

**LA GRANDEE INTERNATIONAL COLLEGE**

**Simalchaur, Pokhara Nepal**

**Final defence**

On

**“Decentralize academy ‘D-academe’”**

**Submitted to:**

Bachelor of Computer Application (BCA) Program

In partial fulfilment of the requirements for the degree of BCA under

Pokhara University

**Submitted by:**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name:** | **Course** | **Semester** | **P.U. Registration Number** |
| Sangam Subedi | BCA | 6th | 2021-1-53-0366 |
| Amit Baral | BCA | 6th | 2021-1-53-0346 |
| Prabin Shrestha | BCA | 6th | 2021-1-53-0358 |

**Date:22/02/2025**

**Acknowledgement**

We would like to express our gratitude to our BCA coordinator **Mr. Kundan Chaudhary**, Project supervisor **Mr. Sunil Sapkota** and LA Grandee International Collage for their support and contributions to the development of our Junkyard Management System.

This project is done for the in partial fulfilment of the requirements for BCA (Bachelor of Computer Application) program under Pokhara University. Our project was made possible by the effort and dedication of our team members. We thank our dedicated team for their hard work and contributions to the game. We are grateful for the guidance and mentorship provided by our respected sir **Mr. Sunil Sapkota.**

Sincerely,

Sangam Subedi

Amit Baral

Prabin Shrestha

**Declaration for**

**“Decentralize academy ‘D-academe’”**

**Student’s Declaration**

We, **Amit Baral**, **Sangam Subedi**, and **Prabin Shrestha** being students of the sixth semester at LA GRANDEE International College, Faculty of Science and Technology ‘kha’, Pokhara University, do hereby declare that the project proposal submitted to the aforementioned institution is an original work completed by us in partial fulfilment of the requirements for the Bachelor of Computer Application (BCA) program, under the supervision of Sir **Mr. Sunil Sapkota.** We further state that no resources other than those specifically listed have been utilized in the completion of this project.

Name: Sangam Subedi Name: Amit Baral

Class Roll No.: 22 Class Roll No.: 2

PU-Registration No.: 2021-1-53-0366 PU-Registration No.: 2021-1-53-0346

Semester: 6th Semester Semester: 6th Semester

**Date: 22/02/2025 Date: 22/02/2025**

Signature: ............... Signature: ...............

Name: Prabin Shrestha

Class Roll No.: 14

PU-Registration No.: 2021-1-53-0358

Semester: 6th Semester

**Date: 22/02/2025**

Signature: ...............

**Supervisor’s Declaration**

I hereby recommend that this project entitled “**Decentralize academy ‘D-academe’**” is done under my supervision by **Prabin Shrestha**, **Sangam Subedi, Amit Baral** during their sixthSemester in partial fulfilment of the requirements for the degree of **BCA** under **Pokhara University** is completed to my satisfaction and be processed for final evaluation.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Mr. Sunil Sapkota**

**Date:22/02/2025**

**Letter of Approval**

We certify that we have examined this report entitled **“Decentralize academy ‘D-academe’”** and are satisfied with the project defence. It is satisfactory in the scope and qualify as project in partial fulfilment of the requirements for the degree of **BCA** under **Pokhara University.**

|  |  |  |
| --- | --- | --- |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  **Supervisor**  **Mr. Sunil Sapkota** | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  **Examiner**  **Er. Ashmit Nepali**  **Date:22/02/2025** | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  **Program Coordinator**  **Mr. Kundan Chaudhary** |

# Abstract

D-Academe is a decentralized online learning management system that truly rethinks ways in which to break traditional barriers to learning. It focuses on providing a smooth, flexible, and inclusive educational experience for both learners and educators. Decentralized technology empowers it to offer transparency, security, and shared control, thus enabling active contribution by users themselves to shaping the learning ecosystem.

Core D-Academe features include live learning sessions, course creation and management, secure blockchain-based certification, and collaborative tools such as discussion forums and peer reviews. Real-time interaction lets the system create an interactive, community-based environment in which learning can take place. It also makes tracking user progress easier, simplifies content delivery, and offers actionable insights with comprehensive reporting.

With D-Academe, the aim is to provide an intuitive, scalable, secure platform that will help make education for all a reality, while fostering innovation and inclusion. It bridges the gap between traditional learning and online learning, contributing toward an equitable and sustainable educational environment.

**TABLE OF CONTENTS**

[Abstract vi](#_Toc191123028)

[1. INTRODUCTION 1](#_Toc191123029)

[2. PROBLEM STATEMENT 2](#_Toc191123030)

[3. OBJECTIVES 3](#_Toc191123031)

[4. BACKGROUND STUDY 4](#_Toc191123032)

[5. FESIBILITY ANALYSIS 5](#_Toc191123033)

[6. REQUIREMENT DOCUMENTATION 7](#_Toc191123034)

[6.1. Functional Requirements: 7](#_Toc191123035)

[6.2. Non-Functional Requirements: 8](#_Toc191123036)

[6.3. Technical Requirements: 9](#_Toc191123037)

[6.3.1. Hardware Requirements 9](#_Toc191123038)

[6.3.2. Software Requirements 10](#_Toc191123039)

[6.3.3. Deployment Requirements 10](#_Toc191123040)

[6.3.4. Dependencies 11](#_Toc191123041)

[6.3.5. Network Requirements 11](#_Toc191123042)

[6.4. Requirements Matrix: 12](#_Toc191123043)

[7. SYSTEM DESIGN 13](#_Toc191123044)

[7.1 Er-Diagram 13](#_Toc191123045)

[7.2 Dataflow Diagram 15](#_Toc191123046)

[8. DEVELOPMENT 19](#_Toc191123047)

[8.1 Development methodology 19](#_Toc191123048)

[8.1.1 Planning Phase 19](#_Toc191123049)

[8.1.2 Risk Analysis 20](#_Toc191123050)

[8.1.3 Engineering 20](#_Toc191123051)

[8.1.4 Evaluation 20](#_Toc191123052)

[8.2 Project Gantt Chart/ Timeline Chart 21](#_Toc191123053)

[8.3 Tools used 22](#_Toc191123054)

[9. TESTING 24](#_Toc191123055)

[9.1 Test Case 26](#_Toc191123056)

[9.2 A Test Matrix with Tested modules or Requirement Traceability Matrix 35](#_Toc191123057)

[9.3 User Acceptance Testing (UAT) 39](#_Toc191123058)

[10. PROJECT RESULTS 41](#_Toc191123059)

[11. FUTURE ENHANCEMENTS 42](#_Toc191123060)

[12. CONCLUSION 43](#_Toc191123061)

[13. REFERENCES 44](#_Toc191123062)

LIST OF FIGURES

[Figure 7.1 ER- Diagram for User 11](#_Toc191086458)

[Figure 7.2 ER- Diagram for Admin 12](#_Toc191086459)

[Figure 7.3 Level-0, DFD 13](#_Toc191086460)

[Figure 7.4 Level-1 DFD for User 13](#_Toc191086461)

[Figure 7.5 Level-1 DFD for Admin 14](#_Toc191086462)

[Figure 7.6 Level-2, DFD for User login 15](#_Toc191086463)

[Figure 7.7 Level-2, DFD for User Register 15](#_Toc191086464)

[Figure 7.8 Level-2, DFD for free course 16](#_Toc191086465)

[Figure 7.9 Level-2, DFD for token purchase 16](#_Toc191086466)

[Figure 8.1 Spiral Model 17](#_Toc191086467)

[Figure 8.2 Timeline chart 19](#_Toc191086468)

[Figure 9.1 Software Testing Life cycle 20](#_Toc191086469)

**List of Tables**

[Table 6‑1 Requirements matrix 12](#_Toc191123134)

[Table 8‑1 Technology stack 23](#_Toc191123135)

[Table 9‑1 Module testing of a login function for user 28](#_Toc191123136)

[Table 9‑2 Module testing of a Main Menu function for user 30](#_Toc191123137)

[Table 9‑3 Module testing of a login function for admin 32](#_Toc191123138)

[Table 9‑4 Module testing of a Main Menu function for admin 34](#_Toc191123139)

[Table 9‑5 Requirement Traceability Matrix 38](#_Toc191123140)

[Table 9‑6 User Acceptance Testing (UAT) 40](#_Toc191123141)

**Abbreviations**

|  |  |
| --- | --- |
| D-Academe | Decentralized Academy |
| GDPR | General Data Protection Regulation |
| NFT | Non Fungible Token |
| API | Application program Interface |
| Q&A | Question and Answer |
| ER | Entity Relationship |
| DFD | Data Flow Diagram |
| DB | Database |
| E-mail | Electronic mail |
| UI | User Interface |
| STLC | Software testing life cycle |
| Etc. | Etcetera |

# INTRODUCTION

With the world in this century changing day in and day out, solutions in the education sector are in demand. Education has to adapt to the needs of all learners while breaking through cultural, geographical, and socio-economic barriers. We're delighted to introduce to you our decentralized online platform, D-Academe, which is redefining education provision and experience across the globe.

