September 14, 2015



**INTERNATIONAL TECHNOLOGICAL UNIVERSITY**

**SWE-690 CAPSTONE PROJECT- FALL-2015**

**SOFTWARE REQUIREMENT SPECIFICATION AND**

**HIGH LEVEL ARCHITECTURE DOCUMENT**

**EMPOWER STUDENTS SKILLS BY SERVICES**

**(ES3)**

**Guided By:**

Professor: Hann So

**Submitted By:**

Desam Syama Sundara Reddy (#86287)

Vijay Krishna Vemuri (#87799)

Dabeeth Nagam (#87873)

Naga Sankara Manoj Hota (#87750)

1. **Introduction**

1.1 Methodology

1.2 Purpose

1.3 Scope

1.4 Definitions, Acronyms and Abbreviations

1.5 Tools and Technologies to be used

1.6 References

1.7 Overview

2. **Overall Description**

2.1 Product Prospective

2.2 Software Interface

2.3 Hardware Interface

2.4 Communication Interface

2.5 Constrains

2.6 ER Diagram

2.7 Use-Case Model Survey

2.8 Architecture Design

2.9 Database Design

3. **Specific Requirements**

3.1 Use-Case Reports

3.2 Activity Diagram

3.3 Sequence Diagrams

**1. Introduction**

**1.1 Methodology**

**RUP (Rational Unified Process)**

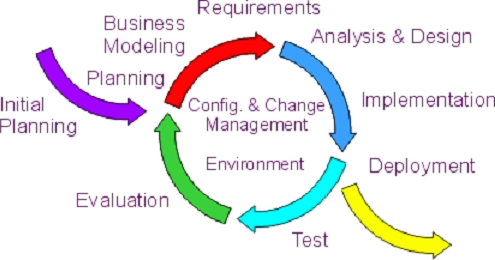
This project would practice Rational Unified Process (RUP), which is a software engineering process that provides disciplined approach towards project implementation. RUP provides guidelines for successful implementation of a project like **Iterative Development** which helps the software requirement specification (SRS) to evolve throughout the development cycle without upsetting the cost of development, **Manage Requirements** which describes methods about documenting functionality and constraints, estimate and plan for alternatives, easily capture and communicate the business requirements, **Verify Quality** to ensure reliable end product by assessing quality in all process and activities, **Control Changes** which helps to track, monitor a change and assess its affect so that there are no roadblocks to the iterative development.

Fig 1.1 Phases of RUP

The RUP process consists of the following four consecutive phases for a development life cycle:-

**Inception:**

In the Inception phase business case is developed to set the boundaries for project scope. Major activities like estimate of resources, initial risk assessment and financial forecast are carried out during this phase. The outcome of this phase is a project plan summary depicting other phases and iterations and a vision document with project’s core requirements and key features.

**Elaboration:**

The Elaboration phase mainly focuses on analyzing the problem, architecture and eradication of highest risk elements in the project. In this phase a prototype of executable software architecture is created depending upon the project scope. The outcome of this phase is use case model, prototype of executable architecture and a development plan for the project.

**Construction**:

In the Construction phase all components needed for the application are developed and integrated. Testing of the components is thoroughly carried out to ensure quality. Construction phase consists of development, which may be sequential or parallel depending upon the need and resource allocation. The outcome of this phase consists of software product, which works on desired functionality, description of the current release and user manuals.

**Transition:**

In the Transition phase consists of beta testing with bug fixes and enhancements. During this phase focus is placed on releasing end product to the user community with documentation to achieve user self-supportability. Training is provided to the users on the usage of the application and parallel operation is carried along with the legacy system (if any).

1.2 **PURPOSE**

‘Empower Students skills by Services’ (ES3) is an online web application to provide a platform for students who wants to improve skills by delivering free lectures for enthusiastic enrolled learners.

The purpose of this project is to help people who lack resources but want to learn advanced technical and communication skills. As one of the oldest English proverb which says ‘Give a man a fish and you feed him for a day. Teach a man to fish and you feed him for a lifetime’, this application helps people to enroll for classes which are taught by the students of ITU. Even though the classes are free of cost, donations for expanding ITU library are highly appreciated. The ES3 application would have the capabilities, which helps interested people to view the upcoming class timetable, register or subscribe for the newsletters, enroll online, cancel enrollment, send reminders for enrolled classes, donate for ITU library and leave feedback. From an ITU student point of view the application possesses options to view the class schedule, number of enrolled students, feedback/comments and number of classes taught. From an admin perspective the applications would have the options to post or cancel the class schedules, monitor login mechanisms, view number of registered students, student contact details, add or edit payment gateway mechanisms for donation.

This application would have the potential for further expansion to support registration of ITU presents. The first phase of this project may involve gathering requirements from college staff, students or other sources, which would help in design. The design would followed by development, QA, Production and documentation phases, thus completing an SDLC life cycle.

As ITU’s mission has always been to empower people and advance global prosperity through inventive, industry-linked Silicon Valley education, this project would not only allow the students to showcase their skills but also help people who are need for advanced technical, communication skills.

1.3 **SCOPE**

* There are three basic users in this project **Admin, Student, User**.
* All the users have their own customized web page.
* User can search for the required course he/she willing to learn and see the student who teaches it.
* Based up on the convenient time they can set up a meeting and exchange their knowledge.
* Admin can monitor all the activities going on, he/she has the permission to authenticate the user and student.
* Admin has the permission to add/remove courses.

1.4 **DEFINITIONS ACRONYMS AND ABBREVIATIONS.**

ES3

Empower Student skills by Services: It is an application, which provides course classes to users, In the mean while empowers the student’s presentation skills.

Admin

Administrator: He/she has the permission to access the number of students and users, and has the permission to modify the number of people and classes being organized.

UML

Unified Modeling Language: It is a standard visual modeling language used for analysis, design and implementation of a software-based system.

HTTPS:

Hyper Text Transfer Protocol Secure: It is a protocol for secure connection over a computer network.

Oracle DBA

Data Base Administrator:

JavaScript:

A scripting language for computers often runs in web browsers to create a dynamic content.

HTML:

Hyper Text Markup Language: It is a standard markup language used to create webpages

CSS

Cascading style sheets: It is a style sheet language used for describing the presentation of the document written in markup language.

1.5 **TOOLS AND TECHNOLOGIES TO BEUSED**

Application Architecture –JAVA, J2EE

Java was introduced by Sun Microsystems in 1995. Java is an Object Oriented Programming and Platform independent language. Java can use to create complete applications that may run on standalone computer or can be distributed among servers and clients in network. J2EE is a java centric environment for developing, building and deploying Web-based enterprise applications. It consists of Application Programming Interfaces, Services and Protocols, which will provide the functionality for developing web-based application.

