

American International University-Bangladesh (AIUB)

**Software Project 1**

**YOU LEARN**

Under the supervision of

**MD. HASIBUL HASAN**

Assistant Professor

Department of Computer Science

American International University-Bangladesh (AIUB)

Spring 2018-2019

**Submitted By:**

|  |  |
| --- | --- |
| Name | ID |
| Tasneem,Shehneeza | 16-32167-2 |
| Khan,Saumik Tanveer | 16-32294-2 |
| Faruk,Md Omar | 16-32350-2 |
|  |  |

**Declaration**

We certify that this project is our original work. No part of this has been submitted elsewhere partially or fully for the award of any other degree. Any material reproduced in this project has been properly acknowledge.

We declare that this project does not contain any content that discloses the secret of any organization or related parties. American International University-Bangladesh (AIUB) will not be held liable for any such activity, as for the project is presented as our original work.

---------------------------------- ------------------------------

Name: Tasneem,Shehneeza Name: Khan ,Saumik Tanveer

ID: 16-32167-2 ID: 16-32294-2  
Department: SE Department: SE

-------------------------------

Name: Faruk,Md Omar

ID: 16-32350-2   
Department: SE

**Approval**

This web based official Application titled **“YOU LEARN”** has been submitted to the following respected member of the board of Examiners of the Faculty of Science and Information Technology in partial fulfillment of the requirements for the degree of Bachelor of Science in Software Engineering on date May 19, 2019 and has been accepted satisfactory.

|  |  |
| --- | --- |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  **MD. Hasibul Hasan**  Assistant Professor  Faculty of Science & Technology American International University-Bangladesh | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  **AZM Ehtesham Chowdhury**  Assistant Professor  Faculty of Science & Technology American International University-Bangladesh |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  **Dr. M. M. Mahbubul Syeed**  Associate Professor & Head  Faculty of Science & Technology American International University-Bangladesh | \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  **Professor Dr. Tafazzal Hossain**  Dean  Faculty of Science & Technology American International University-Bangladesh |

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Dr. Carmen Z. Lamagna**

Vice Chancellor

American International University-Bangladesh

**Acknowledgement**

This Project is our cordial effort and our supervisor’s initiative and constant motivation. But First of all, we would like to be grateful to the Almighty Allah, who gives us the effort to work on this project for the last one semester. Specially thanks goes to our honorable supervisor **MD. HASIBUL HASAN** and Co-supervisor **AZM EHTESHAM CHOWDHURY,** faculty members, American International University-Bangladesh (AIUB) for this enormous support, inspiration, and helpful criticism. Their excellent supervision and constant support has made this project possible. We are very grateful to him for giving us the opportunity to work with him. We thank our co-supervisor for being so supportive and helpful. We must not forget to convey our cordial gratitude to our honorable Vice Chancellor, **Dr. Carmen Z. Lamagna**, Our Dean, Prof. **Dr. Tafazzal Hossain**, Our Associate Dean, **Mashiour Rahman**, Our Director, **Dr. Dip Nandi**, and Our Head of Department, **Dr. M. M. Mahbubul Syeed** for their constant motivation and support.

Last but not least we thank our respectable parents for educating us with aspect from both arts and sciences, for their unconditional support and encouragement to pursue our interests, even when interests went out of boundary.

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Table of Contents**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

Declaration . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .2

Approval . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 3

Acknowledgement . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 4

**Chapter 1 : Project Initition**

* 1. Background to the problem . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 9
  2. Obective of this project . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 9
  3. SDLC Methods . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .10

1.3.1 Phase of SDLC . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 10

1.3.2 SDLC Moels . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .12

**Chapter 2 : Requirements Ecilitation**

2.1 Product Perspective . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 22 2.2 Product Features . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 22 2.3 Operating Environment . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 22 2.4 Design and Implementation Constraints . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 23 2.5 Assumptions and Dependencies . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .23

**Chapter 3 : Requirements Specification**

3.1 User Documentation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 24 3.2 User Interface . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .26 3.3 Hardware Interface . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 26 3.4 Software Interface . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 26 3.5 System Features . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 27 3.5.1 Mentor’s Registration . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 27 3.5.2 Homework . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 28 3.5.3 Yearly Exam . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .29 3.5.4 Upload And Update Notes . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 30 3.6 Other Nonfunctional Requirements . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 31 3.6.1 Performance Requirements . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 31 3.6.2 Safety Requirements . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .32 3.6.3 Security Requirements . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 32

3.7 Software Quality Attributes . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 32 3.8 Business Rules . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 32

**Charter 4 : Software Design Specification**

4.1 Use Case Diagram . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .33 4.2 Er Diagram . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .34 4.3 Activity Diagram . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 35

**Chapter 5 : Reference**

Reference . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .44

**Chapter 6 : Conclusion**

Conclusion . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .45

**List of Tables**

Table 3.1 User Documentation . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 24

**List of Figures**

Fig 1-1 : Phase of SDLC . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 10 Fig 1-2 : Waterfall Model . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .12 Fig 1-3 : Iterative Model . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 13 Fig 1-4 : Spiral Model . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 14 Fig 1-5 : V Model . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 15 Fig 1-6 : Prototype Model . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 16 Fig 1-7 : Agile Model . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 17 Fig 1-8 : Scrum . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .18 Fig 1.9 : Extreme Programming(XP). . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 19 Fig 1.10 : FDD . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 20 Fig 4.1 : Use Case Diagram . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .33 Fig 4.2 : Er Diagram of the system . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 34 Fig 4.3 : Activity Diagram(Log in) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 36 Fig 4.3.1 : Activity Diagram(Registration) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .37 Fig 4.3.2 : Activity Diagram(Admin). . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 38 Fig 4.3.3 : Activity Diagram(Add Tutor). . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .39 Fig 4.4.4 : Activity Diagram(Mentor’s Activity) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 40 Fig 4.4.5 : Activity Diagram(Student Activity) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 41 Fig 4.4.6 : Activity Diagram(Teacher’s Activity) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 42 Fig 4.4.7 :Activity Diagram(Remove User) . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .43

**PROJECT SUMMARY**

It's a SRS documentation for “YOULEARN” which is a web based learning application . The purpose for this project is to build an interactive web application by which the childen of age 3-8 years will be able to learn.

