Software Requirements Specification

for LearnLance

**Version 1.0 approved.**

**Prepared by:**

**Insha Samnani (20K-0247)**

**Ismail Ahmed Ansari (20K-0228)**

**Yusra Adam (20K-0207)**

**FAST-NUCES**

**12th December, 2023**

**Table of Contents**

[**1.** **Introduction** 6](#_Toc153401309)

[**1.1** **Purpose** 6](#_Toc153401310)

[**1.2** **Document** 6](#_Toc153401311)

[**1.3** **Intended Audience and Reading Suggestions** 6](#_Toc153401312)

[**1.4** **Product Scope** 7](#_Toc153401313)

[**1.4.1** **Project Justification:** 7](#_Toc153401314)

[**1.4.2** **Brief Description of Project’s Product:** 7](#_Toc153401315)

[**1.4.3** **Summary of Project Deliverables:** 9](#_Toc153401316)

[**1.4.4** **Statement of Project Success Criteria:** 10](#_Toc153401317)

[**1.4.5** **Project Exclusion:** 10](#_Toc153401318)

[**1.4.6** **Constraints:** 11](#_Toc153401319)

[**1.4.7** **Assumptions:** 11](#_Toc153401320)

[**1.5** **References** 12](#_Toc153401321)

[**2.** **Overall Description** 13](#_Toc153401322)

[**2.1** **Product Perspective** 13](#_Toc153401323)

[**2.2** **Product Functions** 13](#_Toc153401324)

[**2.3** **User classes and Characteristics** 14](#_Toc153401325)

[**2.4** **Operating Environment** 15](#_Toc153401326)

[**2.4.1** **Hardware Platform:** 15](#_Toc153401327)

[**2.4.2** **Operating System:** 15](#_Toc153401328)

[**2.4.3** **Software Components:** 15](#_Toc153401329)

[**2.4.4** **Recommendation Algorithm Library:** 15](#_Toc153401330)

[**2.5** **Design and Implementation Constraints** 15](#_Toc153401331)

[**2.5.1** **Technological Compatibility:** 15](#_Toc153401332)

[**2.5.2** **External APIs:** 16](#_Toc153401333)

[**2.5.3** **Database Structure:** 16](#_Toc153401334)

[**2.5.4** **Development Framework:** 16](#_Toc153401335)

[**2.5.5** **Documentation Standard Constraint:** 16](#_Toc153401336)

[**2.6** **User Documentation** 16](#_Toc153401337)

[**2.7** **Assumptions and Dependencies** 17](#_Toc153401338)

[**3.** **External Interface Requirements** 19](#_Toc153401339)

[**3.1** **User Interfaces** 19](#_Toc153401340)

[**3.2** **Hardware Interfaces** 21](#_Toc153401341)

[**3.3** **Software Interfaces** 21](#_Toc153401342)

[**3.4** **Communication Interfaces** 22](#_Toc153401343)

[**4.** **System Features** 24](#_Toc153401344)

[**4.1** **Registration/Sign in** 24](#_Toc153401345)

[**4.2** **Notifications** 24](#_Toc153401346)

[**4.3** **Schedule Meetings (Calendar)** 25](#_Toc153401347)

[**4.4** **Discussion/Community Forum** 25](#_Toc153401348)

[**4.5** **Chat/Group Chat** 26](#_Toc153401349)

[**4.6** **Call/video Call** 26](#_Toc153401350)

[**4.7** **Create Profile** 27](#_Toc153401351)

[**4.8** **Recommendations** 27](#_Toc153401352)

[**4.9** **Take Courses/Take Sessions** 28](#_Toc153401353)

[**4.10** **Payment** 28](#_Toc153401354)

[**4.11** **Certificates/Badges** 29](#_Toc153401355)

[**4.12** **Post Achievements** 29](#_Toc153401356)

[**4.13** **Student Connecting With Other Students and Following Teachers** 30](#_Toc153401357)

[**4.14** **Bidding** 30](#_Toc153401358)

[**4.15** **Add Courses and Schedule Live Sessions** 31](#_Toc153401359)

[**4.16** **Joint Accounts** 31](#_Toc153401360)

[**4.17** **Remove Students/Teachers** 31](#_Toc153401361)

[**4.18** **Modify Students/Teachers** 32](#_Toc153401362)

[**4.19** **Add Students/Teachers** 32](#_Toc153401363)

[**4.20** **View Revenue** 33](#_Toc153401364)

[**4.21** **Create Posts** 33](#_Toc153401365)

[**4.22** **Create Comments/Like on Posts** 34](#_Toc153401366)

[**4.23** **Virtual Classrooms** 34](#_Toc153401367)

[**5.** **Other Nonfunctional Requirements** 35](#_Toc153401368)

[**5.1** **Performance Requirements** 35](#_Toc153401369)

[**5.2** **Availability Requirements** 35](#_Toc153401370)

[**5.3** **Installability Requirements** 36](#_Toc153401371)

[**5.4** **Integrity Requirements** 36](#_Toc153401372)

[**5.5** **Interoperability Requirements** 37](#_Toc153401373)

[**5.6** **Robustness Requirements** 37](#_Toc153401374)

[**5.7** **Reliability Requirements** 38](#_Toc153401375)

[**5.8** **Safety Requirements** 38](#_Toc153401376)

[**5.9** **Security Requirements** 38](#_Toc153401377)

[**5.10** **Usability Requirements** 39](#_Toc153401378)

[**5.11** **Efficiency Requirements** 39](#_Toc153401379)

[**5.12** **Modifiability Requirements** 39](#_Toc153401380)

[**5.13** **Supportability Requirements** 40](#_Toc153401381)

[**5.14** **Portability Requirements** 40](#_Toc153401382)

[**5.15** **Reusability Requirements** 40](#_Toc153401383)

[**5.16** **Scalability Requirements** 40](#_Toc153401384)

[**5.17** **Business Rules** 40](#_Toc153401385)

[**5.17.1** **Facts:** 40](#_Toc153401386)

[**5.1.2** **Constraints:** 41](#_Toc153401387)

[**5.1.3** **Action Enablers:** 42](#_Toc153401388)

[**5.1.4** **Inferences:** 42](#_Toc153401389)

[**5.1.5** **Computations:** 43](#_Toc153401390)

[**6.** **Other Requirements** 44](#_Toc153401391)

[**6.1** **Database Requirements** 44](#_Toc153401392)

[**6.2** **Internalization Requirements** 44](#_Toc153401393)

[**6.3** **Legal Requirements** 44](#_Toc153401394)

[**6.4** **Reuse Objectives** 44](#_Toc153401395)

[**7.** **Appendices** 44](#_Toc153401396)

[**7.1** **Appendix 1: Requirement Elicitation Technique** 44](#_Toc153401397)

[**7.1.1** **JRP Session:** 45](#_Toc153401398)

[**7.1.2** **Questionnaires:** 48](#_Toc153401399)

[**7.1.3** **Research:** 48](#_Toc153401400)

[**7.2** **Appendix 2: Other Requirements** 49](#_Toc153401401)

[**7.2.1** **Business Requirements:** 49](#_Toc153401402)

[**7.2.2** **Functional Requirements:** 50](#_Toc153401403)

[**7.2.3** **System Requirements:** 53](#_Toc153401404)

[**7.2.4** **Operating Environment Requirements:** 54](#_Toc153401405)

[**7.2.5** **Vision Statement:** 54](#_Toc153401406)

[**7.3** **Appendix 3: Requirements Models** 54](#_Toc153401407)

[**7.3.1** **Business Rules Constraints Table:** 54](#_Toc153401408)

[**7.3.2** **Business Rules Computations Table:** 55](#_Toc153401409)

[**7.3.3** **Business Rules Catalog:** 56](#_Toc153401410)

[**7.3.4** **Decision Tree for Badges Assignment to Students:** 64](#_Toc153401411)

[**7.3.5** **Pareto Bar Chart and Problem Statement Table:** 65](#_Toc153401412)

[**7.3.6** **Ishikawa Diagram:** 66](#_Toc153401413)

[**7.3.7** **Context Diagram:** 67](#_Toc153401414)

[**7.3.8** **DFD Level 0 Diagram:** 68](#_Toc153401415)

[**7.3.9** **ERD Diagram:** 68](#_Toc153401416)

[**7.3.10** **Feature List:** 69](#_Toc153401417)

[**7.3.11** **Data Dictionary:** 69](#_Toc153401418)

[**7.3.12** **Wireframes and Prototypes:** 81](#_Toc153401419)

[**7.3.13** **Report Specification Template for Payment Invoice:** 84](#_Toc153401420)

[**7.3.14** **Use Case Diagram and Use Case Narratives:** 85](#_Toc153401421)

[**7.3.15** **Use Case Glossary:** 108](#_Toc153401422)

[**7.3.16** **User Stories:** 111](#_Toc153401423)

[**7.3.17** **Event List:** 113](#_Toc153401424)

[**7.3.18** **Activity Diagram:** 116](#_Toc153401425)

[**7.3.19** **State Transition Diagrams:** 117](#_Toc153401426)

[**7.3.20** **Crud Matrix:** 130](#_Toc153401427)

[**7.3.21** **Data Structures:** 131](#_Toc153401428)

[**7.3.22** **Scenarios:** 137](#_Toc153401429)

[**7.3.23** **Story Boards:** 138](#_Toc153401430)

[**7.3.24** **User Personas:** 139](#_Toc153401431)

[**7.3.25** **Stakeholders Classification:** 141](#_Toc153401432)

[**7.4** **Appendix 3: Requirement Negotiation and Prioritization** 141](#_Toc153401433)

[**7.4.1** **Requirement Negotiation:** 141](#_Toc153401434)

[**7.4.2** **Requirement Prioritization:** 142](#_Toc153401435)

# **Introduction**

## **Purpose**

The purpose of this Software Requirements Specifications (SRS) document is to outline the development requirements for "LearnLance," an online platform aimed at simplifying educational and professional interactions between students and teachers. This document specifically pertains to the initial release of LearnLance, identified as version 1.0.

LearnLance is envisioned as a comprehensive platform integrating educational, professional, and networking functionalities. The scope of this SRS covers both front-end and backend components, along with external integrations.

*Please refer to Appendix* ***7.3.5*** *and* ***7.3.6*** *that presents the pareto bar chart and problem statement table and Ishikawa diagram respectively.*

## **Document**

1. Headings: Headings and subheadings should be used to organize the document. For

* Heading 1: Arial and 18 points, bold.
* Heading 2: Arial and 16 points, bold.
* Heading 3: Arial and 14 points, bold.
* Heading 3: Arial and 12 points, bold.

1. Body Text: Similarly, Arial is used for body text. The font size for the main content is commonly set between 10 (usually for tables) and 11 points (usually for text content).

## **Intended Audience and Reading Suggestions**

Various categories of intended users, each accompanied by their respective descriptions, are presented in the table below. Additionally, the recommended sections of the Software Requirements Specification (SRS) that may capture the interest of these users are indicated. The SRS follows a Hierarchical Numbering structure.

|  |  |  |
| --- | --- | --- |
| Intended Readers | Description | Intended Section to Read |
| Sponsor | The person who will be responsible for funding LearnLance will review this SRS to finalize the requirements and sign off. | Section 1-6  Appendix |
| Project Manager | The Project Manager will ensure that the business analyst has identified the requirements, prioritize them, and specify the associated costs and time required for completing LearnLance. | Section 1-6  Appendix |
| Business Analysts | The individual who will gather and analyze requirements from various stakeholders through different elicitation processes will document them in this SRS for future reference. | Section 1-6  Appendix |
| Students | The student using the SRS will comprehend all the requirements, including imposed constraints. | Section 1-6  Appendix |
| Teachers | The teacher using the SRS will comprehend all the requirements, including imposed constraints. | Section 1-6  Appendix |
| Developers | Developers will read this SRS to understand the requirements and code accordingly. | Section 2-6  Appendix |
| Testers | Testers will use this SRS to review the non-functional requirements and test all specified requirements. | Section 4-5  Appendix |
| Marketers | Marketers will use the SRS to understand the business objectives and target audience, enabling them to effectively market LearnLance. | Section 1 |
| Subject Matter Experts | The person responsible for ensuring that the advice given on implementation is accurately documented by business analysts will review this SRS. | Section 2-5 |
| JAD Session Leader | JAD session leaders will refer to this SRS to understand the priorities set by the Project Manager and subsequently create the agenda for JAD sessions. | Section 1-6 |
| Scribes | Scribes will use this SRS to check existing documentation and then report all transcriptions to the business analyst. | Section 1-6 |
| UX/UI Designers | UX/UI designers will utilize this SRS to comprehend all user and functional requirements, allowing them to create prototypes and designs for developers and customer review. | Appendix |

The SRS is organized in Hierarchical Numbering.

## **Product Scope**

### **Project Justification:**

The LearnLance platform represents a self-contained and innovative solution for online learning and teaching, with the integration of freelancing elements. Traditional education systems often lack the flexibility and personalization required for effective learning, while traditional freelancing platforms may lack the structured educational environment necessary for skill development. LearnLance addresses this critical gap by offering a versatile and interconnected learning ecosystem, enabling personalized learning, peer-to-peer collaboration, and skill development in an integrated manner. LearnLance is driven by the pressing need to revolutionize the educational experience, combining the best elements of online learning, and freelancing to create a holistic and connected environment.

### **Brief Description of Project’s Product:**

*Please refer to Appendix* ***7.1*** *that presents the JRP session script, research, and questionnaire served as the requirement gathering technique used to collect all the necessary requirements from the users.*

UR-1: Students can create their profiles.

UR-2: Teachers can create their profiles.

UR-3: Students can edit their profiles.

UR-4: Teachers can edit their profiles.

UR-5: Students can view their own profiles.

UR-6: Teachers can view their own profiles.

UR-7: Students can view the profiles of other students.

UR-8: Students can view the profiles of other teachers.

UR-9: Teachers can view the profiles of other students.

UR-10: Teachers can view the profiles of other teachers.

UR-11: Students can send connection requests to other students.

UR-12: Students can follow teachers.

UR-13: Teachers can send connection requests to other teachers.

UR-14: Students can view their connections.

UR-15: Teachers can view their connections.

UR-16: Teachers can view their followers.

UR-17: Teachers can create courses.

UR-18: Teachers can draft courses.

UR-19: Teachers can publish courses.

UR-20: Teachers can view the courses they have posted.

UR-21: The system can offer personalized course recommendations.

UR-22: Students can browse courses.

UR-23: Students can enroll in courses.

UR-24: System can provide certifications for completing courses.

UR-25: System can provide badges for completing a certain number of quizzes.

UR-26: Achievements are made visible on the student’s profile.

UR-27: Certificates are made visible on the student’s profile.

UR-28: Students can post their thoughts and achievements to share with their connections.

UR-29: Teachers can post their thoughts and achievements to share with their connections.

UR-30: Students can like the posts of their connections.

UR-31: Students can like the posts of their followers.

UR-32: Students can comment on the posts of their connections.

UR-33: Students can comment on the posts of their followers.

UR-34: Teachers can like the posts of their connections.

UR-35: Teachers can comment on the posts of their connections.

UR-36: Students can post topic requests.

UR-37: The system can offer personalized job recommendations.

UR-38: Teachers can browse topic requests.

UR-39: Teachers can create proposals for topic requests.

UR-40: Teachers can bid on topic requests from students seeking personalized learning sessions.

UR-41: Teachers can view proposals they have initiated.

UR-42: Students can review teacher bids.

UR-43: Students can select the preferred teacher for personalized learning.

UR-44: Students can end the contract after fulfilling the requirements of a topic request.

UR-45: Teachers can end the contract after fulfilling the requirements of a topic request.

UR-46: Students can view their selected items in the cart.

UR-47: Administrators can approve payments made by students for courses through their individual accounts.

UR-48: Administrators can approve payments made by students for topic requests through their individual accounts.

UR-49: Teachers can schedule live video conferencing sessions.

UR-50: Teachers can schedule interactive webinars.

UR-51: Teachers can initiate meetings.

UR-52: Teachers can end meetings.

UR-53: Students can browse upcoming sessions.

UR-54: Teachers can schedule reminders on the calendar.

UR-55: Teachers can delete reminders on the calendar.

UR-56: Students can schedule reminders on the calendar.

UR-57: Students can delete reminders on the calendar.

UR-58: Students can participate in live video conferencing sessions.

UR-59: Students can participate in interactive webinars.

UR-60: Students can browse the list of communities.

UR-61: Teachers can browse the list of communities.

UR-62: Students can create community forums.

UR-63: Teachers can create community forums.

UR-64: Students can send requests to join a community.

UR-65: Teachers can send requests to join a community.

UR-66: Students can invite their connections to the community they have joined.

UR-67: Teachers can invite their connections to the community they have joined.

UR-68: The creator of the community can accept requests from students and teachers to join.

UR-69: The creator of the community can decline requests from students and teachers to join.

UR-70: Joined participants of the community can create posts in the community.

UR-71: Students can message other students.

UR-72: Students can message other teachers.

UR-73: Teachers can message other students.

UR-74: Teachers can message other teachers.

UR-75: Students can call other students.

UR-76: Students can call other teachers.

UR-77: Teachers can call other students.

UR-78: Teachers can call other teachers.

UR-79: Students can rate courses.

UR-80: Students can provide feedback on courses.

UR-81: Teachers can rate students upon completion of topic requests.

UR-82: Teachers can send joint account requests to other teachers to collaborate in teaching a course together.

UR-83: Teachers can view the joint account requests they have received.

UR-84: Teachers can accept joint account requests from other teachers to collaborate in teaching a course together.

UR-85: Teachers can rate other teachers upon completion of a joint account for courses.

UR-86: Teachers can provide feedback to other teachers upon completion of a joint account for courses.

UR-87: Administrators can delete any student from LearnLance.

UR-88: Administrators can delete any teacher from LearnLance.

UR-89: Administrators can view monthly revenue generation.

UR-90: Administrators can view the number of unique students on LearnLance per month.

UR-91: Administrators can view the number of unique teachers on LearnLance per month.

UR-92: Students can view the LearnLance privacy and policy.

UR-93: Students can view the LearnLance FAQs.

UR-94: Teachers can view the LearnLance privacy and policy.

UR-95: Teachers can view the LearnLance FAQs.

*Please refer to Appendix* ***7.3.14,******7.3.15, 7.3.16, 7.3.20*** *that presents the use case diagram and narratives, use case glossary, user stories, and crud matrix respectively.*

*Please refer to Appendix* ***7.4.1*** *that presents the requirement negotiation.*

*Please refer to Appendix* ***7.4.2.1*** *that presents the user requirement prioritization.*

### **Summary of Project Deliverables:**

**By 1st January 2024 - 30th January 2024:**

1. Brainstorm ideas and collect requirements through questionnaires and conduct JRP sessions.
2. Present the scope and feasibility of LearnLance through a proposal and presentation.

**By 1st February 2024 - 28th February 2024:**

1. Format requirements through document specification (SRS/SDS).
2. Design the database for LearnLance.

**By 1st March 2024 - 30th March 2024:**

1. Design the basic design and wireframes for LearnLance.
2. Prototype LearnLance based on wireframes.

**By 1st April 2024 - 30th April 2024:**

1. Perform backend and frontend development of the following features based on the LearnLance prototype:

* Set up students’ and teachers’ profiles.
* Course management.
* Community forums.
* Administrative tools.
* Achievements and rewards.
* Integrate meeting API for virtual classrooms.
* Joint accounts.

**By 1st May 2024 - 15th July 2024:**

1. Continue backend and frontend development for the remaining features of LearnLance based on LearnLance prototypes.

* Chat and call functionality.
* Bidding system.
* Payment system.
* Social hub.
* Schedule meetings.
* Integrate chat and call APIs.
* Recommendation algorithms for course and job recommendations.

**By 16th July 2024 - 15th August 2024:**

1. Integrate LearnLance components.
2. Conduct Quality Assurance (QA) final testing.
3. Develop manual and automated test cases for the testing of LearnLance.

**By 16th August 2024 - 15th September 2024:**

1. Deploy the Android version on Expo.
2. Conduct beta testing.
3. Perform maintenance of the LearnLance system.
4. Final presentation and report.

### **Statement of Project Success Criteria:**

SM-1: Achieve a 50% increase in the number of registered students and teachers within the first year compared to traditional learning platforms like Coursera and Upwork, demonstrating substantial platform growth.

SM-2: Achieve a LearnLance user retention rate of 70% after six months of the LearnLance platform's official launch, indicating enhanced user experience.

SM-3: At least 90% of the registered teachers actively offer courses or services, indicating a high level of engagement.

SM-4: At least 80% of registered students actively participate in courses and utilize LearnLance's services, indicating a high level of student engagement.

### **Project Exclusion:**

PE-1: LearnLance does not involve the creation or development of specific course content, curriculum, or educational materials. It is not responsible for generating course materials or textbooks.

PE-2: LearnLance platform does not engage in e-commerce for physical products, such as selling physical books, learning materials, or any tangible goods.

PE-3: LearnLance platform does not include in-person or physical educational services, such as traditional classroom-based instruction, on-site workshops, or physical tutoring sessions.

### **Constraints:**

CO-1: LearnLance operates within a budget of $1,000,000, which must be adhered to for all development and implementation phases.

CO-2: LearnLance requires strict alignment with predefined milestone dates for development and launch to ensure project progress and timely delivery.

CO-3: LearnLance requires a non-relational database due to its unstructured nature, facilitating the storage of posts, courses, and other content.

CO-4: Frontend of LearnLance will be implemented using React Native, ensuring cross-platform compatibility and a seamless user experience.

CO-5: Backend APIs will be based on Node.js, providing a robust and scalable foundation for platform functionality.

CO-6: LearnLance will integrate third-party services for calls, messages, and virtual classrooms using the Get Stream open-source API, ensuring efficient and reliable communication features.

CO-7: The system’s design, code, and maintenance documentation shall conform to the *Process Impact Intranet Development Standard*, Version 1.3.

### **Assumptions:**

AS-1: Students and teachers have reliable internet access; otherwise, they will face limited platform access due to poor connectivity.

AS-2: Students and teachers possess Android smartphones; otherwise, they will experience restricted platform access.

AS-3: Students and teachers have pre-existing email addresses for registration; otherwise, they will encounter registration challenges without email addresses.

AS-4: Students and teachers possess basic English language skills for platform interaction; otherwise, they will experience usability issues as non-English speakers.

AS-5: Students and teachers interacting with the platform are human individuals, not automated bots; otherwise, potential automated bot interference may affect platform security and user experience.

AS-6: Students and teachers follow standard security practices, and the platform adheres to data protection standards; otherwise, data breaches, privacy concerns, loss of their trust, and compromised user data may occur in the case of security compliance failure.

* + 1. **Dependencies:**

DE-1: LearnLance relies on utilizing front-end frameworks and design principles that facilitate responsiveness, ensuring a consistent user experience across diverse platforms.

DE-2: The successful implementation of LearnLance depends on the availability of a reliable and scalable database server, with compatibility for the chosen database management system.

DE-3: LearnLance relies on Integration and availability of third-party APIs for chat, call, live sessions, and payments.

DE-4: LearnLance depends on adherence to government rules and regulations for tax implications during payment transactions.

DE-5: LearnLance relies on Successful collaboration and coordination across all roles within the development team.

DE-6: The operation of LearnLance depends on changes being made in the Payroll System to accept payment requests for courses and topic requests with LearnLance.

*Please refer to Appendix* ***2.7*** *that presents the detailed assumptions and dependencies.*

*Please refer to Appendix* ***7.2*** *for detailed information on vision, business requirements, functional requirements, system requirements, operating environment requirements and sources of requirements.*

*Please refer to Appendix* ***7.3.6, 7.3.7, 7.3.10, 7.3.17*** *for the Ishikawa diagram, context diagram, and feature list, and event list which provide a detailed analysis of the problem, vision, and scope.*

## **References**

|  |  |
| --- | --- |
| [1] | K. Wiegers, Software Requirements(Developer Best Practices), Redmond, WA: Microsoft Press, August 15, 2013. |

|  |  |
| --- | --- |
| [2] | [Online], Available: https://www.digitalsamba.com/blog/best-free-video-conferencing-api-for-websiteintegration. |

|  |  |
| --- | --- |
| [3] | [Online], Available: https://developers.dwolla.com/api-reference#tools. |

|  |  |
| --- | --- |
| [4] | [Online], Available://www.weavy.com/free-chat |

|  |  |
| --- | --- |
| [5] | [Online], Available: https://forms.gle/BF9TmHUCoFnSqxAA8 |

# **Overall Description**

## **Product Perspective**

LearnLance is a pioneering and self-contained online learning and freelancing platform that emerged in response to the challenges posed by the fragmented landscape of existing educational and freelancing platforms. Unlike current systems that lack cohesive integration, LearnLance is a standalone product designed to be a unified solution. It is not a replacement for existing systems but aims to redefine the user experience by seamlessly combining social networking, learning, and business elements within a single platform. LearnLance stands alone as a comprehensive solution, eliminating the need for users to navigate multiple platforms for educational and freelancing purposes. This system's perspective is self-contained, offering a unique space where students, educators, and freelancers can access a diverse range of educational services, job opportunities, and networking functionalities in an integrated manner. The platform's major components, subsystem interconnections, and external interfaces will be detailed to provide a clear understanding of its holistic structure and functionality in the larger digital ecosystem.

*Please refer to Appendix* ***7.2.1*** *and* ***7.3.7*** *that presents the business requirements and context diagram respectively.*

## **Product Functions**

LearnLance is designed to offer a comprehensive and unified experience, bringing together students, teachers, and freelancers on a single platform. The product functions are structured to cater to the diverse needs of users, aiming to streamline educational and professional endeavors. From profile management to communication features, a robust bidding system, and immersive virtual classrooms, LearnLance prioritizes user engagement and efficiency. Payment processing, administrative tools, and a recommendation engine further enhance the overall experience, while social hubs, achievement rewards, and joint accounts foster a sense of community and collaboration. LearnLance's functions are intricately designed to create a seamless and integrated ecosystem. The product functions are discussed below:

* Setting up Students and Teachers Profiles: Students and teachers can customize their profiles with personal information, educational achievements, and interests.
* Course Management: Teachers can post courses, and students can enroll in them.
* Chat and Call: Students and teachers can communicate through text chat and video calls.
* Bidding System: Students can post job requests, teachers can bid on them, and students can select a teacher for their needs.
* Payment: Students can make payments for enrolling in paid courses, and they can pay teachers after hiring them.

*Please refer to Appendix* ***7.3.13*** *that presents the report specification template for payment invoice.*

* Virtual Classrooms: Live classes are available for students after hiring teachers, and live sessions are conducted by teachers.
* Community Forums: Students can engage in discussions, seek assistance, and share knowledge.
* Administrative Tools: Admins can manage students and teachers, including adding, removing, and deleting them. Admins can also issue certificates.
* Recommendation Engine: The platform provides course recommendations to students and job recommendations to teachers.
* Social Hub: Students can connect with other students, teachers can connect with other teachers, and students can follow teachers.
* Achievements and Rewards: Students receive rewards upon completing certificates or sessions based on specific criteria.
* Joint Account: Teachers can conduct live sessions and series in collaboration with other teachers.
* View Revenue: Admins can view revenue generated on the platform.
* Schedule Meetings: Students and teachers can schedule meetings for various purposes.

*Please refer to Appendix* ***7.3.8, 7.3.18, 7.3.19,*** *and* ***7.3.9*** *that presents the DFD level 0 diagram, activity diagram, state transition diagrams and ERD diagram respectively.*

## **User classes and Characteristics**

LearnLance caters to diverse user classes, each with unique characteristics and needs. The identification of these user classes is crucial for tailoring the product to meet varying requirements. The primary user classes anticipated for LearnLance are:

|  |  |
| --- | --- |
| User Class | Description |
| Students (Favored) | Students will interact with LearnLance to fulfil their educational and freelancing needs. They can interact with the system by creating a student profile, managing courses (viewing courses, enrolling a course, topic requests), maintaining social hub (connections with other students and teachers), managing bids, gaining certifications, getting personalized recommendations for courses, participating in community forums, enrolling in virtual classrooms and webinars. The user class of students constitutes a significant user base as they actively engage with the core functionalities of the system. The needs and requirements for student's user class involve seamless access to educational content, effective communication, simple and intuitive profile management. |
| Teachers (Favored) | Teachers will interact with LearnLance to utilize educational services and job opportunities. Teachers can interact with the system by creating and managing a teacher profile displaying their experience, certifications, projects, reviews from peer students and teachers, managing social connections with peer students and teachers, offering and publishing educational courses, avail personalized job recommendations based on personal preferences and profile, scheduling live sessions and webinars. The needs and requirements for a teacher’s user class involve efficient course management tools, effective communication and bidding system for job opportunities. |
| Admin (Ignored) | Admins interact with LearnLance as platform overseers. They perform control operations by deleting any student or teacher from the platform, viewing monthly revenue, issuing certifications, controlling payment approvals and performing high level access and control. The needs and requirements involve robust administrative tools for performing access and control operations. |

*Please refer to Appendix* ***7.3.24, 7.3.25*** *that presents the user personas and stakeholders classification.*

## **Operating Environment**

LearnLance is designed to operate within a dynamic and adaptable environment, ensuring compatibility and seamless functionality across various components. The operating environment for LearnLance encompasses the following aspects:

### **Hardware Platform:**

LearnLance is developed to run on a variety of hardware platforms, including smartphones and tablets. The platform aims to provide a consistent and user-friendly experience across different devices, accommodating the diverse preferences of its user base.

### **Operating System:**

The LearnLance mobile application, built using React Native, is compatible with Android operating systems. The Android app supports a range of versions, ensuring accessibility for users with different device capabilities.

### **Software Components:**

LearnLance relies on key software components such as Node.js for backend development and MongoDB as the database. The document-based structure of MongoDB facilitates efficient handling of unstructured data, ensuring rapid processing even with large datasets. External APIs are integrated for payment processing and communication functionalities, enhancing the overall user experience.

### **Recommendation Algorithm Library:**

LearnLance incorporates a recommendation engine that relies on a dedicated algorithm library. This library is seamlessly integrated into the platform to provide personalized course suggestions for students and job recommendations for teachers, enhancing the overall user experience.

## **Design and Implementation Constraints**

LearnLance, while ambitious and versatile, operates within certain constraints that guide the design and implementation processes. These constraints are essential considerations for the development team:

### **Technological Compatibility:**

LearnLance is constrained by the need for compatibility across a diverse range of devices and operating systems. The platform must seamlessly operate on various hardware platforms, such as smartphones and tablets. Ensuring compatibility with different versions of Android is critical for providing a consistent user experience.

### **External APIs:**

The integration of external APIs for payment processing and communication introduces a constraint, as the functionality of LearnLance is dependent on the proper functioning and reliability of these external components. LearnLance will integrate third-party services for calls, messages, and virtual classrooms using the Get Stream open-source API, ensuring efficient and reliable communication features.

### **Database Structure:**

The document-based structure of MongoDB, chosen for its efficiency in handling unstructured data, imposes constraints on the database design. The development team must adhere to MongoDB's data model, potentially limiting flexibility in certain data storage scenarios.

### **Development Framework:**

The use of React Native as the primary development framework constrains the development team to adhere to its conventions. While offering cross-platform advantages, it may limit certain platform-specific optimizations and features. Backend APIs for LearnLance will be based on Node.js, providing a robust and scalable foundation for platform functionality.

### **Documentation Standard Constraint:**

The system’s design, code, and maintenance documentation shall conform to the Process Impact Intranet Development Standard, Version 1.3.

## **User Documentation**

The user documentation for LearnLance is designed to provide comprehensive support for users, facilitating a seamless and intuitive experience on the platform. The following components will be delivered along with the software:

* **User Manuals:** Comprehensive guides for both students and teachers on platform navigation, profile customization, course enrollment, bidding processes, and other key functionalities.
* **Online Help:** In-context helps accessible directly within the LearnLance platform, offering users immediate assistance while interacting with different features. This will include tooltips, contextual pop-ups, and informative links.
* **Tutorials:** Step-by-step tutorials for various actions such as setting up profiles, enrolling in courses, initiating topic requests, and conducting live sessions. These tutorials will be available in video format for visual learners.
* **Knowledge Base:** An online repository of frequently asked questions, troubleshooting guides, and best practices to address common queries and challenges encountered by users.

## **Assumptions and Dependencies**

The successful realization of LearnLance's business requirements is contingent upon several assumptions and dependencies that shape the platform's operational environment and functionality. These assumptions are based on the expected behavior and characteristics of the users engaging with the platform, while dependencies highlight the external factors and integrations critical for the platform's seamless operation.

**Assumptions:**

1. **Internet Access Reliability:** Students and teachers have reliable internet access; otherwise, they will face limited platform access due to poor connectivity. Poor connectivity could hinder user experience and access to educational and professional opportunities.
2. **Android Smartphone Possession:** Students and teachers possess Android smartphones; otherwise, they will experience restricted platform access. Lack of Android devices may limit user access to the platform.
3. **Pre-existing Email Addresses for Registration:** Students and teachers have pre-existing email addresses for registration; otherwise, they will encounter registration challenges without email addresses. Email addresses serve as essential identifiers and communication channels.
4. **Basic English Language Skills:** Students and teachers possess basic English language skills for platform interaction; otherwise, they will experience usability issues as non-English speakers.
5. **Human Interaction:** Students and teachers interacting with the platform are human individuals, not automated bots; otherwise, potential automated bot interference may affect platform security and user experience. Human interaction is crucial for the intended platform dynamics. Automated bot interference poses risks to security, integrity, and the user experience.
6. **Security Practices and Data Protection:** Students and teachers follow standard security practices, and the platform adheres to data protection standards; otherwise, data breaches, privacy concerns, loss of their trust, and compromised user data may occur in the case of security compliance failure.

**Dependencies:**

1. **Front-end Frameworks and Design Principles:** LearnLance relies on utilizing front-end frameworks and design principles that facilitate responsiveness, ensuring a consistent user experience across diverse platforms.
2. **Reliable and Scalable Database Server:** The successful implementation of LearnLance depends on the availability of a reliable and scalable database server, with compatibility for the chosen database management system.
3. **Integration of Third-party APIs:** LearnLance relies on integration and availability of third-party APIs for chat, call, live sessions, and payments.
4. **Adherence to Government Rules and Regulations:** LearnLance depends on adherence to government rules and regulations for tax implications during payment transactions.
5. **Collaboration and Coordination within the Development Team:** LearnLance relies on successful collaboration and coordination across all roles within the development team. Effective collaboration is essential for project success, influencing the quality, timeliness, and cohesiveness of development efforts.
6. **Changes in the Payroll System:** The operation of LearnLance depends on changes being made in the Payroll System to accept payment requests for courses and topic requests with LearnLance. Successful integration with the Payroll System is critical for processing payment requests seamlessly, ensuring a smooth financial interaction between users and the platform.

# **External Interface Requirements**

## **User Interfaces**

The user interfaces for LearnLance are designed to be intuitive, user-friendly, and consistent across the platform. The following outlines the logical characteristics of each interface between the software product and its users:

**User Profiles:**

* Components:

1. Profile customization section for students and teachers.
2. Input fields for personal information, educational achievements, interests, projects and bio information.
3. Options for uploading profile pictures.

* Layout Constraints:

1. Consistent layout for both student and teacher profiles.
2. User-friendly interface with clear navigation options.

* Standard Buttons and Functions:

1. "Save" and "Update" buttons for profile changes.
2. "Upload" button for profile pictures.

**Course Management:**

* Components:

1. Course posting and enrollment interfaces.
2. Course details including title, description, payment details, and enrollment options.

* Layout Constraints:

1. Clear distinction between course creation and course enrollment interfaces.
2. Consistent layout for displaying course details.

* Standard Buttons and Functions:

1. "Create Course" and "Enroll" buttons.
2. Filters and sorting options for course searches.

**Communication Tools (Chat and Call):**

* Components:

1. Text chat and video call interfaces.
2. User contact lists and chat history sections.

* Layout Constraints:

1. Intuitive design for seamless communication.
2. Easy access to chat history and call logs.

* Standard Buttons and Functions:

1. "Send Message," "Start Call," and "End Call" buttons.
2. Notification alerts for new messages and incoming calls.

**Bidding System:**

* Components:

1. Interface for students to post job requests.
2. Bid submission and selection interfaces for teachers and students.

* Layout Constraints:

1. Clear steps for posting job requests and submitting bids.
2. Transparent bidding process for both students and teachers.

* Standard Buttons and Functions:

1. "Post Job," "Submit Bid," and "Accept Bid" buttons.
2. Status indicators for active and closed bids.

**Payment Processing:**

* Components:

1. Payment interfaces for course enrollment and teacher payments.
2. Billing details and transaction history sections.

* Layout Constraints:

1. Secure and straightforward payment process.
2. Detailed transaction records for users.

* Standard Buttons and Functions:

1. "Pay Now" and "Confirm Payment" buttons.
2. Transaction history filters and search options.

**Virtual Classrooms:**

* Components:

1. Interface for scheduling and accessing live classes.
2. Interactive elements for real-time engagement.

* Layout Constraints:

1. Easy-to-follow schedule and class access design.
2. Engaging interface for students and teachers during live sessions.

* Standard Buttons and Functions:

1. "Schedule Class" and "Join Class" buttons.
2. Interactive tools for student-teacher interactions.

These interfaces collectively form the user experience within LearnLance, ensuring a cohesive and efficient platform for students, teachers, and administrators. Detailed design specifications for interface are documented below / appendix:

*Please refer to Appendix* ***7.3.12*** *that**presents the prototypes to depict the user interfaces.*

*Please refer to Appendix* ***7.3.22, 7.3.23*** *that presents the scenarios and storyboards.*

## **Hardware Interfaces**

**Supported Device Types:**

* LearnLance is compatible to run on common smartphones supporting Android platforms.
* LearnLance adapts to different screen sizes and resolutions to ensure a consistent user experience.
* Touchscreen compatibility for mobile devices.

**Data and Control Interactions:**

* Interacts with hardware components for uploading and downloading user profile images.
* Facilitates secure processing of payment transactions.
* Utilizes device-specific APIs for seamless data transfer.
* Implements encryption protocols for secure data interactions.