The application ‘Empower Students skills by Services’ (ES3) is a dynamic web application and is being planned to build with spring MVC (Model, View and Controller), database persistency, with Hibernate Object Relational Mapping (ORM) tool and with RESTFUL Web services. ‘Oracle Web logic’ will be application server to host the application. Junit unit test, TestNG and Selenium Automation testing along with manual testing is required for complete testing of the product. The following complete list of technologies are required to make the application into production:-

Frontend (User Interface).

(a) HTML/JSP

(b) Java Script

(c) CSS

Backend (Server Side Programming).

(a) Java 1.7

(b) Spring MVC

(c) Hibernate- Object Relation Mapping

(d) RESTFUL-web services

Testing

(a) Junit

(b) Test NG

(c) Selenium Auto Testing

Database

Oracle 11g R2

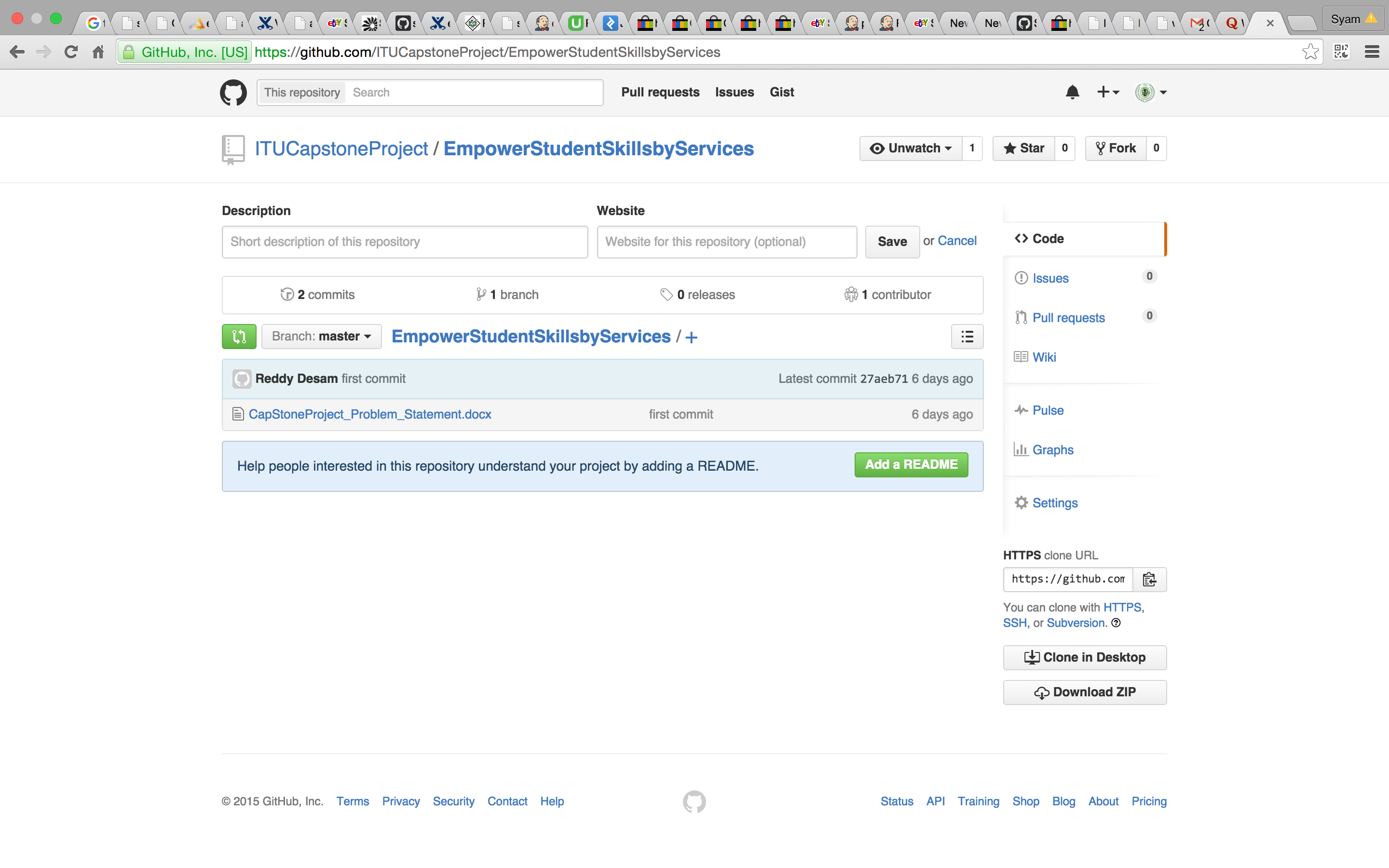
Application Server

Oracle WebLogic

Version Control System.

Version Control is a System that records changes to a file/source code over time so that any version of file/source code can be recalled. Software source code is being version controlled by ‘git’. The following git repository has been created so that the team members do not have any dependency while developing source code.

**https://github.com/ITUCapstoneProject/EmpowerStudentSkillsbyServices**

****

**Build and Deployment:**

Fig 1.2 Version Control System-Git

Building the application using Maven build tool and deployment the application on environments such as QA and Production.

**Project Architecture (Spring MVC Architecture).** The user request (JSP or Html pages) is handled by central servlet is called ‘Dispatcher servlet’ and it is further dispatches to controller where the beans are created and process to Business Layer of BO and DAO where java based ORM tool ie Hibernate that provides framework for mapping application domain objects to the relational database tables and vice versa. Further, the communication for web services’ is through over HTTP protocol and here also the databases is mapped with Hibernate framework for backend persistency.