This is in tune with its goal to create no obstacles-to access, geographic location, course inflexibility-stand in the way of achieving the finest in higher learning. For D-Academe, learning has to be flexible, inclusive, and empowering. D-Academe thus affords an equal opportunity to people of different backgrounds to come forward and excel in academics.

Among its many central features are live learning-creating possibilities for real interactions in real-time between students and educators. Live learning consists of dynamic discussions, active participation, and problem-solving, rather than the simple one-way lecture that students usually see recorded on the screen. All these interactive methods permit improvements in understanding and give an even firmer feeling of community among learners from traditional class to online education.

What makes D-Academe unique is the decentralized model in the hands of the users. Students and educators are the key contributors in the design of the learning ecosystem: course creation, content curation among others. Transparency, security, and equitability have been addressed with a blockchain-based decentralized platform that always bears the seal of trust and accountability.

Furthermore, D-Academe encourages collaboration through the use of discussion forums, peer reviews, and mentorship opportunities. It is in such places that learners will find themselves on a mutual journey of sharing knowledge and growing together.

Our vision is a kind of education that is two-way, interactive, inclusive, with the needs of the modern learner at the forefront. We see D-Academe setting the standard for the educational platform where learning is accessible, meaningful, and transformational to all.

# PROBLEM STATEMENT

* **Centralized Control**: Traditional online learning platforms often face challenges related to centralized control. This can lead to issues with data ownership, censorship, and limited transparency.
* **Limited Accessibility**: Access to quality education can be restricted by geographical location, financial resources, or language barriers. Traditional platforms may not effectively address these limitations.
* **Lack of trust and security**: Concerns regarding data privacy, security breaches, and the potential for fraud can deter individuals from fully engaging in traditional online learning platforms.
* **Interoperability Challenges**: The lack of interoperability between different learning platforms hinders the seamless transfer of credits, credentials, and other learning materials.
* **Live interaction**: There is no live interaction approaches in existing online platform so we cannot determine whether the courses provided by different online platform is effectively working or not.

# OBJECTIVES

1. Create a real-time learning environment for active participation, discussions, and collaboration with educators.
2. Offer flexible, self-paced learning that fits diverse schedules and personal commitments.

# BACKGROUND STUDY

The rise of decentralized technologies, coupled with demands for more accessible and flexible online education, recent growth in the field of decentralized technologies has brought huge potential to change the e-learning platform beyond imagination. There are a bunch of problems that conventional systems face: poor ownership of content, low learner engagement, no personalization, and even questions of credibility about their certificates. The integration of blockchain technology into these challenges is an innovative solution, especially in the use of NFTs and utility tokens. Blockchain allows for secure and transparent ownership of content, enables collaboration through token-based incentivization, and enables personalized learning by securely managing learner data in a privacy-preserving manner.

Besides, blockchain-based credentials verified with NFTs have given an unchangeable solution to authentication problems and build up trust and credibility on a global standard. Besides, the decentralization of platforms seeks to ensure scalability and equity in trying to exclude intermediaries from the process, hence empowering the learner and educators in charge of shaping the education environment. These indeed provide the bases on which our project rests, delivering a transformative and learner-centred platform that redefines how education is accessed, delivered, and experienced.

# FESIBILITY ANALYSIS

A feasibility analysis evaluates the project's potential for success, ensuring its credibility for potential investors and stakeholders. This study examines key areas to identify the practicality and effectiveness of implementing the D-Academe platform. Below are the feasibility studies conducted for the project:

**5.1 Technical Feasibility**

* **Hardware and Software**: Ensure the availability of servers, hosting services, and compatible software resources to support PHP-based development. Utilize technologies such as XAMPP for database management and ensure compatibility with front-end tools like Tailwind CSS.
* **Technical Expertise**: Evaluate the availability of skilled PHP developers within the team and access to relevant technical resources to ensure efficient platform development and maintenance.

**5.2 Operational Feasibility**

* **User Acceptance**: Assess whether the platform is user-friendly and meets the expectations of both learners and educators. Conduct user testing to ensure a seamless and engaging experience.
* **Operational Impact**: Analyze how D-Academe will transform traditional education systems by offering decentralized, transparent, and flexible learning opportunities, and ensure smooth integration into the existing educational environment.
  1. **Economic Feasibility**
* **Cost-Benefit Analysis**: Perform a comprehensive analysis to determine whether the platform's benefits, such as improved access to education, enhanced engagement, and decentralized control, outweigh the development, hosting, and operational costs.
* **Scalability**: Consider the financial viability of scaling the platform to accommodate a growing user base in the long term.

**5.4 Legal and Compliance Feasibility**

* **Data Privacy and Security**: Ensure compliance with global data privacy regulations, such as GDPR, and implement robust security measures to protect user data and platform integrity.
* **Licensing**: Verify that all third-party libraries, frameworks, and tools used in PHP development comply with licensing and legal requirements.

**5.5 Schedule Feasibility**

* **Project Timeline**: Develop a detailed project timeline, including stages for planning, development, testing, and deployment, ensuring realistic milestones are set.
* **Deadlines**: Assess whether the project can be completed within the allocated timeframe without compromising quality and functionality.

# REQUIREMENT DOCUMENTATION

Requirements analysis is the process of defining the expectations and needs of users for the D-Academe platform. In software engineering, this is referred to as requirements engineering, capturing, or gathering. Below is the analysis conducted for D-Academe:

## Functional Requirements:

* **User Management:**
* Allow students, educators, and administrators to create, update, and delete accounts.
* Maintain user profiles with roles and permissions.
* Provide authentication and authorization for secure access.
* **Course Management:**
* Enable educators to create, update, and delete courses.
* Allow students to enroll in courses and track progress.
* Support multimedia content upload (videos, documents, etc.).
* **Live Learning:**
* Provide real-time interactive classes with live chat and Q&A functionalities.
* Enable screen sharing, collaborative whiteboards, and breakout rooms.
* **Decentralized Control:**
* Allow users to participate in decision-making processes, such as course approval or platform updates, through voting mechanisms.
* **Search and Reporting:**
* Advanced search options for courses, instructors, and learners.
* Generate reports on user activity, course performance, and platform usage.
* **Content Certification:**
* Issue blockchain-based digital certificates for completed courses, ensuring authenticity and tamper-proof credentials.
* **Collaboration Features:**
* Provide discussion forums, peer review systems, and mentorship opportunities.
* Enable collaborative learning and resource sharing among users.