**Communication Protocols:**

* Employs WebSocket for live chat and virtual classrooms.
* Utilizes HTTPS for secure data transmission.
* Integrates with device-specific communication protocols.

## **Software Interfaces**

LearnLance interfaces with various software components to deliver its functionalities seamlessly. The connections between LearnLance and other software components, including databases, operating systems, tools, libraries, and integrated commercial components, are described below:

**Database:** MongoDB

* **Data Items In:**
  + User profiles (personal information, educational achievements, interests, projects, ratings, certifications, rewards, feedback).
  + Course details (title, description, enrollment information).
  + Chat and call logs.
  + Bidding system data (job requests from students, bids).
  + Transaction records (payment details).
  + Virtual classrooms for students.
  + Adding, deleting and removing students and teachers by admin.
  + Students' preferences for recommendation of courses.
  + Teachers’ preferences for recommendation of jobs and projects (topic requests).
* **Data Items Out:**
  + Recommended courses for students using recommendation algorithm.
  + Recommended projects (topic requests), job recommendations for teachers.
  + Updated user profiles.
  + Chat and call logs information ensuring the functionality of social hub.
  + Course enrollment information.
  + Bidding system status updates.
  + Payment processing confirmation.
  + Performed operation of adding, deleting or removing student or teacher by admin.

**Libraries**: Python Libraries

* **Data Items In:**
  + Student preferences, skills for recommending courses.
  + Teacher preferences for recommending projects and jobs.
* **Data Items Out:**
  + Personalized course recommendations for students.
  + Project recommendations for teachers.

**External APIs:** Payment Gateway API, Communication API (video SDK)

* **Data Items In:**
  + Payment transaction details.
  + Communication requests (text chat, video call).
* **Data Items Out:**
  + Confirmation of payment processing.
  + Real-time communication data (chat messages, call logs, video calls).

*Please refer to Appendix* ***7.3.8*** *that presents the DFD level 0 diagram.*

## **Communication Interfaces**

LearnLance requires robust communication interfaces to facilitate seamless interactions between users, the platform, and external systems. The communication requirements and associated functions are detailed below:

**Email Communications:**

1. **Message Formatting:**
   * System-generated emails for account verification, password reset, and important notifications.
   * Standardized email templates for consistency.

**Web Browser Communications:**

1. **Protocols:**
   * Utilizes HTTPS for secure data transmission over the web.

**Network Server Communications:**

1. **Protocols:**
   * Employs web Sockets for real-time interactions, such as live chat and virtual classrooms.
   * Utilizes HTTPS for secure communication with the server.

**Electronic Forms:**

1. **Requirements:**
   * User-friendly forms for profile customization, course creation, and job posting.
   * Validation checks to ensure accurate data submission.

**Communication Security and Encryption:**

1. **Security Measures:**
   * + Implements end-to-end encryption for sensitive data, including chat messages and payment transactions.
     + Adheres to industry-standard security practices to safeguard user information.

# **System Features**

## **Registration/Sign in**

Description and Priority:

Description: This feature enables user account creation and authentication within the system.

Priority: High.

Stimulus/Response Sequences:

Sequence 1 - User Registration:

Stimulus: Click "Register" button.

Response: Display registration form for user details.

Sequence 2 - User Sign-In:

Stimulus: Click the “Sign In" button.

Response: Display sign-in form for user credentials.

*Please refer to the Functional Requirements in Appendix* ***7.2.2*** *from [FR-85 to FR-90].*

## **Notifications**

Description and Priority:

Description: This feature provides notifications to users for connection requests, joint account requests, payment verifications, bid topic requests, classes, meetings, and about-to-start sessions.

Priority: Medium.

Stimulus/Response Sequences:

Sequence 1 - Receiving a Notification for Connection Requests:

Stimulus: User receives a connection request.

Response: The system sends a notification to the user.

Sequence 2 - Receiving a Notification for Joint Account Requests:

Stimulus: User receives a joint account request.

Response: The system sends a notification to the user.

Sequence 3 - Receiving a Notification for Payment Verifications:

Stimulus: User's payment requires verification.

Response: The system sends a notification to the user.

Sequence 4 - Receiving a Notification for Bid Topic Requests:

Stimulus: User receives a bid topic request.

Response: The system sends a notification to the user.

Sequence 5 - Receiving a Notification for Classes:

Stimulus: User is scheduled for a class.

Response: The system sends a notification to the user.

Sequence 6 - Receiving a Notification for Meetings:

Stimulus: User is scheduled for a meeting.

Response: The system sends a notification to the user.

Sequence 7 - Receiving a Notification for About-to-Start Sessions:

Stimulus: User has a session about to start.

Response: The system sends an alert notifying the user of the impending session.

*Please refer to the Functional Requirements in Appendix* ***7.2.2*** *from [FR-79 to FR-84].*

## **Schedule Meetings (Calendar)**

Description and Priority:

Description: This feature allows users to schedule meetings using a calendar.

Priority: Medium.

Stimulus/Response Sequences:

Sequence 1 - Scheduling a Meeting:

Stimulus: User wants to schedule a meeting.

Response: The system provides a scheduling interface.

Sequence 2 - Confirming Meeting Details:

Stimulus: User confirms meeting details.

Response: The system schedules meetings on a calendar for the user.

*Please refer to the Functional Requirements in Appendix* ***7.2.2*** *from [FR-41 to FR-44].*

## **Discussion/Community Forum**

Description and Priority:

Description: This feature provides a discussion and community forum for users to engage in discussions, share information, and interact with others.

Priority: High.

Stimulus/Response Sequences:

Sequence 1 - Accessing the Community:

Stimulus: User clicks on the "Community" option.

Response: The system displays the list of communities.

Sequence 2 - Creating a Community:

Stimulus: User initiates the creation of a new community.

Response: The system prompts the user to input details for the new community.

Sequence 3 - Sending Request to Join a Community:

Stimulus: User selects a community and clicks on the "Join" option.

Response: The system sends a request to join the selected community.

Sequence 4 - Creating Posts in a Community:

Stimulus: User navigates to a specific community and selects the "Create Post" option.

Response: The system provides a platform for the user to compose and submit a post in the selected community.

Sequence 5 - Commenting on Posts in a Community:

Stimulus: User views a post within a community and selects the "Comment" option.

Response: The system allows the user to input and submit a comment on the selected post, facilitating the sharing of ideas and discussions within the community.

*Please refer to the Functional Requirements in Appendix* ***7.2.2*** *from [FR-76 to FR-78].*

## **Chat/Group Chat**

Description and Priority:

Description: This feature enables users to engage in one-on-one and group chats within the platform.

Priority: High.

Stimulus/Response Sequences:

Sequence 1 - Accessing Chat:

Stimulus: User selects the "Chat" option.

Response: The system displays the chat interface, showing recent conversations.

Sequence 2 - Starting a New Chat:

Stimulus: User clicks on a chat.

Response: The system presents a list of recent messages within the chat.

Sequence 3 - Sending Messages:

Stimulus: User types and sends a message in a chat.

Response: The system displays the sent message in the chat conversation.

*Please refer to the Functional Requirements in Appendix* ***7.2.2*** *from [FR 59, FR-62 to FR-63, FR-66-67].*

## **Call/video Call**

Description and Priority:

Description: This feature allows users to make audio and video calls within the platform.

Priority: High.

Stimulus/Response Sequences:

Sequence 1 - Initiating a Call:

Stimulus: User selects a contact and clicks on "Call" or "Video Call."

Response: The system establishes a call connection with the selected contact.

Sequence 2 - Receiving a Call:

Stimulus: User receives an incoming call or video call.

Response: The system notifies the user and provides options to accept or decline the call.

Sequence 3 - During a Call:

Stimulus: User is in an active call.

Response: The system displays call controls for muting, ending, and switching (audio/video) a call.

Sequence 4 - Ending a Call:

Stimulus: User decides to conclude an ongoing call.

Response: The system displays an "End Call" option. Upon selecting "End Call," the system terminates the call connection, concluding the communication session.

*Please refer to the Functional Requirements in Appendix* ***7.2.2*** *from [FR-60 to FR-61, FR-64 to FR-65, FR-68 to FR-75].*

## **Create Profile**

Description and Priority

Description: This feature allows users to create a basic profile by providing essential information.

Priority: High.

Stimulus/Response Sequences:

Sequence 1 - Accessing Profile Creation:

Stimulus: User clicks on "Create Profile" or "Edit Profile."

Response: The system presents a profile creation/editing form.

Sequence 2 - Submitting Profile Information:

Stimulus: User fills in and submits the profile creation/editing form.

Response: The system processes the information and updates the user's profile.

*Please refer to the Functional Requirements in Appendix* ***7.2.2*** *from [FR-1, FR-14].*

## **Recommendations**

Description and Priority

Description: This feature provides personalized recommendations to users, including teacher bidding projects and course recommendations for students.

Priority: High.

Stimulus/Response Sequences:

Sequence 1 - Accessing Recommendations:

Stimulus: User navigates to the home page.

Response: The system displays personalized recommendations based on user preferences and history.

Sequence 2 - Browsing Teacher Bidding Projects:

Stimulus: Teacher swipes down to bidding posts on the home page.

Response: The system lists bidding projects that match the teacher's skills with the project’s required skills.

Sequence 3 - Exploring Course Recommendations for Students:

Stimulus: Students swipe down to recommended courses.

Response: The system suggests courses suitable for students based on their experience and interests.

*Please refer to the Functional Requirements in Appendix* ***7.2.2*** *from [FR-23, FR-34].*

## **Take Courses/Take Sessions**

Description and Priority:

Description: This feature allows students to browse and enroll in courses or sessions.

Priority: High.

Stimulus/Response Sequences:

Sequence 1 - Browsing Available Courses/Sessions:

Stimulus: Student navigates to the "Courses" or "Upcoming Sessions" section.

Response: The system displays a list of available courses or upcoming sessions.

Sequence 2 - Enrolling in a Course/Session:

Stimulus: Student selects a course or session and clicks on "Enroll" or "Interested."

Response: The system enrolls the student in the selected course or adds a reminder for a session.

Sequence 3 – Taking Live Session:

Stimulus: Student clicks on a live session.

Response: The system allows access to the user by entering him into the session.

*Please refer to the Functional Requirements in Appendix* ***7.2.2*** *from [FR-24 to FR-25, FR-45 to FR-47].*

## **Payment**

Description and Priority:

Description: This feature enables students to make payments for courses. Payments are initially deposited into the LearnLance account and are subsequently transferred to the teacher's account upon successful course completion.

Priority: Medium.

Stimulus/Response Sequences:

Sequence 1 - Initiating Payment:

Stimulus: Student selects a course to enroll in and clicks on the "Pay" button.

Response: The system prompts the student to proceed with payment.

Sequence 2 - Completing Payment:

Stimulus: Student confirms the payment and submits the transaction.

Response: The system processes the payment and provides payment confirmation.

Sequence 3 - Fund Transfer to Teacher:

Stimulus: Upon successful course completion, the system initiates a fund transfer to the respective teacher's account.

Response: The system updates the teacher's account balance and sends a notification to the teacher.

*Please refer to the Functional Requirements in Appendix* ***7.2.2*** *from [FR-91 to FR-92].*

## **Certificates/Badges**

Description and Priority:

Description: This feature allows students to earn certificates and badges as a recognition of their achievements on the platform.

Priority: Medium.

Stimulus/Response Sequences:

Sequence 1 - Earning Certificates and Badges:

Stimulus: Student completes a course or achieves a specific milestone.

Response: The system evaluates the student's progress and awards certificates or badges accordingly.

Sequence 2 - Viewing and Sharing Certificates/Badges:

Stimulus: Student accesses their profile or achievement section.

Response: The system displays earned certificates and badges, allowing students to view and share them.

*Please refer to the Functional Requirements in Appendix* ***7.2.2*** *from [FR-93 to FR-95].*

## **Post Achievements**

Description and Priority:

Description: This feature enables students to post and showcase their achievements, including badges, certifications, and completed courses, on their profiles or within the community.

Priority: High.

Stimulus/Response Sequences:

Sequence 1 - Posting an Achievement:

Stimulus: Student achieves a milestone, completes a course, or earns a badge.

Response: The student selects the option to post the achievement and enters any additional information or comments.

Sequence 2 - Viewing Achievements:

Stimulus: Other students access a student's profile or achievements section.

Response: The system displays the student's posted achievements, providing details and a visual representation.

*Please refer to the Functional Requirements in Appendix* ***7.2.2*** *from [FR-96 to FR-97].*

## **Student Connecting With Other Students and Following Teachers**

Description and Priority:

Description: This feature enables students to connect with other students and follow teachers on the platform, fostering a sense of community and facilitating interaction.

Priority: High.

Stimulus/Response Sequences:

Sequence 1 - Connecting with Students:

Stimulus: Student searches for other students or receives a connection request.

Response: The system allows students to send or accept connection requests, creating a network of connections.

Sequence 2 - Following Teachers:

Stimulus: Student explores teacher profiles or receives a recommendation to follow a teacher.

Response: The student can choose to follow a teacher, which adds the teacher's updates and courses to their feed.

*Please refer to the Functional Requirements in Appendix* ***7.2.2*** *from [FR-98 to FR-100].*

## **Bidding**

Description and Priority:

Description: This feature allows teachers to bid on topic requests posted by students, creating a competitive marketplace for educational services.

Priority: High.

Stimulus/Response Sequences:

Sequence 1 - Teacher Bidding:

Stimulus: A student posts a topic request with a description and quoted price.

Response: The system allows teachers to view the topic request details and write proposals with their quoted prices and submit bid topic requests.

Sequence 2 - Student Review:

Stimulus: The student reviews incoming bids from teachers.

Response: The system presents bid details, including teacher details, quoted price, and the bid proposal for the student to evaluate and then shortlist.

*Please refer to the Functional Requirements in Appendix* ***7.2.2*** *from [FR-29 to FR-33, FR-35 to FR-40].*

## **Add Courses and Schedule Live Sessions**

Description and Priority:

Description: This feature allows teachers to add new courses to the platform and separately schedule live sessions for different topics.

Priority: High.

Stimulus/Response Sequences:

Sequence 1 - Adding a New Course:

Stimulus: A teacher selects the option to add a new course.

Response: The system provides a course creation interface, allowing the teacher to input course details.

Sequence 2 - Scheduling Live Sessions:

Stimulus: A teacher chooses to schedule live sessions.

Response: The system presents a scheduling tool where the teacher can define session date, time, and other details.

*Please refer to the Functional Requirements in Appendix* ***7.2.2*** *from [FR-15 to FR-22].*

## **Joint Accounts**

Description and Priority:

Description: The "Joint Account" feature enables teachers to collaborate by scheduling live sessions together. Teachers can send joint session requests to each other, and upon acceptance, they can teach an existing course.

Priority: High.

Stimulus/Response Sequences:

Sequence 1 - Sending a Joint Account Request:

Stimulus: A teacher initiates a joint account request, specifying the intended collaborator.

Response: The system sends a request to the intended collaborator.

Sequence 2 - Accepting a Joint Account Request:

Stimulus: A teacher receives a joint session request.

Response: The teacher can accept or decline the request. Upon acceptance, the system confirms the collaboration.

Sequence 3 - Collaborating combinedly on the Course:

Stimulus: They start sharing, uploading, or moderating the content for the course.

Response: The system updates the Course content.

*Please refer to the Functional Requirements in Appendix* ***7.2.2*** *from [FR-101 to FR-102].*

## **Remove Students/Teachers**

Description and Priority:

Description: This feature allows administrators to remove users(student/teacher) or authorized users(students/teachers) to remove their accounts from the platform.

Priority: High.

Stimulus/Response Sequences:

Sequence 1 - Removing a Student:

Stimulus: An administrator selects a student's profile for removal.

Response: The system prompts for confirmation and, upon confirmation, removes the student from the platform.

Sequence 2 - Removing a Teacher:

Stimulus: An administrator selects a teacher's profile for removal.

Response: The system prompts for confirmation and, upon confirmation, removes the teacher from the platform.

Sequence 3 - Removing Oneself:

Stimulus: An authorized member selects to delete his/her profile.

Response: The system will verify the user again and will then remove the account from the system.

*Please refer to the Functional Requirements in Appendix* ***7.2.2*** *from [FR-103 to FR-105].*

## **Modify Students/Teachers**

Description and Priority:

Description: This feature allows authorized users(students/teachers) to modify their respective set of information within the platform.

Priority: High.

Stimulus/Response Sequences:

Sequence 1 - Modifying Student Information:

Stimulus: An authorized student selects his/her profile for modification.

Response: The system displays the student's information and allows the student to make necessary modifications.

Sequence 2 - Modifying Teacher Information:

Stimulus: An authorized teacher selects a his/her profile for modification.

Response: The system displays the teacher's information and allows the teacher to make necessary modifications.

*Please refer to the Functional Requirements in Appendix* ***7.2.2*** *from [FR-106 to FR-108].*

## **Add Students/Teachers**

Description and Priority:

Description: This feature allows administrators to add students or teachers to the platform.

Priority: High.

Stimulus/Response Sequences:

Sequence 1 – Adding a Student:

Stimulus: An administrator adds a new student's profile.

Response: The system prompts for confirmation and, upon confirmation, adds the student in the platform.

Sequence 2 - Adding a Teacher:

Stimulus: An administrator adds a new teacher's profile.

Response: The system prompts for confirmation and, upon confirmation, adds the teacher in the platform.

*Please refer to the Functional Requirements in Appendix* ***7.2.2*** *from [FR-109].*

## **View Revenue**

Description and Priority

Description: This feature allows admins of LearnLance to view revenue-related information, providing insights into earnings and financial performance.

Priority: High.

Stimulus/Response Sequences:

Sequence 1 - Accessing Revenue Information:

Stimulus: Admin selects the "View Revenue" option within their account.

Response: The system displays revenue-related data or reports relevant to the user's account.

*Please refer to the Functional Requirements in Appendix* ***7.2.2*** *from [FR-110].*

## **Create Posts**

Description and Priority:

Description: This feature allows admins of LearnLance to view revenue-related information, providing insights into earnings and financial performance.

Priority: High.

Stimulus/Response Sequences:

Sequence 1 - Accessing Revenue Information:

Stimulus: Admin selects the "View Revenue" option within their account.

Response: The system displays revenue-related data or reports relevant to the user's account.

*Please refer to the Functional Requirements in Appendix* ***7.2.2*** *from [FR-26 to FR-28].*

## **Create Comments/Like on Posts**

Description and Priority:

Description: This feature enables users to create comments and likes on posts. Commenting on or liking a community post can only be done by the respective joined community members, whereas anyone can comment and like on User Post.

Priority: Medium.

Stimulus/Response Sequences:

Sequence 1 – Create a Comment

Stimulus: User enters Comment description and create his/her comment on the post.

Response: The created comment will be visible on the post.

Sequence 2 – Create a Like:

Stimulus: User press like button and put his/her like on the post.

Response: The created like will be visible on the post.

*Please refer to the Functional Requirements in Appendix* ***7.2.2*** *from [FR-111 to FR-112].*

## **Virtual Classrooms**

Description and Priority:

Description: This feature enables students and teachers to redirect to a virtual classroom where they can have live classes on the topic that was requested by the student earlier, following successful payment confirmation by the admin.

Priority: High.

Stimulus/Response Sequences:

Sequence 1 – Conduct Live Classes

Stimulus: Student and Teacher will both decide on a mutual time and join the meeting on the desired date and time

Response: This is where Student and teacher will learn on the topics requested using live one-to-one video session between them.

*Please refer to the Functional Requirements in Appendix* ***7.2.2*** *from [FR-48 to FR-58].*

*Please refer to the Functional Requirements in section* ***1.4.2*** *that presents the user requirements corresponding these features.*

*Please refer to the Functional Requirements in Appendix* ***7.3.10, 7.3.14, 7.3.15, 7.3.16*** *that presents feature list, use case diagram and narratives, use case glossary, and user stories.*

# **Other Nonfunctional Requirements**

**EXTERNAL QUALITY REQUIREMENTS**

## **Performance Requirements**

PER-1: The average response time from backend and database will usually take 3-4 seconds.

PER-2: The average response time for a change done using frontend will be reflected spontaneously within a second because of state management.

PER-3: There will be no hard limit on number of data records to be inserted into the system.

PER-4: There are hard timing requirements for calling and messaging, as it is done in real time.

PER-5: Time delays in communication on video call or audio will range from 1-2 second to reach another end user.

PER-6: Mobile pages shall fully download in an average of 3-4 seconds or less over a 30 megabits/second Internet connection.

PER-7: At least 98 percent of the time, the messaging system shall update the message within 1-2 seconds or less after the initiation.

PER-8: The LearnLance shall display payment confirmation messages to students or receiving confirmation messages to teachers after admin approval, so it will take up to 1-2 days for admin approval.

PER-9: The payment transactions for courses requests shall be processed at an average rate of 1000 per second.

PER-10: The payment transactions for bid topic requests shall be processed at an average rate of 1000 per second.

PER-11: LearnLance can handle up to 5,000 concurrent users during non-peak hours.

PER-12: LearnLance can handle up to 10,000 concurrent users during peak hours.

PER-13: LearnLance can support a maximum of 100 participants in a call, including the host.

PER-14: Audio / Video calls can be conducted with a minimum latency as low as 10 hertz per second.

PER-15: Users can view course content videos in different encoded formats according to their current internet speed for excellent performance output.

## **Availability Requirements**

AVL-1: The system is required to maintain a minimum availability of 95% on weekdays between 8:00 am and midnight Eastern Time, excluding scheduled maintenance windows.

AVL-2: The system is required to maintain a minimum availability of 99% on weekends between 8:00 am and midnight Eastern Time, excluding scheduled maintenance windows.

AVL-3: Downtime, exclusive of scheduled maintenance, occurs during maintenance scheduled from 3:00 am to 6:00 am Eastern Time on Monday, or Thursday.

AVL-4: Scheduled maintenance windows will primarily occur on the last Monday of each month (Gregorian Calendar).

AVL-5: Peak system usage is anticipated to be on weekends, specifically from 6:00 pm to 10:00 pm Eastern Time.

AVL-6: The expected response time, during incidents/ outages/ software failure mechanisms, will be communicated to users through social media platforms or LearnLance notifications, providing details about the next availability time.

AVL-7: LearnLance will not be available when LearnLance goes on updating to the newer version to access additional features.

AVL-8: Users will receive two notifications prior to the maintenance.

## **Installability Requirements**

INS-1: An untrained user shall be able to successfully perform an installation/ bundling of the LearnLance in an average of 3 minutes.

INS-2: When installing/ bundling an upgraded version of LearnLance, all recommendations in the users’ profile will be retained or converted to the new version’s data format if needed.

INS-3: Following successful installation/ bundling, the installation program will automatically delete the temporary files, backup, obsolete and unneeded files associated with LearnLance.

INS-4: In case of incomplete, incorrect, or user-aborted installation/ bundling, LearnLance installation will resume from where it was aborted.

INS-5: In case of reinstallation LearnLance shall retain user-specific data and customizations, even in the reinstallation of the same version, without loss or corruption.

INS-6: Installation of a new version will have new features and these features will work using the same existing data of the user.

INS-7: Installation of older version will not utilize the newer information generated for the user on newer version, but intersecting features will work the same way on older versions of the LearnLance.

INS-8: Updates on the new version availability will be notified to the user also, and user can update LearnLance with just a click.

INS-9: Installing LearnLance will require admin privileges and at least two reviewers to review the production code before going live.

INS-10: the installation program shall verify the correctness of the download of LearnLance before the beginning of the installation process.

INS-11: A complete walkthrough tutorial or user manual will be given for each updated version, which will guide the user through each step of installation to ensure seamless experience.

INS-12: LearnLance installation will be built and made compact and efficient to minimize download time and storage space required for installation.

INS-13: LearnLance installation/bundling will also check the LearnLance compatibility with the android devices allowed.

INS-14: The installation process will automatically create an icon in all apps for convenient access to LearnLance.

INS-15: LearnLance on uninstallation from the phone’s settings will completely remove the application but will have record of all the user’s data.

INS-16: A progress bar or stats indicator will be displayed during the bundling/ uninstalling providing users with real-time feedback on the installation progress.

## **Integrity Requirements**

INT-1: LearnLance shall safeguard against unauthorized additions, deletions, or modifications of data.

INT-2: LearnLance shall validate the encrypted passwords encoded by third-party libraries to ensure security.

INT-3: LearnLance shall perform validation on all input fields on its frontend for smooth interaction between the frontend and backend.

INT-4: LearnLance shall protect against potential attackers from within the LearnLance system by utilizing the Express Validator library on the frontend, implementing necessary checks on the backend.

INT-5: LearnLance will ensure the schemas include all required constraints.

INT-6: In the event of LearnLance uninstallation, user data will be preserved.

INT-7: In the event of LearnLance reinstallation, the user will have access to their previous data.

INT-8: LearnLance's recommendations for courses will be retained until the student modifies their interests or changes their interaction with the system.

INT-9: LearnLance's recommendations for topic requests will persist until the teacher modifies the skills required for the specific topic request.

INT-10: After performing a LearnLance database backup, the system shall verify the backup copy against the original and report any discrepancies to the Admin.

INT-11: LearnLance should report any discrepancies in case of backup to the Admin.

INT-12: LearnLance shall confirm daily that the application executables have not been modified by unauthorized code.

INT-13: LearnLance shall regularly perform database backups on multiple servers.

INT-14: LearnLance should replicate video content encoding for courses to ensure a range of video resolutions from low to high.

## **Interoperability Requirements**

IOP-1: LearnLance shall easily communicate with the 3rd party API of call management systems.

IOP-3: LearnLance shall easily communicate with the 3rd party API of message management systems.

IOP-3: LearnLance shall be designed for easy access to the phone’s camera, allowing access with just a click.

IOP-4: LearnLance shall be designed for easy access to the phone’s microphone, allowing access with just a click.

IOP-5: LearnLance shall provide easy access to Google Collab for running Python scripts for the recommendation of Topic Requests.

IOP-6: LearnLance shall provide easy access to Google Collab for running Python scripts for the recommendation of Courses.

IOP-7: LearnLance shall be designed to easily access and fetch time from the system for user notifications.

IOP-8: LearnLance shall be designed to easily access and fetch time from the system for reminders.

IOP-9: LearnLance's backend and frontend should seamlessly communicate with each other through the HTTP/HTTPS protocol.

IOP-10: LearnLance shall easily communicate with the distributed backup servers, so that if one goes down, the load is shifted to another backup server.

IOP-11: LearnLance should have all the information about servers where the content distribution server has replicated, so that video packets can be obtained from those location servers.

IOP-12: Firewalls and network tools shall be able to easily locate the root server of LearnLance, ensuring easy access to the server.

IOP-13: Web Application Insight resources shall also be easily connected to LearnLance, providing real-time alerts to the administrator.

## **Robustness Requirements**

ROB-1: If the student doesn't fill out all the Topic Request details, it will be drafted.

ROB-2: The next time the user views the draft Topic Request, it will recover the contents previously written until published.

ROB-3: If the teacher doesn't fill out all the Course details, it will be drafted.

ROB-4: The next time the user views the drafted course, it will recover the contents previously written until published.

ROB-5: All date attributes must have a default value specified as the current date of the system's time zone if the user doesn't specify any date by default.

ROB-6: If LearnLance is performing a payment operation and a power/ internet cut is faced the system will roll back the transaction and resumes from the proper operation.

ROB-7: If LearnLance database server gets down, Load balancer will balance the traffic coming.

ROB-8: If LearnLance database server gets down, Load balancer will direct users to the active servers.

ROB-9: If an audio call or video call is interrupted by a network connection request then user will be connected directly to the call and continue.

## **Reliability Requirements**

REL-1: The meantime between failures of the call and message management system shall be at least 30 days.

REL-2: The meantime between failures of the recommendation system shall be at least 90 days.

REL-3: No more than 5 experimental runs out of 50 can fail because of software or hardware mismatches.

REL-4: LearnLance data will be end-to-end encrypted, so no data will be breached, even if data breaches exist.

## **Safety Requirements**

SAF-1: LearnLance should make all Privacy and Policy information visible to all users (Students/Teachers) to ensure compliance with LearnLance rules and regulations.

SAF-2: LearnLance should privilege the community creator to diligently monitor content uploaded on the community wall, ensuring the removal of any member/post/comment irrelevant or vulgar to the Community.

SAF-3: LearnLance will display guidelines for each user action so that users can strictly adhere to them.

SAF-4: LearnLance will terminate resources running within 1 second if measured alerts exceed 25% of the threshold for suspicious activity.

## **Security Requirements**

SEC-1: Only students with Course access privileges shall be able to view the course content.

SEC-2: Only users with Community access shall be able to view community posts.

SEC-3: LearnLance will not allow any unauthorized user to access its services.

SEC-4: Password encryption is implemented to prevent compromising user passwords in case of a data breach.

SEC-5: No LearnLance member can modify content without creation or admin rights.

SEC-6: LearnLance shall allow users to view only the course content they have purchased.

SEC-7: LearnLance shall permit only students, not teachers, to create Topic Requests.

SEC-8: LearnLance shall permit only teachers, not students, to create Courses.

SEC-9: LearnLance shall allow only students, not teachers, to enroll in Courses.

SEC-10: LearnLance shall permit Teachers to send Joint Account Requests only to connected teachers, not to students or unconnected teachers.

SEC-11: LearnLance shall only allow ringing/messaging of the intended user.

SEC-12: LearnLance shall only allow interested students to join the live session, with the teacher having control over the settings.

SEC-13: LearnLance will check on every backend request that user has a valid authentication token generated by JSON Web Token.

SEC-14: LearnLance's firewall will block suspicious IP addresses to prevent LearnLance from threats.

SEC-15: LearnLance shall lock a user's account for 1 hour after five consecutive unsuccessful login attempts within a period of 10 minutes.

SEC-16: The system shall log all attempts to access secure data by users with privilege levels.

SEC-17: The system shall prevent robot attempts to access by using reCAPTCHA during login.

SEC-18: All payment transactions shall be encrypted via the payment gateway.

## **Usability Requirements**

USE-1: LearnLance should display a drawer icon to provide easy access to all visible menu options for user convenience.

USE-2: LearnLance displays FAQs to assist users with any questions they may have.

USE-3: LearnLance will provide user documentation/playbook with a detailed guide on how to use each functionality.

USE-4: LearnLance will display the most useful widgets on the admin profile for ease of understanding LearnLance.

USE-5: Users can directly access a profile by clicking on its profile picture or searching for the profile, allowing for easy access.

USE-6: A trained user should be able to create user posts in an average of 2 minutes, and a maximum of 5 minutes, 95 percent of the time.

USE-7: Users can navigate back using the mobile phone's back option for ease of control, even outside the LearnLance platform.

USE-8: Users can use gestures for handy options like navigating to the home page directly or opening the drawer menu.

USE-9: Tooltips for every input field will be there with 3-4 examples.

USE-10: Ability to carry over a payment from using the previously inserted card information.

USE-11: Error messages coming to the user will be self-explanatory and meaningful, providing clear messages.

**INTERNAL QUALITY REQUIREMENTS**

## **Efficiency Requirements**

EFF-1: LearnLance will be built to be storage-efficient, utilizing less disk space and memory.

EFF-2: LearnLance shall be built to operate efficiently even on low communication bandwidth, up to 1mb/second.

EFF-3: If LearnLance utilizes an excessive number of available resources, users may experience degraded performance.

EFF-4: At least 30 percent of the processor capacity and memory available to LearnLance shall be unused at the planned peak load conditions.

EFF-5: LearnLance application insight resource should provide a warning message when the usage load exceeds 80 percent of the maximum planned capacity.

## **Modifiability Requirements**

MOD-1: Route calls should be nested, and for every other main requirement, a separate file should be created.

MOD-2: All error handling, including try-catch blocks, must also log error messages for easy debugging.

MOD-3: The average time required to add a capability or fix a problem will be determined by starting again from the software development life cycle (SDLC) waterfall cycle, and the average time it will take is 15 days.

MOD-4: The percentage of fixes that are implemented correctly should be above 80% for better sustainability of LearnLance.

MOD-5: All the connection strings should be in a config file which will be modifiable.

## **Supportability Requirements**

SUP-1: LearnLance shall display an error message if the microphone or camera is failing to work or is not found.

SUP-2: The LearnLance team will handle queries through email support and will strive to resolve the problems within an average of 2-3 working days.

## **Portability Requirements**

POR-1: LearnLance is compatible with all Android devices, including mobile phones and tablets, with an Android version greater than Android 8 Oreo.

POR-2: LearnLance may experience issues with auto-rotate features, and mobile pages should be viewed in portrait mode.

## **Reusability Requirements**

REU-1: All routes should be reusable at the code level.

REU-2: At least 30 percent of the application code must not be burdened with frequent insertions of new code.

## **Scalability Requirements**

SCA-1: LearnLance shall be designed to scale and accommodate additional functional requirements until they align with the business objectives.

SCA-2: LearnLance shall have the capability to increase the member limit for joining calls/video calls.

SCA-3: As LearnLance currently operates with the free tier of call and message management systems, it can be made scalable to include a paid tier plan with advanced features in calling and increased user limits.

SCA-4: The system should be made scalable so that future payment gateways can be integrated smoothly once approved by the admin.

## **Business Rules**

### **Facts:**

BRF-1: Each student is assigned a unique identifier.

BRF-2: Each teacher is assigned a unique identifier.

BRF-3: Each admin is assigned a unique identifier.

BRF-4: Every call is associated with a unique identifier.

BRF-5: Calls can be categorized as video sessions.

BRF-6: Calls can be categorized as audio sessions.

BRF-7: Calls can be individual calls.

BRF-8: Calls can be group calls.

BRF-9: Every chat has its own unique identifier.

BRF-10: Chats can be individual conversations.

BRF-11: Chats can be group conversations.

BRF-12: Every topic request is identified by a unique identifier.

BRF-13: Each topic request includes the rate per hour.

BRF-14: Each topic request specifies the estimated hours.

BRF-15: Every topic request tracks the number of bids received.

BRF-16: Each comment is marked by a unique identifier.

BRF-17: Comments can receive likes from users.

BRF-18: Comments can have replies from other users.

BRF-19: Every community is assigned a unique identifier.

BRF-20: Communities consist of members.

BRF-21: Every community is created by a designated community creator.

BRF-22: Each course has a unique identifier.

BRF-23: Each course is associated with specific course fees.

BRF-24: Each course specifies a primary language.

BRF-25: Each course offers the option of earning a certificate.

BRF-26: Every joint account is identified by a unique identifier.

BRF-27: Joint accounts include multiple teachers.

BRF-28: Each lesson item within a course has a defined duration.

BRF-29: Each lesson item in a course is assigned a specific item order.

BRF-30: Every meeting is marked by a unique identifier.

BRF-31: Each meeting designates a meeting organizer.

BRF-32: Meetings have associated titles.

BRF-33: Meetings are scheduled for specific dates.

BRF-34: Meetings are scheduled for specific times.

BRF-35: Every payment is linked to a unique identifier.

BRF-36: Payments correspond to specific amounts.

BRF-37: Each post is identified by a unique identifier.

BRF-38: Posts can receive likes from users.

BRF-39: Posts can have comments from users.

BRF-40: Every quiz is assigned a unique identifier.

BRF-41: Quizzes contain questions.

BRF-42: Each question specifies the number of marks.

BRF-43: Questions have corrected answers.

BRF-44: Questions may also include wrong answers.

### **Constraints:**

BRC-1: Every bid topic must specify the required skills for teachers to teach the topic.

BRC-2: The community creator has the exclusive authority to approve membership requests.

BRC-3: The community creator is solely authorized to remove members from the community.

BRC-4: Communities can be deleted exclusively by their creators.

BRC-5: Joint accounts can only be generated by teachers.

BRC-6: Joint accounts can only be created for courses.

BRC-7: Meetings can only be initiated by the designated meeting organizer.

BRC-8: Comments can be deleted by the post creator.

BRC-9: Comments can be deleted by the comment creator.

BRC-10: Posts can be removed by the post creator.

BRC-11: Community creators have the right to delete community posts.

BRC-12: Quizzes are only created by course creators.

BRC-13: Only admins can access revenue information.

BRC-14: Students are only permitted to connect with other students.

BRC-15: Students can follow teachers.

BRC-16: Teachers can only connect with other teachers.

*Please refer to Appendix* ***7.3.1*** *that presents the business rules constraints table.*

### **Action Enablers:**

BRA-1: If a bid topic is posted by a student, then the recommendation engine will start recommending topic requests to eligible teachers.

BRA-2: If a teacher sends a proposal for a topic request, then the student will be notified of the proposal.

BRA-3: If a student accepts the proposal of a teacher, then the teacher will be notified of their new job.

BRA-4: If a student accepts the proposal of a teacher, then the student will be prompted to make a payment.

BRA-5: If a student's payment for a topic request is verified, then they will be notified.

BRA-6: If both the student and teacher close the topic request, then the payment will be released to the teacher.

BRA-7: If a call is initiated, then the call recipient(s) will be notified.

BRA-8: If a call is disconnected, then the call record will be maintained in the call log.

BRA-9: If a call record is deleted, then it will be removed from the call log.

BRA-10: If a message is sent, then the receiver(s) will be notified.

BRA-11: If a message is sent, then the chat will be updated in the chat log.

BRA-12: If a message chat is deleted, then it will be removed from the chat log.

BRA-13: If a comment is made to a post, then the post creator will be notified.

BRA-14: If a comment is deleted for a post, then the post's comments will be modified.

BRA-15: If a reply for a comment is deleted, then the comment's replies will be modified.

BRA-16: If a community post is created, then community members will be notified.

BRA-17: If an individual post is created, then it will appear on the feed of the creator's connections.

BRA-18: If a post is liked, then the post creator will be notified.

BRA-19: If a post is unliked, then the total number of likes for that post will be updated.

BRA-20: If a post for a community is deleted, then community posts will be modified.

BRA-21: If a community is deleted, then all its posts will be deleted.

