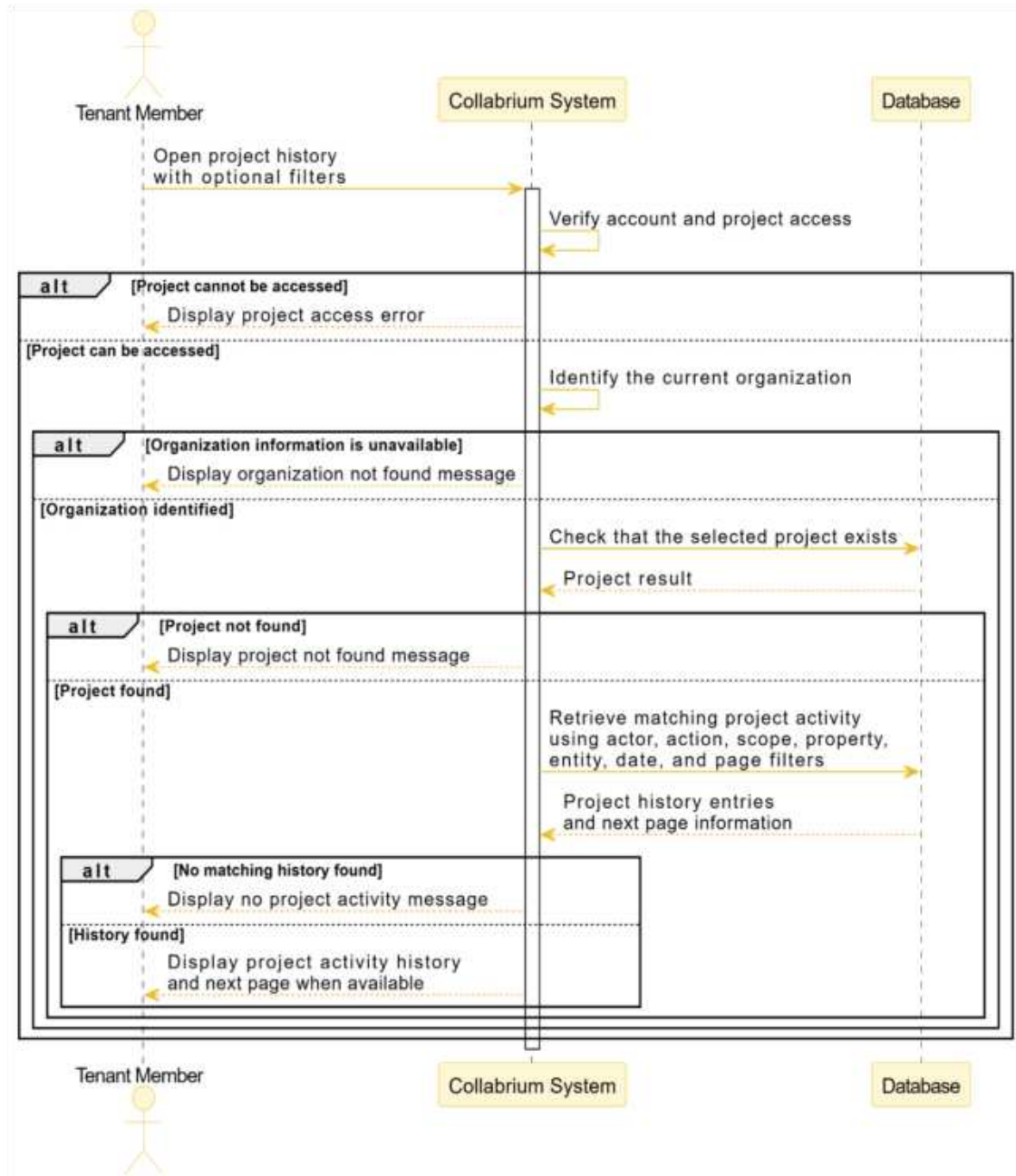


Figure 5.8.6 7 View Project History System Sequence Diagram

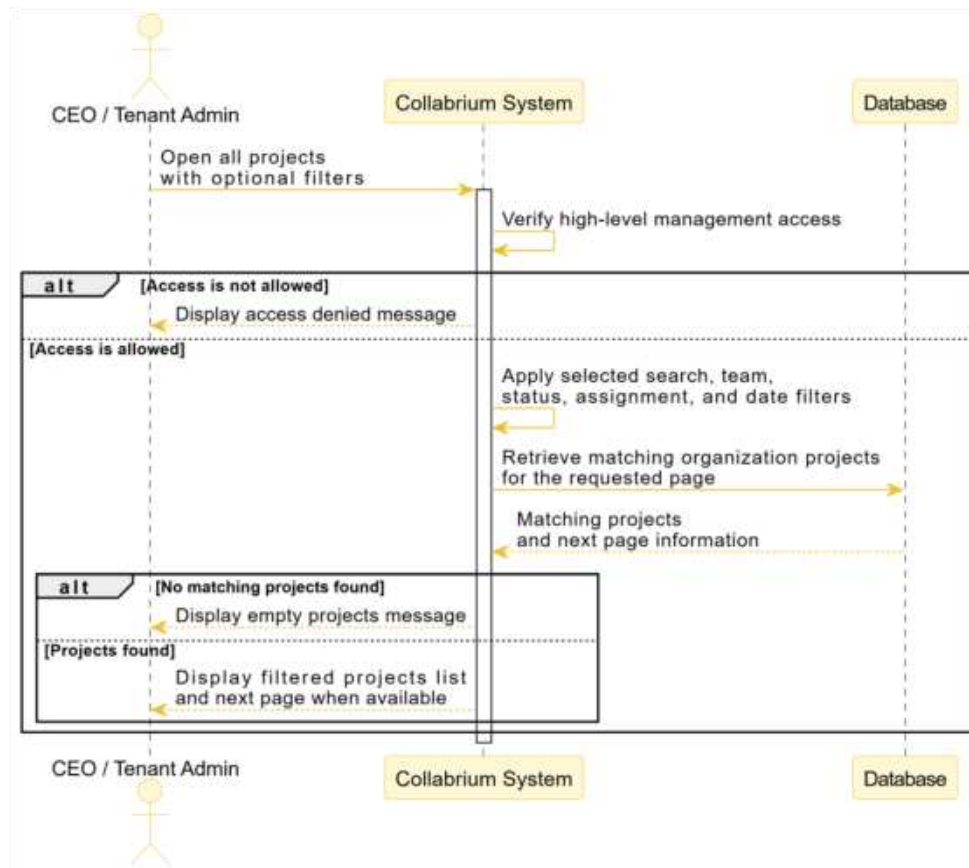


Figure 5.8.6 8 Browse All Projects History System Sequence Diagram

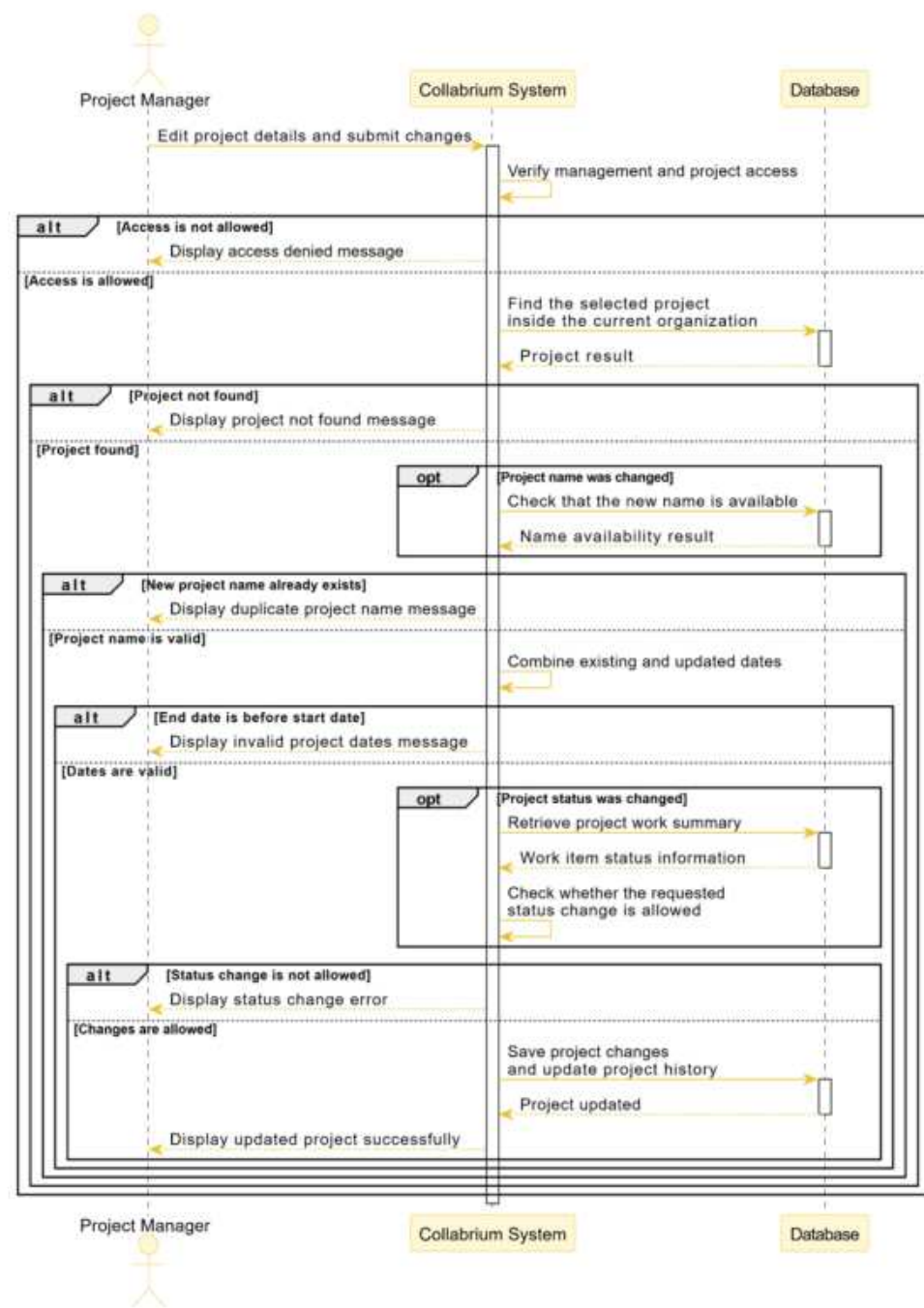


Figure 5.8.6 9 Edit Project Details System Sequence Diagram

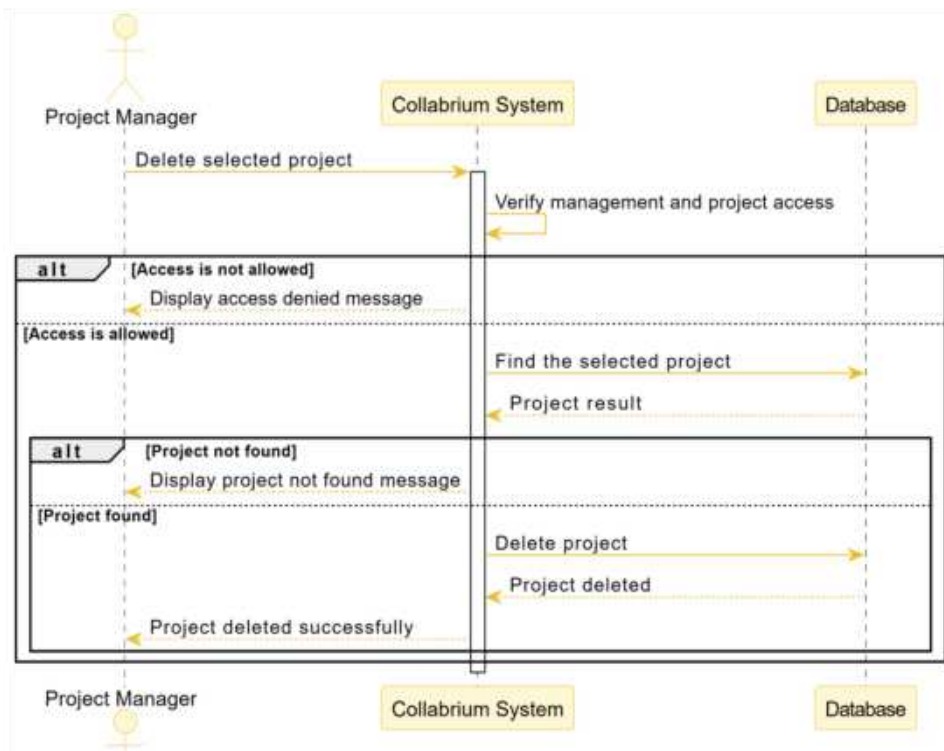


Figure 5.8.6 10 Delete Selected Project System Sequence Diagram

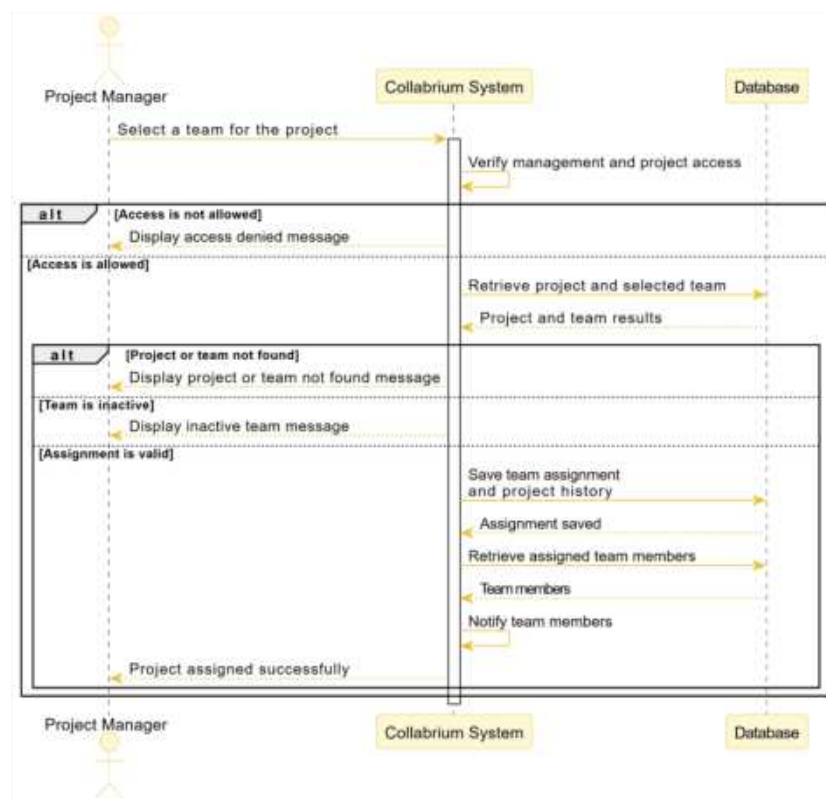


Figure 5.8.6 11 Select A Team for A Project System Sequence Diagram

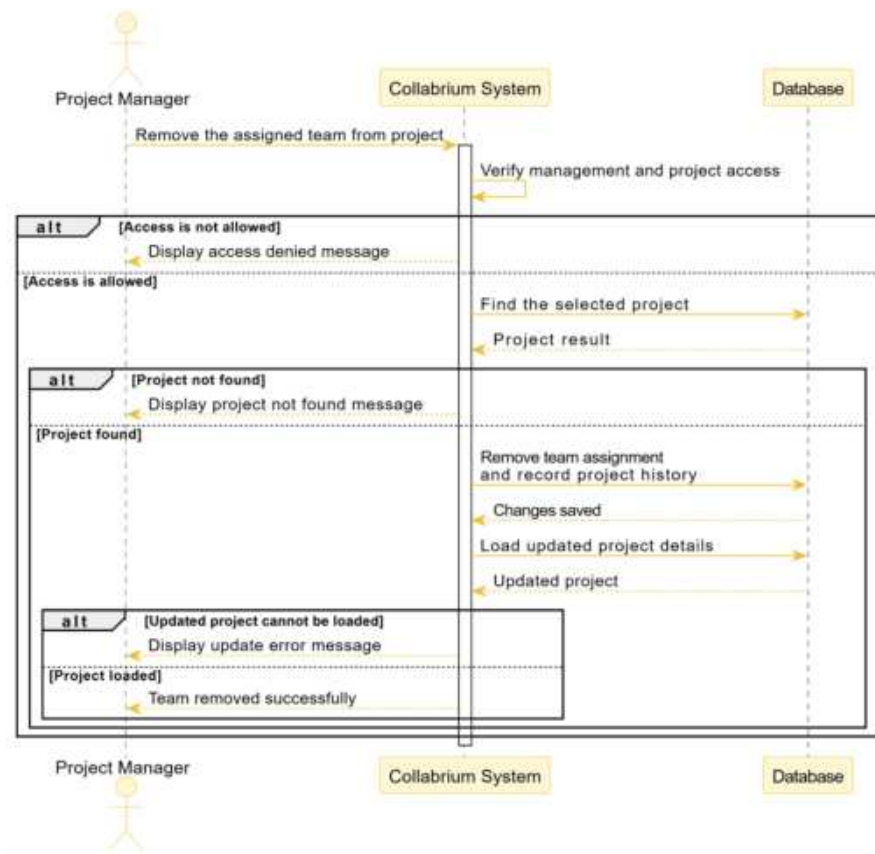


Figure 5.8.6 12 Remove the Assigned Team from A Project System Sequence Diagram

## 5.8.7 Work Item Core Management Subsystem System Sequence Diagram

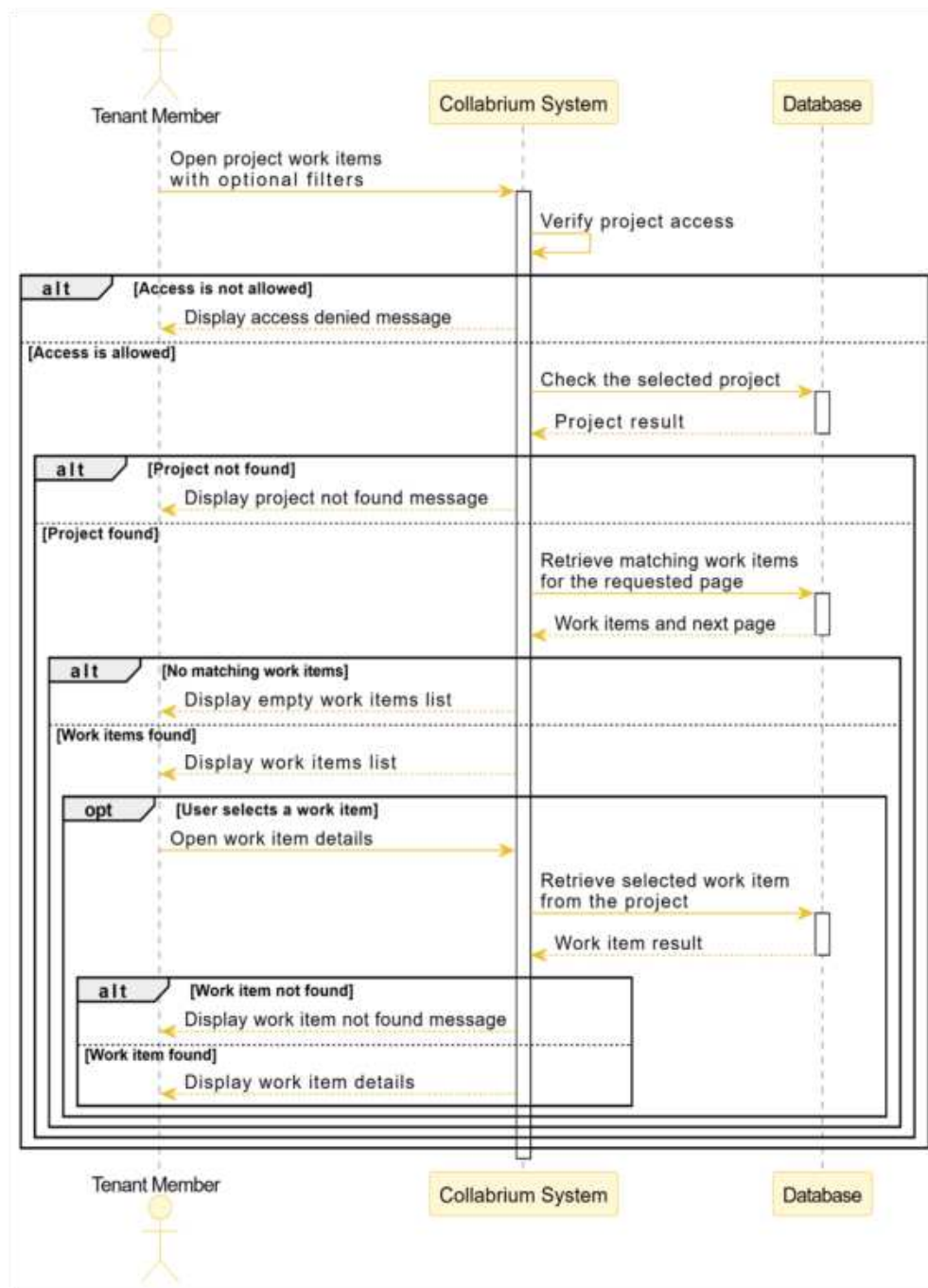


Figure 5.8.7 1 Browse and Filter Work Items System Sequence Diagram

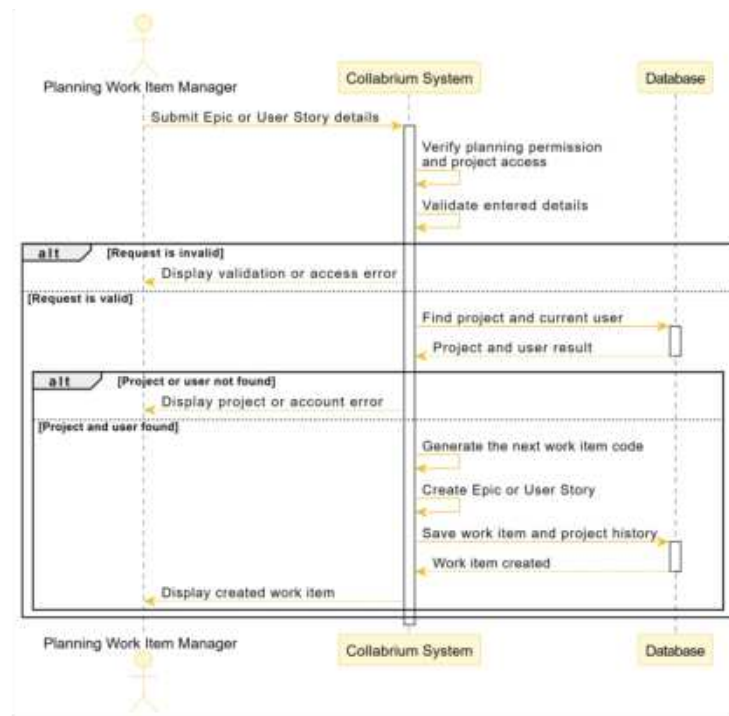


Figure 5.8.7 2 Create Planning Work Item System Sequence Diagram

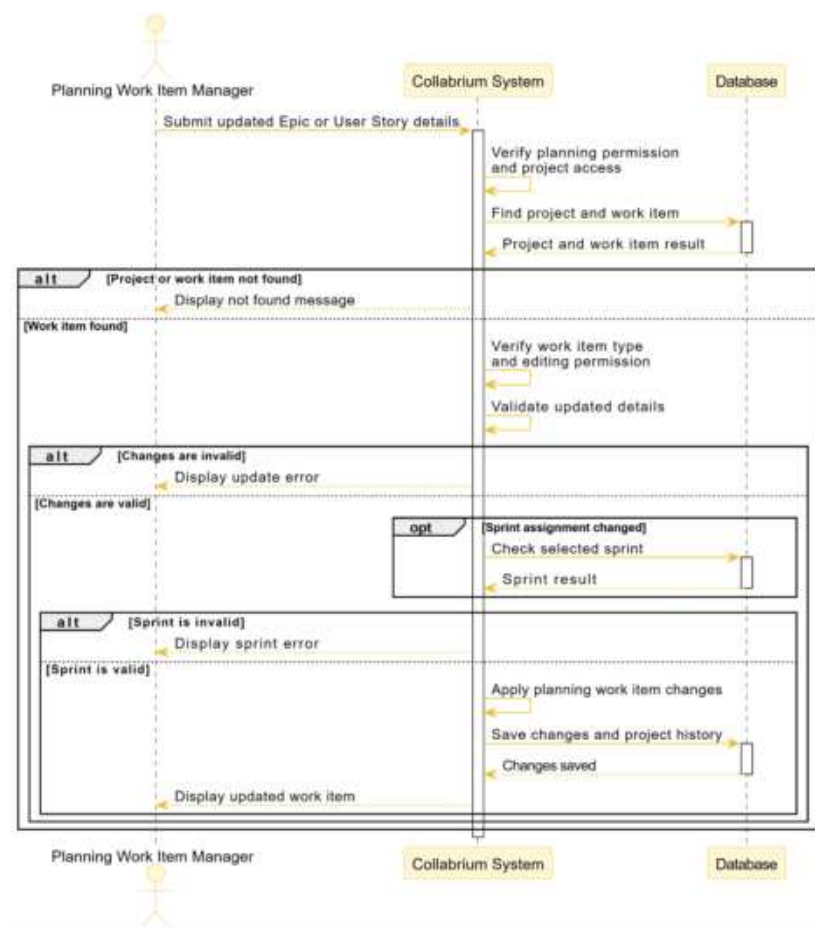


Figure 5.8.7 3 Update Planning Work Item System Sequence Diagram

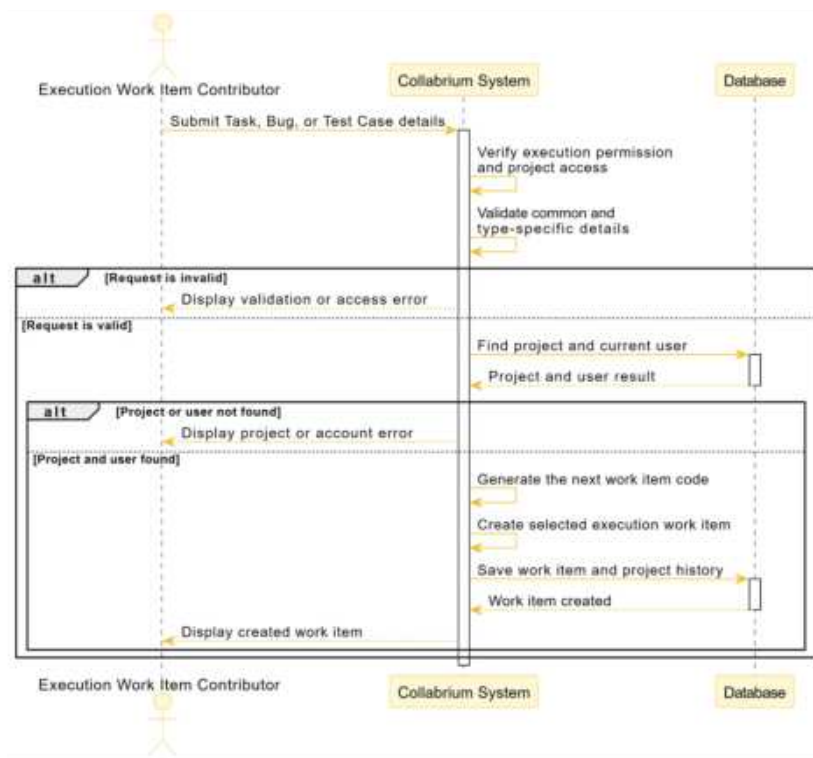


Figure 5.8.7 4 Create Execution Work Item System Sequence Diagram

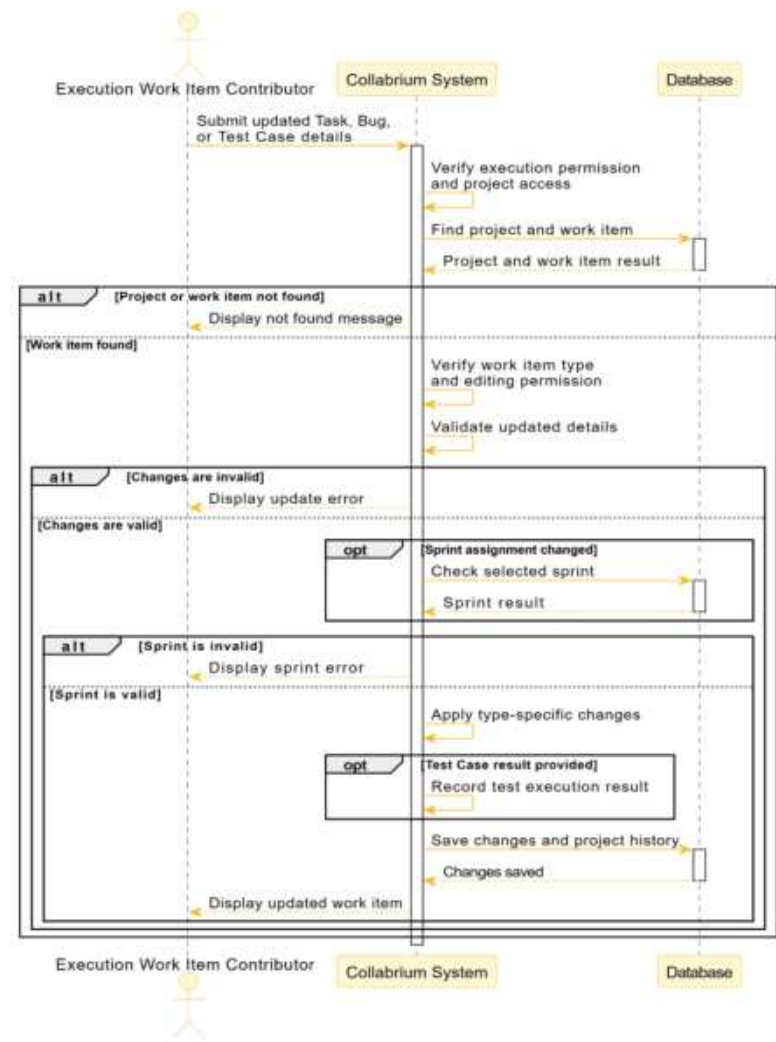


Figure 5.8.7.5 Update Execution Work Item System Sequence Diagram

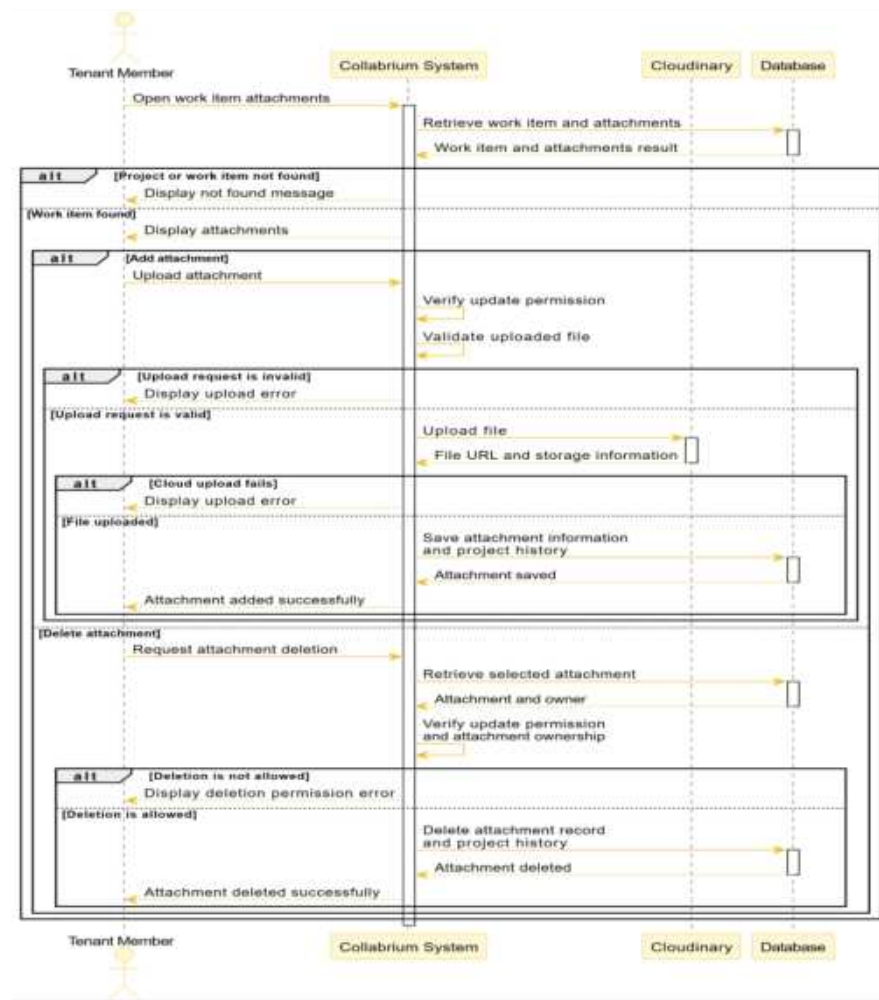


Figure 5.8.7 6 Manage Work item Attachments System Sequence Diagram

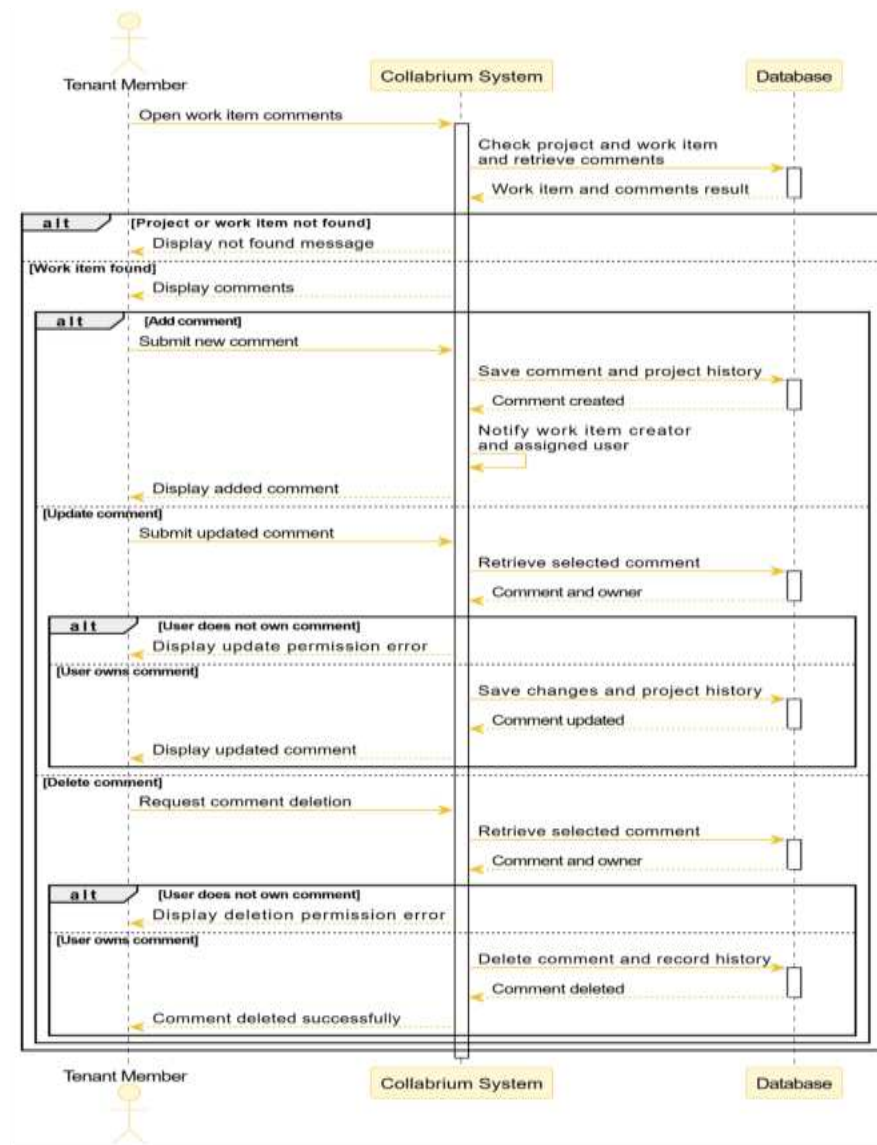


Figure 5.8.77 Manage Work item Comment System Sequence Diagram

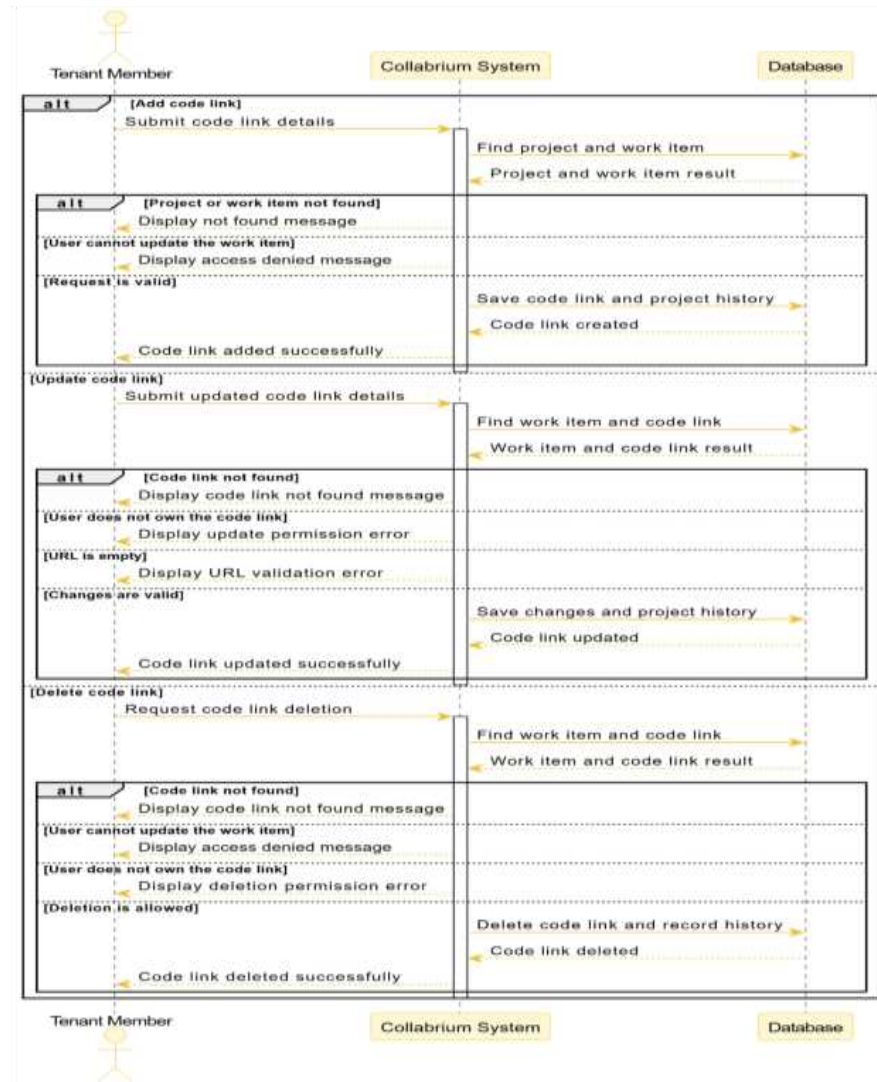


Figure 5.8.7 8 Manage Work item Code Links System Sequence Diagram

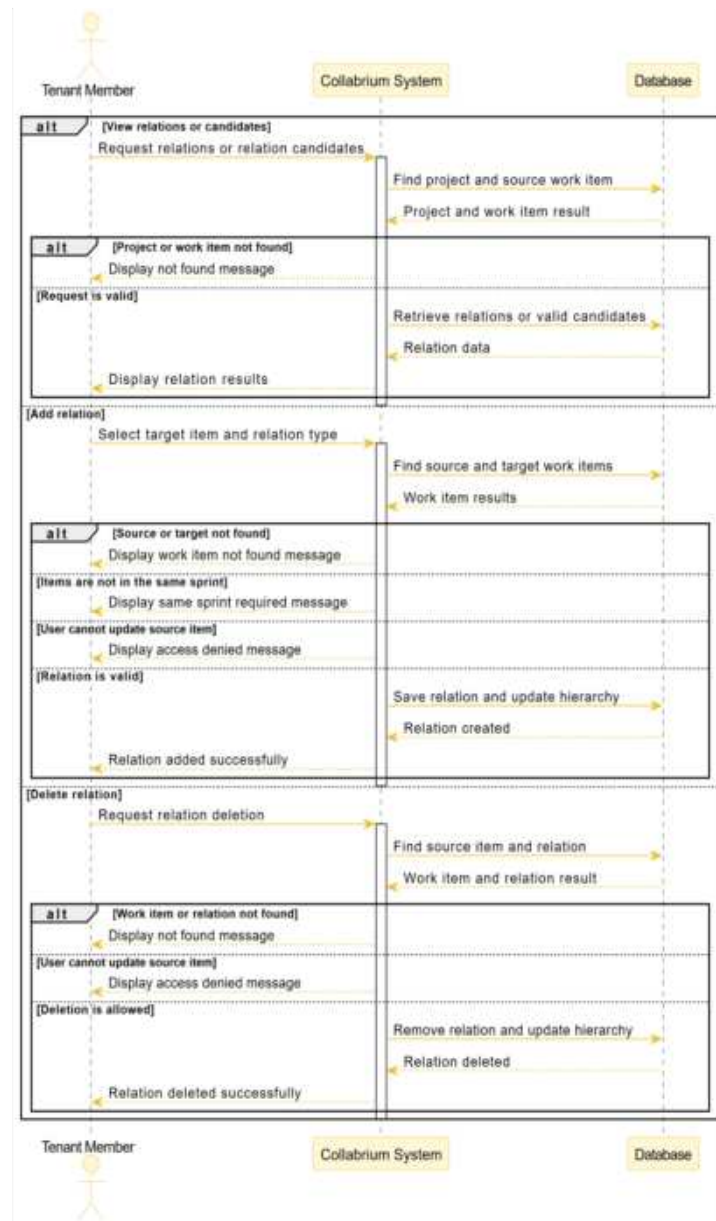


Figure 5.8.7 9 Manage Work item Relations System Sequence Diagram

## 5.8.8 Sprint Management Subsystem System Sequence Diagram

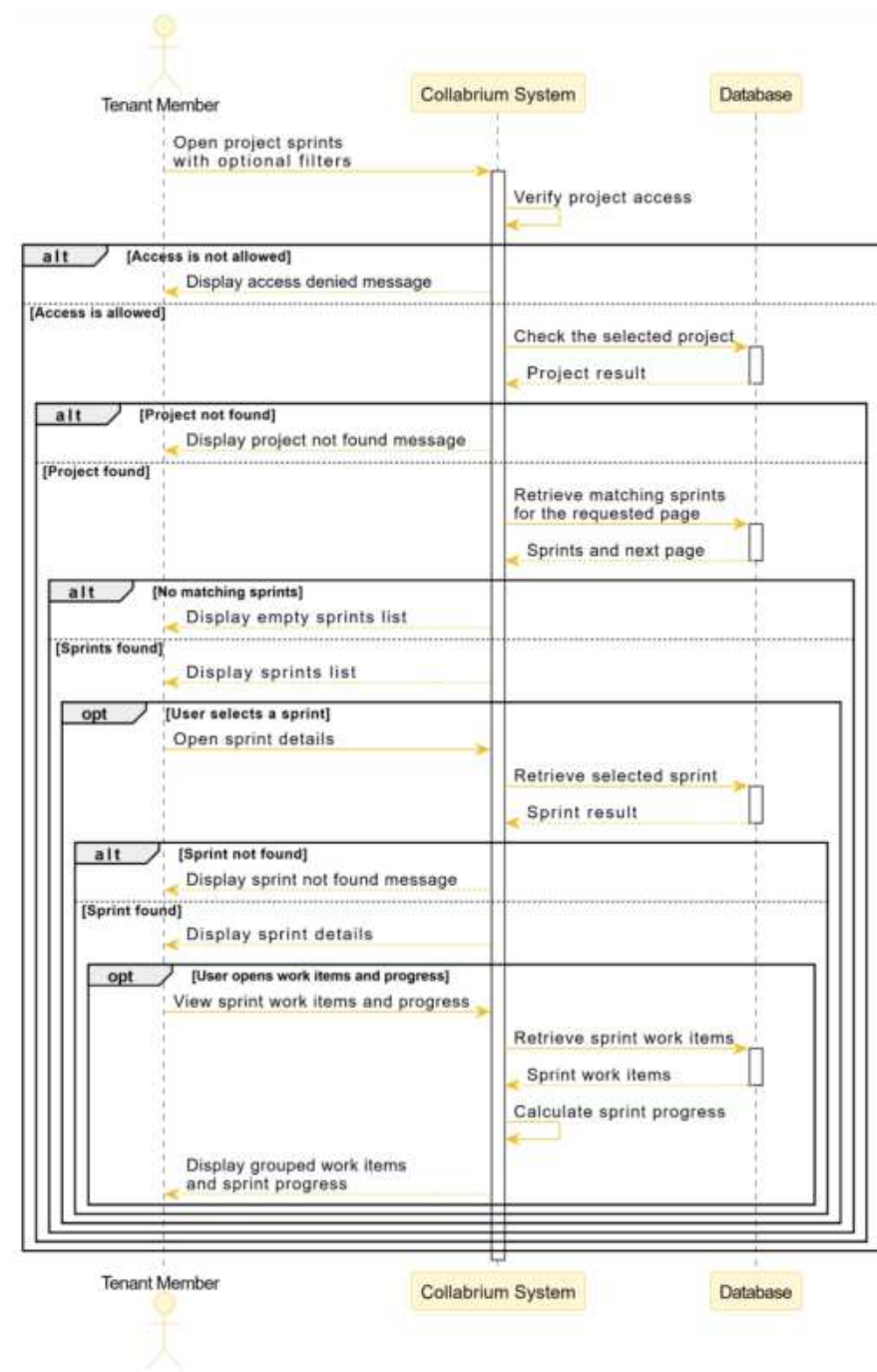


Figure 5.8.8 1 Browse and Filter Sprints System Sequence Diagram

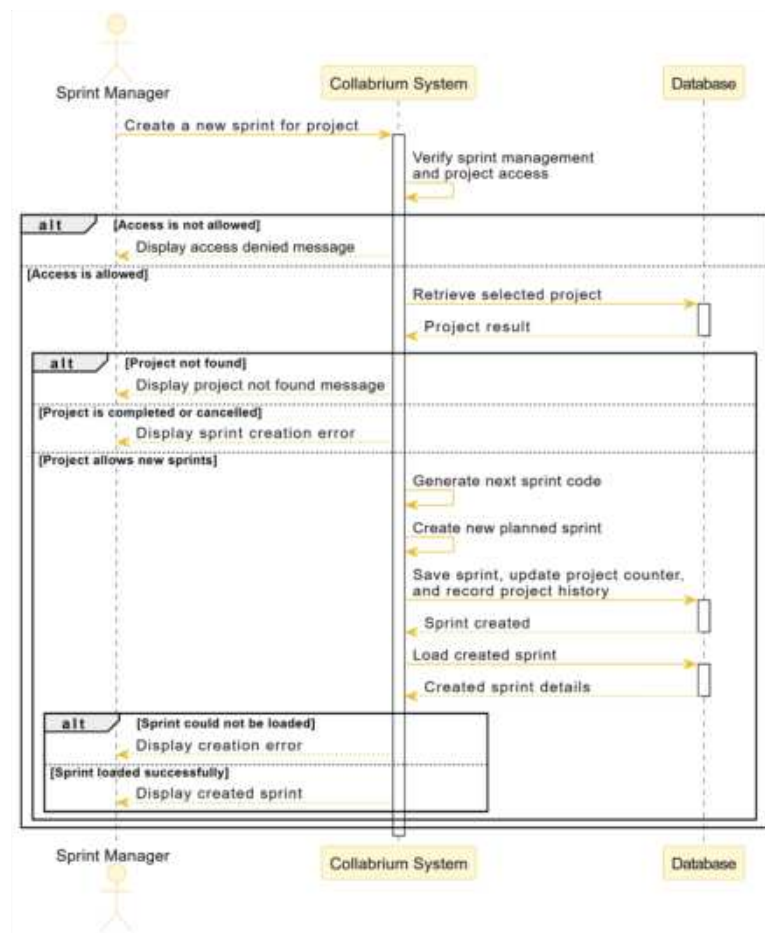


Figure 5.8.8 2 Create Sprint System Sequence Diagram

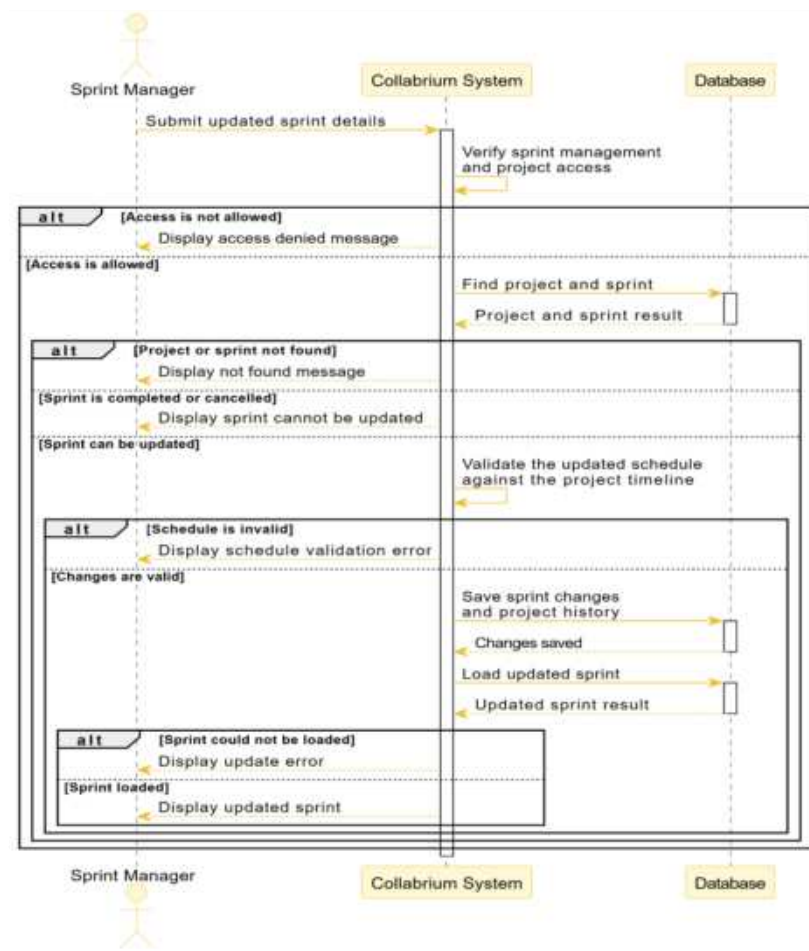


Figure 5.8.8 3 Update Sprint System Sequence Diagram

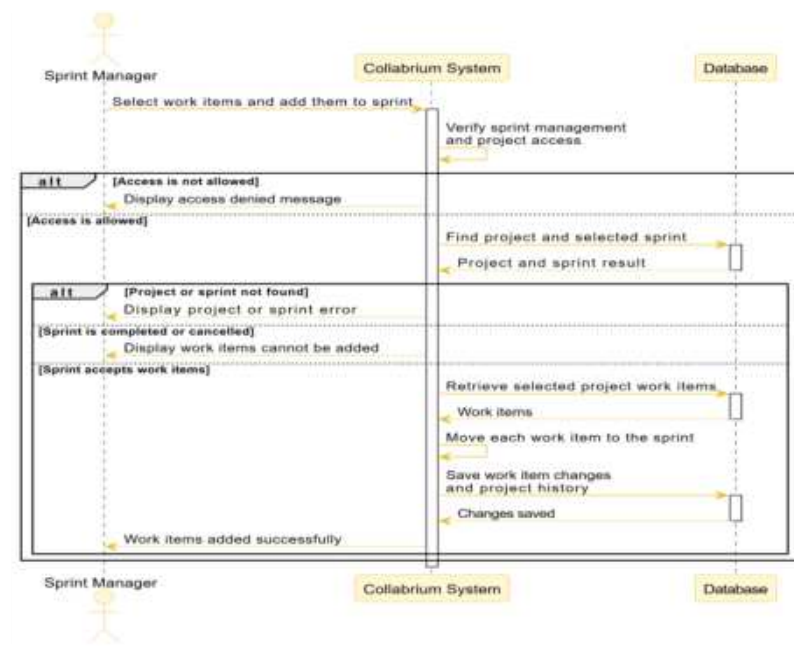


Figure 5.8.8 4 Add Work item to sprint System Sequence Diagram

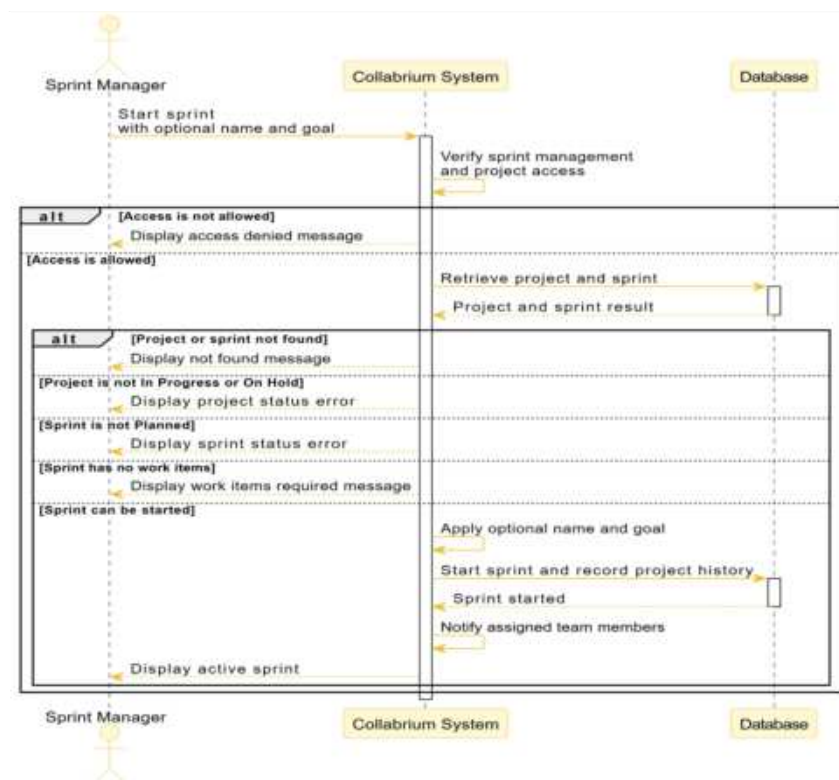


Figure 5.8.8 5 Start Sprint System Sequence Diagram

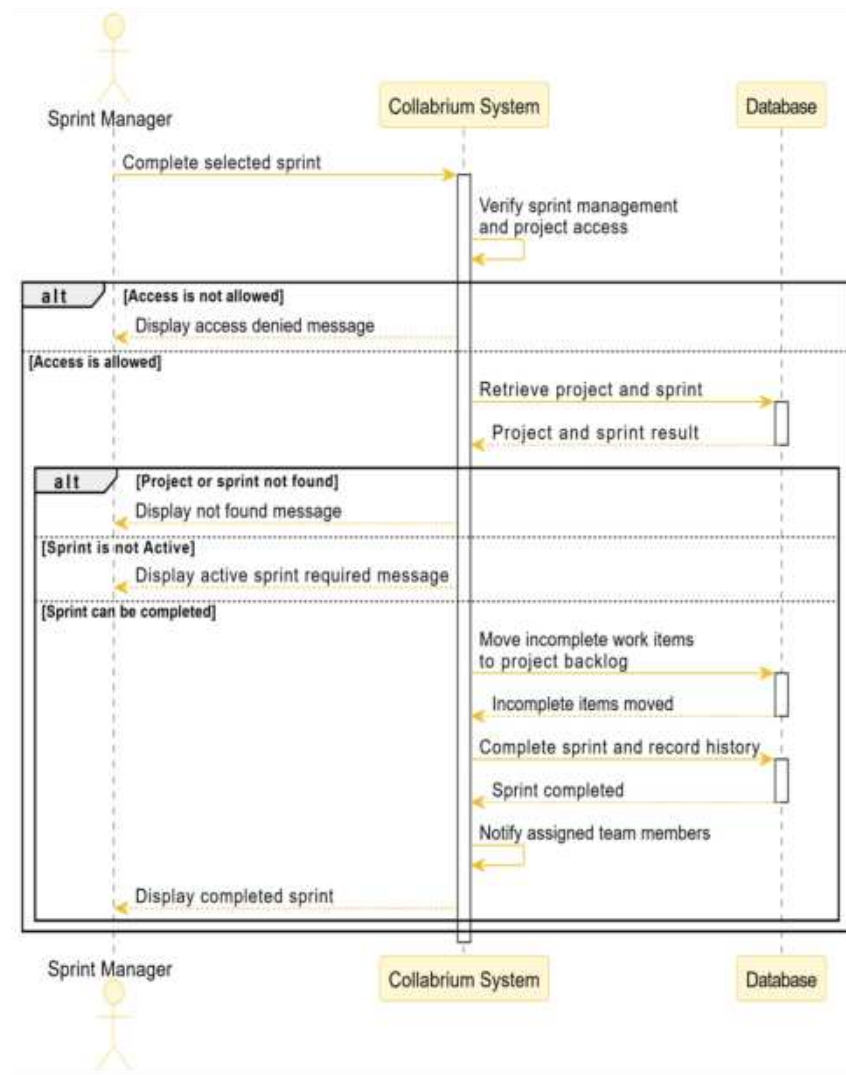


Figure 5.8.8 6 Complete Sprint System Sequence Diagram

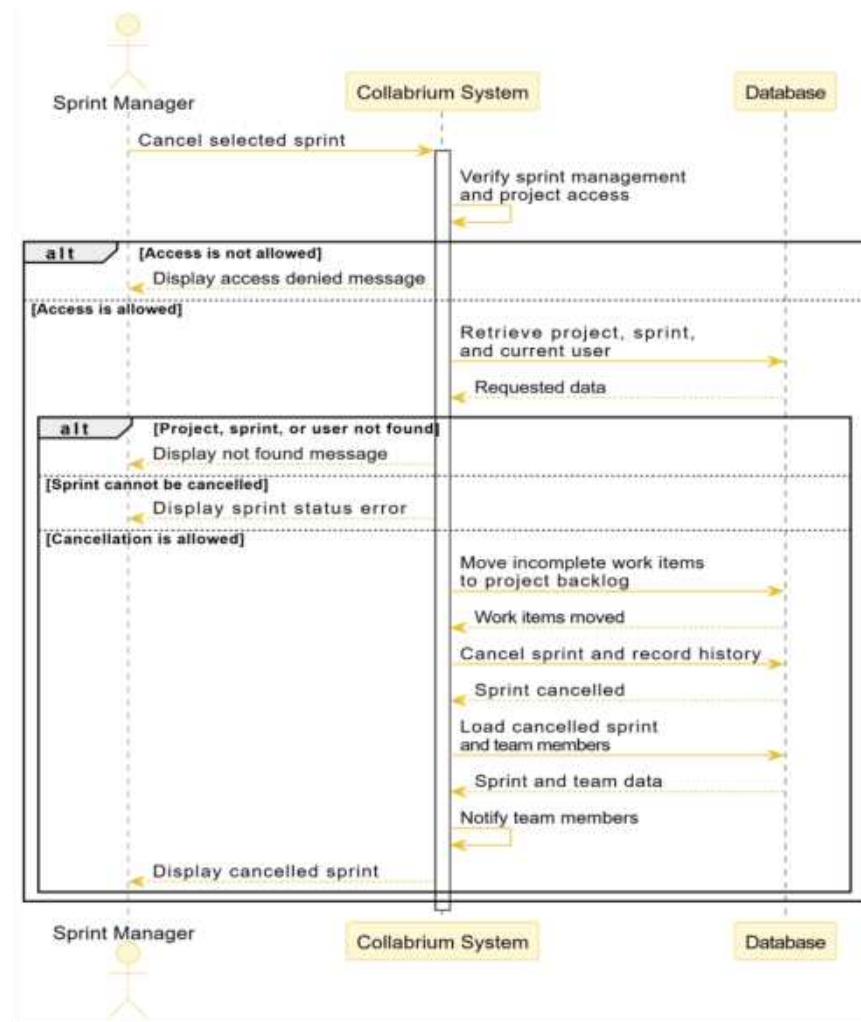


Figure 5.8.8 7 Cancel Sprint System Sequence Diagram

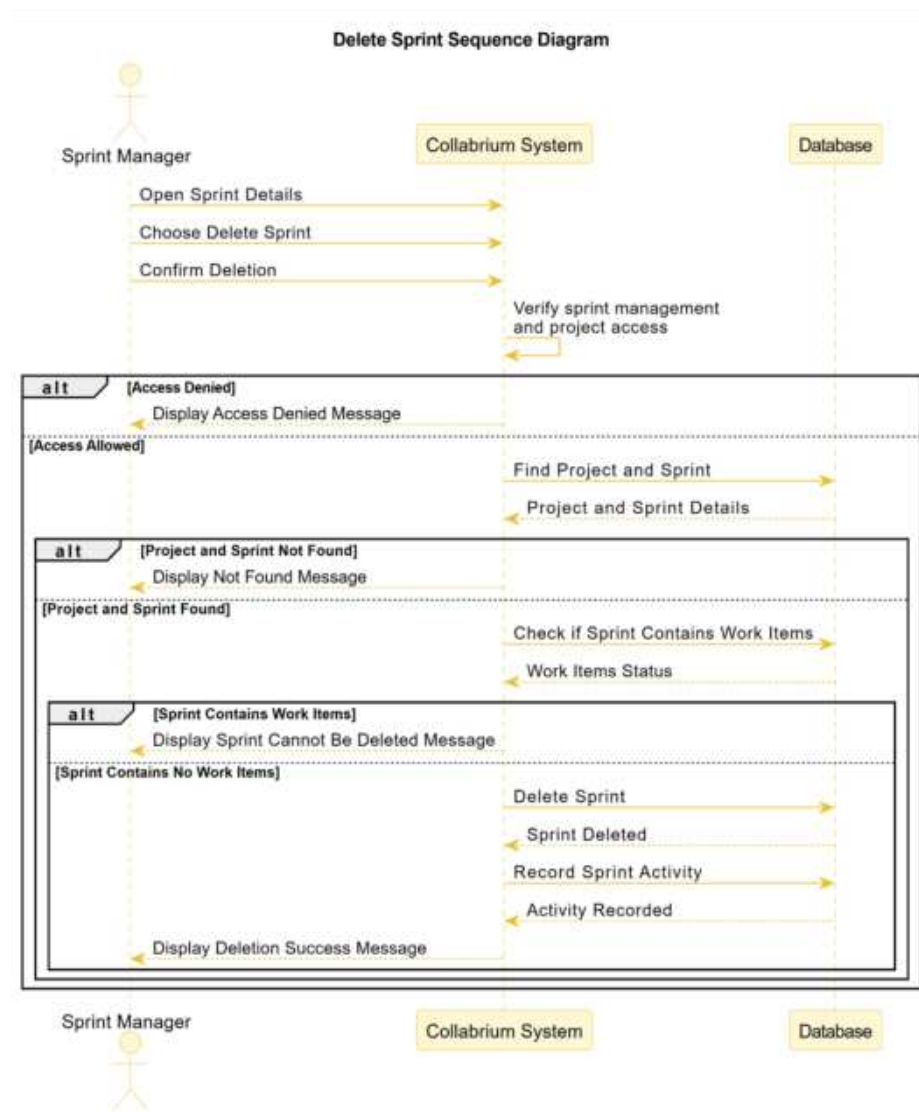


Figure 5.8.8 8 Delete Sprint System Sequence Diagram

## 5.8.9 Reports and Analytics Subsystem Activity Diagrams

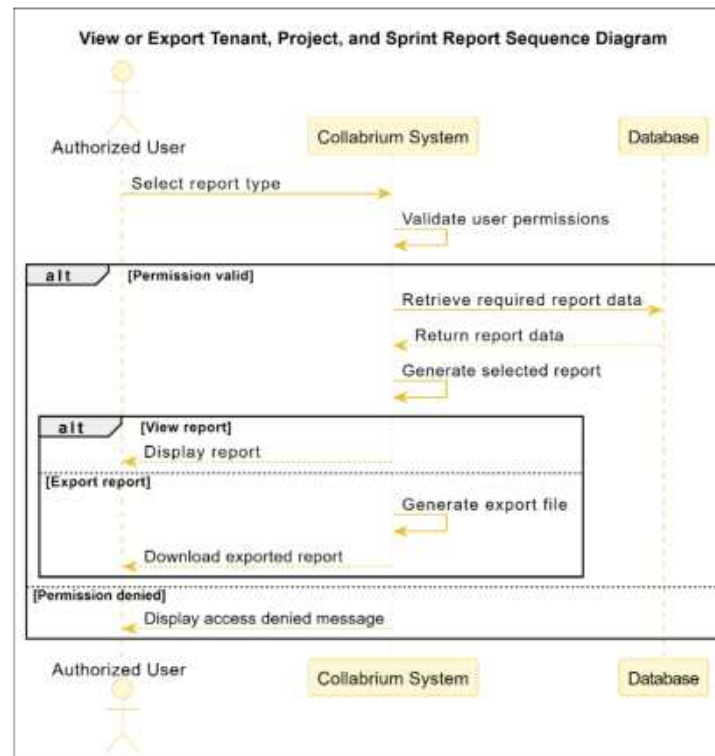


Figure 5.8.9 1 View or Export Tenant, Project, and Sprint Report Sequence Diagram