## Non-Functional Requirements:

* **Performance:**
* The platform should handle real-time interactions and content delivery seamlessly, even during peak usage periods.
* Optimize database queries and API responses for faster performance.
* **Security:**
* Ensure data privacy through compliance with regulations like GDPR.
* Protect user data with encryption and secure communication protocols.
* Implement role-based access control and multi-factor authentication.
* **Scalability:**
* The system architecture should support a growing user base and additional features without compromising performance.
* Handle increasing volumes of courses, certifications, and user interactions efficiently.
* **Reliability:**
* Guarantee minimal downtime with robust server and database management.
* Include backup and disaster recovery mechanisms to ensure data integrity.
* **Usability:**
* Design an intuitive user interface tailored for students and educators.
* Minimize the learning curve through simple navigation and responsive design.
* **Maintainability:**
* Ensure the platform is easy to update and enhance, with modular and well-documented code.
* Facilitate efficient debugging and integration of new technologies or features. Non-Functional Requirements:

## Technical Requirements:

### Hardware Requirements

* + Development Environment
* Processor: Intel Core i5 or equivalent (64-bit, 2.5 GHz or higher).
* RAM: 8 GB minimum (16 GB recommended for multitasking).
* Storage: 100 GB SSD (for OS, tools, and local blockchain node data).
* GPU: Not mandatory, but recommended for video encoding/streaming (NVIDIA/AMD with 4GB VRAM).
* Production Server
* Processor: 4-core CPU (e.g., AWS EC2 t3.xlarge, Azure F4s\_v2).
* RAM: 16 GB minimum (32 GB for high-traffic scalability).
* Storage: 500 GB+ SSD (scalable for blockchain data, videos, and databases).
* Network: Dedicated bandwidth (1 Gbps recommended).
* Client Devices
* Browser: Latest versions of Chrome, Firefox, or Brave (with MetaMask extension).
* OS: Windows 10/11, macOS Monterey+, or Linux (Ubuntu 22.04+).
* RAM: 4 GB minimum (8 GB recommended).

### Software Requirements

* Blockchain Development:
  + Solidity Compiler (v0.8.x).
  + Remix IDE / VS Code with Solidity extensions.
  + MetaMask (browser extension for wallet integration).
* Backend Development:
  + PHP (v8.0+), XAMPP (Apache, MySQL, PHP).
* Frontend Development:
  + JavaScript (ES6+), React.js (optional for advanced UI).
* Video Tools:
  + OBS Studio (v28.0+ for recording/streaming).
  + Livepeer API/SDK integration.
* Diagramming:
  + SmartDraw (for DFDs, ER diagrams).

### Deployment Requirements

* Blockchain:
  + Ethereum Node (Infura/Alchemy for remote access).
  + OpenSea API (for NFT marketplace integration).
* Web Server:
  + Apache/Nginx (with PHP 8.0+ support).
* Database:
  + MySQL (v8.0+) or MariaDB.
* Decentralized Storage:
  + Livepeer (video streaming), IPFS/Filecoin (optional for video storage).

### Dependencies

* Libraries:
  + Web3.js (v5.x) / ethers.js (v6.x) for Ethereum interaction.
  + OpenZeppelin Contracts (v4.x) for ERC721 NFT templates.
* APIs:
  + Livepeer API (for video streaming).
  + OpenSea API (NFT display and trading).

### Network Requirements

* Bandwidth:
  + Minimum 10 Mbps upload/download for video streaming.
  + Stable connection for blockchain transactions (Ethereum RPC calls).
* Security:
  + HTTPS (SSL/TLS) encryption for all web traffic.
  + Firewall rules to restrict unauthorized access to servers.
* Blockchain Network:
  + Ethereum Mainnet (production) / sepholia Testnet (development).

## Requirements Matrix:

|  |  |  |  |
| --- | --- | --- | --- |
| SN. | Required modules, system, and features. | Description for the modules | Priority (High, Moderate, low) |
| 1. | Login and security system | Users have to login in the system | High |
| 2. | Registration function | Registration contains Name , password, gender, DOB, phone number, email,photo | High |
| 3. | Couse buy | Couse id, title, price | high |
| 4. | Live stream | Admin will provide the detail | high |
| 5. | Token purchase | Token purchase by user | high |
| 7. | Design interface | Designing the interface | high |

Table 6‑1 Requirements matrix

# SYSTEM DESIGN

Er-Diagrams, Dataflow, Algorithm and Flowchart are used for understanding the system's design and its functionalities, and both are important for creating proper documentation.

## 7.1 Er-Diagram

An Entity-Relationship (ER) diagram is a visual representation of a database's structure. It uses entities (objects or concepts) and their relationships to illustrate how data is organized and connected within a database system. ER diagrams are widely used in database design and modelling to help understand and plan data relationships.

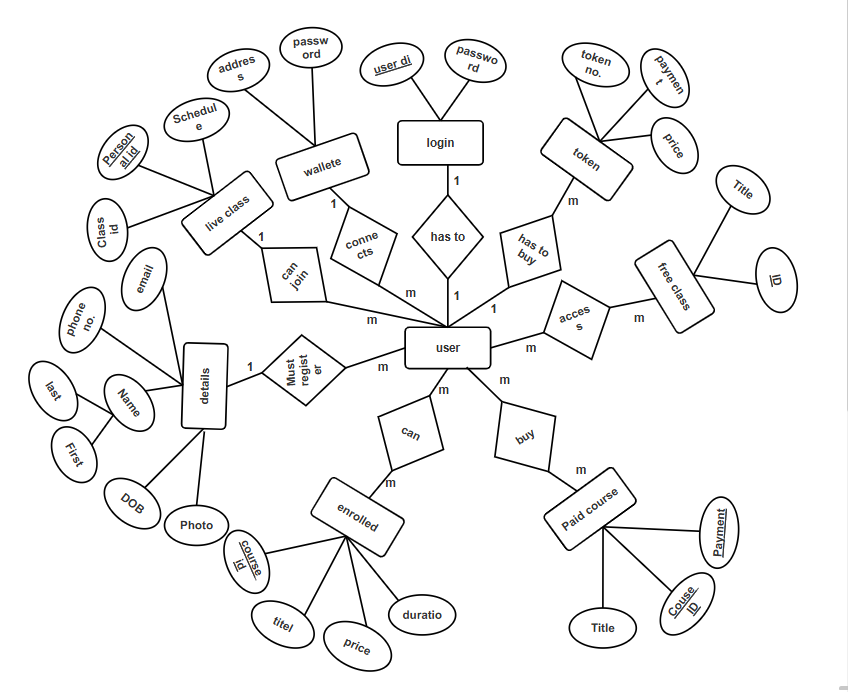


Figure 7.1 ER- Diagram for User

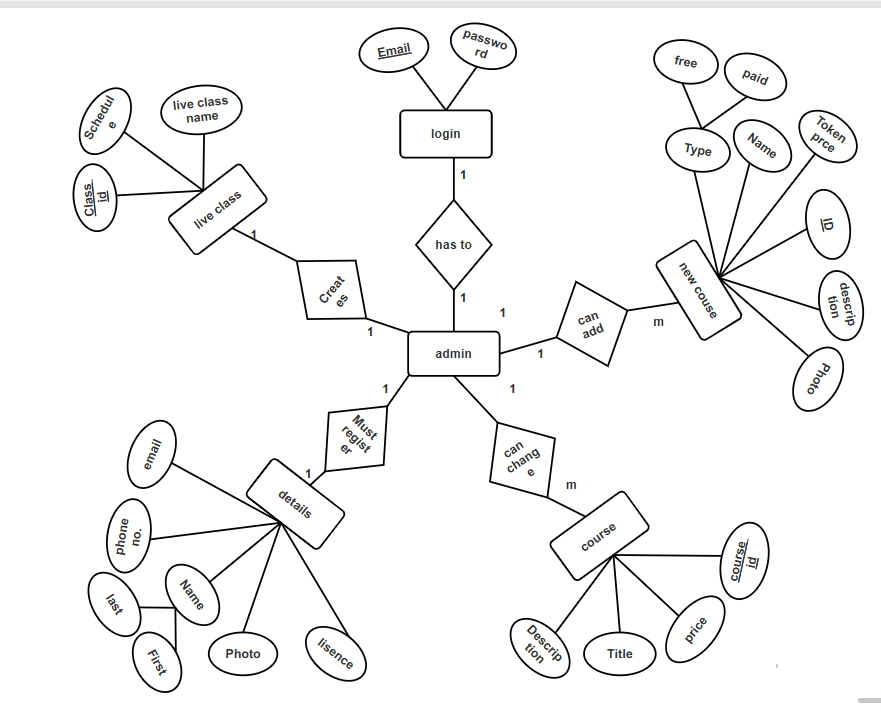


Figure 7.2 ER- Diagram for Admin

## Dataflow Diagram

It is a diagrammatic representationthat portrays the flow of datain a system or a process. Helps communicates the general data flowstructure of a proposed system to the system designer, programmer, and end-users.