**2. OVERALL DESCRIPTION**

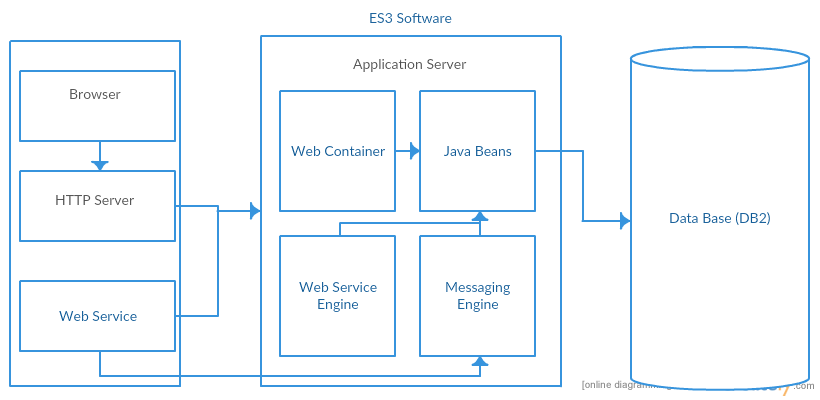
2.1 Product Perspective

Fig 2.1 Product Perspective

**2.2 Software Interface:**

* Client on Internet:

Web Browser, operating system

* Client on Intranet:

Web browser, operating system

* Web server:

WASCE, operating system

* Database server:

DB2, operating system

* Development End:

Eclipse Luna (Java, J2EE, java Bean, servlets, HTML5, XML, AJAX), DB2, OS (Windows), Web Logic (Web server)

**2.3 Hardware Interface:**

Minimum Requirements:

|  |  |  |  |
| --- | --- | --- | --- |
| Client Side | | | |
|  | Processor | RAM | Disk Space |
| IE 8, Chrome, Firefox | Intel Pentium iv | 512 MB | 150 MB |

|  |  |  |  |
| --- | --- | --- | --- |
| Server Side | | | |
|  | Processor | RAM | Disk Space |
| Eclipse Luna | Intel Pentium iv | 1 GB | 3.5 GB |
| DB2-9.5 | Intel Pentium iv | 512 MB | 500 MB |

Recommended Requirements:

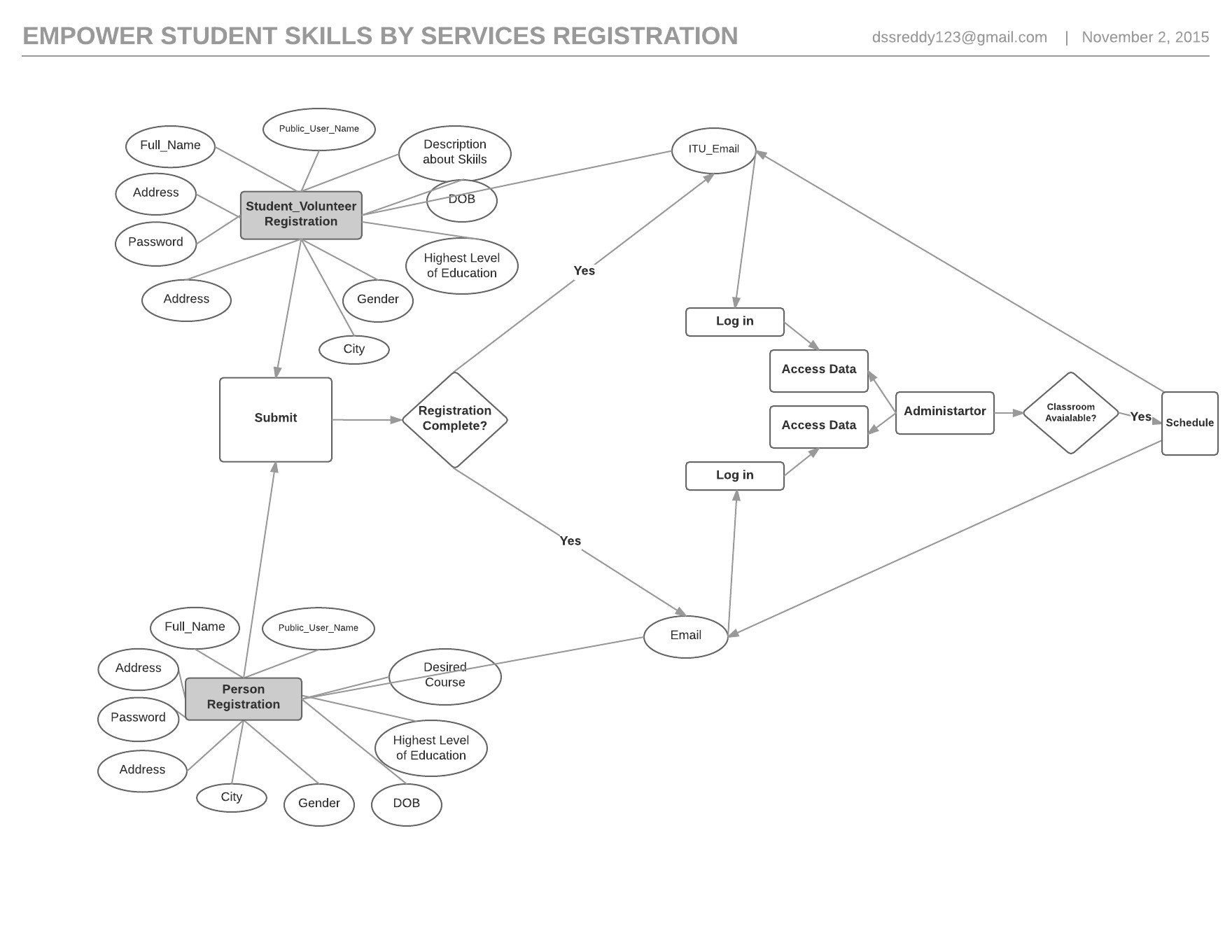
|  |  |  |  |
| --- | --- | --- | --- |
| Client Side | | | |
|  | Processor | RAM | Disk Space |
| IE 9, Chrome, Firefox | Intel core2duo or more | 2 GB | 150 MB |
| Server Side | | | |
|  | Processor | RAM | Disk Space |
| Eclipse Luna | Intel 3rd gen (2.4 GHz or more) | 2 GB | 3.5 GB |
| DB2-9.5 | Intel 3rd gen (2.4 GHz or more) | 2 GB | 500 MB |

**2.4 Communication Interface:**

* Student on Internet will be using HTTP/HTTPS protocols only.
* User on Internet will be using HTTP/HTTPS protocol.
* Admin on the Internet will be using HTTP/HTTPS protocol.

**2.5 Constraints:**

* GUI is limited to English language only.
* Login and password is necessary for all the users, students and the admin.
* Only ITU registered students and users can avail this service.
* This system is limited to single server.