This system is designed to assist kids in learning alphabets, words, numbers, tables, mathematical operations and much more. By adding graphical features such as both still and moving images, short animations it will be more visually attractive and helpful for kids to learn fast and make it more fun for them. There will also some audio features such as sound, music, voice overs that will enhance the learning tools and also help the children learn proper pronunciation.

Moreover, this system will provide drawing and coloring tools. There will be also quiz system so that children can learn and practice at the same time.

**Keywords:** web based learning application, graphical features, voice overs etc.

**Chapter 1**

**PROJECT INITITION**

## 1.1 Background to the problem

Now a days children gets more familier with gazettes than books. So if we can make our education system online based children will get more interest in their studies . We have come up with this thought to make an interactive online education application to remove the monotony from studies and make it more fun and attractive .

## 1.2 Objective of this project

This system is designed to assist kids in learning alphabets, words, numbers, tables, mathematical operations and much more. By adding graphical features such as both still and moving images, short animations it will be more visually attractive and helpful for kids to learn fast and make it more fun for them. There will also some audio features such as sound, music, voice overs that will enhance the learning tools and also help the children learn proper pronunciation.

## 1.3 SDLC methods

SDLC stands for Software Development Life Cycle. SDLC is a process to develop high qualified software. SDLC have six stages. The following figure is a graphical representation of the stages of a typical SDLC.

### 1.3.1 Phase of SDLC:



**Fig 1-1: Phase of SDLC**

Descriptions are given below:

1. **Planning:** This stage contains requirement analysis. This is the most important part of SDLC. In this stage senior members talk to the customer and collects the requirement of the software. Project related study, risk and revenue calculation are also done in this stage.

**2. Defining**: This is the documentation stage. After requirement analysis, requirements are clearly documented and approved from the customer. It’s called SRS (Software Requirement Specification). It contains all the product requirements.

**3. Designing**: Based on the requirements, more than one design is proposed. Designed are also documented. It’s called DDS (Design Document Specification). After analysis the DDS, the best design approach is selected for the product.

**4. Building**: In this stage programmer start programming with the chosen tools (programming language, compiler etc.). Usually tools are defined by the organization. In this stage actual development is start. The programmer implement the DDS. And the product is built.

**5. Testing**: In this stage tester try to find bugs and check if the software meets the requirements or not.

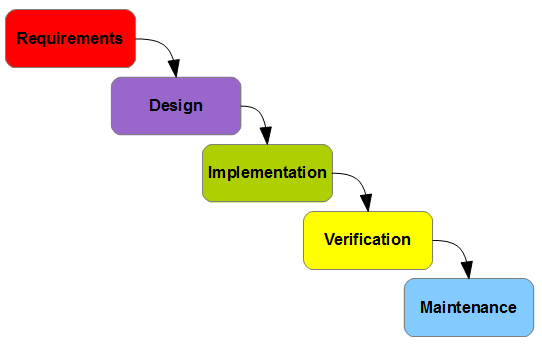
**6. Deployment**: After the product is tested then it’s ready to release. Release date depends on organization and market conditions. This stage is also contain maintenance. Based on feedback and customer demand, software maintenance happen.

### 1.3.2 SDLC Models:

There are different kinds of SDLC process model. Some are traditional and some agile and other.

Traditional Models:

**1.Waterfall :**



**Fig 1-2: Waterfall**

When all requirements are very clear then this model is chosen. The main thing is that when the working is started it in phase by phase there is no way to move back to the previous phases. Most of time this model is chosen for small or medium projects. When the requirements are clear and fixed there is no option to change it while tasks are processing on this model.