BRA-22: If a course is created by a teacher, then the recommendation engine will start recommending the course to students.

BRA-23: If a student enrolls in a course, then the course creator will be notified.

BRA-24: If a student enrolls in a course, then the student will be prompted to make a payment.

BRA-25: If a student's payment for a course is verified, then they will be notified.

BRA-26: If a student's payment for a course is verified, then they can access the course resources.

BRA-27: If a joint account request has been sent by a teacher, then the invited teacher will be notified.

BRA-28: If the invited teacher for a joint account accepts the request, then the request sender will be notified.

BRA-29: If a meeting is started, then all its attendees will be notified.

BRA-30: If a reminder is scheduled, then it will start appearing on the calendar.

BRA-31: If a payment is made by a student, then the admin will be notified.

BRA-32: If a quiz question is marked correctly, then the student's marks will be incremented.

BRA-33: If a quiz question is marked wrong, then the student's marks will be decreased.

BRA-34: If a connection request is sent, then the recipient will be notified.

BRA-35: If a connection request is accepted, then the request sender will be notified.

BRA-36: If a student follows a teacher, then the teacher will be notified.

### **Inferences:**

BRI-1: If a student doesn't pay for a topic request, then he/she is considered a 'no-show.'

BRI-2: If a post has many likes, then it is considered a valuable post.

BRI-3: If a post has fewer likes, then it is considered an invaluable post.

BRI-4: If a community has many members, then it is considered an active community.

BRI-5: If a community has fewer members, then it is considered an inactive community.

BRI-6: If a course has many members enrolled, then it is considered an informative course.

BRI-7: If a course has fewer members enrolled, then it is considered an uninformative course.

BRI-8: If a meeting has many attendees, then it is considered an informative meeting.

BRI-9: If a meeting has fewer attendees, then it is considered an uninformative meeting.

BRI-10: If a student scores well in a course, then he/she is considered intelligent.

BRI-11: If a student scores less in a course, then he/she is considered below average.

BRI-12: If a student has more connections, then he/she is considered socially active.

BRI-13: If a student has fewer connections, then he/she is considered socially inactive.

BRI-14: If a teacher has more connections, then he/she is considered socially active.

BRI-15: If a teacher has fewer connections, then he/she is considered socially inactive.

BRI-16: If a teacher has more followers, then he/she is considered well-known.

### **Computations:**

BRCo-1: The payment for a topic request will be calculated by multiplying the estimated hours and rate per hour requested by the teacher who is hired.

BRCo-2: The total marks of a quiz are calculated by multiplying the number of questions and points for each question.

BRCo-3: The marks obtained in a quiz are calculated by multiplying the number of correctly marked questions and points for each question.

BRCo-4: If the Number of Quizzes Passed is 0, then the Badge is None.

BRCo-5: If the Number of Quizzes Passed is 30 or more, the Badge is Bronze.

BRCo-6: If the Number of Quizzes Passed is 40 or more, the Badge is Silver.

BRCo-7: If the Number of Quizzes Passed is 50 or more, the Badge is Gold.

*Please refer to Appendix* ***7.3.2*** *that presents the business rules computations table.*

*Please refer to Appendix* ***7.3.3*** *that presents the business rules catalog.*

*Please refer to Appendix* ***7.3.4*** *for a decision tree outlining the process by which badges are assigned to students.*

# **Other Requirements**

## **Database Requirements**

DBR-1: The database should be non-relational.

DBR-2: The database should be hosted on a cloud service.

DBR-3: The database should be designed to accept media files.

DBR-4: The database should only be accessible from specific IP addresses assigned to administrators.

DBR-5: The database should be secured with two-factor authentication for access.

DBR-6: The database should be replicated on at least three servers.

DBR-7: The database should perform regular backups at intervals of at least one day.

## **Internalization Requirements**

INR-1: LearnLance should switch to different languages without compromising the user experience.

INR-2: LearnLance should switch to different time zones and adjust accordingly.

INR-3: LearnLance should be able to accept payments in various international currencies.

## **Legal Requirements**

LGR-1: Users must respect and adhere to the intellectual property rights owned by LearnLance.

LGR-2: Users must respect copyright infringement, or legal actions will be taken.

LGR-3: Users must not post any discriminatory content towards any culture or society.

## **Reuse Objectives**

REO-1: Backend APIs should be written using microservices for scalability and reusability.

REO-2: Services should be divided into middleware for reusing the same backend functions.

REO-3: Teachers can reuse the old format of a proposal to bid for a new topic request.

REO-4: Students can reuse the old format of a Topic Request to open a new topic request.

*Please refer to Appendix* ***7.3.11, 7.3.21*** *and* ***7.3.9*** *that presents the data dictionary, data structures, and ERD respectively.*

# **Appendices**

## **Appendix 1: Requirement Elicitation Technique**

As a Requirement Gathering Technique, we conducted a JRP (Joint Requirement Planning) Session, for developing an online learning and freelancing platform called LearnLance. The script includes roles like Session Leader, SME, Coder, Tester, Teacher, Scribe, and Sponsor, engaging in discussions on various aspects such as teacher profiles, UI design, and frontend requirements.

Additionally, the use of Questionnaires is introduced as a technique for gathering insights from LearnLance users. Below sections provides links to the questionnaire and response forms.

Lastly, the importance of Research is emphasized for LearnLance customers, aiming to benchmark against industry leaders and competitors to enhance competitiveness.

### **JRP Session:**

Please refer to this link to watch complete JRP video: *https://drive.google.com/file/d/1Gy1Y80WImlzDparig\_stsRoN4HRYhDbX/view?usp=drivesdk*

**Roles:**

* Session Leader (Lara): -> Yusra
* SME (Anna): -> Insha
* Coder (Moajam): -> Ismail
* Tester (Alia): -> Insha
* Teacher (Camilla): -> Yusra
* Scribe (Aaiz): -> Ismail
* Sponsor (UAC): -> Insha
* Analyst (Rafi Ujair): -> Ismail

**Script:**

**Session Leader:** Good morning, everyone! Welcome to our LearnLance JAD session. My name is Lara. Today, we have a critical task ahead—to create a revolutionary online learning and freelancing platform named LearnLance. Let's dive into our detailed discussion. I'll start by introducing myself as the session leader. Now, let's go around the room.

**Sponsor:** Good morning. My name is UAC. As the sponsor, my role involves aligning our features with business objectives.

**Analyst:** Hello, everyone. I'm Rafi Ujair, and I'll be taking on the role of the analyst. My responsibility is to gather and analyze the requirements. Let's get started.

**Scribe:** Good morning, everyone! My name is Aaiz. I'm the scribe writer, responsible for documenting our discussions. Let's ensure we capture the essence of what's being discussed throughout the JAD sessions.

**Subject Matter Expert (SME):** Morning, everyone. I'm Anna, and my role is to guide us in ensuring that the content and features align with the educational and employment needs and expertise of our target audience such as teachers and students.

**Teacher:** Hey everyone. My name is Camilla. I'm representing the user class of teacher for LearnLance.

**Coder:** Hello, team. My name is Moajam. As the Coder, I'll be diving into the technical aspects of development. Let's discuss how we can implement each feature seamlessly to create a robust and user-friendly LearnLance platform.

**Tester:** Hello, team. I am Alia. As the Tester, I'll be looking after the testing aspects of LearnLance.

**Session Leader:** Fantastic, everyone. Let's kick off with the teacher profile. Camilla, could you share your thoughts on what information and features you'd find valuable to include in your profile?

**Teacher:** Thank you. I would like to include personal information such as my name, profile picture, bio information, connections, followers’ number, rating, and location. Additionally, I'd like to highlight certifications, experiences and projects, educational background, feedback from peer teachers and students, languages spoken, as well as honors and awards. But I'm a bit confused about which fields to include in the experience section, especially considering its significance for the teacher profile.

**Session Leader:** Noted. Anna, could you please specify the specific details that should be included in the experience section?

**Subject Matter Expert (SME):** Certainly. When adding experience in the teacher profile, make sure to include the job title, company name, start date, end date, and location.

Analyst: Alright. UAC, what about mandatory and optional details?

**Sponsor:** Every piece of information is mandatory, except for feedback. Feedback will be included in a teacher's profile only when it is submitted by other teachers or students.

**Analyst:** Okay. So, are teachers also capable of editing, deleting, or adding this information?

**Teacher:** Yes, of course. I think it should be.

**Sponsor:** No Camilla, please note that feedback is read only to ensure that students and teachers have an accurate representation of reality.

**Analyst:** Alright, let's consider UAC's opinion, as he is more familiar with the business objectives of LearnLance. Now, Mr. UAC, could you please provide guidance on the extent to which you want the system to handle errors? I'm considering the system's robustness.

**Sponsor:** Well, I believe Alia would be better suited to address that.

**Tester:** Thank you. Input validation is crucial for every field in each section. I recommend Moajam incorporate try-catch blocks in coding to handle unexpected network, memory, and hardware errors. Additionally, errors should be displayed in red for better visibility.

**Session Leader:** Good input. Mr. UAC, could you please specify the information you would like to display prominently on the profile?

**Sponsor:** Certainly. The profile will feature three records for each section. However, the comprehensive list of records will be accessible by clicking the "View All" button.

**Analyst:** Alright.

**Session Leader:** Moajam, could you share your thoughts on whether the current UI requirements align with market trends?

**Coder:** In my opinion, displaying the three most recent records is a good idea, and it's also relatively easy to implement.

**Sponsor:** Yes, that sounds like a good approach.

**Teacher:** Great. Let's ensure the UI is user-friendly and the system is easy to use.

Scribe: Thank you, everyone. Kindly wait for two minutes so that I can document all the discussions.

**After 1-2 Minutes…**

**Session Leader:** Alright, Mr. UAC, what is your preference regarding frontend requirements? Would you prefer to review prototypes first, or do you plan to articulate your own set of requirements?

**Sponsor:** I would like to see the prototypes.

**Analyst:** Certainly, here they are. I prepared them in advance as instructed by Mr. Lara before the meeting.

**Sponsor:** Great. The design looks good, but I would like to have some changes incorporated into it.

**Analyst:** Of course, I'm open to making any necessary adjustments. Please let me know the specific changes you have in mind.

**Sponsor:** Firstly, I would like to have a moderate amount of content on the profile. For example, in the experiences section, I prefer having the title and company visible. When a user clicks on "View Details," the complete information for that record should be displayed.

**Session Leader:** Great input. Camilla, do you find it feasible to work with such a UI-based system?

**Teacher:** Yes, I agree. Having less content on the profile will likely engage viewers more, allowing them to focus on the details that specifically interest them.

Analyst: Alright, Mr. UAC, please proceed with any additional feedback or requirements you may have.

**Sponsor:** Secondly, I'd like all the buttons to be rounded, taking up 90% of the screen width, and centered.

**Analyst:** Certainly. Any more suggestions?

**Sponsor:** Additionally, I would prefer the font for the headings to be "Sohne" and for the descriptions to be "Calibri".

**Scribe:** Okay, I have compiled comprehensive notes of all our discussions. I would now like to request Mr. Lara to email them to all the session participants once the meeting concludes.

Session Leader: Does anyone have any concerns or questions regarding today's discussion?

**Teacher:** No, everything has been resolved.

**Sponsor:** Yes, we are all satisfied.

**Session Leader:** Great. To summarize today's session, we engaged in collaborative discussions on teacher profiles, UI design, and frontend requirements. The comprehensive notes will be shared with all participants for confirmation, ensuring alignment as we progress in building this innovative online learning and freelancing platform.

**Everyone says yes, alright.**

**Session Leader:** Let's take a lunch break. We will resume the student profile discussion shortly after lunch. Meanwhile, Ms. Kaleem Ujairni will be joining us as a student surrogate. Thank you.

### **Questionnaires:**

Why Questionnaires?

For users of LearnLance:

RR-1: Questionnaires efficiently gather insights from a diverse user base while being cost-effective.

RR-2: They are easily administered across geographical boundaries, facilitating feedback collection from various regions.

RR-3: Questionnaires provide structured, standardized data for systematic analysis, offering valuable insights for LearnLance's development.

RR-4: Standardized questions minimize interviewer bias, ensuring impartial feedback collection.

Questionnaires:

For Questionnaire: [*https://docs.google.com/forms/d/e/1FAIpQLSeJWGUGHT3FOMudetZf9ZuDbQ2pcbElKvfGXIJ0vVLJbqOTsg/viewform*](https://docs.google.com/forms/d/e/1FAIpQLSeJWGUGHT3FOMudetZf9ZuDbQ2pcbElKvfGXIJ0vVLJbqOTsg/viewform)

For Responses: [*https://docs.google.com/forms/d/1D2DhlWGSHxV6gvpyR7hJ2gACM4xYlusczdT7gZLaSL4/edit*](https://docs.google.com/forms/d/1D2DhlWGSHxV6gvpyR7hJ2gACM4xYlusczdT7gZLaSL4/edit)

### **Research:**

Why Research?

**For customers of LearnLance:**

RR-1: Researching traditional platforms enables LearnLance to benchmark itself against industry leaders and competitors, identifying features, functionalities, and user experiences for enhanced competitiveness.

RR-2: Examining these platforms provides insights into best practices in areas such as user interface design, engagement, and content delivery, facilitating improvements in user experience and platform performance.

**Description:**

In the realm of online education and freelancing, existing platforms like Coursera, Udemy, and edX have made significant strides in structured online education, offering a wide array of courses and certifications. However, a noticeable gap exists in the integration of freelancing and learning functionalities within these platforms. On the other hand, platforms like Upwork primarily cater to freelancers but lack a comprehensive educational component. LearnLance aims to bridge this gap by offering a unified platform that seamlessly combines learning and earning journeys. It not only provides a robust educational ecosystem with features like course recommendations, personalized learning, and assessments but also integrates freelancing and job bidding, addressing the limitations of the current disjointed systems.A black and white chart with black text

Description automatically generated

## **Appendix 2: Other Requirements**

### **Business Requirements:**

#### **Business Opportunities:**

BOO-1: Tapping into the need for a centralized and user-friendly profile management solution for students and teachers.

BOO-2: Addressing the demand for more efficient and accurate suggestions, potentially capturing a larger share of the market by offering superior recommendation experience.

BOO-3: Tapping the opportunity to ease and improve student-teacher interactions, positioning LearnLance as a user-centric solution for effective educational engagement.

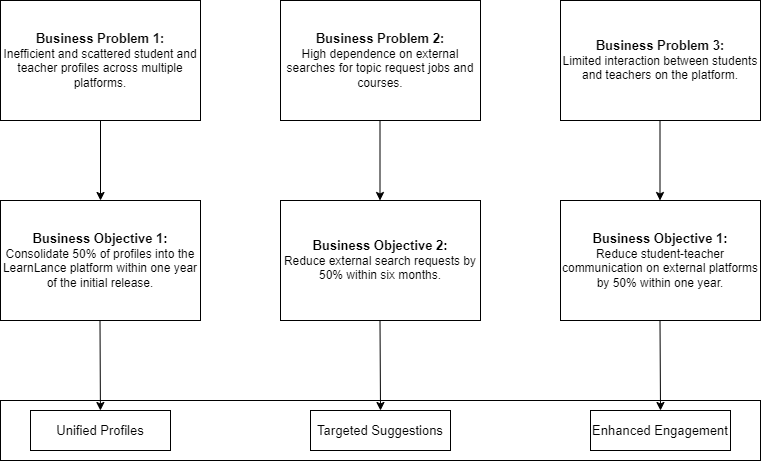
#### **Business Objectives:**

BO-1: Streamline students’ and teachers’ profiles by consolidating 50% of multiple profiles from various traditional platforms into profiles on single LearnLance platform within one year of initial release.

BO-2: Improve the personalized recommendation system for topic request jobs and courses to reduce external search requests by 50% within the six months of initial release.

BO-3: Enhance student-teacher interaction on the platform to reduce student-teacher communication on external platforms by 50% within the one year of initial release.

Following are the important business benefits that the product will provide in a quantitative and measurable way.



#### **Customer Needs:**

LearnLance aims to meet the critical needs of both students and teachers.

CN-1: Students require a seamless learning experience that includes webinars, topic-based learning, peer-to-peer collaboration, and community engagement.

CN-2: Teachers, on the other hand, need a platform to showcase their expertise, manage courses, conduct live sessions, and access a network of students seeking personalized learning.

CN-3: Furthermore, the current market necessitates a comprehensive and interconnected learning environment.

#### **Business Risks:**

BR-1: Potential failure to gain students’ and teachers’ adoption, reducing platform usage and limiting growth, impacting the platform's viability and success.

BR-2: Competitive market dynamics and the need to differentiate itself from other platforms, leading to difficulties in establishing a strong market presence and potentially reducing user acquisition.

BR-3: Reliance on third-party service providers for communication features, leading to potential disruptions in LearnLance communication system, students’ and teachers’ dissatisfaction, and hindered platform usage.

BR-4: Budget overruns and project delays impacting financial health, timeline, and the ability to meet students’, teachers’, and market demands when aligning with predefined milestones and allocated budget.

*Please refer to Appendix* ***7.3.5, 7.3.6*** *that presents the pareto bar chart and problem statement table, and Ishikawa diagram.*

### **Functional Requirements:**

*Please note that:*

* *The requirements were conveyed through various stakeholders, including student and teacher surrogate, the project sponsor, and analysts who presented prototypes. This communication process considered the competing systems also.*
* *The requirements, as communicated in the script attached in Appendix* ***7.1.1****, were managed by the leader. The developer then validated these requirements by listing the range of acceptable implementations, and the tester/QA ensured that the testing criteria for the requirements were delineated during the JRP session, as detailed in the script.*

FR-1: Students have the option to enrich their profiles by providing country.

FR-2: Students have the option to enrich their profiles by providing profile pictures.

FR-3: Students have the option to enrich their profiles by providing language.

FR-4: Students have the option to enrich their profiles by providing interests.

FR-5: Students have the option to enrich their profiles by providing education.

FR-6: Students have the option to enrich their profiles by providing bio information.

FR-7: Teachers have the option to enrich their profiles by providing profile pictures.

FR-8: Teachers have the option to enrich their profiles by providing language.

FR-9: Teachers have the option to enrich their profiles by providing education.

FR-10: Teachers have the option to enrich their profiles by providing experience.

FR-11: Teachers have the option to enrich their profiles by providing honors and awards.

FR-12: Teachers have the option to enrich their profiles by providing projects.

FR-13: Teachers have the option to enrich their profiles by providing certificates.

FR-14: Teachers have the option to enrich their profiles by providing skills.

FR-15: Teachers can include duration in their created courses.

FR-16: Teachers can include objectives in their created courses.

FR-17: Teachers can include descriptions in their created courses.

FR-18: Teachers can include videos in their created courses.

FR-19: Teachers can include quizzes in their created courses.

FR-20: Teachers can include sessions in their created courses.

FR-21: Teachers can include topics in their created courses.

FR-22: Teachers can include price in their created courses.

FR-23: Course recommendations are based on student learning history.

FR-24: Students can browse courses by topic.

FR-25: Students can browse courses by fees.

FR-26: Posts can include educational resources.

FR-27: Posts can include documents.

FR-28: Posts can include multimedia.

FR-29: Students can post topic requests by specifying the topic's title.

FR-30: Students can post topic requests by specifying the topic's description.

FR-31: Students can post topic requests by specifying the topic's estimated hours.

FR-32: Students can post topic requests by specifying the topic's rate per hour.

FR-33: Students can post topic requests by specifying the topic's required skills.

FR-34: Job recommendations are based on teacher profile.

FR-35: Teachers can browse topic requests based on skills.

FR-36: Teachers can browse topic requests based on price.

FR-37: Teachers can browse topic requests based on the number of proposals.

FR-38: Teachers can view active proposals they have initiated.

FR-39: Teachers can view requested proposals they have initiated.

FR-40: Teachers can view closed proposals they have initiated.

FR-41: Teachers can include the duration while scheduling live sessions.

FR-42: Teachers can include the topic while scheduling live sessions.

FR-43: Teachers can include the duration when scheduling webinars.

FR-44: Teachers can include the topic while scheduling webinars.

FR-45: Students can browse upcoming sessions by date.

FR-46: Students can browse upcoming sessions by time.

FR-47: Students can browse upcoming sessions by topic.

FR-48: Students can share their screens in virtual classrooms.

FR-49: Students can mute their microphones in virtual classrooms.

FR-50: Students can stop their video feeds in virtual classrooms.

FR-51: Students can raise their hands in virtual classrooms.

FR-52: Students can view all the participants in virtual classrooms.

FR-53: Teachers can share their screens in virtual classrooms.

FR-54: Teachers can mute their microphones in virtual classrooms.

FR-55: Teachers can stop their video feeds in virtual classrooms.

FR-56: Teachers can raise their hands in virtual classrooms.

FR-57: Teachers can view all the participants in virtual classrooms.

FR-58: Virtual classroom hosts can record the class for later review.

FR-59: Messages can include emojis.

FR-60: Calls can be audio calls.

FR-61: Calls can be video calls.

FR-62: Students can message each other in groups.

FR-63: Students can message each other individually.

FR-64: Students can call each other in groups.

FR-65: Students can call each other individually.

FR-66: Teachers can message each other in groups.

FR-67: Teachers can message each other individually.

FR-68: Teachers can call each other in groups.

FR-69: Teachers can call each other individually.

FR-70: Students can switch to audio/video calls.

FR-71: Students can mute their microphones.

FR-72: Students can stop their video feeds.

FR-73: Teachers can switch to audio/video calls.

FR-74: Teachers can mute their microphones.

FR-75: Teachers can stop their video feeds.

FR-76: Communities can include community names.

FR-77: Communities can include community descriptions.

FR-78: Communities can include community cover pictures.

FR-79: Notifications can be made pushed in the Notification window of Mobile screen.

FR-80: Notifications can be set to related to scheduled meetings.

FR-81: Notifications can be set for notifying user about scheduled maintenance window.

FR-82: Notifications sound can be made musical.

FR-83: Notifications sound can be made distinctive from systems sound.

FR-84: Notifications can be turned on from LearnLance system.

FR-85: Notifications can be turned off from LearnLance system.

FR-86: Notifications can also be viewed in the Drawer of LearnLance.

FR-87: Users can provide a valid email address.

FR-88: Users can provide a unique email address.

FR-89: Users can provide a strong password.

FR-90: Password can comprise of capital letters.

FR-91: Password can comprise of special characters.

FR-92: Password can comprise of number.

FR-93: Users can provide their First Name

FR-94: Users can provide their Last Name

FR-95: Users can provide their preference of using LearnLance.

FR-96: Payment can be encrypted end-to-end.

FR-97: Payment can be made via Account Number entered by Student.

FR-98: Certificate can be done with Course Title.

FR-99: Certificate can be made with Student Name on it.

FR-100: Certificate can include LearnLance logo for authentication.

FR-101: Badges can be made to have ranks such as Gold, Silver, and Bronze.

FR-102: Students can showcase their LearnLance associated certificates on their profiles.

FR-103: Students can showcase their LearnLance associated badges on their profiles.

FR-104: Student connecting other students can make posts visible on connected students’ feed.

FR-105: Student connecting other students can make posts visible on their profile.

FR-106: Students following other teachers can make posts visible on feed of the student feed.

FR-107: Teachers connecting other teachers can make posts visible on connected teachers’ feed.

FR-108: Teachers can write description to invite teacher for Joint Account.

FR-109: Teachers can Update the Course Content to the course he/she was invited.

FR-110: Admin can have a dashboard to remove Student from LearnLance.

FR-111: Admin can have a dashboard to remove Teacher from LearnLance.

FR-112: Student can have the option to delete his/her account from LearnLance drawer.

FR-113: Teacher can have the option to delete his/her account from LearnLance drawer.

FR-114: Student can update their interests by clicking on Pencil Icon on their own profile screen.

FR-115: Teacher can update their certificates by clicking on Pencil Icon on their own profile screen.

FR-116: Teacher can update their experience by clicking on Pencil Icon on their own profile screen.

FR-117: Admin can have a dashboard to add Student on LearnLance.

FR-118: Admin can have a dashboard to add Teacher on LearnLance.

FR-119: Admin can have dashboard to view Revenue-related data with specific time periods.

FR-120: User can do commenting in a hierarchical manner.

FR:121: User can like post by just a click.

FR:122: User can like comment by just a click.

*Please refer to Appendix* ***7.3.12*** *that presents the prototypes to model functional requirements.*

*Please refer to Appendix* ***7.3.22, 7.3.23*** *that presents the scenarios and story boards.*

*Please refer to Appendix* ***7.4.2.2*** *that presents the functional requirement prioritization.*

*Please refer to Appendix* ***7.3.20*** *that presents the crud matrix.*

### **System Requirements:**

SR-1: The system must operate 24/7 to ensure continuous user access.

SR-2: The system must minimize scheduled maintenance or downtime.

SR-3: The system must communicate any scheduled downtime in advance.

SR-4: The system's servers should efficiently handle peak usage times, including live webinars and virtual classrooms.

SR-5: The system shall grant administrators the authority to manage students within the LearnLance platform.

SR-6: The system shall grant administrators the authority to manage teachers within the LearnLance platform.

SR-7: The system shall grant teachers the authority to generate courses.

SR-8: The system shall grant teachers the authority to conduct live sessions.

SR-9: The system shall grant teachers the authority to bid on topic requests.

SR-10: The system shall grant teachers the authority to create joint accounts.

SR-11: The system shall grant students the authority to enroll in courses.

SR-12: The system shall grant students the authority to post topic requests.

SR-13: The system will automatically assign certificates and badges to students without user intervention.

SR-14: The system will undergo scheduled maintenance every night between 3 AM and 4 AM. During this period, server access will be temporarily closed to users to ensure system reliability and performance.

### **Operating Environment Requirements:**

OE-1: LearnLance shall operate on Apache HTTP server.

OE-2: LearnLance shall permit user access from the corporate intranet; from a VPN internet connection; and by Android 8 Oreo smartphones and tablets.

*Please refer to Section* ***5*** *for more information on NFR’s.*

### **Vision Statement:**

**For** students and teachers **who** are seeking a seamless, connected, and user-centric learning and teaching experience, **the** LearnLance **is** an internet-based and android mobile revolutionary application **that** caters for the diverse needs of students by providing webinars, topic-based learning, peer-to-peer collaboration, and community engagement and serves teachers by offering a platform to showcase their expertise, manage courses, conduct live sessions, and access a network of students seeking personalized learning. **Unlike** traditional learning and freelancing platforms, **students and teachers** who uses LearnLance no longer need to navigate multiple platforms and maintain separate profiles to access learning and teaching services; instead, it will offer a unified application that includes enhanced student-teacher interaction and improved personalized recommendations for topic request jobs and courses, revolutionizing the way students and teachers explore and engage with teaching and learning services, making the experience more efficient and user-centric.

*Please refer to Appendix* ***7.3.6*** *for the Ishikawa diagram, which provides a detailed analysis of the problem and the business opportunity that can be tapped.*

## **Appendix 3: Requirements Models**

### **Business Rules Constraints Table:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Roles and Matrix Representation | Comment Creator | Post Creator | Community Creator | Student | Teacher | Admin | System |
| Social Hub | | | | | | | |
| Connect and disconnect with teachers. |  |  |  |  |  |  |  |
| Follow and unfollow teachers. |  |  |  |  |  |  |  |
| Connect and disconnect with students. |  |  |  |  |  |  |  |
| Course Management | | | | | | | |
| Create course. |  |  |  |  |  |  |  |
| Enroll in course. |  |  |  |  |  |  |  |
| Pay for course. |  |  |  |  |  |  |  |
| Recommend course. |  |  |  |  |  |  |  |
| Community and Post Management | | | | | | | |
| Create community. |  |  |  |  |  |  |  |
| Join community. |  |  |  |  |  |  |  |
| Delete community. |  |  |  |  |  |  |  |
| Manage members in community. |  |  |  |  |  |  |  |
| Delete community posts. |  |  |  |  |  |  |  |
| Delete community post comments. |  |  |  |  |  |  |  |
| Delete posts. |  |  |  |  |  |  |  |
| Delete post comments. |  |  |  |  |  |  |  |
| Leave community. |  |  |  |  |  |  |  |
| Admin Operations | | | | | | | |
| Access revenue information. |  |  |  |  |  |  |  |
| Manage users in LearnLance. |  |  |  |  |  |  |  |
| Manage payments. |  |  |  |  |  |  |  |
| Live Sessions Management | | | | | | | |
| Initiate sessions. |  |  |  |  |  |  |  |
| Manage sessions. |  |  |  |  |  |  |  |
| End sessions. |  |  |  |  |  |  |  |
| Join Accounts Management | | | | | | | |
| Create and manage joint accounts. |  |  |  |  |  |  |  |
| Profile Management | | | | | | | |
| Manage profiles. |  |  |  |  |  |  |  |
| Communication Management | | | | | | | |
| Audio/Video call student/teacher. |  |  |  |  |  |  |  |
| Message student/teacher. |  |  |  |  |  |  |  |
| Topic Request Management | | | | | | | |
| Post and manage topic request. |  |  |  |  |  |  |  |
| Recommend topic requests. |  |  |  |  |  |  |  |
| Bid on topic request. |  |  |  |  |  |  |  |
| Select teacher for topic request. |  |  |  |  |  |  |  |
| Pay for topic request. |  |  |  |  |  |  |  |
| System Operations | | | | | | | |
| Issue certificate to students. |  |  |  |  |  |  |  |
| Issue badges to students. |  |  |  |  |  |  |  |
| Notify LearnLance users for different purposes. |  |  |  |  |  |  |  |
| Meeting Scheduling Management | | | | | | | |
| Schedule and unscheduled  meetings. |  |  |  |  |  |  |  |

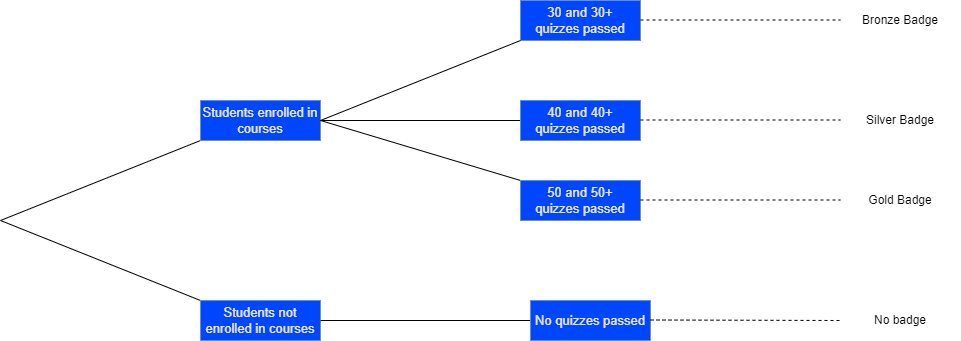
### **Business Rules Computations Table:**

|  |  |  |
| --- | --- | --- |
| ID | Number of Quizzes Passed | Badge |
| DISC-1 | 0 | None |
| DISC-2 | 30 and 30+ | Bronze |
| DISC-3 | 40 and 40+ | Silver |
| DISC-4 | 50 and 50+ | Gold |

### **Business Rules Catalog:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID | Rule Definition | Type of Rule | Static or Dynamic | Source |
| BRA-1 | If a bid topic is posted by a student, then the recommendation engine will start recommending topic requests to eligible teachers. | Action Enabler | Static | Sponsor |
| BRA-2 | If a teacher sends a proposal for a topic request, then the student will be notified of the proposal. | Action Enabler | Static | Student Surrogate |
| BRA-3 | If a student accepts the proposal of a teacher, then the teacher will be notified of their new job. | Action Enabler | Static | Teacher Surrogate |
| BRA-4 | If a student accepts the proposal of a teacher, then the student will be prompted to make a payment. | Action Enabler | Static | Sponsor |
| BRA-5 | If a student's payment for a topic request is verified, then they will be notified. | Action Enabler | Static | Student Surrogate |
| BRA-6 | If both the student and teacher close the topic request, then the payment will be released to the teacher. | Action Enabler | Dynamic | Sponsor |
| BRA-7 | If a call is initiated, then the call recipient(s) will be notified. | Action Enabler | Static | Sponsor |
| BRA-8 | If a call is disconnected, then the call record will be maintained in the call log. | Action Enabler | Static | Sponsor |
| BRA-9 | If a call record is deleted, then it will be removed from the call log. | Action Enabler | Static | Sponsor |
| BRA-10 | If a message is sent, then the receiver(s) will be notified. | Action Enabler | Static | Sponsor |
| BRA-11 | If a message is sent, then the chat will be updated in the chat log. | Action Enabler | Static | Sponsor |
| BRA-12 | If a message chat is deleted, then it will be removed from the chat log. | Action Enabler | Static | Sponsor |
| BRA-13 | If a comment is made to a post, then the post creator will be notified. | Action Enabler | Static | Sponsor |
| BRA-14 | If a comment is deleted for a post, then the post's comments will be modified. | Action Enabler | Static | Sponsor |
| BRA-15 | If a reply for a comment is deleted, then the comment's replies will be modified. | Action Enabler | Static | Sponsor |
| BRA-16 | If a community post is created, then community members will be notified. | Action Enabler | Static | Sponsor |
| BRA-17 | If an individual post is created, then it will appear on the feed of the creator's connections. | Action Enabler | Static | Sponsor |
| BRA-18 | If a post is liked, then the post creator will be notified. | Action Enabler | Static | Sponsor |
| BRA-19 | If a post is unliked, then the total number of likes for that post will be updated. | Action Enabler | Static | Sponsor |
| BRA-20 | If a post for a community is deleted, then community posts will be modified. | Action Enabler | Static | Sponsor |
| BRA-21 | If a community is deleted, then all its posts will be deleted. | Action Enabler | Dynamic | Sponsor |
| BRA-22 | If a course is created by a teacher, then the recommendation engine will start recommending the course to students. | Action Enabler | Static | Sponsor |
| BRA-23 | If a student enrolls in a course, then the course creator will be notified. | Action Enabler | Static | Teacher Surrogate |
| BRA-24 | If a student enrolls in a course, then the student will be prompted to make a payment. | Action Enabler | Static | Sponsor |
| BRA-25 | If a student's payment for a course is verified, then they will be notified. | Action Enabler | Static | Student Surrogate |
| BRA-26 | If a student's payment for a course is verified, then they can access the course resources. | Action Enabler | Static | Sponsor |
| BRA-27 | If a joint account request has been sent by a teacher, then the invited teacher will be notified. | Action Enabler | Static | Teacher Surrogate |
| BRA-28 | If the invited teacher for a joint account accepts the request, then the request sender will be notified. | Action Enabler | Static | Teacher Surrogate |
| BRA-29 | If a meeting is started, then all its attendees will be notified. | Action Enabler | Static | Sponsor |
| BRA-30 | If a reminder is scheduled, then it will start appearing on the calendar. | Action Enabler | Static | Sponsor |
| BRA-31 | If a payment is made by a student, then the admin will be notified. | Action Enabler | Static | Sponsor |
| BRA-32 | If a quiz question is marked correctly, then the student's marks will be incremented. | Action Enabler | Dynamic | Sponsor |
| BRA-33 | If a quiz question is marked wrong, then the student's marks will be decreased. | Action Enabler | Dynamic | Sponsor |
| BRA-34 | If a connection request is sent, then the recipient will be notified. | Action Enabler | Static | Sponsor |
| BRA-35 | If a connection request is accepted, then the request sender will be notified. | Action Enabler | Static | Sponsor |
| BRA-36 | If a student follows a teacher, then the teacher will be notified. | Action Enabler | Static | Teacher Surrogate |
| BRF-1 | Each student is assigned a unique identifier. | Fact | Static | Developer |
| BRF-2 | Each teacher is assigned a unique identifier. | Fact | Static | Developer |
| BRF-3 | Each admin is assigned a unique identifier. | Fact | Static | Developer |
| BRF-4 | Every call is associated with a unique identifier. | Fact | Static | Developer |
| BRF-5 | Calls can be categorized as video sessions. | Fact | Static | Sponsor |
| BRF-6 | Calls can be categorized as audio sessions. | Fact | Static | Sponsor |
| BRF-7 | Calls can be individual calls. | Fact | Static | Sponsor |
| BRF-8 | Calls can be group calls. | Fact | Dynamic | Sponsor |
| BRF-9 | Every chat has its own unique identifier. | Fact | Static | Developer |
| BRF-10 | Chats can be individual conversations. | Fact | Static | Sponsor |
| BRF-11 | Chats can be group conversations. | Fact | Dynamic | Sponsor |
| BRF-12 | Every topic request is identified by a unique identifier. | Fact | Static | Developer |
| BRF-13 | Each topic request includes the rate per hour. | Fact | Dynamic | Sponsor |
| BRF-14 | Each topic request specifies the estimated hours. | Fact | Dynamic | Sponsor |
| BRF-15 | Every topic request tracks the number of bids received. | Fact | Dynamic | Sponsor |
| BRF-16 | Each comment is marked by a unique identifier. | Fact | Static | Developer |
| BRF-17 | Comments can receive likes from users. | Fact | Dynamic | Sponsor |
| BRF-18 | Comments can have replies from other users. | Fact | Dynamic | Sponsor |
| BRF-19 | Every community is assigned a unique identifier. | Fact | Static | Developer |
| BRF-20 | Communities consist of members. | Fact | Static | Sponsor |
| BRF-21 | Every community is created by a designated community creator. | Fact | Static | Sponsor |
| BRF-22 | Each course has a unique identifier. | Fact | Static | Developer |
| BRF-23 | Each course is associated with specific course fees. | Fact | Static | Sponsor |
| BRF-24 | Each course specifies a primary language. | Fact | Dynamic | Sponsor |
| BRF-25 | Each course offers the option of earning a certificate. | Fact | Dynamic | Sponsor |
| BRF-26 | Every joint account is identified by a unique identifier. | Fact | Static | Developer |
| BRF-27 | Joint accounts include multiple teachers. | Fact | Dynamic | Teacher Surrogate |
| BRF-28 | Each lesson item within a course has a defined duration. | Fact | Dynamic | Sponsor |
| BRF-29 | Each lesson item in a course is assigned a specific item order. | Fact | Dynamic | Sponsor |
| BRF-30 | Every meeting is marked by a unique identifier. | Fact | Static | Developer |
| BRF-31 | Each meeting designates a meeting organizer. | Fact | Static | Sponsor |
| BRF-32 | Meetings have associated titles. | Fact | Dynamic | Sponsor |
| BRF-33 | Meetings are scheduled for specific dates. | Fact | Static | Sponsor |
| BRF-34 | Meetings are scheduled for specific times. | Fact | Static | Sponsor |
| BRF-35 | Every payment is linked to a unique identifier. | Fact | Static | Developer |
| BRF-36 | Payments correspond to specific amounts. | Fact | Static | Sponsor |
| BRF-37 | Each post is identified by a unique identifier. | Fact | Static | Developer |
| BRF-38 | Posts can receive likes from users. | Fact | Dynamic | Sponsor |
| BRF-39 | Posts can have comments from users. | Fact | Dynamic | Sponsor |
| BRF-40 | Every quiz is assigned a unique identifier. | Fact | Static | Developer |
| BRF-41 | Quizzes contain questions. | Fact | Static | Sponsor |
| BRF-42 | Each question specifies the number of marks. | Fact | Dynamic | Sponsor |
| BRF-43 | Questions have corrected answers. | Fact | Static | Sponsor |
| BRF-44 | Questions may also include wrong answers. | Fact | Dynamic | Sponsor |
| BRC-1 | Every bid topic must specify the required skills for teachers to teach the topic. | Constraint | Static | Sponsor |
| BRC-2 | The community creator has the exclusive authority to approve membership requests. | Constraint | Static | Sponsor |
| BRC-3 | The community creator is solely authorized to remove members from the community. | Constraint | Dynamic | Sponsor |
| BRC-4 | Communities can be deleted exclusively by their creators. | Constraint | Dynamic | Sponsor |
| BRC-5 | Joint accounts can only be generated by teachers. | Constraint | Dynamic | Sponsor |
| BRC-6 | Joint accounts can only be created for courses. | Constraint | Dynamic | Sponsor |
| BRC-7 | Meetings can only be initiated by the designated meeting organizer. | Constraint | Static | Sponsor |
| BRC-8 | Comments can be deleted by the post creator. | Constraint | Dynamic | Sponsor |
| BRC-9 | Comments can be deleted by the comment creator. | Constraint | Static | Sponsor |
| BRC-10 | Posts can be removed by the post creator. | Constraint | Dynamic | Sponsor |
| BRC-11 | Community creators have the right to delete community posts. | Constraint | Dynamic | Sponsor |
| BRC-12 | Quizzes are only created by course creators. | Constraint | Static | Sponsor |
| BRC-13 | Only admins can access revenue information. | Constraint | Static | Sponsor |
| BRC-14 | Students are only permitted to connect with other students. | Constraint | Dynamic | Sponsor |
| BRC-15 | Students can follow teachers. | Constraint | Static | Sponsor |
| BRC-16 | Teachers can only connect with other teachers. | Constraint | Dynamic | Sponsor |
| BRCo-1 | The payment for a topic request will be calculated by multiplying the estimated hours and rate per hour requested by the teacher who is hired. | Computation | Dynamic | Sponsor |
| BRCo-2 | The total marks of a quiz are calculated by multiplying the number of questions and points for each question. | Computation | Dynamic | Sponsor |
| BRCo-3 | The marks obtained in a quiz are calculated by multiplying the number of correctly marked questions and points for each question. | Computation | Dynamic | Sponsor |
| BRI-1 | If a student doesn't pay for a topic request, then he/she is considered a 'no-show.' | Inference | Dynamic | Sponsor |
| BRI-2 | If a post has many likes, then it is considered a valuable post. | Inference | Dynamic | Sponsor |
| BRI-3 | If a post has fewer likes, then it is considered an invaluable post. | Inference | Dynamic | Sponsor |
| BRI-4 | If a community has many members, then it is considered an active community. | Inference | Dynamic | Sponsor |
| BRI-5 | If a community has fewer members, then it is considered an inactive community. | Inference | Dynamic | Sponsor |
| BRI-6 | If a course has many members enrolled, then it is considered an informative course. | Inference | Dynamic | Sponsor |
| BRI-7 | If a course has fewer members enrolled, then it is considered an uninformative course. | Inference | Dynamic | Sponsor |
| BRI-8 | If a meeting has many attendees, then it is considered an informative meeting. | Inference | Dynamic | Sponsor |
| BRI-9 | If a meeting has fewer attendees, then it is considered an uninformative meeting. | Inference | Dynamic | Sponsor |
| BRI-10 | If a student scores well in a course, then he/she is considered intelligent. | Inference | Dynamic | Sponsor |
| BRI-11 | If a student scores less in a course, then he/she is considered below average. | Inference | Dynamic | Sponsor |
| BRI-12 | If a student has more connections, then he/she is considered socially active. | Inference | Dynamic | Sponsor |
| BRI-13 | If a student has fewer connections, then he/she is considered socially inactive. | Inference | Dynamic | Sponsor |
| BRI-14 | If a teacher has more connections, then he/she is considered socially active. | Inference | Dynamic | Sponsor |
| BRI-15 | If a teacher has fewer connections, then he/she is considered socially inactive. | Inference | Dynamic | Sponsor |
| BRI-16 | If a teacher has more followers, then he/she is considered well-known. | Inference | Dynamic | Sponsor |