2.6 **ER Diagram**

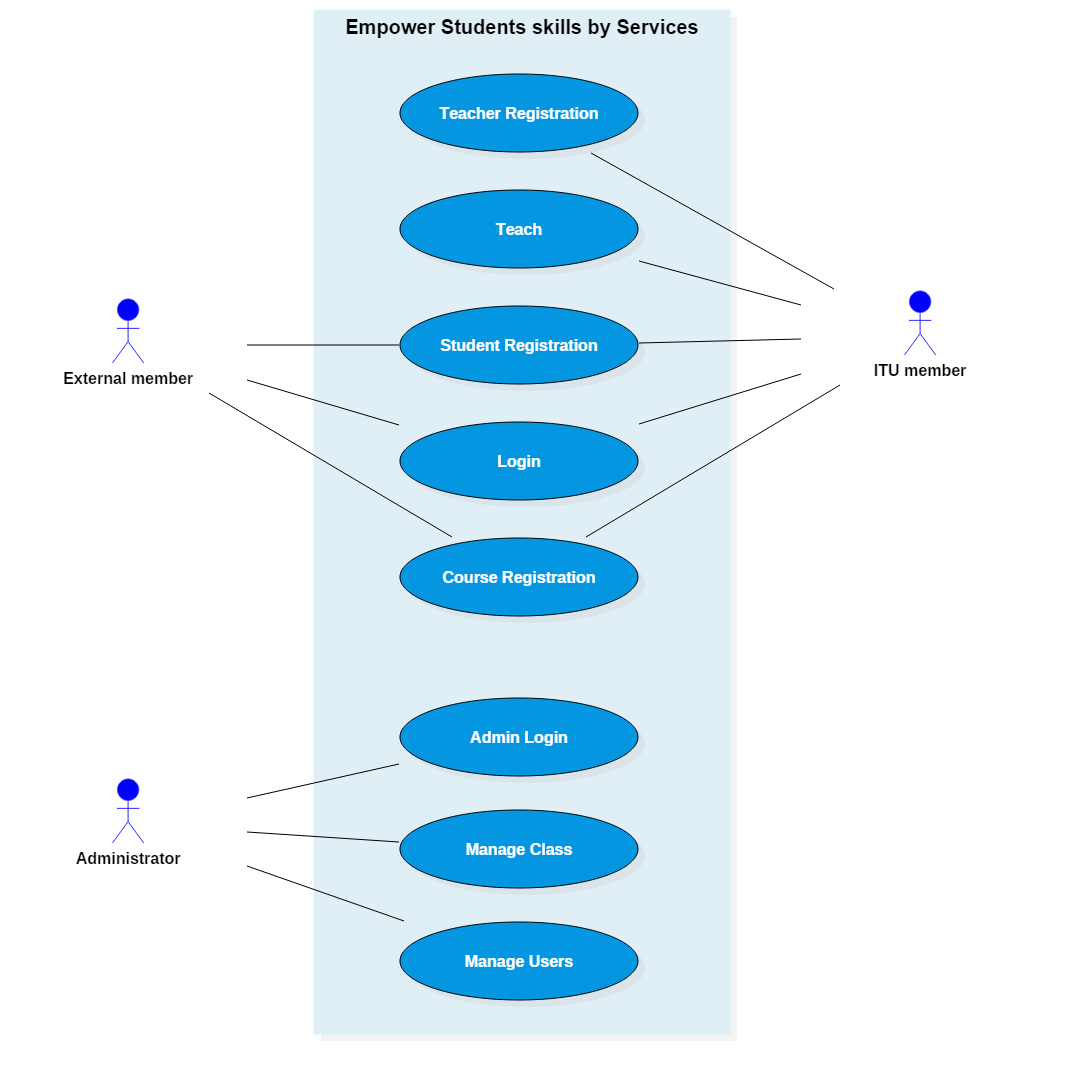
**2.7** **Use Case Model Survey**

Fig 2.2 ER Diagram

Fig 2.3 Use Case Model Survey Diagram

**Student:**

Students can fill registration form for skills sets and can propose for course teaching. Students can view the learners who are all registered course and approved by admin.

**Course Learners.**

Course Learners can search for all courses and can register for their desired course.

**Admin.**

Admin can view all profiles of students and course learners. Admin can co-ordinate the courses and will inform to course learners and students. Subsequently, admin will arrange classroom for conducting class.

**2.8 ARCHITECTURE DIAGRAM**

VO

Spring Controller

Dispatcher Servlet

BO

DAO(Hibernate)