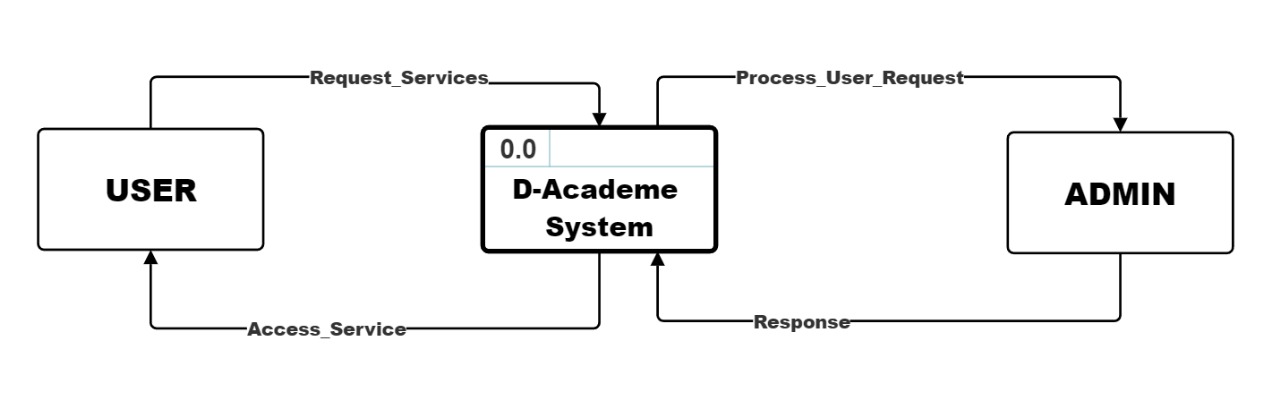


Figure 7.3 Level-0, DFD

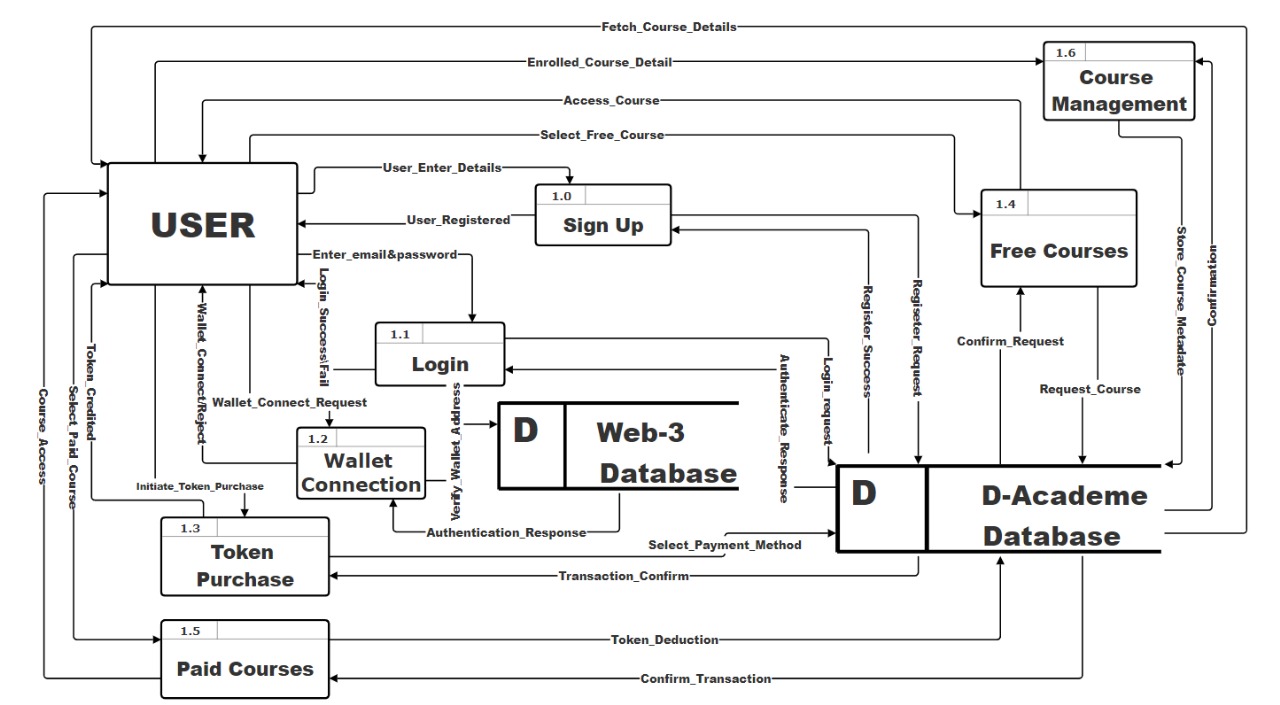
**

Figure 7.4 Level-1 DFD for User

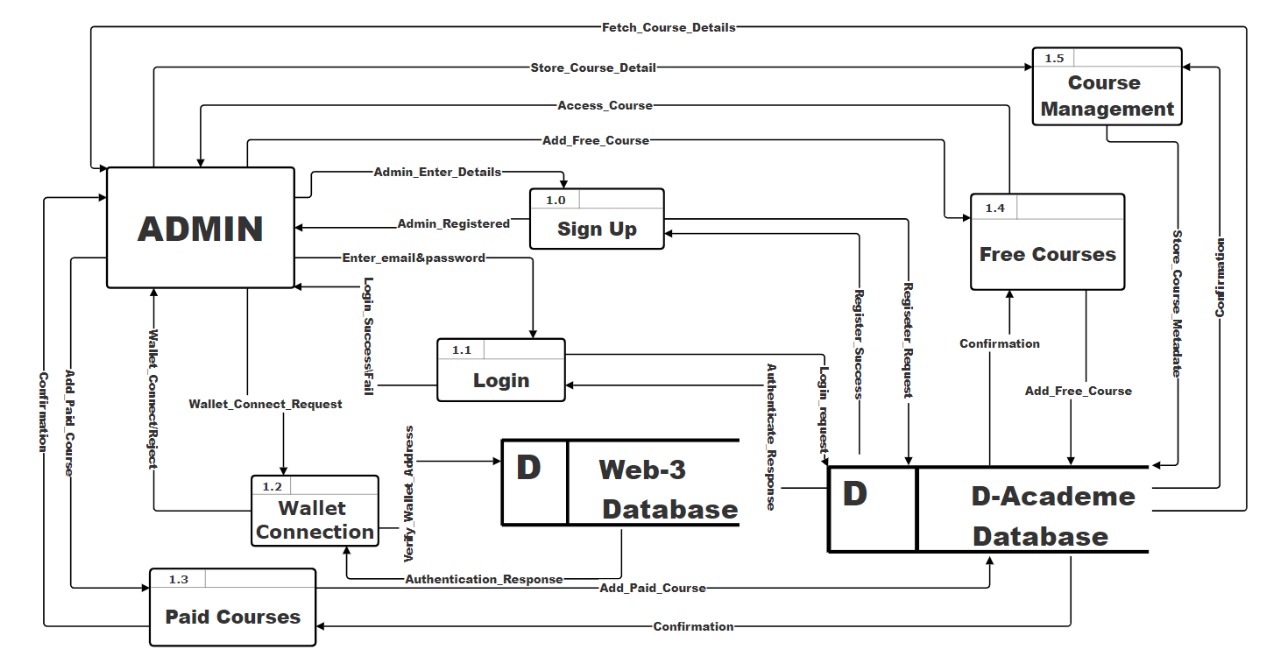
**

Figure 7.5 Level-1 DFD for Admin

**DFD Index:**

|  |  |
| --- | --- |
| INDEX | |
| Function | SN |
| Sign up | 1.0 |
| Login | 1.1 |
| Wallet Connection | 1.2 |
| Token Purchase | 1.3 |
| Free course | 1.4 |
| Paid Courses | 1.5 |
| Course Management | 1.6 |