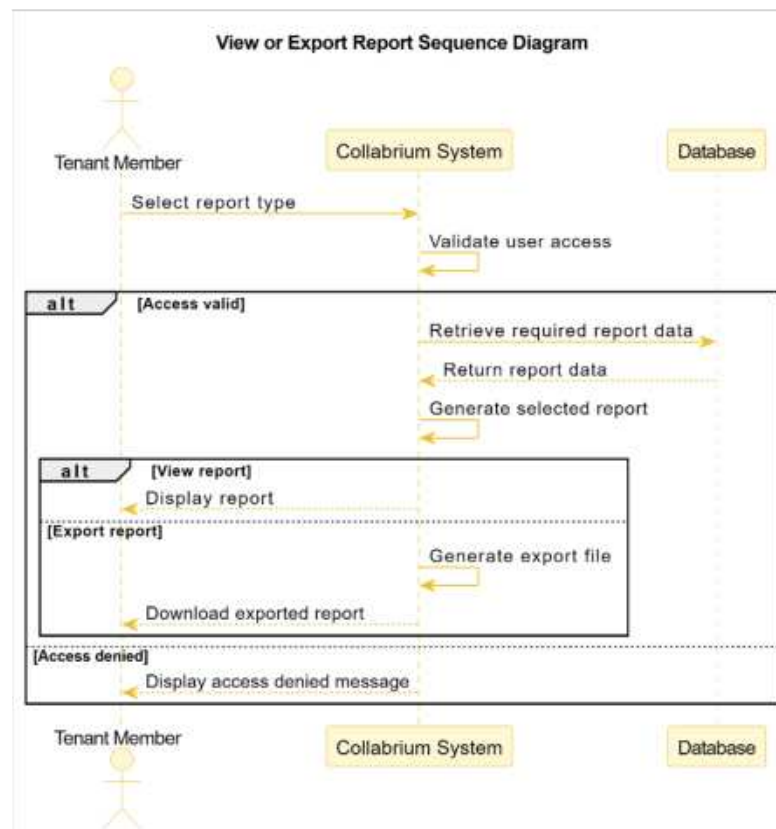


Figure 5.8.9 2 View or Export Report Sequence Diagram

## 5.8.10 Learning hub Subsystem System Sequence Diagram

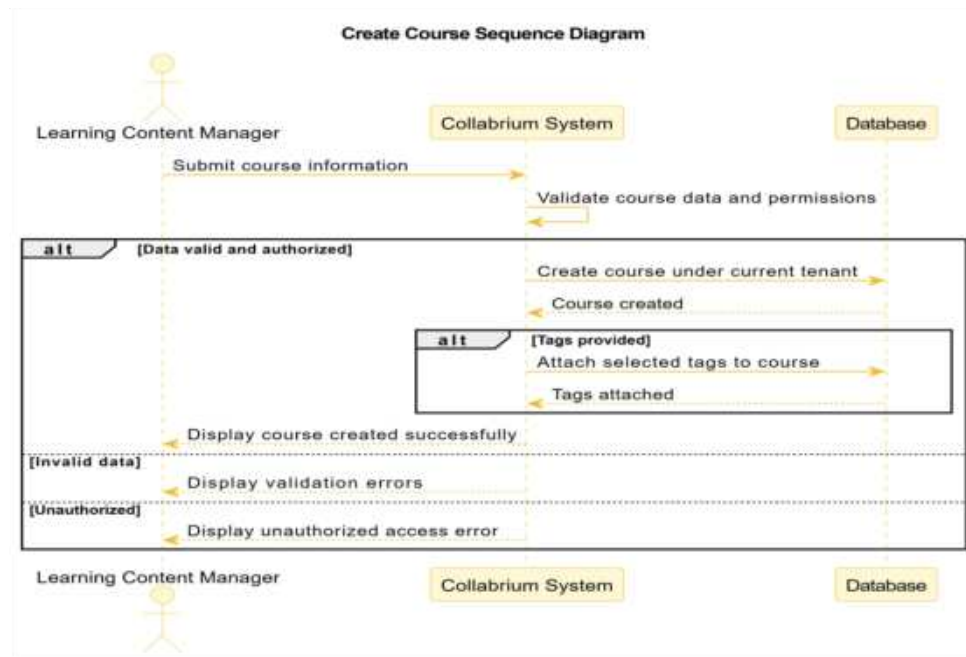


Figure 5.8.10 1 Create Course System Sequence Diagram

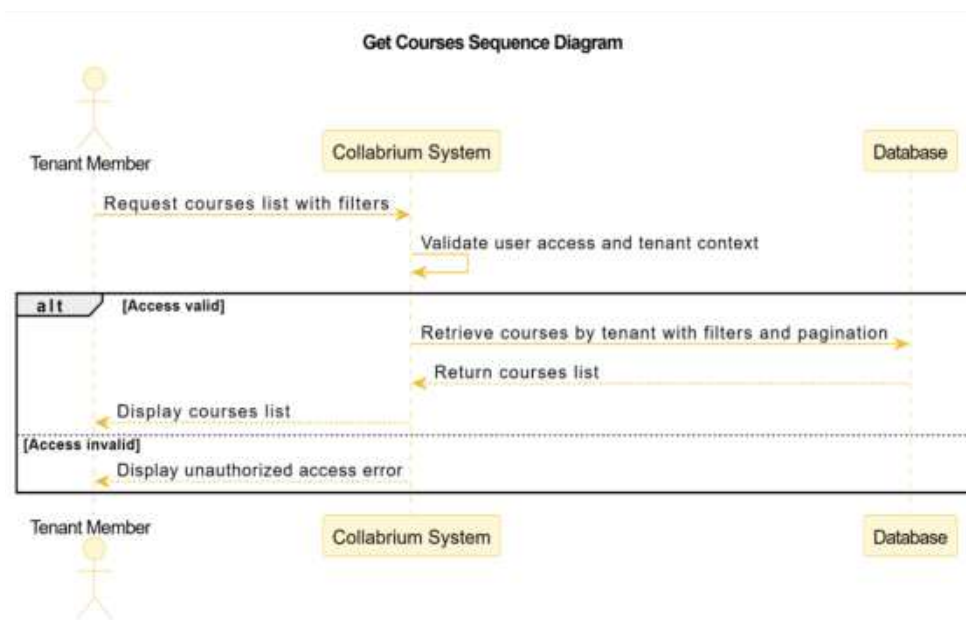


Figure 5.8.10 2 Get Courses System Sequence Diagram

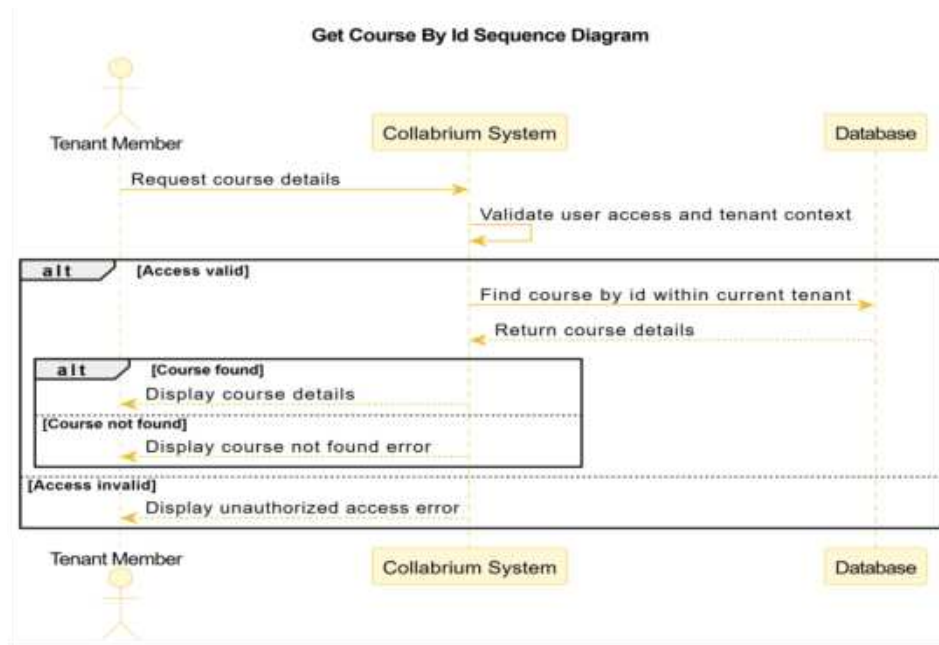


Figure 5.8.10 3 Get Course by id System Sequence Diagram

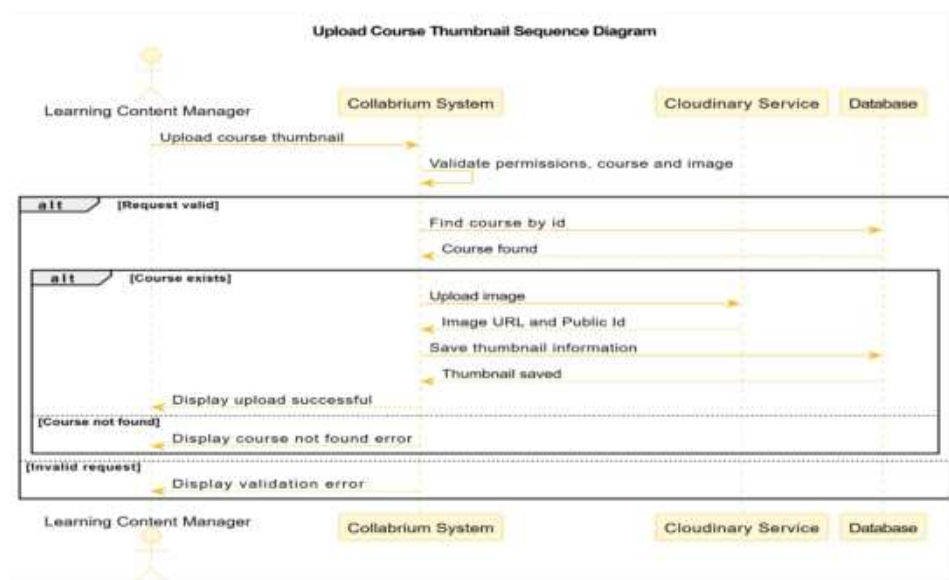


Figure 5.8.10 4 Upload Course Trumbull System Sequence Diagram

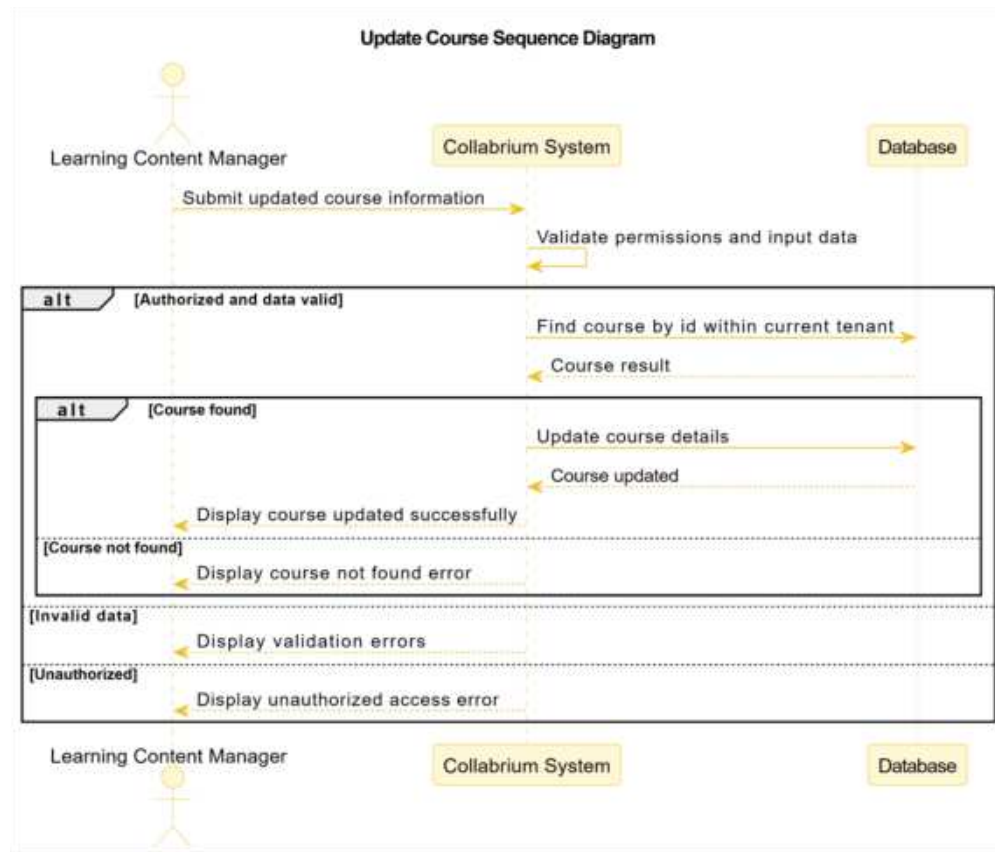


Figure 5.8.10 5 Update Course System Sequence Diagram

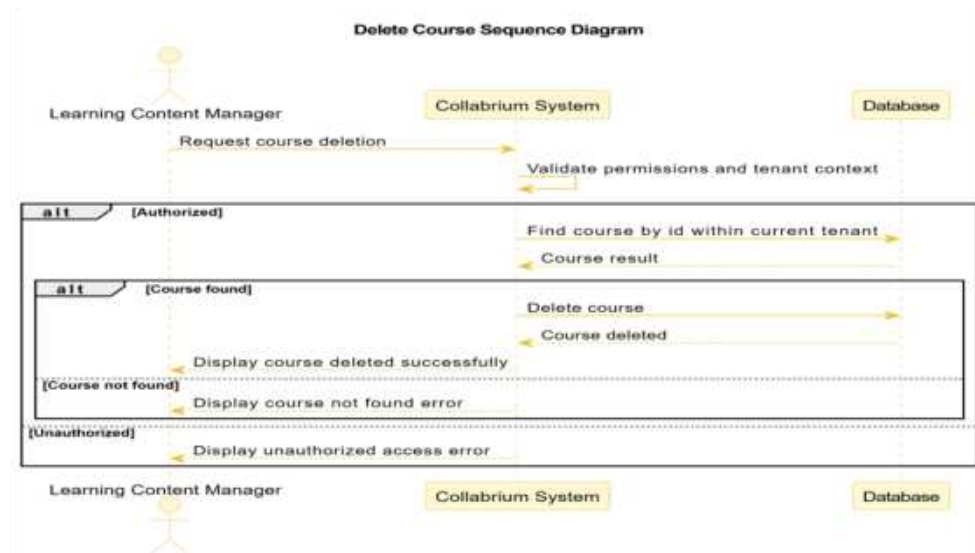


Figure 5.8.10 6 Delete Course System Sequence Diagram

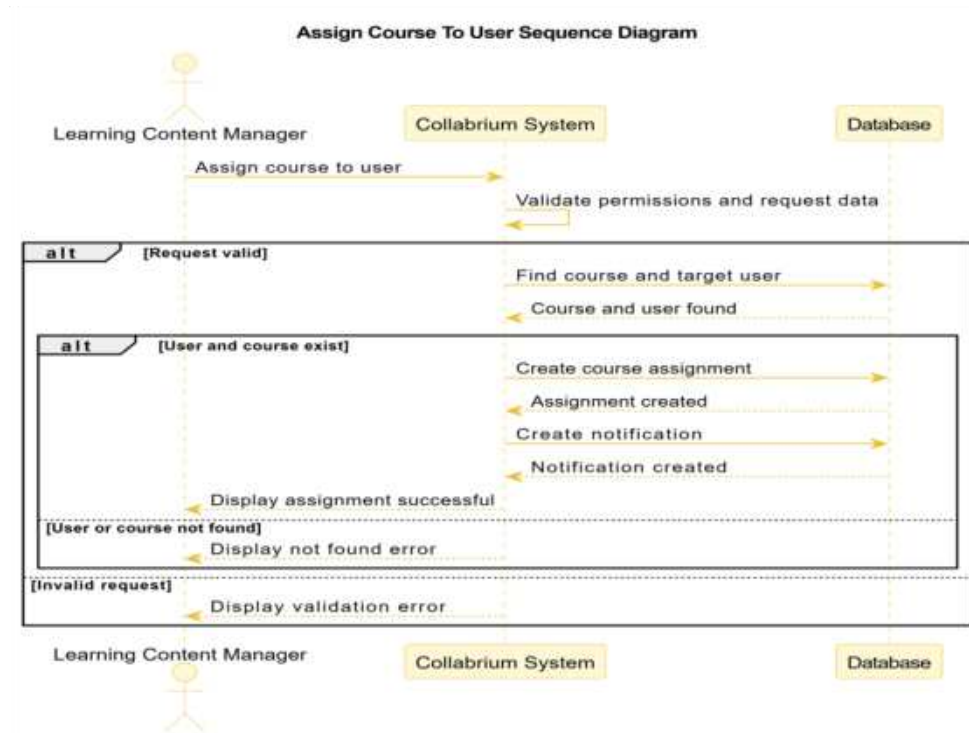


Figure 5.8.10 7 Assign Course to User Sequence Diagram

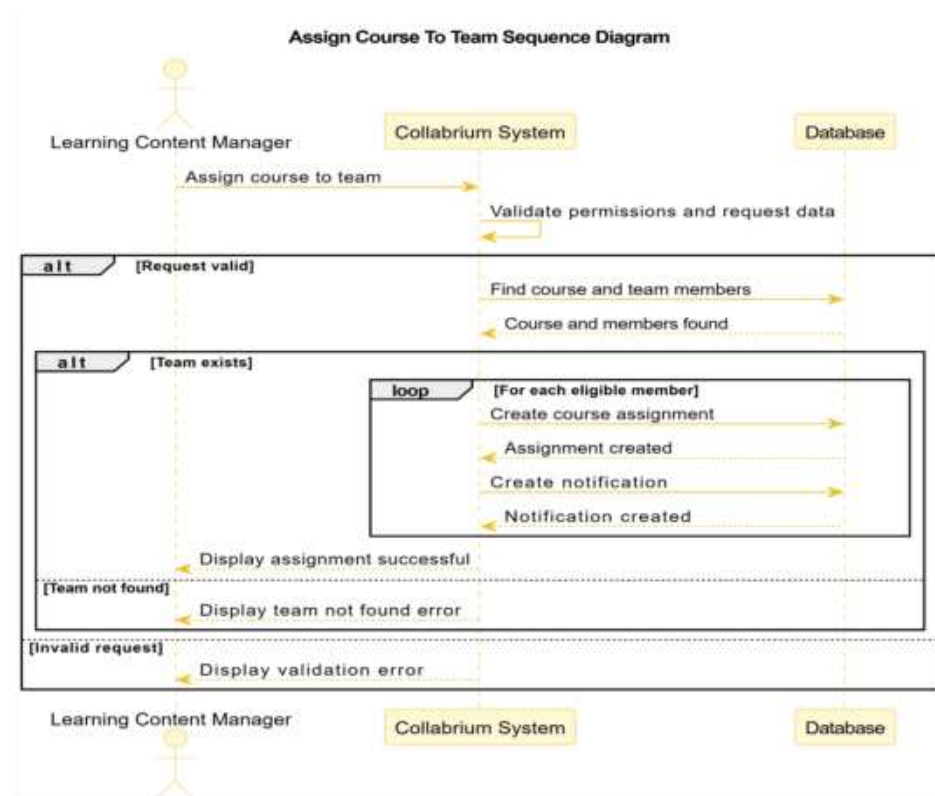


Figure 5.8.10 8 Assign Course to Team System Sequence Diagram

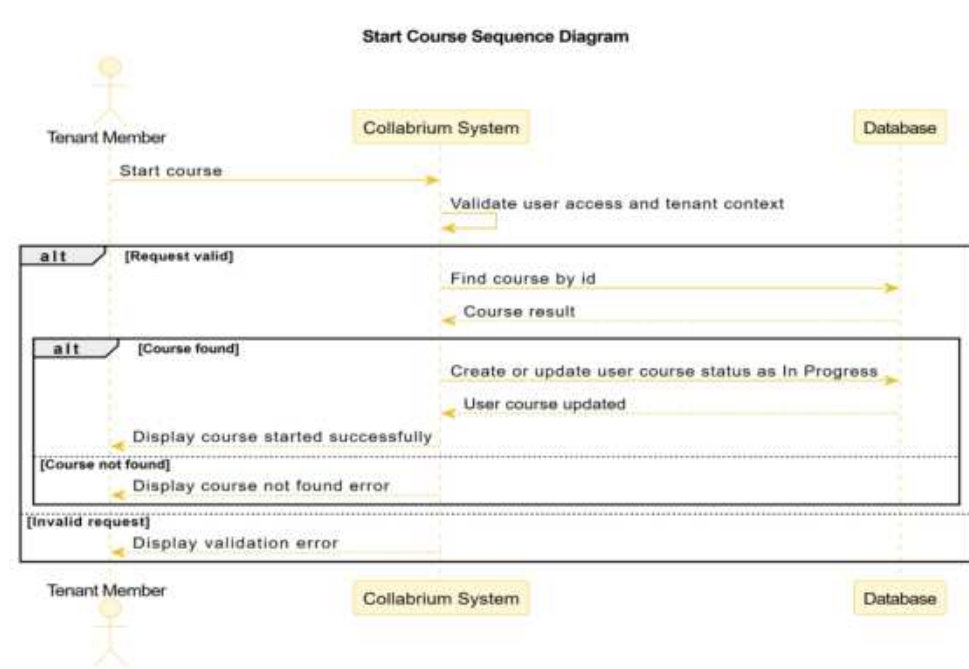


Figure 5.8.10 9 Start Course System Sequence Diagram

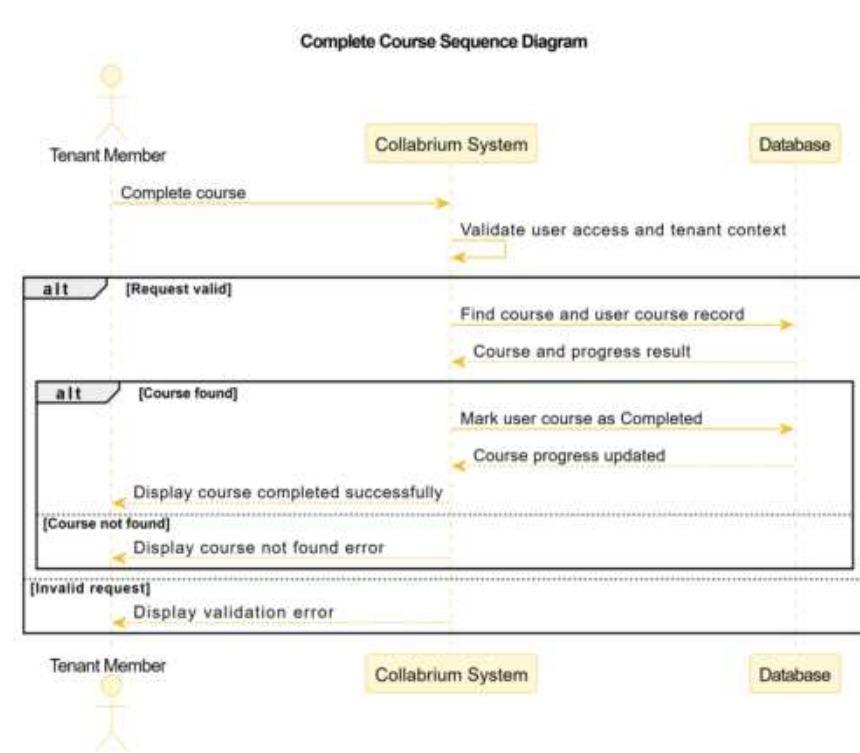


Figure 5.8.10 10 Complete Course System Sequence Diagram

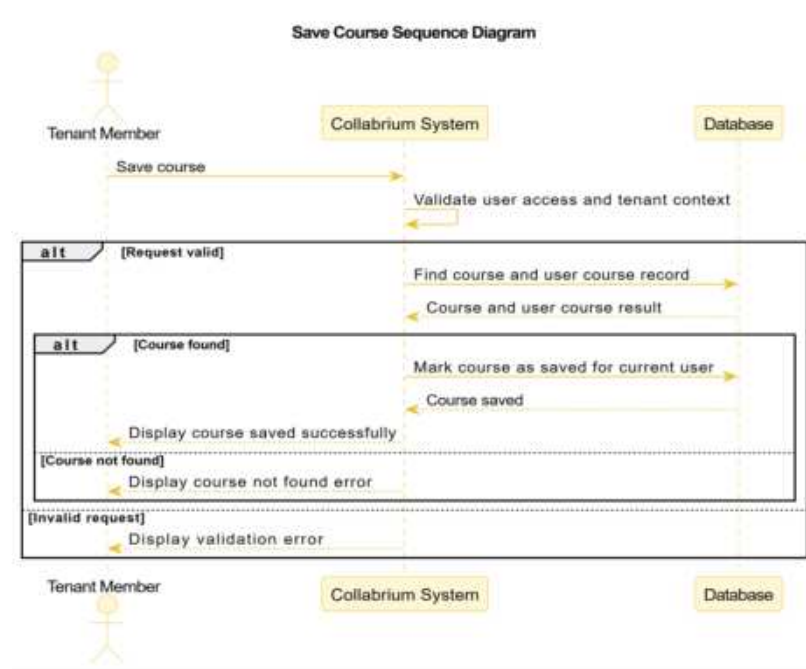


Figure 5.8.10 11 Save Course System Sequence Diagram

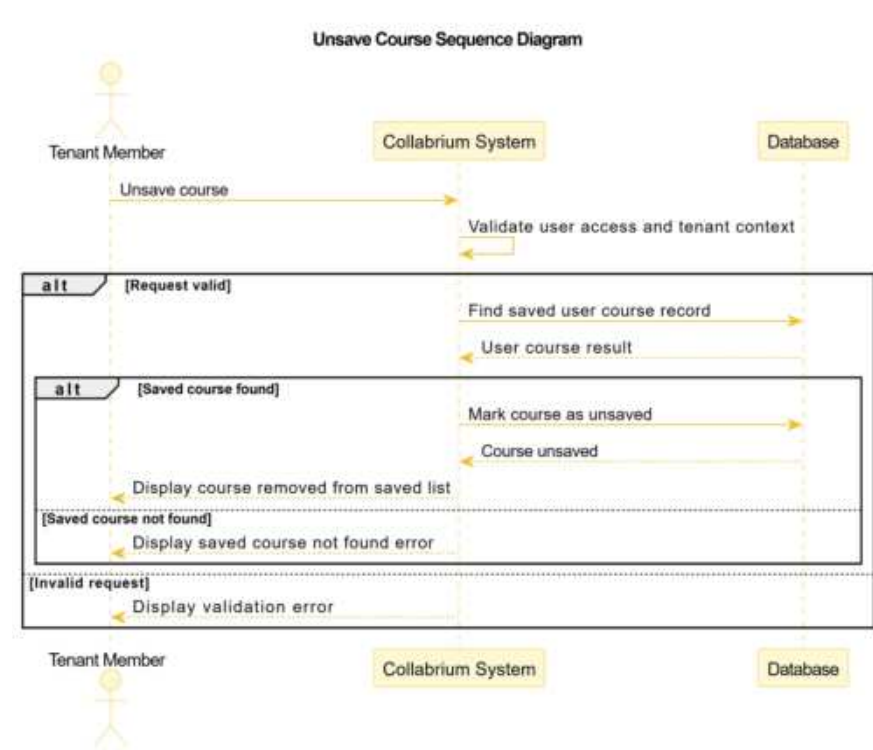


Figure 5.8.10 12 Un Save Course System Sequence Diagram

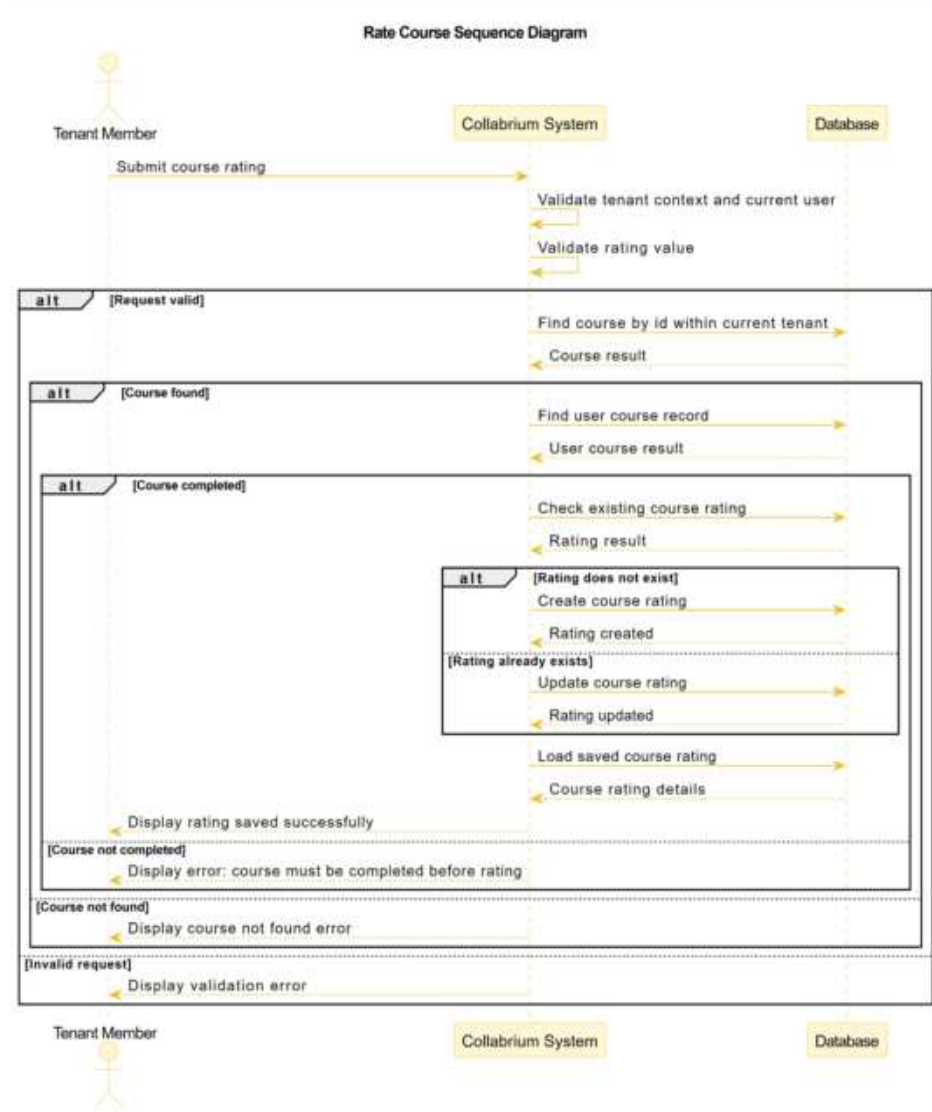


Figure 5.8.10 13 Rate Course System Sequence Diagram

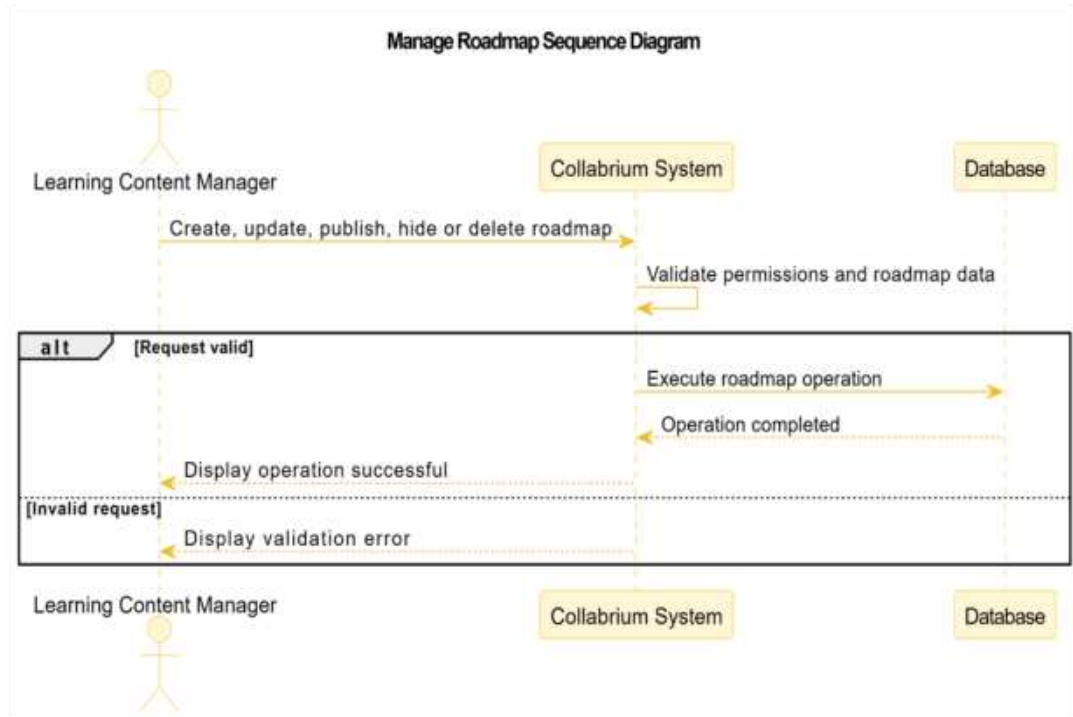


Figure 5.8.10 14 Manage Roadmap System Sequence Diagram

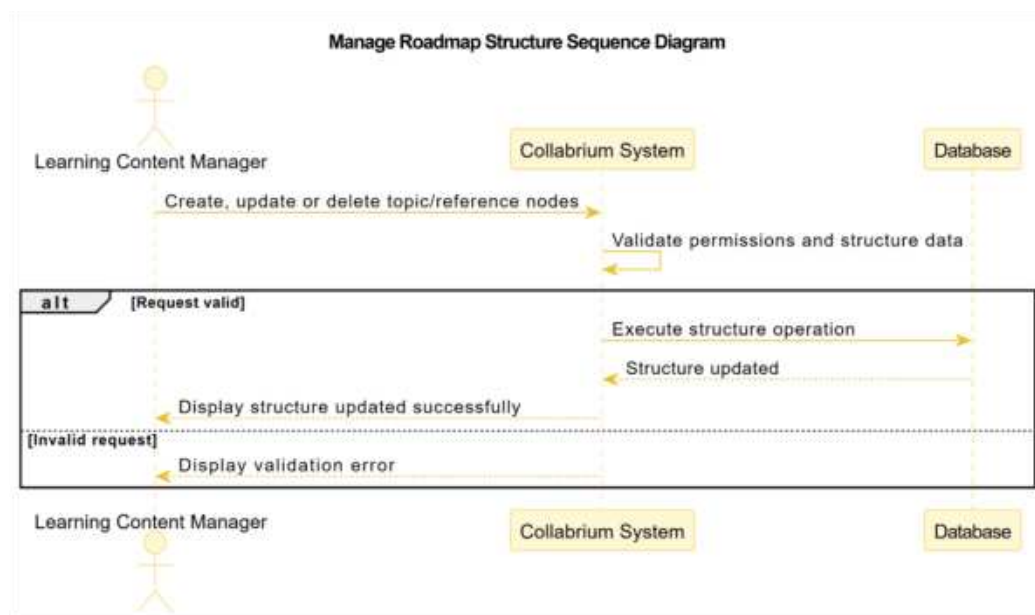


Figure 5.8.10 15 Manage Roadmap Structure System Sequence Diagram

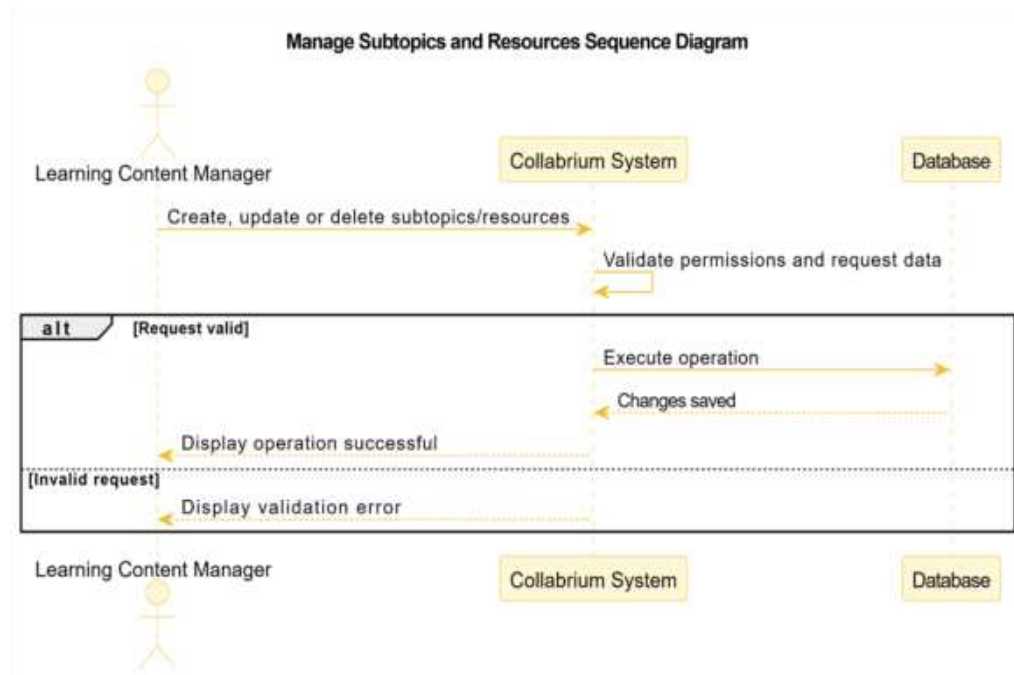


Figure 5.8.10 16 Manage Subtopics and Resource System Sequence Diagram

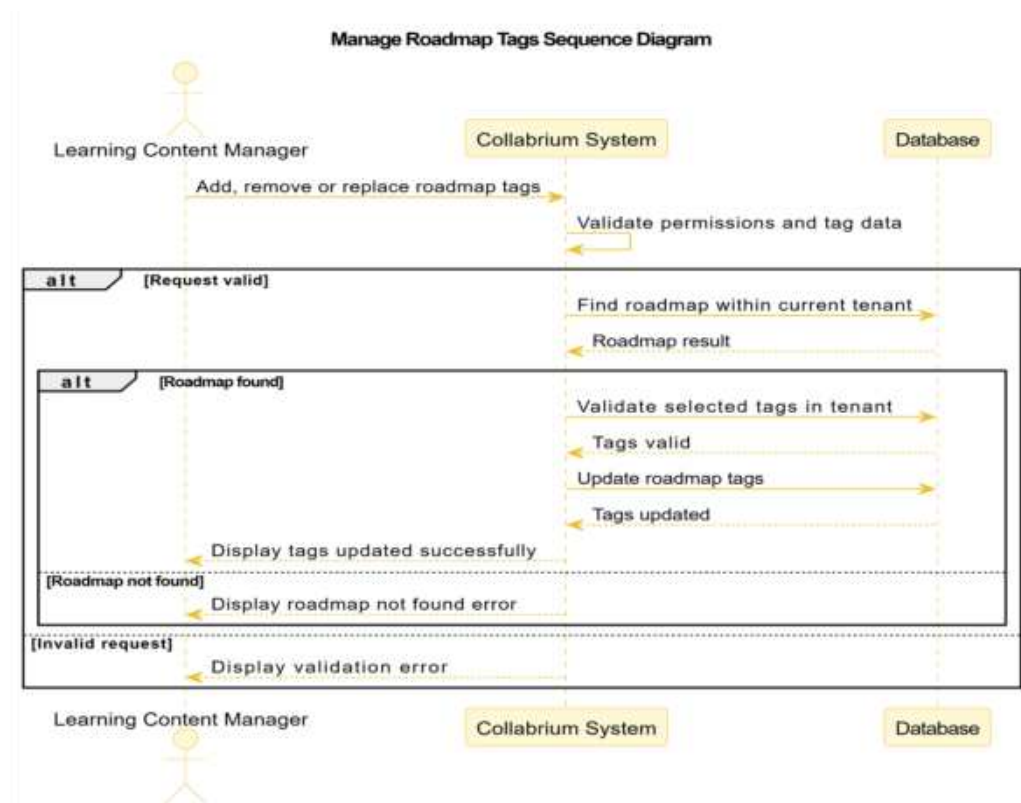


Figure 5.8.10 17 Manage Roadmap Tags Sequence Diagram

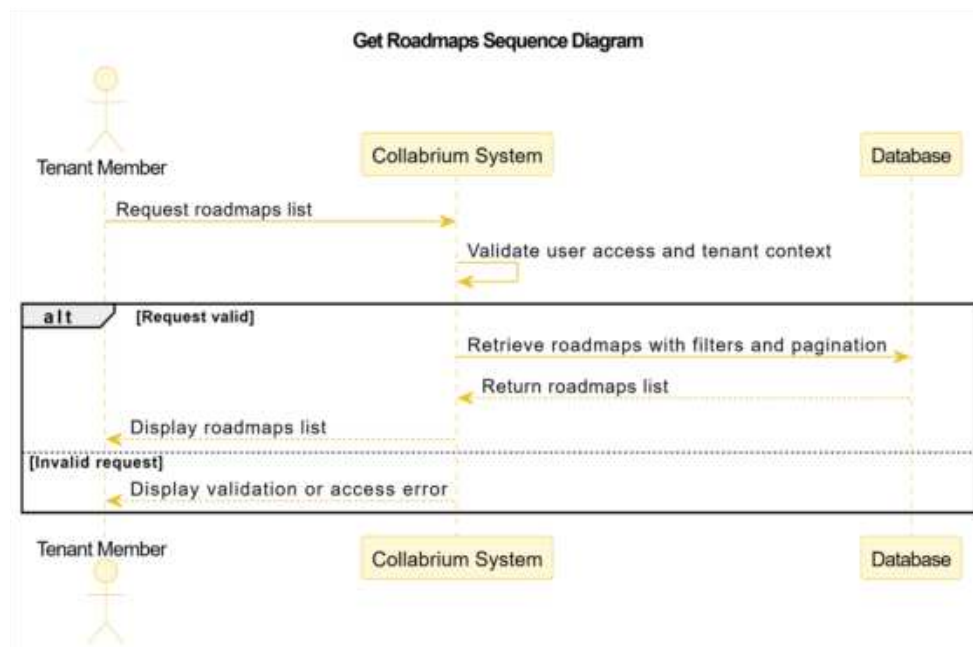


Figure 5.8.10 18 Get Roadmaps System Sequence Diagram

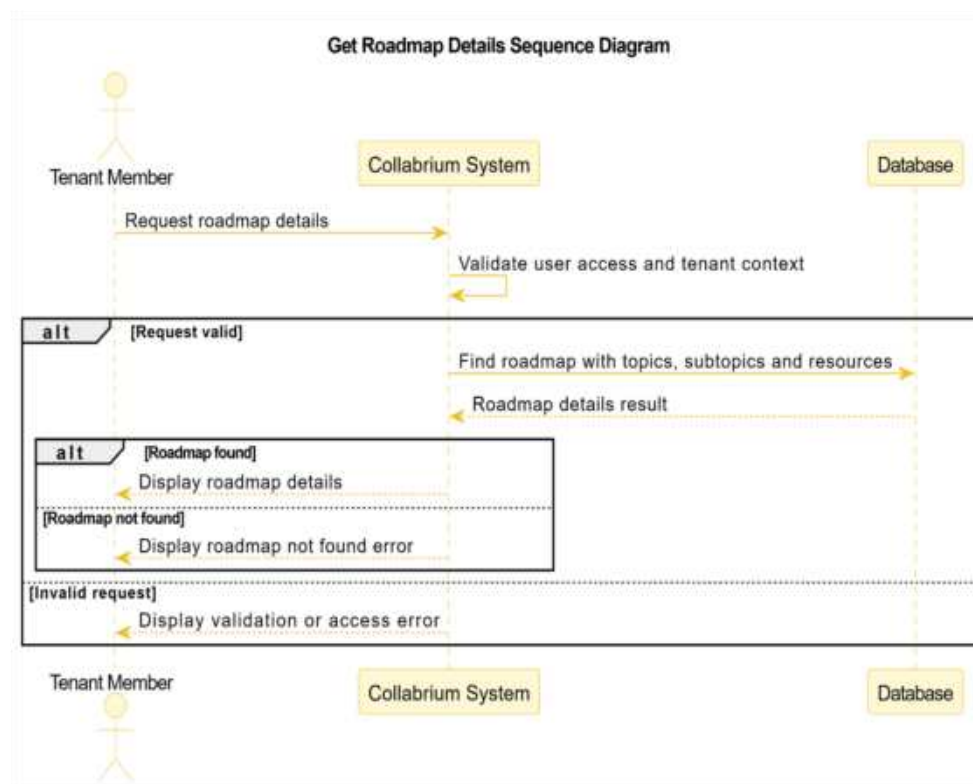


Figure 5.8.10 19 Get Roadmap Tag System Sequence Diagram