JSP

Hibernate

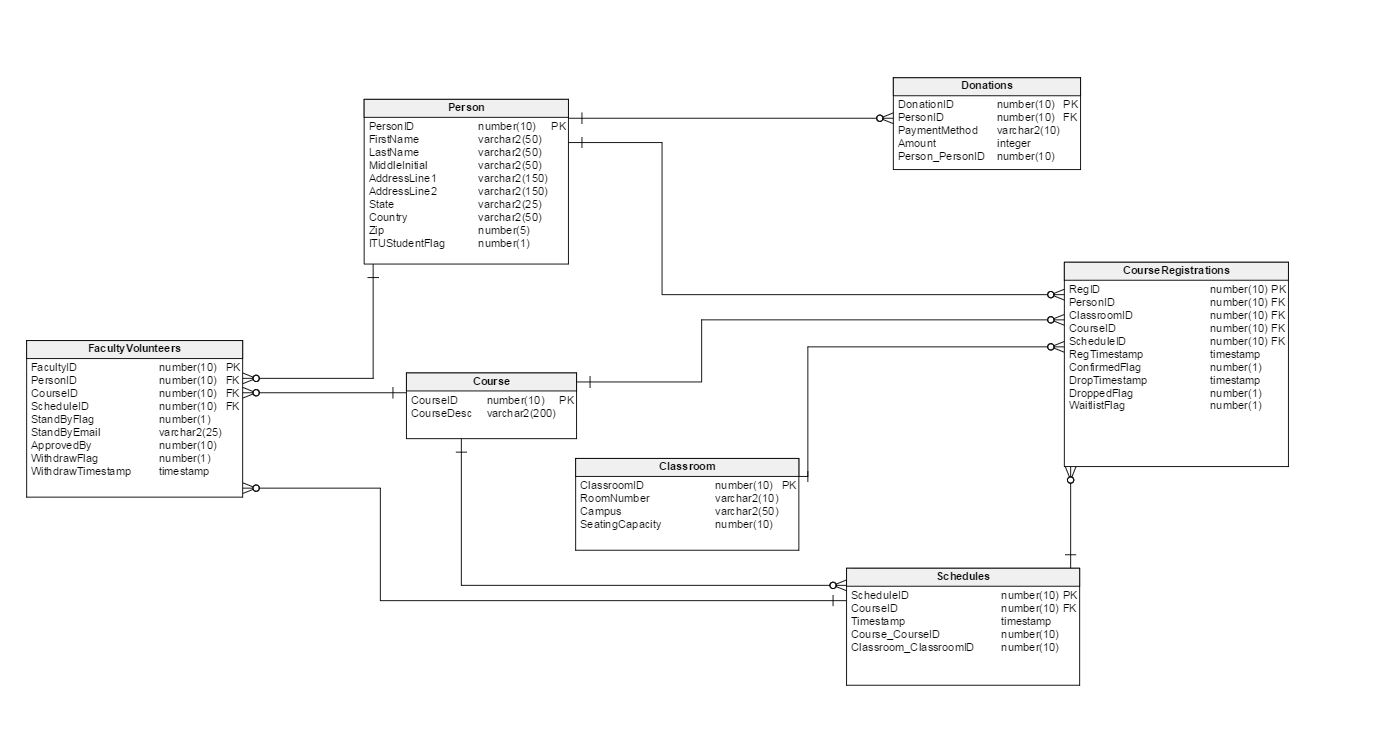
db

RESTFUL Web services

db

Hibernate

Fig 2.4 Architecture Diagram

**2.9** **Database Design**

**3. SPECIFIC REQUIREMENTS**

Fig 2.5 Database Design

Fig 2.4 Architecture Diagram

**3.1 Use Case Reports**

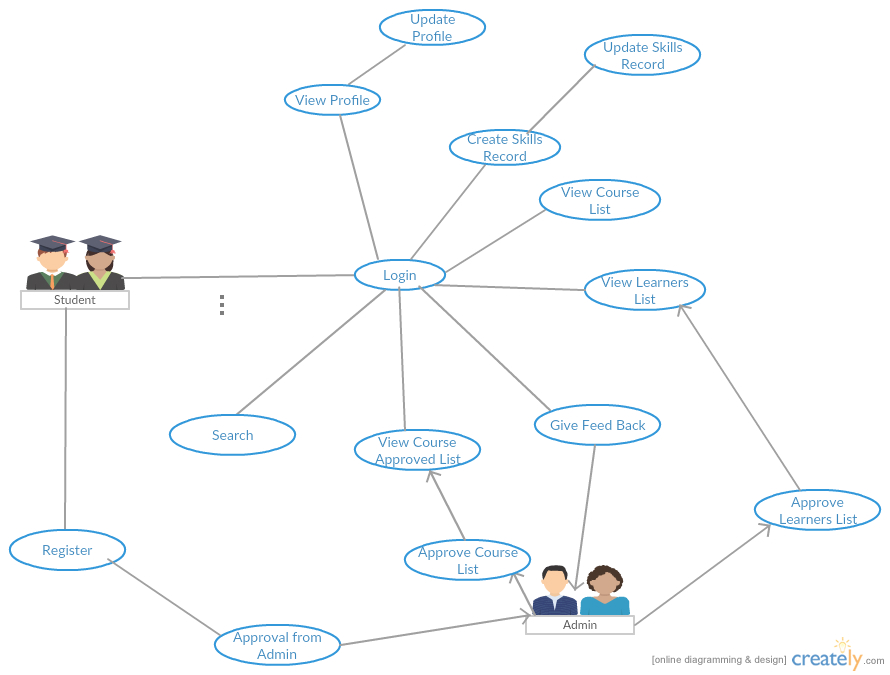
**3.1.1** **Student Use Case Report**

Fig 3.1 Use Case diagram for Student

|  |  |
| --- | --- |
| **USE CASE** | **DESCRIPTION** |
| Register | ITU Student registers into ES3 application |
| Approval from Admin | ITU Student gets approval from Admin |
| Login | ITU Student logs into ES3 application |
| View Profile | ITU Student can view his profile details |
| Update Profile | ITU Student can update his profile details |
| Create Skills Record | ITU Student provides a list of his skills |
| Update Skills Record | ITU Student can update his skills list |
| View Course List | See the list of courses |
| View Learners List | View list of enrolled learners for the course |
| View Course Approved List | View the list of approved courses |
| Give Feedback | Provide feedback on ES3 application |
| Approve Course List | Get approval for courses |
| Approve Learners List | Get approved learners list from Admin |

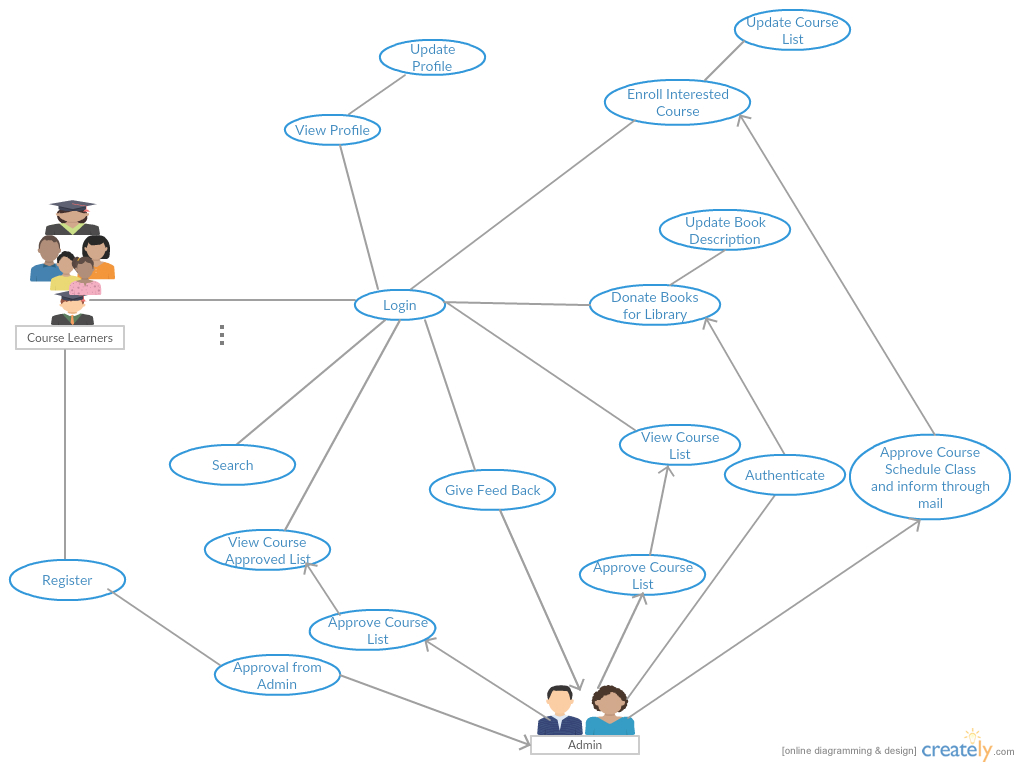
**3.1.2** **Learners Use Case Report**

Fig 3.2 Use Case diagram for Course Learners

|  |  |
| --- | --- |
| **USE CASE** | **DESCRIPTION** |
| Register | Course learner registers into ES3 application |
| Approval from Admin | Course learner gets approval from Admin |
| Login | Course learner logs into ES3 application |
| View Profile | Course learner can view his profile details |
| Update Profile | Course learner can update his profile details |
| Search | Search for available courses |
| Give Feedback | Provide feedback on ES3 application |
| View Course List | See the list of courses |
| Approve Course List | Get approval for courses |
| View Course Approved List | View the list of approved courses |
| Enroll Interested Course | Course Learner can enroll for an interested course |
| Update Course List | Update already registered courses |
| Approve Course Schedule class and inform through email | Course approval and class schedule details are emailed |
| Donate Books for library | Course Learners can donate books for ITU library |
| Authenticate | Course Learner is authenticated by Admin before donating |
| Update Book Description | Course Learners can update the book description for the books donated |