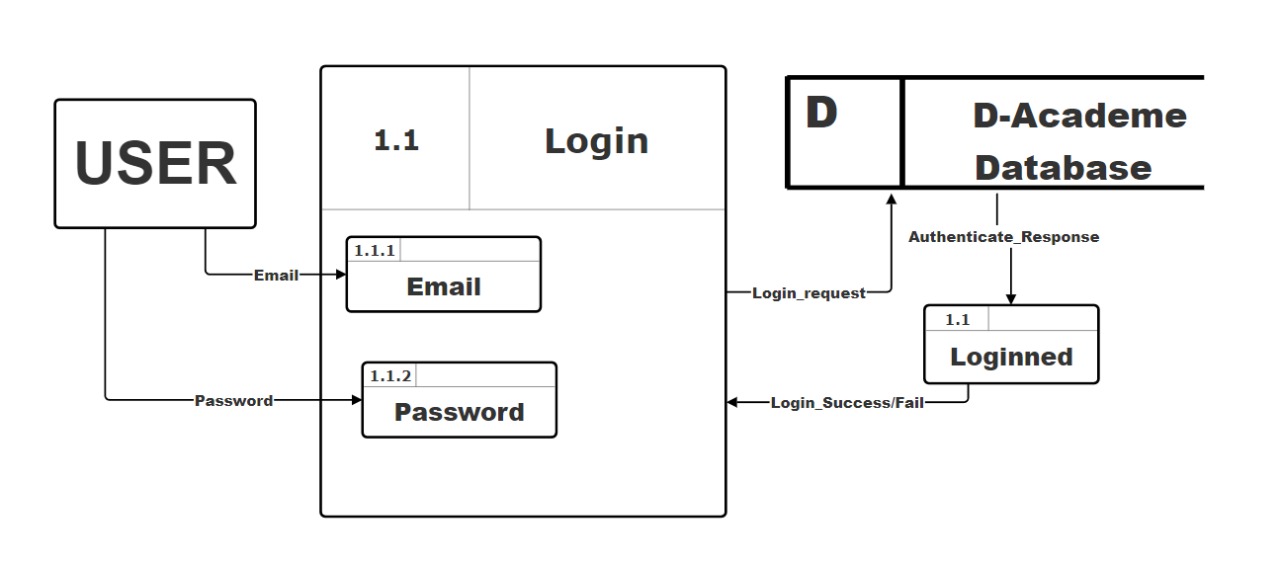


Figure 7.6 Level-2, DFD for User login

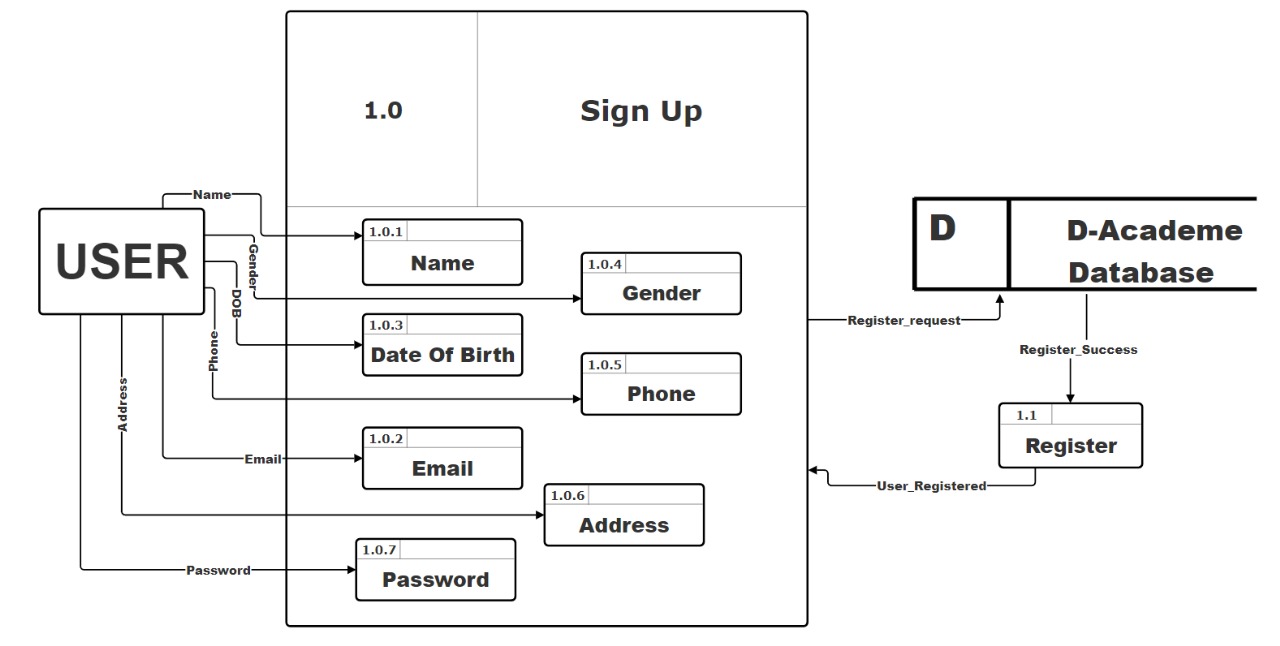


Figure 7.7 Level-2, DFD for User Register

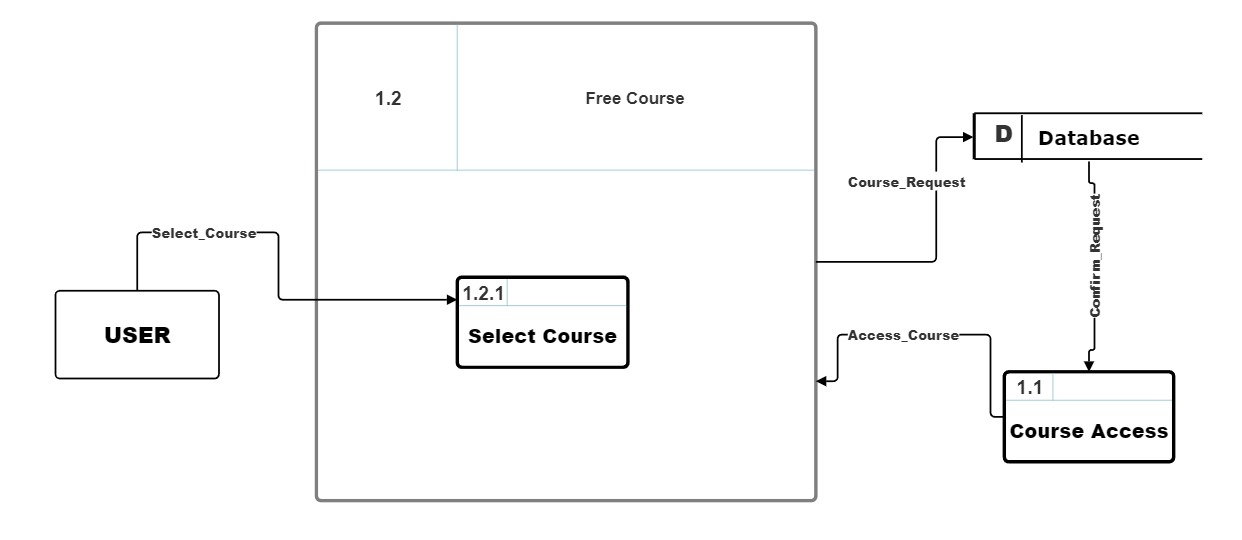


Figure 7.8 Level-2, DFD for free course

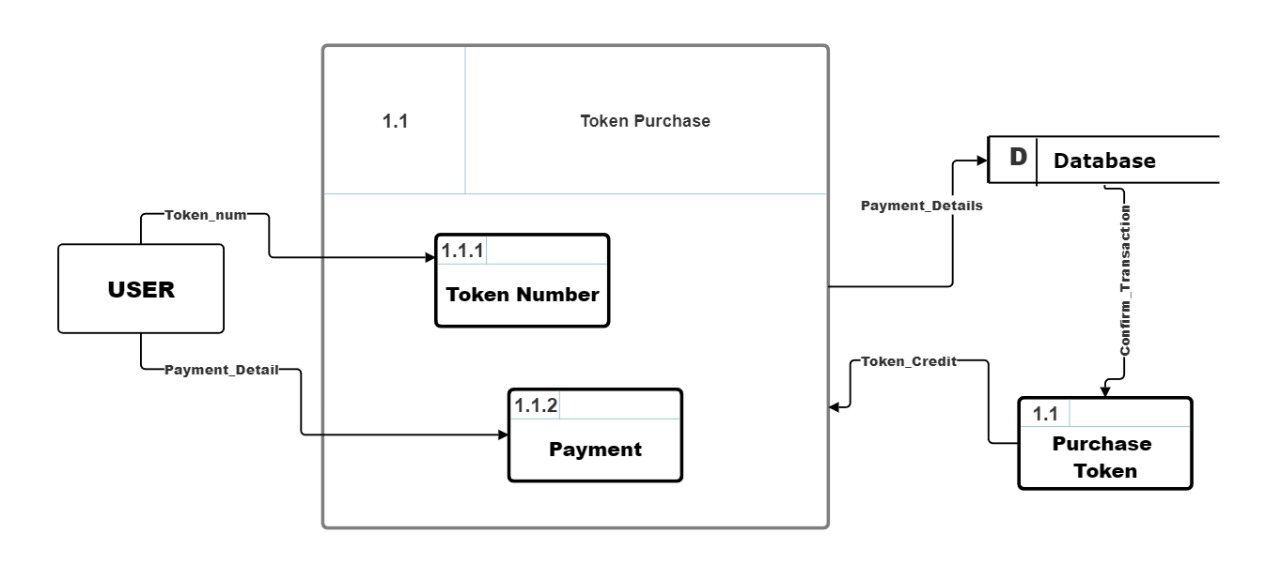


Figure 7.9 Level-2, DFD for token purchase

# DEVELOPMENT

## Development methodology

The Spiral Model is used in this project. A sophisticated method for developing software, the spiral model offers a framework for taking on challenging tasks and refining and assessing risks iteratively. The model is shown as a spiral, where each loop or phase denotes a different step in the process of evolution. It serves as the foundation for the majority of software development processes, which include planning, risk analysis, engineering, and evaluation.

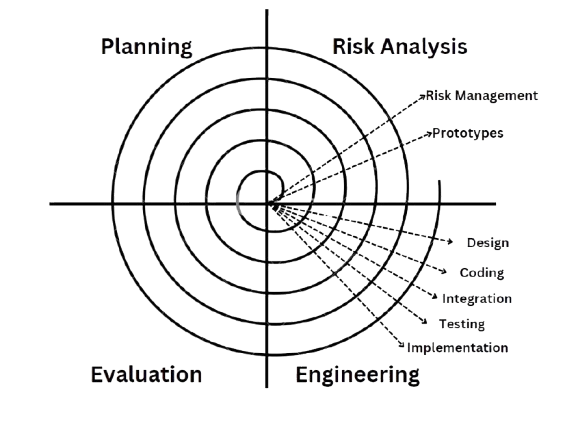


Figure 8.1 Spiral Model

The phases in Spiral model are: -

### 8.1.1 Planning Phase

The project's goals, parameters, and extent are specified at the planning phase. In order to determine the resources, deadlines, and deliverables, stakeholders collaborate. In order to create a strong basis for the project, requirements collecting and preliminary feasibility study are also completed during this phase.

### 8.1.2 Risk Analysis

The spiral model's most unique characteristic is its risk analysis. Prototypes and models are used in this phase to identify, assess, and mitigate potential risks. To lower the chance of failure, the development team evaluates operational, financial, schedule-related, and technological risks.

### 8.1.3 Engineering

This is the real development phase, during which system design, testing, and coding are done. The product is built incrementally, with every cycle resulting in a deliverable or prototype that changes with each iteration.

### 8.1.4 Evaluation

Stakeholders assess the current build or prototype at the end of each cycle. Testers, clients, and end users all provide feedback. In the following iteration, the product might see improvements or adjustments in response to this input. This stage assists in guaranteeing that the product meets the expectations of the user.