### **Decision Tree for Badges Assignment to Students:**



### **Pareto Bar Chart and Problem Statement Table:**

The Pareto bar chart illustrates the percentage of the problems:

A graph of problems and problems

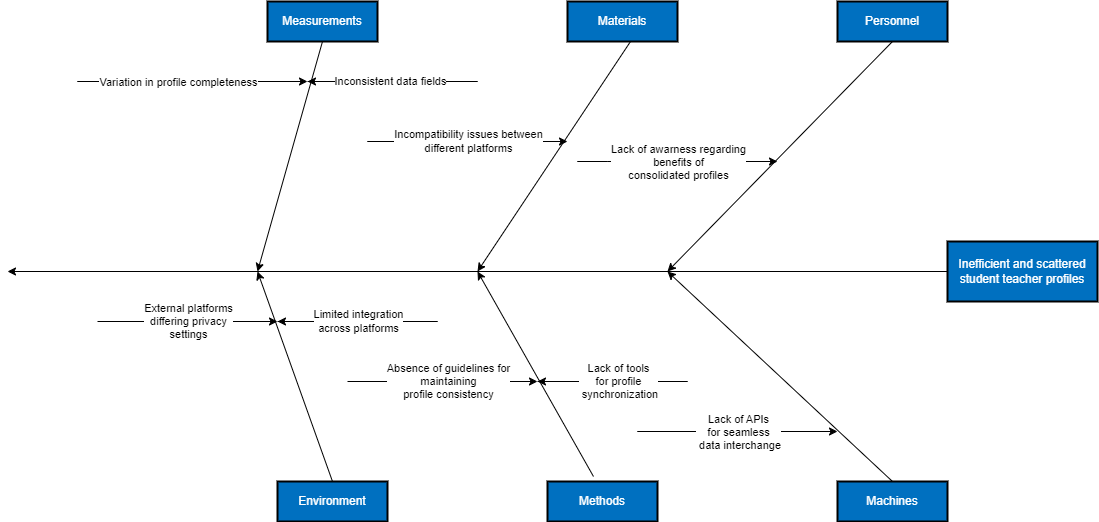
Description automatically generated

Based on the highest percentage among the problems, the below problem statement table describes the problem of “**Inefficient and scattered student teacher profile across platforms**”.

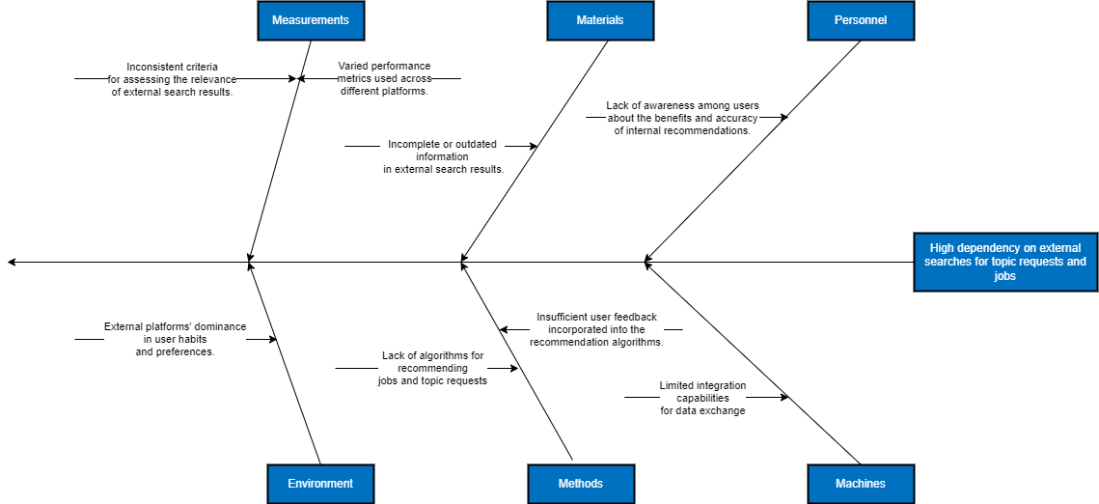
|  |  |
| --- | --- |
| Element | Description |
| What the problem is? | The problem at hand involves the inefficiency and dispersal of student and teacher profiles across multiple online platforms. Users currently face the challenge of maintaining and updating separate profiles on various educational and professional networks, leading to a fragmented online presence. |
| Who is affected by this? | This issue impacts both students and teachers participating in online education and professional networking. Students struggle with managing multiple profiles across diverse platforms, hindering their ability to showcase their educational achievements and interests effectively. Similarly, teachers face challenges in establishing a cohesive professional identity across various platforms, affecting their ability to attract students and job opportunities. |
| What is the impact? | The impact of this problem is multifold. Users experience a loss of time and effort in duplicating profile information across platforms. Additionally, the scattered nature of profiles diminishes the effectiveness of educational and professional networking, limiting the opportunities available to both students and teachers. The lack of a centralized and streamlined profile management system hampers the overall user experience. |
| Is there a proposed solution? | Yes, the proposed solution is the implementation of LearnLance, a unified platform that consolidates student and teacher profiles. LearnLance aims to streamline the profile management process by providing a single platform for users to showcase their educational achievements, interests, and professional endeavors. |
| What are the key benefits? | The key benefits of addressing this problem include:   * Efficiency: Users can manage a single profile, eliminating the need for redundant data entry and updates across multiple platforms. * Visibility: Consolidated profiles enhance the visibility of students and teachers, increasing their chances of connecting with relevant opportunities. * User Experience: LearnLance provides a seamless and user-friendly environment, improving the overall experience of both students and teachers in online education and professional networking space. * Networking Opportunities: A unified platform fosters better connections between students and teachers, creating a more conducive environment for learning and collaboration. |

### **Ishikawa Diagram:**

**Inefficient and scattered student teacher profiles**



**High dependency on external searches for topic requests and jobs**

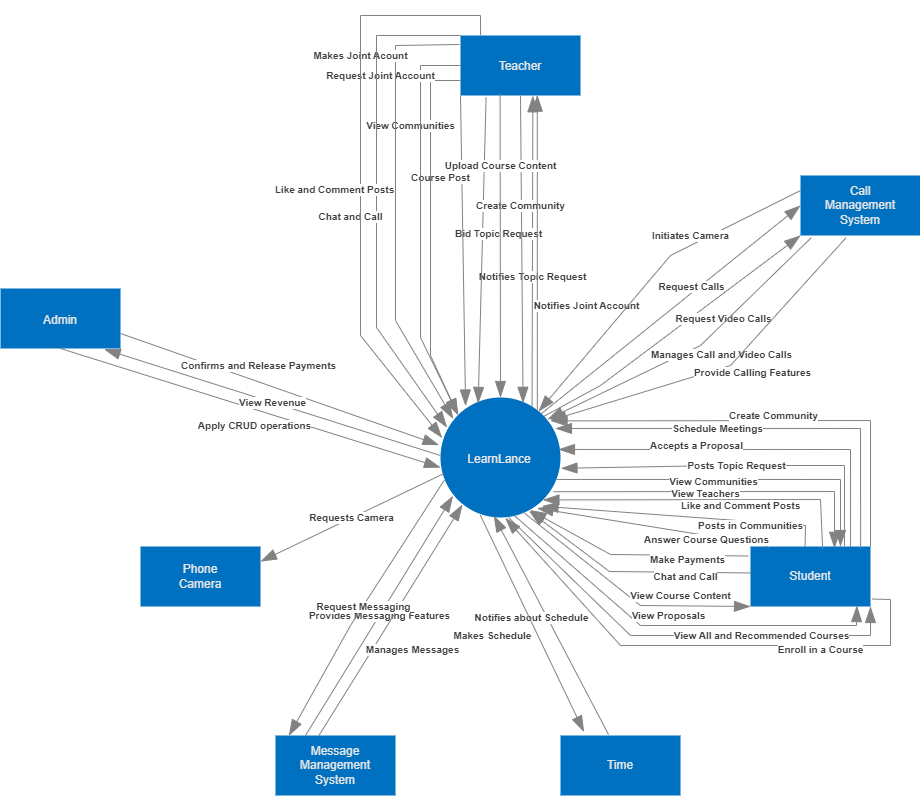


**Limited interaction between students and teachers**

A diagram of a diagram

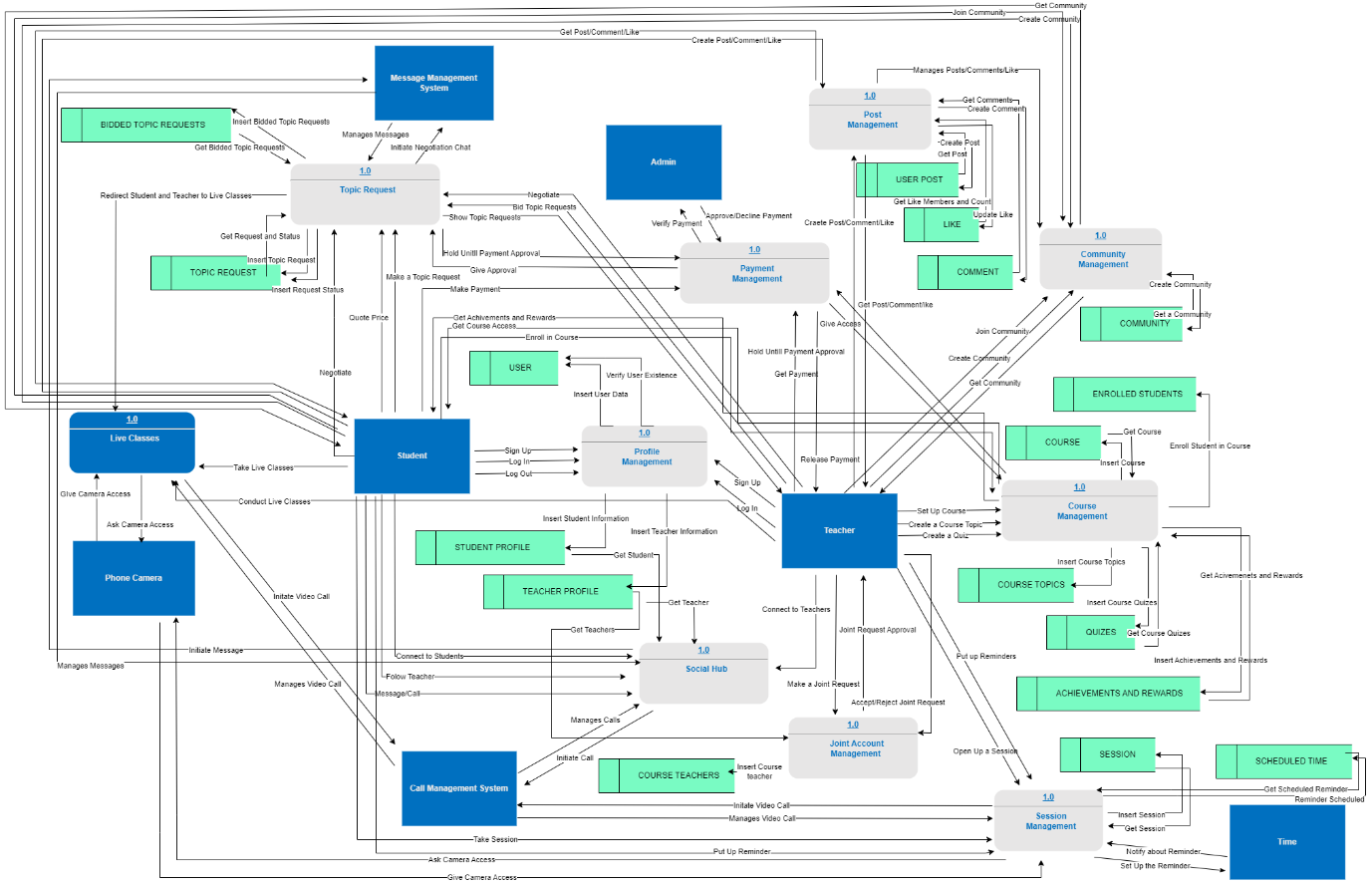
Description automatically generated

### **Context Diagram:**



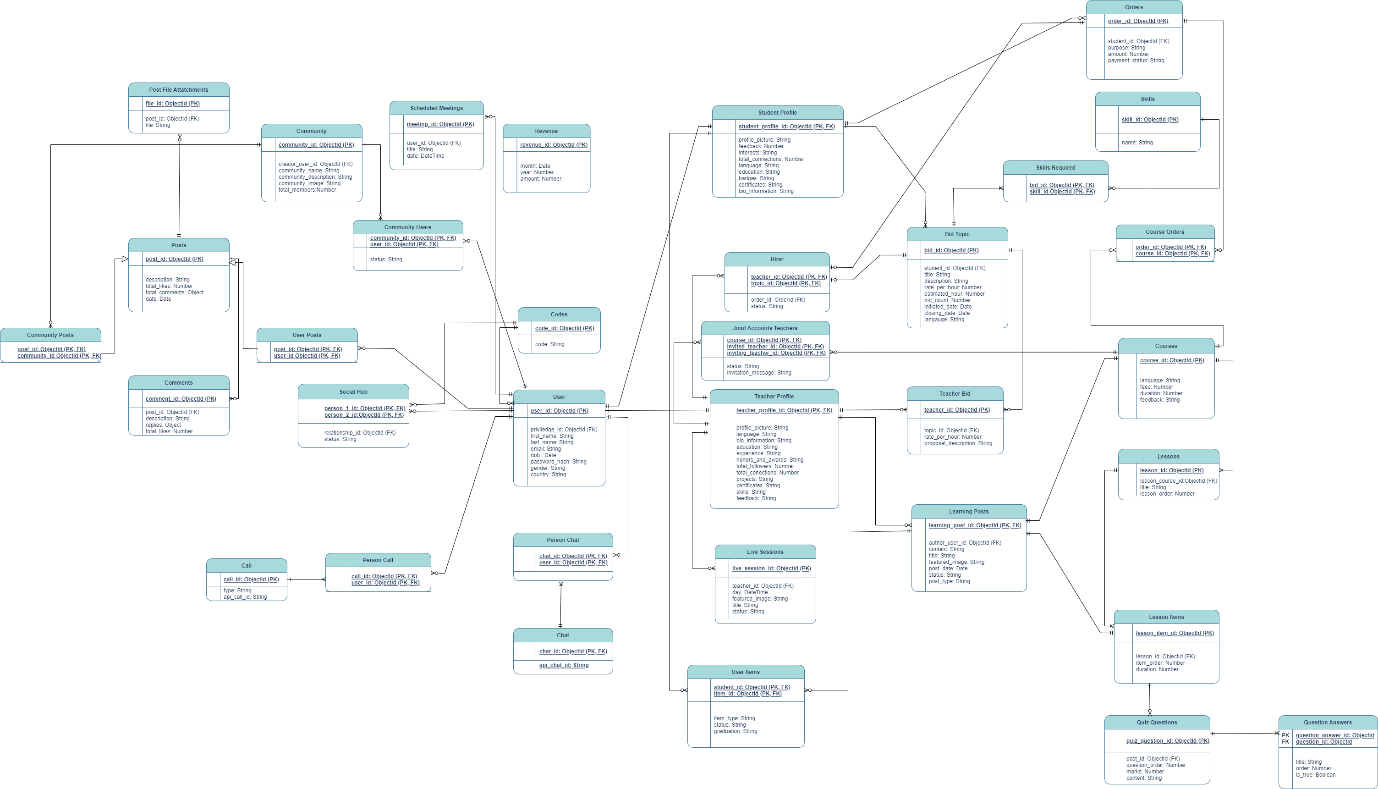
Please refer to this link if the image is not clear: *https://drive.google.com/file/d/1V5Rv3AySh-ObEQ8TEAUQOCeuo4g\_4qNN/view?usp=sharing*

### **DFD Level 0 Diagram:**



Please refer to this link if the image is not clear: *https://drive.google.com/file/d/1bZ4vV\_\_ojtI0JVEtVLEXo15LsSp-qJY3/view?usp=sharing*

### **ERD Diagram:**



Please refer to this link if the image is not clear: [*https://drive.google.com/file/d/1wWy9QD0ftcC32xqEArjzdWXxdNyFugjp/view?usp=sharing*](https://drive.google.com/file/d/1wWy9QD0ftcC32xqEArjzdWXxdNyFugjp/view?usp=sharing)

### **Feature List:**

A screenshot of a computer

Description automatically generated

Please refer to this link if the image is not clear: [*https://drive.google.com/file/d/1\_NdDeKWt-4ZTLmu0U7OOJOKdLtppj7jN/view?usp=sharing*](https://drive.google.com/file/d/1_NdDeKWt-4ZTLmu0U7OOJOKdLtppj7jN/view?usp=sharing)

### **Data Dictionary:**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Orders | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| order\_id | ObjectID | 24 | 12 | Primary Key, Not Null | - | Order Id | 5f84841f4d4a07e12345a678 |
| student\_id | ObjectID | 24 | 12 | Foreign Key (References Student\_Profile), Not Null | - | Student Id | 5f84841f4d4a07e12345a679 |
| purpose | String | Varies | Varies | Not Null | - | - | bidding |
| amount | Number | Varies | 8 | Not Null | - | - | 150 |
| Payment\_status | String | Varies | - | Not Null, default: “Pending” | - | Payment Status | paid |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Course Orders | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| order\_id | ObjectID | 24 | 12 | Foreign Key (References Orders), Not Null | - | Order Id | 5f84841f4d4a07e12345a678 |
| course\_id | ObjectID | 24 | 12 | Foreign Key (References Courses), Not Null | - | Course Id | 5f84841f4d4a07e12345a679 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Courses | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| course\_id | ObjectID | 24 | 12 | Primary Key, Not Null | - | Course Id | 5f84841f4d4a07e12345a680 |
| language | String | Varies | Varies | Not Null | - | - | English |
| fees | Number | Varies | 8 | Not Null | - | - | 100 |
| duration | Number | Varies | 8 | Not Null | - | - | 8 |
| feedback | String | Varies | Varies | - | - | - | Excellent course, highly recommended |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Lessons | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| lesson\_id | ObjectID | 24 | 12 | Primary Key, Not Null | - | Lesson Id | 5f1a8b9d2c4c5129a34e5911 |
| lesson\_course\_id | ObjectID | 24 | 12 | Foreign Key (References Courses), Not Null | - | Lesson Course Id | 5f1a8c1d2c4c5129a34e5912 |
| title | String | Varies | Varies | Not Null | - | - | Introduction to Programming |
| lesson\_order | Number | Varies | 8 | Not Null | - | Lesson Order | 1 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Lesson Items | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| lesson\_item\_id | ObjectID | 24 | 12 | Primary Key, Not Null | - | Lesson Item Id | 5f1a8b9d2c4c5129a34e5911 |
| lesson\_id | ObjectID | 24 | 12 | Foreign Key (References Lessons), Not Null | - | Lesson Id | 5f1a8c1d2c4c5129a34e5912 |
| item\_order | Number | Varies | 8 | Not Null | - | Item Order | 1 |
| duration | Number | Varies | 8 | Not Null | - | - | 30 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Quiz Questions | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| quiz\_question\_id | ObjectID | 24 | 12 | Primary Key, Not Null | - | Quiz Question Id | 5f1a8b9d2c4c5129a34e5911 |
| post\_id | ObjectID | 24 | 12 | Foreign Key (References Lesson Items), Not Null | - | Post Id | 5f1a8c1d2c4c5129a34e5912 |
| question\_order | Number | Varies | 8 | Not Null | - | Question Order | 1 |
| marks | Number | Varies | 8 | Not Null | - | - | 10 |
| content | String | Varies | Varies | Not Null | - | - | What is the capital of France? |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Question Answers | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| question\_answer\_id | ObjectID | 24 | 12 | Primary Key, Not Null | - | Question Answer Id | 5f1a8b9d2c4c5129a34e5911 |
| question\_id | ObjectID | 24 | 12 | Foreign Key (References Quiz Questions), Not Null | - | Question Id | 5f1a8c1d2c4c5129a34e5912 |
| title | String | Varies | Varies | Not Null | - | - | Paris |
| order | Number | Varies | 8 | Not Null | - | - | 1 |
| is\_true | Boolean | Varies | Varies | Not Null | true/false | Is True | true |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Bid Topic | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| bid\_id | ObjectID | 24 | 12 | Primary Key, Not Null | - | Bid Id | 5f1a8b9d2c4c5129a34e5911 |
| student\_id | ObjectID | 24 | 12 | Foreign Key (References Student\_Profile), Not Null | - | Student Id | 5f1a8c1d2c4c5129a34e5912 |
| title | String | Varies | Varies | Not Null | - | - | Programming Project |
| description | String | Varies | Varies | Not Null | - | - | Develop a web application |
| rate\_per\_hour | Number | Varies | 8 | Not Null | - | Rate per Hour | 25 |
| estimated\_hour | Number | Varies | 8 | Not Null | - | Estimated Hours | 20 |
| bid\_count | Number | Varies | 8 | Not Null | - | Bid Count | 5 |
| initiated\_date | Date | 10 | 8 |  | >= Current Date | Initiated Date | 2023-04-01 |
| closing\_date | Date | 10 | 8 |  | >= Current Date | Closing Date | 2023-05-01 |
| language | String | Varies | Varies | Not Null | - | - | English |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Teacher Bid | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| teacher\_id | ObjectID | 24 | 12 | Primary Key, Not Null | - | Teacher Id | 5f1a8b9d2c4c5129a34e5911 |
| topic\_id | ObjectID | 24 | 12 | Foreign Key (References Bid Topic), Not Null | - | Topic Id | 5f1a8c1d2c4c5129a34e5912 |
| rate\_per\_hour | Number | Varies | 8 | Not Null | - | Rate per Hour | 30 |
| proposal\_description | String | Varies | Varies | Not Null | - | Proposal Description | Experienced teacher in Math |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Learning Posts | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| learning\_post\_id | ObjectID | 24 | 12 | Primary Key, Foreign Key (References Lesson Items or Courses), Not Null | - | Learning Post Id | 5f1a8b9d2c4c5129a34e5911 |
| author\_user\_id | ObjectID | 24 | 12 | Foreign Key (References Teacher\_Profile), Not Null | - | Author User Id | 5f1a8c1d2c4c5129a34e5912 |
| content | String | Varies | Varies | Not Null | - | - | Introduction to Programming |
| title | String | Varies | Varies | Not Null | - | - | Learn Python Basics |
| featured\_image | String | Varies | Varies | Not Null | - | Featured Image | /images/python\_logo.png |
| post\_date | Date | 10 | 8 | Not Null, Default: CURRENT\_DATE | >= Current Date | Post Date | 2023-12-11 |
| status | String | Varies | Varies | Not Null | - | - | Draft |
| post\_type | String | Varies | Varies | Not Null | - | Post Type | Course |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Hirer | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| teacher\_id | ObjectID | 24 | 12 | Primary Key, Foreign Key (References Teacher\_Profile), Not Null | - | Teacher Id | 5f1a8b9d2c4c5129a34e5911 |
| topic\_id | ObjectID | 24 | 12 | Primary Key, Foreign Key (References Bid Topic), Not Null | - | Topic Id | 5f1a8c1d2c4c5129a34e5912 |
| order\_id | ObjectID | 24 | 12 | Foreign Key (References Order), Not Null | - | Order Id | 5f1a8d5e2c4c5129a34e5913 |
| status | String | Varies | Varies | Not Null,  Default: In Progress | - | - | Closed |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Student Profile | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| student\_profile\_id | ObjectID | 24 | 12 | Primary Key, Foreign Key (References Users), Not Null | - | Student Profile Id | 5f1a8b9d2c4c5129a34e5911 |
| profile\_picture | String | Varies | Varies | Not Null | - | Profile Picture | /images/profiles/john\_doe.jpg |
| feedback | Number | Varies | 8 | Not Null,  Default: 5 | - | - | 5 |
| interests | String | Varies | Varies | Not Null | - | - | Passionate web app developer skilled in creating robust applications. Expertise in the end-to-end development process, delivering user-centric solutions with a keen eye for innovation |
| total\_connections | Number | Varies | 8 | Not Null, Default: 0 | - | Total Connections | 50 |
| language | String | Varies | Varies | Not Null | - | - | Arabic |
| education | String | Varies | Varies | Not Null | - | - | Student at FAST-NUCES |
| badges | String | Varies | Varies | - | - | - | Bronze |
| certificates | String | Varies | Varies | - | - | - | Cloud Foundations |
| bio\_information | String | Varies | Varies | Not Null | - | Bio Information | Passionate about technology |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Joint Accounts | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| course\_id | ObjectID | 24 | 12 | Primary Key, Foreign Key (References Courses), Not Null | - | Course Id | 5f1a8b9d2c4c5129a34e5911 |
| invited\_teacher\_id | ObjectID | 24 | 12 | Primary Key, Foreign Key (References Teacher\_Profile), Not Null | - | Invited Teacher Id | 5f1a8c1d2c4c5129a34e5912 |
| inviting\_teacher\_id | ObjectID | 24 | 12 | Primary Key, Foreign Key (References Teacher\_Profile), Not Null | - | Inviting Teacher Id | 5f1a8d5e2c4c5129a34e5913 |
| status | String | Varies | Varies | Not Null, default: “Pending” | - | - | Pending |
| invitation\_message | String | Varies | Varies | Not Null | - | - | Join my course! |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Teacher Profile | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| teacher\_profile\_id | ObjectID | 24 | 12 | Primary Key, Foreign Key (References Users), Not Null | - | Teacher Profile Id | 5f1a8b9d2c4c5129a34e5911 |
| profile\_picture | String | Varies | Varies | Not Null | - | Profile Picture | /images/profiles/jane\_doe.jpg |
| feedback | String | Varies | Varies | - | - | - | I had the pleasure to work with Muhammad for the IEEEXtreme Programming contest 7.0, where Muhammad participated as a Judge and provided us with appreciable insights |
| total\_followers | Number | Varies | 8 | Not Null, Default: 0 | - | Total Followers | 100 |
| total\_connections | Number | Varies | 8 | Not Null, Default: 0 | - | Total Connections | 50 |
| language | String | Varies | Varies | Not Null | - | - | Arabic |
| education | String | Varies | Varies | Not Null | - | - | Student at FAST-NUCES |
| experience | String | Varies | Varies | Not Null | - | - | Teacher at FAST-NUCES |
| certificates | String | Varies | Varies | Not Null | - | - | Microsoft Certified Trainer - MySQL |
| bio\_information | String | Varies | Varies | Not Null | - | Bio Information | Passionate about teaching |
| honors\_and\_awards | String | Varies | Varies | Not Null | - | Honors and Awards | 3rd Position in Science and Commerce |
| Projects | String | Varies | Varies | Not Null | - | - | A search space based GUI test case generation and testing by using Tabu Search |
| Skills | String | Varies | Varies | Not Null | - | - | Problem Solver |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Live Sessions | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| live\_session\_id | ObjectID | 24 | 12 | Primary Key, Not Null | - | Live Session Id | 5f1a8b9d2c4c5129a34e5911 |
| teacher\_id | ObjectID | 24 | 12 | Foreign Key (References Teacher\_Profile), Not Null | - | Teacher Id | 5f1a8c1d2c4c5129a34e5912 |
| day | DateTime | 10 | 8 | Not Null | >= Current Date and Time | - | 2023-12-15T14:30:00 |
| featured\_image | String | Varies | Varies | Not Null | - | Featured Image | /images/live\_sessions/session1.jpg |
| title | String | Varies | Varies | Not Null | - | - | Introduction to Python |
| status | String | Varies | Varies | Not Null, default: “To Start” | - | - | Started |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| User Items | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| student\_id | ObjectID | 24 | 12 | Primary Key, Foreign Key (References Students), Not Null | - | Student Id | 5f1a8b9d2c4c5129a34e5911 |
| item\_id | ObjectID | 24 | 12 | Primary Key, Foreign Key (References Items), Not Null | - | Item Id | 5f1a8c1d2c4c5129a34e5912 |
| item\_type | String | Varies | Varies | Not Null | - | Item Type | Exam |
| status | String | Varies | Varies | Not Null,  Default: In Progress | - | - | In Progress |
| graduation | String | Varies | Varies |  | - | - | Pass |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Revenue | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| revenue\_id | ObjectID | 24 | 12 | Primary Key, Not Null | - | Revenue Id | 5f1a8b9d2c4c5129a34e5911 |
| month | Number | Varies | 8 | Not Null | >= 1 and <=12 | - | 12 |
| year | Number | Varies | 8 | Not Null | Current Year or Later Years | - | 2023 |
| amount | Number | Varies | 8 | Not Null | - | - | 5000 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Scheduled Meetings | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| meeting\_id | ObjectID | 24 | 12 | Primary Key, Not Null | - | Meeting Id | 5f1a8b9d2c4c5129a34e5911 |
| user\_id | ObjectID | 24 | 12 | Foreign Key (References Users), Not Null | - | User Id | 5f1a8c1d2c4c5129a34e5912 |
| title | String | Varies | Varies | Not Null | - | - | Team Briefing |
| date | DateTime | 10 | 8 | Not Null | >= Current Date and Time | - | 2023-12-15T14:30:00 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Codes | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| code\_id | ObjectID | 24 | 12 | Primary Key, Not Null | - | Code Id | 5f1a8b9d2c4c5129a34e5911 |
| code | String | Varies | Varies | Not Null | - | - | ABC123 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| User | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| user\_id | ObjectID | 24 | 12 | Primary Key, Not Null | - | User Id | 5f1a8b9d2c4c5129a34e5911 |
| privilege\_id | ObjectID | 24 | 12 | Foreign Key (References Codes), Not Null | - | Privilege Id | 5f1a8c1d2c4c5129a34e5912 |
| first\_name | String | Varies | Varies | Not Null | - | First Name | John |
| last\_name | String | Varies | Varies | Not Null | - | Last Name | Doe |
| email | String | Varies | Varies | Not Null | - | - | john.doe@email.com |
| dob | Date | 10 | 8 | Not Null | >= Current Date | - | 1990-05-15 |
| password\_hash | String | Varies | Varies | Not Null | - | Hashed Password | tbhifulieudie |
| gender | String | Varies | Varies | Not Null | - | - | Male |
| country | String | Varies | Varies | Not Null | - | - | United States |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Post File Attachments | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| file\_id | ObjectID | 24 | 12 | Primary Key, Not Null | - | File Id | 5f1a8b9d2c4c5129a34e5911 |
| post\_id | ObjectID | 24 | 12 | Foreign Key (References Posts), Not Null | - | Post Id | 5f1a8c1d2c4c5129a34e5912 |
| file | String | Varies | Varies | Not Null | - | - | /uploads/post\_attachments/file1.pdf |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Chat | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| chat\_id | ObjectID | 24 | 12 | Primary Key, Not Null | - | Chat Id | 5f1a8b9d2c4c5129a34e5911 |
| api\_chat\_id | String | Varies | Varies | Not Null | - | Api Chat Id | fewjjfhjurgfjirgf |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Person Chat | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| chat\_id | ObjectID | 24 | 12 | Primary Key, Foreign Key (References Chat), Not Null | - | Chat Id | 5f1a8b9d2c4c5129a34e5911 |
| user\_id | ObjectID | 24 | 12 | Primary Key, Foreign Key (References Users), Not Null | - | User Id | 5f1a8c1d2c4c5129a34e5912 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| User Posts | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| post\_id | ObjectID | 24 | 12 | Primary Key, Foreign Key (References Posts), Not Null | - | Post Id | 5f1a8b9d2c4c5129a34e5911 |
| user\_id | ObjectID | 24 | 12 | Primary Key, Foreign Key (References Users), Not Null | - | User Id | 5f1a8c1d2c4c5129a34e5912 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Person Call | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| call\_id | ObjectID | 24 | 12 | Primary Key, Foreign Key (References Calls), Not Null | - | Call Id | 5f1a8b9d2c4c5129a34e5911 |
| user\_id | ObjectID | 24 | 12 | Primary Key, Foreign Key (References Users), Not Null | - | User Id | 5f1a8c1d2c4c5129a34e5912 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Calls | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| call\_id | ObjectID | 24 | 12 | Primary Key, Not Null | - | Call Id | 5f1a8b9d2c4c5129a34e5911 |
| type | String | Varies | Varies | Not Null | - | - | Voice |
| Api\_call\_id | String | Varies | Varies | Not Null | - | Api Call Id | dfhfjfvfvvffvfvnjfv |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Community Posts | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| community\_id | ObjectID | 24 | 12 | Primary Key, Foreign Key (References Communities), Not Null | - | Community Id | 5f1a8b9d2c4c5129a34e5911 |
| post\_id | ObjectID | 24 | 12 | Primary Key, Foreign Key (References Posts), Not Null | - | Post id | 5f1a8c1d2c4c5129a34e5912 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Posts | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| post\_id | ObjectID | 24 | 12 | Primary Key, Not Null | - | Post Id | 5f1a8b9d2c4c5129a34e5911 |
| description | String | Varies | Varies | Not Null | - | - | This is a sample post |
| total\_likes | Number | Varies | 8 | Not Null, Default: 0 | - | Total Likes | 50 |
| total\_comments | Object | Varies | Varies | - | - | Total Comments | {"comment1": "Great post!", "comment2": "Nice content!"} |
| date | Date | 10 | 8 | Not Null,  Default: CURRENT\_DATE | >= Current Date | - | 2023-12-15 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Comments | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| comment\_id | ObjectID | 24 | 12 | Primary Key, Not Null | - | Comme d Id | 5f1a8b9d2c4c5129a34e5911 |
| post\_id | ObjectID | 24 | 12 | Foreign Key (References Posts), Not Null | - | Post Id | 5f1a8c1d2c4c5129a34e5912 |
| description | String | Varies | Varies | Not Null | - | - | This is a sample comment |
| replies | Object | Varies | Varies | - | - | - | {"comment1": "Great post!", "comment2": "Nice content!"} |
| total\_likes | Number | Varies | 8 | Not Null,  Default: 0 | - | Total Likes | 20 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Social Hub | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| person\_1\_id | ObjectID | 24 | 12 | Primary Key, Foreign Key (References Users), Not Null | - | Person 1 Id | 5f1a8b9d2c4c5129a34e5911 |
| person\_2\_id | ObjectID | 24 | 12 | Primary Key, Foreign Key (References Users), Not Null | - | Person 2 Id | 5f1a8c1d2c4c5129a34e5912 |
| relationship\_id | ObjectID | 24 | 12 | Foreign Key (References Codes), Not Null | - | Relationship Id | 5f1a8d5e2c4c5129a34e5913 |
| status | String | Varies | Varies | Not Null,  Default: “Pending” | - | - | Connected |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Community | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| community\_id | ObjectID | 24 | 12 | Primary Key, Not Null | - | Community Id | 5f1a8b9d2c4c5129a34e5911 |
| creator\_user\_id | Object | Varies | Varies | Foreign Key (References Users), Not Null | - | Creator Id | 5f1a8b9d2c4c5129a34e5911 |
| community\_name | String | Varies | Varies | Not Null | - | Community Name | Tech Enthusiasts |
| community\_description | String | Varies | Varies | Not Null | - | Community Description | A community for tech discussions |
| community\_image | String | Varies | Varies | Not Null | - | Community Image | /images/tech\_community.jpg |
| total\_members | Number | Varies | 8 | Not Null,  Default: 0 | - | Total Members | 500 |