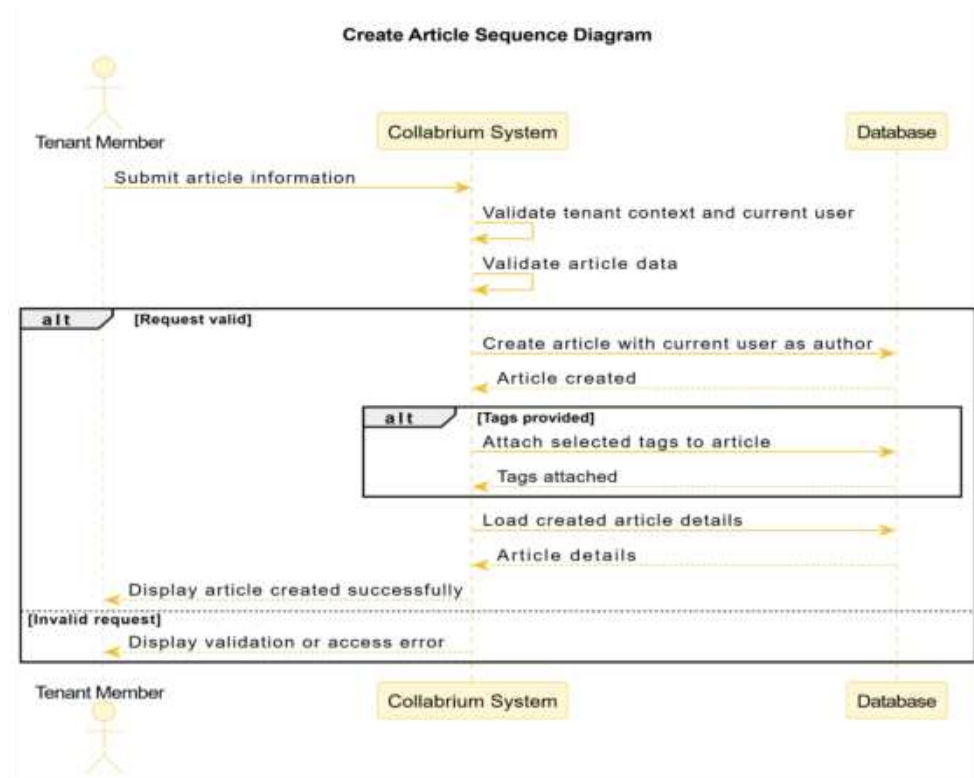


Figure 5.8.10 20 Create Article System Sequence Diagram

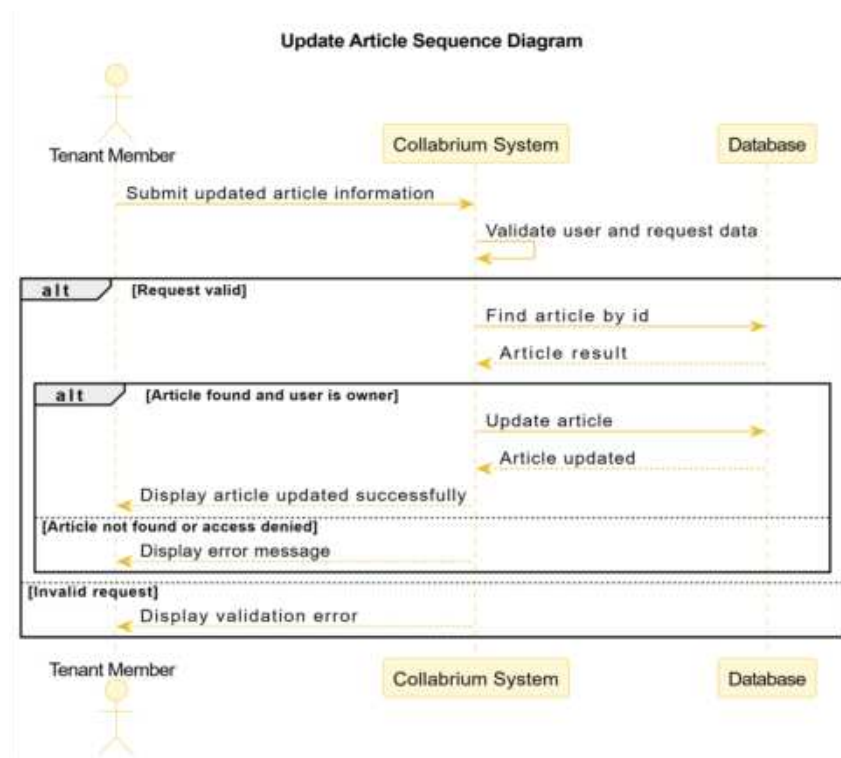


Figure 5.8.10 21 Update Article System Sequence Diagram

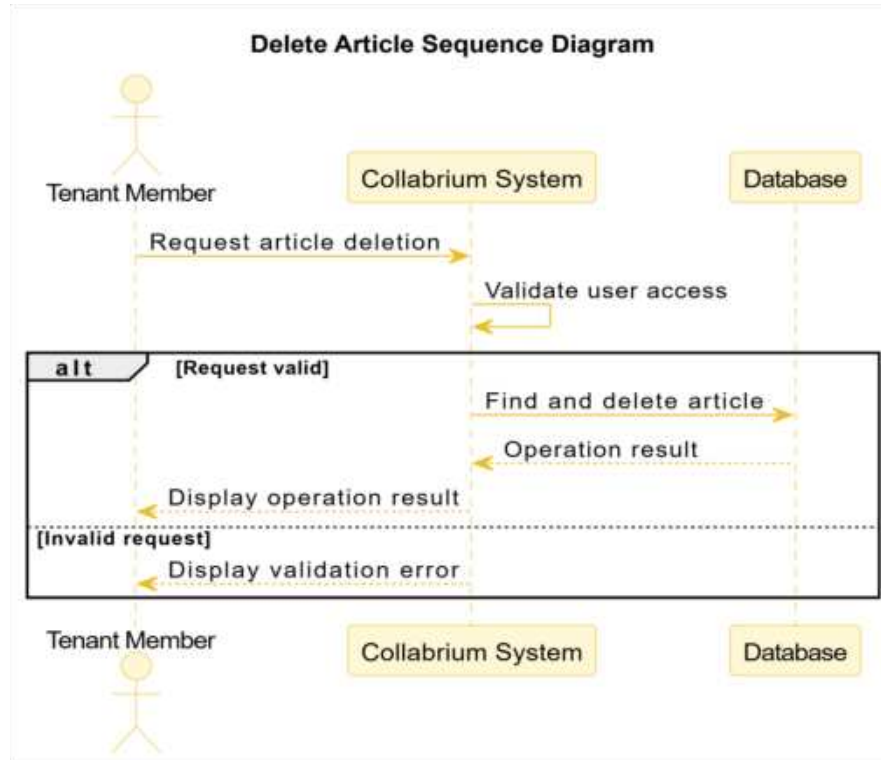


Figure 5.8.10 22 Delete Article System Sequence Diagram

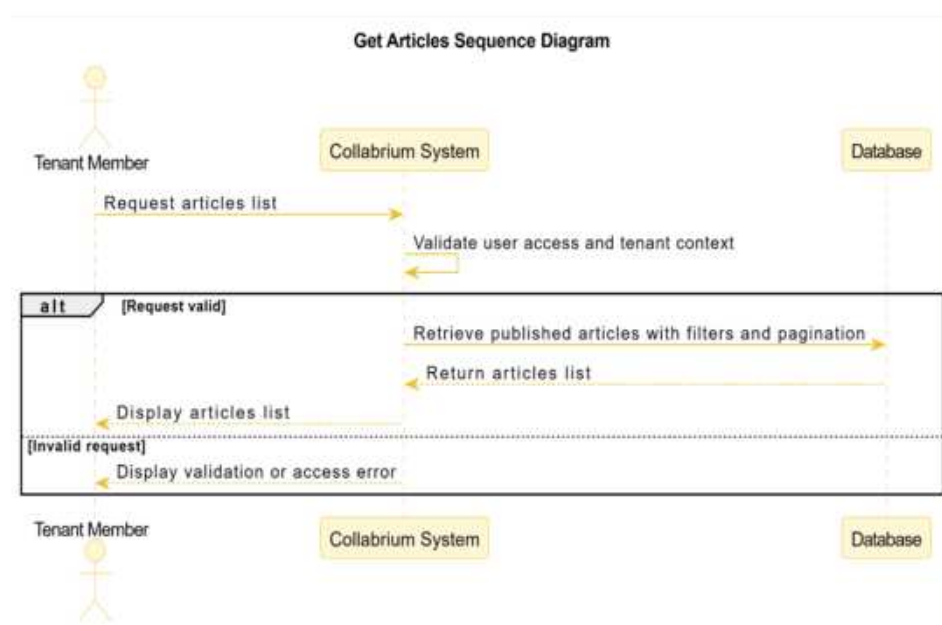


Figure 5.8.10 23 Get Articles System Sequence Diagram

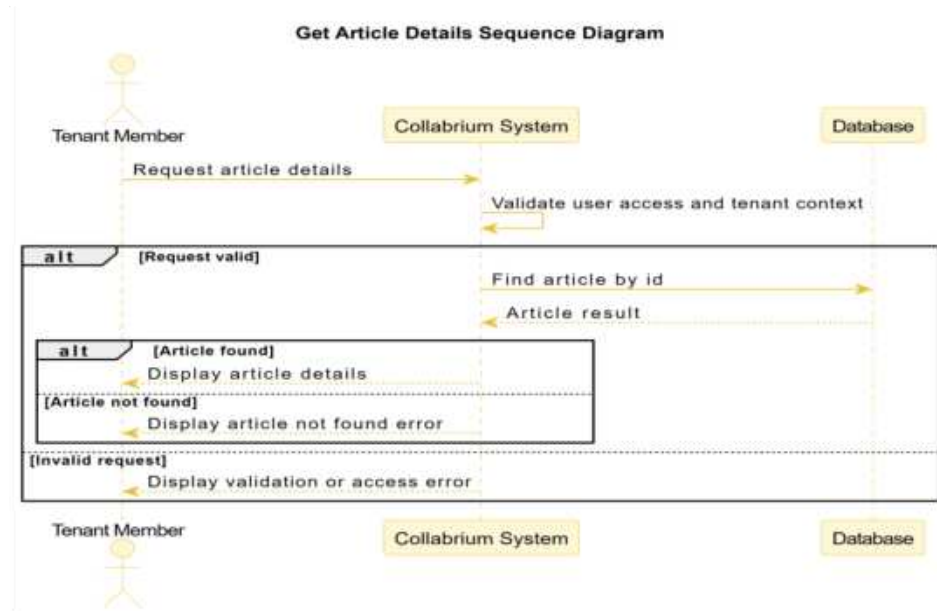


Figure 5.8.10 24 Get Article Details System Sequence Diagram

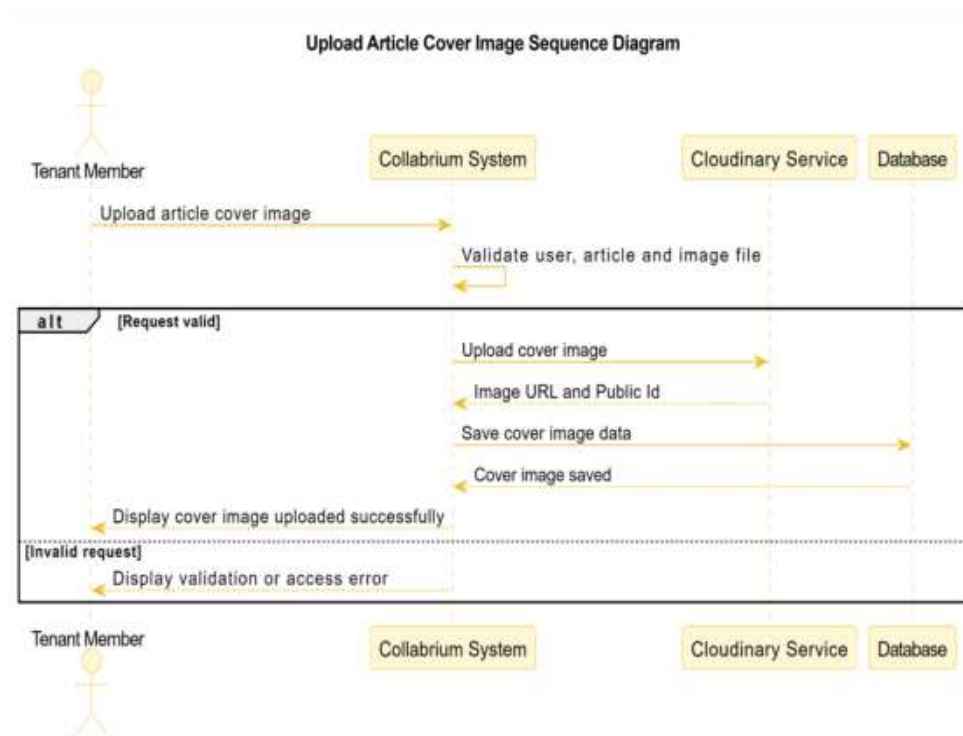


Figure 5.8.10 25 Upload Article Cover Page System Sequence Diagram

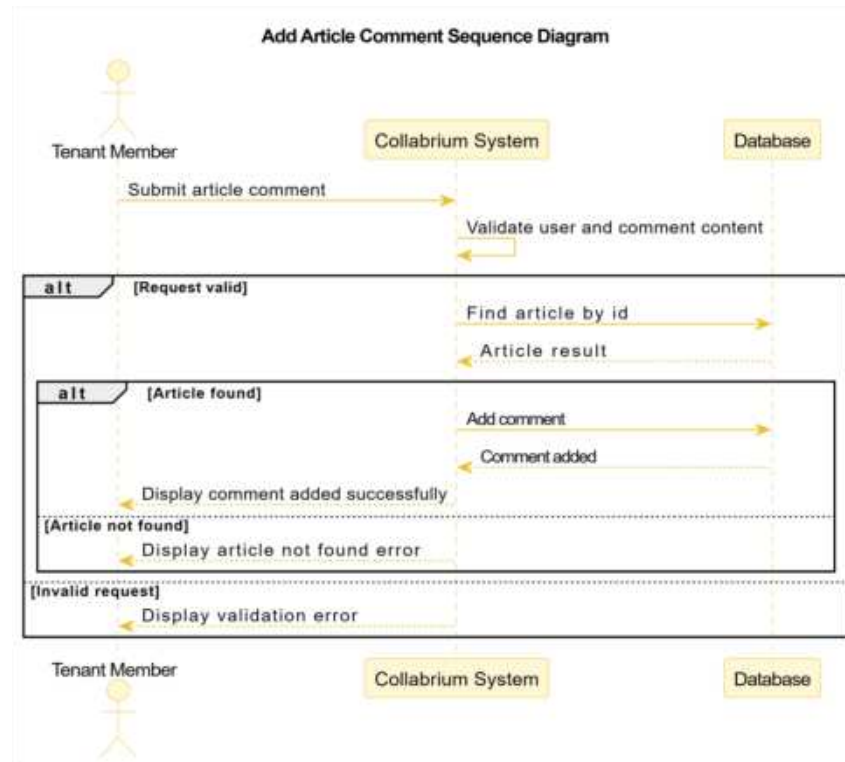


Figure 5.8.10 26 Add Article Comment System Sequence Diagram

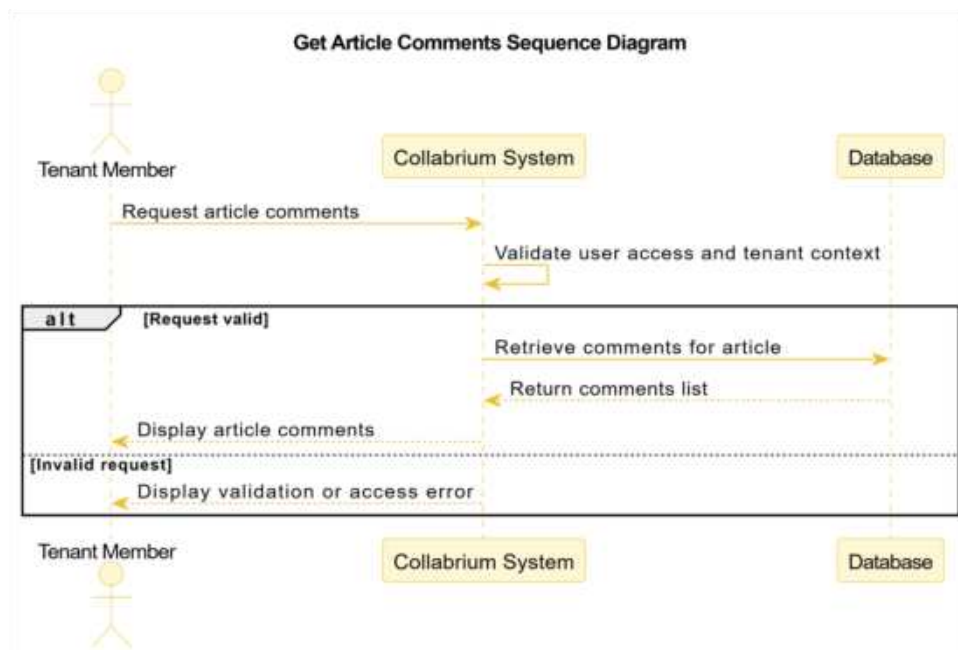


Figure 5.8.10 27 Get Article Comments System Sequence Diagram

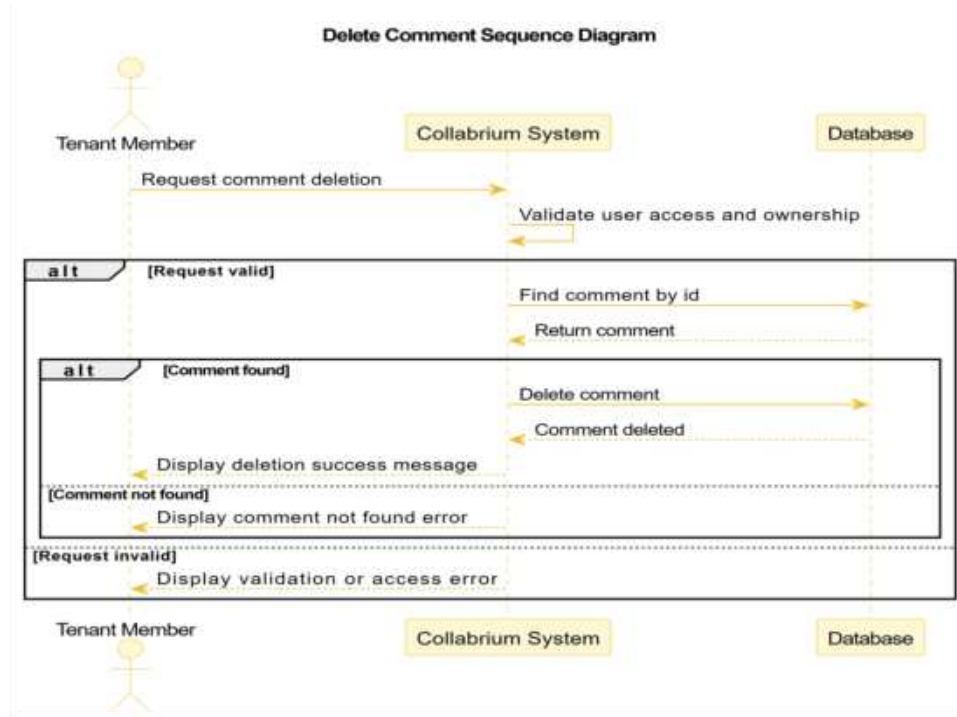


Figure 5.8.10 28 Delete Article Comments System Sequence Diagram

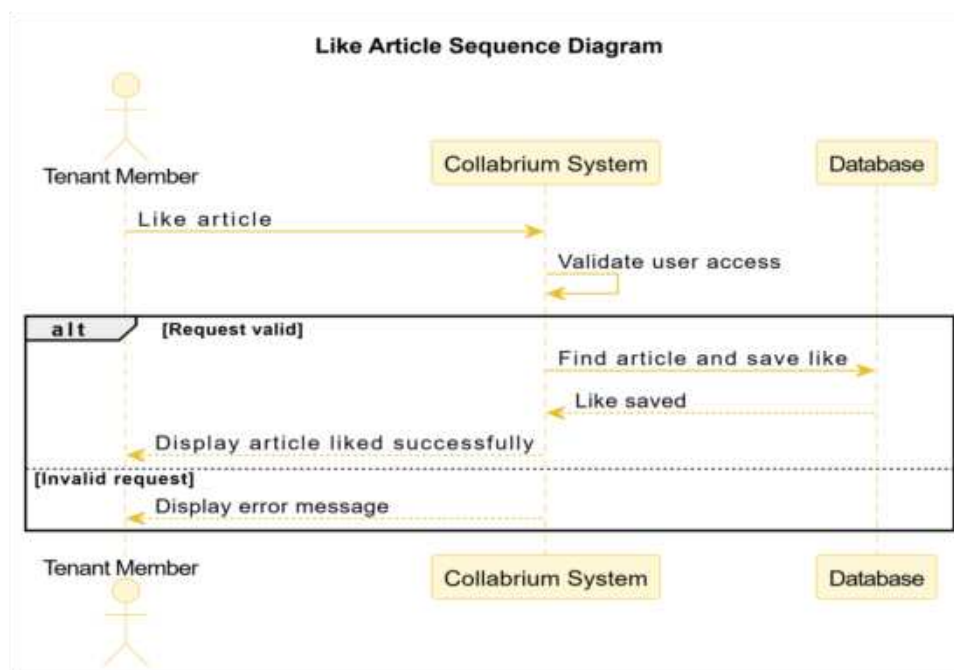


Figure 5.8.10 29 Like Article System Sequence Diagram

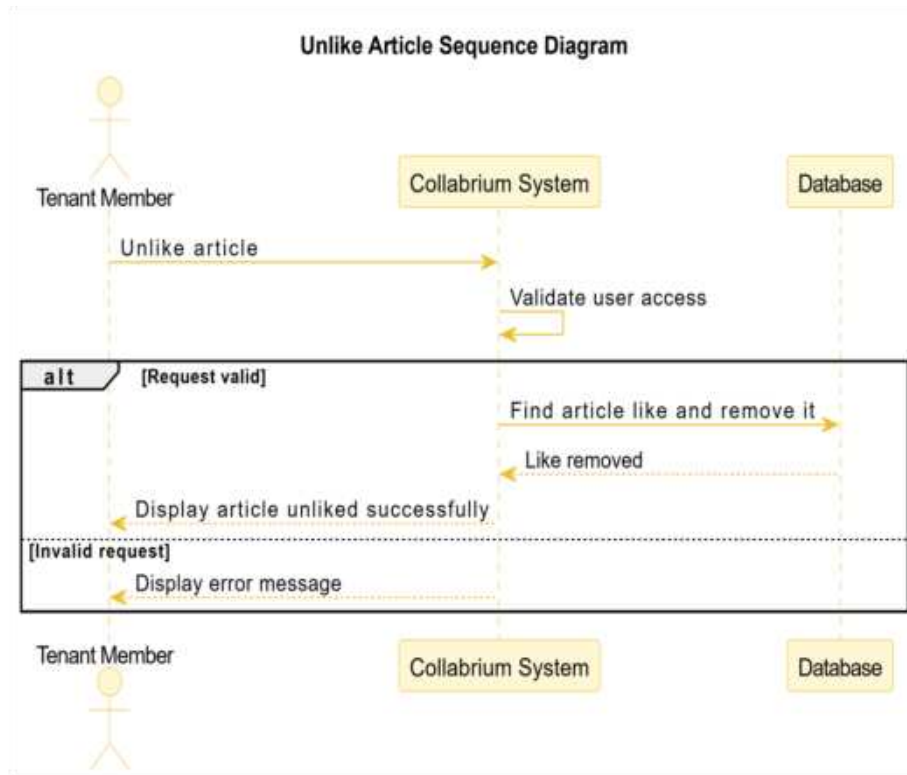


Figure 5.8.10 30 Unlike Article System Sequence Diagram

### 5.8.11 Community hub Subsystem System Sequence Diagram

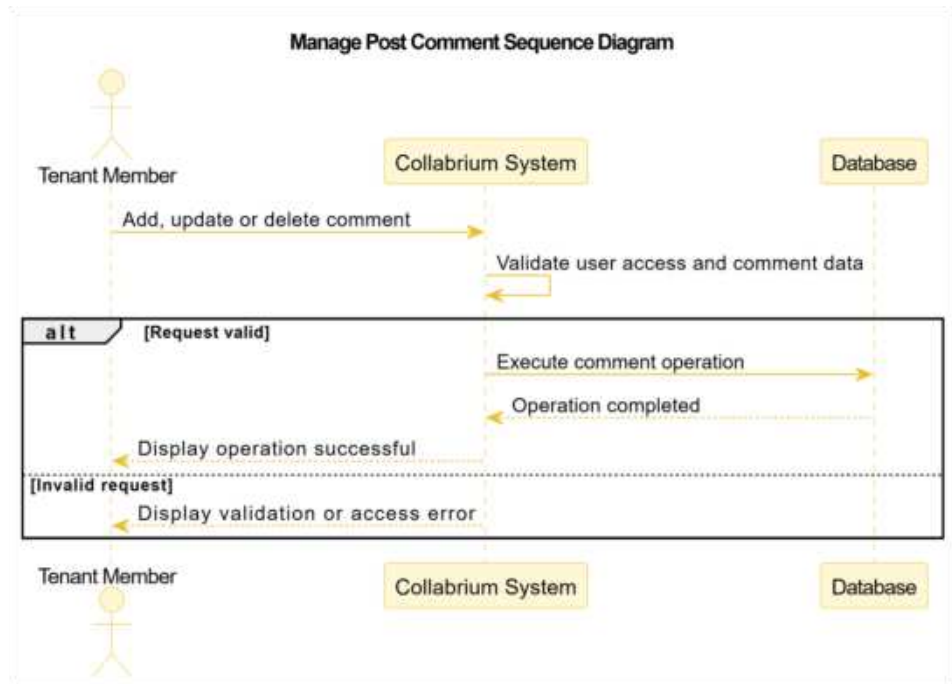


Figure 5.8.11 1 Manage Post System Sequence Diagram

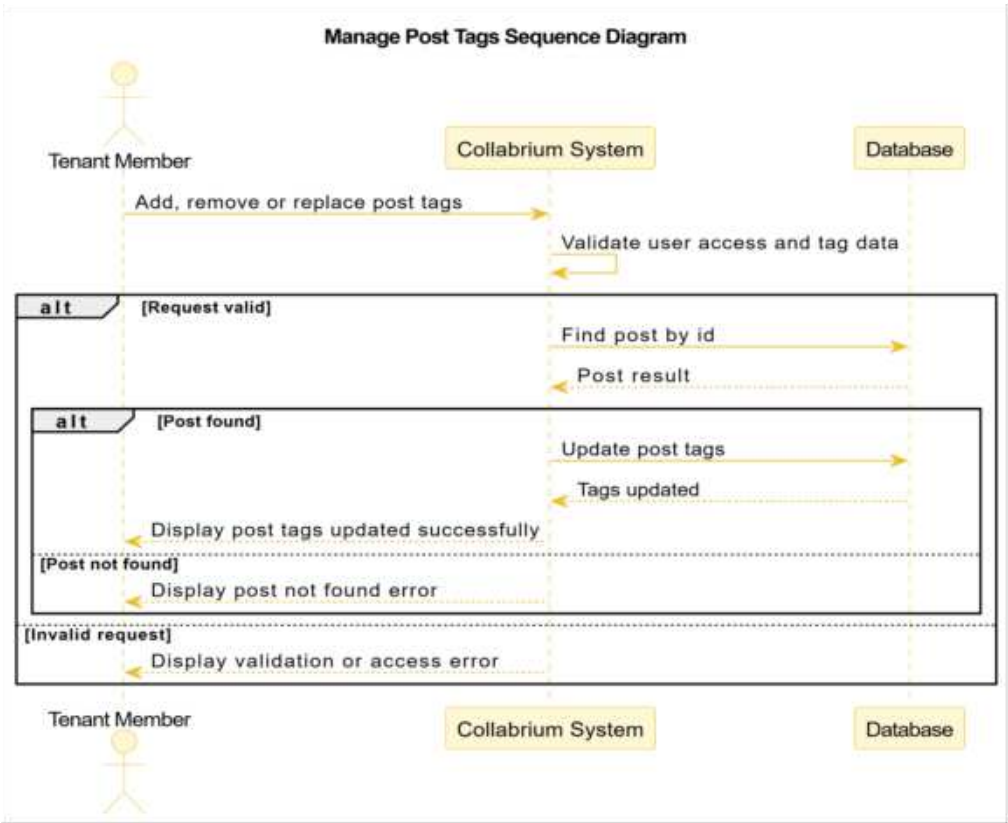


Figure 5.8.11 2 Manage Post Tag System Sequence Diagram

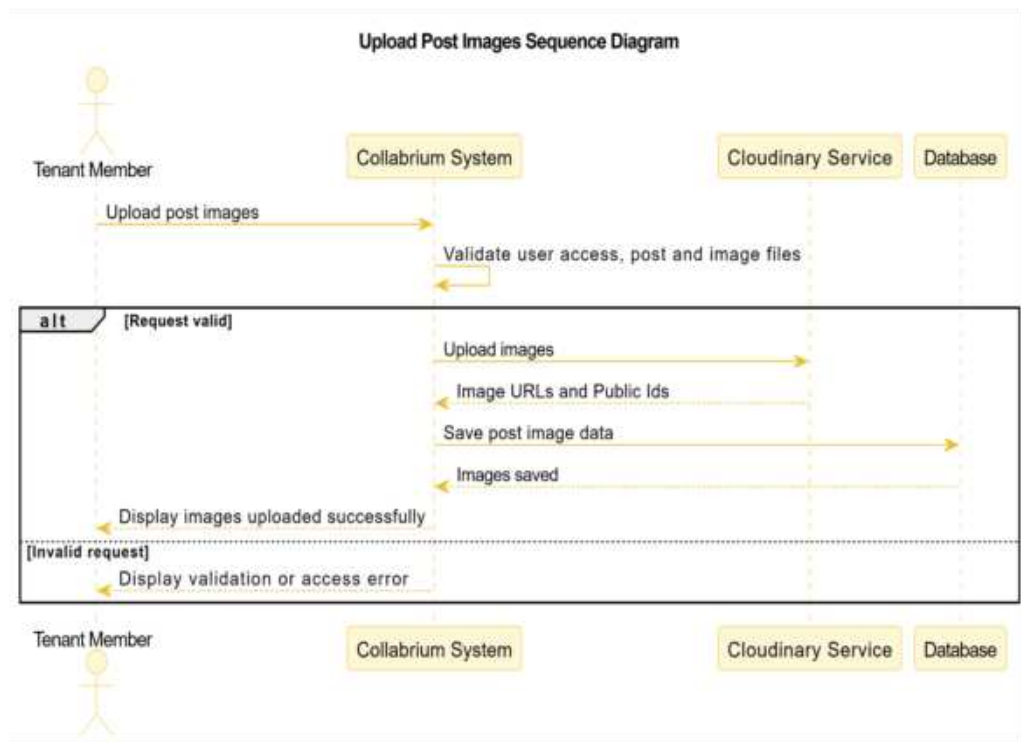


Figure 5.8.11 3 Upload Post Images Tag System Sequence Diagram

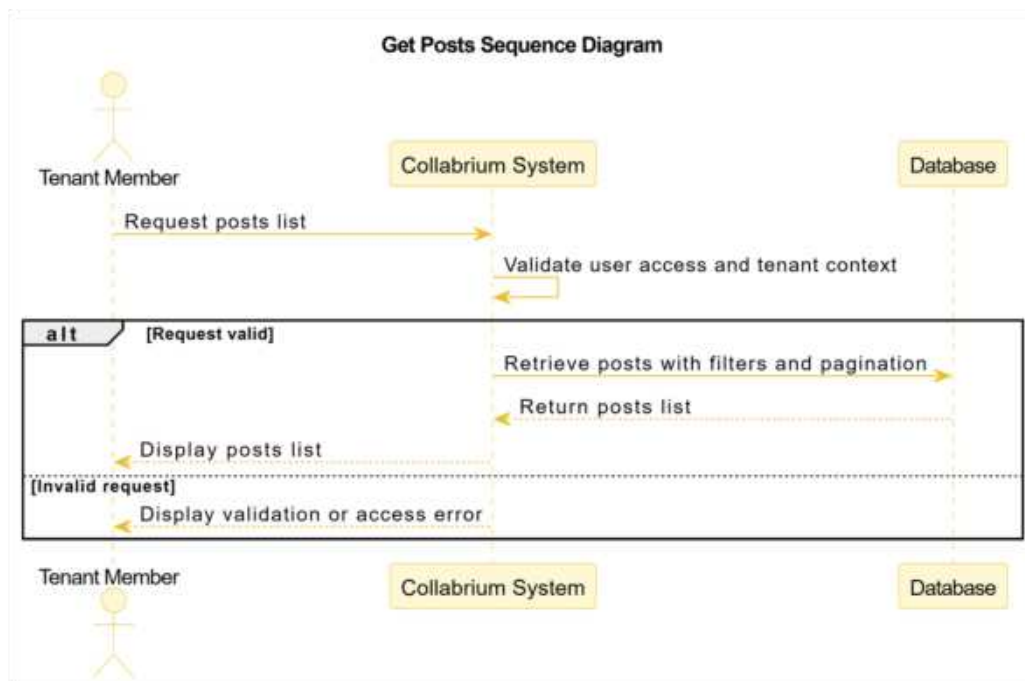


Figure 5.8.11 4 Get Posts System Sequence Diagram

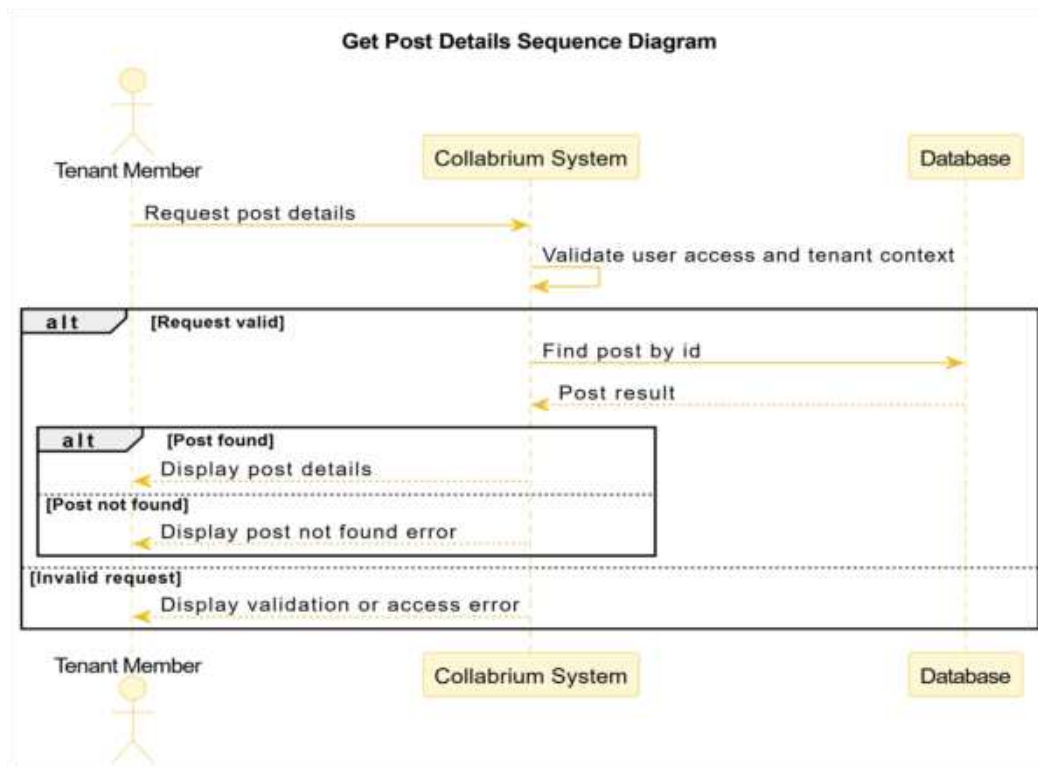


Figure 5.8.11 5 Get Post Details System Sequence Diagram

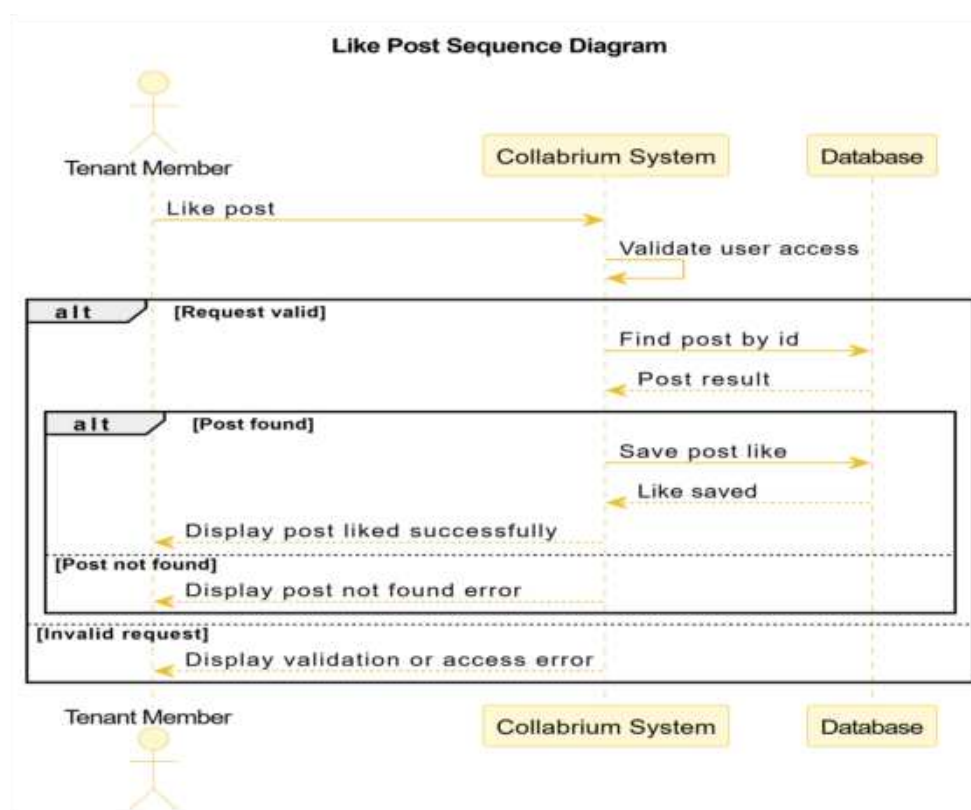


Figure 5.8.11 6 Like Post System Sequence Diagram

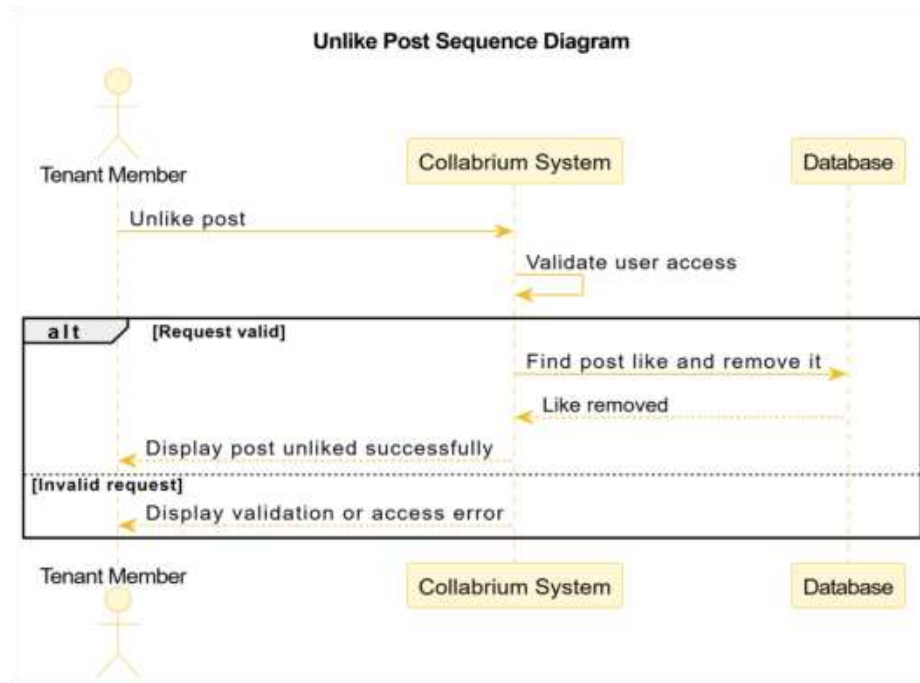


Figure 5.8.11 7 Unlike Post System Sequence Diagram

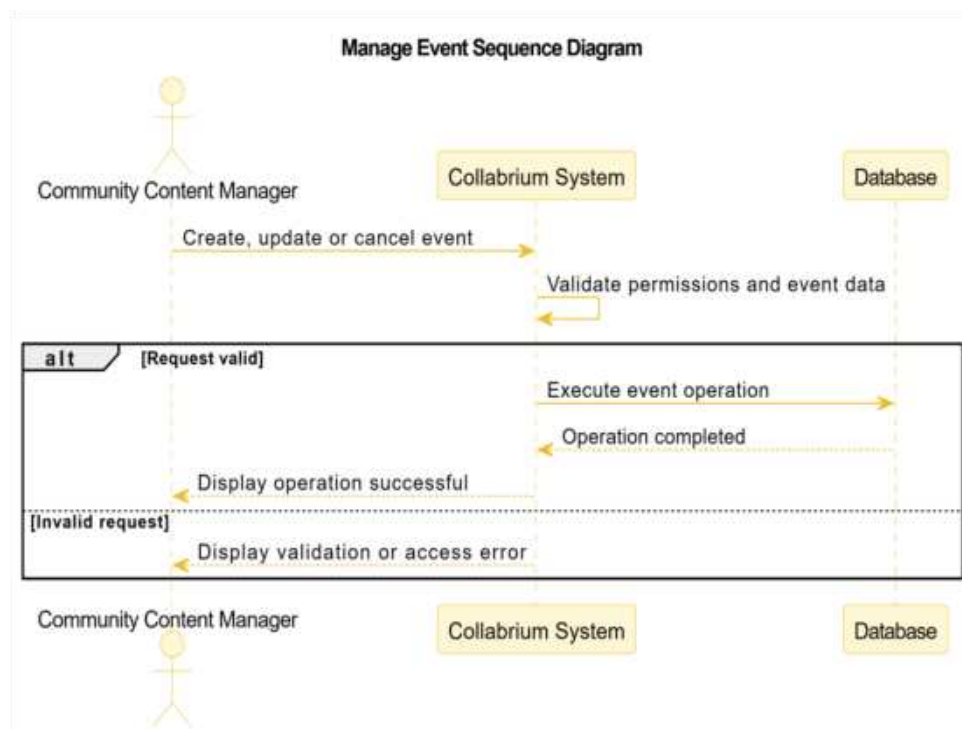


Figure 5.8.11 8 Manage Event System Sequence Diagram

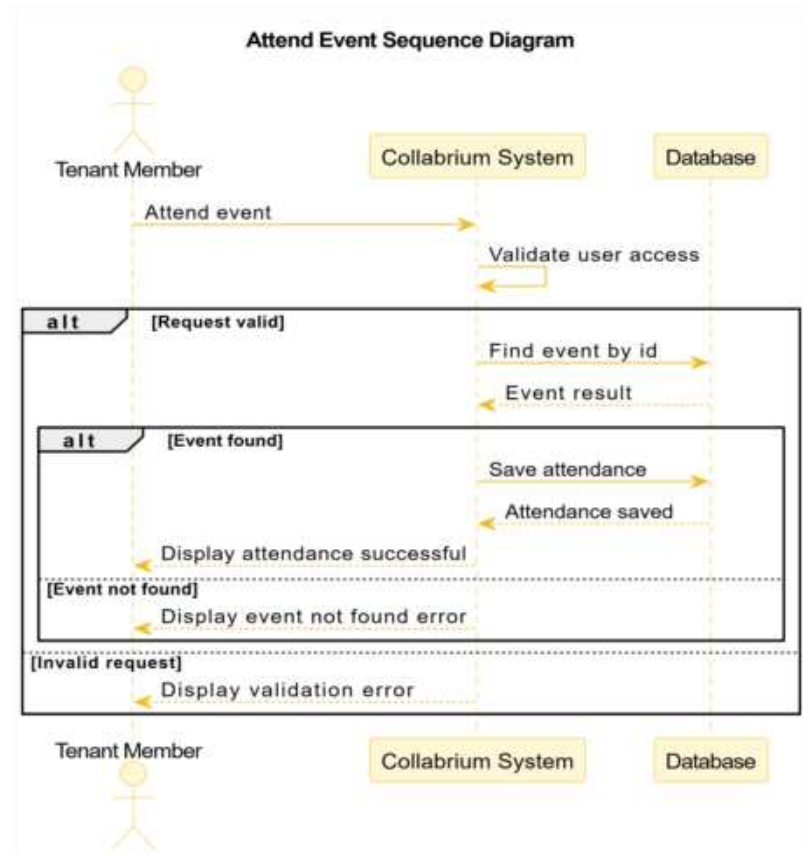


Figure 5.8.11 9 Attend Event System Sequence Diagram