## Project Gantt Chart/ Timeline Chart

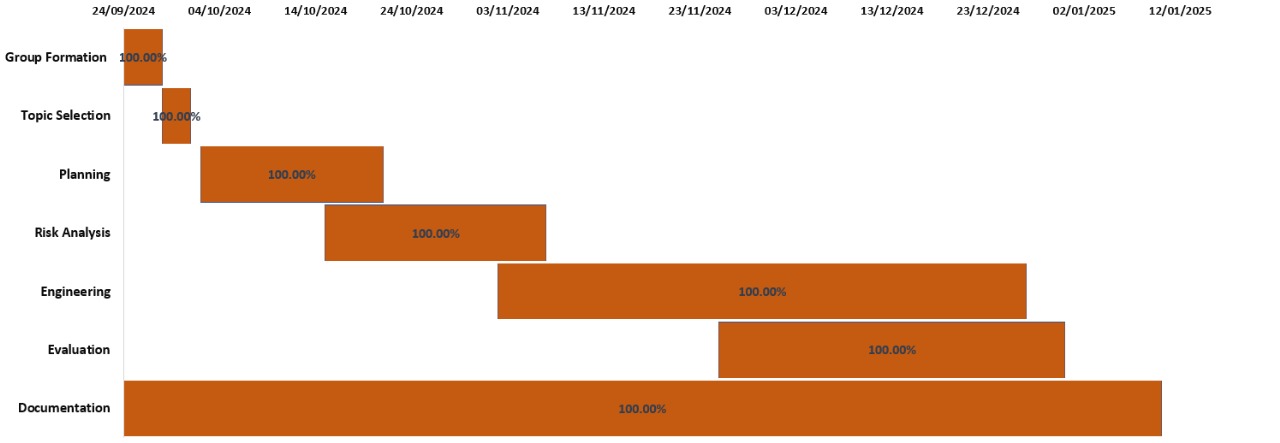


Figure 8.2 Timeline chart

|  |  |  |
| --- | --- | --- |
| INDEX | | |
| S. No. | Color | Work Status |
| 1 | Orange | Completed |
| 2 | Blue | Not completed |

## Tools used

* **System Overview**

D-Academe utilizes blockchain technology for transparency and security. The platform integrates:

* **Ethereum Blockchain** for smart contract execution.
* **Solidity** for writing smart contracts.
* **PHP** for backend development.
* **Livepeer** for decentralized video streaming.
* **OBS Studio** for video recording and streaming.
* **VS Code** and **Remix IDE** for development.
* **XAMPP for localhost, Database.**
* Technology Stack

|  |  |  |
| --- | --- | --- |
| **Component** | **Technology Used** | **Description** |
| Blockchain Platform | Ethereum | Hosts smart contracts for academic operations (e.g., certifications, NFTs). |
| Smart Contracts | Solidity | Programming language for writing Ethereum-based smart contracts. |
| Backend Development | PHP | Server-side logic, APIs, and integration with blockchain. |
| Local Server/Database | XAMPP (Apache, MySQL) | Local development environment and database management. |
| Frontend Development | JavaScript | User interface development for interacting with blockchain and streaming. |
| Video Streaming | Livepeer | Decentralized video streaming protocol for lectures and tutorials. |
| Recording/Streaming Tool | OBS Studio | Captures, records, and streams high-quality educational content. |
| Development Tools | VS Code, Remix IDE | Code editing (PHP/JS) and smart contract development/debugging (Solidity). |
| Documentation | MS Word, MS PowerPoint | System documentation, user guides, and presentations. |
| Diagramming Tools | SmartDraw | Design of DFDs, ER diagrams, and system architecture visuals. |

Table 8‑1 Technology stack

# TESTING

The procedure of software testing is also known as STLC (Software Testing Life Cycle) which includes phases of the testing process. The testing process is executed in a well-planned and systematic manner. All activities are done to improve the quality of the software product.