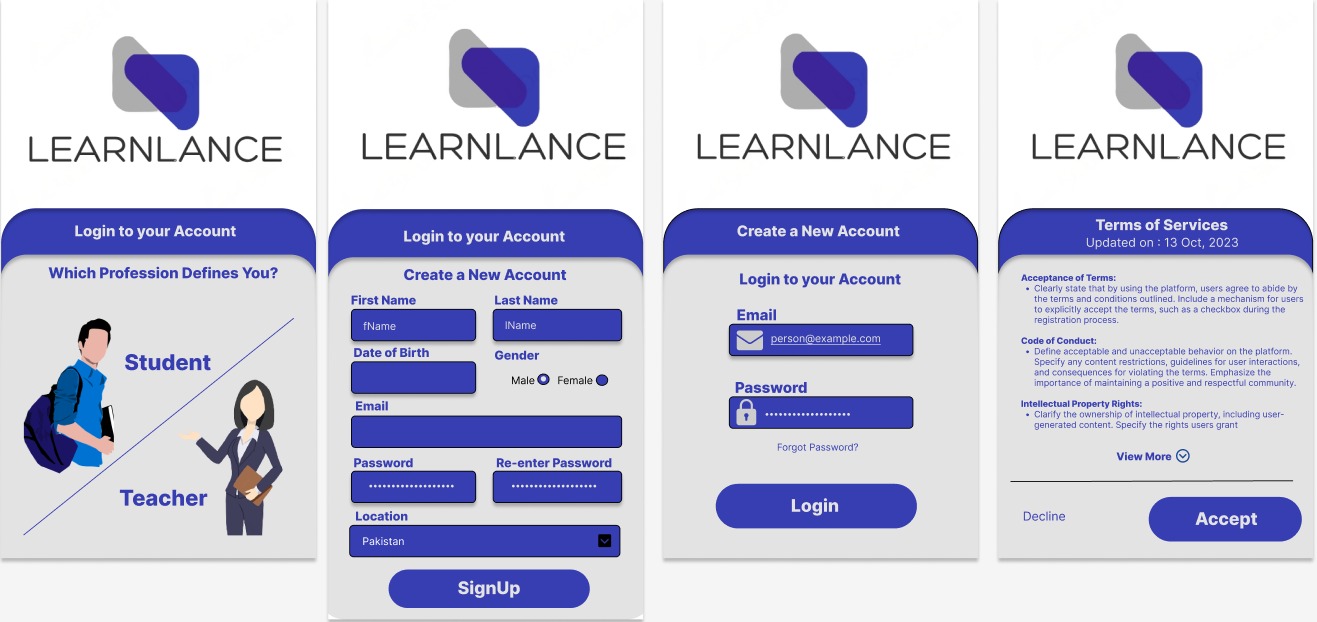
|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Community Users | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| community\_id | ObjectID | 24 | 12 | Primary Key, Foreign Key (References Community), Not Null | - | Community Id | 5f1a8b9d2c4c5129a34e5911 |
| user\_id | ObjectID | 24 | 12 | Primary Key, Foreign Key (References Users), Not Null | - | User Id | 5f1a8c1d2c4c5129a34e5912 |
| status | String | Varies | Varies | Not Null | - | - | Accepted |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Skills | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| skill\_id | ObjectID | 24 | 12 | Primary Key, Not Null | - | Skill Id | 5f1a8b9d2c4c5129a34e5911 |
| name | String | Varies | Varies | Not Null | - | - | Programming |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Skills Required | | | | | | | |
| Field Name | Data Type | Size | Bytes | Constraints | Range | Alias | E.g. of Legitimate Value |
| skill\_id | ObjectID | 24 | 12 | Primary Key, Foreign Key (References Skills), Not Null | - | Skill Id | 5f1a8b9d2c4c5129a34e5911 |
| bid\_id | ObjectID | 24 | 12 | Foreign Key (References Bid Topic), Not Null | - | Bid Id | 5f1a8c1d2c4c5129a34e5912 |

### **Wireframes and Prototypes:**

Here are a few wireframes for your reference.



Screens screenshots of a phone

Description automatically generated

A screenshot of a cell phone

Description automatically generated

A screenshot of a social media post

Description automatically generated

A screenshot of a cell phone

Description automatically generated

For detailed prototypes and wireframes, please refer to the links below.

For teacher portal GUI, please refer to: [*https://www.figma.com/file/autIYX24gtnqeD80YnBGHP/Teacher-Screens?type=design&node-id=0-1&mode=design&t=DHPjkrraxuD8ZUi9-0*](https://www.figma.com/file/autIYX24gtnqeD80YnBGHP/Teacher-Screens?type=design&node-id=0-1&mode=design&t=DHPjkrraxuD8ZUi9-0)

For teacher portal prototypes, please refer to: *https://www.loom.com/share/c5820ab5afb64d37bde8ae82cda2a88a?sid=18edf4e7-a44c-4635-9ca5-86cf2eaa3a8d*

For student portal GUI, please refer to: [*https://www.figma.com/file/AAxadUoBXs4thN3cgk1i7v/Student-Screens?type=design&node-id=0-1&mode=design&t=gSjMdQUTGIoYbKo8-0*](https://www.figma.com/file/AAxadUoBXs4thN3cgk1i7v/Student-Screens?type=design&node-id=0-1&mode=design&t=gSjMdQUTGIoYbKo8-0)

For student portal prototypes, please refer to: *https://www.loom.com/share/413214ee36c3425eb676ff9933f8c15c?sid=0ebde987-6ccf-4d8c-a7ec-f6ce0dcc111a*

For admin portal GUI, please refer to: [*https://www.figma.com/file/XxTUAkDdXYv6Ly4GwSQD8a/Admin-Screens?type=design&node-id=0-1&mode=design&t=U8Qb7aI3ffxWfigq-0*](https://www.figma.com/file/XxTUAkDdXYv6Ly4GwSQD8a/Admin-Screens?type=design&node-id=0-1&mode=design&t=U8Qb7aI3ffxWfigq-0)

For admin portal prototypes, please refer to: [*https://www.loom.com/share/edc17200ffc14f639bd011701f8e4744?sid=14135bf1-0884-4464-8de0-e4844dca64f7*](https://www.loom.com/share/edc17200ffc14f639bd011701f8e4744?sid=14135bf1-0884-4464-8de0-e4844dca64f7)

### **Report Specification Template for Payment Invoice:**

|  |  |
| --- | --- |
| Report Element | Element Description |
| Report ID | INVOICE-01 |
| Report Title | Payment Invoice  (Title will be presented on center at top of the page)  Query Parameters:   * For Course Enrollment: Student ID, Course ID, Date Range * For Topic Request: Student ID, Teacher ID, Topic ID, Date Range |
| Report Purpose | This report serves to generate payment invoices for students who enroll in courses or request topics. It includes details of the course or topic, along with associated charges, providing a transparent overview of financial transactions on LearnLance. |
| Decisions Made from Report | * Approval of student payments for course enrollment or topic requests. * Tracking and verification of financial transactions on LearnLance. |
| Priority | High |
| Report Users | * Admins for approval * Students for payment verification |
| Data Sources | LearnLance Database |
| Frequency and disposition | Dynamic report  Generated on-demand.  Triggered by course enrollment or topic request teacher selection.  Automatic generation after admin approval  Report will be received by admin and student via email. |
| Latency | Immediate delivery upon request |
| Visual Layout | Portrait  Current data on runtime  A4 paper size for hard-copy reports  Dotted line graph with payment on x and date on y scaling appearing at top of the report |
| Header and Footer | Report Title: Bold, 16pt, centered.  Page numbering: "Page x of y," 10pt, right aligned.  Report run timestamp: 10pt, left-aligned.  Organization identification: Logo and company name, centered.  If organization name is exceeding 10 characters so truncate it. |
| Report Body | * Fields to Include:  1. For Course Enrollment: 2. Student Name 3. Course Name 4. Course Charges 5. For Topic request: 6. Student Name 7. Teacher Name 8. Topic Name 9. Teacher's Bid Amount  * Record Selection Criteria: Logic for selecting enrolled courses or requested topics. * User-specified Text or Parameters: Customizable field labels for student and teacher names. * Column and Row Heading:  1. Text: Bold, 12pt, left-aligned 2. Data: Regular, 11pt, left-aligned  * Display Format:  1. Currency: Bold, 11pt, right aligned 2. Numeric Rounding: Two decimal places  * Calculations: Total charges for courses or bids * Sort Criteria: By student name * Filter Criteria: Approval status (admin-approved) * Grouping and Subtotals: Subtotals for total charges * Paging Criteria: None (Single page) |
| End-of-Report Indicator | Horizontal line |
| Interactivity | None |
| Security Access Restrictions | N/A |

### **Use Case Diagram and Use Case Narratives:**

**Profile Management System**



1-

|  |  |  |
| --- | --- | --- |
| **Use-Case Name:** | Sign Up | **Use-Case Type:**  User Requirement |
| **Use-Case ID:** | LL-UUC-1.0.0-1 |
| **Priority:** | High |
| **Source:** | Requirement LL-001 |
| **Primary Business Actors:** | Student/Teacher | |
| **Other Interested Stakeholders:** | Marketing: Marketing is interested in understanding the sign-up process as it influences user acquisition and may inform promotional strategies based on user demographics and preferences. | |
| **Description:** | This use case delineates the process for a new user within the LearnLance system to sign up. During the sign-up process, the user is prompted to provide basic information, including first name, last name, email, date of birth, password, gender, and country. For students, additional information such as interests, language, education, and bio information is required. Meanwhile, teachers are asked to provide information such as language, bio information, education, experience, honors and awards, projects, certificates, and skills. After successful validation, user will be the part of LearnLance system. | |
| **Precondition:** | The sign-up process is initiated by a new user, either a student or a teacher, interested in joining LearnLance. | |
| **Trigger:** | This use case is initiated when a new user clicks on the "Sign Up" button on the LearnLance platform. | |
| **Typical Course of Events:** | **Actor Action** | **System response** |
| **Step 1:** The new user begins the sign-up process by selecting the role as student or teacher and then clicking the ‘Sign Up’ button, on navigating user will provide basic information and additional details based on the selected role. | **Step 2**: LearnLance responds by validating the provided information and creating a new user profile.  **Step 3:** Upon successful validation, the user receives confirmation of their sign-up, and their account is activated. |
| **Alternate Courses:** | **Alt-Step 1:** If the system's backend server is down or not responding, the sign-up process is interrupted. **Alt-Step 2:** If the provided information does not meet validation requirements, the system prompts the user to correct the errors. | |
| **Conclusion:** | This use case concludes with the new user receiving confirmation of their successful sign-up and the activation of their LearnLance account. | |
| **Postcondition:** | The user now has an active LearnLance account with the provided information and is ready to explore the platform. | |
| **Business Rules:** | N/A | |
| **Implementation Constraints and Specifications:** | **IC-:** Sign Up button must be prominently displayed on the LearnLance platform.  **IC-:** The sign-up process includes fields for both basic and additional information based on user type (student or teacher). | |
| **Assumptions:** | The sign-up request is made by an individual interested in joining LearnLance, either as a student or a teacher. | |
| **Open Issues:** | N/A | |

**Course Management System**

A diagram of a diagram

Description automatically generated

1-

|  |  |  |
| --- | --- | --- |
| **Use-Case Name:** | Create Course | **Use-Case Type:**  User Requirement |
| **Use-Case ID:** | LL-UUC-1.0.0-2 |
| **Priority:** | High |
| **Source:** | Requirement LL-001 |
| **Primary Business Actors:** | Teacher | |
| **Other Interested Stakeholders:** | Marketing: Interested in sales activity planning new promotions.  Student: Eager to enroll in courses that align with their interests. | |
| **Description:** | This use case illustrates the process where a validated teacher initiates the creation of a new course on LearnLance. Upon verification, the teacher can create a course and then the teacher is prompted to input essential details such as course name, course description, course cover photo, language, fees, and duration. Input validation mechanisms are implemented to ensure the accuracy of the provided details. After creating a course, the course creator has the option to either leave it in draft status or publish the course. If left in draft, they can return at any time to finalize the course details and publish it. Once published, they have the flexibility to make updates whenever necessary. Additionally, if no students have enrolled in the course, they retain the ability to delete it. Furthermore, the course creator has the capability to send joint account requests to other teachers, inviting them to become course partners. Additionally, they can view a list of course partners, see requested teachers, and withdraw the request if it has not been accepted by the other teacher. Conversely, teachers who receive requests to join a course as instructors can either accept or decline the invitation. They also have visibility into the joint account requests they are receiving. | |
| **Precondition:** | The creation of course must be made by a registered teacher of LearnLance. | |
| **Trigger:** | This use case is initiated when a Create New Course Button is pressed on the Administrative Tools Page. | |
| **Typical Course of Events:** | **Actor Action** | **System response** |
| **Step 1:** The LearnLance teacher initiates the course creation process by clicking the "Create Course" button. They proceed to input the course name, course description, language, fees, and duration, and upload a cover photo for the course. Subsequently, the user clicks on the "Create Course" button to finalize the creation process. | **Step 2**: LearnLance responds by verifying the provision of all required information.  **Step 3**: The system records the request for creating a course and renders the course visible to all LearnLance students.  **Step 4:** By default, the system designates the creator of the course as the course creator.  **Step 5:** Upon community creation, the creator is granted full privileges to monitor and manage course content. |
| **Alternate Courses:** | **Alt-Step 2:** If the teacher misses out on any required field or inserts incorrect or incomplete data, the teacher is notified of the discrepancy and prompted to resubmit.  **Alt-Step 3:** If the system's backend server is down or not responding, the requested course creation is interrupted. | |
| **Conclusion:** | This use case concludes with the teacher receiving confirmation of the successful creation of a new course. | |
| **Postcondition:** | The course has been successfully created, granting the creator privileges to monitor course content. The course is recommended to students whose interests align with the nature of the course. | |
| **Business Rules:** | **Facts:**  **BRF-22:** Each course has a unique identifier.  **BRF-23:** Each course is associated with specific course fees.  **BRF-24:** Each course specifies a primary language.  **BRF-25:** Each course offers the option of earning a certificate.  **BRF-28:** Each lesson item within a course has a defined duration.  **BRF-29:** Each lesson item in a course is assigned a specific item order.  **Inference:**  **BRI-6:** If a course has many members enrolled, then it is considered an informative course.  **BRI-7:** If a course has fewer members enrolled, then it is considered an uninformative course.  **Constraints:**  **BRC-6:** Joint accounts can only be created for courses.  **BRC-13:** Quizzes are only created by course creators.  **Action Enablers:**  **BRA-22:** If a course is created by a teacher, then the recommendation engine will start recommending the course to students. | |
| **Implementation Constraints and Specifications:** | **IC-1:** Create Course Button must be floating at Administrative Tools page.  **IC-2:** Creating Course must allow to preview what is being entered. | |
| **Assumptions:** | Create Course request is made by authorized teacher of LearnLance | |
| **Open Issues:** | N/A | |

2-

|  |  |  |
| --- | --- | --- |
| **Use-Case Name:** | Enroll in Course | **Use-Case Type:**  User Requirement |
| **Use-Case ID:** | LL-UUC-1.0.0-3 |
| **Priority:** | High |
| **Source:** | Requirement LL-001 |
| **Primary Business Actors:** | Student | |
| **Other Interested Stakeholders:** | Marketing: Interested in sales activity planning new promotions.  Teacher: Eager to create courses that resonate with the interests of students. | |
| **Description:** | This use case illustrates the process where a validated student initiates the enrollment in a course on LearnLance. Initially, the system proactively suggests courses to students based on their interests. Students have the capability to explore courses using filters such as topic and fees. Once a decision is made, the student can initiate enrolment by clicking the "Enrol" button. Subsequently, the system prompts the student to proceed with the payment for the selected course. At this juncture, the student can opt to either retain the course in the cart or progress to complete the payment. If the student completes the payment, the transaction enters a pending status until the admin verifies the payment. The admin holds the authority to either approve or reject the payment. Upon approval, the student gains access to all course content. | |
| **Precondition:** | The enrollment in a course must be made by a registered student of LearnLance. | |
| **Trigger:** | This use case is initiated when an Enroll in Course Button is pressed on the Course page or E-Learning page. | |
| **Typical Course of Events:** | **Actor Action** | **System response** |
| **Step 1:** The LearnLance student initiates the course enrollment process by clicking the "Enroll in Course" button. They navigate to the cart page where they are prompted to input their account number. Subsequently, the student clicks on the "Make Payment" button to conclude the enrolment process. | **Step 2**: LearnLance prompts the admin to verify the payment.  **Step 3**: The system records the enrolment request, notifying the student daily about the payment status.  **Step 4:** Once verified, the system adds the course to the student's "My Courses" section.  **Step 5:** Upon course enrollment, the student is granted full privileges to access the course content. |
| **Alternate Courses:** | **Alt-Step 2:** If the admin does not verify the student's payment, access to the course content will be restricted.  **Alt-Step 3:** If student provides incorrect account information, the admin may reject the payment, leading to non-access to the course content.  **Alt-Step 4:** If the system's backend server is down or not responding, the requested course enrollment is interrupted. | |
| **Conclusion:** | This use case concludes with the student receiving confirmation of the successful enrollment in a course. | |
| **Postcondition:** | The student has been successfully enrolled in a course, granting the student privileges to access course content. The course will now appear in the "My Courses" section for the student. | |
| **Business Rules:** | **Facts:**  **BRF-35:** Every payment is linked to a unique identifier.  **BRF-36:** Payment corresponds to specific amounts.  **Inference:**  **BRI-10:** If a student scores well in a course, then he/she is considered intelligent.  **BRI-11:** If a student scores less in a course, then he/she is considered below average.  **Action Enablers:**  **BRA-23:** If a student enrols in a course, then the course creator will be notified.  **BRA-26:** If a student's payment for a course is verified, then they can access the course resources.  **BRA-31:** If a payment is made by a student, then the admin will be notified.  **BRA-24:** If a student enrols in a course, then the student will be prompted to make a payment.  **BRA-25:** If a student's payment for a course is verified, then they will be notified. | |
| **Implementation Constraints and Specifications:** | **IC-1:** Enroll in Course Button must be floating at E-Learning page or Course page. | |
| **Assumptions:** | Enroll in Course request is made by authorized student of LearnLance | |
| **Open Issues:** | N/A | |

**Community and Post Management System**

A diagram of a diagram

Description automatically generated

1-

|  |  |  |
| --- | --- | --- |
| **Use-Case Name:** | Create Community | **Use-Case Type:**  User Requirement |
| **Use-Case ID:** | LL-UUC-1.0.0-4 |
| **Priority:** | High |
| **Source:** | Requirement LL-001 |
| **Primary Business Actors:** | Student/ Teacher | |
| **Other Interested Stakeholders:** | Marketing: Interested in sales activity planning new promotions. | |
| **Description:** | This use case illustrates the process where a validated User, whether a student or Teacher, initiates the creation of a new community on LearnLance. Upon verification, the User can create a Community and then the User is prompted to input essential details such as Community Name, Community Description, and Community Cover Photo. Input validation mechanisms are implemented to ensure the accuracy of the provided details. After successful creation, the new community becomes a part of LearnLance's community network, initially with a single member—the creator. The creator has the authority to generate Community Posts, visible to all community members. As the community creator, the User holds the privilege to edit or delete any community post and its associated comments. While the community creator cannot exit the community, they can resolve it by deleting it. The act of creating a community grants these specified rights. | |
| **Precondition:** | The creation of community request must be made by a registered user (Student/Teacher) of LearnLance. | |
| **Trigger:** | This use case is initiated when a Create New Community Button is pressed on the All-Communities Page. | |
| **Typical Course of Events:** | **Actor Action** | **System response** |
| **Step 1:** The LearnLance user initiates the community creation process by clicking the "Create Community" button. They proceed to input the Community Name, Community Description, and upload a cover photo for the community. Subsequently, the user clicks on the "Create Community" button to finalize the creation process. | **Step 2**: LearnLance responds by verifying the provision of all required information.  **Step 3**: The system records the request for creating a community and renders the community visible to all LearnLance members.  **Step 4:** By default, the system designates the creator of the community as the community's creator.  **Step 5:** The system includes the creator in the list of community members.  **Step 6:** Upon community creation, the creator is granted full privileges to monitor and manage community posts. |
| **Alternate Courses:** | **Alt-Step 2:** If the member misses out on any required field or inserts incorrect or incomplete data, the user (student/teacher) is notified of the discrepancy and prompted to resubmit.  **Alt-Step 3:** If the system's backend server is down or not responding, the requested community creation is interrupted. | |
| **Conclusion:** | This use case concludes with the user (student/teacher) receiving confirmation of the successful creation of a new community. | |
| **Postcondition:** | The community has been successfully created, granting the creator privileges to monitor community posts. The creator is automatically added as a member, and the newly created community is also included in the list of joined communities. | |
| **Business Rules:** | **Facts:**  **BRF-19:** Every community is assigned a unique identifier.  **BRF-20:** Communities consist of members.  **BRF-21:** Every community is created by a designated community creator.  **Inference:**  **BRI-4:** If a community has many members, then it is considered an active community.  **BRI-5:** If a community has fewer members, then it is considered an inactive community.  **Constraints:**  **BRC-2:** The community creator has the exclusive authority to approve membership requests.  **BRC-3:** The community creator is solely authorized to remove members from the community.  **BRC-4:** Communities can be deleted exclusively by their creators.  **Action Enablers:**  **BRA-21:** If a community is deleted, then all its posts will be deleted. | |
| **Implementation Constraints and Specifications:** | **IC-1:** Create Community Button must be floating at All Communities page.  **IC-2:** Creating Community must allow to preview what is being entered. | |
| **Assumptions:** | Create Community request is made by authorized member of LearnLance | |
| **Open Issues:** | N/A | |

2-

|  |  |  |
| --- | --- | --- |
| **Use-Case Name:** | Create Post | **Use-Case Type:**  User Requirement |
| **Use-Case ID:** | LL-UUC-1.0.0-5 |
| **Priority:** | High |
| **Source:** | Requirement LL-001 |
| **Primary Business Actors:** | Student/ Teacher | |
| **Other Interested Stakeholders:** | N/A | |
| **Description:** | This use case illustrates the process where a validated User, whether a student or Teacher, initiates the creation of a new post on its own profile of LearnLance. Upon verification, the User can create a post and then the User is prompted to input essential details such as Post Description, and any media (Photo/Video). Input validation mechanisms are implemented to ensure the accuracy of the provided details. After successful creation, the new post becomes a part of ones’ post. This post will be visible to all connected members or followers. As the creator, the User holds the privileges to edit or delete his/her post and its associated comments. | |
| **Precondition:** | The creation of post request must be made by a registered user (Student/Teacher) of LearnLance. | |
| **Trigger:** | This use case is initiated when a Create New Post Button is pressed on the User Profile Page. | |
| **Typical Course of Events:** | **Actor Action** | **System response** |
| **Step 1:** The LearnLance user initiates the post creation process by clicking the "Create Post" button. Then proceed to input the Post Description and upload media for the post. Subsequently, the user clicks on the "Create Post" button to finalize the creation process. | **Step 2**: LearnLance responds by verifying the provision of all required information.  **Step 3**: The system records the request for creating a post and renders the created post visible to all connected/follower members.  **Step 4:** Upon post creation, the creator is granted privileges to edit and delete posts. |
| **Alternate Courses:** | **Alt-Step 2:** If the member misses out on any required field or inserts incorrect or incomplete data, the user (student/teacher) is notified of the discrepancy and prompted to resubmit.  **Alt-Step 3:** If the system's backend server is down or not responding, the requested post creation is interrupted. | |
| **Conclusion:** | This use case concludes with the user (student/teacher) receiving confirmation of the successful creation of new user’s post. | |
| **Postcondition:** | The post has been successfully created, and the newly created post comes on connected/following users’ feed. | |
| **Business Rules:** | **Facts:**  **BRF-37:** Each post is identified by a unique identifier.  **BRF-38:** Posts can receive likes from users.  **BRF-39:** Posts can have comments from users.  **Inference:**  **BRI-2:** If a post has many likes, then it is considered a valuable post.  **BRI-3:** If a post has fewer likes, then it is considered an invaluable post.  **Constraints:**  **BRC-11:** Posts can be removed by the post creator.  **Action Enablers:**  **BRA-17:** If an individual post is created, then it will appear on the feed of the creator's connections.  **BRA-18:** If a post is liked, then the post creator will be notified.  **BRA-19:** If a post is unliked, then the total number of likes for that post will be updated.  **BRA-13:** If a comment is made to a post, then the post creator will be notified.  **BRA-14:** If a comment is deleted for a post, then the post's comments will be modified.  **BRA-15:** If a reply for a comment is deleted, then the comment's replies will be modified. | |
| **Implementation Constraints and Specifications:** | **IC-3:** Social Hub and Connections must be made (Separate Use Cases)  **IC-4:** News feed refreshes after reloading by the user itself. | |
| **Assumptions:** | Create Post request is made by authorized member of LearnLance | |
| **Open Issues:** | N/A | |

3-

|  |  |  |
| --- | --- | --- |
| **Use-Case Name:** | Join Community | **Use-Case Type:**  User Requirement |
| **Use-Case ID:** | LL-UUC-1.0.0-6 |
| **Priority:** | High |
| **Source:** | Requirement LL-001 |
| **Primary Business Actors:** | Student/ Teacher | |
| **Other Interested Stakeholders:** | Marketing: Marketing is interested in optimizing community join processes to enhance platform visibility and user engagement, influencing promotional strategies.  Management: Management is concerned with overseeing community join processes to ensure user satisfaction, contributing to strategic decisions and overall platform success. | |
| **Description:** | This use case outlines the procedure for any user within the LearnLance system to join a community. Users have the option to join a community either through recommended communities or by utilizing the community search page. Upon selecting a specific community, a user can send a request to join, and upon acceptance, they become a member of that community. Subsequently, the user gains the ability to publish posts within the community and has the option to leave the community at any time. | |
| **Precondition:** | The join request must be made by a registered user (Student/Teacher) of LearnLance to an approved community in LearnLance system. | |
| **Trigger:** | This use case is initiated when a Join Community Button is pressed on the Single Community Page. | |
| **Typical Course of Events:** | **Actor Action** | **System response** |
| **Step 1:** The LearnLance user begins the community joining process by exploring recommended communities or utilize the community search page to find a specific community. After choosing a community, the user can click the "Join Community" button to send a join request. | **Step 2**: LearnLance responds by adding new request to pending requests of community creator.  **Step 3:** Upon approval, the user becomes a member of the community, gaining the ability to publish posts. |
| **Alternate Courses:** | **Alt-Step 1:** If the system's backend server is down or not responding, community’s joining process is interrupted. **Alt-Step 2:** If community creator rejects the request, the request will be removed from pending requests. | |
| **Conclusion:** | This use case concludes with the user (student/teacher) receiving notification of the approval of his request to join a community. | |
| **Postcondition:** | The user is successfully added to the community, granting privileges to manage posts, including the ability to add, edit, and delete them. Additionally, the user retains the option to leave the community anytime. | |
| **Business Rules:** | **Facts:**  **BRF-20:** Communities consist of members.  **Constraints:**  **BRC-2:** The community creator has the exclusive authority to approve membership requests. | |
| **Implementation Constraints and Specifications:** | **IC-:** Join Community Button must be floating at Single Community page.  **IC-:** The Community will be listed in all community’s page. | |
| **Assumptions:** | Join Community request is made by authorized member of LearnLance | |
| **Open Issues:** | N/A | |

**Joint Accounts Management System**

A diagram of a function

Description automatically generated

1-

|  |  |  |
| --- | --- | --- |
| **Use-Case Name:** | Manage Invitation Request | **Use-Case Type:**  User Requirement |
| **Use-Case ID:** | LL-UUC-1.0.0-7 |
| **Priority:** | High |
| **Source:** | Requirement LL-001 |
| **Primary Business Actors:** | Teacher | |
| **Other Interested Stakeholders:** | N/A | |
| **Description:** | This use case illustrates the process where a validated teacher manages the joint account request, upon the invitation of some other teacher for a course on LearnLance. Initially, the teacher clicks on "Joint Account Requests." Upon clicking, they can view all joint account requests, organized by the name of the teacher who sent the request and the course for which the request was made. The teacher can click on any request to access a detailed message from the other teacher, including information about the course and the teacher who created it for joint account purposes. The teacher has the option to either accept or decline the request, and upon acceptance, both teachers can communicate through chat and call for further discussions. | |
| **Precondition:** | The joint account requests must be viewed by a registered teacher of LearnLance. | |
| **Trigger:** | This use case is initiated when a Joint Account Request Button is pressed on the main menu on the home page. | |
| **Typical Course of Events:** | **Actor Action** | **System response** |
| **Step 1:** The LearnLance teacher initiates the viewing of joint account requests by clicking the "Joint Account Request" button. They go to the joint account requests page where they can view the details of individual requests. Subsequently, the teacher clicks on "Accept" or "Decline" to confirm or reject the request, respectively, concluding the process of managing joint account requests. | **Step 2**: LearnLance prompts the teacher to confirm the acceptance or rejection of the request.  **Step 3**: The system records the final decision of teacher, notifying the course teacher about the joint account request.  **Step 4:** Once completed, the system updated the joint account requests of the teacher in "Joint Account Requests" section. |
| **Alternate Courses:** | **Alt-Step 2:** If the system's backend server is down or not responding, the requested management of joint account requests is interrupted. | |
| **Conclusion:** | This use case concludes with the course teacher and teacher receiving confirmation of the acceptance or rejection of joint account request for a course. | |
| **Postcondition:** | The teacher has successfully managed the joint account request for a course. The joint account request will now be removed from the "Joint Account Requests" section for the teacher. | |
| **Business Rules:** | **Facts:**  **BRF-26:** Every joint account is identified by a unique identifier.  **BRF-27:** Joint accounts include multiple teachers.  **Constraints:**  **BRC-5:** Joint accounts can only be generated by teachers.  **BRC-6:** Joint accounts can only be created for courses.  **Action Enablers:**  **BRA-27:** If a joint account request has been sent by a teacher, then the invited teacher will be notified.  **BRA-28:** If the invited teacher for a joint account accepts the request, then the request sender will be notified. | |
| **Implementation Constraints and Specifications:** | **IC-1:** Joint Account Request Button must be floating at main menu on home page for a teacher. | |
| **Assumptions:** | Manage Joint Accounts Request is made by authorized teacher of LearnLance | |
| **Open Issues:** | N/A | |

**Communication Management System**

A diagram of communication management system

Description automatically generated

1-

|  |  |  |
| --- | --- | --- |
| **Use-Case Name:** | Send Message | **Use-Case Type:**  User Requirement |
| **Use-Case ID:** | LL-UUC-1.0.0-8 |
| **Priority:** | High |
| **Source:** | Requirement LL-001 |
| **Primary Business Actors:** | Student/ Teacher | |
| **Other Interested Stakeholders:** | Customer Support: Customer support is interested in the send message process as it may involve resolving user issues related to messaging, ensuring a seamless communication experience, and addressing any technical difficulties users may encounter. | |
| **Description:** | This use case details the process for a user to send messages within the LearnLance chat system. Users can initiate conversations with other users by selecting the "Send Message" option or by opening the existing chat. During the message creation process, the sender inputs the message content and selects the recipient. The sender and receiver may be either students or teachers. The system facilitates real-time communication between users by delivering messages promptly. | |
| **Precondition:** | The sender and receiver must be registered users on LearnLance. | |
| **Trigger:** | This use case is initiated when a user selects the "Send Message" option or opens the existing chat within the LearnLance chat system. | |
| **Typical Course of Events:** | **Actor Action** | **System response** |
| **Step 1:** The sender initiates the message creation process by selecting the "Send Message" option or by opening an existing chat, user then types in the message and sends it to the recipient. | **Step 2**: LearnLance responds by delivering the message promptly to the selected recipient. **Step 3:** The recipient receives a notification of the incoming message and can view the message content.  **Step 4**: LearnLance updates the chat log of sender. |
| **Alternate Courses:** | **Alt-Step 1:** If the system's backend server is down or not responding, the message creation process is interrupted. | |
| **Conclusion:** | This use case concludes with the sender successfully sending a message to the selected recipient and updating chat log, facilitating real-time communication within the LearnLance chat system. | |
| **Postcondition:** | The message is delivered to the recipient, who can view and respond to it within the LearnLance chat system. | |
| **Business Rules:** | **Action Enablers:**  **BRA-10:** If a message is sent, then the receiver(s) will be notified.  **BRA-11:** If a message is sent, then the chat will be updated in the chat log. | |
| **Implementation Constraints and Specifications:** | **IC-:** "Send Message" option must be easily accessible within the LearnLance chat interface.  **IC-:** Existing chats should be listed in “My Chats” page.  **IC-:** The system must provide real-time delivery of messages. | |
| **Assumptions:** | The sender and recipient are both registered users on LearnLance and have active accounts. | |
| **Open Issues:** | N/A | |

2-

|  |  |  |
| --- | --- | --- |
| **Use-Case Name:** | Initiate Video Call | **Use-Case Type:**  User Requirement |
| **Use-Case ID:** | LL-UUC-1.0.0-9 |
| **Priority:** | High |
| **Source:** | Requirement LL-001 |
| **Primary Business Actors:** | Student/ Teacher | |
| **Other Interested Stakeholders:** | Customer Support: Customer support is interested in the initiate video call process as it may involve addressing user inquiries or issues related to video call functionality, ensuring a smooth and positive user experience.  Technical Support: Technical support has a stake in the initiate video call process to troubleshoot and resolve any technical issues that users may encounter during video call initiation, ensuring the reliability and functionality of the video call feature. | |
| **Description:** | This use case outlines the process for a user to initiate a video call within the LearnLance platform. Users can seamlessly initiate video calls by selecting the "Video Call" option. During the video call initiation, the caller selects the desired recipient or list of recipients for a group call, and the system establishes a secure video connection when receiver answers the call. The participants, whether students or teachers, engage in real-time video communication for an enhanced learning experience. After successful initiation, the call entry will be saved in LearnLance system to maintain history of user’s past calls | |
| **Precondition:** | The caller and receiver(s) must be registered users on LearnLance with devices equipped for video calls. | |
| **Trigger:** | This use case is initiated when a user selects the "Video Call" option within the LearnLance platform. | |
| **Typical Course of Events:** | **Actor Action** | **System response** |
| **Step 1:** The caller starts the video call initiation process by selecting the "Video Call" option. | **Step 2**: The system prompts the caller to select the recipient(s) for the video call.  **Step 3:** The recipient(s) receives a notification of the incoming video call. Upon receiving, the system establishes a secure video connection with the recipient.  **Step 4:** The system stores the call information in the system as a history record for user. |
| **Alternate Courses:** | **Alt-Step 1:** If the system's backend server is down or not responding, the video call initiation process is interrupted. **Alt-Step 2:** If recipient(s) rejects or missed the call, the video call will not be initiated. | |
| **Conclusion:** | This use case concludes with the caller successfully initiating a video call with the selected recipient and storing call data in the system. | |
| **Postcondition:** | The video call session is established, enabling participants to engage in real-time visual communication. | |
| **Business Rules:** | **Facts:**  **BRF-4:** Every call is associated with a unique identifier.  **BRF-7:** Calls can be individual calls.  **BRF-8:** Calls can be group calls.  **Action Enablers:**  **BRA-7:** If a call is initiated, then the call recipient(s) will be notified.  **BRA-8:** If a call is disconnected, then the call record will be maintained in the call log. | |
| **Implementation Constraints and Specifications:** | **IC-:** "Video Call" option must be shown on every chat header in chat log.  **IC-:** The system must establish a secure video connection between the caller and the recipient. | |
| **Assumptions:** | The caller and recipient(s) are both registered users on LearnLance with devices capable of supporting video calls. | |
| **Open Issues:** | N/A | |