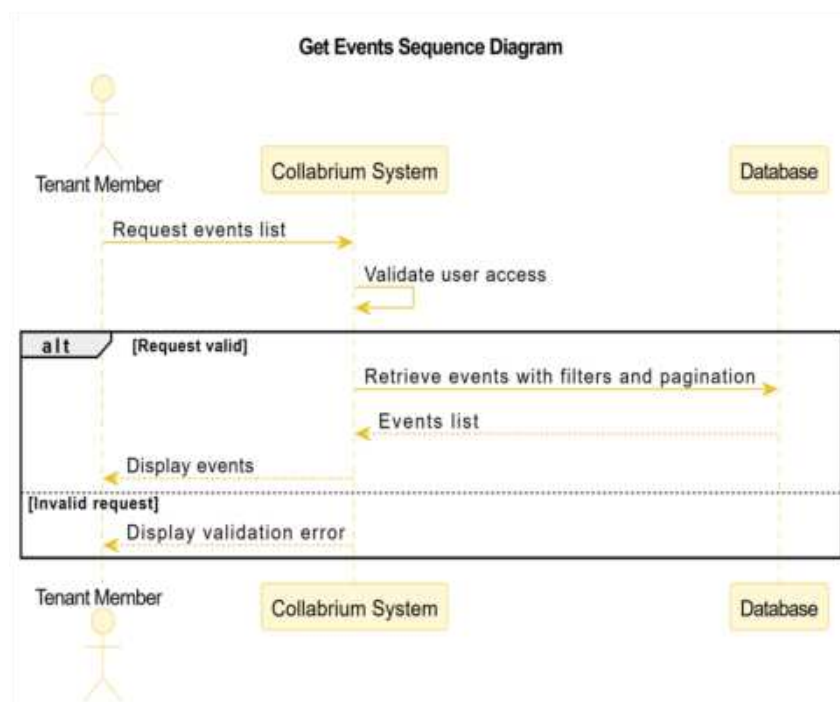


Figure 5.8.11 10 Get Events System Sequence Diagram

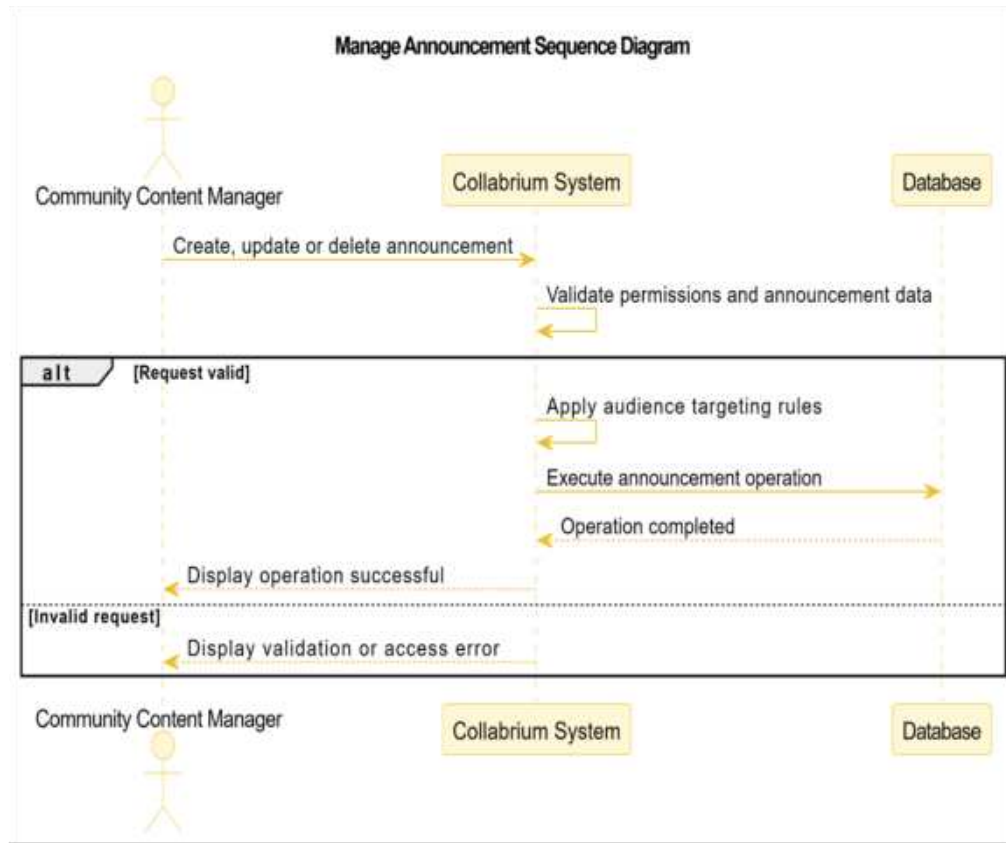


Figure 5.8.11 11 Manage Announcement System Sequence Diagram

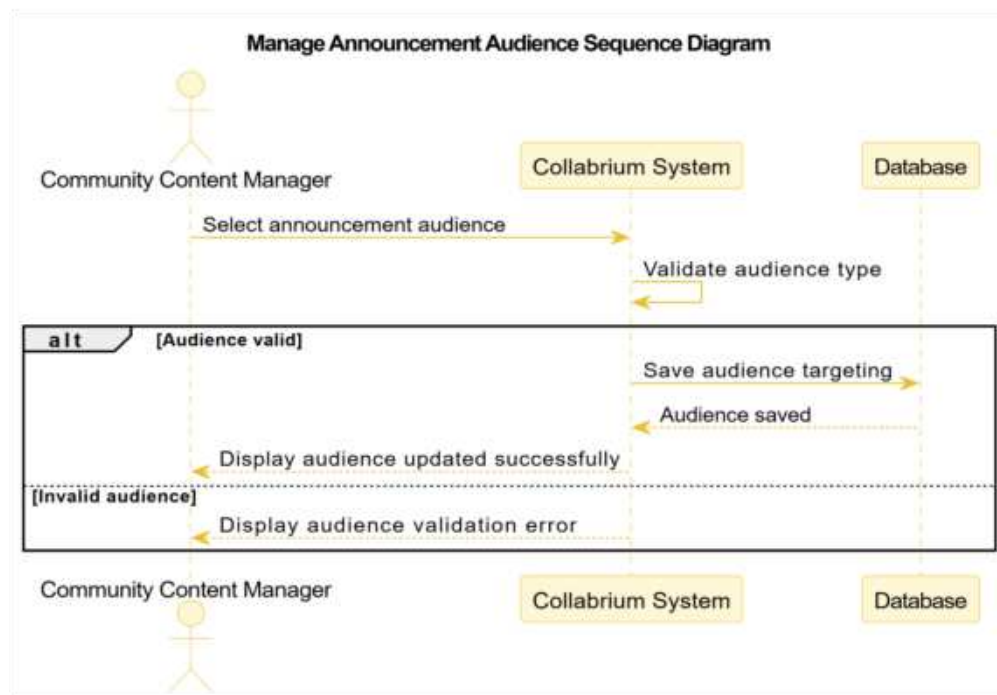


Figure 5.8.11 12 Manage Announcement Audience System Sequence Diagram

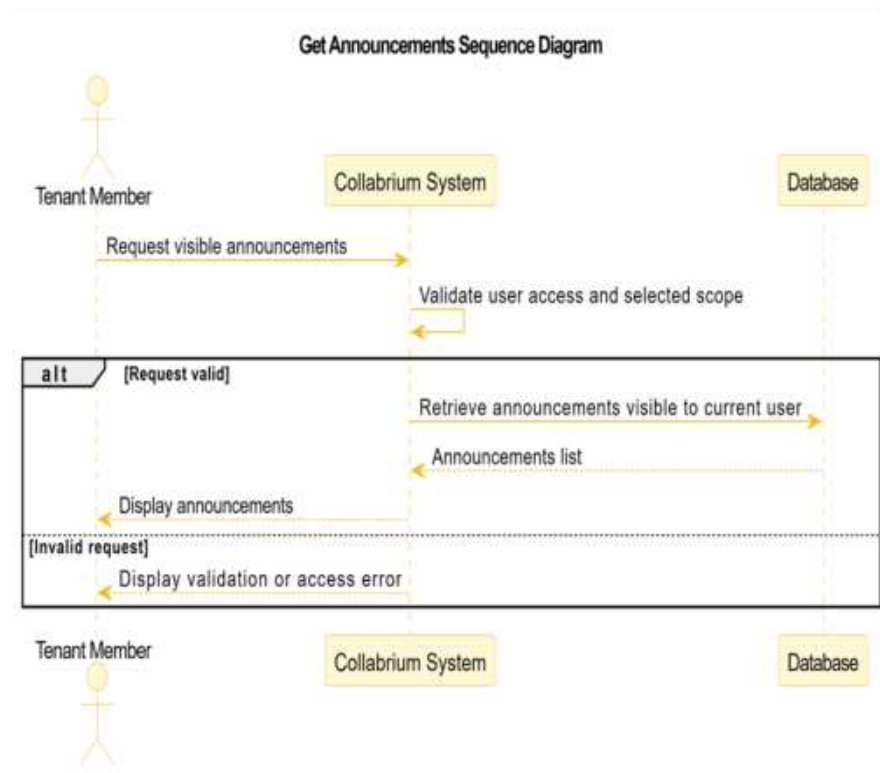


*Figure 5.8.11 13 Get Announcement System Sequence Diagram*

5.8.12 AI Subsystem System Sequence Diagram

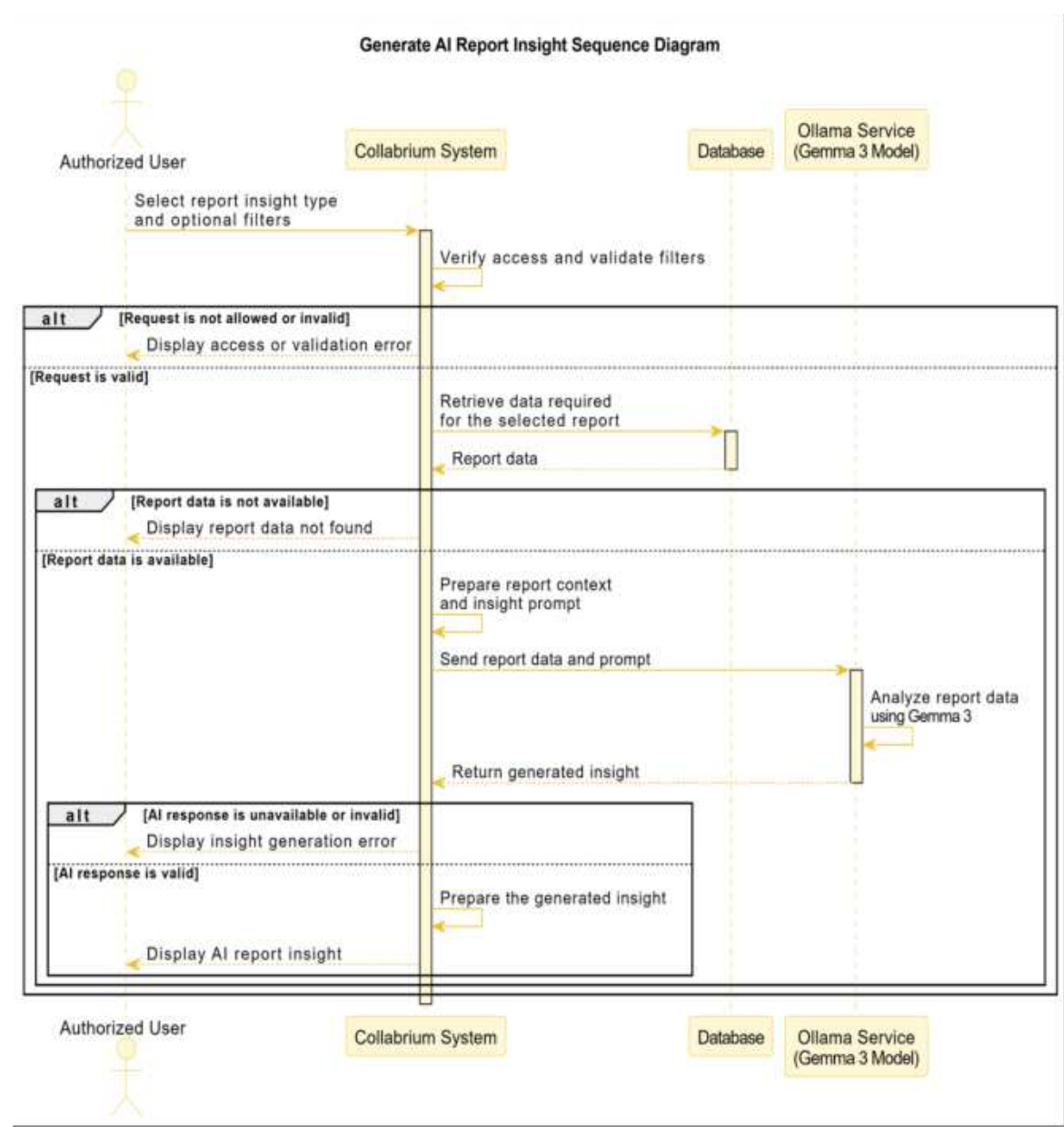


Figure 5.8.12 1 Generate AI Report System Sequence Diagram

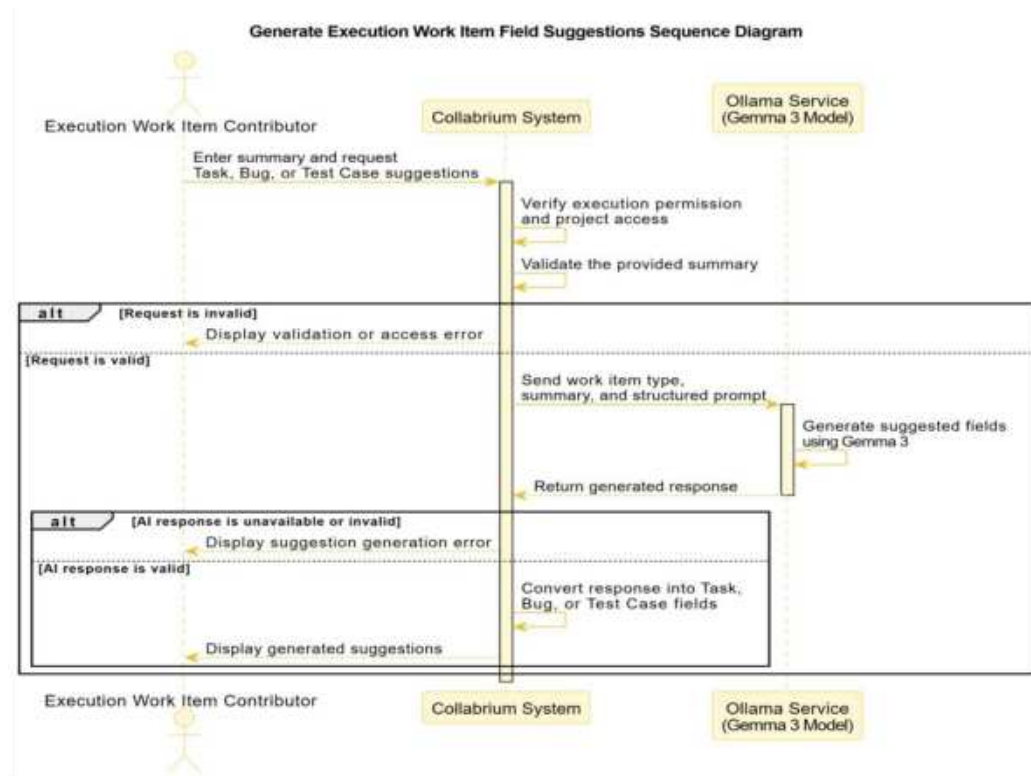


Figure 5.8.12 2 Generate Execution Work Item Field System Sequence Diagram

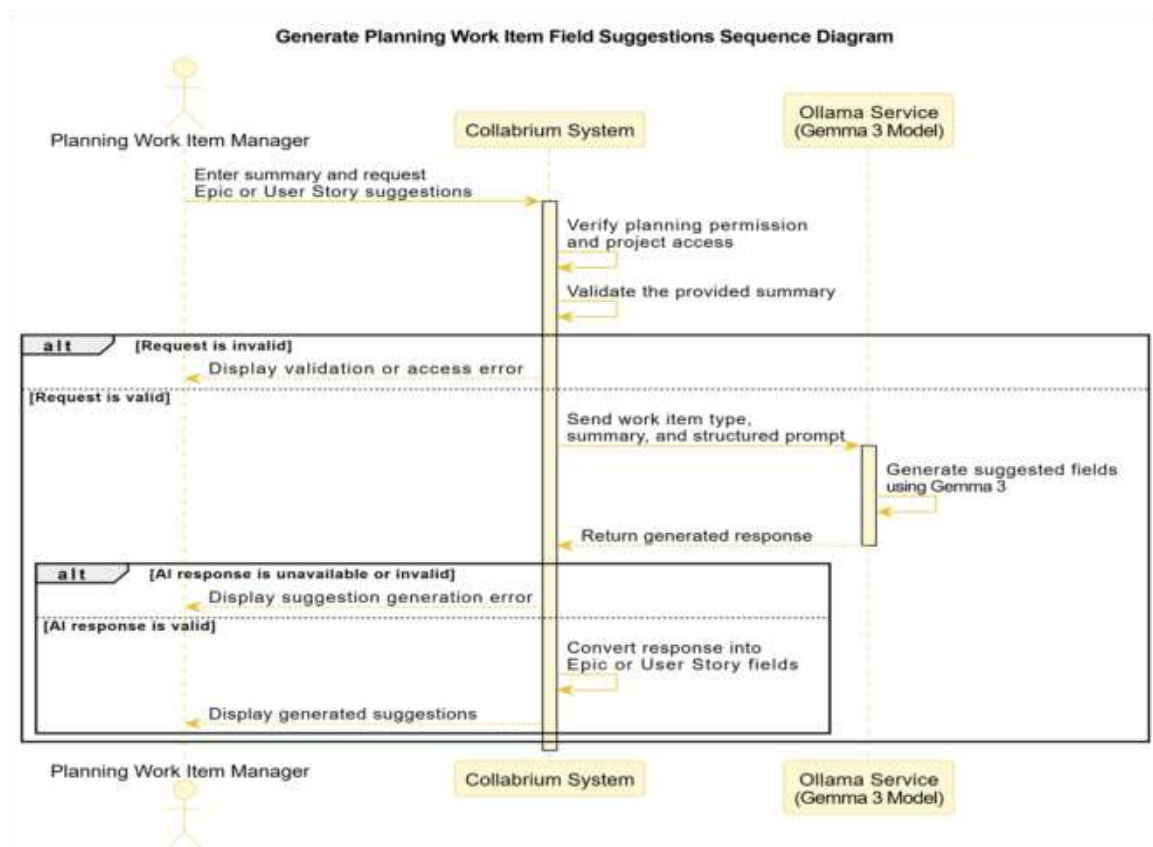


Figure 5.8.12 3 Generate Planning Work Item Field System Sequence Diagram

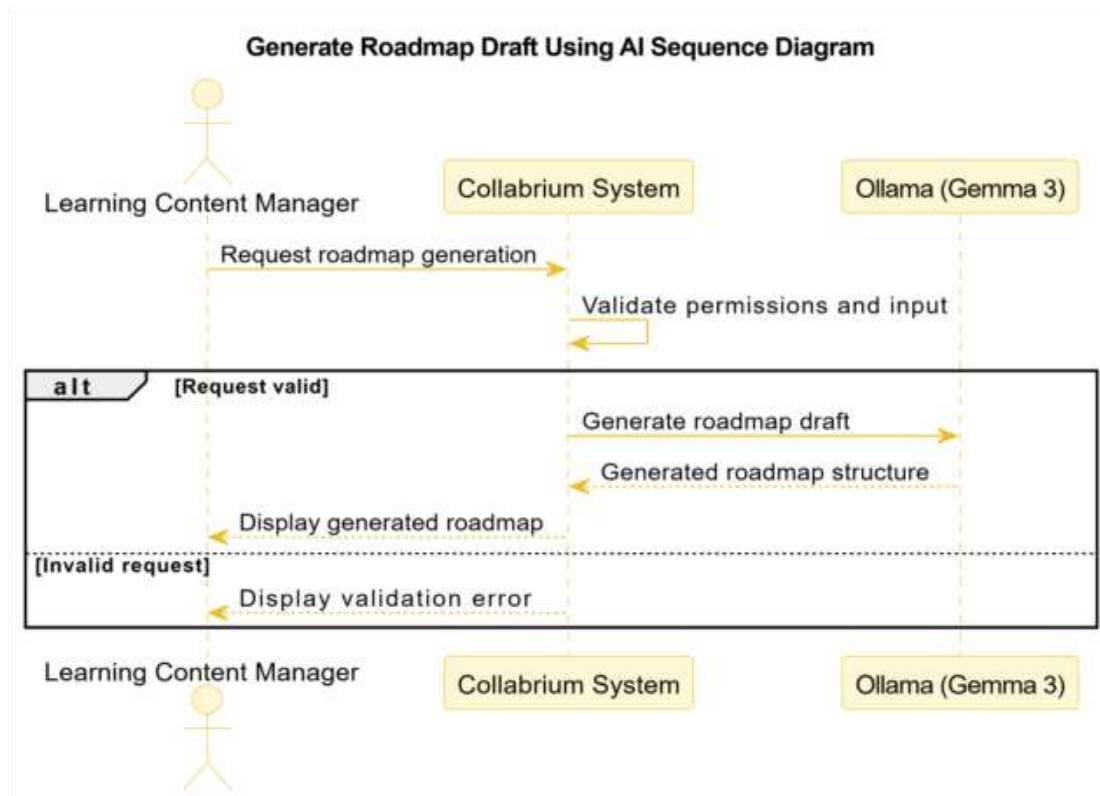


Figure 5.8.12 4 Generate Roadmap Draft Using AI Field System Sequence Diagram

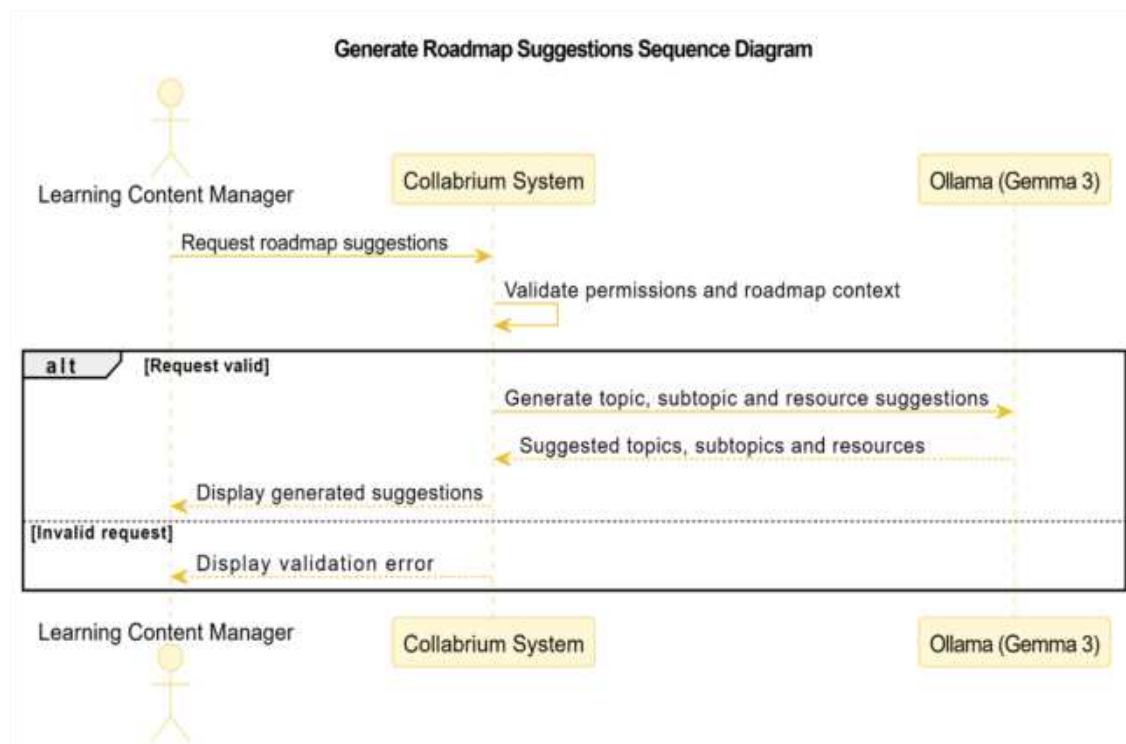


Figure 5.8.12 5 Generate Roadmap Suggestions Using AI Field System Sequence Diagram

5.8.13 Notification Subsystem System Sequence Diagram

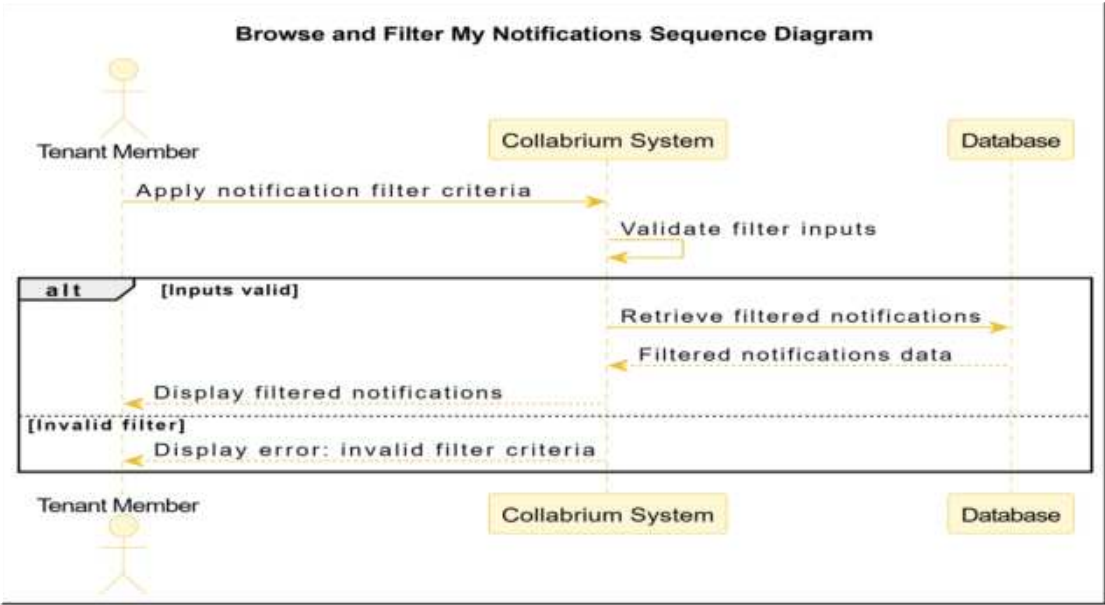


Figure 5.8.13 1 Browse and Filter My Notifications System Sequence Diagram

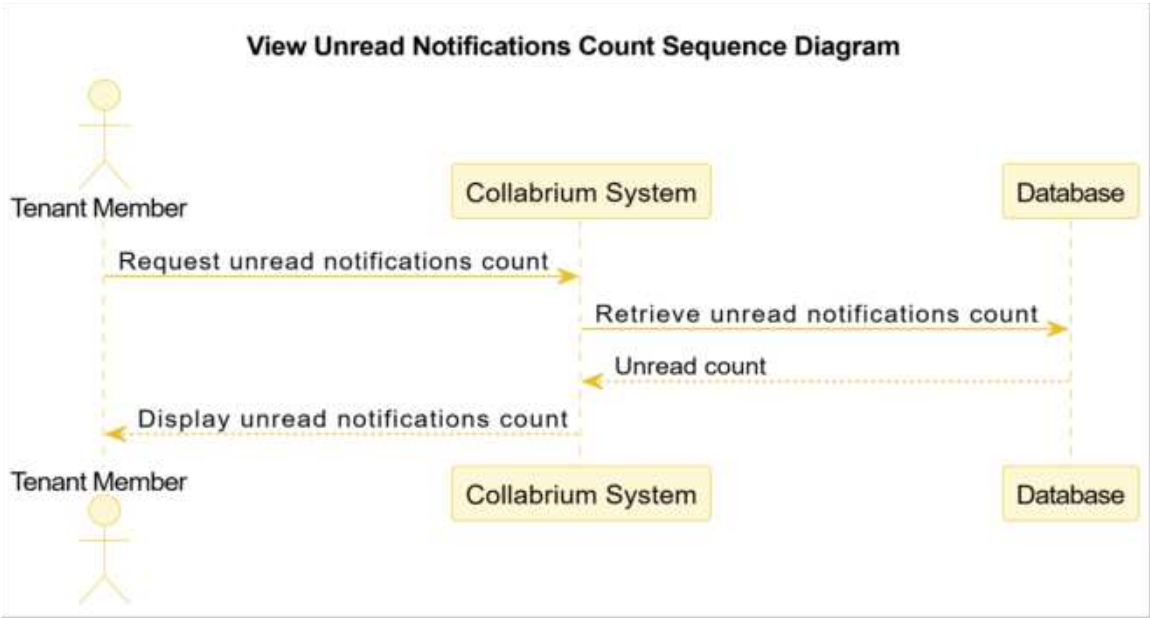


Figure 5.8.13 2 View Unread Notifications Count System Sequence Diagram

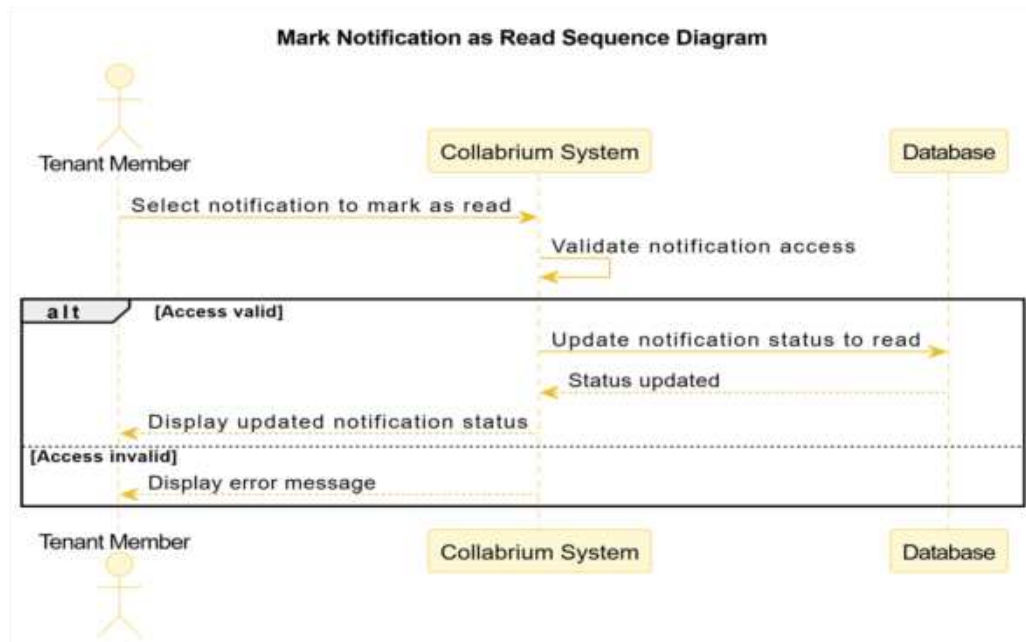


Figure 5.8.13 3 Mark Notification as Read System Sequence Diagram

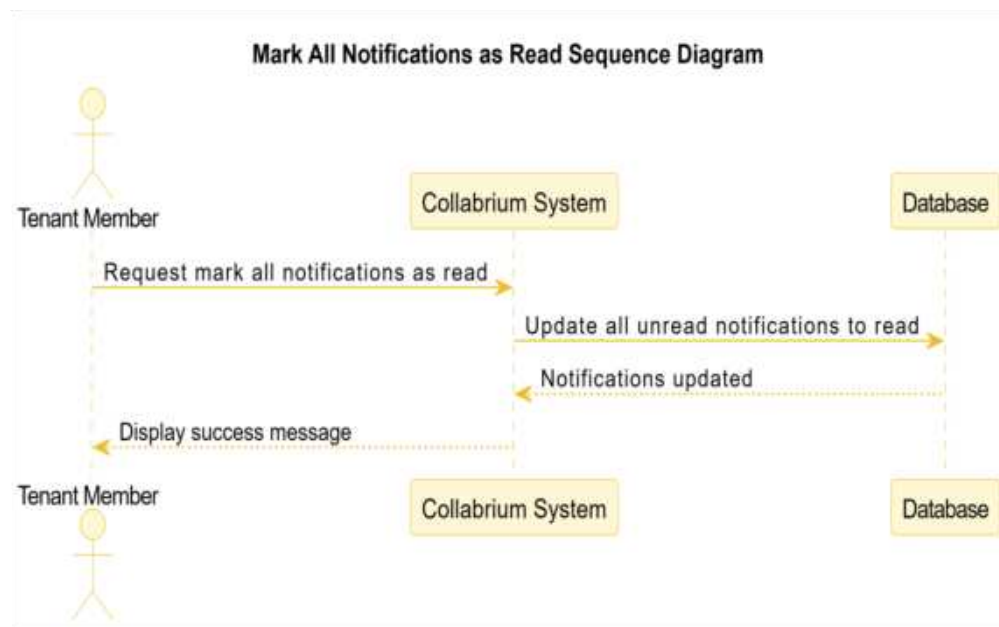


Figure 5.8.13 4 Mark All Notification As Read System Sequence Diagram

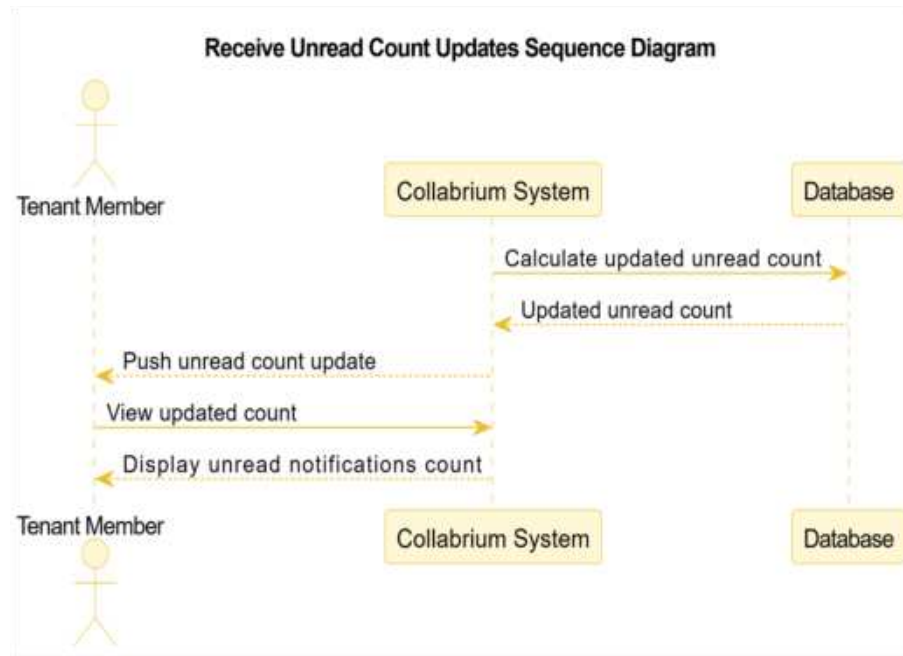


Figure 5.8.13 5 View Unread Count Notifications Update System Sequence Diagram

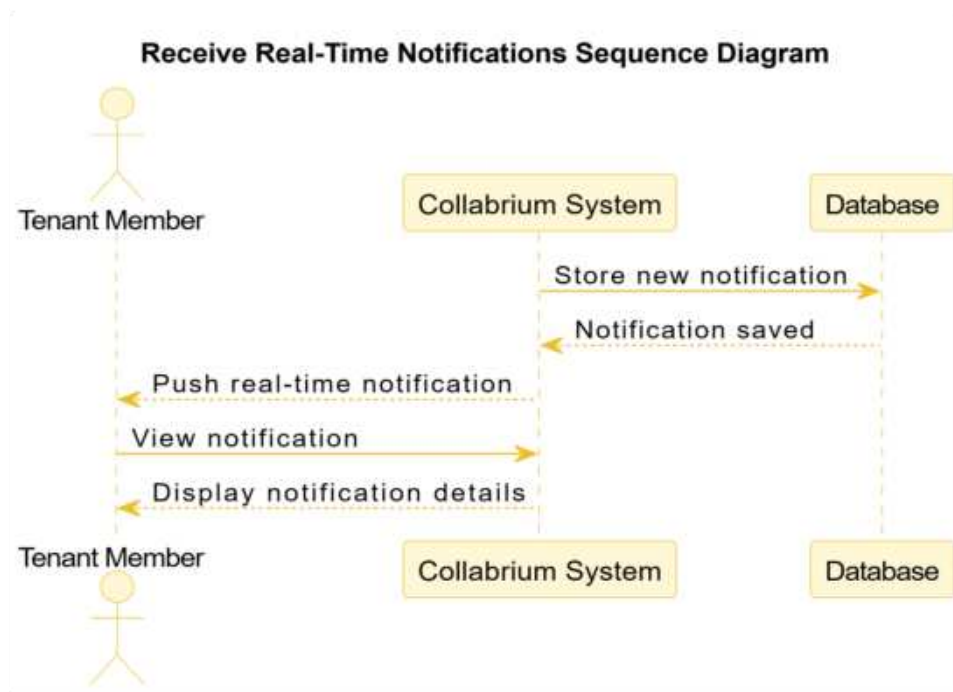


Figure 5.8.13 6 Receive Real-Time Notifications System Sequence Diagram

## CHAPTER SIX

### **USER INTERFACE**

This chapter presents the user interface design of the Collabrium platform. The chapter explains the design principles, visual identity, and the interfaces developed for the different system modules.