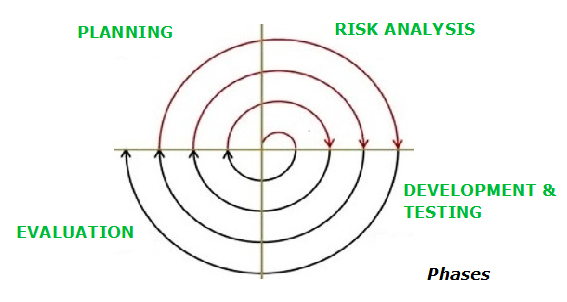
1. **Iterative Model:**



**fig : iterative model**

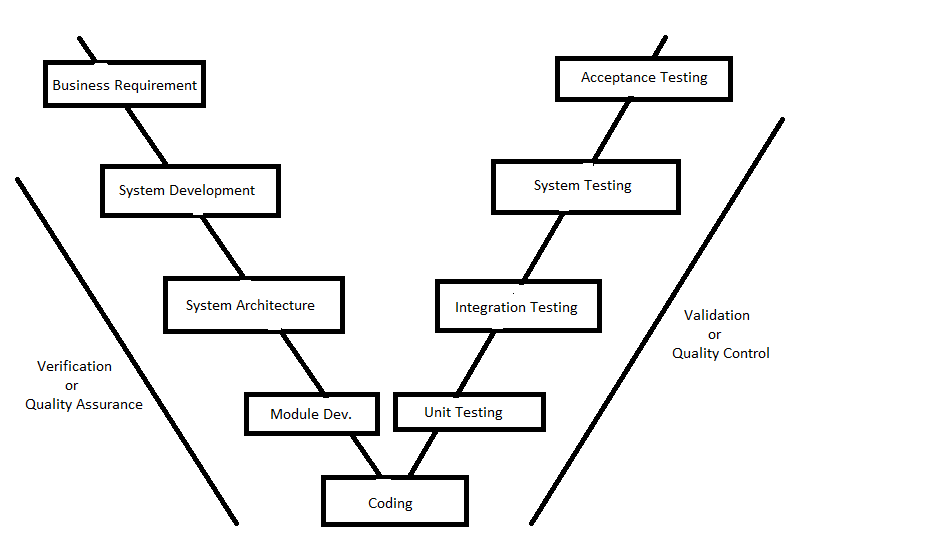
**I**terative process is start from implementation. It starts implement a subset of the software. This method is based on repeated cycle (iterative). The main plot of this method is to develop a system through repeated cycles (iterative) and in smaller portions at a time (incremental). Iterative process is an incremental process. More than one iteration may process at a time. Every iteration includes requirement analysis, design, development, testing and implementation. In this model risk can easily analyze. But this model is so complex. Skilled people need for management

1. **Spiral Model:**



This Spiral model is a combination of iterative development process model and sequential linear development model i.e. the waterfall model with a very high emphasis on risk analysis. It allows incremental releases of the product or incremental refinement through each iteration around the spiral. This model has four phases. A software project tasks repeatedly passes through these phases in iterations called Spirals. This one is mainly used for medium to high-risk projects, requirements are complex, and significant changes are expected in the product during the development cycle. Spiral model is used for big projects and when requirements are not stable.

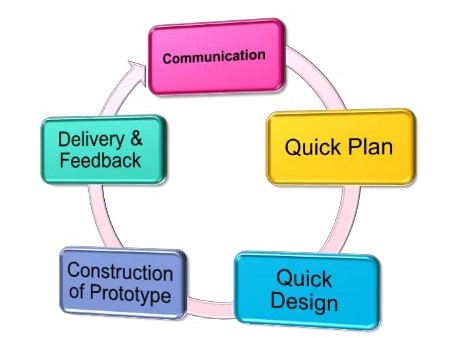
1. **V-Model :**



**fig : v-model**

This is also known as Verification and Validation Model. The V-Model is an extension of the waterfall model and is based on the association of a testing phase for each corresponding development stage. This means that for every single phase in the development cycle, there is a directly associated testing phase. Like waterfall model here requirements are well defined and fixed and there will be no undefined or ambiguous requirements. This is mainly used for short projects.

**5.Prototyping Model**: This model is mainly used for understanding the user requirements clearly. This one helps developer to understand what functionality and system look customer is expecting to build. In simple words, the prototyping refers to building software application prototypes which displays the functionality of the product under development, but does not hold the exact logic of the original software. Iteration occurs as the prototype is tuned to satisfy the needs of the customer. Prototyping model is used when the project is short.



**Fig 1-6: Prototyping Model**

**6.Agile Methods**: In Agile model, the tasks are divided to time boxes (small time frames) to deliver specific features for a release. These are done in iteration process. Each iteration process has time limit from 1 week to 4 weeks. Basically now-a-days this model is used in most of the projects. Mainly for big projects this one is used. This model gives flexibility to developers and here the resource requirements are minimum. But it will be hard to manage or processing the tasks if the project has complex dependencies



**fig : Agile model**

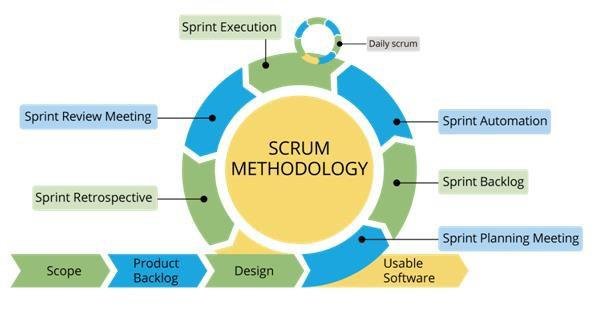
**1.Scrum:** Scrum have three phase.