**Social Hub Management System**

A diagram of a social hub

Description automatically generated

1-

|  |  |  |
| --- | --- | --- |
| **Use-Case Name:** | Manage Connection Request | **Use-Case Type:**  User Requirement |
| **Use-Case ID:** | LL-UUC-1.0.0-10 |
| **Priority:** | High |
| **Source:** | Requirement LL-001 |
| **Primary Business Actors:** | Student/ Teacher | |
| **Other Interested Stakeholders:** | N/A | |
| **Description:** | This use case illustrates the process where a validated user (student/teacher) manages the connection request, upon the invitation of some other user (student/teacher). Initially, the user (student/teacher) clicks on "Connect." It should be noted that only two students or two teachers can connect. Upon clicking, the connection request is sent to the second user (student/teacher). The use-case initiates when second user (student/teacher) can either accept or decline the request by clicking on “Accept” or “Decline” respectively. Upon acceptance, it is notified to the first user (student/teacher). The first user (student/teacher) can withdraw the request by clicking on “Withdraw” unless it is not accepted by the second user (student/teacher). Moreover, if two users (student/teacher) are friends, then either of them can unfriend the other. Apart from this, two friends (connected) can communicate with one another through chats and calls. At last, the system updated the connections of the user (student/teacher) in "My Connections" section upon acceptance and the connection requests in “Pending Connections” upon rejection. | |
| **Precondition:** | The connection requests must be managed by a registered user (student/teacher) of LearnLance. | |
| **Trigger:** | This use case is initiated when a user (student/teacher) accepts or decline the connection request by other user (student/teacher) | |
| **Typical Course of Events:** | **Actor Action** | **System response** |
| **Step 1:** The LearnLance user (student/teacher) initiates the managing of connection requests by clicking the "Accept" or “Decline” button respectively, concluding the process of managing connection requests. | **Step 2**: LearnLance records the final decision of user (student/teacher), notifying the second user (student/teacher) about the acceptance of connection request if accepted.  **Step 3:** Once done, the system updated the connections of the user (student/teacher) in "My Connections" section upon acceptance and the connection requests in “Pending Connections” upon rejection. |
| **Alternate Courses:** | **Alt-Step 2:** If the system's backend server is down or not responding, the requested management of connection requests is interrupted. | |
| **Conclusion:** | This use case concludes with the second user (student/teacher) receiving confirmation of the acceptance of connection request. | |
| **Postcondition:** | The user (student/teacher) has successfully managed the connection request. The connection request will now be removed from the "Pending Requests" section for the user (student/ teacher) and connections of the user (student/teacher) will be updated. | |
| **Business Rules:** | **Constraints:**  **BRC-15:** Students are only permitted to connect with other students.  **BRC-17:** Teachers can only connect with other teachers.  **Inferences:**  **BR-12:** If a teacher has fewer connections, then he/she is considered socially inactive.  **BR-13:** If a student has fewer connections, then he/she is considered socially inactive.  **BR-14:** If a teacher has more connections, then he/she is considered socially active.  **BR-15**: If a teacher has fewer connections, then he/she is considered socially inactive.  **Action Enablers:**  **BRA-34:** If a connection request is sent, then the recipient will be notified.  **BRA-35:** If a connection request is accepted, then the request sender will be notified. | |
| **Implementation Constraints and Specifications:** | **IC-1:** Connect, Unfriend, Withdraw, and Decline Button must be floating at the profile of a user (student/teacher) or on the list of “My Connections” and “Pending Requests”. | |
| **Assumptions:** | Manage Connection Request is made by authorized user (student/teacher) of LearnLance | |
| **Open Issues:** | N/A | |

**Topic Request Management System**

A diagram of a diagram

Description automatically generated

1-

|  |  |  |
| --- | --- | --- |
| **Use-Case Name:** | Bid Topic Request | **Use-Case Type:**  User Requirement |
| **Use-Case ID:** | LL-UUC-1.0.0-11 |
| **Priority:** | High |
| **Source:** | Requirement LL-001 |
| **Primary Business Actors:** | Teacher | |
| **Other Interested Stakeholders:** | Student: The one who will post the Topic Request for bidding.  Marketing: Interested in sales activity planning new promotions. | |
| **Description:** | This use case depicts the process where a validated teacher initiates the bidding for an open topic request made by a student on LearnLance. After verification, the teacher can create a bid topic request and is prompted to write a proposal, including essential details such as a proposal description with duration and quoted price. Input validation mechanisms are in place to ensure the accuracy of the provided details. Once created successfully, the student who initiated the bid topic request can view the proposal and shortlist the teacher. Shortlisting the teacher initiates a chat where both parties can negotiate and communicate regarding the price, duration, or course content for the topic request. After approval from the student, the student pays the fees, and the teacher is notified, and the topic request is closed. They can then be redirected to the live virtual classes, and the payment will be released to the teacher by the admin. | |
| **Precondition:** | The bidding of topic request must be made by a registered teacher of LearnLance.  There must be Open Topic Request that can be bided. | |
| **Trigger:** | This use case is initiated when topic requests are recommended, and teacher selects the Topic Request to bid and click on Bid Topic Request. | |
| **Typical Course of Events:** | **Actor Action** | **System response** |
| **Step 1:** The LearnLance user initiates the community creation process by clicking the "Create Community" button. They proceed to input the Community Name, Community Description, and upload a cover photo for the community. Subsequently, the user clicks on the "Create Community" button to finalize the creation process. | **Step 2**: LearnLance responds by verifying the provision of all required information.  **Step 3**: The system records the request for bidding topic request and notifies student about a bided proposal by a Teacher on Topic Request  **Step 4:** The student selects and shortlists the teacher.  **Step 5:** Both the Teacher and Student communicate.  **Step 6:** Students make payment and so topic request closes.  **Step 7:** They both are directed towards Virtual Classes. |
| **Alternate Courses:** | **Alt-Step 2:** If the member misses out on any required field or inserts incorrect or incomplete data, the teacher is notified of the discrepancy and prompted to resubmit.  **Alt-Step 3:** If the system's backend server is down or not responding, the requested bid topic request is interrupted.  **Alt-Step 5:** If they both are agreed, they might not negotiate.  **Alt-Step 6:** If the topic is not closed the system may malfunction as the topic request is still open.  **Alt-Step 7:** If the redirection fails, User must navigate manually. | |
| **Conclusion:** | This use case concludes with the teacher receiving confirmation of the successful bidding of topic request. | |
| **Postcondition:** | The Bidding on Topic Request has been successfully created, notifying the student requested, and then he will shortlist the teacher making the topic request close. | |
| **Business Rules:** | **Facts:**  **BRF-12:** Every topic request is identified by a unique identifier.  **BRF-13:** Each topic request includes the rate per hour.  **BRF-14:** Each topic request specifies the estimated hours.  **BRF-15:** Every topic request tracks the number of bids received.  **BRF-35:** Every payment is linked to a unique identifier.  **BRF-36:** Payments correspond to specific amounts.  **Inference:**  **BRI-1:** If a student doesn't pay for a topic request, then he/she is considered a 'no-show.'  **Constraints:**  **BRC-1:** Every bid topic must specify the required skills for teachers to teach the topic.  **Action Enablers:**  **BRA-1:** If a bid topic is posted by a student, then the recommendation engine will start recommending topic requests to eligible teachers.  **BRA**-2: If a teacher sends a proposal for a topic request, then the student will be notified of the proposal.  **BRA-3:** If a student accepts the proposal of a teacher, then the teacher will be notified of their new job.  **BRA**-4: If a student accepts the proposal of a teacher, then the student will be prompted to make a payment.  **BRA**-4: If a student's payment for a topic request is verified, then they will be notified.  **Computations:**  **BRCo-1:** The payment for a topic request will be calculated by multiplying the estimated hours and rate per hour requested by the teacher who is hired. | |
| **Implementation Constraints and Specifications:** | **IC-5:** Payment and Topic Request must be created (Separate Use Cases)  **IC-6:** Topic Requests must generate a notification on Student requested for Topic | |
| **Assumptions:** | Bid Topic request is made by authorized teacher of LearnLance | |
| **Open Issues:** | N/A | |

**Session Management System**

A diagram of a diagram

Description automatically generated

1-

|  |  |  |
| --- | --- | --- |
| **Use-Case Name:** | Create Session | **Use-Case Type:**  User Requirement |
| **Use-Case ID:** | LL-UUC-1.0.0-12 |
| **Priority:** | High |
| **Source:** | Requirement LL-001 |
| **Primary Business Actors:** | Teacher | |
| **Other Interested Stakeholders:** | Marketing: Marketing is interested in the create session process as it may influence promotional strategies, content planning, and user engagement activities based on the popularity and dynamics of upcoming live sessions.  Student: Eager to join session that aligns with their interests | |
| **Description:** | This use case outlines the process for a teacher to create an upcoming live session within the LearnLance system. Teachers can seamlessly create sessions by providing details such as the featured image, title, date, and time. Once validated, the session will be created. After successful creation, the session is displayed on the students' upcoming sessions if they follow the teacher and is listed in the teacher's "My Upcoming Sessions" page. On the teacher's backend admin tools, they can edit and delete sessions. Additionally, teachers can transition sessions between draft and published states. | |
| **Precondition:** | The teacher must be a registered user on LearnLance with the appropriate permissions to create and manage sessions. | |
| **Trigger:** | This use case is initiated when a teacher selects the "Create Session" option within the LearnLance platform. | |
| **Typical Course of Events:** | **Actor Action** | **System response** |
| **Step 1:** The teacher initiates the session creation process by selecting the "Create Session" option. | **Step 2**: The system prompts the teacher to input session details, including the featured image, title, date, and time.  **Step 3:** The teacher confirms the details, and the system creates the session, displaying it on the students' upcoming sessions if they follow the teacher and listing it in the teacher's "My Upcoming Sessions" page. |
| **Alternate Courses:** | **Alt-Step 1:** If the system's backend server is down or not responding, the session creation process is interrupted. **Alt-Step 2:** If the provided session details do not meet validation requirements, the system prompts the teacher to correct the errors. | |
| **Conclusion:** | This use case concludes with the teacher successfully creating an upcoming live session, making it visible to students who follow the teacher and listing it in the teacher's "My Upcoming Sessions" page. The teacher can further manage the session through backend admin tools. | |
| **Postcondition:** | The session is created and can be viewed by students following the teacher, and it is listed in the teacher's "My Upcoming Sessions" page. The teacher has the ability to edit, delete, and transition the session between draft and published states. | |
| **Business Rules:** | N/A | |
| **Implementation Constraints and Specifications:** | **IC-:** "Create Session" option must be easily accessible on teacher administrative panel on session’s page.  **IC-:** The system must validate session details to ensure accuracy. | |
| **Assumptions:** | The teacher is a registered user on LearnLance with the necessary permissions to create and manage sessions. | |
| **Open Issues:** | N/A | |

2-

|  |  |  |
| --- | --- | --- |
| **Use-Case Name:** | Join Session | **Use-Case Type:**  User Requirement |
| **Use-Case ID:** | LL-UUC-1.0.0-13 |
| **Priority:** | High |
| **Source:** | Requirement LL-001 |
| **Primary Business Actors:** | Student | |
| **Other Interested Stakeholders:** | Teacher: Teachers have a direct interest in the join session process as it involves student participation in their live sessions. Teachers may use insights from session participation for instructional improvements and content planning.  Marketing: Marketing is interested in the join session process as it may influence promotional strategies and user engagement activities based on the popularity and attendance of live sessions.  Technical Support: Technical support has a stake in the join session process to address any technical issues students or teachers may encounter during the session, ensuring a smooth and positive user experience. | |
| **Description:** | This use case outlines the process for a student to join a live session within the LearnLance system. Live sessions are listed on the student's home page, allowing them to seamlessly click and join. Once joined, students can actively participate by chatting in the session and listening to the content, similar to a Google Meet experience. | |
| **Precondition:** | The student must be a registered user on LearnLance with the necessary permissions to join sessions. | |
| **Trigger:** | This use case is initiated when a student selects a live session to join from the list on their home page. | |
| **Typical Course of Events:** | **Actor Action** | **System response** |
| **Step 1:** The student selects a desired live session from home page listing to join. | **Step 2**: The system prompts the student to confirm their intention to join the session.  **Step 3:** The student confirms, and the system connects them to the live session.  **Step 4:** Once joined, system will allow user to actively participate by chatting and listening to the content. |
| **Alternate Courses:** | **Alt-Step 1:** If the system's backend server is down or not responding, the live session joining process is interrupted. | |
| **Conclusion:** | This use case concludes with the student successfully joining the live session, allowing them to actively participate in the session's activities. | |
| **Postcondition:** | The student is connected to the live session and can engage in chat and content listening features. | |
| **Business Rules:** | **Facts:**  **BRF-5:** Calls can be categorized as video sessions. | |
| **Implementation Constraints and Specifications:** | **IC-:** Live sessions must be prominently displayed on the student's home page.  **IC-:** The system must handle a smooth transition for students joining live sessions. | |
| **Assumptions:** | The student is a registered user on LearnLance with the necessary permissions to join sessions. | |
| **Open Issues:** | N/A | |

### **Use Case Glossary:**

|  |  |  |
| --- | --- | --- |
| Use-Case Name | Use-Case Description | Participating Actors and Roles |
| Signup | Users can create an account by providing necessary information. | User (Primary Actor) |
| Login | Registered users can log into their accounts securely. | User (Primary Actor) |
| Logout | Users can log out of their accounts, terminating the session. | User (Primary Actor) |
| Update Profile | Users can modify and update their profile information. | User (Primary Actor) |
| Search Courses | Students can search and discover courses based on specific criteria. | Student (Primary Actor) |
| Enroll In Course | Students can enroll in a chosen course to access its content. | Student (Primary Actor) |
| Make Course Payment | Students can make secure payments for course enrollment. | Student (Primary Actor)  Admin (Secondary Actor) |
| Access To Course Content | Students can access the content of the course they've enrolled in. | Student (Primary Actor) |
| Complete Course | Students can mark a course as completed upon finishing its content | Student (Primary Actor) |
| Give Feedback | Students can provide feedback on courses they've completed. | Student (Primary Actor)  Teacher (Secondary Actor) |
| Get Certificate | Students receive a certificate upon successfully completing a course. | Student (Primary Actor) |
| Create Course | Teachers can create and publish new courses. | Teacher (Primary Actor) |
| Update Course | Teachers can modify and update course details. | Teacher (Primary Actor) |
| View Drafted Course | Teachers can preview and edit courses in draft mode. | Teacher (Primary Actor) |
| Complete Course Details | Teachers can input all necessary details when creating a course. | Teacher (Primary Actor) |
| View Published Course | Teachers can view details of published courses. | Teacher (Primary Actor) |
| Create Community | Users can establish new online communities. | Student (Primary Actor)  Teacher (Primary Actor) |
| Manage Community | Community creators can oversee and moderate community activities. | Student (Primary Actor)  Teacher (Primary Actor) |
| Update Community | Community creators can modify community details. | Student (Primary Actor)  Teacher (Primary Actor) |
| Search Communities | Users can search and discover communities of interest. | Student (Primary Actor)  Teacher (Primary Actor) |
| Leave Community | Users can exit communities they've joined. | Student (Primary Actor)  Teacher (Primary Actor) |
| Manage Post | Community creators can oversee and moderate community posts. | Student (Primary Actor)  Teacher (Primary Actor) |
| Join Community | Users can become members of communities. | Student (Primary Actor)  Teacher (Primary Actor) |
| Create Community Post | Joined Members can publish new posts within a community. | Student (Primary Actor)  Teacher (Primary Actor) |
| Comment Post | Users can comment on posts within a community. | Student (Primary Actor)  Teacher (Primary Actor) |
| Like Post | Users can express approval by liking community posts. | Student (Primary Actor)  Teacher (Primary Actor) |
| Create Post | Users can create and publish standalone posts. | Student (Primary Actor)  Teacher (Primary Actor) |
| View Connections | Users can see a list of their connections. | Student (Primary Actor)  Teacher (Primary Actor) |
| Invite Connection | Teachers can send connection invitations to others. | Student (Primary Actor)  Teacher (Primary Actor) |
| View Invitation | Teachers can see incoming connection invitations. | Student (Primary Actor)  Teacher (Primary Actor) |
| Withdraw Invitation | Teachers can cancel sent connection invitations. | Student (Primary Actor)  Teacher (Primary Actor) |
| Manage Invitation Request | Teachers can accept or decline connection invitations. | Student (Primary Actor)  Teacher (Primary Actor) |
| View Chats | Users can access their message conversations. | Student (Primary Actor)  Teacher (Primary Actor) |
| Initiate Audio Calls | Users can start audio calls with connections. | Student (Primary Actor)  Teacher (Primary Actor) |
| Send Message | Users can send text messages to connections. | Student (Primary Actor)  Teacher (Primary Actor) |
| Initiate Video Call | Users can start video calls with connections. | Student (Primary Actor)  Teacher (Primary Actor) |
| End Video Call | Users can terminate ongoing video calls. | Student (Primary Actor)  Teacher (Primary Actor) |
| End Audio Call | Users can terminate ongoing audio calls. | Student (Primary Actor)  Teacher (Primary Actor) |
| View My Followers | Users can see a list of individuals following them. | Teacher (Primary Actor) |
| View My Connections | Users can see a list of their connections. | Student (Primary Actor)  Teacher (Primary Actor) |
| Withdraw Connection Requests | Users can cancel pending connection requests. | Student (Primary Actor)  Teacher (Primary Actor) |
| Send Connection Requests | Users can send connection Users to others. | Student (Primary Actor)  Teacher (Primary Actor) |
| Manage Connection Requests | Users can accept or decline connection requests. | Student (Primary Actor)  Teacher (Primary Actor) |
| Follow Teacher | Students can follow teachers to receive updates. | Student (Primary Actor) |
| Search Topic Request | Teachers can search and discover topic requests. | Teacher (Primary Actor) |
| Bid Topic Requests | Teachers can bid on topic requests to offer their expertise. | Teacher (Primary Actor) |
| View Proposals | Students can review proposals received for their topic requests. | Teacher (Primary Actor) |
| Shortlist Teacher | Students can shortlist teachers based on proposals. | Student (Primary Actor) |
| Close Topic Request | Students can close a topic request after selecting a teacher. | Student (Primary Actor) |
| Conduct Live Classes | Teachers can conduct live classes for selected topics. | Teacher (Primary Actor) |
| Create Topic Request | Students can conduct live classes for selected topics. | Student (Primary Actor) |
| View My Topic Request | Students can view a list of their created topic requests. | Student (Primary Actor) |
| Update Topic Request | Students can update details of their topic requests. | Student (Primary Actor) |
| Communicate | Users involved in a topic request can communicate for collaboration. | Student (Primary Actor)  Teacher (Primary Actor) |
| Create Session | Teachers can create sessions for live classes. | Teacher (Primary Actor) |
| Update Session | Teachers can update details of live sessions. | Teacher (Primary Actor) |
| View Drafted Session | Teachers can preview and edit drafted sessions. | Teacher (Primary Actor) |
| Complete Session Details | Teachers provide comprehensive details for live sessions. | Teacher (Primary Actor) |
| View Published Session | Teachers can view details of published live sessions. | Teacher (Primary Actor) |
| Search Session | Teachers can search and discover live sessions. | Student (Primary Actor) |
| Join Session | Teachers can join live sessions. | Student (Primary Actor) |
| End Session | Teachers can join live sessions. | Student (Primary Actor) |
| Start Session | Teachers can start live sessions at scheduled times. | Teacher (Primary Actor) |
| View Revenue | Admin can view reports on Learn Lance's revenue | Admin (Primary Actor) |
| View Reports | Admin can access detailed reports on system activities. | Admin (Primary Actor) |
| Perform Crud Operation | Admin can perform CRUD operations for system management. | Admin (Primary Actor) |

### **User Stories:**

|  |  |  |
| --- | --- | --- |
| Application | Sample Use Case | Corresponding User Story |
| Profile Management System | Signup | As a new user, I want to sign up for LearnLance so that I can create an account and access the platform. |
| Login | As a registered user (Student/Teacher), I want to log in to my profile so that I can access personalized features and information. |
| Logout | As a user (Student/Teacher), I want to log out of my profile to secure my account and data. |
| Update Profile | As a user (Student/Teacher), I want to update my profile information to keep it accurate and relevant. |
| Course Management System | Search Courses | As a Student, I want to search for courses to find courses from the recommended courses. |
| Enroll In Course | As a Student, I want to enroll in a Course of my choice. |
| Make Course Payment | As a Student, I will pay the fees for the Course. |
| Access To Course Content | As a Student, I will get access to the course content to which I paid the fee.  As an Admin, I will approve the payment, so that accessing to course content can be made |
| Complete Course | As a Student, I want to complete the course content (Quiz/Videos/etc.) |
| Give Feedback | As a Student I want to give feedback to the teacher of the course I completed. |
| Get Certificate | As a student, I should receive the certificate with my name on it after the completion of the course. |
| Create Course | As a teacher, I want to create a course and publish that course, so that it can be accessible by students. |
| Update Course | As a teacher, I want to Update the published course content to keep updated content intact. |
| View Drafted Course | As a teacher, I want to make my unpublished course drafted by default by LearnLance, so that later I can make it publish |
| Complete Course Details | As a teacher, I want to input all the course details for an earlier drafted course. |
| View Published Course | As a teacher, I want to view all the published courses I have created on LearnLance. |
| Community and Post Management System | Create Community | As a User (Student/Teacher), I want to create a community. |
| Manage Community | As a creator of community, I should be able to manage community detail so members can stick to the community guidelines. |
| Update Community | As a Creator of community, I can edit the Community details, and can make requested members joined. |
| Search Communities | As a User (Student/Teacher), I should be able to search all, joined, pending communities of mine. |
| Leave Community | As a Joined Member of Community, I should be able to leave the Community. |
| Manage Post | As creator of Community, I can manage what is being post in the the community by deleting it. |
| Join Community | As a User, I can join community by sending pending request,  As the creator, I should be able to add the requested member to Join community. |
| Create Community Post | As a Joined Community Member, I should be able to create a post on the community. |
| Comment Post | As a Joined Member of the Community, I should be able to comment on Community Post. |
| Like Post | As a Joined Member of the Community, I should be able to like on Community Post. |
| Create Post | As a User (Student/Teacher), I should be able to create a Post on my profile so that everyone can see what’s on my mind. |
| Joint Accounts Management System | View Connections | As a User (Student/Teacher), I can view my connections to know to whom I have connected to. |
| Invite Connection | As a Teacher, I can invite any connected teacher to jointly make account for a course. |
| View Invitation | As a Teacher, I can view to whom in the past I have requested too. |
| Withdraw Invitation | As a Teacher, I can withdraw the invitation being sent to the teacher. |
| Manage Invitation Request | As a Teacher, I can edit and delete the invitation request. |
| Communication Management System | View Chats | As a User(student/Teacher), I can view all my previous Single Chats and Group Chats along with their history. |
| Initiate Audio Calls | As a User (Student/Teacher), I can initiate audio call to anyone on LearnLance. |
| Send Message | As a User (Student/Teacher), I can send message to anyone on LearnLance. |
| Initiate Video Call | As a User (Student/Teacher), I can initiate audio call to anyone on LearnLance. |
| End Video Call | As a User (Student/Teacher), I can end video call that I am currently active on. |
| End Audio Call | As a User (Student/Teacher), I can end audio call that I am currently active on. |
| Social Hub Management System | View My Followers | As a teacher, I should be able to view all my following students. |
| View My Connections | As a User (Student/Teacher), I should be able to view all the connections I am connected to. |
| Withdraw Connection Requests | As a User (Student/Teacher), I can disconnect/unfollow myself from others account. |
| Send Connection Requests | As a Student I should be able to send connection request to students.  As a Teacher I should be able to send connection request to Teachers |
| Manage Connection Requests | As a User (Student/Teacher), I should be able to manage my connection request and allow users to connect. |
| Follow Teacher | As a Student, I should be able to follow up teachers to view their posts, but this will be 1 way relation. |
| Topic Request Management System | Search Topic Request | As a teacher, I want to search among the recommended topic requests. |
| Bid Topic Requests | As a teacher, I want to bid on topic requests to offer my expertise. |
| View Proposals | As a student who created a topic request, I want to view proposals from teachers who have bid on my request. |
| Shortlist Teacher | As a student with a topic request, I want to shortlist teachers based on their proposals for further consideration. |
| Close Topic Request | As a student who created a topic request, I want to close the request after selecting a teacher for the collaboration. |
| Conduct Live Classes | As a teacher, I want to conduct live classes to interact with students in real-time and deliver engaging content. |
| Create Topic Request | As a student, I want to create a topic request to seek expertise or knowledge in a specific area. |
| View My Topic Request | As a student, I want to view a list of topic requests I've created to track their status and manage them. |
| Update Topic Request | As a student who created a topic request, I want to update the details to provide more information or make changes. |
| Communicate | As a student involved in a topic request, I want to communicate with teachers to discuss details and ensure a successful collaboration. |
| Session Management System | Create Session | As a teacher, I want to create a session to plan and organize a live class for my students. |
| Update Session | As a teacher, I want to update session details to ensure accurate information is provided to participants. |
| View Drafted Session | As a teacher, I want to view a drafted session before publishing it to ensure everything is set up correctly. |
| Complete Session Details | As a teacher, I want to complete all necessary session details to provide participants with comprehensive information. |
| View Published Session | As a student, I want to view a published session to verify its availability. |
| Search Session | As a student, I want to search for sessions to find live classes or events that align with my interests. |
| Join Session | As a student, I want to join a session to actively participate in a live class. |
| End Session | As a teacher, I want to end a session after the class has concluded. |
| Start Session | As a teacher, I want to start a session to initiate the live class at the scheduled time. |
| Administrative Management System | View Revenue | As an administrator, I want to view detailed reports to gain insights into user activities and system performance. |
| View Reports | As an admin, I want to view detailed reports to gain insights into user activities and system performance. |
| Perform Crud Operation | As an administrator, I want to perform CRUD operations on Learn Lance functionalities to manage and maintain the system effectively. |

### **Event List:**

**Creating and Managing Communities:**

EL-1: Users (Students/teachers) create a community.

EL-2: LearnLance creates a community.

EL-3: Users (Students/teachers) can search communities using predefined filters.

EL-4: Users (Students/teachers) can view the community.

EL-5: Users (Students/teachers) can request to join the community.

EL-6: The creator of the community can admit requested Users (students/teachers).

EL-7: Joined Users (students/teachers) can post in the community.

EL-8: LearnLance creates a community post.

EL-9: LearnLance makes the community post visible to all joined members' feeds.

EL-10: LearnLance makes the community post visible on the community wall.

EL-11: Joined Users (students/teachers) can comment community posts.

EL-12: Joined Users (students/teachers) can like community posts.

EL-13: The creator of the community can remove joined community members.

EL-14: The creator of the community can delete a community post.

**Topic Requests and Bidding:**

EL-15: Students create a topic request.

EL-16: LearnLance creates the topic request.

EL-17: Teachers can search topic requests using predefined filters.

EL-18: LearnLance recommends the topic request to teachers.

EL-19: Teachers selects a Topic Request. and bid on the topic request.

EL-20: Teachers bids on the topic request.

EL-21: Students are notified about the teacher's proposal.

EL-22: Students select the topic request.

EL-23: Students shortlist the topic request.

EL-24: Students can communicate and negotiate on payment/duration.

EL-25: Teachers can communicate and negotiate on payment/duration.

EL-26: Students will pay the amount for the topic request.

EL-27: The admin approves the payment for the topic request.

EL-28: LearnLance automatically confirms the student and teacher.

EL-29: LearnLance redirects students to a virtual classroom.

EL-30: LearnLance redirects teachers to a virtual classroom.

**Live Classes and Virtual Classroom:**

EL-31: Teachers schedule live classes.

EL-32: Students can set up reminders for live classes.

EL-33: Teachers can set up reminders for live classes.

EL-34: LearnLance fetches time from the system.

EL-35: LearnLance notifies students of live virtual classes.

EL-36: LearnLance notifies teachers of live virtual classes.

EL-37: During live classes, participants can interact with each other in real-time.

EL-38: They can also communicate through message support during live classes.

**Course Creation and Enrollment:**

EL-39: Teachers create a course.

EL-40: Teachers upload course content.

EL-41: Teachers upload quizzes.

EL-42: Students enroll in a course.

EL-43: The admin approves the payment made by students for course enrollment.

EL-44: LearnLance notifies the student about the success of enrollment in the course.

EL-45: Students can search courses using predefined filters.

EL-56: Students can access course content.

EL-47: Students can watch videos or lectures for that course.

EL-48: Students can give quizzes assigned to them.

EL-49: LearnLance updates the progress of the student.

EL-50: LearnLance rewards students with badges on successful quiz completion.

EL-51: On successful completion of a course, LearnLance generates a certificate for the student.

**Connections and Feeds:**

EL-52: Teachers can also invite another teacher for a joint account on a course.

EL-53: The teacher accepts the request on the invited course.

EL-54: The teacher updates the course content on the invited course.

EL-55: Students can follow teachers.

EL-56: Students can connect with other students.

EL-57: Teachers can connect with students.

EL-58: Students/teachers can view their connected student/teacher posts on the feed.

EL-59: Students can view posts from the following teachers on the feed.

EL-60: Students/teachers can go directly to any profile by clicking on the profile picture icon.

EL-61: Students/teachers can search profiles using predefined filters.

EL-62: Students/teachers can message/audio call/video call someone directly on their profiles.

**Messaging and Calls:**

EL-63: The message management system manages messages.

EL-64: The message management system retrieves the chat.

EL-65: The phone camera is requested for video calls.

EL-66: The call management system manages audio calls.

EL-67: The call management system manages video calls.

**Sessions:**

EL-68: Teachers can create a session.

EL-69: Students can search sessions using some filters.

EL-70: Students can mark sessions as interesting.

EL-71: Students/teachers can set reminders on the date and time of the schedule.

EL-72: LearnLance fetches time from the system and generates notifications for teachers and students.

EL-73: The call management system manages the session.

EL-74: The call management system manages the participants of sessions.

**Administration:**

EL-75: The admin performs CRUD operations on all LearnLance functionalities.

EL-76: The admin has a dashboard to view statistics.

EL-77: The admin has a dashboard to view reports.

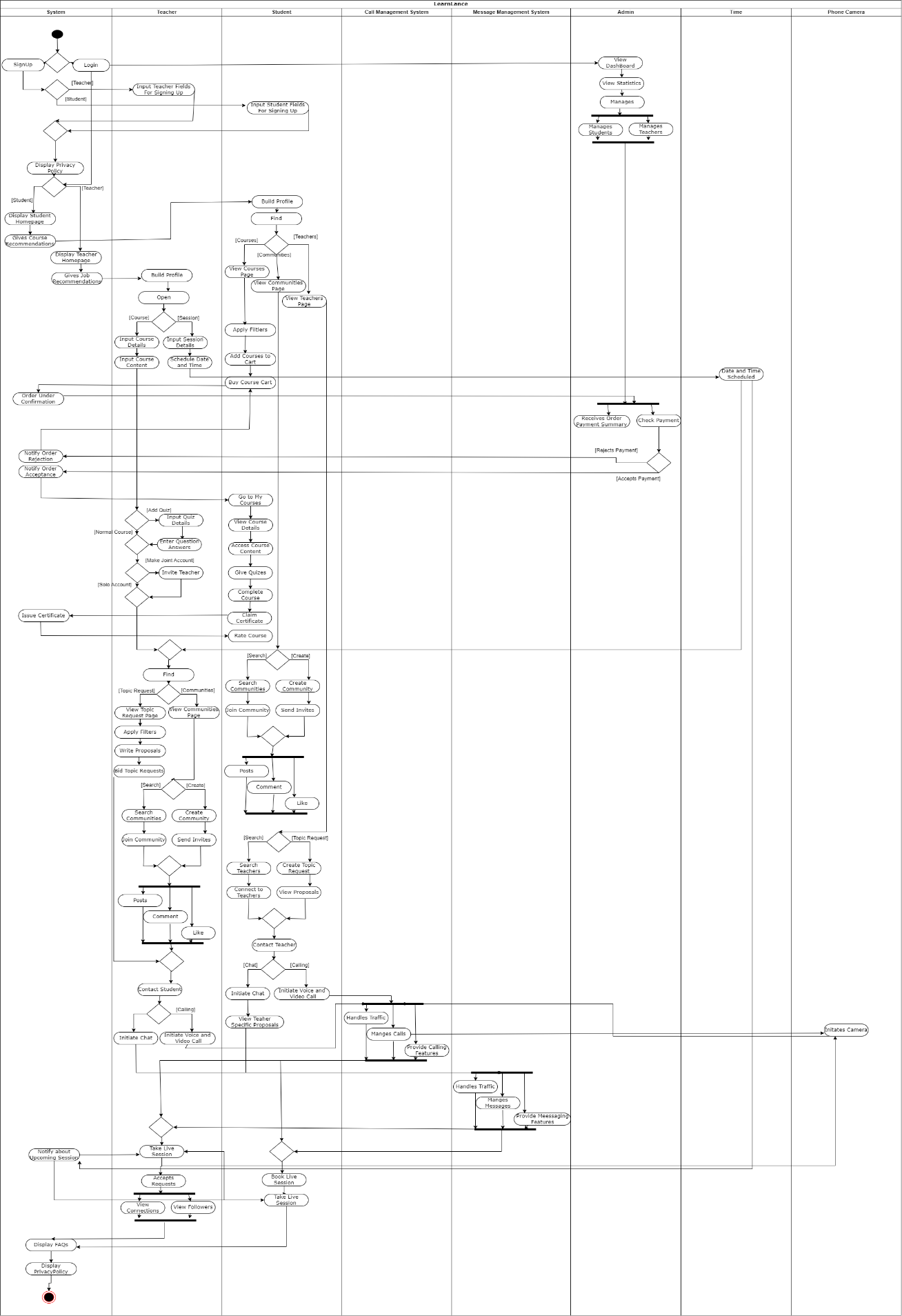
EL-78: The admin can also view the revenue generated by LearnLance for some specific time.

**Information and Policies:**

EL-79: LearnLance displays FAQs.

EL-80: LearnLance displays privacy and policy.