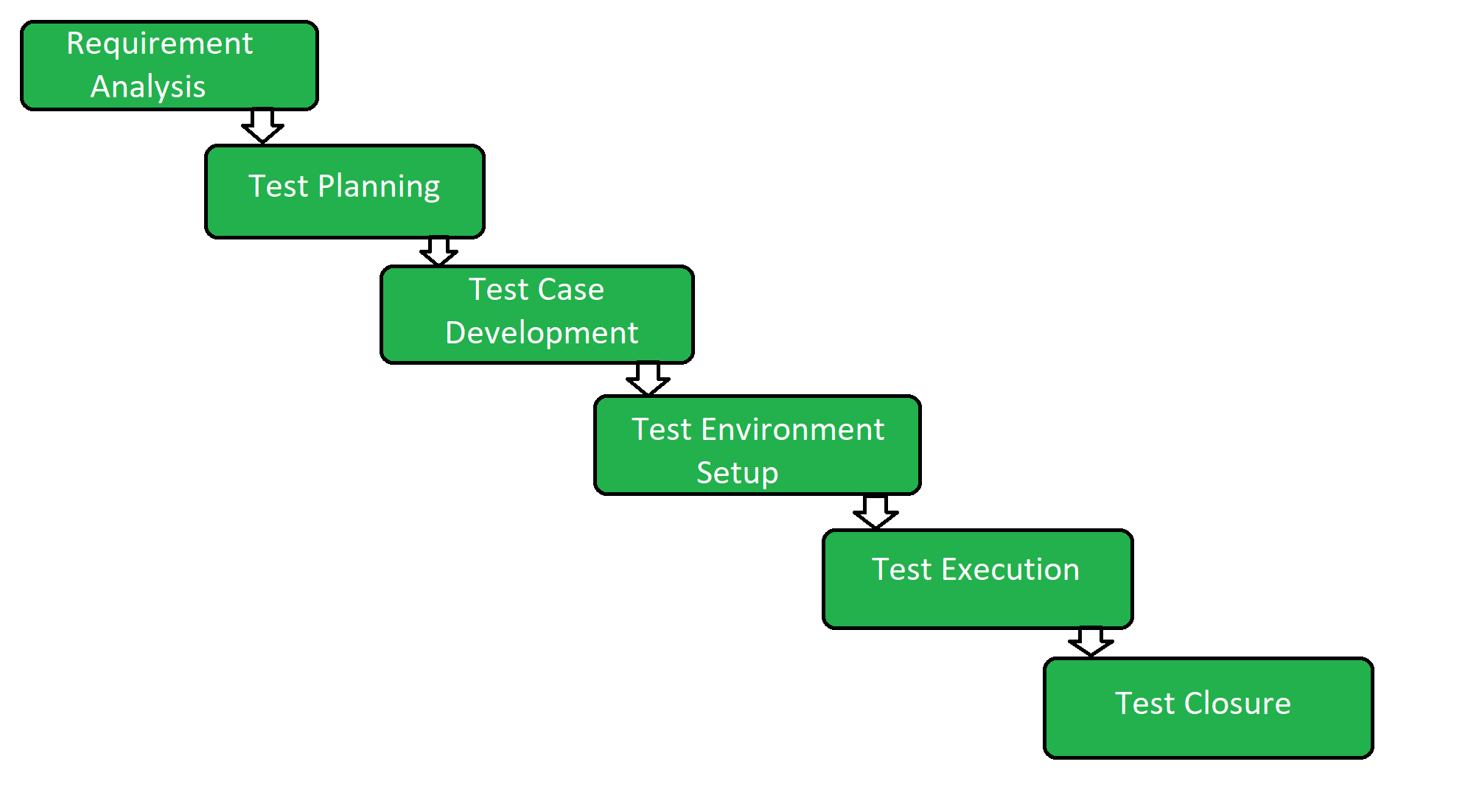


Figure 9.1 Software Testing Life cycle

There are 6 major phases of STLC −

* **Requirement Analysis**: Requirement Analysis is the first step of Software Testing Life Cycle (STLC). In this phase our team understands the requirements like what is to be tested.
* **Test Planning or Execution**: Test Planning is most efficient phase of software testing life cycle where all testing plans are defined. This phase gets started once the requirement gathering phase is completed.
* **Test Case Development:** The test case development phase gets started once the test planning phase is completed. In this phase our team noted down the detailed test cases. We also prepared the required test data for the testing.
* **Test Environment Setup**: Test environment setup is the vital part of the STLC. Basically test environment decides the conditions on which software is tested. This is independent activity and can be started along with test case development. In this process the testing team is not involved either the developer or the customer creates the testing environment.
* **Test Execution**: After the test case development and test environment setup test execution phase gets started. In this phase our team start executing test cases based on prepared test cases in the earlier step.
* **Test Closure**: This is the last stage of STLC in which the process of testing is analyzed.

## Test Case

**Title**: Module testing of a Login function– Authentication of users

**TID**: 001, 002

**Description:**A user should be able to log in with proper username and password

**Precondition:** The user must have username and password that is previously registered.

**Test Case for User**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test cases# | Test ID | Test Case | Test Steps | Test Data | Expected Results |
| 1. | 001 | Verify Login | Step1: Runs the system.  Step2: Enter Email and password.  Step3: Press the enter button | Email= Sprabin@gmail.com  Password= password | Login successful. |
| 2. | 002 | Verify Login | Step1: Runs the system.  Step2: Enter the incorrect Email or password.  Step3: Press enter button | Email= Sprabin@gmail.com  Password: password | Login unsuccessful. |
| 3. | 003 | Register /. Sign up | Step1: click login.  Step2: click sign up  Step3: enter name, address, DOB, phone, email, picture,  password.  Step4: press sign up. | Name: Ram Thapa  Email:  [Ram009@gmail.com](mailto:Ram009@gmail.com)  Address: Pokhara  DOB:2001/03/02  Phone: 9846123456  Password:  Ram1234 | Sign successful |
| 4. | 004 | Forger password. | Step1:enter email.  Step2: press request for reset.  Step3:check the mail.  Step4: change the password. | [Ram009@gmail.com](mailto:Ram009@gmail.com) | Successful |

Table 9‑1 Module testing of a login function for user

Title: Module Testing of Main menu of user

TID: 005,006,007,008,009

Description: A user should use only verified Email ID and password.