These are: 1.1. Pre-game

1.2. Development

1.3. Post-game

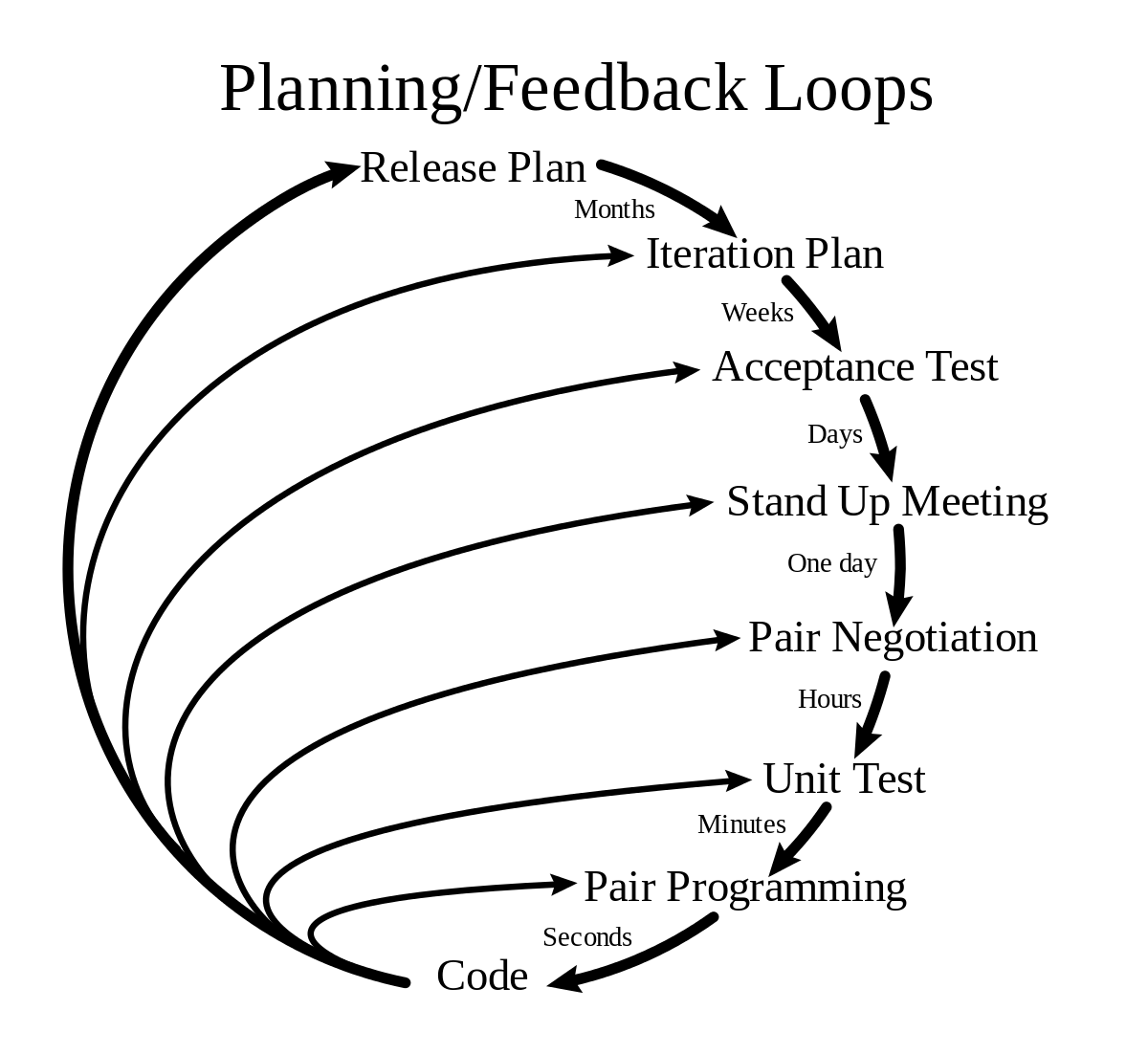
In pre-game phase planning and design are happen. It includes requirements analysis, risk calculation and other documentation part. And in development phase, project is built. Post-game phase come when there is no more requirement. And then the project is being released.



**fig : SCRUM**

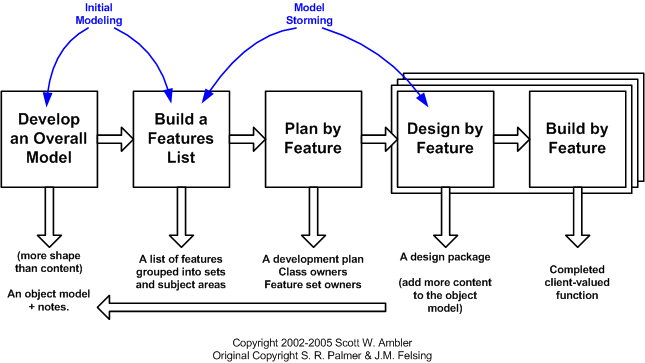
Product backlog and sprint are special features of scrum. Product backlog is a list of priority based requirements. And sprint is iteration cycle.

**2. Extreme Programming (XP):** Agile Modeling (AM) is a practices-based software process. In XP method, software may release after every iteration.



**fig : Extreme Programming**

1. **Feature Driven Development (FDD):** FDD is iterative and incremental software development process. FDD have short iteration process. FDD consist five basic activities.

****

**fig:FDD**

First two activities are the overall model of the project. The final three activities are iterated for each feature.

**Our Chosen SDLC model:** SCRUM

**Why we chosen SCRUM as our SDLC?**

● Scrum ensures effective use of time and money

● Large projects are divided into easily manageable sprint

● Works well for fast-moving development projects

● The team gets clear visibility through scrum meetings

● Scales very well to multiple teams and geographic locations.

● Short sprints enable changes based on feedback a lot more easily

● The individual effort of each team member is visible during daily scrum meetings

● Issues are identified well in advance through the daily meetings and hence can be resolved speedily

● It is easier to deliver a quality product in a scheduled time.

● It is good for small, fast moving projects as it works well only with small team.

**Chapter 2**

**REQUIREMENTS ELICITATION**

## 2.1 Product Perspective:

YOU LEARN is being is developed for children who are of age 3-6 years. It will provide a platform to the children to learn basic primary education through a web site .It is an open source project and it has a very active developer team to support it and provide feedback to users.

## 

## 2.2 Product Features:

* Powerful and supportive data base.
* Easy navigation
* Interactive Content
* Mock Test
* Game session
* Painting

## 

## 2.3 Operating Environment

The software will run in a hosting service named Xyz with the domain named youlearn.com and this domain can be accessed via any windows or android platform through any browser like Google Chrome, Mozilla, and Microsoft Edge.

## 

## 2.4 Design and Implementation Constraints:

## YOU LEARN can be run in any processing system like dual-core, quad-core, hexa-core, octa-core etc. Much higher configured system and well developed operating system will help to run this system more smoothly.

## 2.5 Assumptions and Dependencies:

There is a dependency of the software and it is,

A good structured and secured hosting service to store the data & files.