## 6.1 Design System

### 6.1.1 Visual Design and Color Scheme

#### Primary Color (#2A3F58)

The primary color was selected as the main visual identity of the system. This dark blue shade conveys professionalism, trust, and stability, making it suitable for enterprise collaboration platforms. It is primarily used in navigation bars, headers, and important interface elements.



#### Accent Color (#D4A72C)

The accent color serves as the highlight color throughout the system. The golden tone was chosen to create visual emphasis and guide user attention toward important actions such as buttons, active states, and notifications.



#### Secondary Gray (#9A9A9A)

This neutral gray color is used to reduce visual dominance and provide supporting information without distracting the user.



#### Background Color (FFFFFF)

White was selected as the primary background color to maximize readability and maintain a clean user interface.



# 6.2 System Interfaces

## 6.2.1 Authentication Subsystem Interfaces



Figure 6.2.1 1 Login Interface



Figure 6.2.1 2 Forgot Password Interface



Figure 6.2.1 3 Email Confirm Message Interface

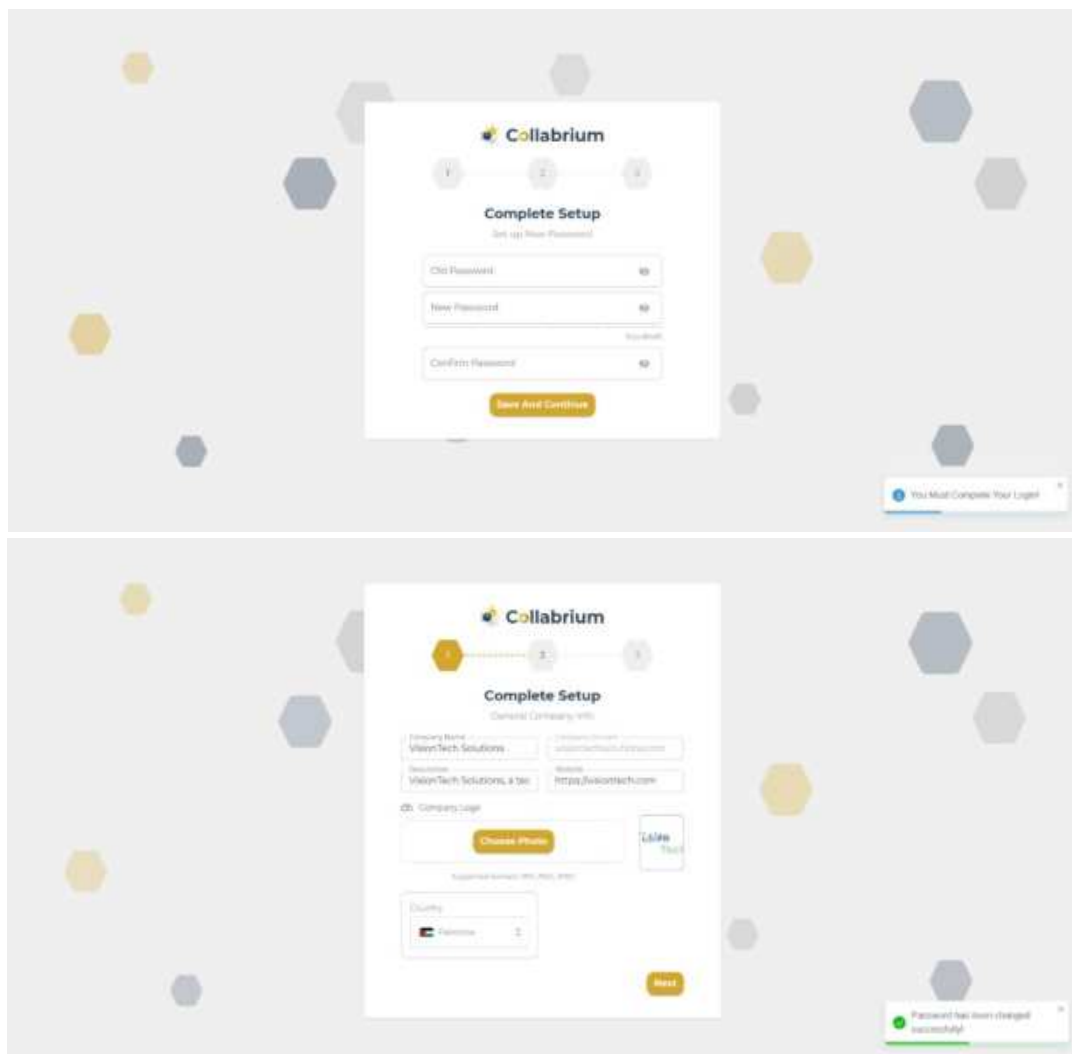


Figure 6.2.1 4 Complete Tenant Setup Interface

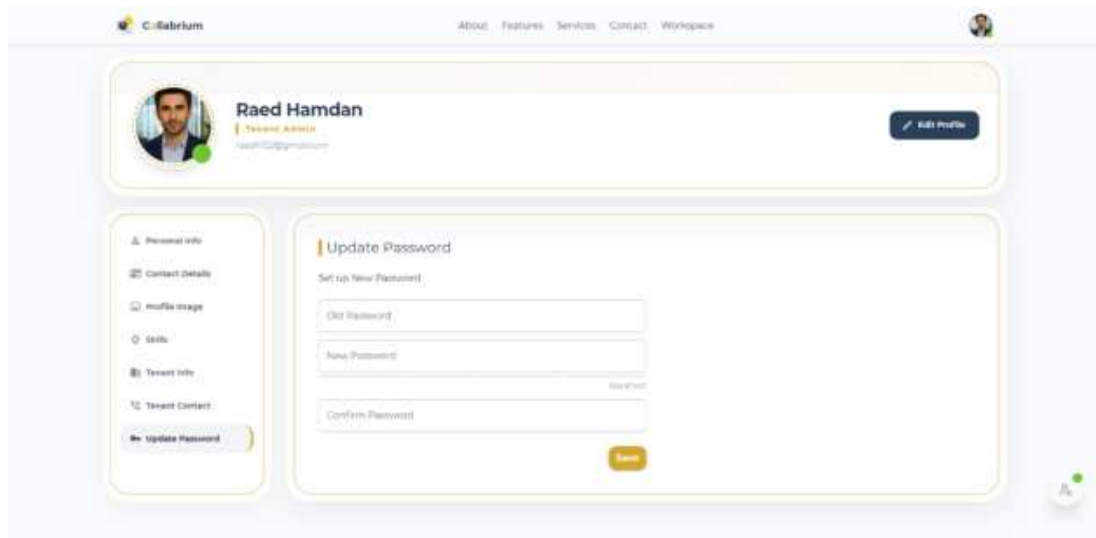


Figure 6.2.1 5 Reset Password Interface

## 6.2.2 Platform Services Interfaces

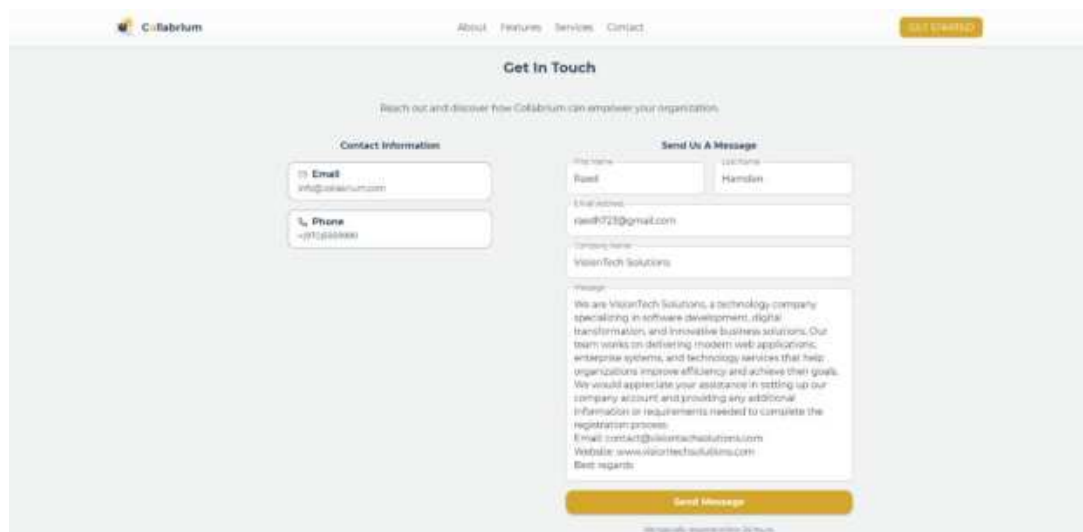


Figure 6.2.2 1 Send Contact Interface

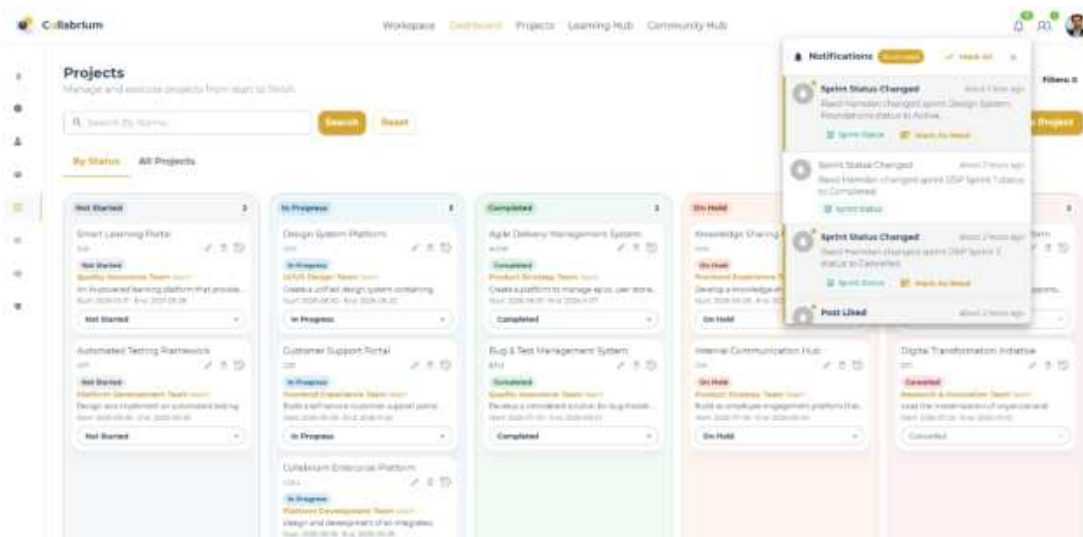


Figure 6.2.2 2 Notifications Dialog

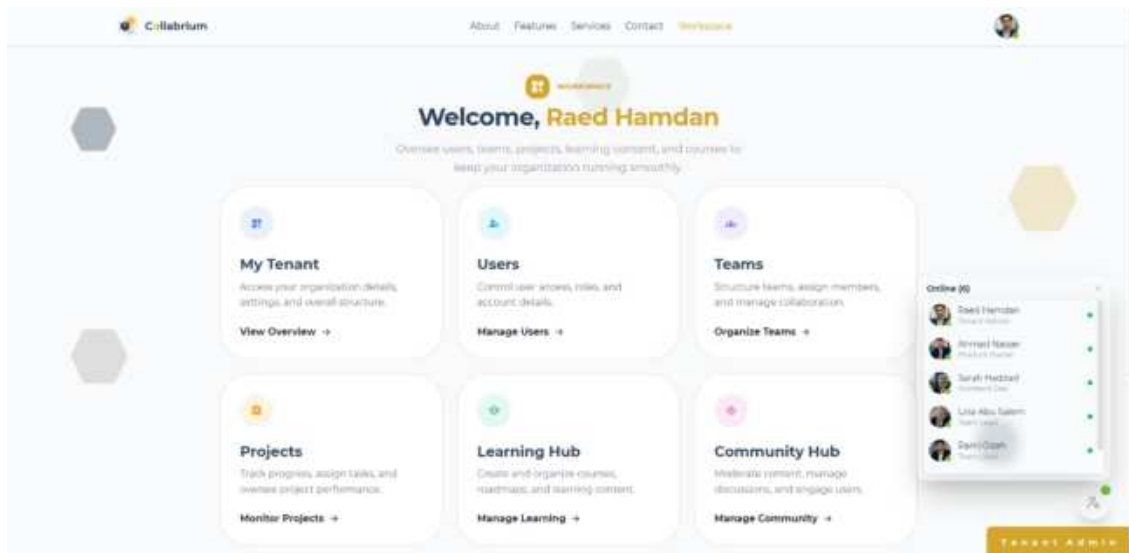


Figure 6.2.2 3 View Online Users Dialog

## 6.2.3 Tenant Interfaces

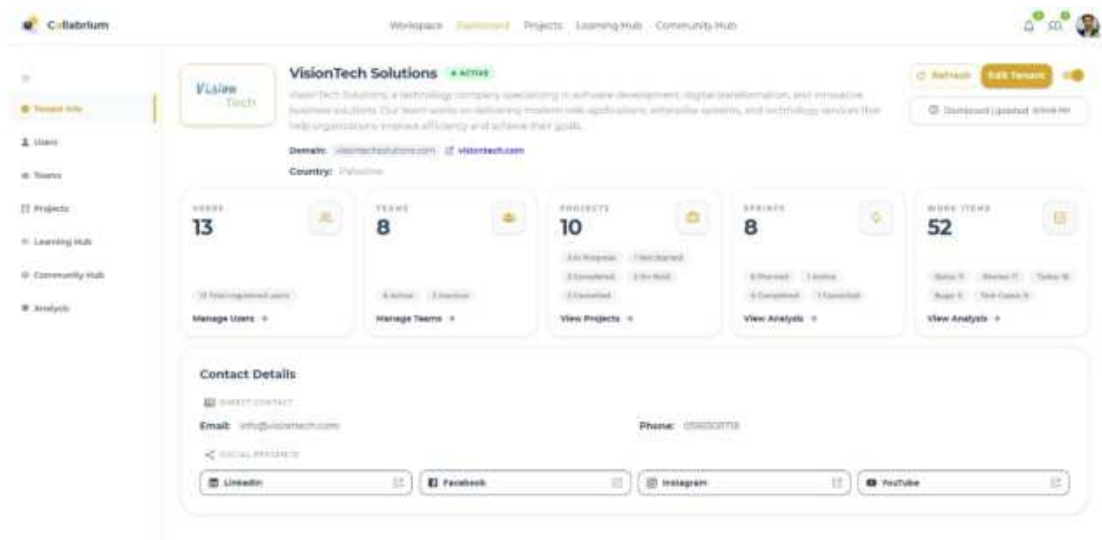


Figure 6.2.5 1 Tenant Main Dashboard

| Collabrium     |      | Workspace Dashboard      |                         |           |                                                         | Total: 6 Filter: 0                 |         |
|----------------|------|--------------------------|-------------------------|-----------|---------------------------------------------------------|------------------------------------|---------|
| Tenants        |      | Manage Tenants           |                         |           |                                                         |                                    |         |
| Search By Name |      | Search Reset             |                         |           |                                                         | Filter + Add Tenant                |         |
| #              | Logo | Name                     | Domain                  | Country   | Website                                                 | Description                        | Actions |
| 1              |      | Platform Root tenant     | global.platform         | Palestine | <a href="https://collabrium.com">collabrium.com</a>     | Global tenant for SuperAdmin a...  |         |
| 2              |      | TechFusion Ltd           | techfusion.com          | Palestine | <a href="https://techfusion.com">techfusion.com</a>     | Software development compan...     |         |
| 3              |      | AgileTech Systems        | agiletech.io            | Palestine | <a href="https://agiletech.io">agiletech.io</a>         | Enterprise software company fo...  |         |
| 4              |      | CloudSphere Technologies | cloudsphere.tech        | Palestine | <a href="https://cloudsphere.tech">cloudsphere.tech</a> | Cloud Infrastructure, DevOps a...  |         |
| 5              |      | NexusSoft Systems        | nexussoft.com           | Palestine | <a href="https://nexussoft.com">nexussoft.com</a>       | Custom software development, w...  |         |
| 6              |      | VisionTech Solutions     | visiontechsolutions.com | Palestine | <a href="https://visiontech.com">visiontech.com</a>     | VisionTech Solutions, a technol... |         |
| No more...     |      |                          |                         |           |                                                         |                                    |         |

Figure 6.2.3 1 Super Admin- Tenants Table

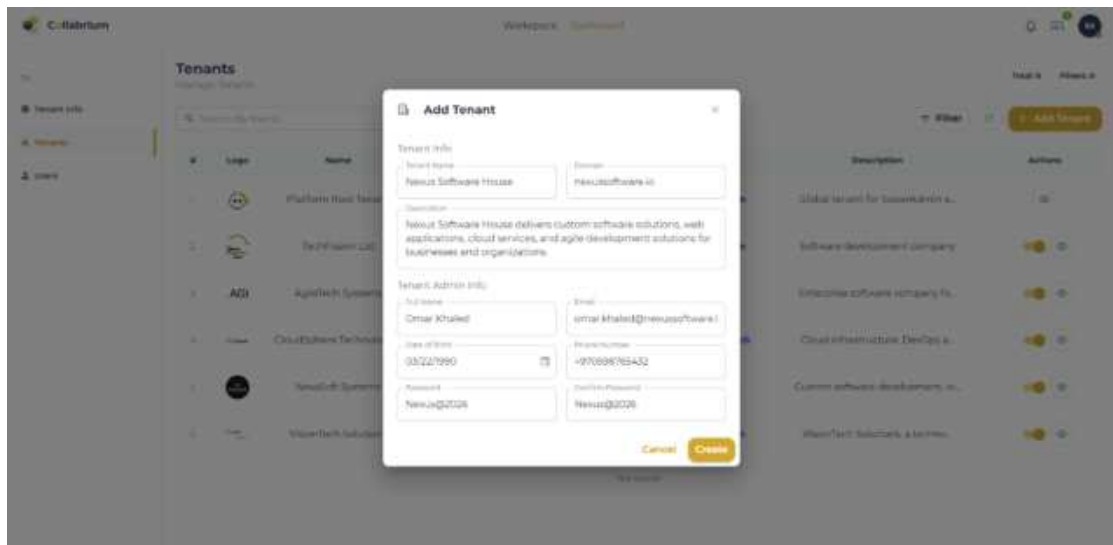


Figure 6.2.3 2 Super Admin- Add Tenant Dialog

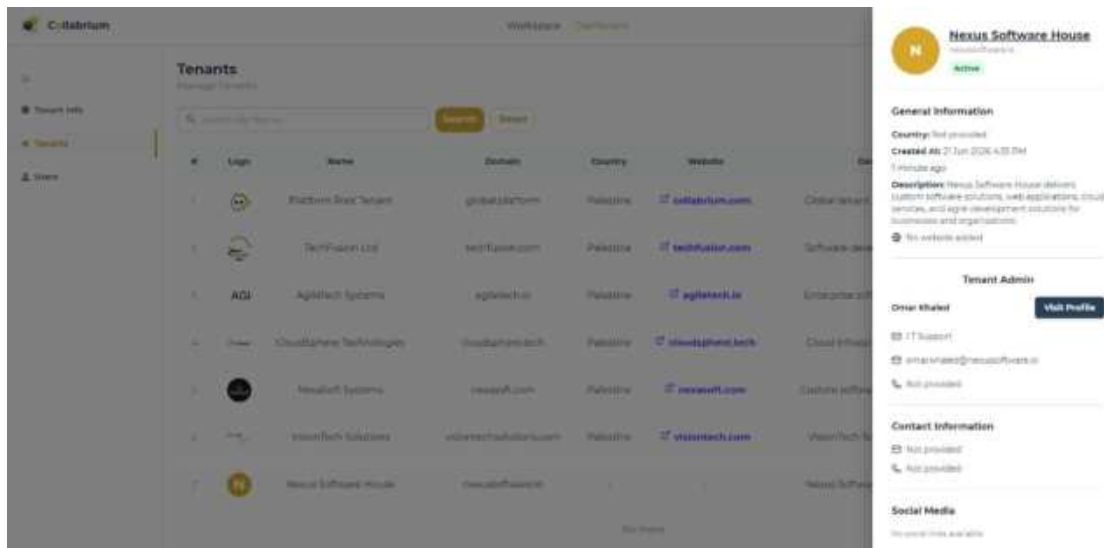


Figure 6.2.3 3 Super Admin- Tenant Details Drawer

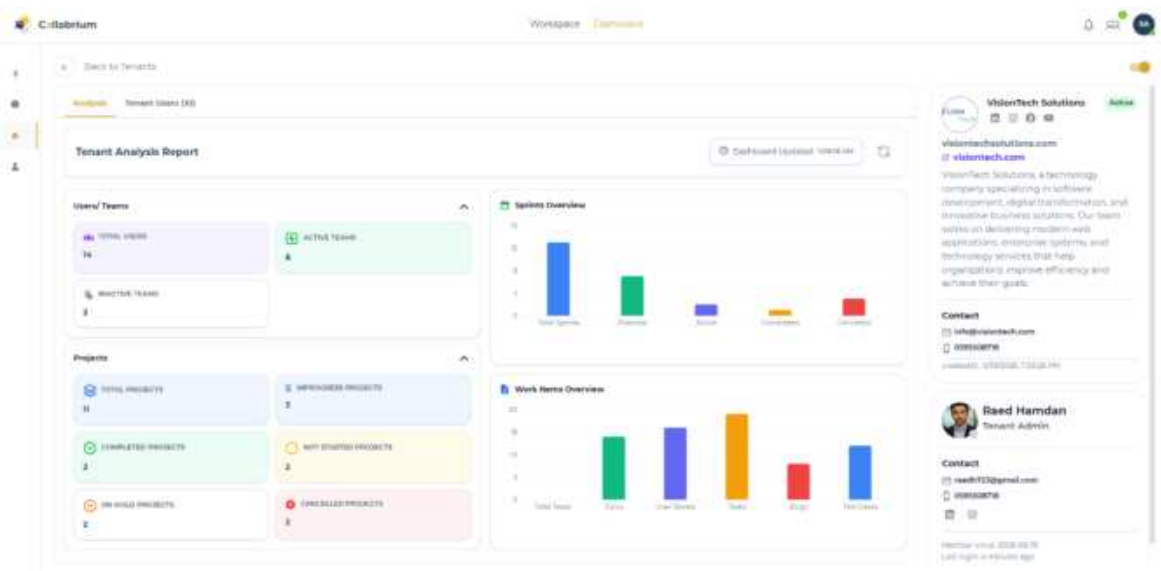


Figure 6.2.3 4 Super Admin- Tenant Profile Interface

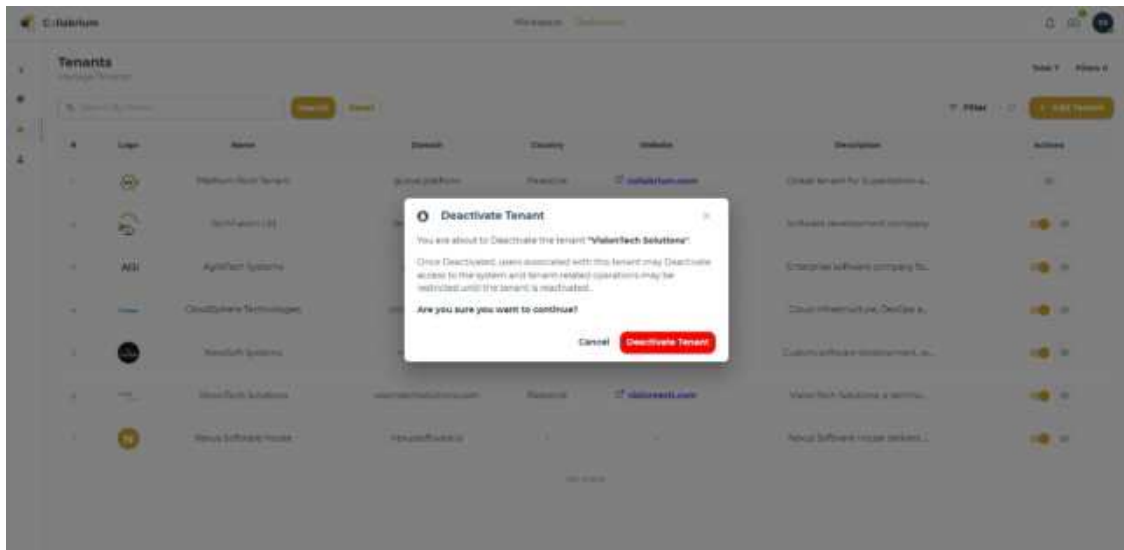


Figure 6.2.3 5 Super Admin- Activate & Deactivate Tenant Dialog

## 6.2.4 Users Interfaces

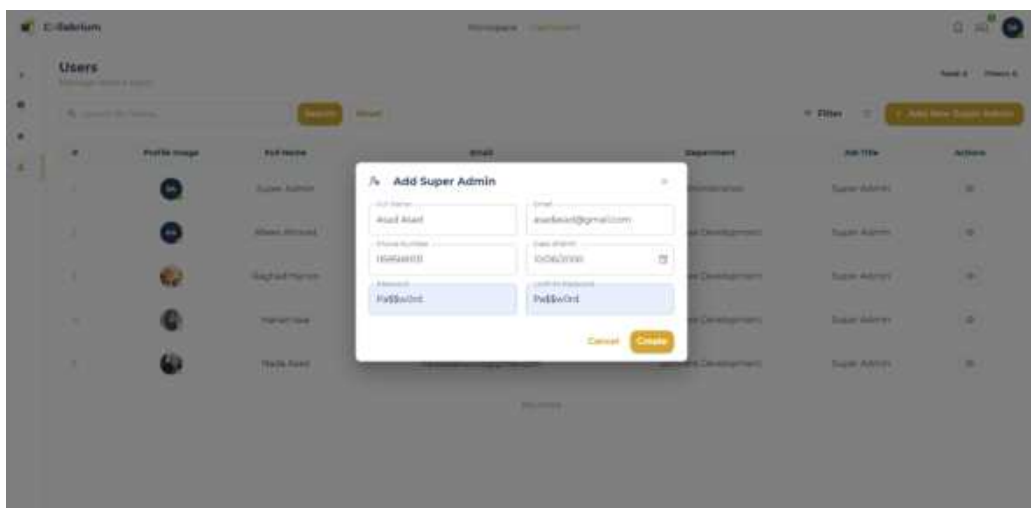


Figure 6.2.4 1 Super Admin- Add New Super Admin Dialog

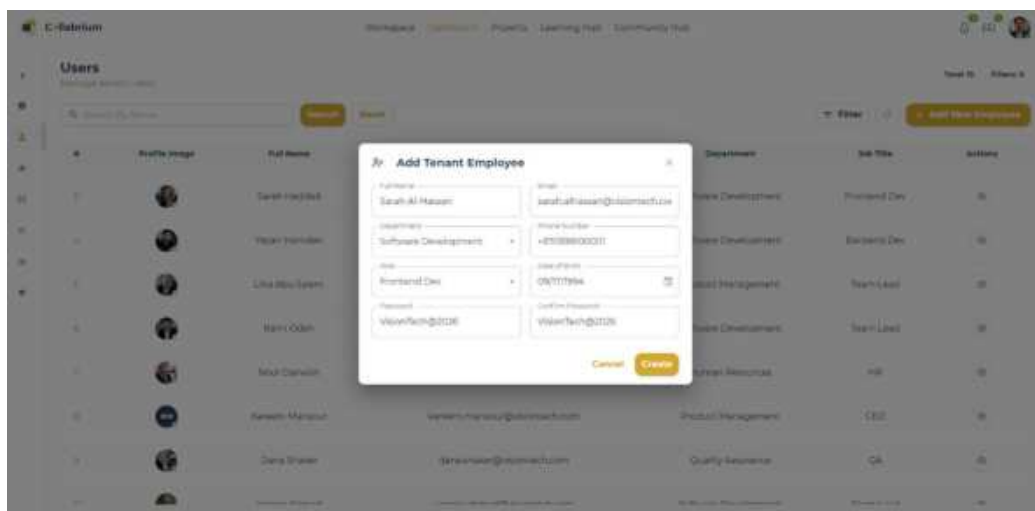


Figure 6.2.4 2 Tenant Manager- Add New Tenant Member Dialog

| #  | Profile Image | Full Name        | Email                           | Department           | Job Title    | Actions |
|----|---------------|------------------|---------------------------------|----------------------|--------------|---------|
| 1  |               | Sarah Haddad     | sarah.haddad@visiortech.com     | Software Development | Frontend Dev |         |
| 4  |               | Yazan Hamdan     | yazan.hamdan@visiortech.com     | Software Development | Backend Dev  |         |
| 7  |               | Lina Abu Salem   | lina.abusalem@visiortech.com    | Product Management   | Team Lead    |         |
| 8  |               | Sami Odeh        | sami.odeh@visiortech.com        | Software Development | Team Lead    |         |
| 9  |               | Noor Darwish     | noor.darwish@visiortech.com     | Human Resources      | HR           |         |
| 10 |               | Kareem Mansour   | kareem.mansour@visiortech.com   | Product Management   | CEO          |         |
| 11 |               | Dana Shaker      | dana.shaker@visiortech.com      | Quality Assurance    | QA           |         |
| 12 |               | Youssef Alkhatib | youssef.alkhatib@visiortech.com | Software Development | Team Lead    |         |