### **Activity Diagram:**



Please refer to this link if the image is not clear: *https://drive.google.com/file/d/1UZdC\_pzybtaFA-jAqusVoMah9o-nH5gb/view?usp=sharing*

### **State Transition Diagrams:**

**Community State Diagram**

******

Created

|  |  |  |
| --- | --- | --- |
|  | Created | Deleted |
| Created | No | Community Creator decides to delete the community |
| Deleted | No | No |

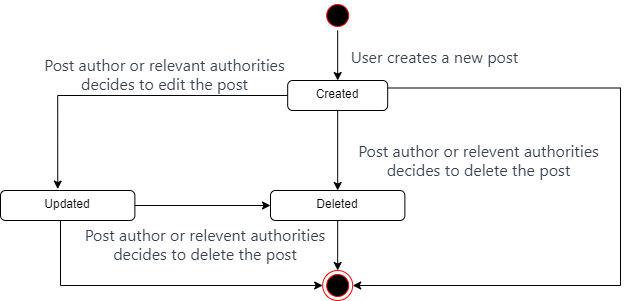
**Community Request**

A diagram of a process

Description automatically generated

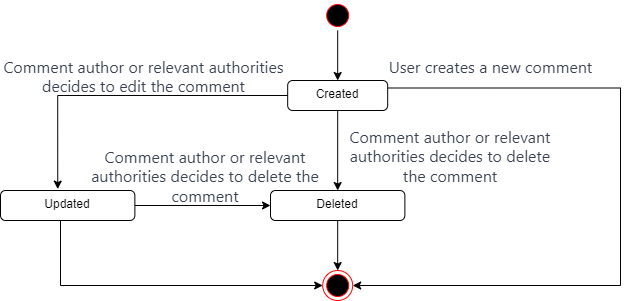
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Active | Pending | Approved | Rejected | Withdraw |
| Active | No | Request is Sent | No | No | No |
| Pending | No | No | Community Creator approve the request | Community Creator rejects the request | No |
| Approved | No | No | No | No | User left the Community |
| Rejected | No | No | No | No | No |
| Withdraw | No | No | No | No | No |

**Post State Diagram**

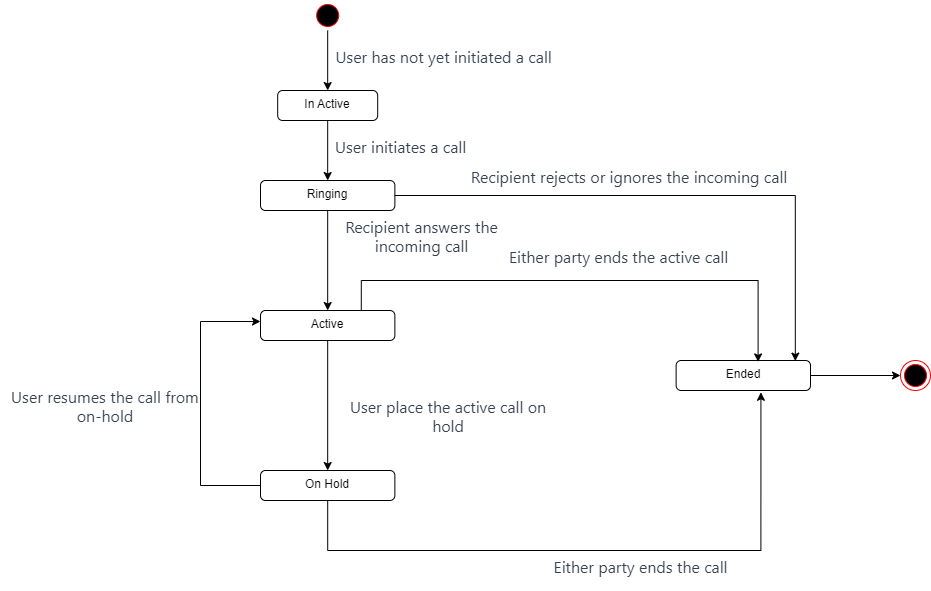
****

|  |  |  |  |
| --- | --- | --- | --- |
|  | Created | Updated | Deleted |
| Created | No | Post Author or relevant authorities decides to edit the post | Post author or relevant authorities decides to delete the post |
| Updated | No | No | Post author or relevant authorities decides to delete the post |
| Deleted | No | No | No |

**Comment State Diagram**

****

|  |  |  |  |
| --- | --- | --- | --- |
|  | Created | Updated | Deleted |
| Created | No | Comment Author or relevant authorities decides to edit the Comment | Comment Author or relevant authorities decides to delete the Comment |
| Updated | No | No | Comment Author or relevant authorities decides to delete the Comment |
| Deleted | No | No | No |

****Call State Diagram**

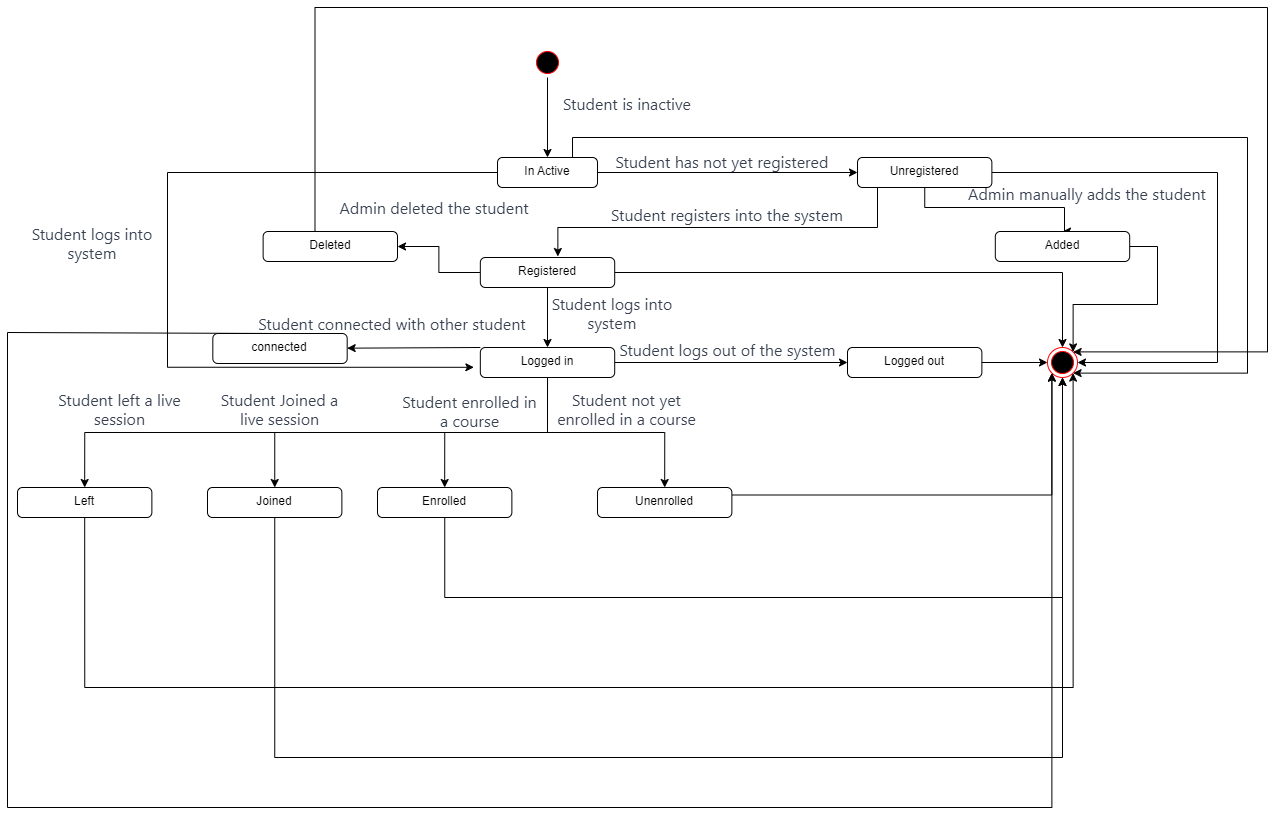
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Active | Pending | Approved | Rejected | Withdraw |
| In Active | No | Request is Sent | No | No | No |
| Ringing | No | No | Community Creator approve the request | Community Creator rejects the request | No |
| Active | No | No | No | No | User left the Community |
| Rejected | No | No | No | No | No |
| Withdraw | No | No | No | No | No |

**Teacher State Diagram**

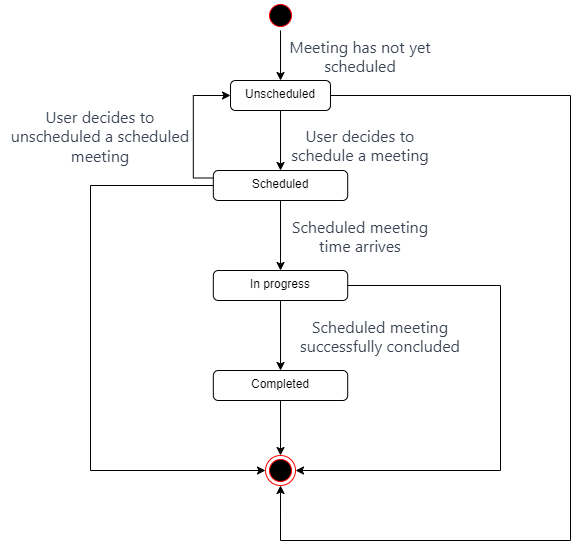
**A diagram of a company

Description automatically generated**

**Student State Diagram**

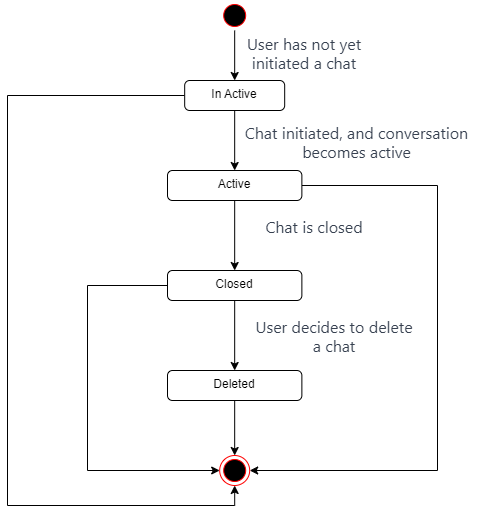
****

**Meeting State Diagram**

******

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Unscheduled | Scheduled | In Progress | Completed |
| Unscheduled | No | User decides to schedule a meeting | Scheduled meeting time arrives | No |
| Scheduled | User decides to unscheduled a scheduled meeting | No | Scheduled Meeting time arrives | No |
| In Progress | No | No | No | Scheduled meeting successful concluded |
| Completed | No | No | No | No |

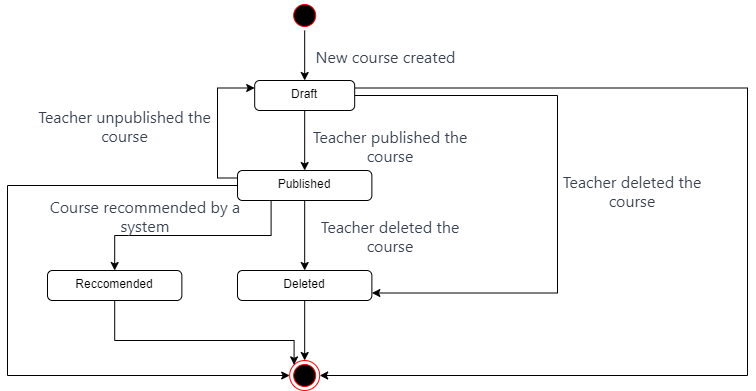
**Chat State Diagram**

******

**Order State Diagram**

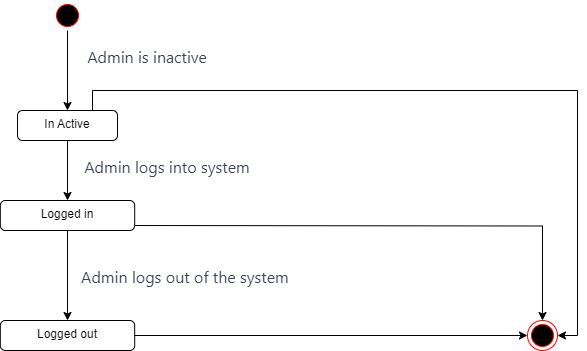
******

|  |  |  |
| --- | --- | --- |
|  | Pending | Approved |
| Pending | No | Admin manually marks the order as Paid |
| Approved | No | No |

****Course State Diagram**

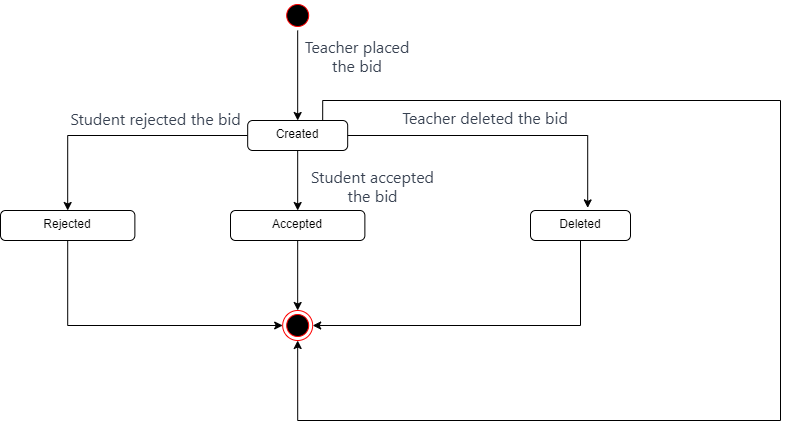
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Draft | Published | Recommended | Deleted |
| Draft | No | Teacher published the Course | No | Teacher decides to delete the Course |
| Published | No | No | Course Recommended by a System | Teacher decides to delete the Course |
| Recommended | No | No | No | No |
| Deleted | No | No | No | No |

**Admin State Diagram**

****

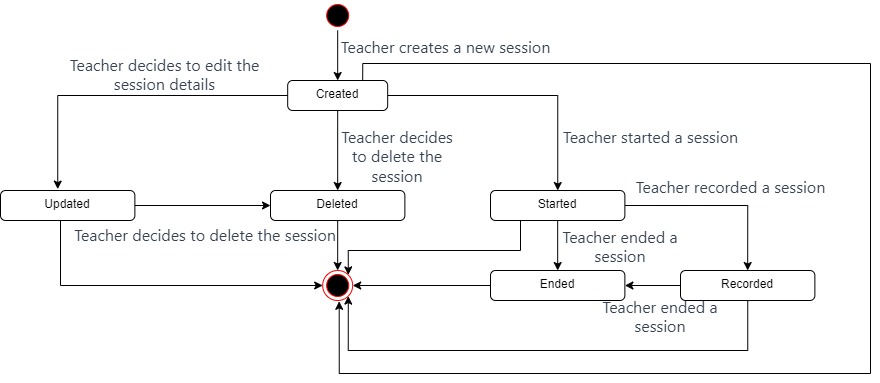
|  |  |  |  |
| --- | --- | --- | --- |
|  | In Active | Logged In | Logged Out |
| In Active | No | Admin logs into the system | No |
| Logged In | No | No | Admin logs out of the system |
| Logged Out | No | No | No |

**Bid State Diagram**

****

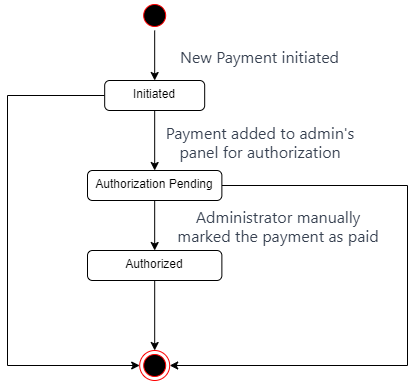
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Created | Deleted | Accepted | Rejected |
| Created | No | Teacher deleted the bid | Student accepted the bid | Student rejected the bid |
| Deleted | No | No | No | No |
| Accepted | No | No | No | No |
| Rejected | No | No | No | No |

**Session State Diagram**

****

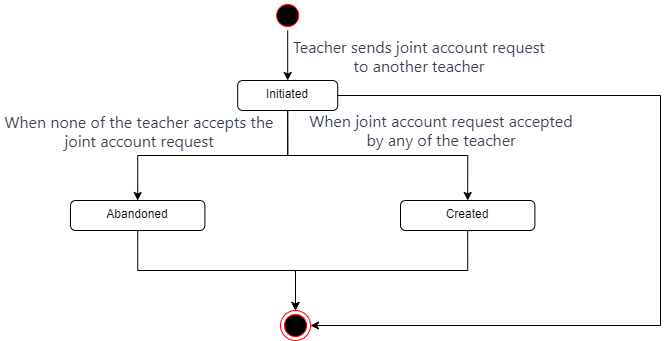
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Created | Deleted | Updated | Started | Recorded | Ended |
| Created | No | Teacher decides to Delete the session | Teacher decides to edit the session details | Teacher started a Session | No | No |
| Deleted | No | No | No | No | No | No |
| Updated | No | No | No | No | No | No |
| Started | No | No | No | No | Teacher recorded a session | Teacher ended a session |
| Recorded | No | No | No | No | No | No |
| Ended | No | No | No | No | No | No |

**Payment State Diagram**

****

|  |  |  |  |
| --- | --- | --- | --- |
|  | Initiated | Authorization Pending | Authorized |
| Initiated | No | Payment added to admin’s panel for authorization | No |
| Authorization Pending | No | No | Admin manually marks the payment as paid |
| Authorized | No | No | No |

**Joint Account State Diagram**

****

|  |  |  |  |
| --- | --- | --- | --- |
|  | Initiated | Abandoned | Created |
| Initiated | No | When teacher don’t accept the joint account request | When teacher accepts the joint account request |
| Abandoned | No | No | No |
| Created | No | No | No |

### **Crud Matrix:**

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Student | Teacher | Admin | Community | Post | Joint  Account | Session | Social Hub | Course | Call | Messages |
| Signup | C | C |  |  |  |  |  |  |  |  |  |
| Login | R | R |  |  |  |  |  |  |  |  |  |
| Logout | R | R |  |  |  |  |  |  |  |  |  |
| Update Profile | R,U,D | R,U,D |  |  |  |  |  |  |  |  |  |
| Search Courses | R |  |  |  |  |  |  |  | R |  |  |
| Enroll In Course | R |  |  |  |  |  |  |  | R,U |  |  |
| Make Course Payment | R |  | R |  |  |  |  |  | R |  |  |
| Access To Course Content | R |  |  |  |  |  |  |  | R,U |  |  |
| Complete Course | R |  |  |  |  |  |  |  | R,U |  |  |
| Give Feedback | R | R |  |  |  |  |  |  | R |  |  |
| Get Certificate | R | R |  |  |  |  |  |  | R |  |  |
| Create Course | R | R |  |  |  |  |  |  | C |  |  |
| Update Course | R | R |  |  |  |  |  |  | U |  |  |
| View Drafted Course | R | R |  |  |  |  |  |  | R |  |  |
| Complete Course Details | R | R |  |  |  |  |  |  | R,U,C |  |  |
| View Published Course | R | R |  |  |  |  |  |  | R |  |  |
| Create Community | R | R |  | C |  |  |  |  |  |  |  |
| Manage Community | R | R |  | R,U,D |  |  |  |  |  |  |  |
| Update Community | R | R |  | R,U,D |  |  |  |  |  |  |  |
| Search Communities |  | R |  | R |  |  |  |  |  |  |  |
| Leave Community | R | R |  | R |  |  |  |  |  |  |  |
| Manage Post | R | R |  | R | R,U,D |  |  |  |  |  |  |
| Join Community | R | R |  | R |  |  |  |  |  |  |  |
| Create Community Post | R | R |  | R | C |  |  |  |  |  |  |
| Comment Post | R | R |  | R | R,U |  |  |  |  |  |  |
| Like Post | R | R |  | R | R,U |  |  |  |  |  |  |
| Create Post | R | R |  | R | C |  |  | R |  |  |  |
| View Connections | R | R |  |  | R |  |  | R |  |  |  |
| Invite Connection |  | R |  |  |  | C,R |  |  |  |  |  |
| View Invitation |  | R |  |  |  | R |  |  |  |  |  |
| Withdraw Invitation |  | R |  |  |  | D |  |  |  |  |  |
| Manage Invitation Request |  | R |  |  |  | R,U,D |  |  |  |  |  |
| View Chats | R | R |  |  |  |  |  | R |  |  | R |
| Initiate Audio Calls | R | R |  |  |  |  |  | R |  | C,R,U |  |
| Send Message | R | R |  |  |  |  |  | R |  |  | C,R,U,D |
| Initiate Video Call | R | R |  |  |  |  |  | R |  | C,R,U |  |
| End Video Call | R | R |  |  |  |  |  | R |  | U |  |
| End Audio Call | R | R |  |  |  |  |  | R |  | U |  |
| View My Followers | R |  |  |  |  |  |  | R |  |  |  |
| View My Connections | R | R |  |  | R |  |  | R |  |  |  |
| Withdraw Connection Requests | R | R |  |  | R |  |  | U,D |  |  |  |
| Send Connection Requests | R | R |  |  |  |  |  | C |  |  |  |
| Manage Connection Requests | R | R |  |  | R |  |  | R,U,D |  |  |  |
| Follow Teacher | R |  |  |  |  |  |  | C |  |  |  |
| Search Topic Request | R |  |  |  |  |  |  |  |  |  |  |
| Bid Topic Requests | R | R |  |  |  |  |  |  |  |  |  |
| View Proposals |  | R |  |  |  |  |  |  |  |  |  |
| Shortlist Teacher |  | R |  |  |  |  |  |  |  |  |  |
| Close Topic Request | R | R |  |  |  |  |  |  |  |  |  |
| Conduct Live Classes | R | R |  |  |  |  |  |  |  | C,R,U |  |
| Create Topic Request | R | R |  |  |  |  |  |  |  |  |  |
| View My Topic Request | R | R |  |  |  |  |  |  |  |  |  |
| Update Topic Request | R | R |  |  |  |  |  |  |  |  |  |
| Communicate | R | R |  |  |  |  |  | R |  | C,R,U | C,R,U,D |
| Create Session |  | R |  |  |  |  | C |  |  |  |  |
| Update Session |  | R |  |  |  |  | R,U |  |  |  |  |
| View Drafted Session |  | R |  |  |  |  | R,U |  |  |  |  |
| Complete Session Details |  | R |  |  |  |  | R,U |  |  |  |  |
| View Published Session |  | R |  |  |  |  | R |  |  |  |  |
| Search Session | R | R |  |  |  |  | R |  |  |  |  |
| Join Session | R | R |  |  |  |  | R,U |  |  | R,U |  |
| End Session |  | R |  |  |  |  | R,U |  |  | U |  |
| Start Session |  | R |  |  |  |  | R,U |  |  | C |  |
| View Revenue |  |  | R |  |  |  |  |  |  |  |  |
| View Reports | R | R | R |  |  |  |  |  |  |  |  |
| Perform Crud Operation | C,R,U,D | C,R,U,D | R | R,D | R,D |  |  |  |  |  |  |

### **Data Structures:**

**Orders:**

A close-up of a code

Description automatically generated

**Courses:**

****

**Lessons:**

A computer code with text

Description automatically generated with medium confidence

**Questions:**

A computer screen shot of a question

Description automatically generated

**Question Answers:**

A white rectangular frame with black text

Description automatically generated

**Teacher Bid:**

A close-up of a computer code

Description automatically generated

**Bid Topic:**

A close-up of a code

Description automatically generated

**Learning Posts:**

A screen shot of a computer program

Description automatically generated

**Student Profile:**

A close-up of a computer screen

Description automatically generated

**Joint Account Teachers:**

A computer screen shot of a computer screen

Description automatically generated

**Live Sessions:**

A screen shot of a computer code

Description automatically generated

**Teacher Profile:**

A screenshot of a computer

Description automatically generated

**Revenue:**

A white background with black text

Description automatically generated

**Scheduled Meetings:**

A white rectangular object with black text

Description automatically generated

**User:**

A computer screen shot of a computer code

Description automatically generated

**Posts:**

A white rectangle with black text

Description automatically generated

**Comments:**

A white rectangle with black text

Description automatically generated

**Social Hub:**

A computer screen shot of a computer program

Description automatically generated

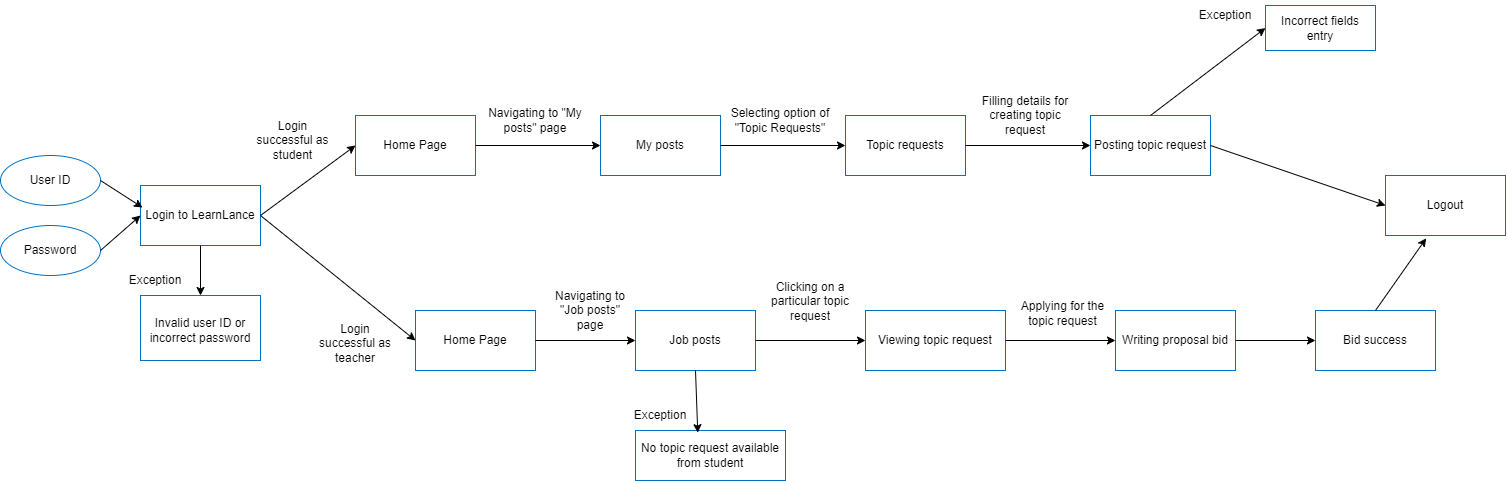
**Community:**

A computer code with text

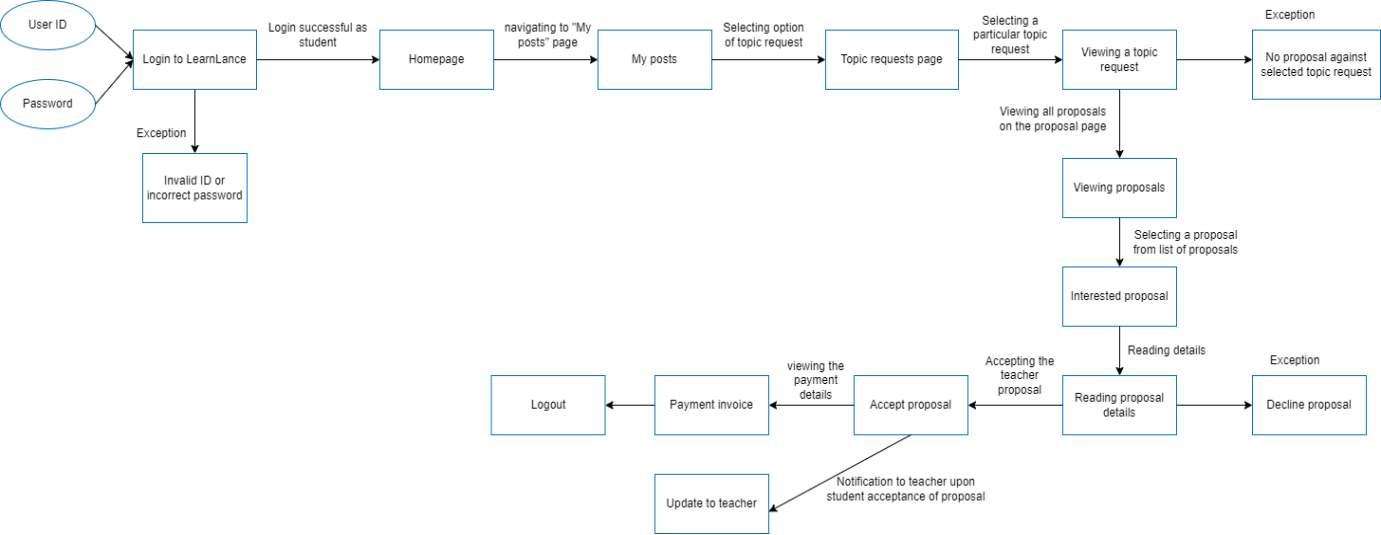
Description automatically generated

### **Scenarios:**

**Topic Request Scenarios:**

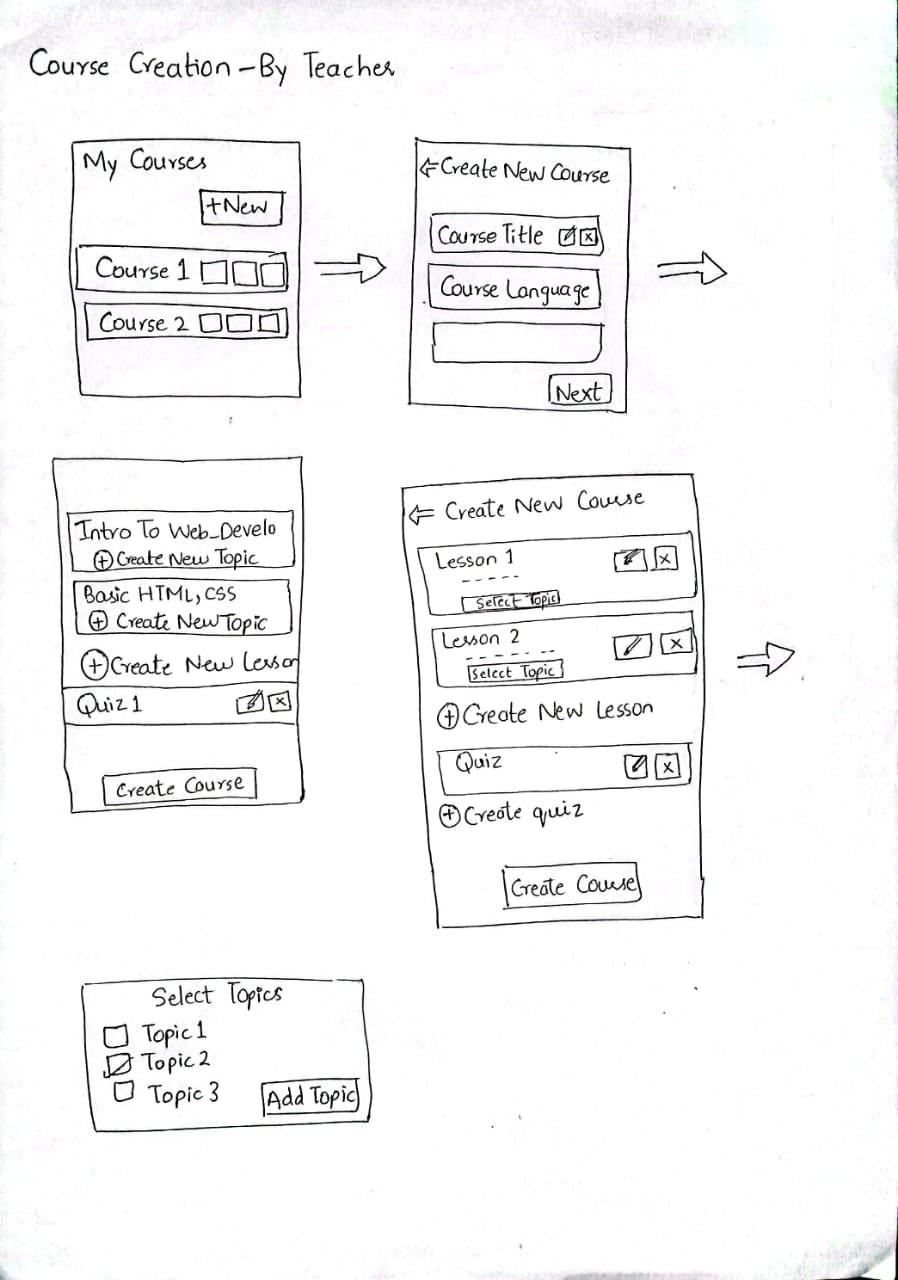


**Proposal Acceptance Scenarios:**



### **Story Boards:**

**Course Creation by Teacher:**



**Course Viewing by Students:**

A diagram of a computer program

Description automatically generated with medium confidence

### **User Personas:**

The user personas for describing each user class interactions with the system are described below:

**User Persona of Student:**

Insha Samnani, a 22-year-old undergraduate student majoring in Computer Science. Insha is driven by a strong motivation to excel in academic studies while concurrently acquiring practical skills essential for a future career in the tech industry. Always on the lookout for opportunities that align with academic goals and career aspirations, Insha will actively engage with the LearnLance platform to fulfill educational and freelancing needs.

Insha's primary goals include achieving educational excellence, developing new skills, expanding professional networks, and exploring freelancing opportunities. To realize these goals, Insha relies on LearnLance for seamless access to a diverse range of courses, efficient communication with teachers and peers through integrated tools like chat and video calls, and the ability to manage a profile that showcases academic achievements and interests.

One of Insha's key needs is a user-friendly platform that simplifies course navigation, ensuring easy access to relevant content. Efficient communication tools within the platform enable Insha to collaborate with peers, seek guidance from teachers, and participate in community forums. Insha appreciates the personalized LearnLance's recommendation engine, which tailors course suggestions based on academic performance, interests, and career objectives.

**User Persona of Teacher:**

Dr. Muhammad Rafi, an experienced professional in the field of artificial intelligence and machine learning with a passion for sharing knowledge and fostering the growth of aspiring learners. Dr. Rafi uses LearnLance to bring a wealth of expertise and actively engages with the platform to leverage its educational services and job opportunities.

As a teacher at LearnLance, Dr. Rafi places high importance on efficient course management tools to showcase his skills and credentials effectively. The platform's user-friendly interface allows him to effortlessly create and manage a detailed teacher profile, highlighting his experience, certifications, completed projects, and valuable reviews from both students and fellow teachers. Dr. Rafi finds immense satisfaction in building a positive online reputation within the LearnLance community.

Dr. Rafi’s uses LearnLance to effectively manage communication and networking. LearnLance's integrated social hub enables him to connect with like-minded peers, expanding his professional network and fostering collaborative opportunities. The bidding system for job opportunities is a crucial feature for him, as it provides a streamlined and transparent process for connecting with students seeking his expertise. Dr. Rafi appreciates the platform's commitment to personalized recommendations, and he uses this feature of LearnLance offering him job opportunities aligned with his expertise and preferences.

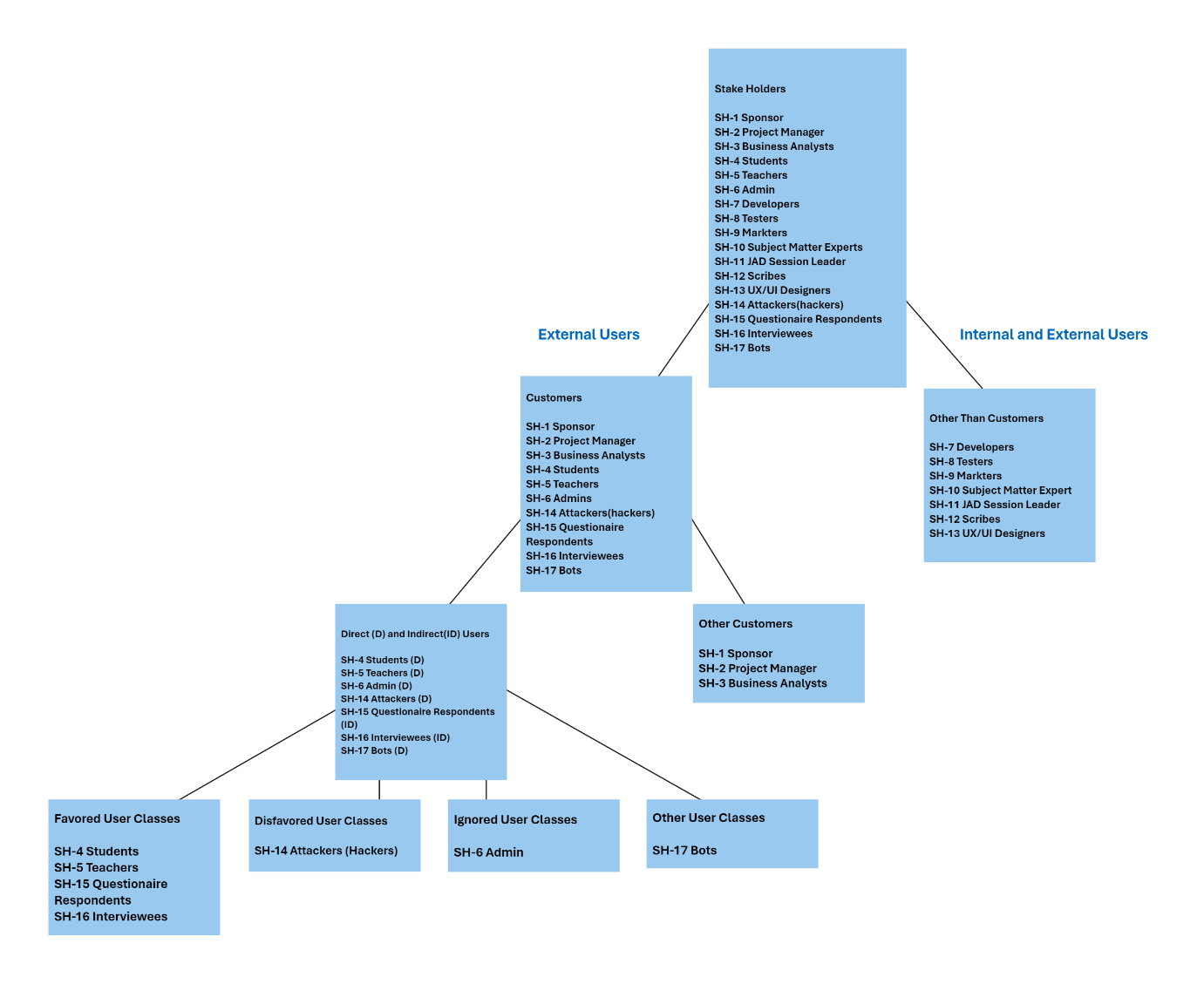
**User Persona of Admin:**

Ismail Ahmed Ansari, a seasoned professional with a background in system administration and a keen eye for maintaining order and efficiency. Ismail interacts with LearnLance as an administrator, overseeing the platform's operations and ensuring a smooth experience for both students and teachers.

In his capacity as an admin, Ismail relies on LearnLance's robust administrative tools to perform crucial control operations. Ismail uses the LearnLance platform for managing user profiles, overseeing monthly revenue, issuing certifications, approving payments, and maintaining a high level of access and control over the platform. Ismail values the platform's user-friendly interface, which streamlines his administrative tasks and ensures efficient navigation.

Ismail’s key needs revolve around maintaining the integrity and security of the LearnLance platform. The administrative tools provided by LearnLance empower him to execute control operations swiftly, whether it's adding or removing users, monitoring monthly revenue, or ensuring that payment approvals align with the platform's guidelines.

### **Stakeholders Classification:**



Please refer to this link if the image is not clear: *https://drive.google.com/file/d/1uslKNHVisFNb2tc76A\_N77egll\_crocR/view?usp=sharing*

## **Appendix 3: Requirement Negotiation and Prioritization**

### **Requirement Negotiation:**

R1: LearnLance is compatible with all Android devices, including mobile phones and tablets, with an Android version greater than Android 8 Oreo.

R2: LearnLance shall permit user access from the corporate intranet; from a VPN internet connection; and by Android 8 Oreo smartphones and tablets.

R3: LearnLance will display the most useful widgets on the admin profile for ease of understanding LearnLance.

R4: Users can directly access a profile by clicking on its profile picture or searching for the profile, allowing for easy access.

R5: LearnLance will display the most useful widgets on the admin profile and drawer icon for ease of understanding LearnLance.

R6: When installing/ bundling an upgraded version of LearnLance, all recommendations in the users’ profile will be retained and converted to the new version’s data format if needed.

R7: LearnLance's recommendations for courses will be retained until the student modifies their interests or changes their interaction with the system.

R8: LearnLance's recommendations for topic requests will persist until the teacher modifies the skills required for the specific topic request.

R9: LearnLance shall operate on Apache HTTP server.

R10: The system can offer personalized job recommendations.

R11: The system can offer personalized course recommendations.

R12: LearnLance's backend and frontend should seamlessly communicate with each other through the HTTP/HTTPS protocol.

R13: Teachers can browse the list of communities.

R14: If a student enrolls in a course, then the course creator and student will be notified.

R15: If a student enrolls in a course, then the course creator will be notified.