Good internet connection

**Chapter 3**

**REQUIREMENTS SPECIFICATION**

## 

## 3.1 USER DOCUMENTATION:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Serial** | **As a/an** | **I want to** | **So that** | **Acceptance Criteria** |
| 01 | As a admin | Know the result | I can know the industrious child | Sorted and showing highest mark obtained child info |
| 02 | As a admin | Know the result | I can inform the guardian of failed student | Lowest marks obtained child info. |
| 03 | As a admin | Know the student who scored more than passed score | I can promote children to the next class | Promoted successfully |
| 04 | As a admin | Upload marks | Mentors can see their child’s marks | Uploaded marks successfully |
| 05 | As a admin | Update marks | Marks can be updated if necessary | Updated marks successfully |
| 06 | As a admin | Upload question set | All the student can give test | Question uploaded successfully |
| 07 | As a admin | Accept mentor’s request  after registration | the student can use this website. | Student enrolled successfully |
| 08 | As a admin | Check student answer sheet | I can check their marks. | Question paper uploaded successfully |
| 09 | As a admin | Check student homework | I can know about their improvement | Checked student homework |
| 10 | As a mentor | Registration | Log in in the software | Log in successfully |
| 11 | As a mentor | Know test result | I can know children performance | Displayed result |
| 12 | As a mentor | Hear student pronunciation | I can know their spelling is right or not | Audio player played successfully |
| 13 | As a mentor | See sample question set | We can taught them | Sample set showed successfully |
| 14 | As a mentor | Know exam date | I can keep up to date with exam | Exam date showed successfully |
| 15 | As a mentor | Check child daily homework | I can know whether he is doing his/her daily work or not. | Show child daily homework |
| 16 | As a student | Do homework | I can complete my daily task | Daily homework notification |
| 17 | As a student | Give test | I can promote to next class | Yearly exam taken |
| 18 | As a student | Learn from this software | I can increase my primary knowledge | Showing all the learning option |
| 19 | As a student | Write in this software | I can practice | Writing dashboard displayed |
| 20 | As a student | Submit my homework | My mentor can see my daily work | Submit homework successfully |
| 21 | As a student | Know my marks | I can ensure I am doing right | Showed marked successfully |
| 22 | As a student | Recheck my marks | My marks can increase if I feel I was given less number by the software | Rechecked marks |
| 23 | As a student | pronounce | I can know my pronunciation is right or not | Sound recorded successfully |
| 24 | As a admin | Notified the student as well as mentor | They can know their performance or any kinds of notice | Notified successfully |
| 25 | As a student | I want to download all the notes | I can go through the notes | Download successfully |

## 3.2 User Interface

The user interface for the software shall be compatible to any browser such as Internet Explorer, Mozilla or Netscape Navigator by which user can access to the system.

The user interface shall be implemented using Microsoft C# .Net.

## 3.3 Hardware Interfaces

Since the application must run over the internet, all the hardware shall require to connect internet will be hardware interface for the system. As for e.g. Modem, WAN – LAN, Ethernet Cross-Cable.

## Software Interfaces

* The Youlearn website shall communicate with the student to get all kinds of learning instrument.
* The YouLearn website shall communicate with the mentor to get the student’s specifications and promotions.
* The YouLearn website shall communicate with the database to check student’s Performance.
* The YouLearn website shall communicate to admin for any kind of notice..
* The YouLearn website shall communicate with admin any things needed for Yearly exam or Monthly test.
* The YouLearn website shall communicate with the mentor for student’s registration.
* The YouLearn website shall communicate with the mentor for notifying student’s performance in the exam.
* The YouLearn website shall communicate with student for submitting homework after completing.

## 

## 3.5 System Features:

The major services and functional requirements for the product can be illustrated by system features. This section is organized by use cases for major system features. In the following, necessary description is provided for each use cases in the system. Each use case description provides information of the associated actors, triggering condition, preconditions, post conditions, response sequences, exceptions and functional requirements (assumptions). Being a major important section of the SRS, this section is expected to go through iterative improvement to make the most logical sense for the intended product.

### 3.5.1 Mentor’s Registration

**Descriptoin and Priority**

First of all, Mentor needs to be log in as the student can’t be registered as their own. So, the mentor will first registered themselves and then they’ll registered student. After admin approval, admin get access student’s information.

**Stimulus and Response Sequences**

* Mentor registered their student by log in this website.
* Student request approval
* Get notified when homework uploaded

**Functional Requirements**

Mentors can see student registration page at first. After admin approval, mentor can view student information alongside their homework when it will be finished and it will be uploaded by student.

**Enter the site**

* For using this website user have to enter this site. One should have a clear idea how to use this application and what purpose it is served. One can use this website for learning purpose.

**Student needs to be capable of devices for proper use**

* For using this site, user should have the idea how to use his/her cellphone. User should know how to use this software.

### 3.5.2 Homework

**Description and Priority**

After student’s acceptance approval, an interface will be shown to student which will contain all the subject information. After clicking any subject, all the subject information will be provided through website.

**Stimulus and Response Sequences**

* Choose any course
* 4 skills of learning will be provided
* Record when reading and speaking

**Functional Requirements**

Student will see a list of courses in the dashboard accordingly to their class. Then they have to choose which they’ll learn on that particular day. There will be 4 types of learning skills provided in our website.

**Listening**

* For listening, the system will play a recorded file of story or a letter or a word. The system will also provide an image according to the letter or word played. Students have to choose the image. Then student have to upload in upload section.

**Reading**

* For reading, the system will provided a story or an image of any letter/word, which student have to record and uploaded in record section for reviewing the recorded file. It will notify the mentors that today’s homework is finished and uploaded successfully.

**Speaking**

* For speaking, the system will generate a letter or word. Then student have to pronounce the letter or word and record the pronunciation and uploaded in record section.

**Writing**

* For writing, the system will provide a letter or a word. Students have to write the letter or the word in the answer script after the picture shown.

### 3.5.3 Yearly Exam

**Description and Priority**

In this system, there will be two exams in a year known as Midyear exam and Final exam. The exam notice will be provided one month before the exam. All the students have to give on a proper date and time.