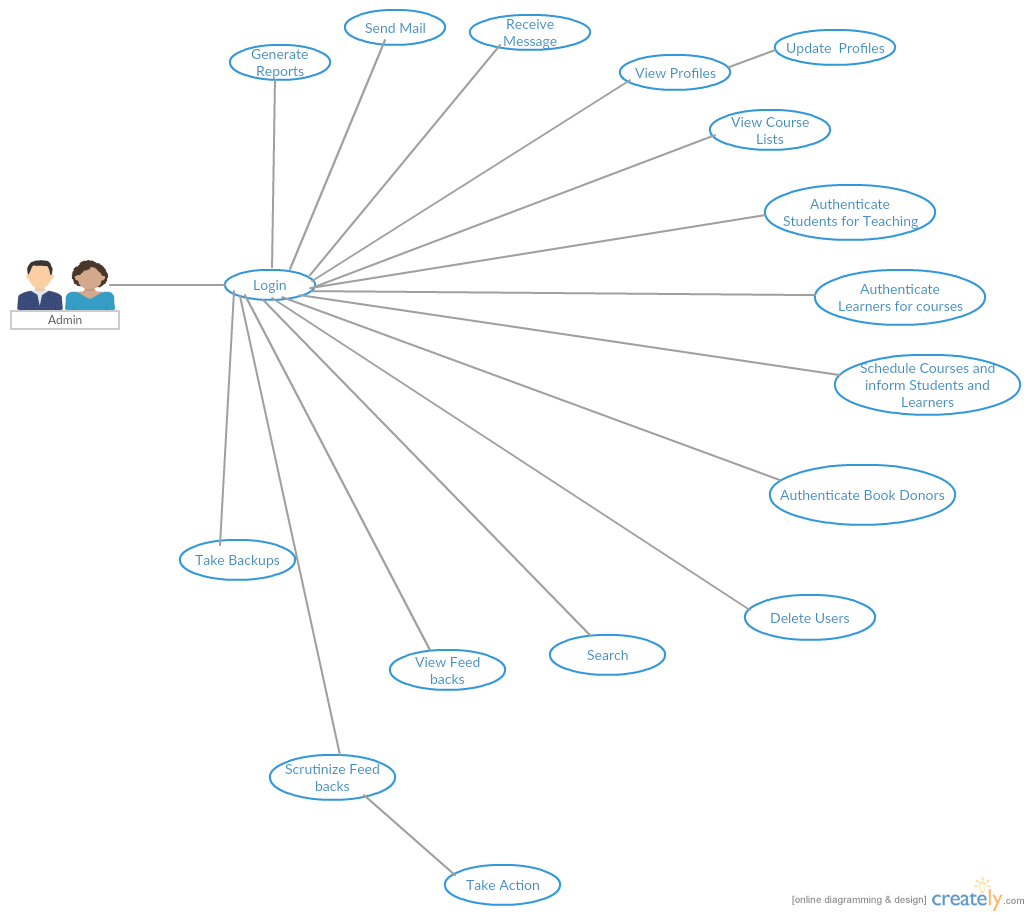
**3.1.3** **Admin Use Case Rep**

Fig 3.3 Use Case diagram for Admin

|  |  |
| --- | --- |
| **USE CASE** | **DESCRIPTION** |
| Generate Reports | Generate all reports |
| Send Mail | Send confirmation email to Students and Course Learners |
| View Profiles | View profiles of Students and Course Learners |
| Update Profiles | Update profiles of Students and Course Learners |
| View Course Lists | View list all available courses |
| Authenticate Students for Teaching | Authenticate enrolled students |
| Authenticate Learners for courses | Authenticate enrolled course learners |
| Schedule Courses and inform Students and Learners | Take action and sent mails accordingly |
| Authenticate Book Donors | Authenticate book donors before accepting donation for ITU Library |
| Delete Users | Delete Users (Students or Course Learners) |
| Search | Search for a specified course |
| View Feedbacks | View Feedbacks from Students or Course Learners |
| Scrutinize Feedbacks | Scrutinize Feedbacks to get rid of irrelevant feedbacks |
| Take Action | Take action based on the provided feedbacks |

3.2 **Activity Diagrams**

3.2.1 **User Registration Activity**

Initially User is required to fill all mandatory fields in Registration Form. Once Students/Course learners. When the users click submit button, the user name is verified. If the username is already exists, and then the user is again need to fill with new one. If the username validation is passed then password will be checked and finally check for all mandatory fields are filled or not. If the mandatory fields are left blank and filled in correct then the user will be informed to enter the correct values. Once all these verifications are succeeded, then the registration is process is completed.