Precondition: The main menu should show multiple option to perform different task.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test cases# | Test ID | Test Case | Test Steps | Test Data | Expected Results |
| 1. | 005 | Verify Dashboard | Step1: Login to the system with the verified Email and password. | Email:  Ram009@gmail.com  Password: Ram1234 | Dashboard showed. |
| 2. | 006 | Free course | Step1: choose one free course among the different options. | blockchain | Free course material accessed. |
| 3. | 007 | Token purchase | Step1:click to the token purchase option.  Step2: Enter the required number of token.  Step3: choose the payment method.  Step4: Press buy token. | Entered 15 | Successful |
| 4. | 008 | Buy course | Step1: choose the course among the different course.  Step2: check the required number of token.  Step3: if available, press the buy course. | clarity | Successful |
| 5. | 009 | Live class | Step1: click live class  Step2: enter key provided by admin.  Step4: press join class. | Key inserted. | Successful. |

Table 9‑2 Module testing of a Main Menu function for user

**Title**: Module testing of a Login function– Authentication of users

**TID**: 001, 002,003,004

**Description:**A user should be able to log in with proper Email and password

**Precondition:** The user must have Email and password that is previously registered.

**Test Case for admin**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test cases# | Test ID | Test Case | Test Steps | Test Data | Expected Results |
| 1. | 001 | Verify admin Login | Step1: Runs the system.  Step2: Enter Email and password.  Step3: Press the enter button | Email=  Ramesh98@gmail.com  Password= Ramesh1234 | Login successful. |
| 2. | 002 | Verify admin Login | Step1: Runs the system.  Step2: Enter the incorrect Email or password.  Step3: Press enter button | Email=  Ramesh  Password:  Ramesh1234 | Login unsuccessful. |
| 3. | 003 | Register /. Sign up | Step1: click login.  Step2: click sign up  Step3: enter name, address, phone, email, license, password.  Step4: press sign up. | Name: Ramesh Thapa  Address: pokhrara 17  Phone: 9846234567  Email:  Ramesh98@gmail.com  License:  Password: Ramesh1234 |  |
| 4. | 004 | Forger password. | Step1:enter email.  Step2: press request for reset.  Step3:check the mail. Go to link  Step4: change the password. | Ramesh98@gmail.com |  |

Table 9‑3 Module testing of a login function for admin

**Title**: Module Testing of Main menu of admin

**TID**: 005,006,007

**Description**: A user should use only verified Email and password.

**Precondition**: The main menu should show multiple option to perform different task.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test cases# | Test ID | Test Case | Test Steps | Test Data | Expected Results |
| 1. | 005 | Verify Dashboard | Step1: Login to the system with the verified Email and password. | Email:  Ram009@g,ail.com  Password: Ram1234 | Dashboard showed. |
| 2. | 006 | Live class | Step1: choose live class.  Step2: enter class name.  Step3: press create stream. | Hello class | Live class generated. |
| 3. | 007 | Add course | Step1:choose course.  Step2: Enter the course name to add.  Step3: enter course name, course type, course image, description and file.  Step4: Press add course. | blockchain | Successful |

Table 9‑4 Module testing of a Main Menu function for admin

## A Test Matrix with Tested modules or Requirement Traceability Matrix

**Requirement Traceability Matrix (RTM)**is a document that maps and traces the user requirement with different test cases.

The traceability matrix is typically a worksheet that contains the requirements with its all possible test scenarios and cases and their current state, i.e. if they have been passed or failed. This would help to understand the level of testing activities done for the specific product. It is delivered at the conclusion of software development life cycle.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| S.N | Test case | Test case | Expected result | Actual result | Test result |
| 101 | 001 | Enter valid Email and password | System should direct to Dashboard | Display Dashboard | Successful |
| 102 | 002 | Enter wrong Email and password | System should not move to Dashboard | Display invalid password or Email | Successful |
| 103 | 003 | Enter correct Email and password after TC 002 | System should move to Dashboard | Display  Dashboard | Successful |
| 104 | 004 | Register for user and admin. | System should show name, address, email, phone, picture, Document, DOB and password to sign up. | Display all the expected result | Successful |
| 105 | 005 | Connect wallet | Connect Meta mask wallet in extension and successful login with metamask. | Connects the wallet and shows the wallet details and tokens. | Successful |
| 106 | 006 | Buy token | System should show the place to enter the number of token. | Token purchase shown as expected. | Successful |
| 107 | 007 | Buy course | System should show option to choose the course . | Shown as expected and purchased successfully. | Successful |
| 108 | 008 | Live class | System should show the place to enter the key provide by the admin. | It showed all the necessary things to join the live class. | Issue while connecting |
| 109 | 009 | Live class creation from admin | System should show the place to enter live class name | Program showed all the list of the necessary approaches. | Successful |
| 1110 | 010 | Add course | System should show the all the details that are need to fill up to add the new course. | It showed all the detail required to add new course. | Successful |
| 111 | 011 | About | Program should show information about our system | Showed as expected. | Successful |

Table 9‑5 Requirement Traceability Matrix

## User Acceptance Testing (UAT)

User Acceptance Testing (UAT) also known as beta or end-user testing, is defined as testing the software by the user or client to determine whether it can be accepted or not. This is the final testing performed once the functional, system and regression testing are completed.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| No. | Acceptance requirement (USR) | Critical (DEV) | | Test Results (DEV) | | Comments (USR) |
| Yes | No | Accept | Reject |
| 1. | The login system should ensure security | Affirm. |  | Affirm |  | The system is felt secured. |
| 2. | Registration of new user | Affirm. |  | Affirm. |  | The registration is easy and simple |
| 3. | Different modes such as add, buy, stream etc. should be included | Affirm. |  | Affirm. |  | modes is included and can be enhanced. |
| 4. | User details and transaction should be able to be in record | Affirm. |  | Affirm. |  | Details can be filled and stored |
| 5. | Changes should immediately display to owner after transaction | Affirm. |  | Affirm. |  | changes are displayed after the transaction is committed. |
| 7. | User should be able to view detail of course | Affirm. |  | Affirm. |  | User can easily view the course details. |
| 8. | User should be able to delete account | Affirm. |  | Affirm. |  | User can easily delete account |

Table 9‑6 User Acceptance Testing (UAT)

# PROJECT RESULTS

The D-Academe platform has ensured transformative outcomes in the realm of online education. Its architecture and real-time interactivity ensure extended access to learning that is more engaging and inclusive. It further provided seamless course management for both learners and educators, bright live learning sessions, and blockchain certification for transparency and trust in credentials.

These discussion forums and peer reviews are part of the collaborative tools that have grown a sense of community and knowledge sharing, hence enriching the learning experience. The decentralized model empowers users in active participation during decision-making, hence making the platform align with the principles of equity and fairness. This is showing great potential for scalability and adaptability for diverse educational objectives without losing performance and security.

D-Academe addressed the constraints of the conventional education system quite well with sustainable and innovative learning practices. This platform, using advanced technology, will facilitate an engaging, inclusive, and learner-centered environment; hence, it can change the educational environment in a way that will be commensurate with global objectives of equitable and lifelong learning opportunities.

# FUTURE ENHANCEMENTS

1. Chat box Integration:

Introduce an interactive chat box for real-time communication between learners, educators, and support teams.

1. AI Integration:

Implement AI-powered features like personalized course recommendations, automated feedback, and intelligent chat assistants for 24/7 user support.

1. Advance Learning

Implementing Advanced class features at live class and enhancing live class feature, enabling better experience and features.

# CONCLUSION

In conclusion, D-Academe successfully establishes a decentralized, interactive, and secure platform that addresses the limitations of traditional education systems. By leveraging innovative technologies and fostering collaboration, the platform enhances accessibility, engagement, and inclusivity, paving the way for a more sustainable and learner-centric educational future.

# REFERENCES

* eSatya. (n.d.). Blockchain in Nepal | eSatya - Blockchain initiative in Nepal. Retrieved February 21, 2025, from <https://esatya.io/>
* Infinite IT Solutions Nepal. (n.d.). Blockchain development company - Infinite IT Solutions Nepal. Retrieved February 21, 2025, from <https://www.infiniteitsolutionsnepal.com/blockchain-development.php>
* Anticipation Hub. (2024, February 27). Using blockchain technology to streamline anticipatory action in Nepal. Retrieved from <https://www.anticipation-hub.org/news/using-blockchain-technology-to-streamline-anticipatory-action-in-nepal>
* Digital Gurkha. (2024, May 24). Blockchain in Nepal: Its future and career in Nepal. Retrieved from <https://digitalgurkha.com/blog/blockchain-in-nepal>
* OpenZeppelin. (2024). ERC721 - OpenZeppelin Docs. OpenZeppelin Contracts Documentation. <https://docs.openzeppelin.com/contracts/4.x/erc721>