**Stimulus and Response Sequences**

* View exam schedule
* Exam syllabus
* View Question Paper
* Upload answer script

**Functional Requirements**

At the midyear and end of the year there will be an examination for student. On the performance of the exam student will promote to next class. And the failure student will remain in the class.

**View Exam Date**

* A month before the exam, the examination date along all the information including exam syllabus will be provided through notice. On the due date students have to sit for exam.

**View sample question set**

* A week before the exam, a sample question set will be provided in our website. It can only accessible by mentor. It will contain what kinds of question will be asked in the exam.

**Write Answer**

* An answer script alongside question will be provided on examination date. Students have to answer in this answer script. Student can write, erase and draw figure in script. The starting time will be counted after viewing the question paper and the students have to finish writing before due time.

**Upload Answer script**

* After completing the exam, students have to upload answer script in the uploaded section. Once uploaded one can’t rewrite or can’t give exam for the second time

### 3.5.4 Upload and Update result

**Description and Priority**

After uploading the answer script, the result will be published in a short time. All the students who scored more than passing grades, will promote to next class. The failure students will remain in the class.

**Stimulus and Response sequences**

* Exam Details
* View Answer Script
* Get Notify

**Functional Requirements**

After uploading the result, mentors will get notified by our website. The mentors can see student’s result as well as answer script.

**View answer script**

* After checking all the answer script, result will be provided through our website. Mentors can see student’s result as well as answer script. Student also can see their marks.

**Get Notify**

* After uploading, mentors will get a notification for result publication. Thus mentor can know student’s marks. Any kinds of notice including exam schedule, result etc., the system will notify all the mentors.

**Request Admin**

* If any mentors find that any student didn’t scored as his/her performance, mentors can request admin for rechecking students answer script. Then admin again will check the answer and if he finds the marks will increase, he can update the marks.

## 3.6 Other Nonfunctional Requirements

### 3.6.1  Performance Requirements

The system must be interactive and the delays involved must be less .So in every action-response of the system, there are no immediate delays. In case of opening windows forms, of popping error messages and saving the settings or sessions there is delay much below 2 seconds, In case of opening databases, sorting questions and evaluation there are no delays and the operation is performed in less than 2 seconds for opening ,sorting, computing, posting > 95% of the files. Also when connecting to the server the delay is based editing on the distance of the 2 systems and the configuration between them so there is high probability that there will be or not a successful connection in less than 20 seconds for sake of good communication.

### 

### 3.6.2    Safety Requirements:

YouLearnis 100% safe to use. Besides, itis an open-source project, so anyone who has doubts is able to check the source code.

### 3.6.3 Security Requirements

This program uses object oriented mechanisms to protect its data passed using methods also there is no currently a security schema of this program.

# 3.7     Software Quality Attributes

**Availability:**If the internet service gets disrupted while sending information to the server, the information can be send again for verification.

**Security:**The main security concern is for users account hence proper login mechanism should be used to avoid hacking. The tablet id registration is way to spam check for increasing the security. Hence, security is provided from unwanted use of recognition software.

**Usability:**As the system is easy to handle and navigates in the most expected way with no delays. In that case the system program reacts accordingly and transverses quickly between its states

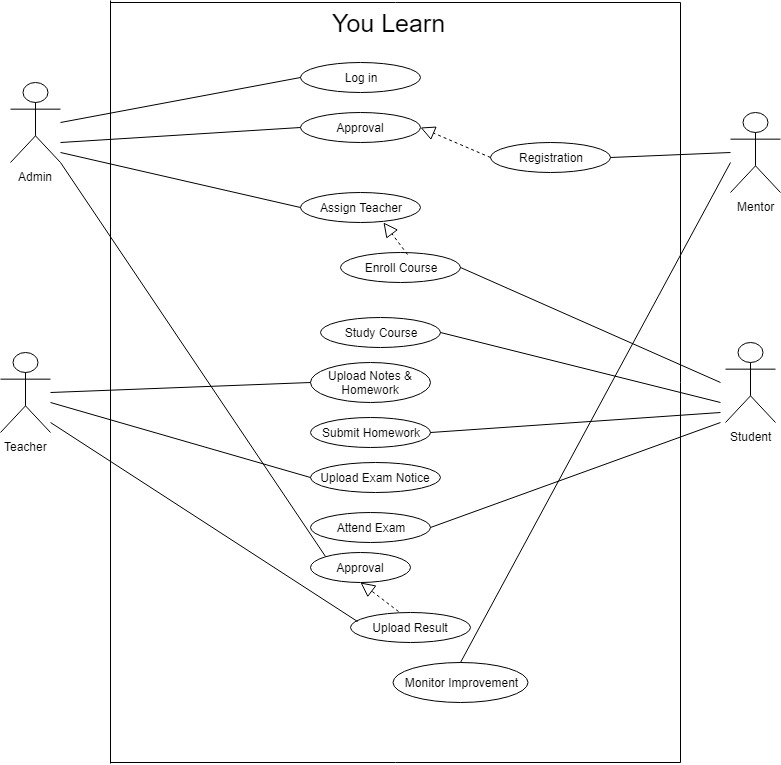
# 3.8   Business Rules

We want to implement a Online Learning System which reduce the time of being teaching by the parent. Also student can know details about primary knowledge. The parent does not have to give time instead they will check their child’s performance. It will reduce the time that parent have to give. Also we are thinking to provide a new way learning for the children so they can find a easier way to learn. Also when a person who will be browsing in the website,the Owner of the system can earn form the ads by google.