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| REQUIREMENTS | R1 | R2 | R3 | R4 | R5 | R6 | R7 | R8 | R9 | R10 | R11 | R12 | R13 | R14 | R15 |
| R1 | 0 | 1000 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R2 | 1000 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R3 | 0 | 0 | 0 | 0 | 1000 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R4 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R5 | 0 | 0 | 1000 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R6 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R7 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R8 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R9 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1000 | 0 | 0 | 0 |
| R10 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R11 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| R12 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1000 | 0 | 0 | 0 | 0 | 0 | 0 |
| R13 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |  |
| R14 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1000 |
| R15 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1000 | 0 |

### **Requirement Prioritization:**

#### **User Requirement Prioritization:**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Relative Weights | 2 | 1 |  |  | 1 |  | 0.5 |  |  |
| Feature | Relative Benefit | Relative Penalty | Total Value | Value % | Relative Cost | Cost % | Relative Risk | Risk % | Priority |
| Students can create their profiles. | 9 | 9 | 27 | 1.436170213 | 7 | 1.559020045 | 3 | 0.940438871 | 0.707738158 |
| Teachers can create their profiles. | 9 | 9 | 27 | 1.436170213 | 7 | 1.559020045 | 3 | 0.940438871 | 0.707738158 |
| Students can edit their profiles. | 9 | 9 | 27 | 1.436170213 | 5 | 1.113585746 | 3 | 0.940438871 | 0.906784641 |
| Teachers can edit their profiles. | 9 | 9 | 27 | 1.436170213 | 5 | 1.113585746 | 3 | 0.940438871 | 0.906784641 |
| Students can view their own profiles. | 8 | 8 | 24 | 1.276595745 | 4 | 0.890868597 | 2 | 0.626959248 | 1.059988899 |
| Teachers can view their own profiles. | 8 | 8 | 24 | 1.276595745 | 4 | 0.890868597 | 2 | 0.626959248 | 1.059988899 |
| Students can view the profiles of other students. | 7 | 3 | 17 | 0.904255319 | 4 | 0.890868597 | 2 | 0.626959248 | 0.75082547 |
| Students can view the profiles of other teachers. | 9 | 9 | 27 | 1.436170213 | 4 | 0.890868597 | 2 | 0.626959248 | 1.192487512 |
| Teachers can view the profiles of other students. | 9 | 9 | 27 | 1.436170213 | 4 | 0.890868597 | 2 | 0.626959248 | 1.192487512 |
| Teachers can view the profiles of other teachers. | 7 | 3 | 17 | 0.904255319 | 4 | 0.890868597 | 2 | 0.626959248 | 0.75082547 |
| Students can send connection requests to other students. | 7 | 7 | 21 | 1.117021277 | 4 | 0.890868597 | 3 | 0.940438871 | 0.820682608 |
| Students can follow teachers. | 9 | 9 | 27 | 1.436170213 | 4 | 0.890868597 | 3 | 0.940438871 | 1.055163353 |
| Teachers can send connection requests to other teachers. | 7 | 7 | 21 | 1.117021277 | 4 | 0.890868597 | 3 | 0.940438871 | 0.820682608 |
| Students can view their connections. | 7 | 7 | 21 | 1.117021277 | 3 | 0.668151448 | 2 | 0.626959248 | 1.137923716 |
| Teachers can view their connections. | 7 | 7 | 21 | 1.117021277 | 3 | 0.668151448 | 2 | 0.626959248 | 1.137923716 |
| Teachers can view their followers. | 7 | 7 | 21 | 1.117021277 | 3 | 0.668151448 | 2 | 0.626959248 | 1.137923716 |
| Teachers can create courses. | 9 | 9 | 27 | 1.436170213 | 8 | 1.781737194 | 6 | 1.880877743 | 0.527581677 |
| Teachers can draft courses. | 4 | 3 | 11 | 0.585106383 | 3 | 0.668151448 | 2 | 0.626959248 | 0.59605528 |
| Teachers can publish courses. | 9 | 9 | 27 | 1.436170213 | 3 | 0.668151448 | 2 | 0.626959248 | 1.463044778 |
| Teachers can view the courses they have posted. | 7 | 7 | 21 | 1.117021277 | 3 | 0.668151448 | 2 | 0.626959248 | 1.137923716 |
| The system can offer personalized course recommendations. | 9 | 9 | 27 | 1.436170213 | 9 | 2.004454343 | 8 | 2.507836991 | 0.440763008 |
| Students can browse courses. | 7 | 7 | 21 | 1.117021277 | 3 | 0.668151448 | 2 | 0.626959248 | 1.137923716 |
| Students can enroll in courses. | 9 | 9 | 27 | 1.436170213 | 4 | 0.890868597 | 3 | 0.940438871 | 1.055163353 |
| System can provide certifications for completing courses. | 7 | 7 | 21 | 1.117021277 | 4 | 0.890868597 | 3 | 0.940438871 | 0.820682608 |
| System can provide badges for completing a certain number of quizzes. | 4 | 2 | 10 | 0.531914894 | 4 | 0.890868597 | 3 | 0.940438871 | 0.390801242 |
| Achievements are made visible on the student’s profile. | 3 | 2 | 8 | 0.425531915 | 1 | 0.222717149 | 1 | 0.313479624 | 1.121423398 |
| Certificates are made visible on the student’s profile. | 3 | 2 | 8 | 0.425531915 | 1 | 0.222717149 | 1 | 0.313479624 | 1.121423398 |
| Students can post their thoughts and achievements to share with their connections. | 7 | 7 | 21 | 1.117021277 | 7 | 1.559020045 | 3 | 0.940438871 | 0.550463012 |
| Teachers can post their thoughts and achievements to share with their connections. | 7 | 7 | 21 | 1.117021277 | 7 | 1.559020045 | 3 | 0.940438871 | 0.550463012 |
| Students can like the posts of their connections. | 3 | 2 | 8 | 0.425531915 | 3 | 0.668151448 | 3 | 0.940438871 | 0.373807799 |
| Students can like the posts of their followers. | 3 | 2 | 8 | 0.425531915 | 3 | 0.668151448 | 3 | 0.940438871 | 0.373807799 |
| Students can comment on the posts of their connections. | 3 | 2 | 8 | 0.425531915 | 3 | 0.668151448 | 3 | 0.940438871 | 0.373807799 |
| Students can comment on the posts of their followers. | 3 | 2 | 8 | 0.425531915 | 3 | 0.668151448 | 3 | 0.940438871 | 0.373807799 |
| Teachers can like the posts of their connections. | 3 | 2 | 8 | 0.425531915 | 3 | 0.668151448 | 3 | 0.940438871 | 0.373807799 |
| Teachers can comment on the posts of their connections. | 3 | 2 | 8 | 0.425531915 | 3 | 0.668151448 | 3 | 0.940438871 | 0.373807799 |
| Students can post topic requests. | 9 | 9 | 27 | 1.436170213 | 7 | 1.559020045 | 3 | 0.940438871 | 0.707738158 |
| The system can offer personalized job recommendations. | 9 | 9 | 27 | 1.436170213 | 9 | 2.004454343 | 8 | 2.507836991 | 0.440763008 |
| Teachers can browse topic requests. | 7 | 7 | 21 | 1.117021277 | 3 | 0.668151448 | 2 | 0.626959248 | 1.137923716 |
| Teachers can create proposals for topic requests. | 9 | 9 | 27 | 1.436170213 | 6 | 1.336302895 | 3 | 0.940438871 | 0.794991674 |
| Teachers can bid on topic requests from students seeking personalized learning sessions. | 9 | 9 | 27 | 1.436170213 | 1 | 0.222717149 | 1 | 0.313479624 | 3.78480397 |
| Teachers can view proposals they have initiated. | 7 | 3 | 17 | 0.904255319 | 4 | 0.890868597 | 2 | 0.626959248 | 0.75082547 |
| Students can review teacher bids. | 9 | 9 | 27 | 1.436170213 | 4 | 0.890868597 | 2 | 0.626959248 | 1.192487512 |
| Students can select the preferred teacher for personalized learning. | 9 | 9 | 27 | 1.436170213 | 2 | 0.445434298 | 2 | 0.626959248 | 1.892401985 |
| Students can end the contract after fulfilling the requirements of a topic request. | 9 | 9 | 27 | 1.436170213 | 2 | 0.445434298 | 1 | 0.313479624 | 2.384975023 |
| Teachers can end the contract after fulfilling the requirements of a topic request. | 9 | 9 | 27 | 1.436170213 | 2 | 0.445434298 | 1 | 0.313479624 | 2.384975023 |
| Students can view their selected items in the cart. | 8 | 8 | 24 | 1.276595745 | 7 | 1.559020045 | 3 | 0.940438871 | 0.629100585 |
| Administrators can approve payments made by students for courses through their individual accounts. | 9 | 9 | 27 | 1.436170213 | 2 | 0.445434298 | 2 | 0.626959248 | 1.892401985 |
| Administrators can approve payments made by students for topic requests through their individual accounts. | 9 | 9 | 27 | 1.436170213 | 2 | 0.445434298 | 2 | 0.626959248 | 1.892401985 |
| Teachers can schedule live video conferencing sessions. | 8 | 8 | 24 | 1.276595745 | 8 | 1.781737194 | 8 | 2.507836991 | 0.420533774 |
| Teachers can schedule interactive webinars. | 8 | 8 | 24 | 1.276595745 | 8 | 1.781737194 | 8 | 2.507836991 | 0.420533774 |
| Teachers can initiate meetings. | 8 | 8 | 24 | 1.276595745 | 8 | 1.781737194 | 7 | 2.194357367 | 0.443429332 |
| Teachers can end meetings. | 8 | 8 | 24 | 1.276595745 | 8 | 1.781737194 | 7 | 2.194357367 | 0.443429332 |
| Students can browse upcoming sessions. | 7 | 7 | 21 | 1.117021277 | 3 | 0.668151448 | 2 | 0.626959248 | 1.137923716 |
| Teachers can schedule reminders on the calendar. | 4 | 4 | 12 | 0.638297872 | 6 | 1.336302895 | 4 | 1.253918495 | 0.325121062 |
| Teachers can delete reminders on the calendar. | 4 | 4 | 12 | 0.638297872 | 6 | 1.336302895 | 4 | 1.253918495 | 0.325121062 |
| Students can schedule reminders on the calendar. | 4 | 4 | 12 | 0.638297872 | 6 | 1.336302895 | 4 | 1.253918495 | 0.325121062 |
| Students can delete reminders on the calendar. | 4 | 4 | 12 | 0.638297872 | 6 | 1.336302895 | 4 | 1.253918495 | 0.325121062 |
| Students can participate in live video conferencing sessions. | 8 | 8 | 24 | 1.276595745 | 8 | 1.781737194 | 8 | 2.507836991 | 0.420533774 |
| Students can participate in interactive webinars. | 8 | 8 | 24 | 1.276595745 | 8 | 1.781737194 | 8 | 2.507836991 | 0.420533774 |
| Students can browse the list of communities. | 7 | 7 | 21 | 1.117021277 | 3 | 0.668151448 | 2 | 0.626959248 | 1.137923716 |
| Teachers can browse the list of communities. | 7 | 7 | 21 | 1.117021277 | 3 | 0.668151448 | 2 | 0.626959248 | 1.137923716 |
| Students can create community forums. | 8 | 7 | 23 | 1.223404255 | 7 | 1.559020045 | 3 | 0.940438871 | 0.602888061 |
| Teachers can create community forums. | 8 | 7 | 23 | 1.223404255 | 7 | 1.559020045 | 3 | 0.940438871 | 0.602888061 |
| Students can send requests to join a community. | 8 | 7 | 23 | 1.223404255 | 4 | 0.890868597 | 2 | 0.626959248 | 1.015822695 |
| Teachers can send requests to join a community. | 8 | 7 | 23 | 1.223404255 | 4 | 0.890868597 | 2 | 0.626959248 | 1.015822695 |
| Students can invite their connections to the community they have joined. | 1 | 1 | 3 | 0.159574468 | 3 | 0.668151448 | 2 | 0.626959248 | 0.162560531 |
| Teachers can invite their connections to the community they have joined. | 1 | 1 | 3 | 0.159574468 | 3 | 0.668151448 | 2 | 0.626959248 | 0.162560531 |
| The creator of the community can accept requests from students and teachers to join. | 8 | 8 | 24 | 1.276595745 | 3 | 0.668151448 | 2 | 0.626959248 | 1.300484247 |
| The creator of the community can decline requests from students and teachers to join. | 8 | 8 | 24 | 1.276595745 | 3 | 0.668151448 | 2 | 0.626959248 | 1.300484247 |
| Joined participants of the community can create posts in the community. | 7 | 7 | 21 | 1.117021277 | 7 | 1.559020045 | 3 | 0.940438871 | 0.550463012 |
| Students can message other students. | 3 | 2 | 8 | 0.425531915 | 8 | 1.781737194 | 8 | 2.507836991 | 0.140177925 |
| Students can message other teachers. | 9 | 9 | 27 | 1.436170213 | 8 | 1.781737194 | 8 | 2.507836991 | 0.473100496 |
| Teachers can message other teachers | 9 | 9 | 27 | 1.436170213 | 8 | 1.781737194 | 8 | 2.507836991 | 0.473100496 |
| Teachers can message other students. | 9 | 9 | 27 | 1.436170213 | 8 | 1.781737194 | 8 | 2.507836991 | 0.473100496 |
| Students can call other students | 3 | 2 | 8 | 0.425531915 | 8 | 1.781737194 | 8 | 2.507836991 | 0.140177925 |
| Students can call other teachers | 9 | 9 | 27 | 1.436170213 | 8 | 1.781737194 | 8 | 2.507836991 | 0.473100496 |
| Teachers can call other students | 9 | 9 | 27 | 1.436170213 | 8 | 1.781737194 | 8 | 2.507836991 | 0.473100496 |
| Teachers can call other teachers | 3 | 2 | 8 | 0.425531915 | 8 | 1.781737194 | 8 | 2.507836991 | 0.140177925 |
| Students can rate courses | 8 | 8 | 24 | 1.276595745 | 5 | 1.113585746 | 2 | 0.626959248 | 0.894560103 |
| Students can provide feedback on courses | 1 | 1 | 3 | 0.159574468 | 5 | 1.113585746 | 2 | 0.626959248 | 0.111820013 |
| Teachers can rate students upon completion of topic requests. | 8 | 8 | 24 | 1.276595745 | 5 | 1.113585746 | 2 | 0.626959248 | 0.894560103 |
| Teachers can send joint account requests to other teachers to collaborate in teaching a course together. | 9 | 9 | 27 | 1.436170213 | 3 | 0.668151448 | 2 | 0.626959248 | 1.463044778 |
| Teachers can view the joint account requests they have received. | 8 | 8 | 24 | 1.276595745 | 3 | 0.668151448 | 2 | 0.626959248 | 1.300484247 |
| Teachers can accept joint account requests from other teachers to collaborate in teaching a course together. | 8 | 8 | 24 | 1.276595745 | 3 | 0.668151448 | 2 | 0.626959248 | 1.300484247 |
| Teachers can rate other teachers upon completion of a joint account for courses. | 1 | 1 | 3 | 0.159574468 | 5 | 1.113585746 | 2 | 0.626959248 | 0.111820013 |
| Teachers can provide feedback to other teachers upon completion of a joint account for courses. | 1 | 1 | 3 | 0.159574468 | 5 | 1.113585746 | 2 | 0.626959248 | 0.111820013 |
| Administrators can delete any student from LearnLance. | 9 | 9 | 27 | 1.436170213 | 5 | 1.113585746 | 2 | 0.626959248 | 1.006380116 |
| Administrators can delete any teacher from LearnLance. | 9 | 9 | 27 | 1.436170213 | 4 | 0.890868597 | 2 | 0.626959248 | 1.192487512 |
| Administrators can view monthly revenue generation. | 7 | 7 | 21 | 1.117021277 | 5 | 1.113585746 | 6 | 1.880877743 | 0.543820783 |
| Administrators can view the number of unique students on LearnLance per month. | 3 | 3 | 9 | 0.478723404 | 4 | 0.890868597 | 2 | 0.626959248 | 0.397495837 |
| Administrators can view the number of unique teachers on LearnLance per month | 3 | 3 | 9 | 0.478723404 | 4 | 0.890868597 | 2 | 0.626959248 | 0.397495837 |
| Students can view the LearnLance privacy and policy. | 7 | 3 | 17 | 0.904255319 | 3 | 0.668151448 | 2 | 0.626959248 | 0.921176342 |
| Teachers can view the LearnLance privacy and policy. | 7 | 3 | 17 | 0.904255319 | 3 | 0.668151448 | 2 | 0.626959248 | 0.921176342 |
| Teachers can view the LearnLance FAQs. | 7 | 3 | 17 | 0.904255319 | 3 | 0.668151448 | 2 | 0.626959248 | 0.921176342 |
| Students can view the LearnLance FAQs. | 7 | 3 | 17 | 0.904255319 | 3 | 0.668151448 | 2 | 0.626959248 | 0.921176342 |
| Totals | 642 | 596 | 1880 | 100 | 449 | 100 | 319 | 100 |  |

#### **Functional Requirement Prioritization:**

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Relative Weights | 2 | 1 |  |  | 1 |  | 0.5 |  |  |
| Feature | Relative Benefit | Relative Penalty | Total Value | Value % | Relative Cost | Cost % | Relative Risk | Risk % | Priority |
| Students have the option to enrich their profiles by providing country | 9 | 9 | 27 | 1.436170213 | 6 | 1.336302895 | 2 | 0.626959248 | 0.87052093 |
| Students have the option to enrich their profiles by providing profile pictures | 9 | 9 | 27 | 1.436170213 | 6 | 1.336302895 | 2 | 0.626959248 | 0.87052093 |
| Students have the option to enrich their profiles by providing language | 9 | 9 | 27 | 1.436170213 | 6 | 1.336302895 | 2 | 0.626959248 | 0.87052093 |
| Students have the option to enrich their profiles by providing interests | 9 | 9 | 27 | 1.436170213 | 6 | 1.336302895 | 2 | 0.626959248 | 0.87052093 |
| Students have the option to enrich their profiles by providing education | 9 | 9 | 27 | 1.436170213 | 6 | 1.336302895 | 2 | 0.626959248 | 0.87052093 |
| Students have the option to enrich their profiles by providing bio information | 9 | 9 | 27 | 1.436170213 | 6 | 1.336302895 | 2 | 0.626959248 | 0.87052093 |
| Teachers have the option to enrich their profiles by providing profile pictures | 9 | 9 | 27 | 1.436170213 | 6 | 1.336302895 | 2 | 0.626959248 | 0.87052093 |
| Teachers have the option to enrich their profiles by providing language | 9 | 9 | 27 | 1.436170213 | 6 | 1.336302895 | 2 | 0.626959248 | 0.87052093 |
| Teachers have the option to enrich their profiles by providing education | 9 | 9 | 27 | 1.436170213 | 6 | 1.336302895 | 2 | 0.626959248 | 0.87052093 |
| Teachers have the option to enrich their profiles by providing experience | 9 | 9 | 27 | 1.436170213 | 6 | 1.336302895 | 2 | 0.626959248 | 0.87052093 |
| Teachers have the option to enrich their profiles by providing honors and awards | 9 | 9 | 27 | 1.436170213 | 6 | 1.336302895 | 2 | 0.626959248 | 0.87052093 |
| Teachers have the option to enrich their profiles by providing projects | 9 | 9 | 27 | 1.436170213 | 6 | 1.336302895 | 2 | 0.626959248 | 0.87052093 |
| Teachers have the option to enrich their profiles by providing certificates | 9 | 9 | 27 | 1.436170213 | 6 | 1.336302895 | 2 | 0.626959248 | 0.87052093 |
| Teachers have the option to enrich their profiles by providing skills | 9 | 9 | 27 | 1.436170213 | 6 | 1.336302895 | 2 | 0.626959248 | 0.87052093 |
| Teachers can include duration in their created courses | 4 | 4 | 12 | 0.638297872 | 3 | 0.668151448 | 1 | 0.313479624 | 0.773796382 |
| Teachers can include objectives in their created courses | 4 | 4 | 12 | 0.638297872 | 3 | 0.668151448 | 1 | 0.313479624 | 0.773796382 |
| Teachers can include descriptions in their created courses | 4 | 4 | 12 | 0.638297872 | 3 | 0.668151448 | 1 | 0.313479624 | 0.773796382 |
| Teachers can include videos in their created courses | 9 | 9 | 27 | 1.436170213 | 8 | 1.781737194 | 3 | 0.940438871 | 0.637743282 |
| Teachers can include quizzes in their created courses | 8 | 7 | 23 | 1.223404255 | 6 | 1.336302895 | 3 | 0.940438871 | 0.67721513 |
| Teachers can include sessions in their created courses | 4 | 4 | 12 | 0.638297872 | 3 | 0.668151448 | 1 | 0.313479624 | 0.773796382 |
| Teachers can include topics in their created courses | 9 | 9 | 27 | 1.436170213 | 7 | 1.559020045 | 3 | 0.940438871 | 0.707738158 |
| Teachers can include price in their created courses | 8 | 8 | 24 | 1.276595745 | 3 | 0.668151448 | 1 | 0.313479624 | 1.547592764 |
| Course recommendations are based on student learning history | 9 | 9 | 27 | 1.436170213 | 9 | 2.004454343 | 9 | 2.821316614 | 0.420533774 |
| Students can browse courses by topic | 4 | 3 | 11 | 0.585106383 | 3 | 0.668151448 | 2 | 0.626959248 | 0.59605528 |
| Students can browse courses by fees | 4 | 3 | 11 | 0.585106383 | 3 | 0.668151448 | 2 | 0.626959248 | 0.59605528 |
| Posts can include educational resources | 7 | 5 | 19 | 1.010638298 | 3 | 0.668151448 | 3 | 0.940438871 | 0.887793524 |
| Posts can include documents | 7 | 5 | 19 | 1.010638298 | 3 | 0.668151448 | 3 | 0.940438871 | 0.887793524 |
| Posts can include multimedia | 7 | 5 | 19 | 1.010638298 | 3 | 0.668151448 | 3 | 0.940438871 | 0.887793524 |
| Students can post topic requests by specifying the topic's title | 9 | 9 | 27 | 1.436170213 | 3 | 0.668151448 | 1 | 0.313479624 | 1.74104186 |
| Students can post topic requests by specifying the topic's description | 4 | 4 | 12 | 0.638297872 | 3 | 0.668151448 | 1 | 0.313479624 | 0.773796382 |
| Students can post topic requests by specifying the topic's estimated hours | 9 | 9 | 27 | 1.436170213 | 3 | 0.668151448 | 1 | 0.313479624 | 1.74104186 |
| Students can post topic requests by specifying the topic's rate per hour | 9 | 9 | 27 | 1.436170213 | 3 | 0.668151448 | 1 | 0.313479624 | 1.74104186 |
| Students can post topic requests by specifying the topic's required skills | 9 | 9 | 27 | 1.436170213 | 3 | 0.668151448 | 1 | 0.313479624 | 1.74104186 |
| Job recommendations are based on teacher profile | 9 | 9 | 27 | 1.436170213 | 9 | 2.004454343 | 9 | 2.821316614 | 0.420533774 |
| Teachers can browse topic requests based on skills | 4 | 3 | 11 | 0.585106383 | 3 | 0.668151448 | 2 | 0.626959248 | 0.59605528 |
| Teachers can browse topic requests based on price | 4 | 3 | 11 | 0.585106383 | 3 | 0.668151448 | 2 | 0.626959248 | 0.59605528 |
| Teachers can browse topic requests based on the number of proposals | 4 | 3 | 11 | 0.585106383 | 3 | 0.668151448 | 2 | 0.626959248 | 0.59605528 |
| Teachers can view active proposals they have initiated | 7 | 7 | 21 | 1.117021277 | 3 | 0.668151448 | 1 | 0.313479624 | 1.354143669 |
| Teachers can view requested proposals they have initiated | 7 | 7 | 21 | 1.117021277 | 3 | 0.668151448 | 1 | 0.313479624 | 1.354143669 |
| Teachers can view closed proposals they have initiated | 3 | 3 | 9 | 0.478723404 | 3 | 0.668151448 | 1 | 0.313479624 | 0.580347287 |
| Teachers can include the duration while scheduling live sessions | 3 | 2 | 8 | 0.425531915 | 3 | 0.668151448 | 1 | 0.313479624 | 0.515864255 |
| Teachers can include the topic while scheduling live sessions | 7 | 8 | 22 | 1.170212766 | 3 | 0.668151448 | 1 | 0.313479624 | 1.418626701 |
| Teachers can include the duration while scheduling webinars | 3 | 2 | 8 | 0.425531915 | 3 | 0.668151448 | 1 | 0.313479624 | 0.515864255 |
| Teachers can include the topic while scheduling webinars | 7 | 8 | 22 | 1.170212766 | 3 | 0.668151448 | 1 | 0.313479624 | 1.418626701 |
| Students can browse upcoming sessions by date | 4 | 3 | 11 | 0.585106383 | 3 | 0.668151448 | 2 | 0.626959248 | 0.59605528 |
| Students can browse upcoming sessions by time | 4 | 3 | 11 | 0.585106383 | 3 | 0.668151448 | 2 | 0.626959248 | 0.59605528 |
| Students can browse upcoming sessions by topic | 4 | 3 | 11 | 0.585106383 | 3 | 0.668151448 | 2 | 0.626959248 | 0.59605528 |
| Students can share their screens in virtual classrooms | 7 | 7 | 21 | 1.117021277 | 8 | 1.781737194 | 8 | 2.507836991 | 0.367967053 |
| Students can mute their microphones in virtual classrooms | 6 | 6 | 18 | 0.957446809 | 8 | 1.781737194 | 8 | 2.507836991 | 0.315400331 |
| Students can stop their video feeds in virtual classrooms | 6 | 6 | 18 | 0.957446809 | 8 | 1.781737194 | 8 | 2.507836991 | 0.315400331 |
| Students can raise their hands in virtual classrooms | 6 | 6 | 18 | 0.957446809 | 8 | 1.781737194 | 8 | 2.507836991 | 0.315400331 |
| Students can view all the participants in virtual classrooms | 3 | 2 | 8 | 0.425531915 | 8 | 1.781737194 | 8 | 2.507836991 | 0.140177925 |
| Teachers can share their screens in virtual classrooms | 8 | 9 | 25 | 1.329787234 | 8 | 1.781737194 | 8 | 2.507836991 | 0.438056015 |
| Teachers can mute their microphones in virtual classrooms | 3 | 2 | 8 | 0.425531915 | 8 | 1.781737194 | 8 | 2.507836991 | 0.140177925 |
| Teachers can stop their video feeds in virtual classrooms | 8 | 9 | 25 | 1.329787234 | 8 | 1.781737194 | 8 | 2.507836991 | 0.438056015 |
| Teachers can raise their hands in virtual classrooms | 3 | 2 | 8 | 0.425531915 | 8 | 1.781737194 | 8 | 2.507836991 | 0.140177925 |
| Teachers can view all the participants in virtual classrooms | 5 | 7 | 17 | 0.904255319 | 8 | 1.781737194 | 8 | 2.507836991 | 0.29787809 |
| Virtual classroom hosts can record the class for later review | 9 | 8 | 26 | 1.382978723 | 8 | 1.781737194 | 8 | 2.507836991 | 0.455578256 |
| Messages can include emojis | 2 | 4 | 8 | 0.425531915 | 8 | 1.781737194 | 8 | 2.507836991 | 0.140177925 |
| Calls can be audio calls | 8 | 8 | 24 | 1.276595745 | 9 | 2.004454343 | 9 | 2.821316614 | 0.373807799 |
| Calls can be video calls | 2 | 4 | 8 | 0.425531915 | 9 | 2.004454343 | 9 | 2.821316614 | 0.1246026 |
| Students can message each other in groups | 9 | 8 | 26 | 1.382978723 | 9 | 2.004454343 | 9 | 2.821316614 | 0.404958449 |
| Students can message each other individually | 6 | 7 | 19 | 1.010638298 | 9 | 2.004454343 | 9 | 2.821316614 | 0.295931175 |
| Students can call each other in groups | 9 | 8 | 26 | 1.382978723 | 9 | 2.004454343 | 9 | 2.821316614 | 0.404958449 |
| Students can call each other individually | 6 | 7 | 19 | 1.010638298 | 9 | 2.004454343 | 9 | 2.821316614 | 0.295931175 |
| Teachers can message each other in groups | 9 | 8 | 26 | 1.382978723 | 9 | 2.004454343 | 9 | 2.821316614 | 0.404958449 |
| Teachers can message each other individually | 9 | 8 | 26 | 1.382978723 | 9 | 2.004454343 | 9 | 2.821316614 | 0.404958449 |
| Teachers can call each other in groups | 9 | 8 | 26 | 1.382978723 | 9 | 2.004454343 | 9 | 2.821316614 | 0.404958449 |
| Teachers can call each other individually | 9 | 8 | 26 | 1.382978723 | 9 | 2.004454343 | 9 | 2.821316614 | 0.404958449 |
| Students can switch to audio/video calls | 2 | 3 | 7 | 0.372340426 | 8 | 1.781737194 | 8 | 2.507836991 | 0.122655684 |
| Students can mute their microphones | 2 | 3 | 7 | 0.372340426 | 8 | 1.781737194 | 8 | 2.507836991 | 0.122655684 |
| Students can stop their video feeds | 2 | 3 | 7 | 0.372340426 | 8 | 1.781737194 | 8 | 2.507836991 | 0.122655684 |
| Teachers can switch to audio/video calls | 2 | 3 | 7 | 0.372340426 | 8 | 1.781737194 | 8 | 2.507836991 | 0.122655684 |
| Teachers can mute their microphones | 2 | 3 | 7 | 0.372340426 | 8 | 1.781737194 | 8 | 2.507836991 | 0.122655684 |
| Teachers can stop their video feeds | 2 | 3 | 7 | 0.372340426 | 8 | 1.781737194 | 8 | 2.507836991 | 0.122655684 |
| Communities can include community names | 8 | 8 | 24 | 1.276595745 | 3 | 0.668151448 | 2 | 0.626959248 | 1.300484247 |
| Communities can include community descriptions | 2 | 3 | 7 | 0.372340426 | 3 | 0.668151448 | 2 | 0.626959248 | 0.379307905 |
| Communities can include community cover pictures | 5 | 6 | 16 | 0.85106383 | 3 | 0.668151448 | 2 | 0.626959248 | 0.866989498 |
| Notifications can be made pushed in the Notification window of Mobile screen | 2 | 3 | 7 | 0.372340426 | 3 | 0.668151448 | 2 | 0.626959248 | 0.379307905 |
| Notifications can be set to related to scheduled meetings | 6 | 8 | 20 | 1.063829787 | 6 | 1.336302895 | 3 | 0.940438871 | 0.588882722 |
| Notifications can be set for notifying user about scheduled maintenance window | 9 | 9 | 27 | 1.436170213 | 6 | 1.336302895 | 3 | 0.940438871 | 0.794991674 |
| Notifications sound can be made musical | 2 | 3 | 7 | 0.372340426 | 7 | 1.559020045 | 4 | 1.253918495 | 0.170331177 |
| Notifications sound can be made distinctive from system's sound | 2 | 3 | 7 | 0.372340426 | 7 | 1.559020045 | 4 | 1.253918495 | 0.170331177 |
| Notifications can be turned on from LearnLance system | 2 | 3 | 7 | 0.372340426 | 4 | 0.890868597 | 4 | 1.253918495 | 0.245311368 |
| Notifications can be turned off from LearnLance system | 2 | 3 | 7 | 0.372340426 | 4 | 0.890868597 | 4 | 1.253918495 | 0.245311368 |
| Notifications can also be viewed in the Drawer of LearnLance | 8 | 8 | 24 | 1.276595745 | 7 | 1.559020045 | 3 | 0.940438871 | 0.629100585 |
| Users can provide a valid email address | 9 | 9 | 27 | 1.436170213 | 4 | 0.890868597 | 2 | 0.626959248 | 1.192487512 |
| Users can provide a unique email address | 9 | 9 | 27 | 1.436170213 | 4 | 0.890868597 | 2 | 0.626959248 | 1.192487512 |
| Users can provide a strong password | 9 | 8 | 26 | 1.382978723 | 5 | 1.113585746 | 3 | 0.940438871 | 0.873200024 |
| Password can comprise of capital letters | 7 | 8 | 22 | 1.170212766 | 5 | 1.113585746 | 3 | 0.940438871 | 0.738861559 |
| Password can comprise of special characters | 7 | 8 | 22 | 1.170212766 | 5 | 1.113585746 | 3 | 0.940438871 | 0.738861559 |
| Password can comprise of number | 7 | 8 | 22 | 1.170212766 | 5 | 1.113585746 | 3 | 0.940438871 | 0.738861559 |
| Users can provide their First Name | 9 | 9 | 27 | 1.436170213 | 3 | 0.668151448 | 1 | 0.313479624 | 1.74104186 |
| Users can provide their Last Name | 7 | 6 | 20 | 1.063829787 | 3 | 0.668151448 | 1 | 0.313479624 | 1.289660637 |
| Users can provide their preference of using LearnLance | 8 | 9 | 25 | 1.329787234 | 3 | 0.668151448 | 2 | 0.626959248 | 1.35467109 |
| Payment can be encrypted end-to-end | 9 | 9 | 27 | 1.436170213 | 9 | 2.004454343 | 9 | 2.821316614 | 0.420533774 |
| Payment can be made via Account Number entered by Student | 9 | 9 | 27 | 1.436170213 | 8 | 1.781737194 | 7 | 2.194357367 | 0.498857999 |
| Certificate can be made with Course Title | 9 | 8 | 26 | 1.382978723 | 3 | 0.668151448 | 2 | 0.626959248 | 1.408857934 |
| Certificate can be made with Student Name on it | 9 | 9 | 27 | 1.436170213 | 3 | 0.668151448 | 2 | 0.626959248 | 1.463044778 |
| Certificate can include LearnLance logo for authentication | 9 | 9 | 27 | 1.436170213 | 3 | 0.668151448 | 2 | 0.626959248 | 1.463044778 |
| Badges can be made to have ranks such as Gold, Silver, and Bronze | 3 | 2 | 8 | 0.425531915 | 5 | 1.113585746 | 3 | 0.940438871 | 0.268676931 |
| Students can showcase their LearnLance associated certificates on their profiles | 9 | 9 | 27 | 1.436170213 | 6 | 1.336302895 | 2 | 0.626959248 | 0.87052093 |
| Students can showcase their LearnLance associated badges on their profiles | 9 | 9 | 27 | 1.436170213 | 6 | 1.336302895 | 2 | 0.626959248 | 0.87052093 |
| Student connecting other students can make posts visible on connected students’ feed | 6 | 4 | 16 | 0.85106383 | 7 | 1.559020045 | 3 | 0.940438871 | 0.41940039 |
| Student connecting other students can make posts visible on their profile | 6 | 4 | 16 | 0.85106383 | 7 | 1.559020045 | 3 | 0.940438871 | 0.41940039 |
| Students following other teachers can make posts visible on feed of the student feed | 9 | 8 | 26 | 1.382978723 | 7 | 1.559020045 | 3 | 0.940438871 | 0.681525634 |
| Teachers connecting other teachers can make posts visible on connected teachers’ feed | 6 | 4 | 16 | 0.85106383 | 7 | 1.559020045 | 3 | 0.940438871 | 0.41940039 |
| Teachers can write description to invite teacher for Joint Account | 9 | 8 | 26 | 1.382978723 | 6 | 1.336302895 | 3 | 0.940438871 | 0.765547538 |
| Teachers can Update the Course Content to the course he/she was invited | 5 | 6 | 16 | 0.85106383 | 7 | 1.559020045 | 5 | 1.567398119 | 0.363280356 |
| Admin can have a dashboard to remove Student from LearnLance | 3 | 2 | 8 | 0.425531915 | 3 | 0.668151448 | 1 | 0.313479624 | 0.515864255 |
| Admin can have a dashboard to remove Teacher from LearnLance | 3 | 2 | 8 | 0.425531915 | 3 | 0.668151448 | 1 | 0.313479624 | 0.515864255 |
| Student can have the option to delete his/her account from LearnLance drawer | 7 | 9 | 23 | 1.223404255 | 7 | 1.559020045 | 3 | 0.940438871 | 0.602888061 |
| Teacher can have the option to delete his/her account from LearnLance drawer | 7 | 9 | 23 | 1.223404255 | 7 | 1.559020045 | 3 | 0.940438871 | 0.602888061 |
| Student can update their interests by clicking on Pencil Icon on their own profile screen | 9 | 9 | 27 | 1.436170213 | 6 | 1.336302895 | 2 | 0.626959248 | 0.87052093 |
| Teacher can update their certificates by clicking on Pencil Icon on their own profile screen | 9 | 9 | 27 | 1.436170213 | 6 | 1.336302895 | 2 | 0.626959248 | 0.87052093 |
| Teacher can update their experience by clicking on Pencil Icon on their own profile screen | 9 | 9 | 27 | 1.436170213 | 6 | 1.336302895 | 2 | 0.626959248 | 0.87052093 |
| Admin can have a dashboard to add Student on LearnLance | 3 | 2 | 8 | 0.425531915 | 3 | 0.668151448 | 1 | 0.313479624 | 0.515864255 |
| Admin can have a dashboard to add Teacher on LearnLance | 3 | 2 | 8 | 0.425531915 | 3 | 0.668151448 | 1 | 0.313479624 | 0.515864255 |
| Admin can have dashboard to view Revenue-related data with specific time periods | 3 | 2 | 8 | 0.425531915 | 3 | 0.668151448 | 1 | 0.313479624 | 0.515864255 |
| User can do commenting in the hierarchical manner | 4 | 6 | 14 | 0.744680851 | 7 | 1.559020045 | 4 | 1.253918495 | 0.340662354 |
| User can like post by just a click | 5 | 6 | 16 | 0.85106383 | 5 | 1.113585746 | 2 | 0.626959248 | 0.596373402 |
| User can like comment by just a click | 2 | 3 | 7 | 0.372340426 | 5 | 1.113585746 | 2 | 0.626959248 | 0.260913363 |
| Totals | 772 | 568 | 2312 | 122.9787234 | 679 | 151.2249443 | 458 | 143.5736677 |  |