Figure 6.2.4 3 Tenant Users Table

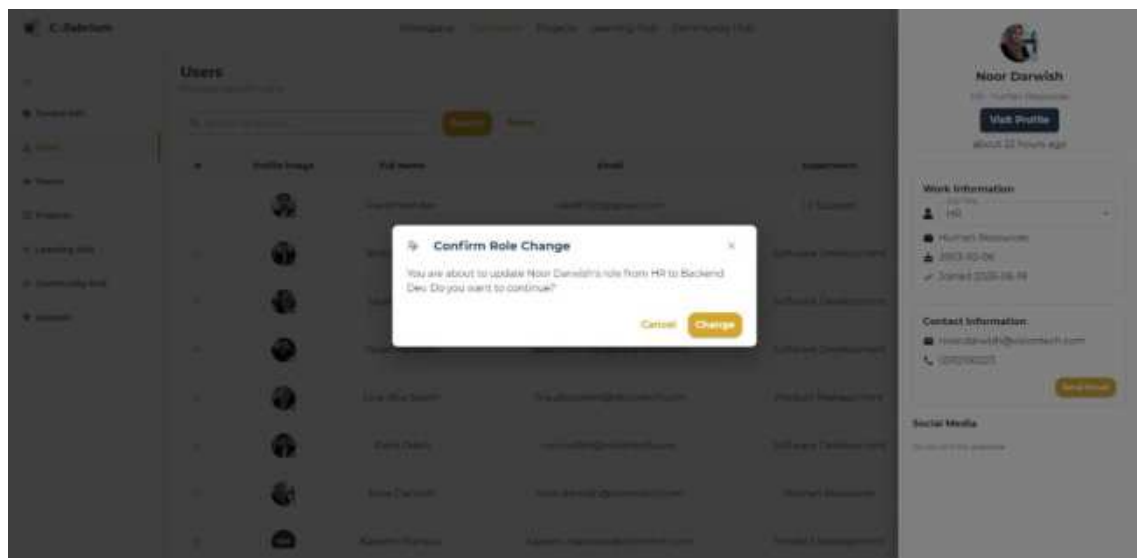


Figure 6.2.4 4 Change User Role Dialog

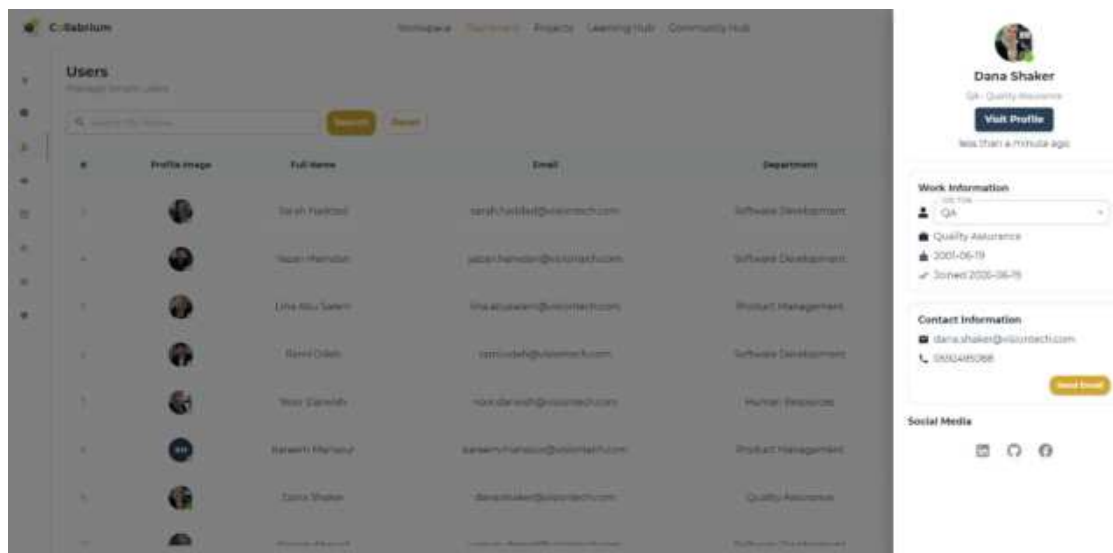


Figure 6.2.4 5 User Details Drawer

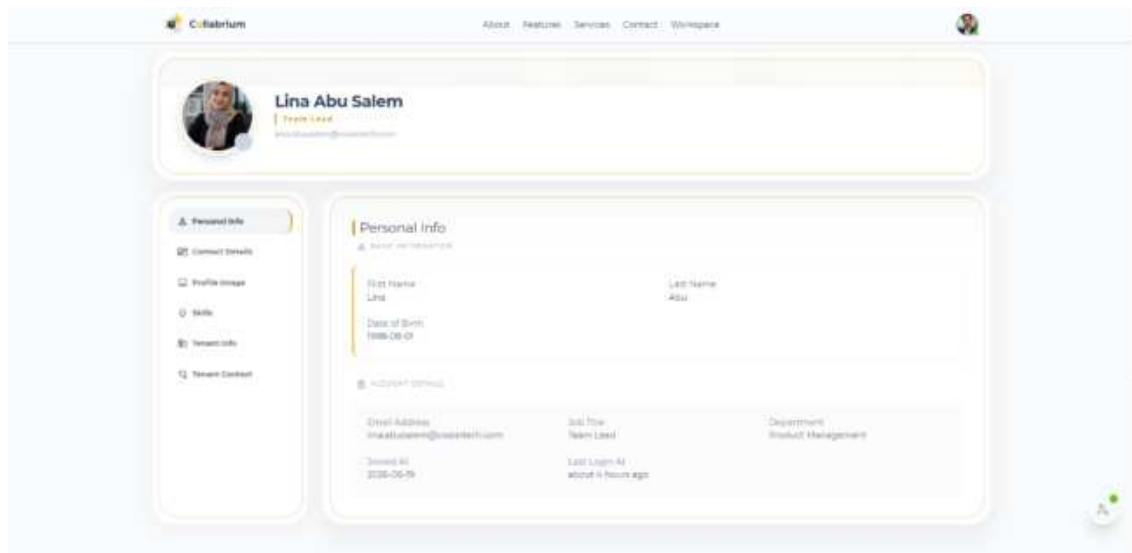


Figure 6.2.4 6 User Profile Interface

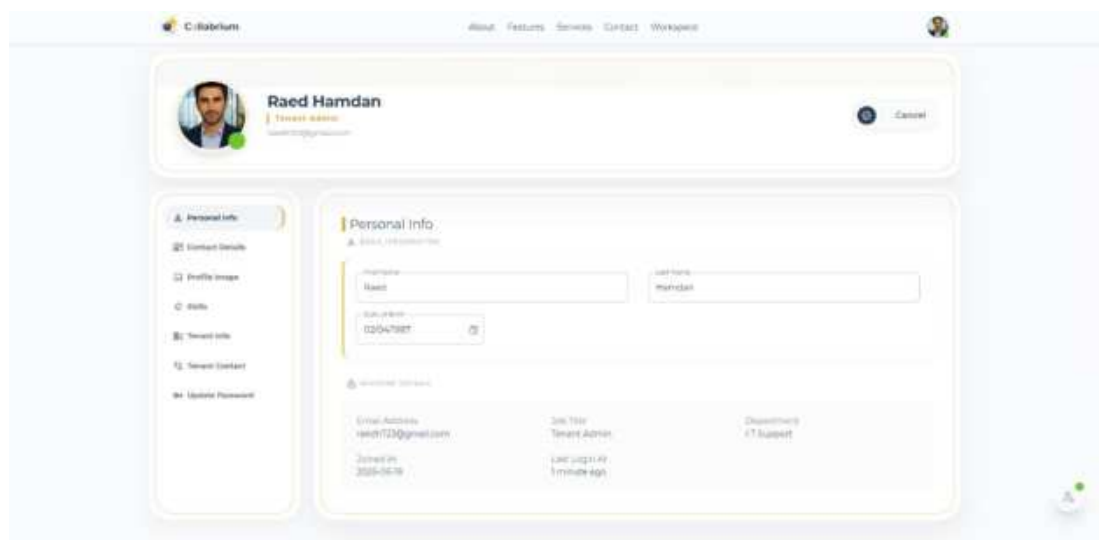


Figure 6.2.4 7 Edit Profile Interface

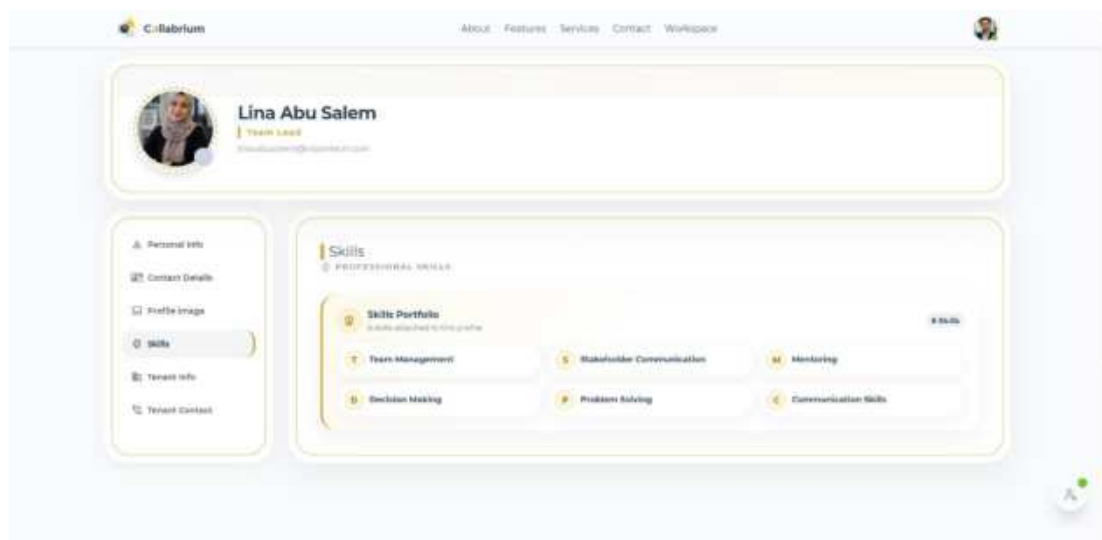


Figure 6.2.4 8 View User Skills

6.2.5 Teams Interfaces

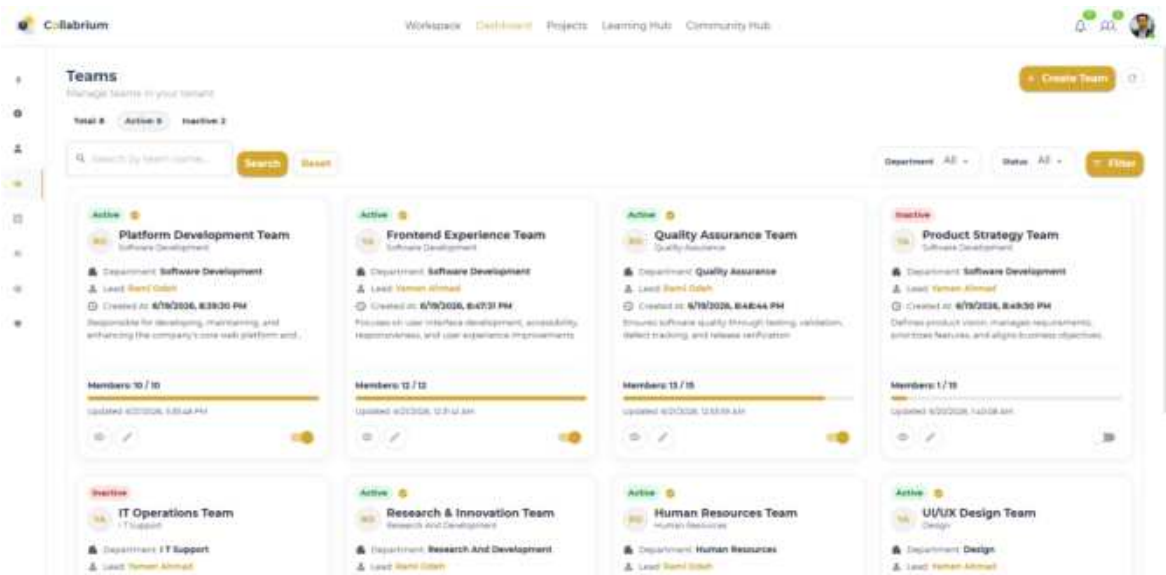


Figure 6.2.5 2 Teams Board

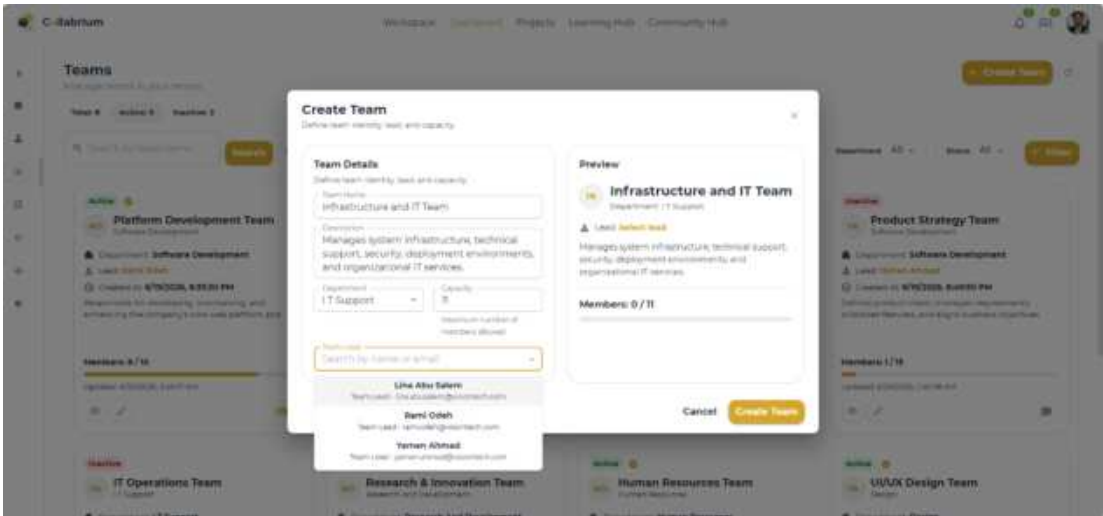


Figure 6.2.5 3 Tenant Manager- Create & Edit Team Dialog

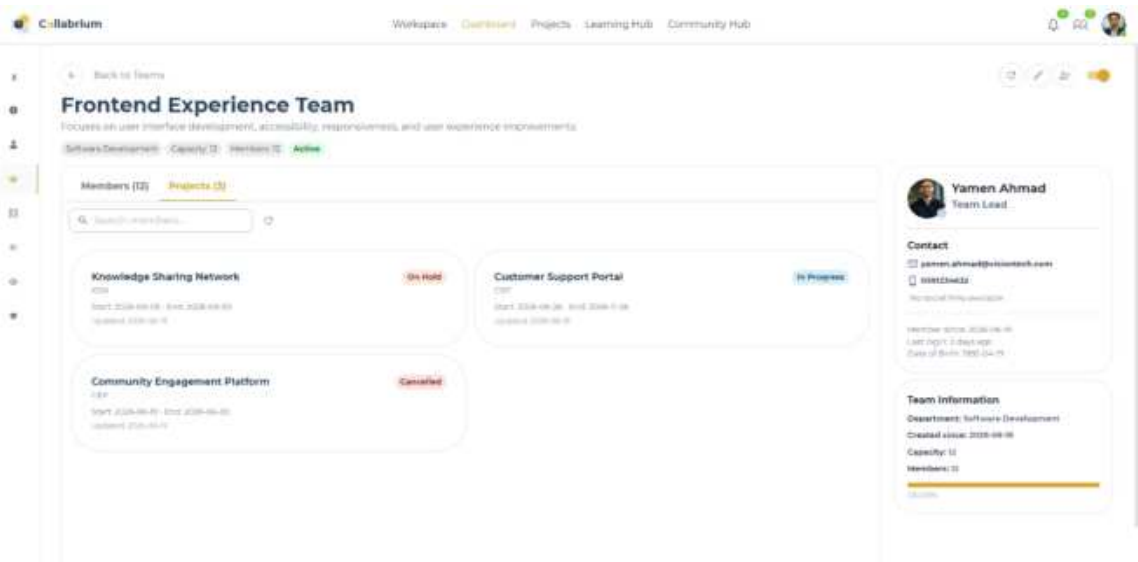


Figure 6.2.5 4 Team Details Page

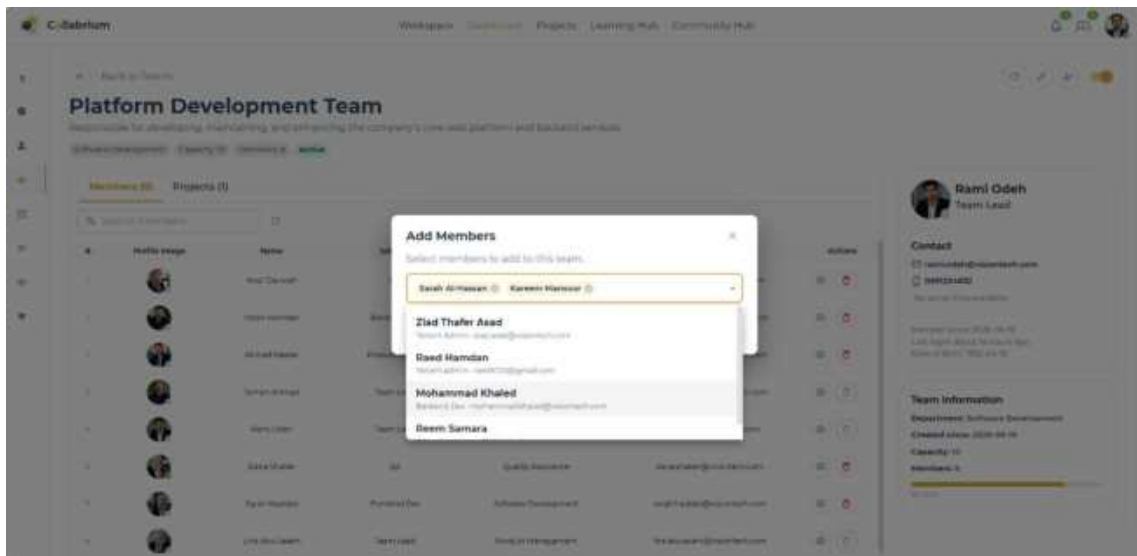


Figure 6.2.5 5 Tenant Manager- Add Team Member Dialog

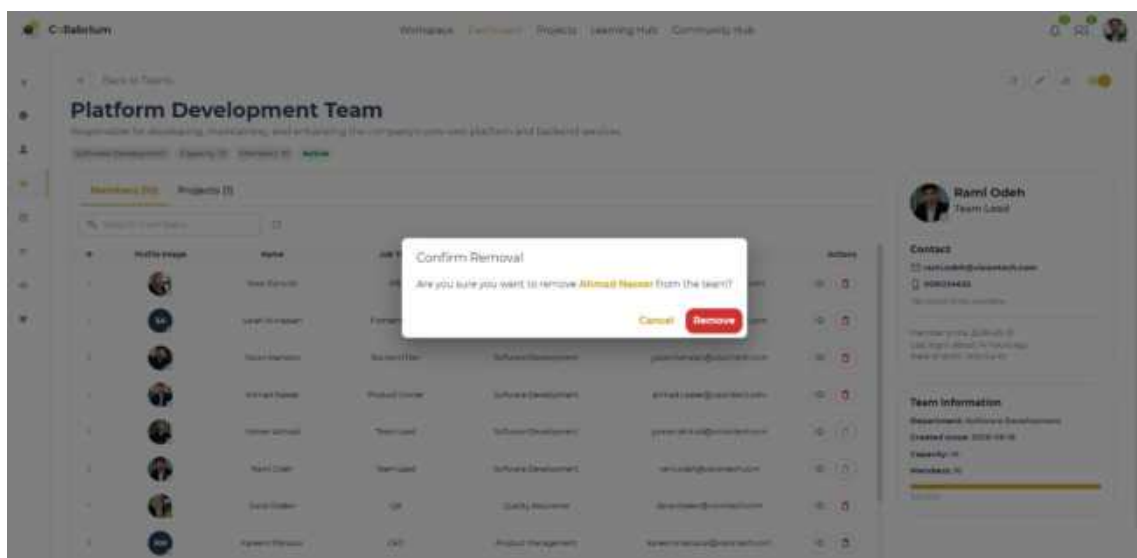


Figure 6.2.5 6 Tenant Manager- Remove Team Member Dialog

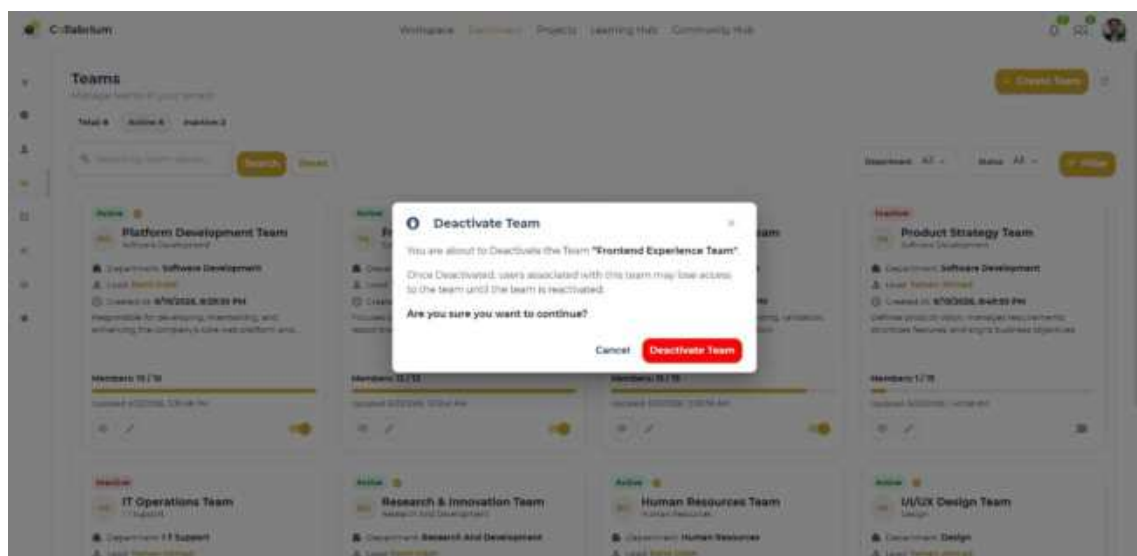


Figure 6.2.5 7 Tenant Manager- Activate & Deactivate Team Dialog

### 6.2.6 Projects Interfaces

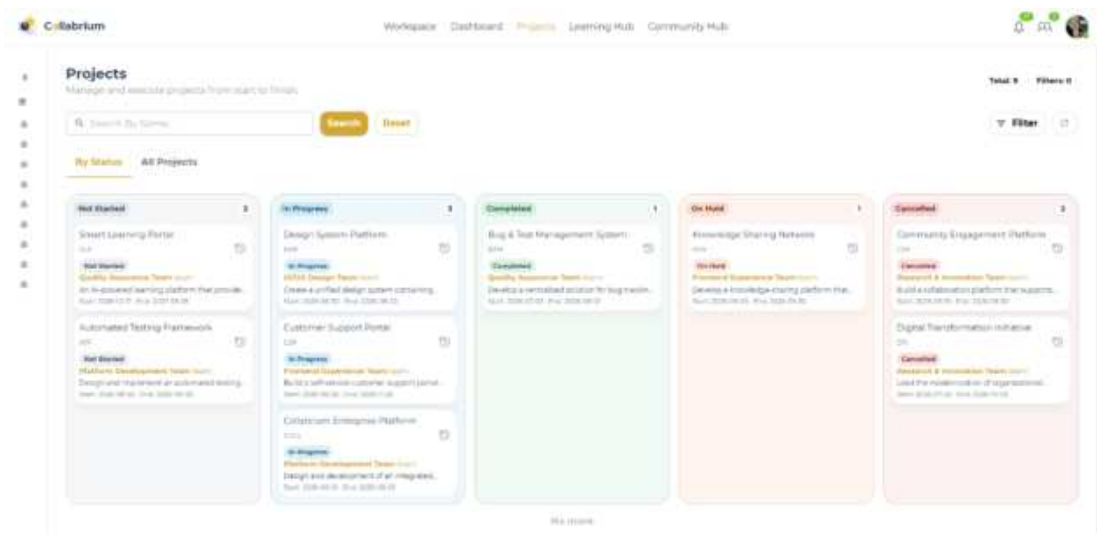


Figure 6.2.6 1 User Projects Board - Card View

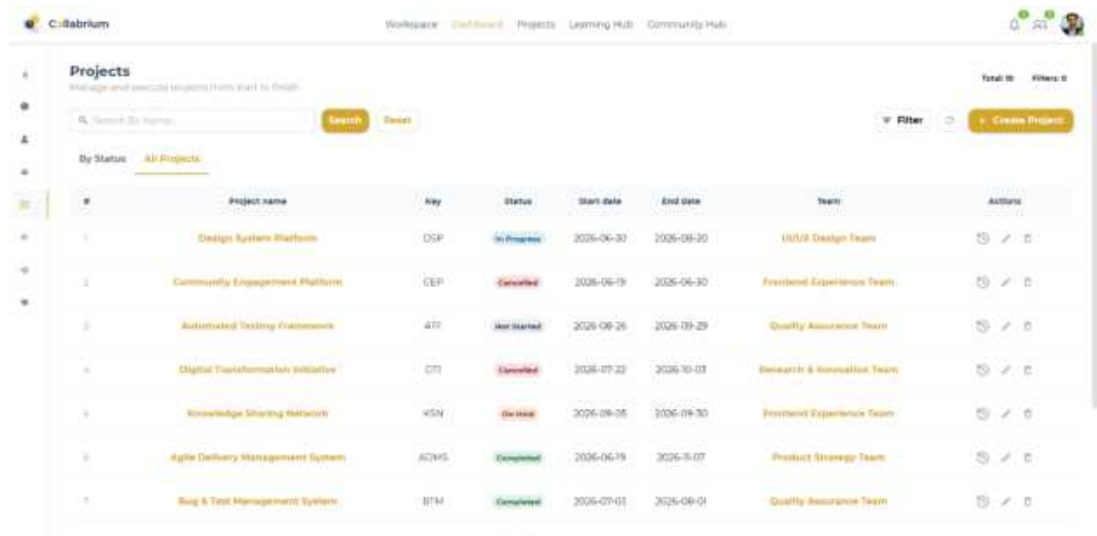


Figure 6.2.6 2 User Projects Board- Table View

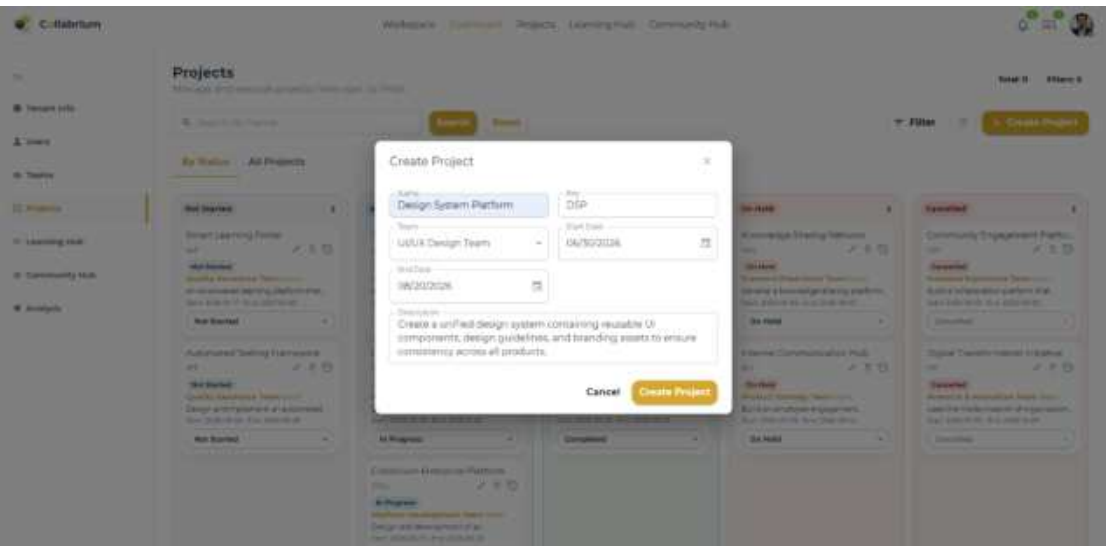


Figure 6.2.6 3 Tenant Manager- Create Project Dialog

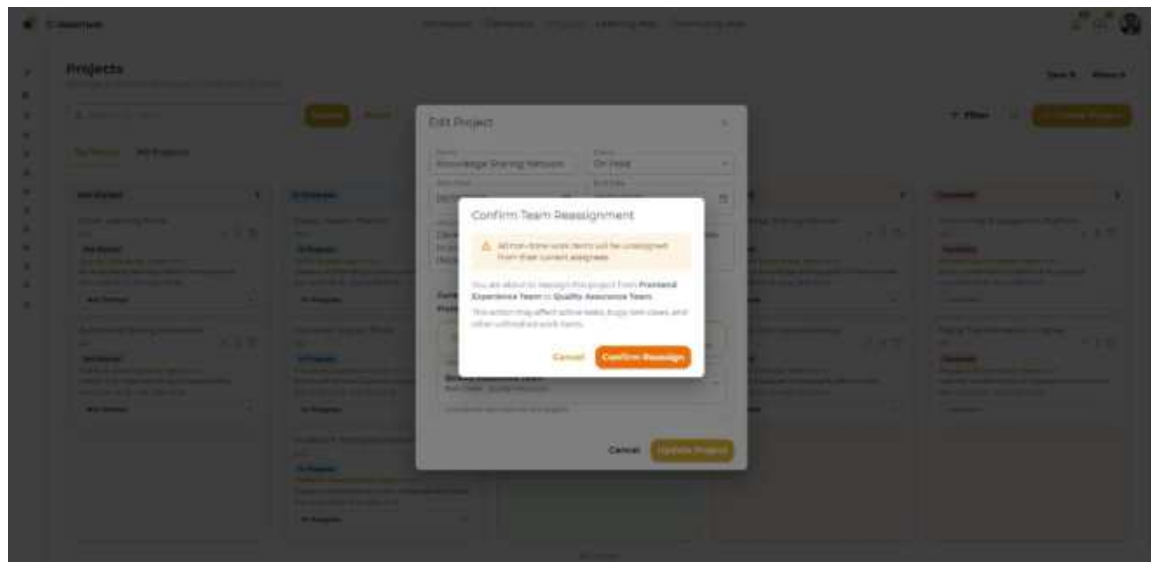


Figure 6.2.6 4 Tenant Manager- Confirm Reassign Team To Project

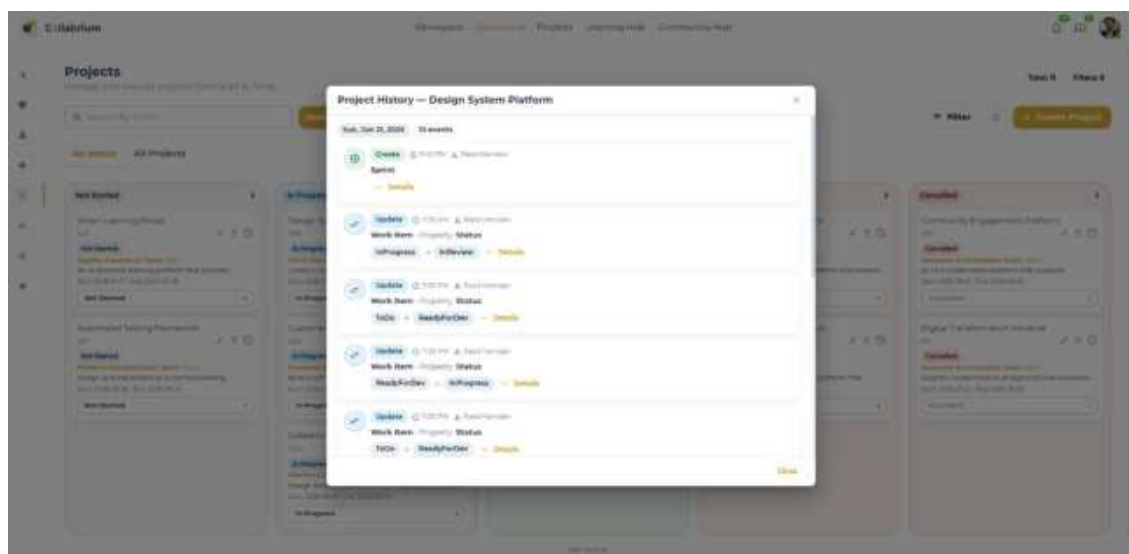


Figure 6.2.6 5 Project History Dialog

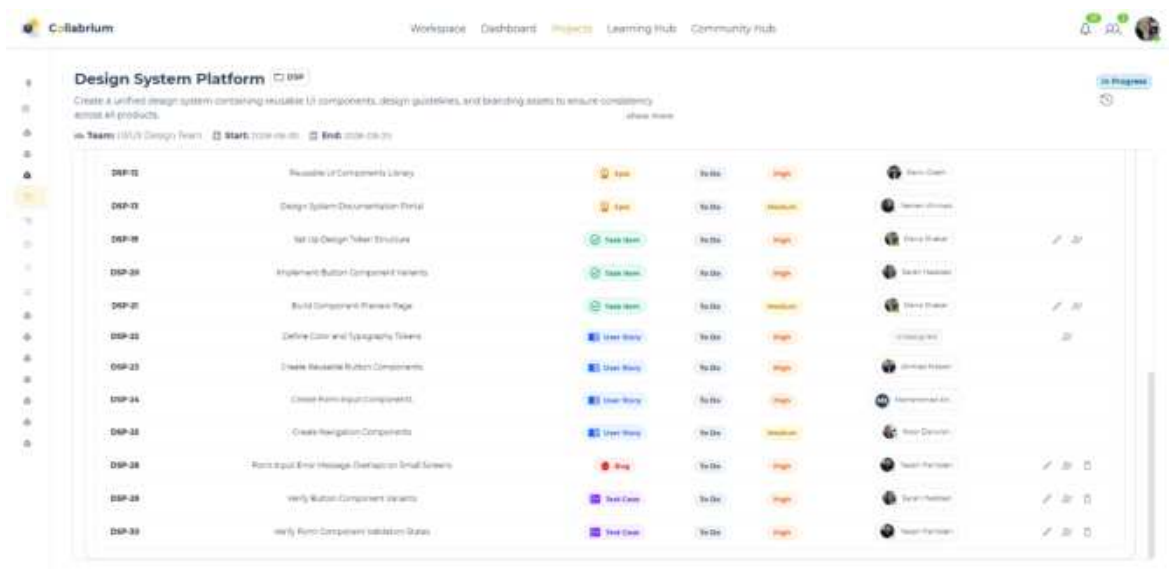


Figure 6.2.6 6 Project Backlog Board

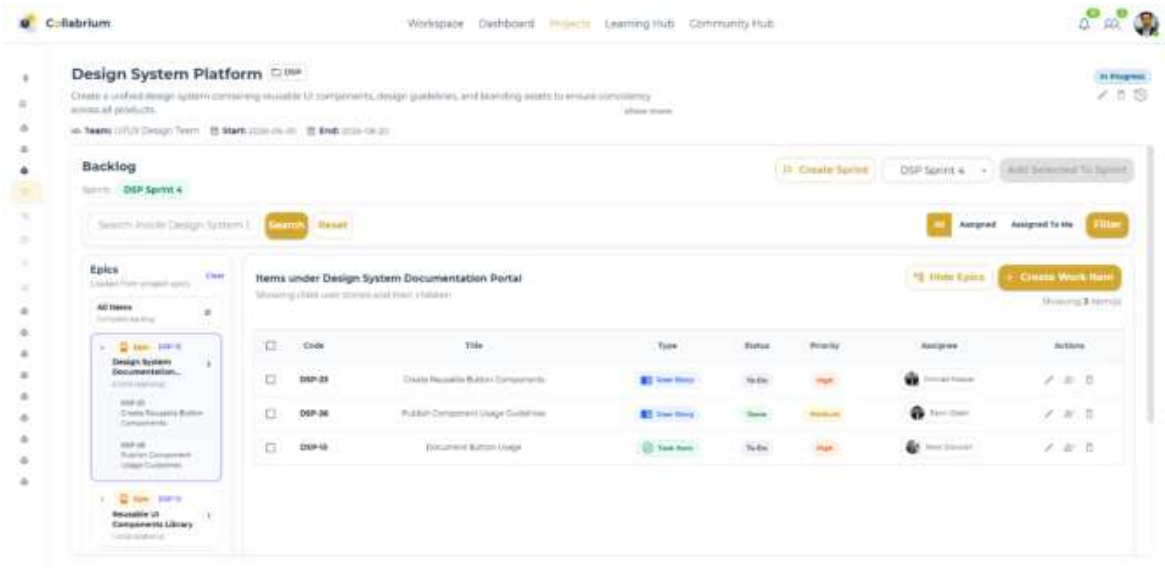


Figure 6.2.6 7 Project Epics & Related Work Items Board

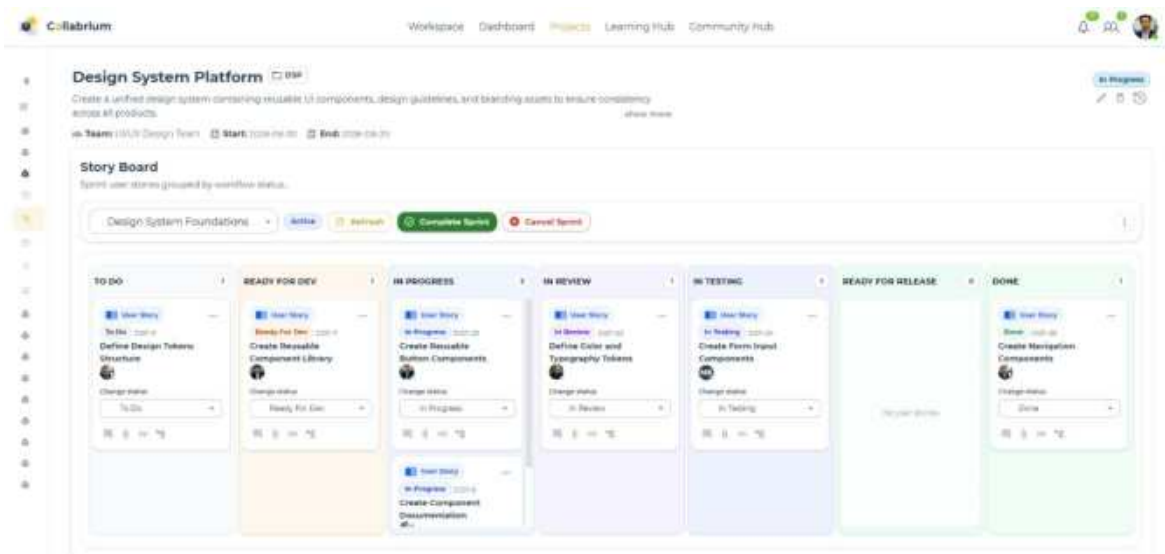


Figure 6.2.6 8 Project Story Board

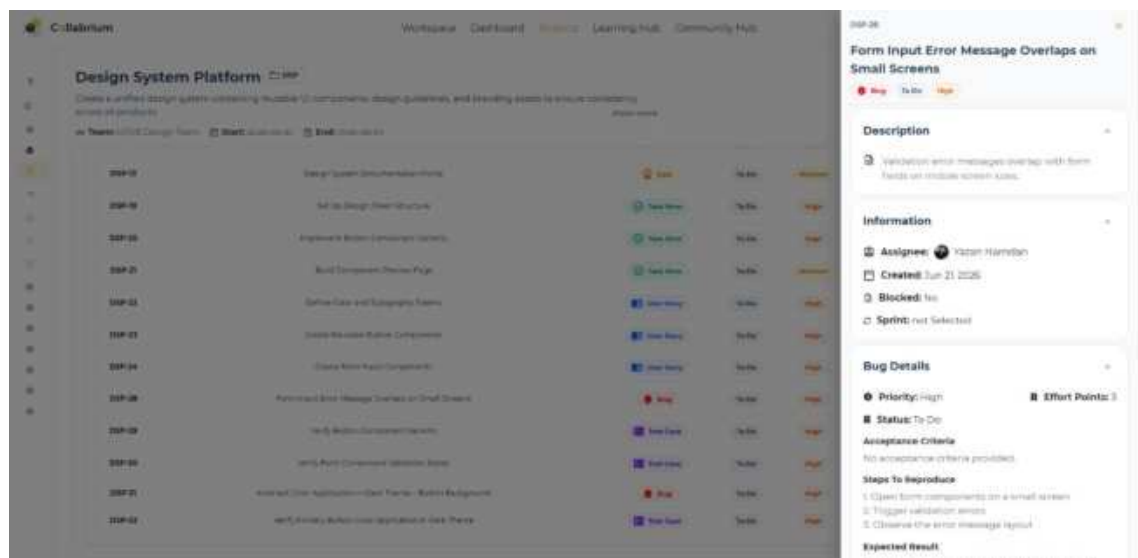


Figure 6.2.6 9 View Work Item Details Drawer

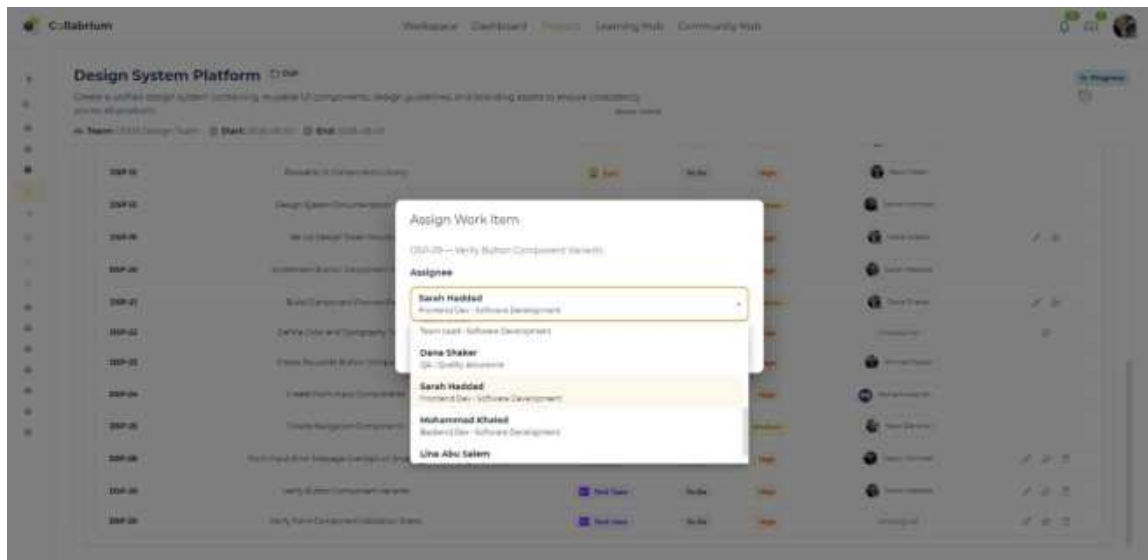


Figure 6.2.6 10 Assign Work Item Dialog

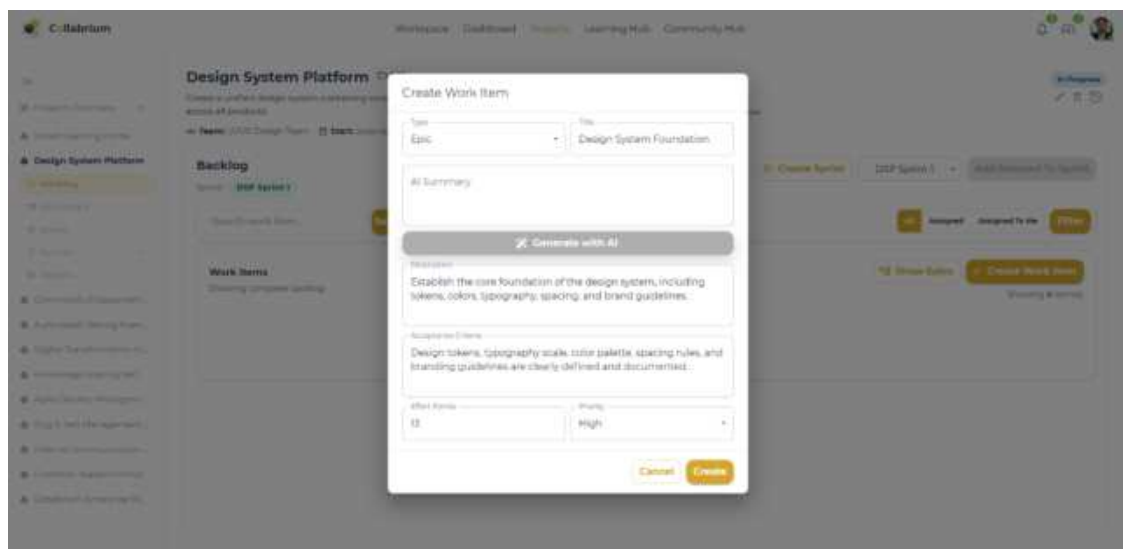


Figure 6.2.6 11 Product Team Manager- Create Planning Work Item Dialog

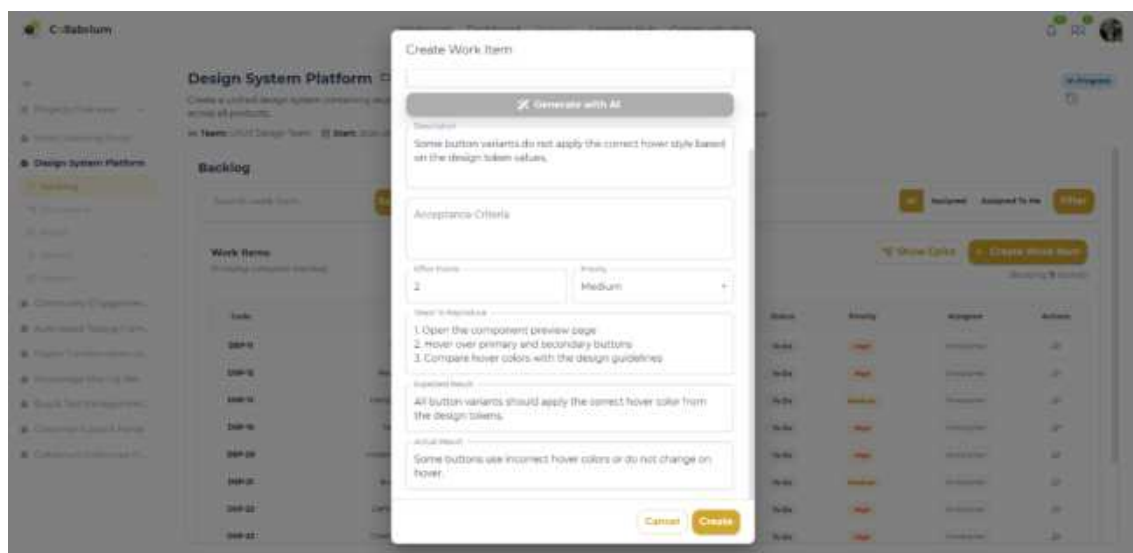


Figure 6.2.6 12 Product Team Member- Create Execution Work Item Dialog

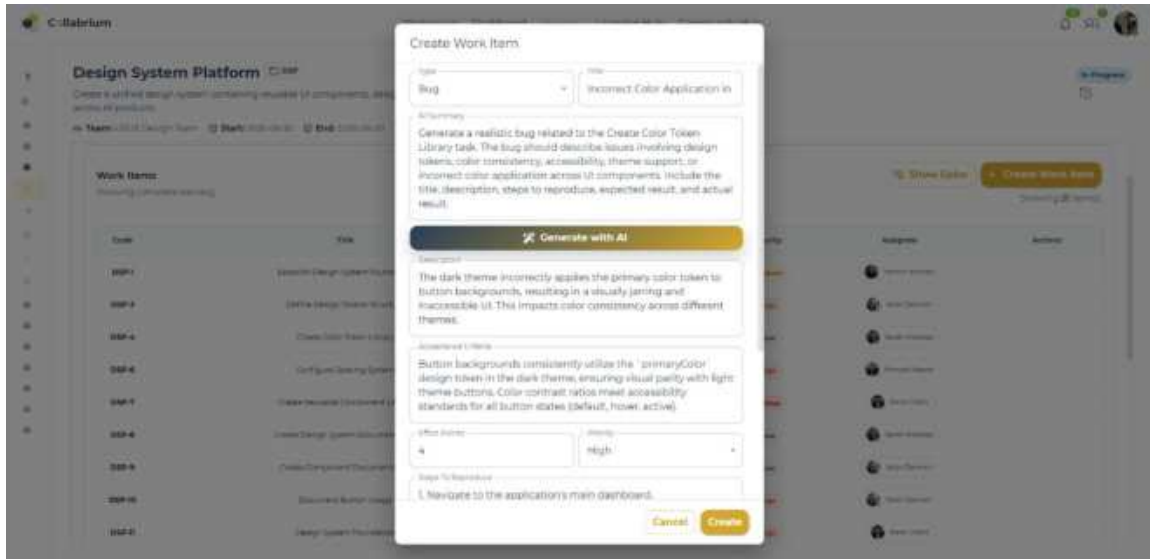


Figure 6.2.7 1 Create Work Item Using AI

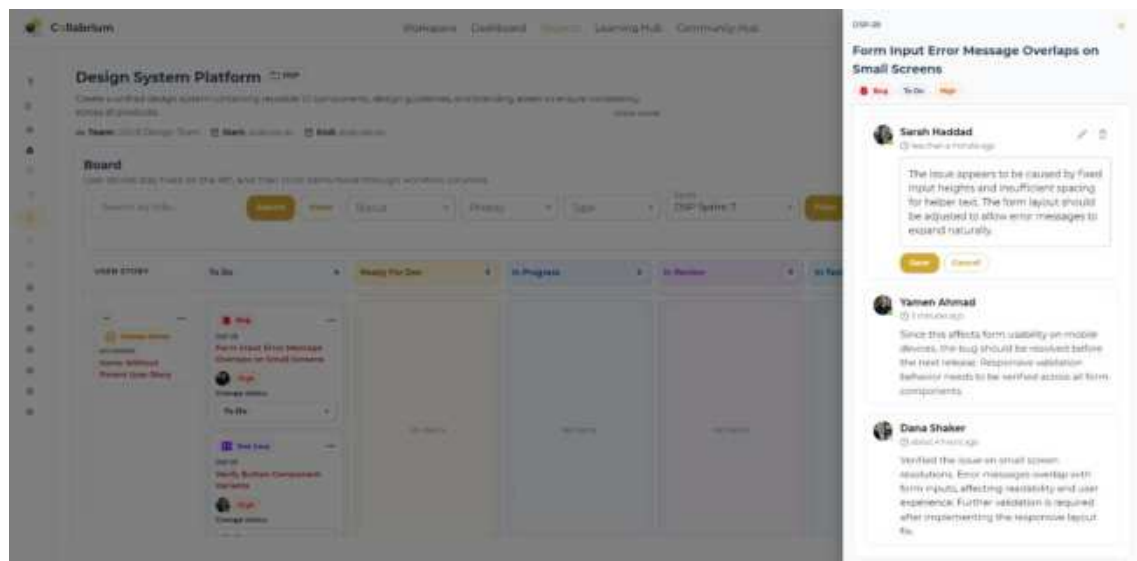


Figure 6.2.6 13 Manage Work Item Comments Interface

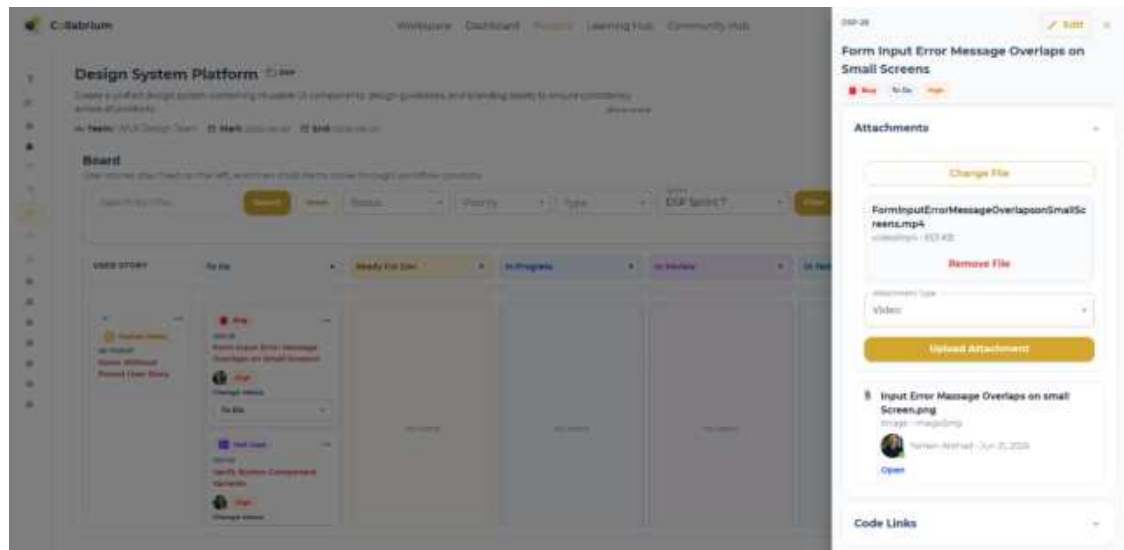


Figure 6.2.6 14 Manage Work Item Attachments Interface

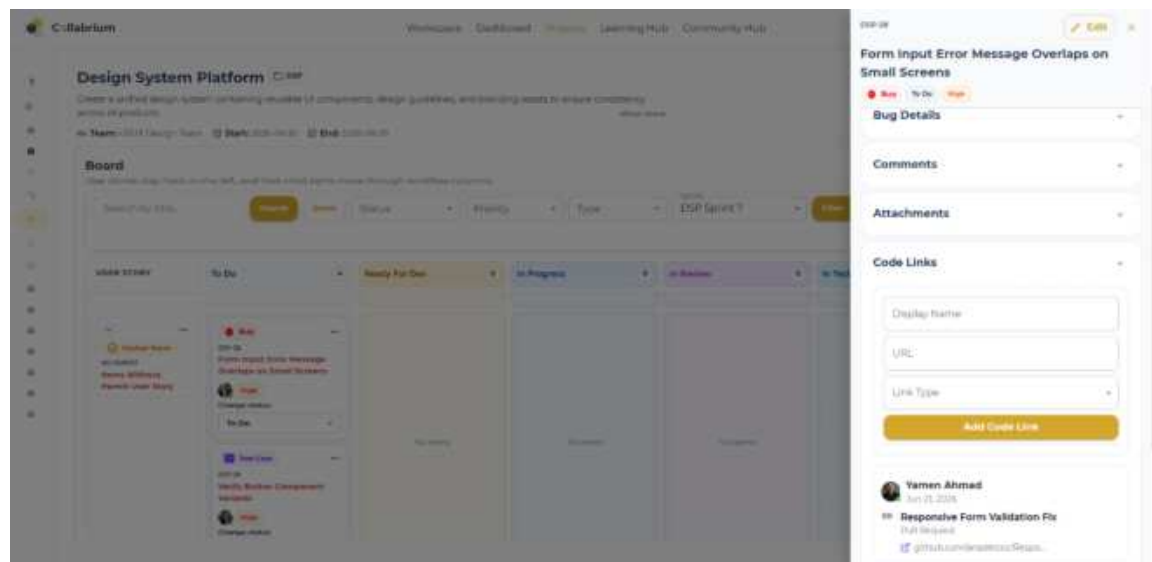


Figure 6.2.6 15 Manage Code Links Interface

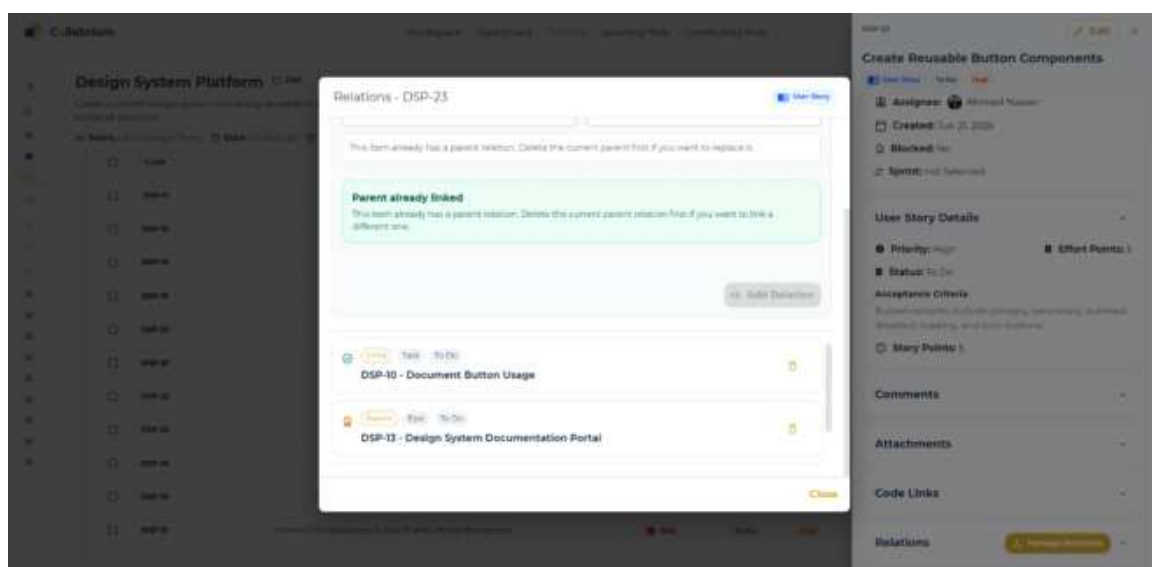


Figure 6.2.6 16 Manage Work Item Relations Dialog

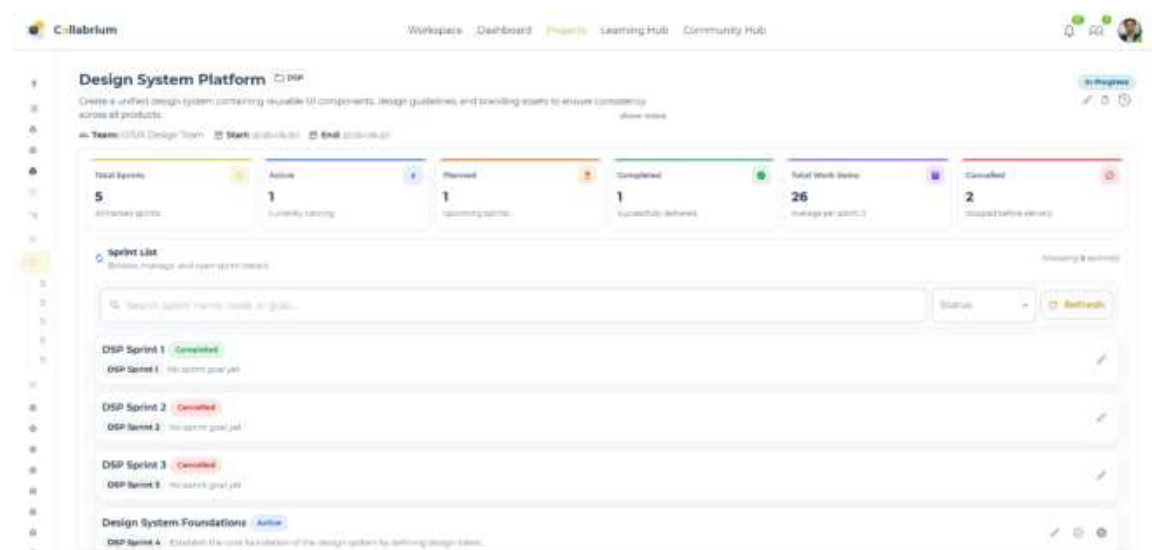


Figure 6.2.6 17 Sprints Interface

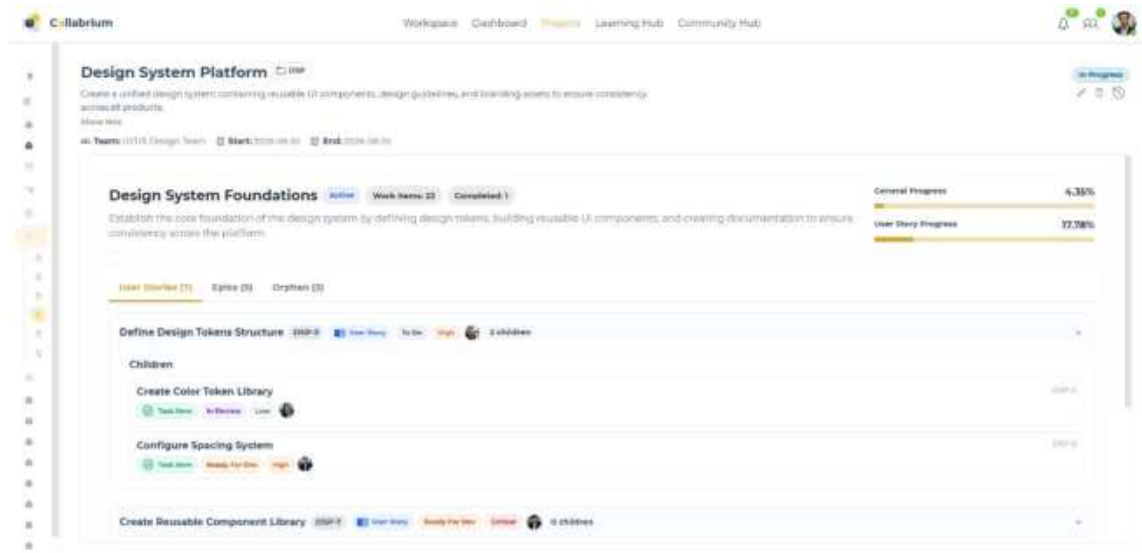


Figure 6.2.6 18 Sprint Details Interface

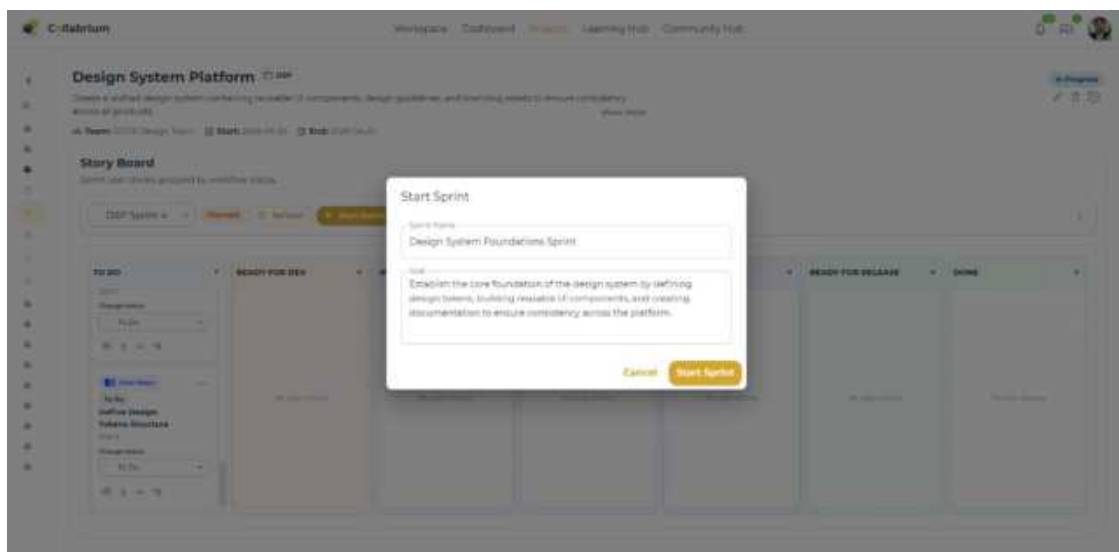


Figure 6.2.6 19 Start Sprint Dialog

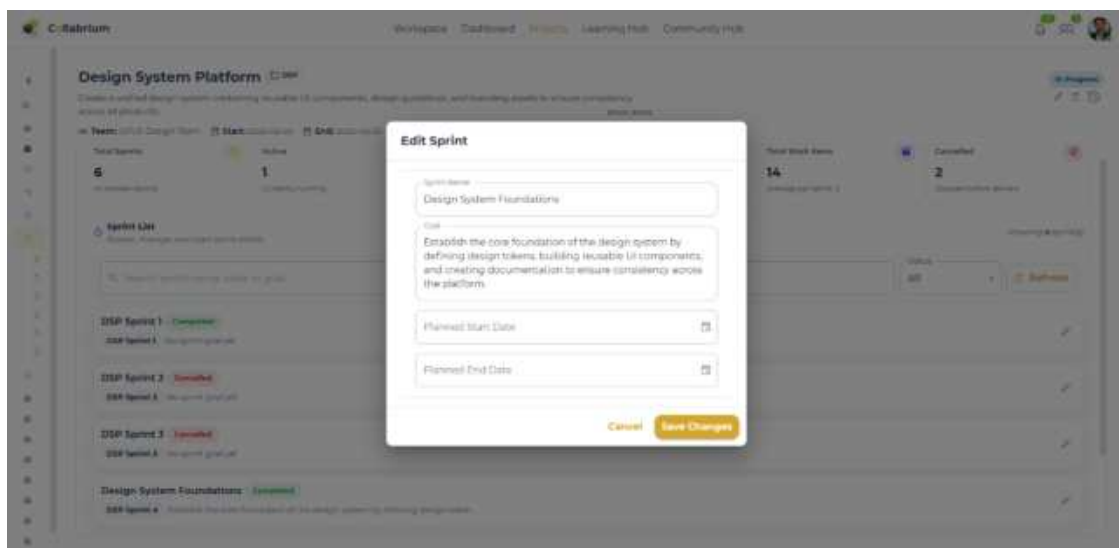


Figure 6.2.6 20 Edit Sprint Dialog

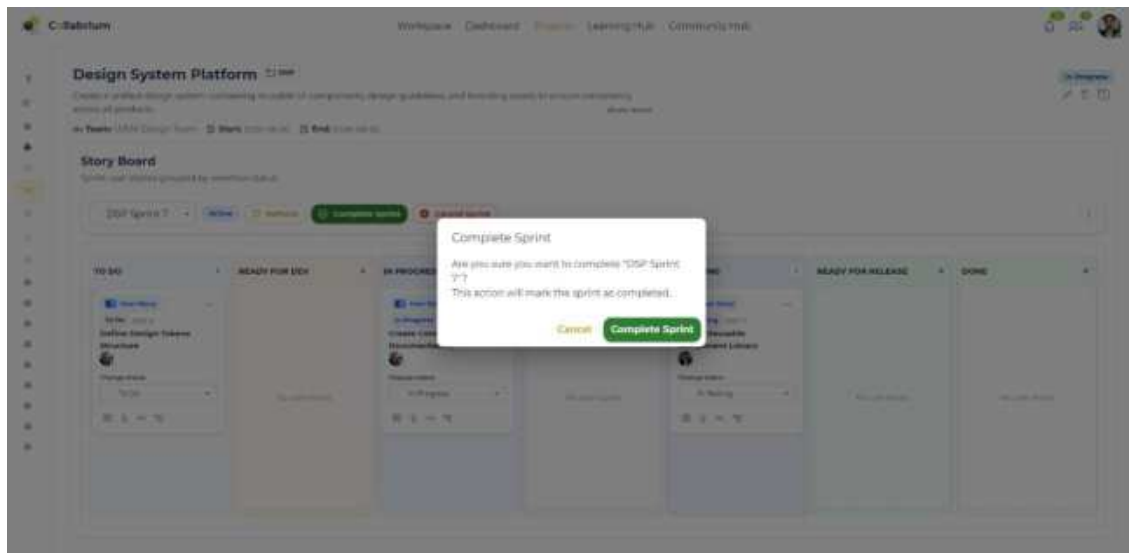


Figure 6.2.6 21 Complete Sprint Dialog

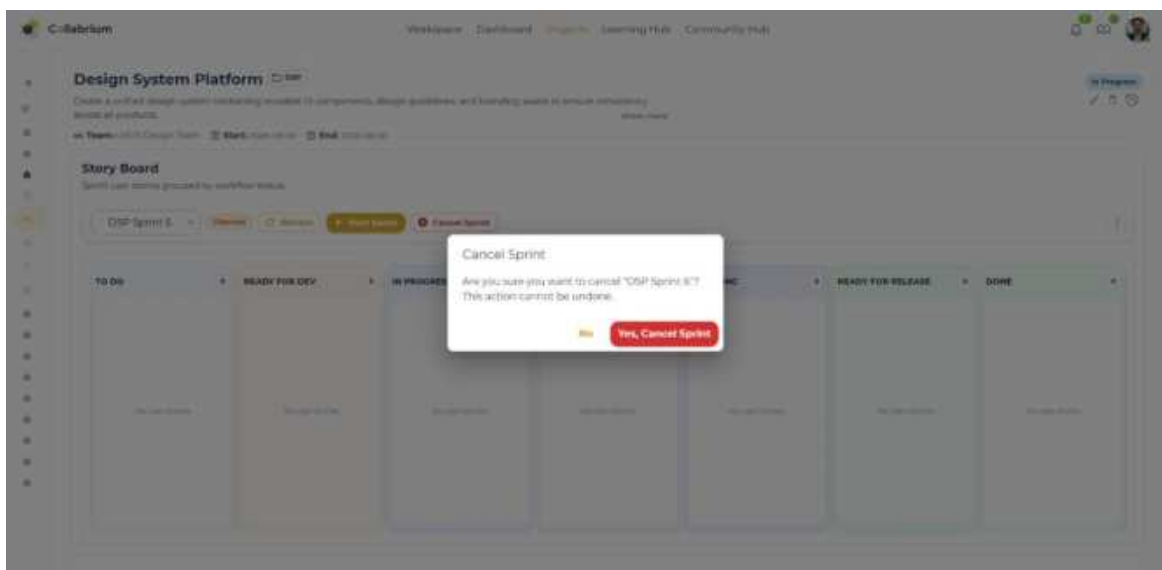


Figure 6.2.6 22 Cancel Sprint Dialog

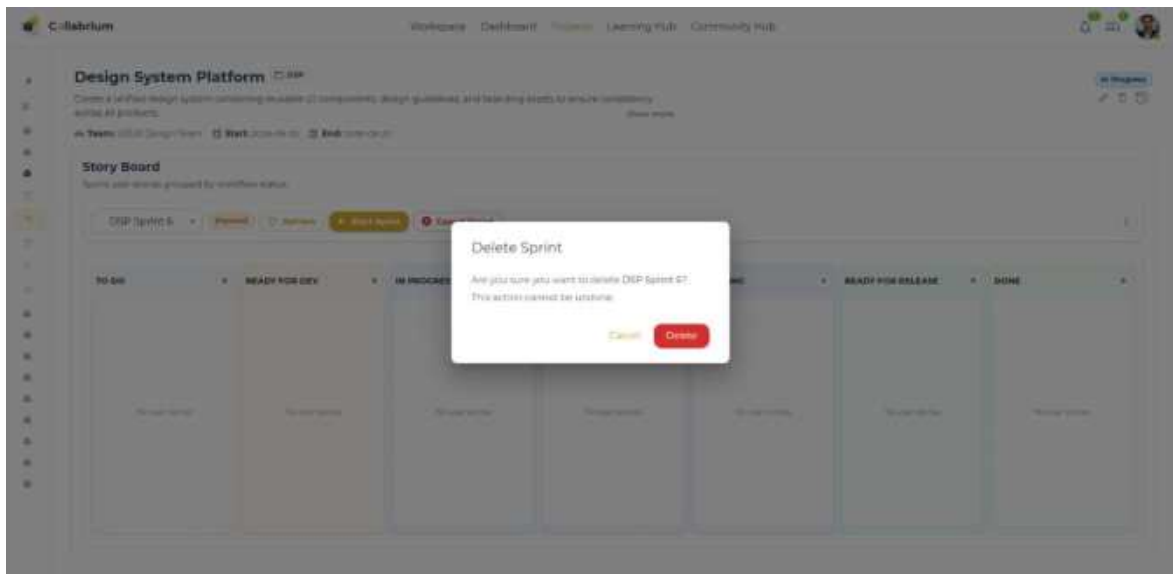


Figure 6.2.6 23 Delete Sprint Dialog