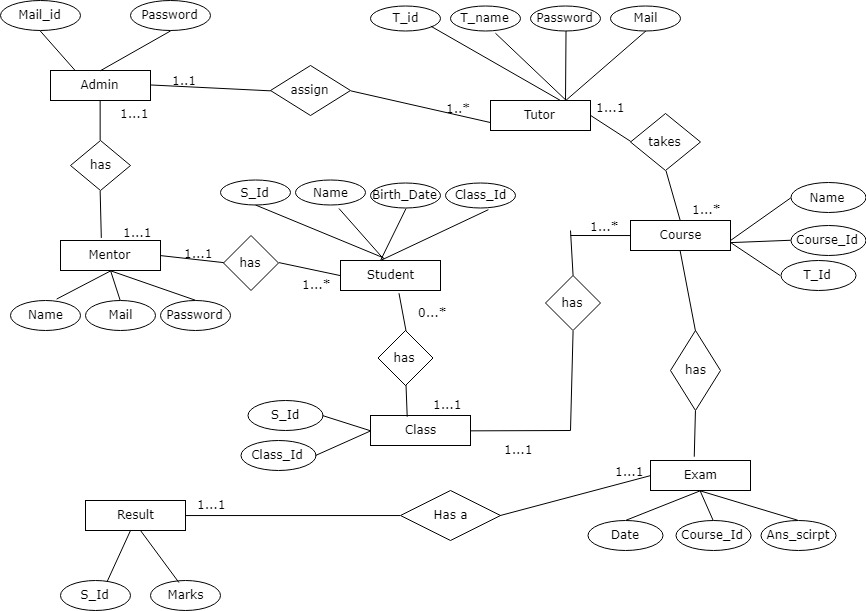
**Chapter 4**

**SOFTWARE DESIGN SPECIFICATION**

# 4.1 Use Case Diagram

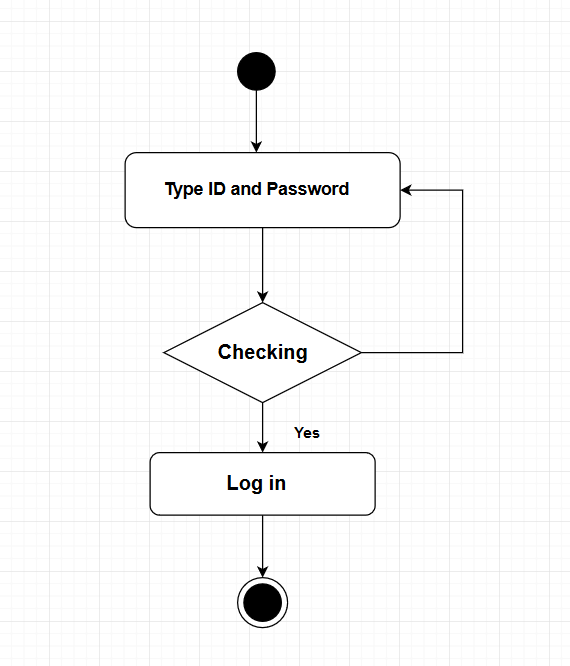
****

# 4.2 ER Diagram:

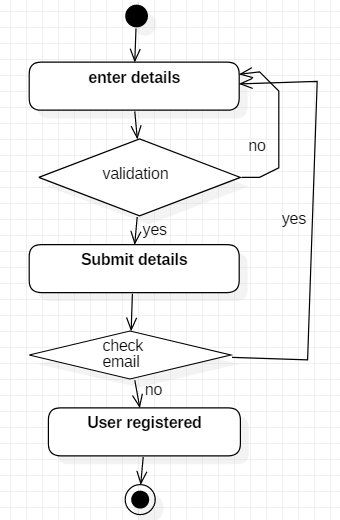
****

# 4.3 Activity Diagram:

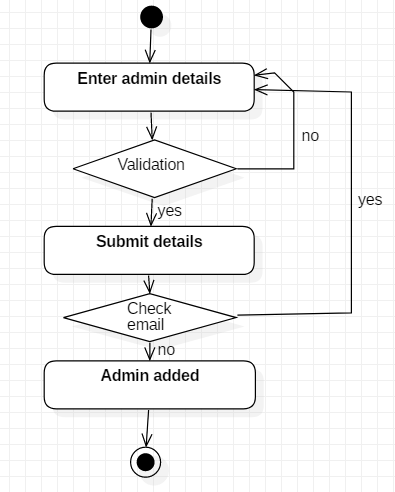
## 4.3.1 LOG IN :

****

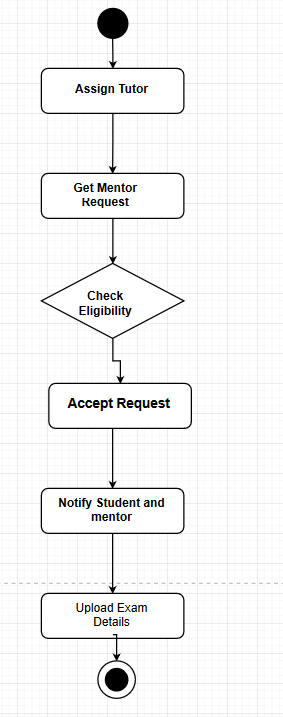
## 4.3.2 Registration:

****

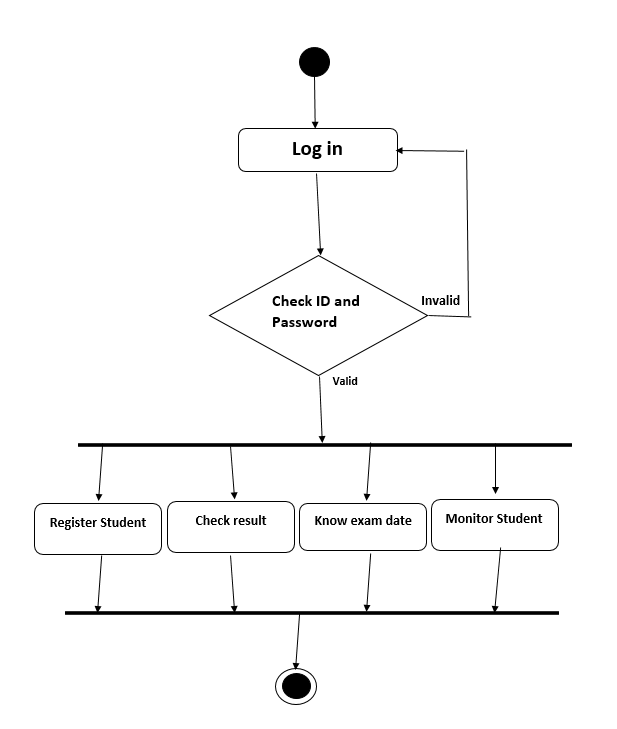
## 4.3.3 Add Admin :

****

## 4.3.4: Add tutor :

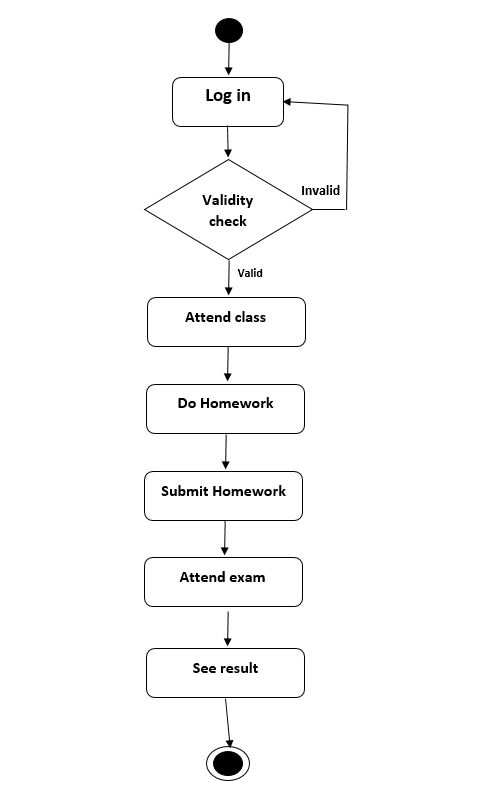
****

## 4.3.5 Mentor’s activity :

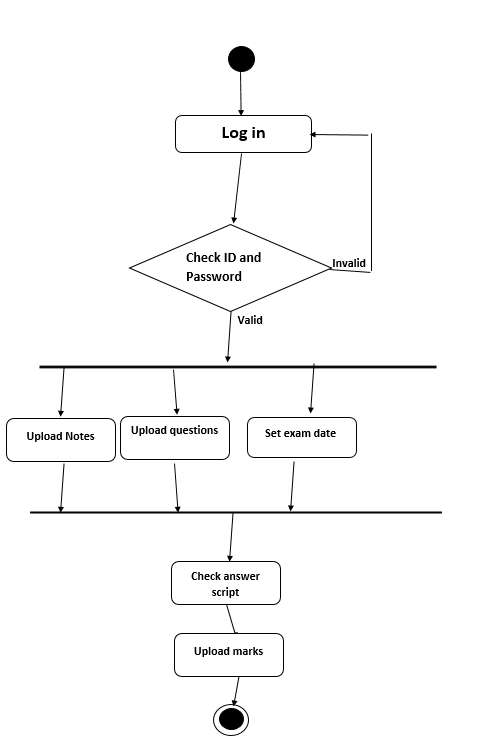
****

## 

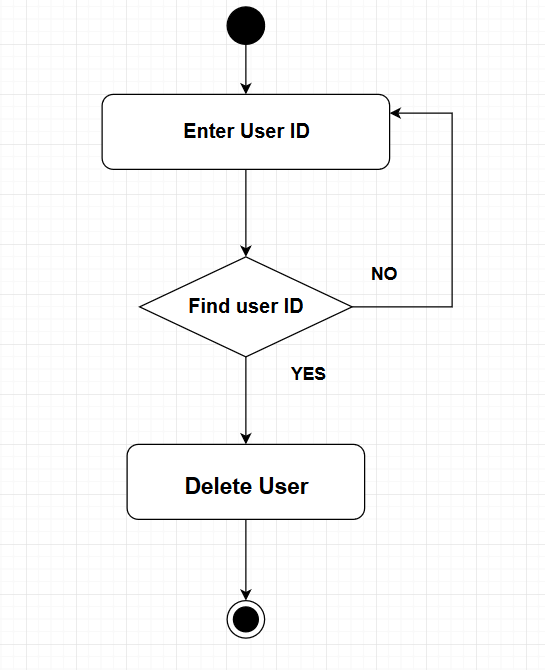
## 4.3.6 Student’s Activity :

****

## 4.3.7: Teacher’s Activity :

****

## 4.3.8 Remove User :

****

**Chapter 5**

**REFERENCE**

1. <https://www.khanacademy.org/>
2. <https://www.w3schools.com/>
3. https://www.udemy.com/
4. <https://www.codecademy.com/>
5. [https://www.coursera.org](https://www.coursera.org/)
6. https://join.skillshare.com/
7. https://www.lynda.com/
8. <https://www.bloc.io/>
9. https://www.udacity.com/

**Chapter 6**

**CONCLUSION**

This project has been implemented specially for Primary level students to introduce online learning system . Students often face difficulties to carry books and notes. With the system, students do not need any books or notes because everything is online. Besides, parent does not have to give time for their child. Because the system is designed in an attractive way so that students will find it helpful. And furthermore we think “YouLearn” can replace the old learning system that exist in our education system. It is a matter of joy and honor that we have got the opportunity to work with the project and have implemented the topic.