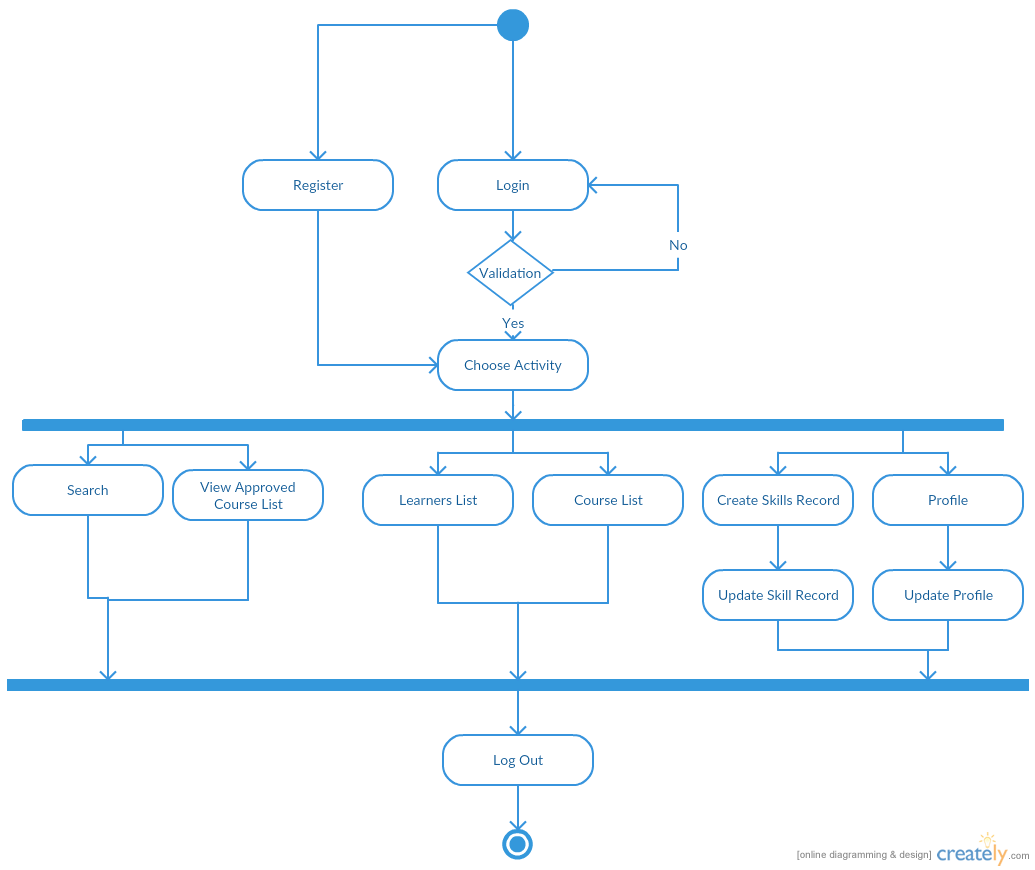


Fig 3.4 Student Registration Activity

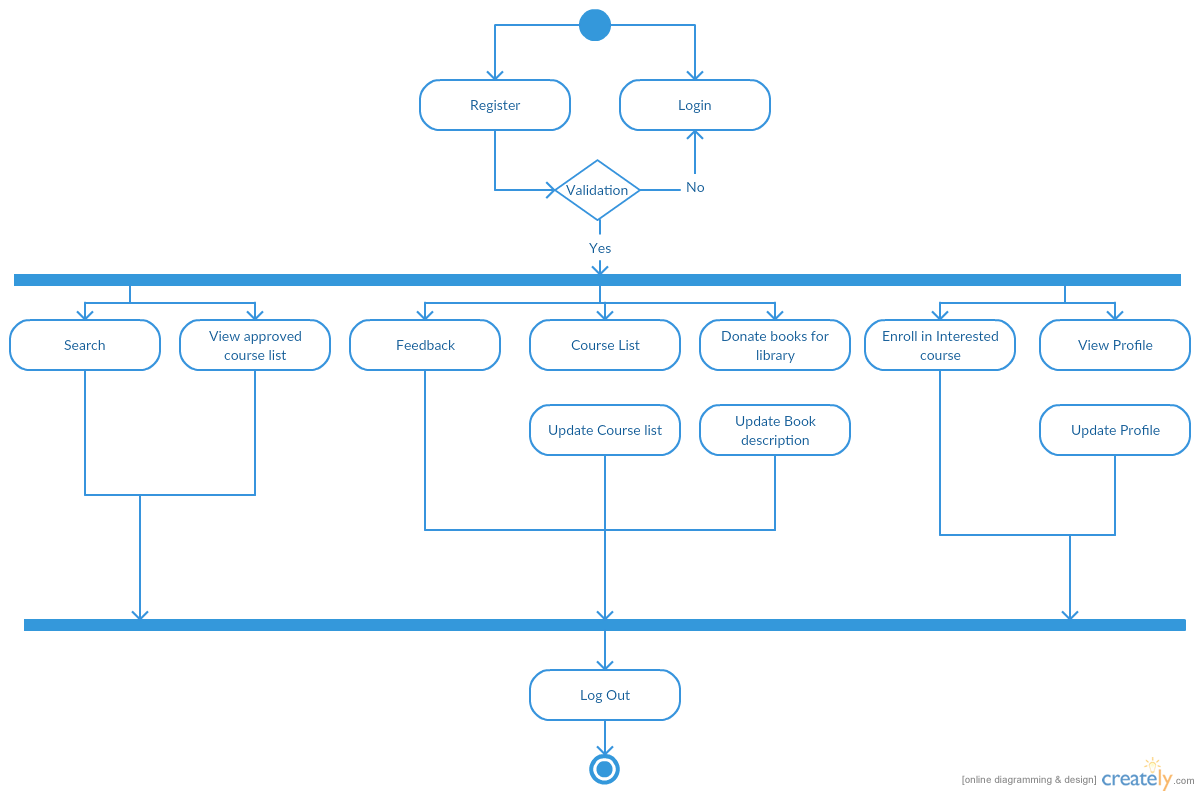


Fig 3.5 Course Learners Registration Activity

**3.3 SEQUENCE DIAGRAMS**

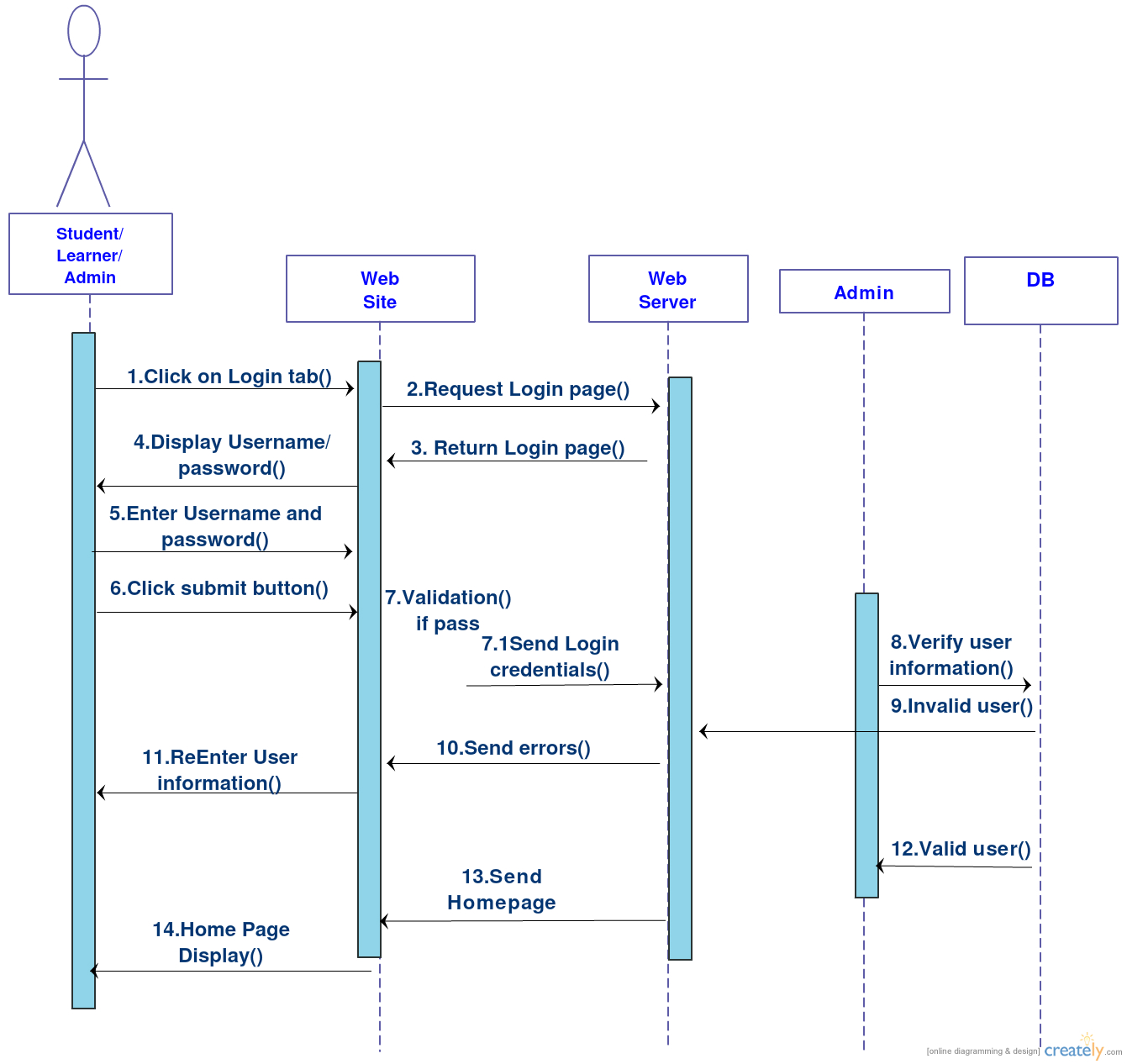