# 6.2.7 Analysis Interfaces

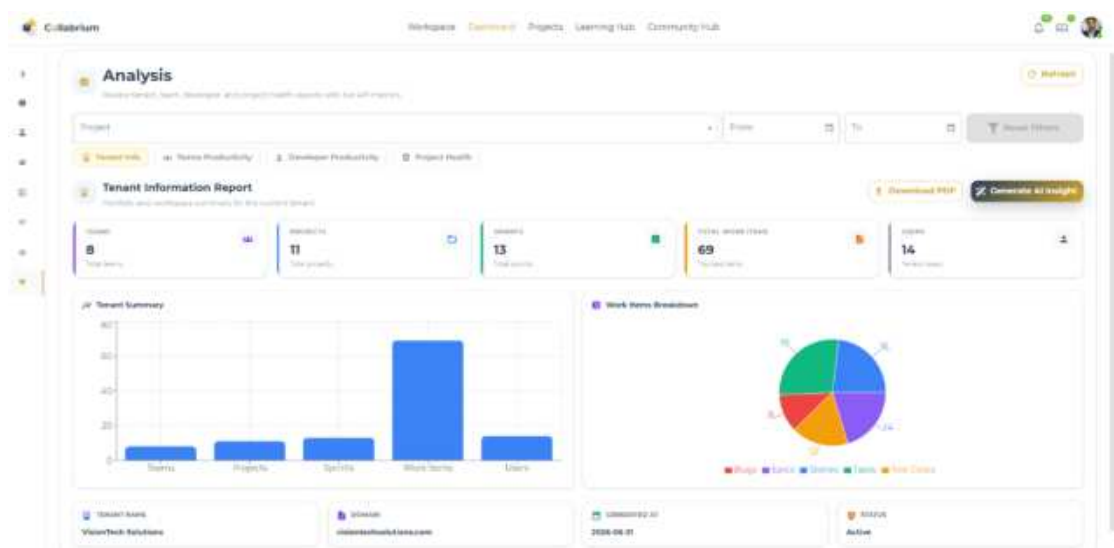


Figure 6.2.7 2 Tenant Information Report

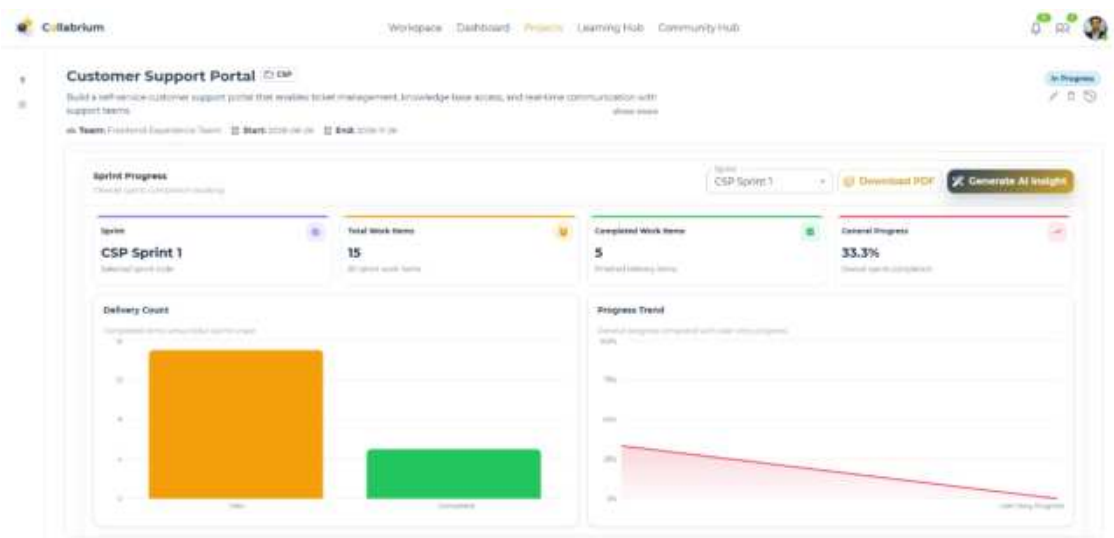


Figure 6.2.7 3 Sprint Progress Report

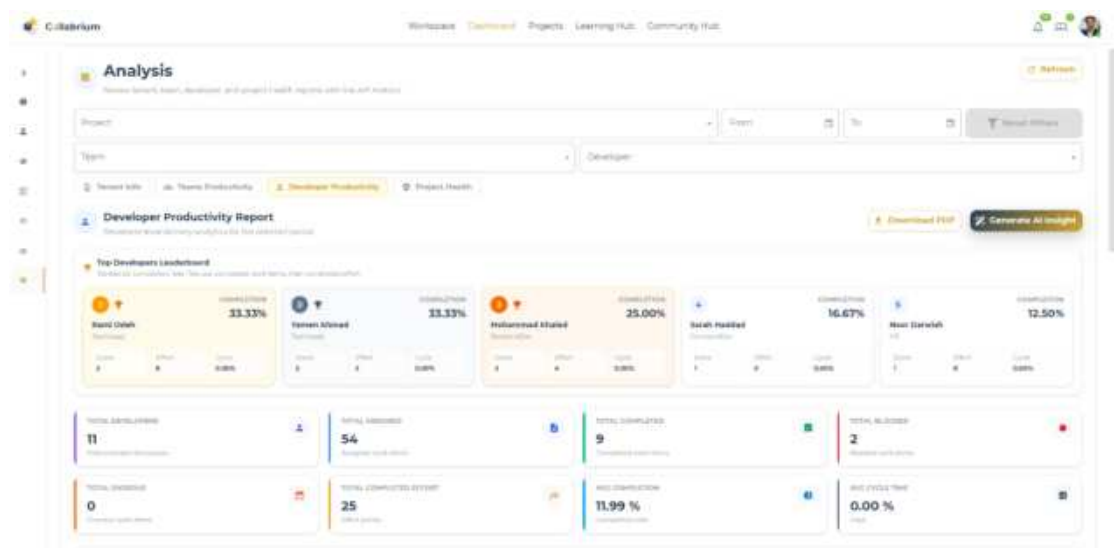


Figure 6.2.7 4 Developer Productivity Report

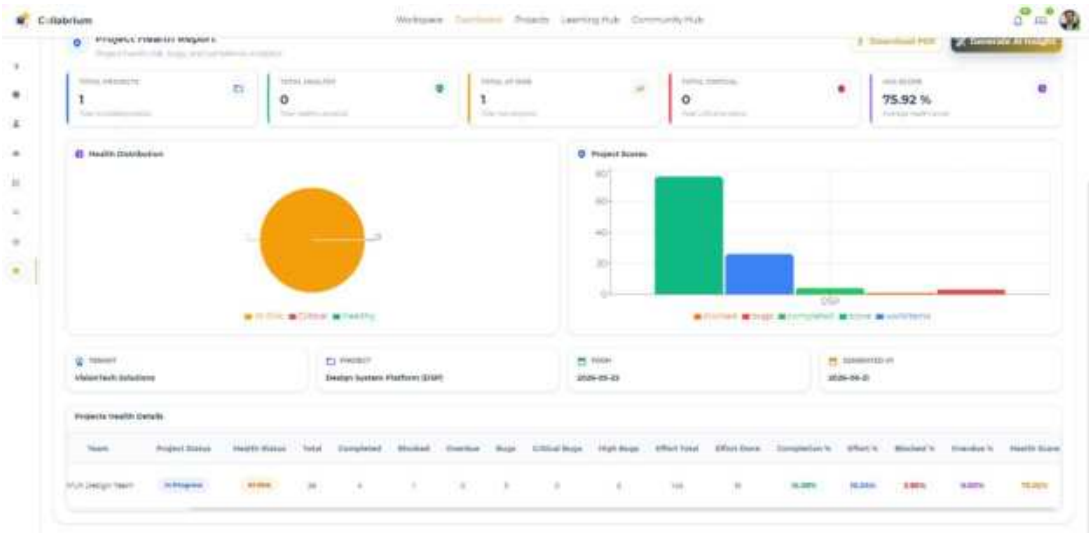


Figure 6.2.7 5 Project Health Report

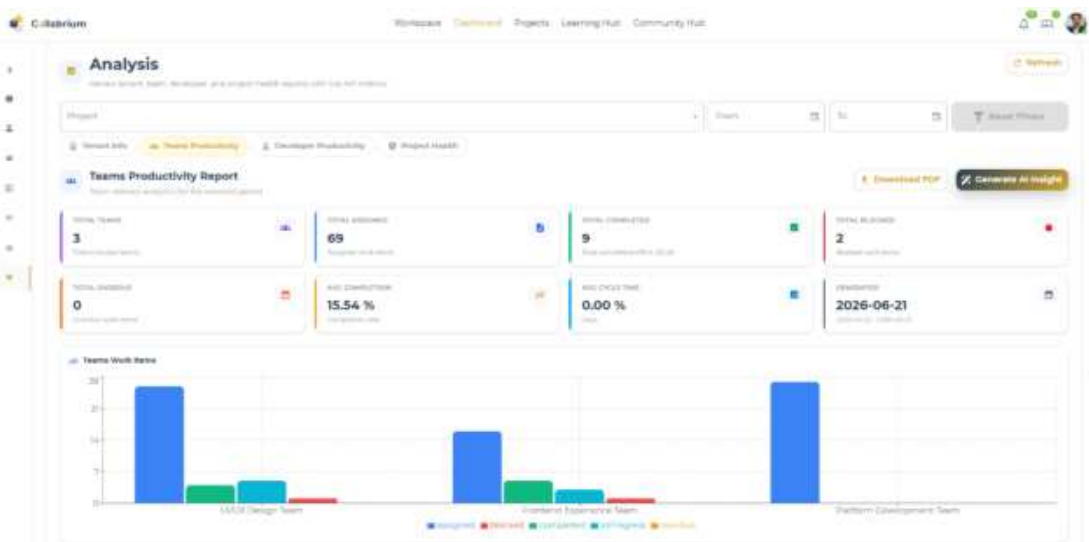


Figure 6.2.7 6 Teams Productivity Report

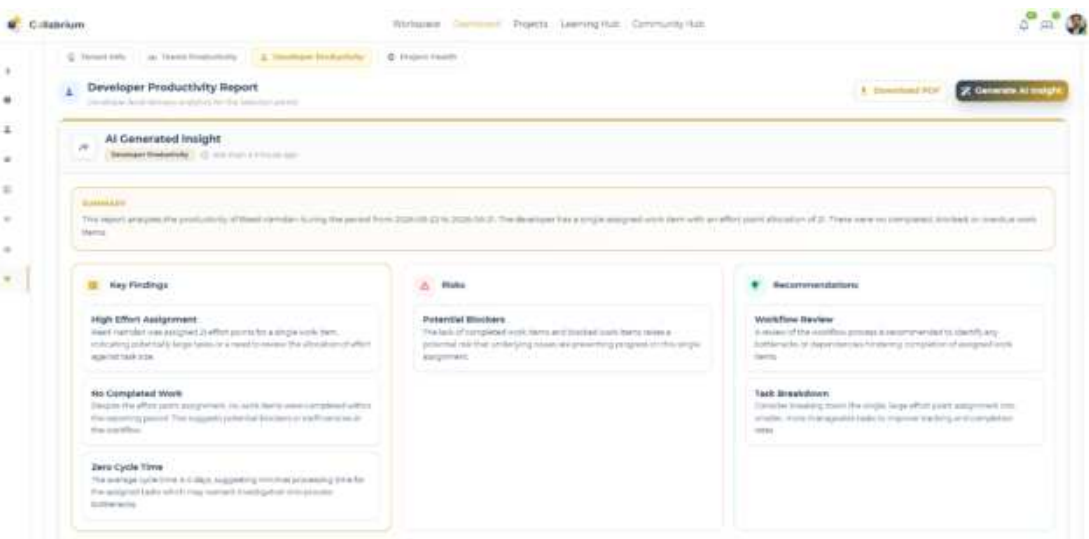


Figure 6.2.7 7 AI Generated Report

6.2.8 Learning Hub Interfaces

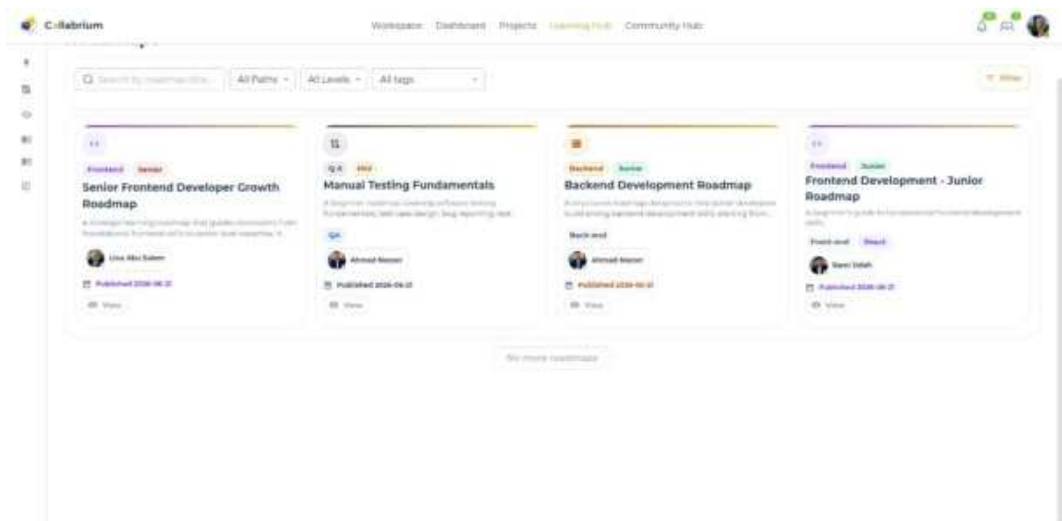


Figure 6.2.8 1 Tenant Member- Roadmap Page

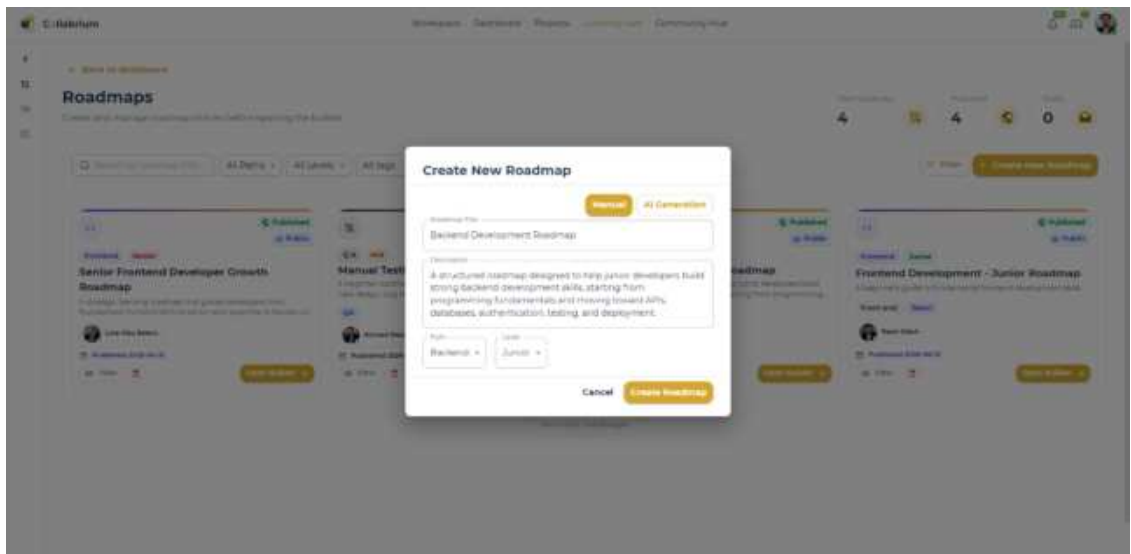


Figure 6.2.8 2 Tenant Manager- Create Roadmap Dialog

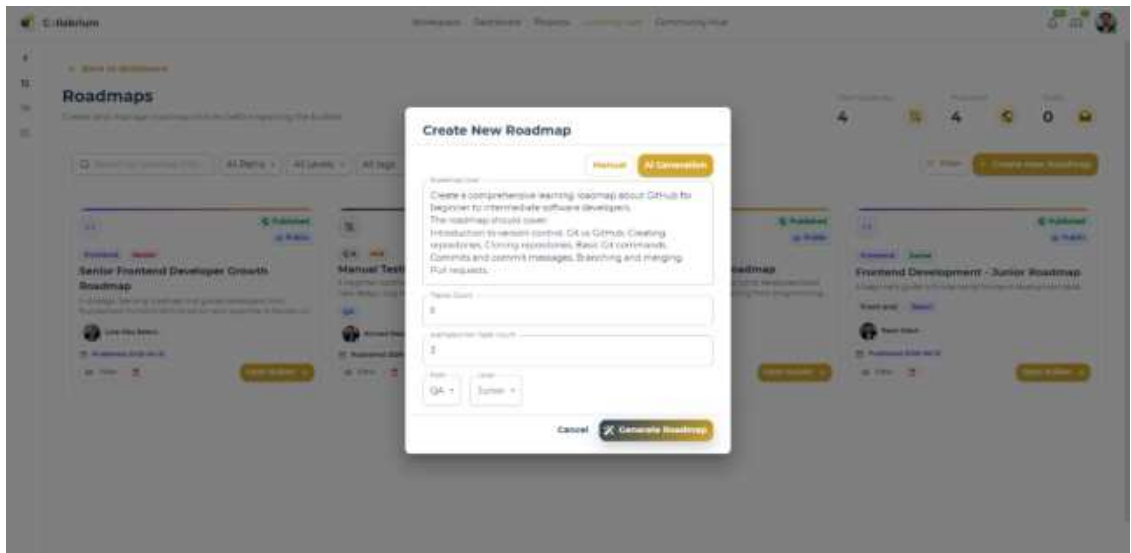


Figure 6.2.8 3 Tenant Manager- Create Roadmap Using AI Dialog

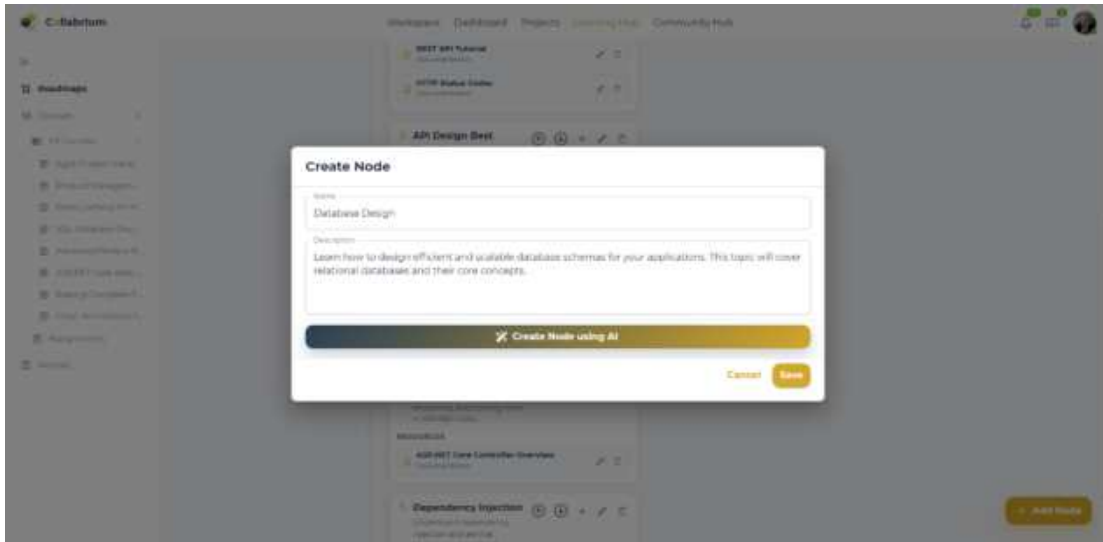


Figure 6.2.8 4 Tenant Manager- Create Roadmap Node Using AI Dialog

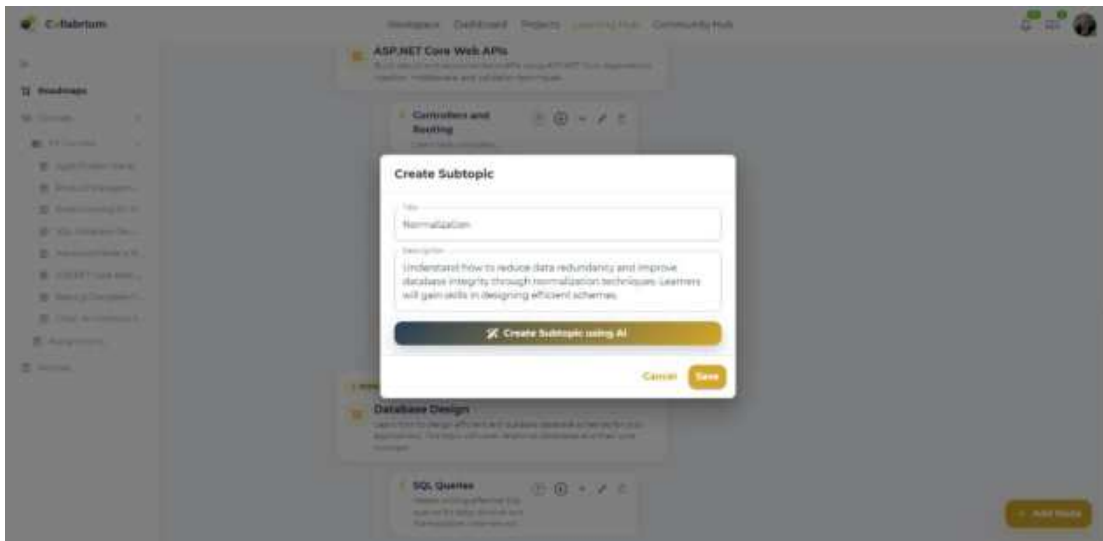


Figure 6.2.8 5 Tenant Manager- Create Roadmap Subtopic Using AI Dialog

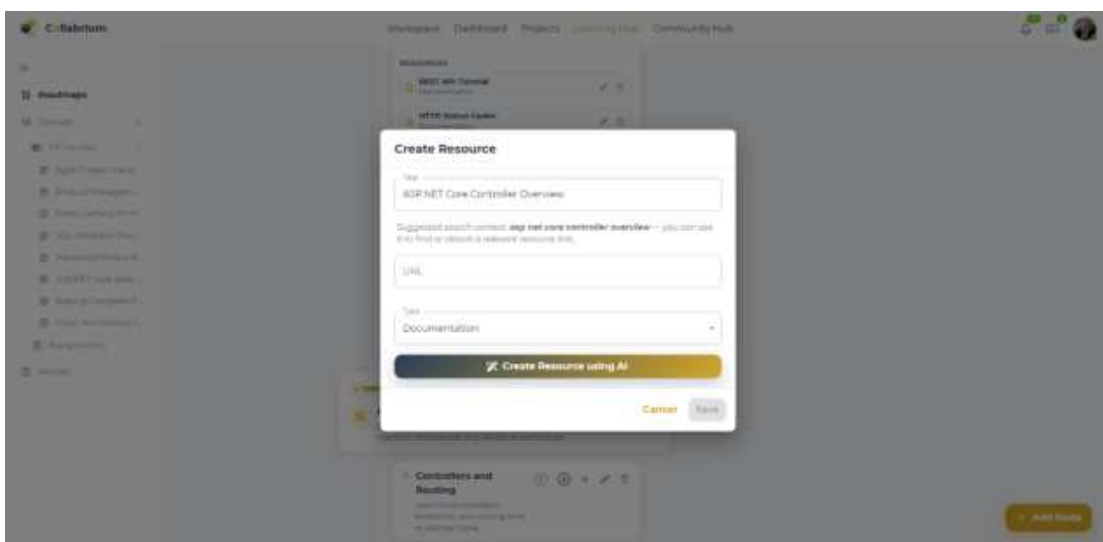


Figure 6.2.8 6 Tenant Manager- Create Roadmap Resource Using AI Dialog

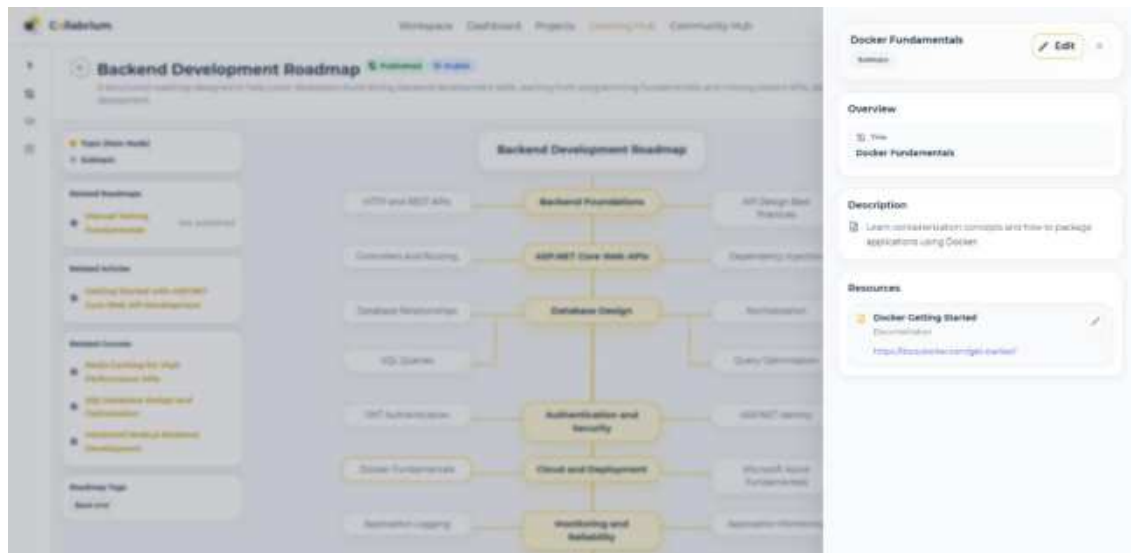


Figure 6.2.8 7 View Roadmap Node Details Drawer

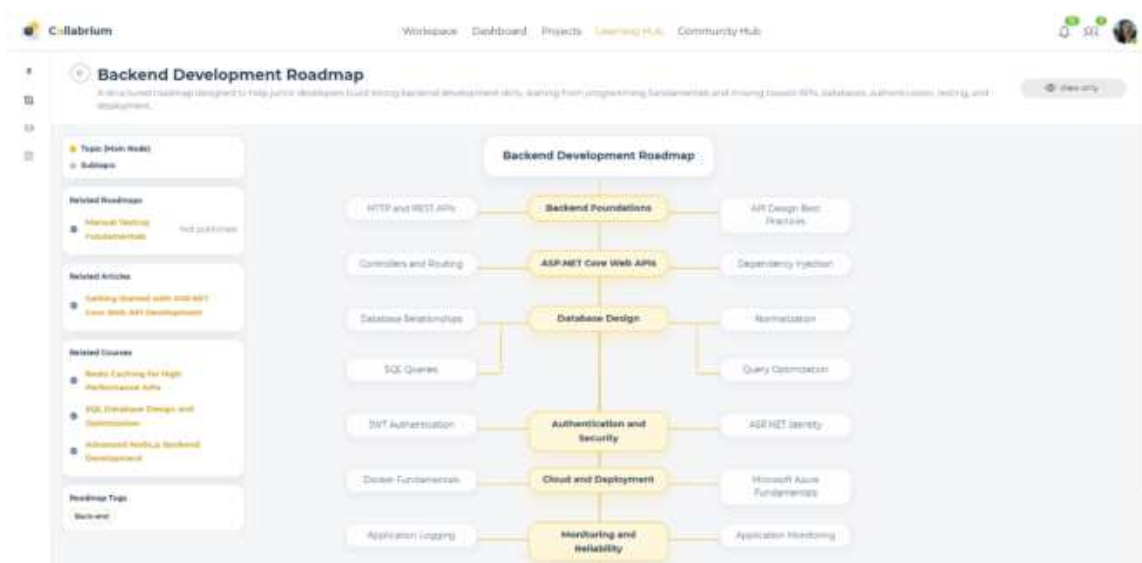


Figure 6.2.8 8 Roadmap View Page

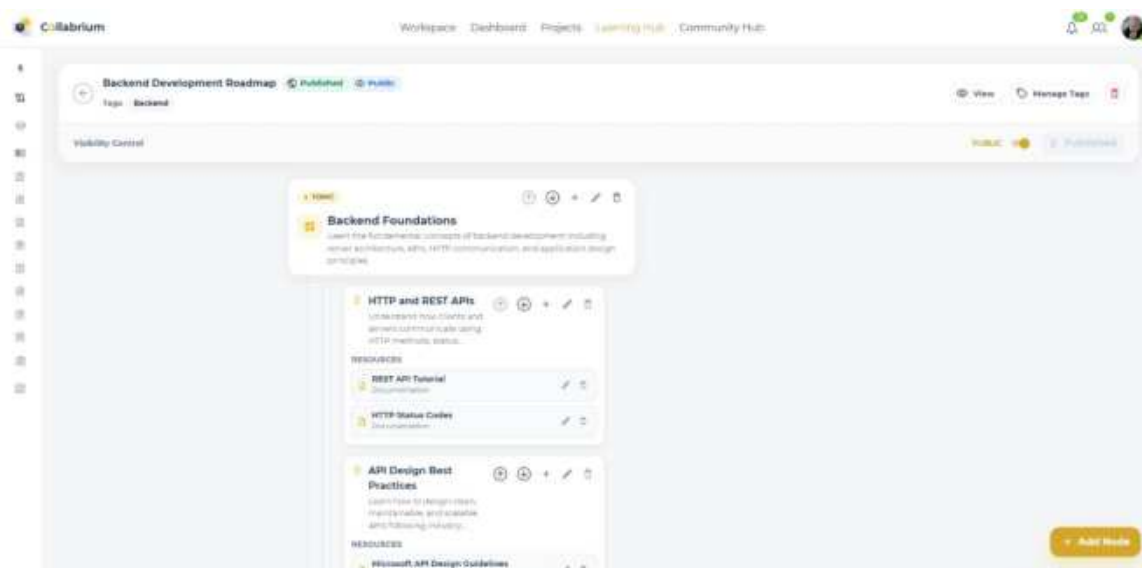


Figure 6.2.8 9 Roadmap Builder Page

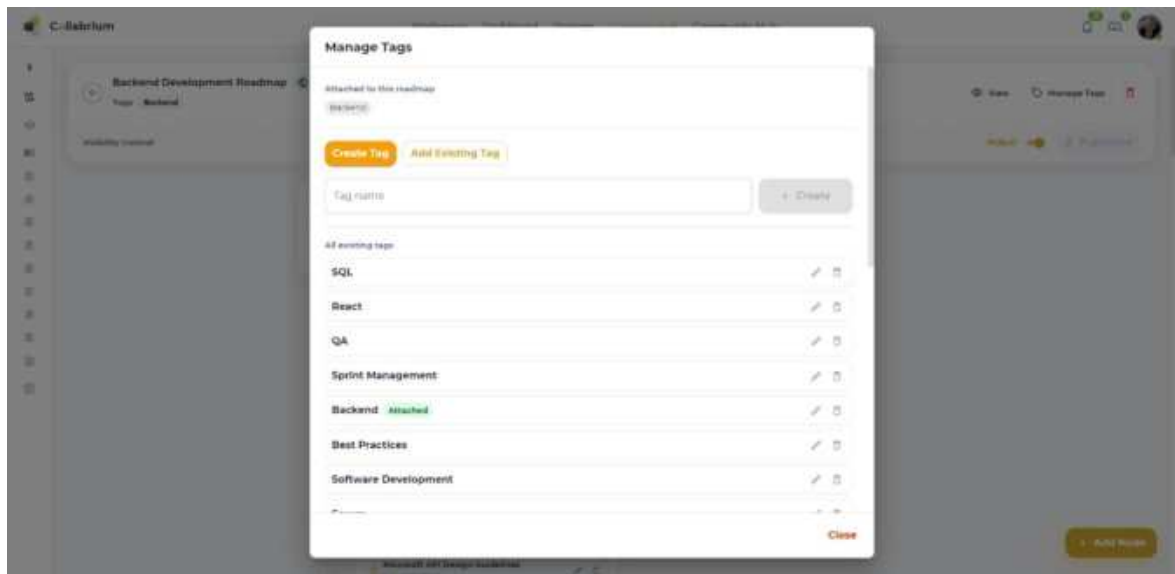


Figure 6.2.8 10 Manage Roadmap Tags Dialog

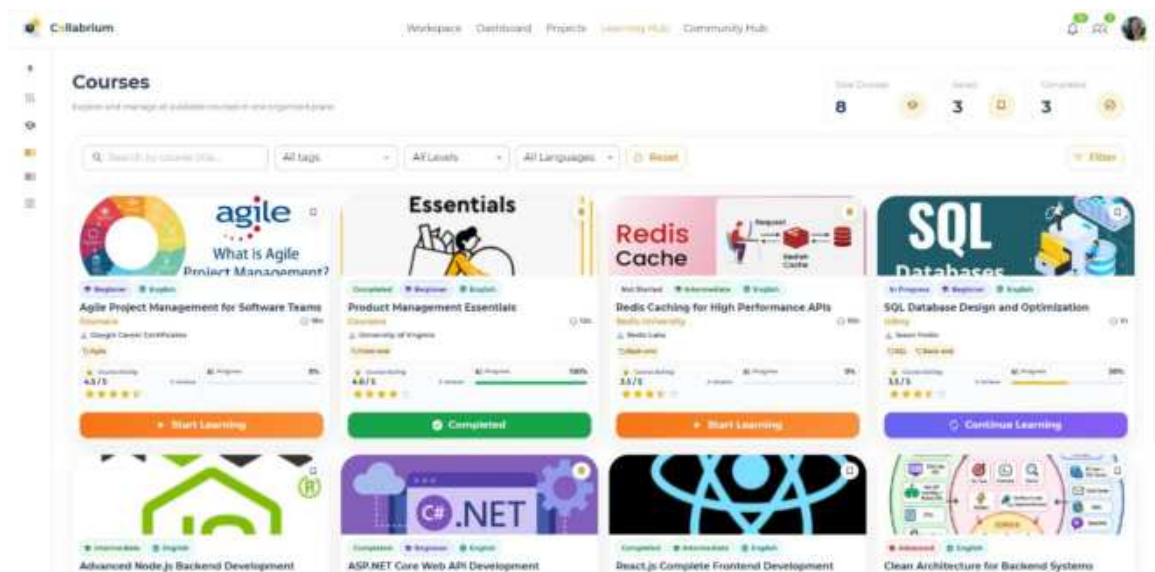


Figure 6.2.8 11 Courses Page

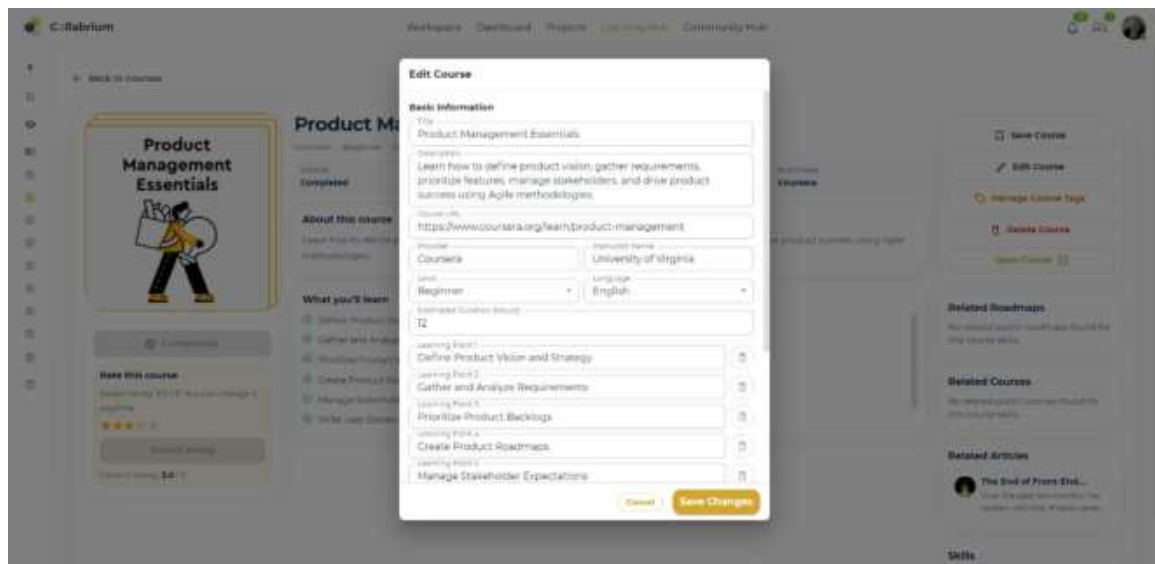


Figure 6.2.8 12 Tenant Manager- Create & Edit Course Dialog

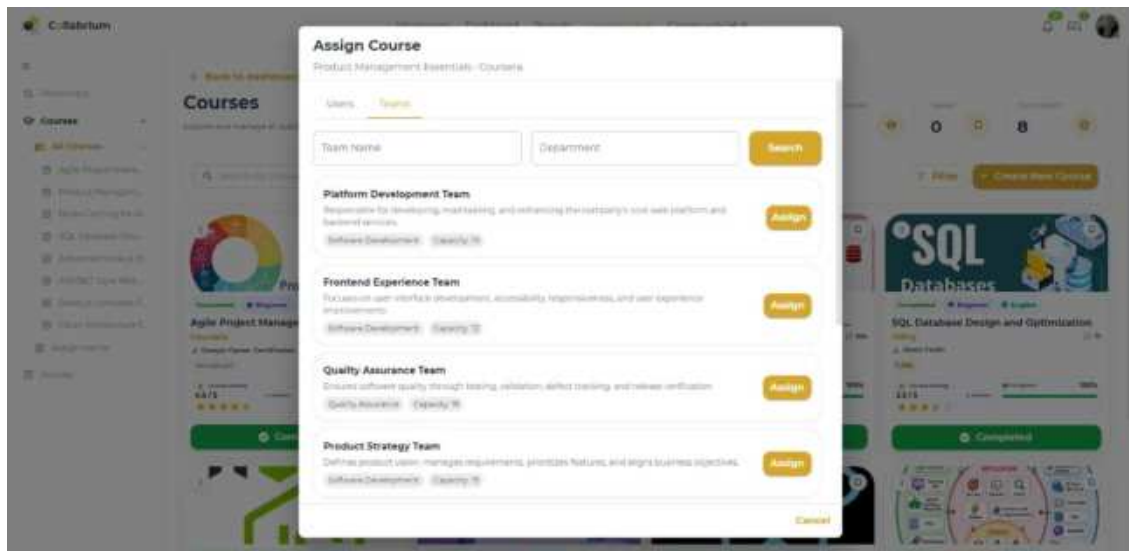


Figure 6.2.8 13 Tenant Manager- Assign Course To Team/ Member Dialog

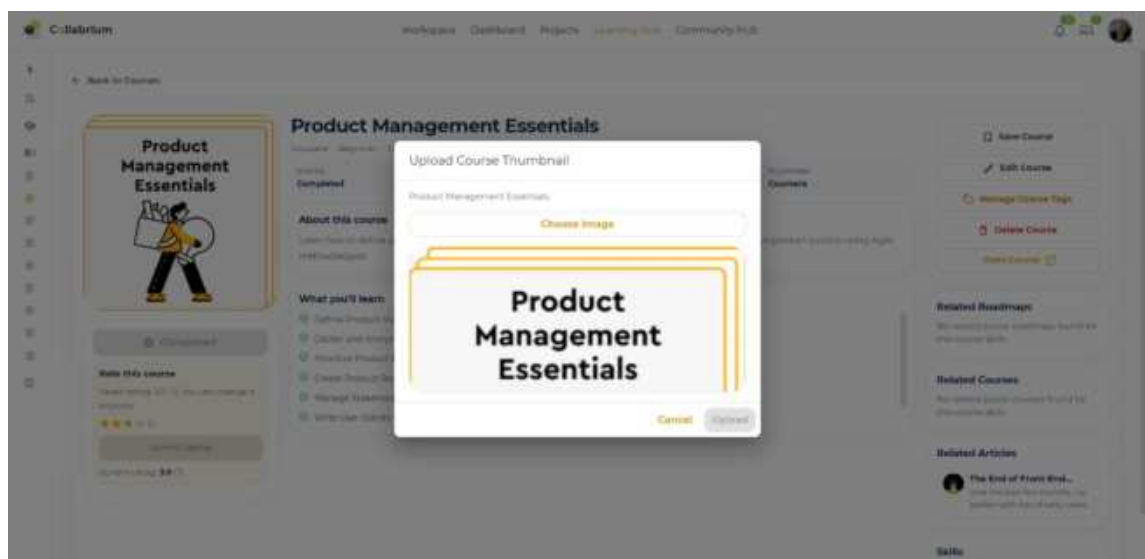


Figure 6.2.8 14 Tenant Manager- Change Course Thumbnail Dialog

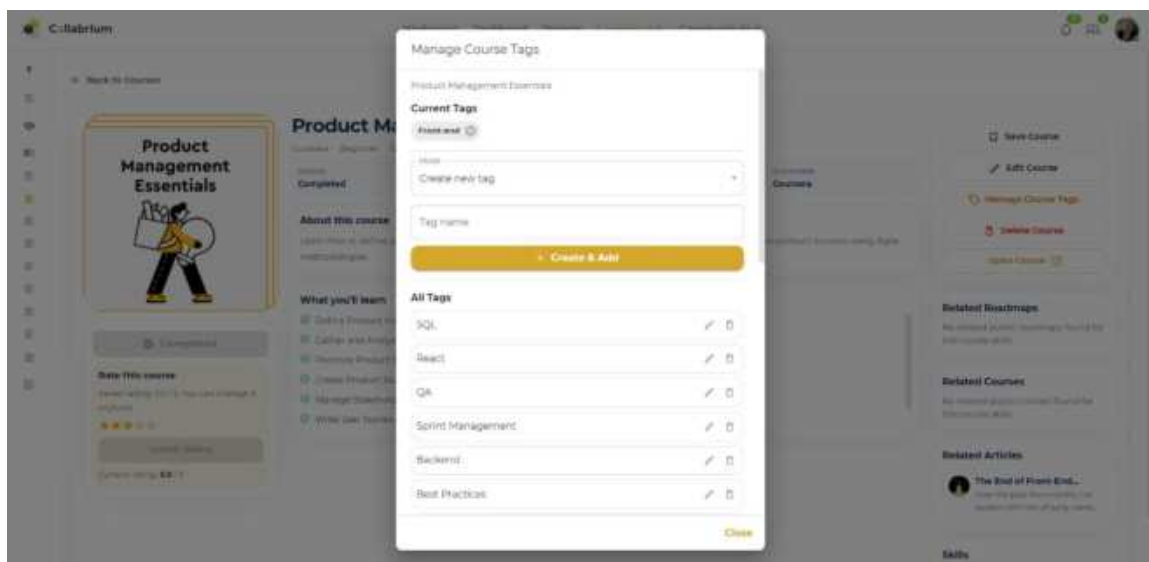


Figure 6.2.8 15 Manage Roadmap Tags Dialog

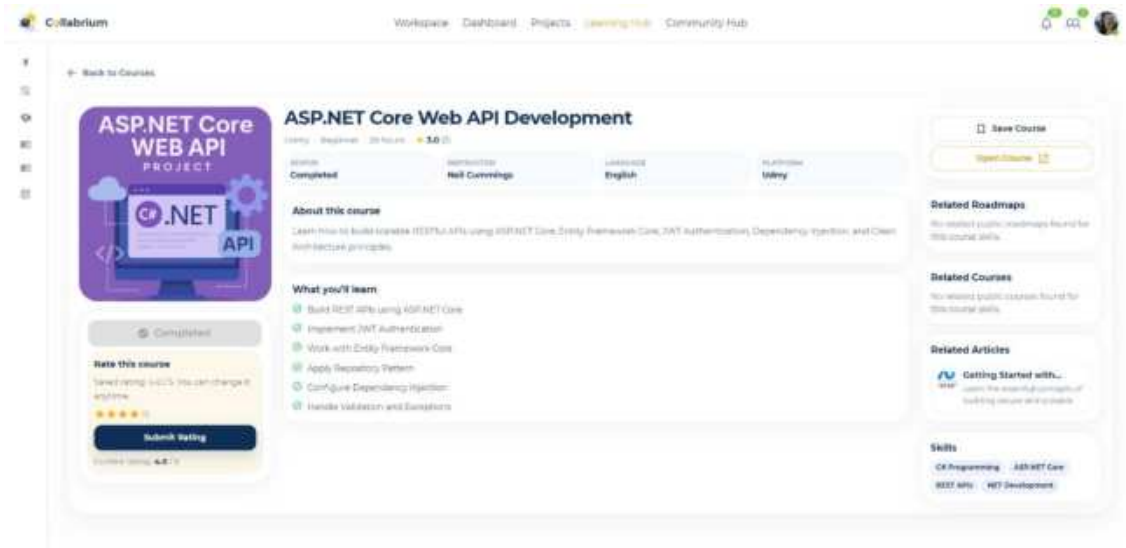


Figure 6.2.8 16 Submit Course Rating Interface

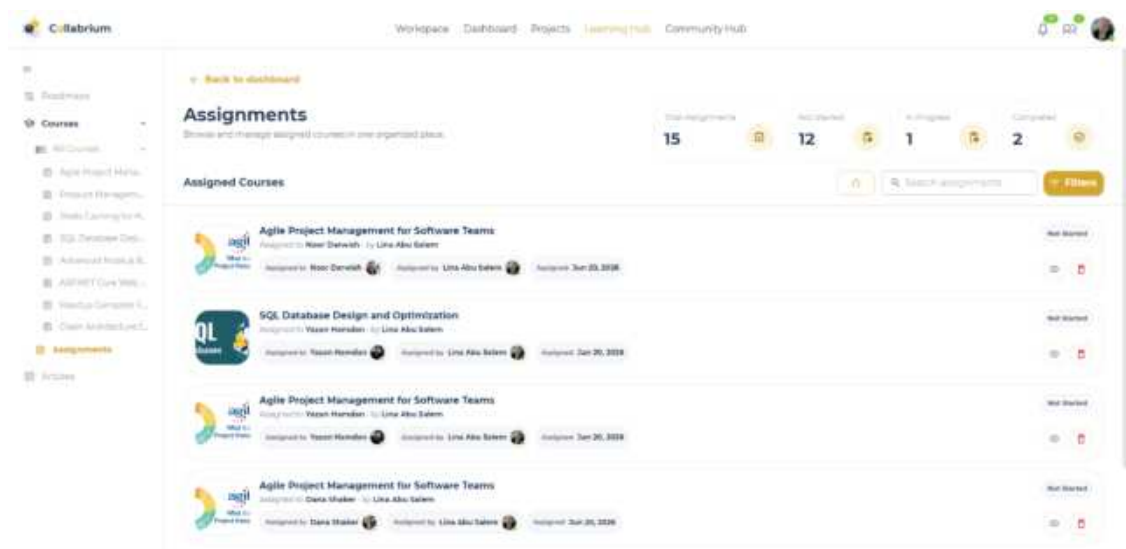


Figure 6.2.8 17 Tenant Manager- Assignments Page

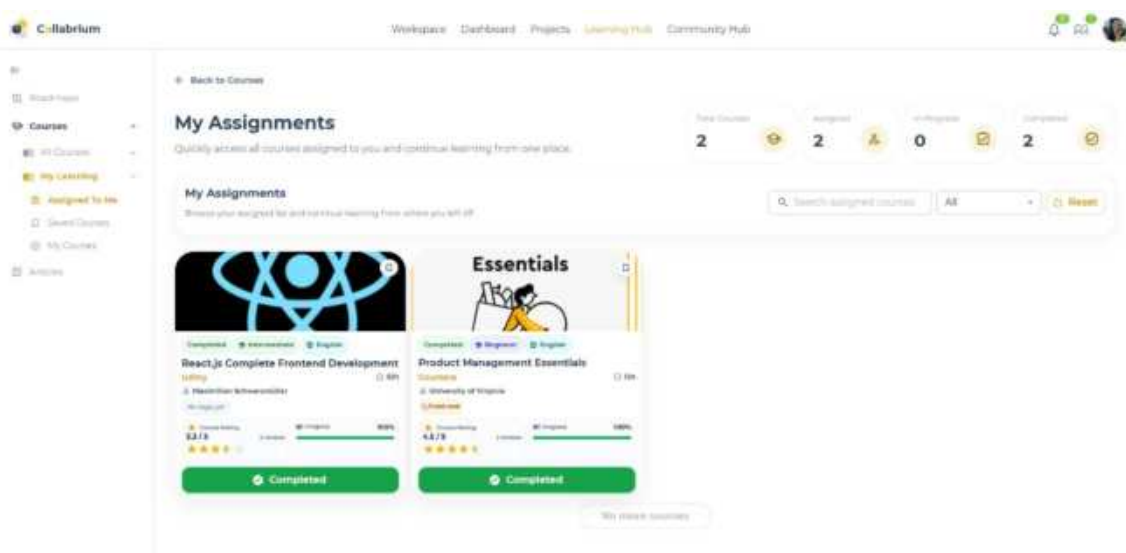


Figure 6.2.8 18 Tenant Member- Assign to Me Page

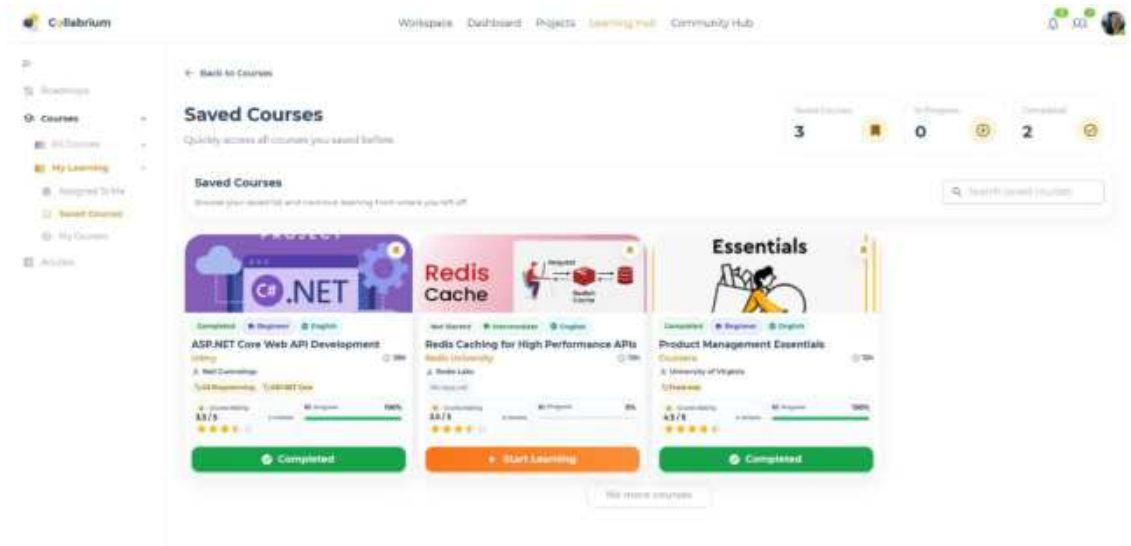


Figure 6.2.8 19 Tenant Member- Saved Courses Interface

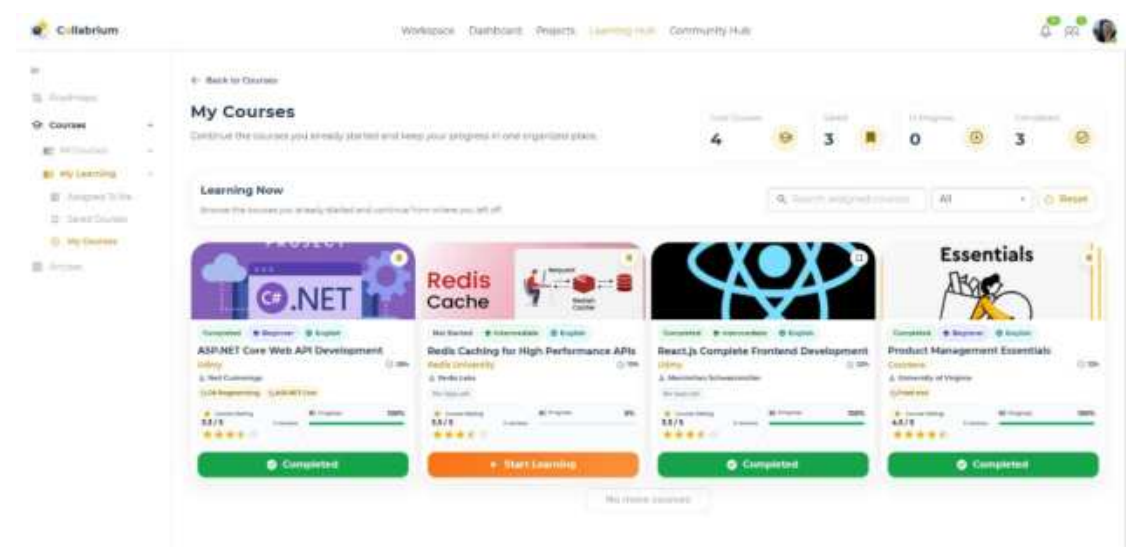


Figure 6.2.8 20 Tenant Member- My Courses Interface

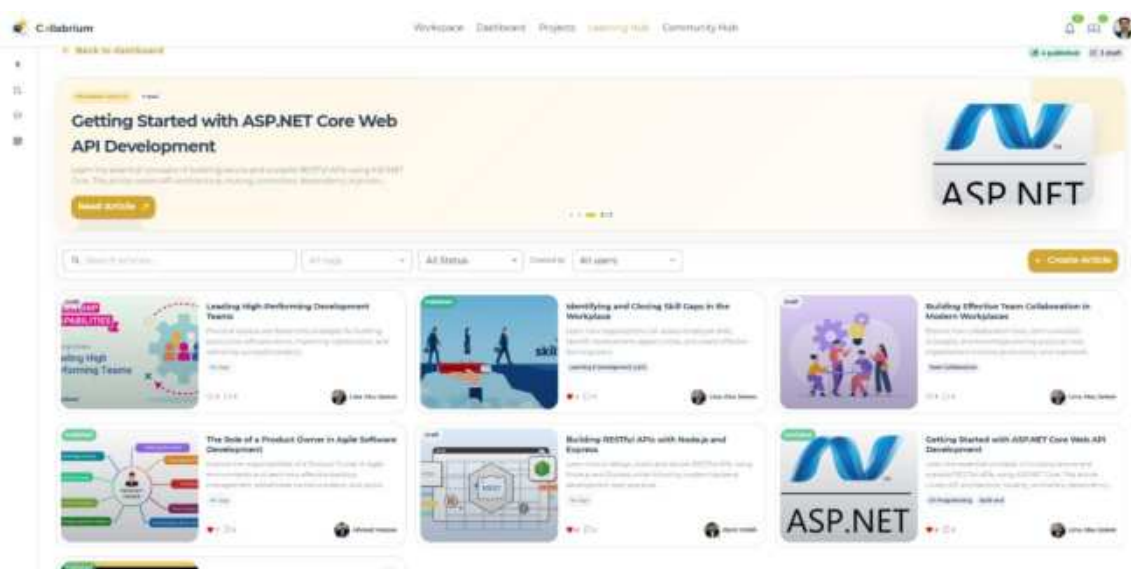


Figure 6.2.8 21 Tenant Manager- Articles Page

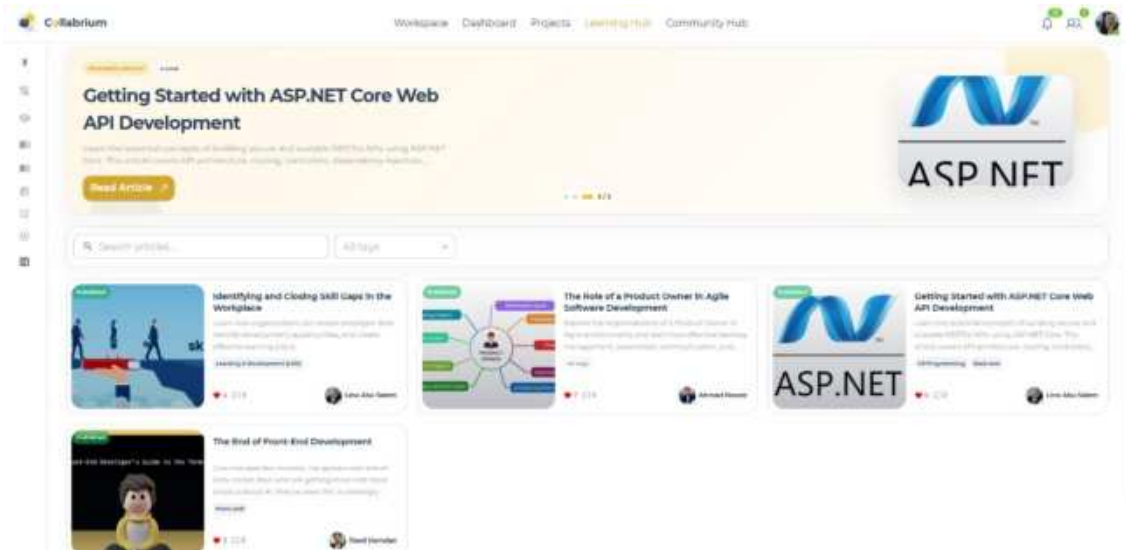


Figure 6.2.8 22 Tenant Member- Articles Page

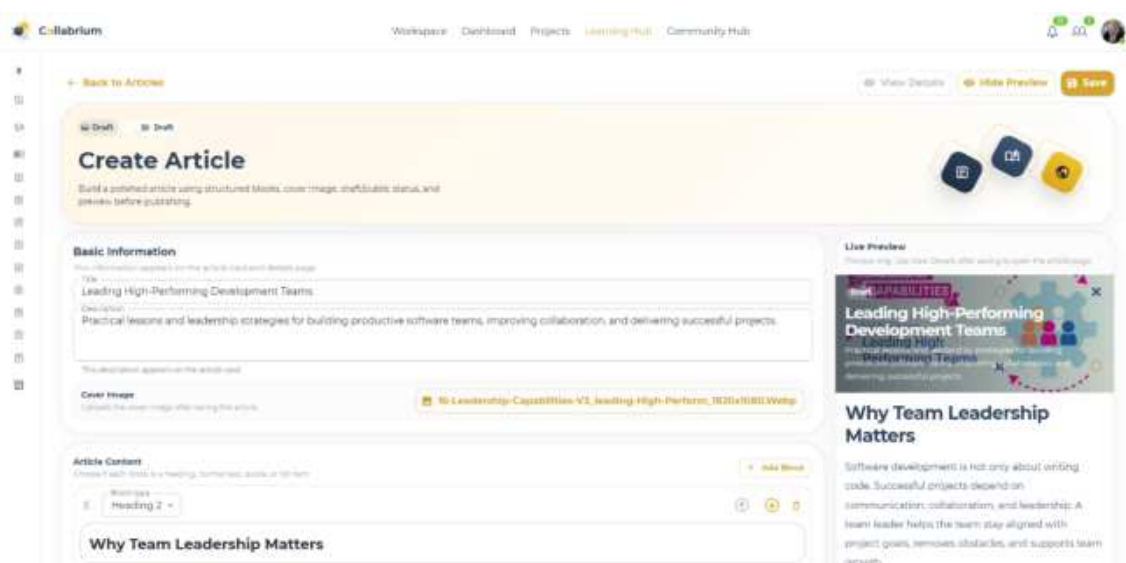


Figure 6.2.8 23 Tenant Manager- Create Article Page

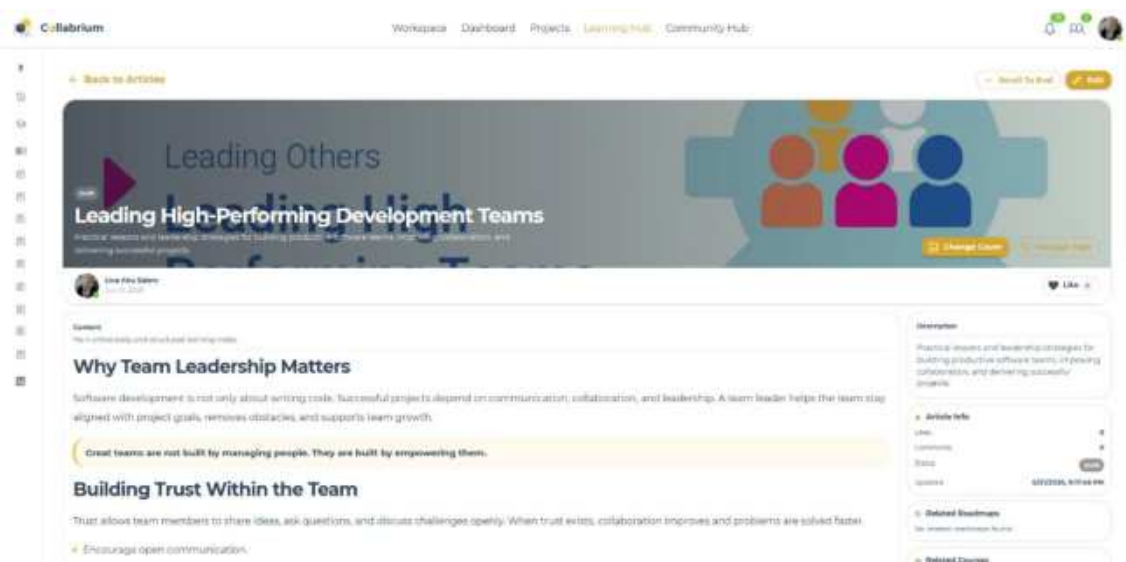


Figure 6.2.8 24 Detailed Article Page

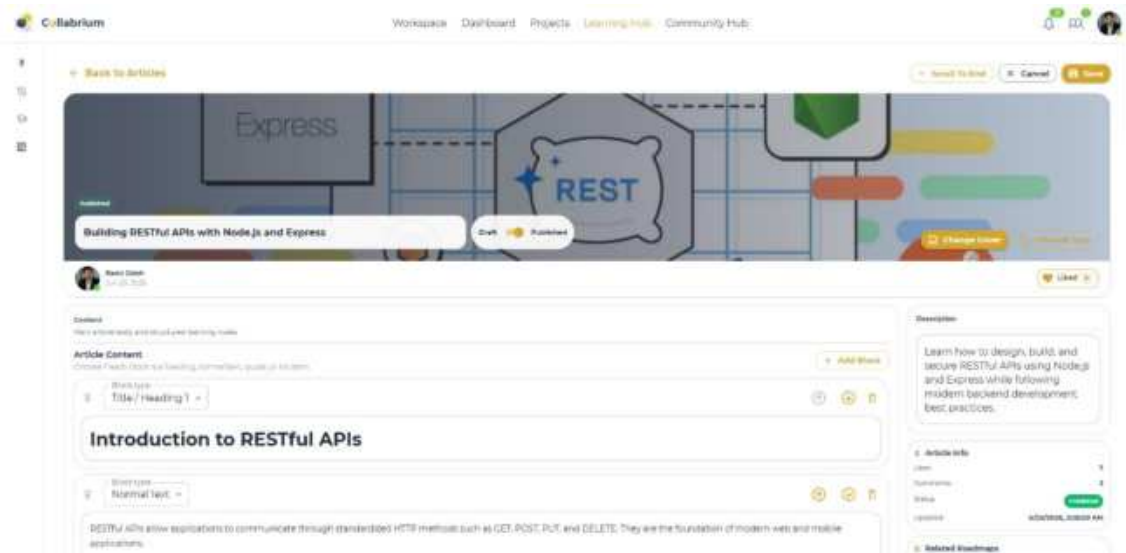


Figure 6.2.8 25 Tenant Manager- Edit Article Interface

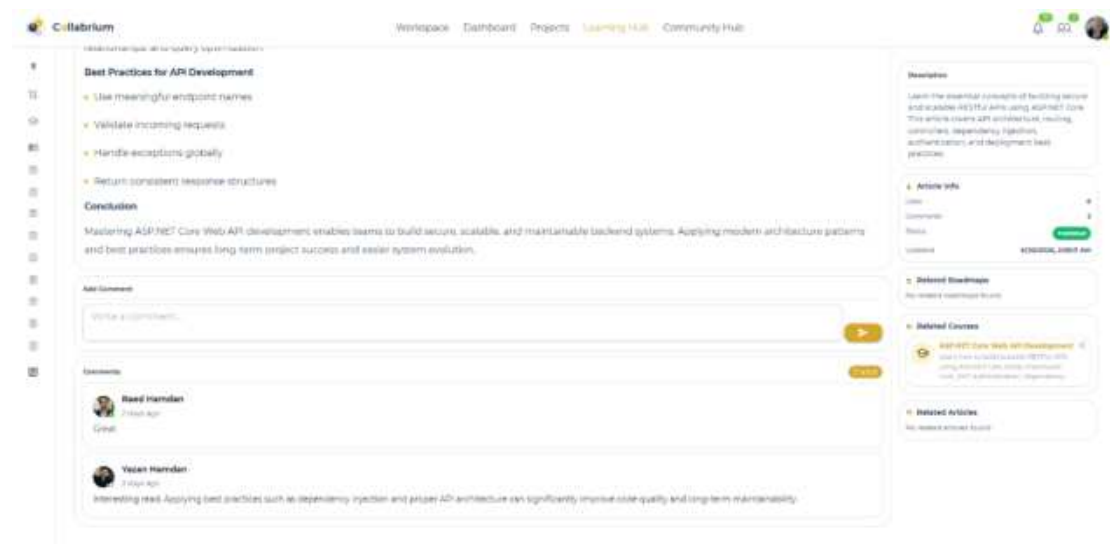


Figure 6.2.8 26 Article Comments Section

## 6.2.9 Community Hub Interfaces

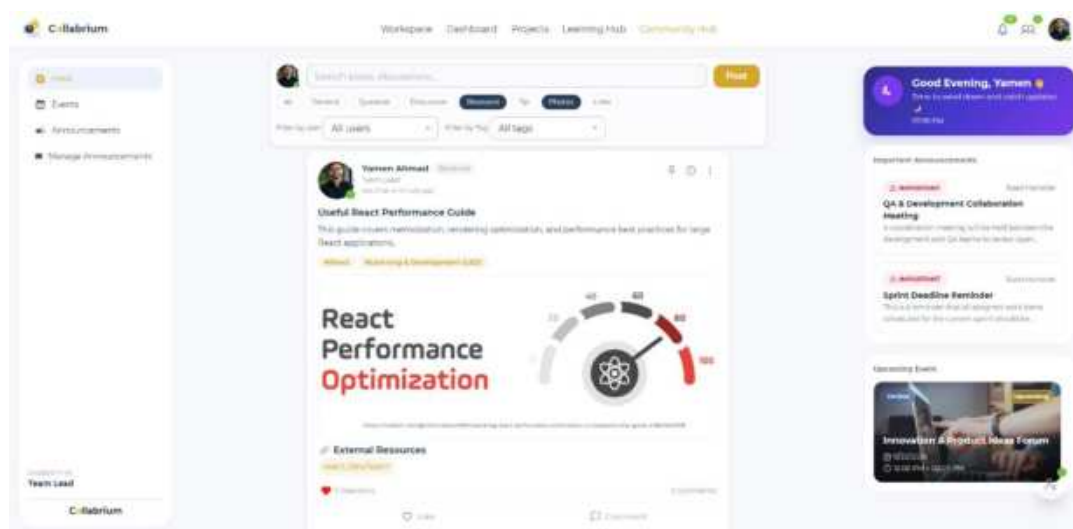


Figure 6.2.9 1 Community Feed Interface



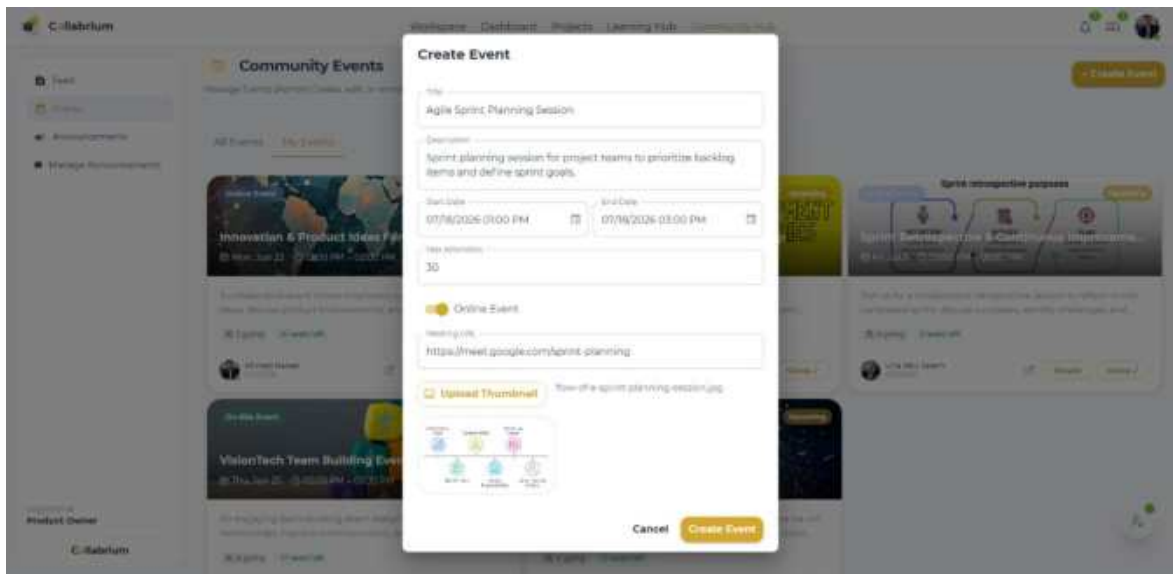


Figure 6.2.9 5 Create & Edit Event Dialog

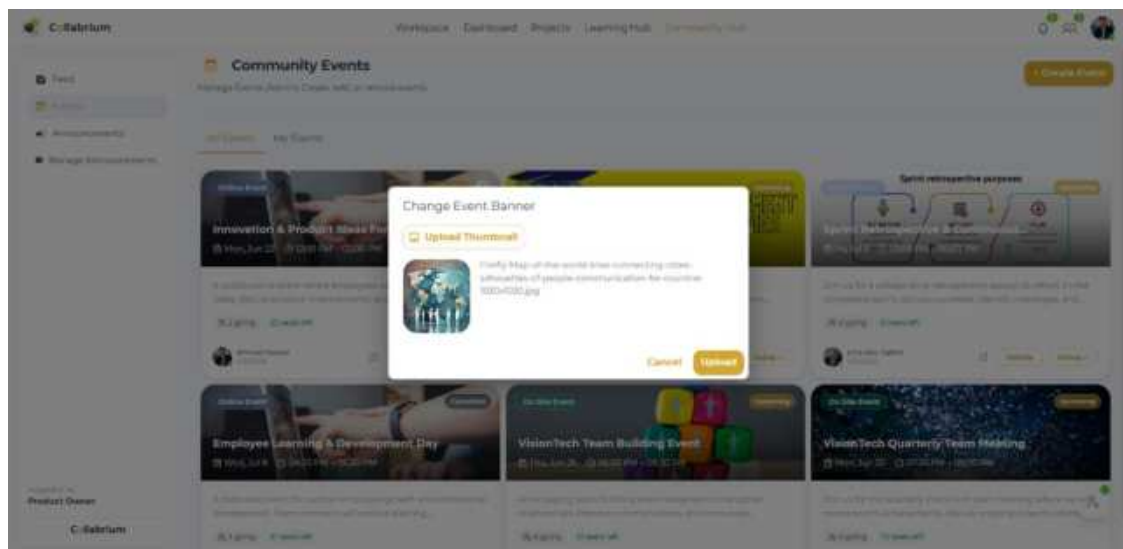


Figure 6.2.9 6 Change Event Banner Dialog

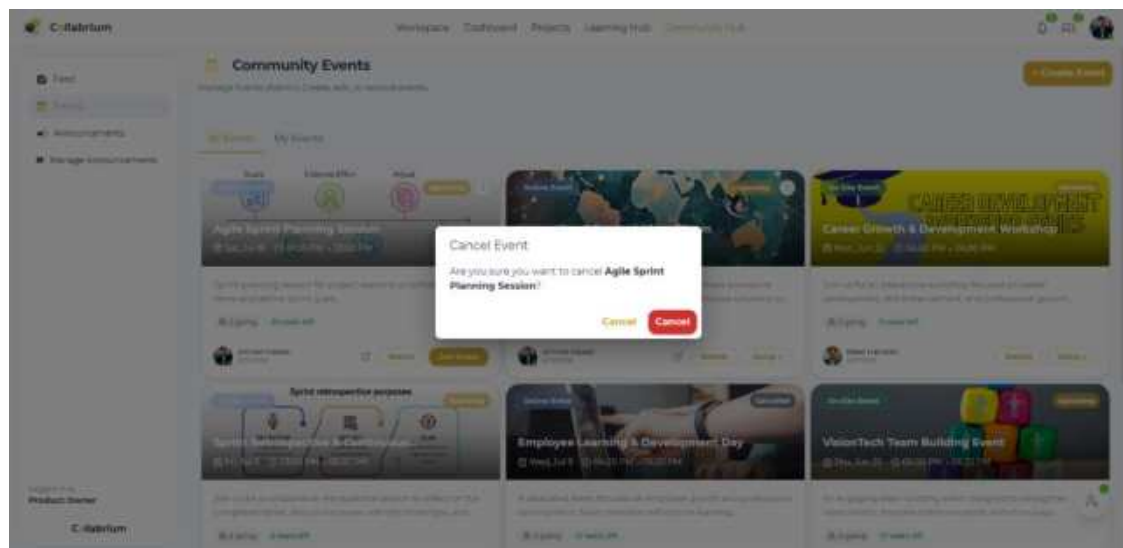


Figure 6.2.9 7 Cancel Event Dialog

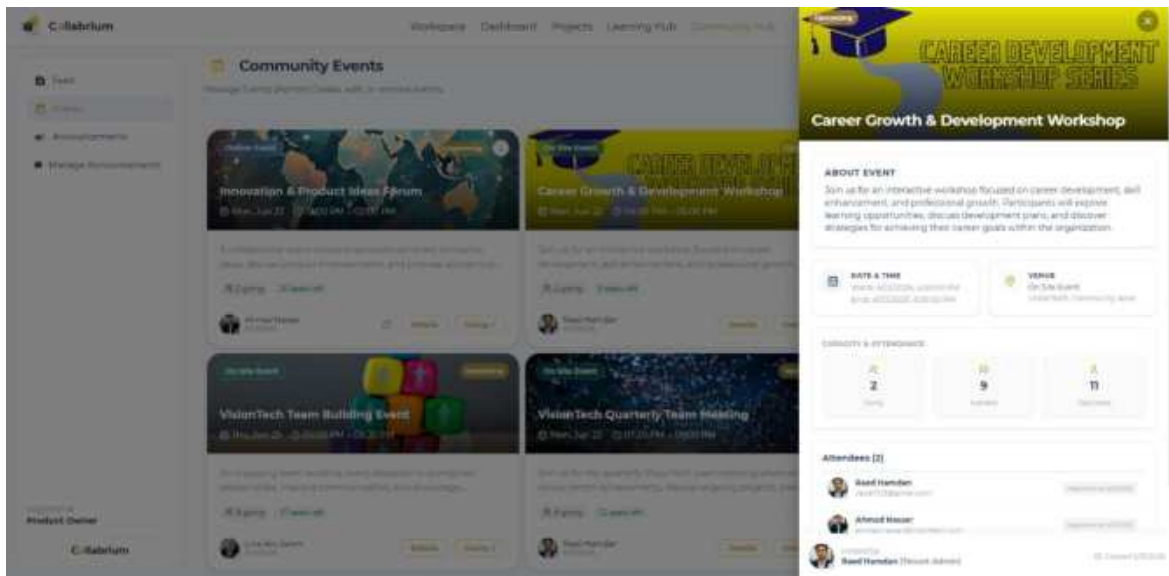


Figure 6.2.9 8 Detailed Event Drawer

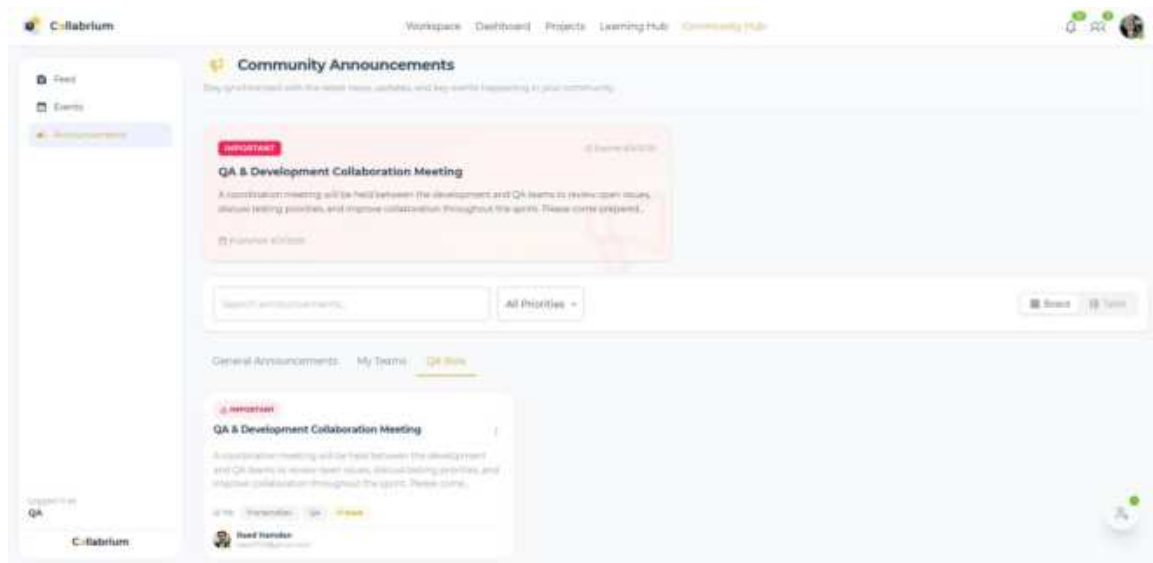


Figure 6.2.9 9 Tenant Member- Announcements Interface

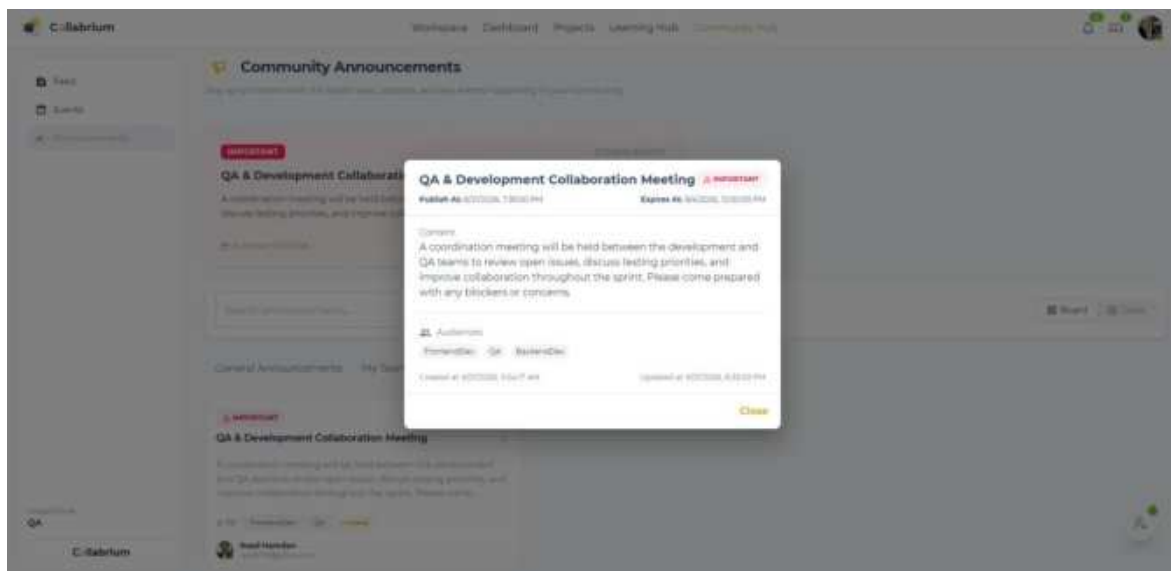


Figure 6.2.9 10 Announcement Details Dialog

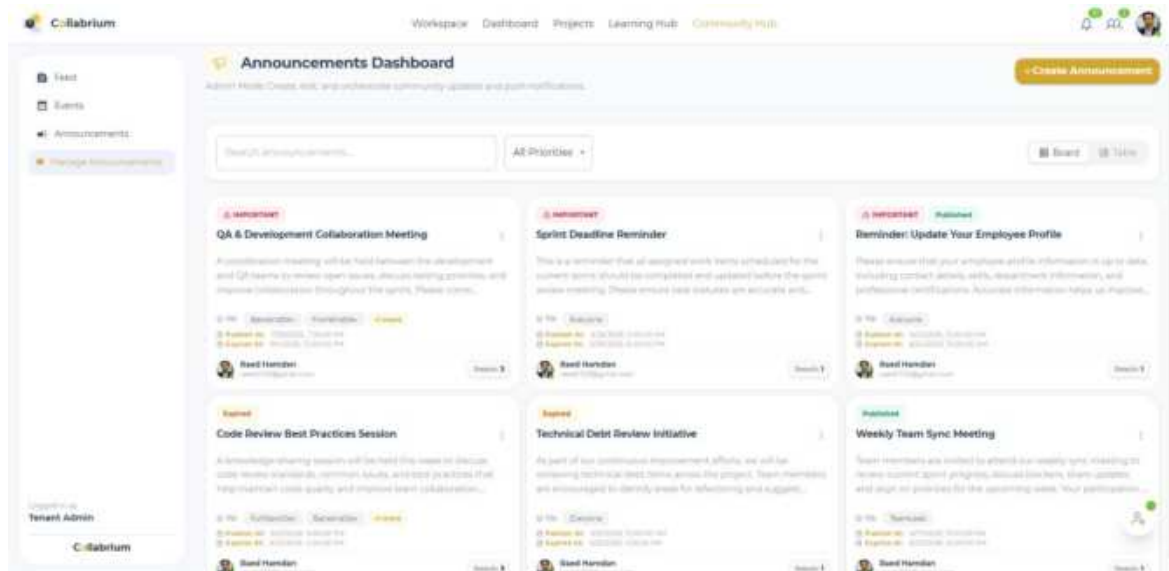


Figure 6.2.9 11 Tenant Manager- Announcements Dashboard Card View

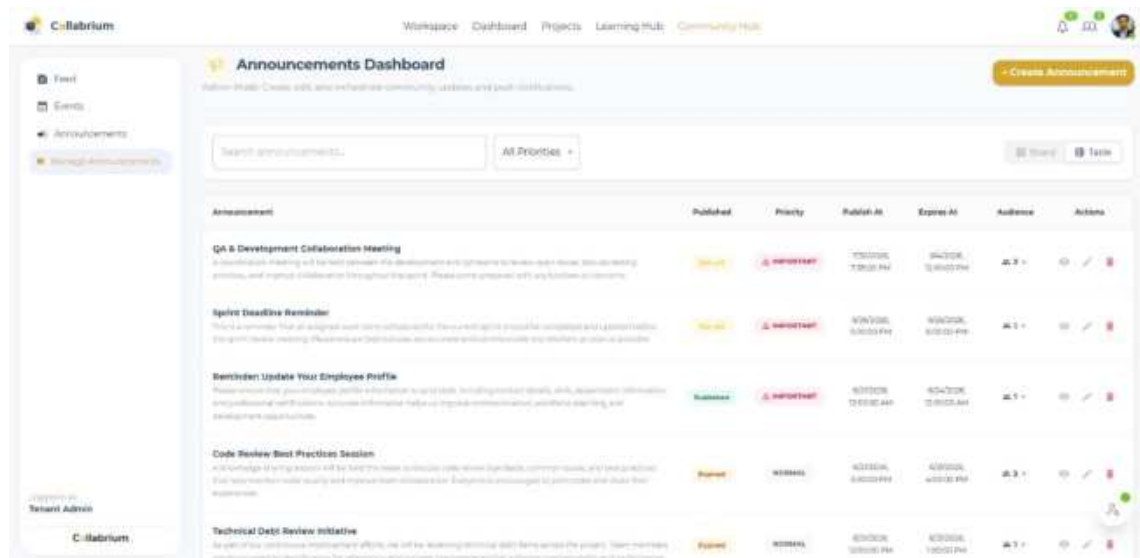


Figure 6.2.9 12 Tenant Manager- Announcements Dashboard Table View

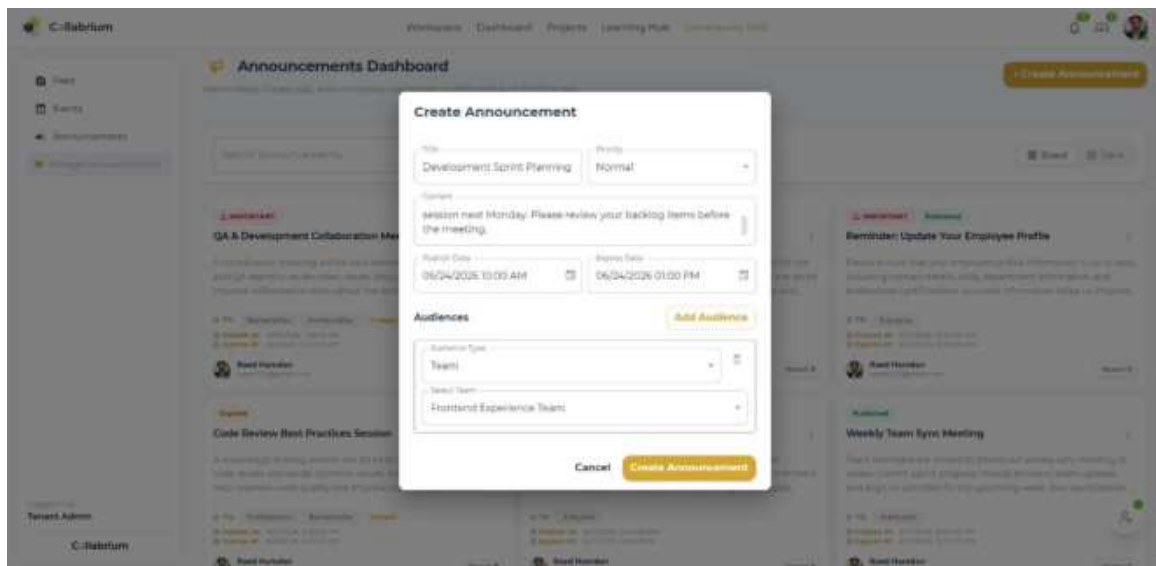


Figure 6.2.9 13 Create & Edit Announcement Dialog

## CHAPTER SEVEN

### **FUTURE VISION**

This chapter presents the future vision of the Collabrium platform and discusses the potential directions for system expansion, adoption, commercialization, and long-term development.

## 7.1 System Adoption and Evaluation

The system will be introduced to selected organizations and teams for real-world use and evaluation. User feedback will be collected to assess system performance, usability, and effectiveness in supporting project management, workforce development, and organizational communication. The collected feedback will be used to guide future improvements and enhancements.

## 7.2 Marketing the System

The next step is to market the system to organizations, companies, and start-ups interested in improving collaboration, learning, and project management processes. Interested clients can request a demonstration of the system and deploy it within their own organizational environment. Ongoing maintenance, technical support, and future updates will be provided to ensure reliable operation and continuous improvement.

## 7.3 Subscription and Payment System

A future enhancement of Collabrium is the integration of a subscription and payment system that enables organizations to subscribe to the platform according to their needs. A future enhancement of Collabrium is the integration of a subscription and payment system that enables organizations to subscribe to the platform according to their needs.

## References

- [1] "<https://www.atlassian.com/software/jira>," [Online].
- [2] "<https://azure.microsoft.com/en-us/products/devops>," [Online].
- [3] "<https://www.notion.com/product>," [Online].
- [4] "<https://trello.com>," [Online].
- [5] "<https://slack.com>," [Online].
- [6] "<https://www.atlassian.com/agile/project-management>," [Online].
- [7] "<https://www.ibm.com/think/topics/artificial-intelligence>," [Online].
- [8] "<https://learn.microsoft.com/en-us/aspnet/core/web-api>," [Online].
- [9] "<https://learn.microsoft.com/en-us/dotnet/architecture/modern-web-apps-azure/common-web-application-architectures>," [Online].
- [10] "[www.jwt.io/introduction](http://www.jwt.io/introduction)," [Online].
- [11] "<https://react.dev>," [Online].
- [12] "<https://mui.com>," [Online].
- [13] "<https://swagger.io/specification/>," [Online].
- [14] "<https://www.zaproxy.org>," [Online].
- [18] "<https://www.zaproxy.org>," [Online].