**3.3.1 Student/Learners/Admin Login Sequence Diagram**

Fig 3.6 Sequence Diagram Login

**3.3.2 Student Registration Sequence Diagram**

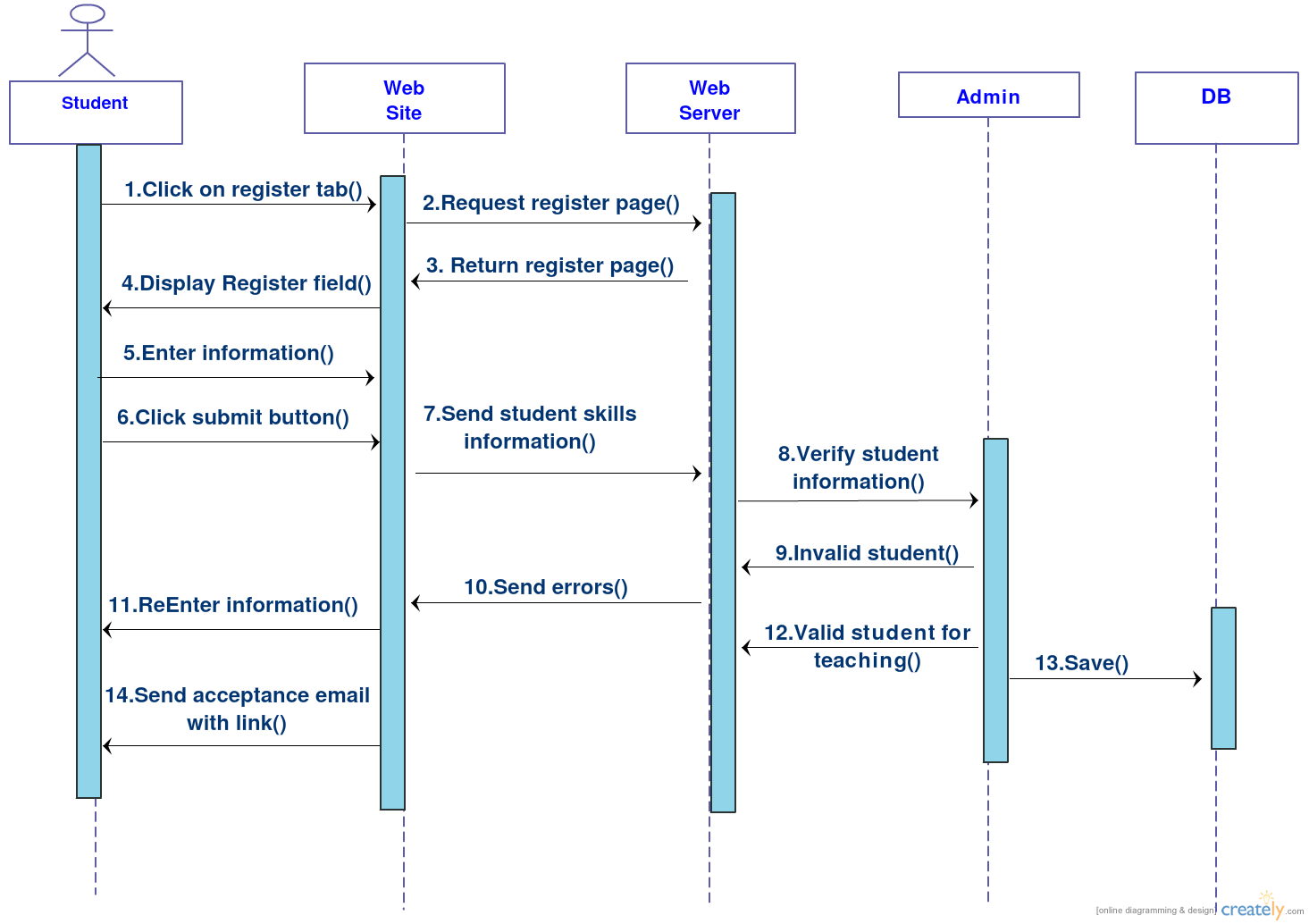
****

Fig 3.7 Sequence Diagram Student Registration

**3.3.3 Learners Course Registration Sequence Diagram**

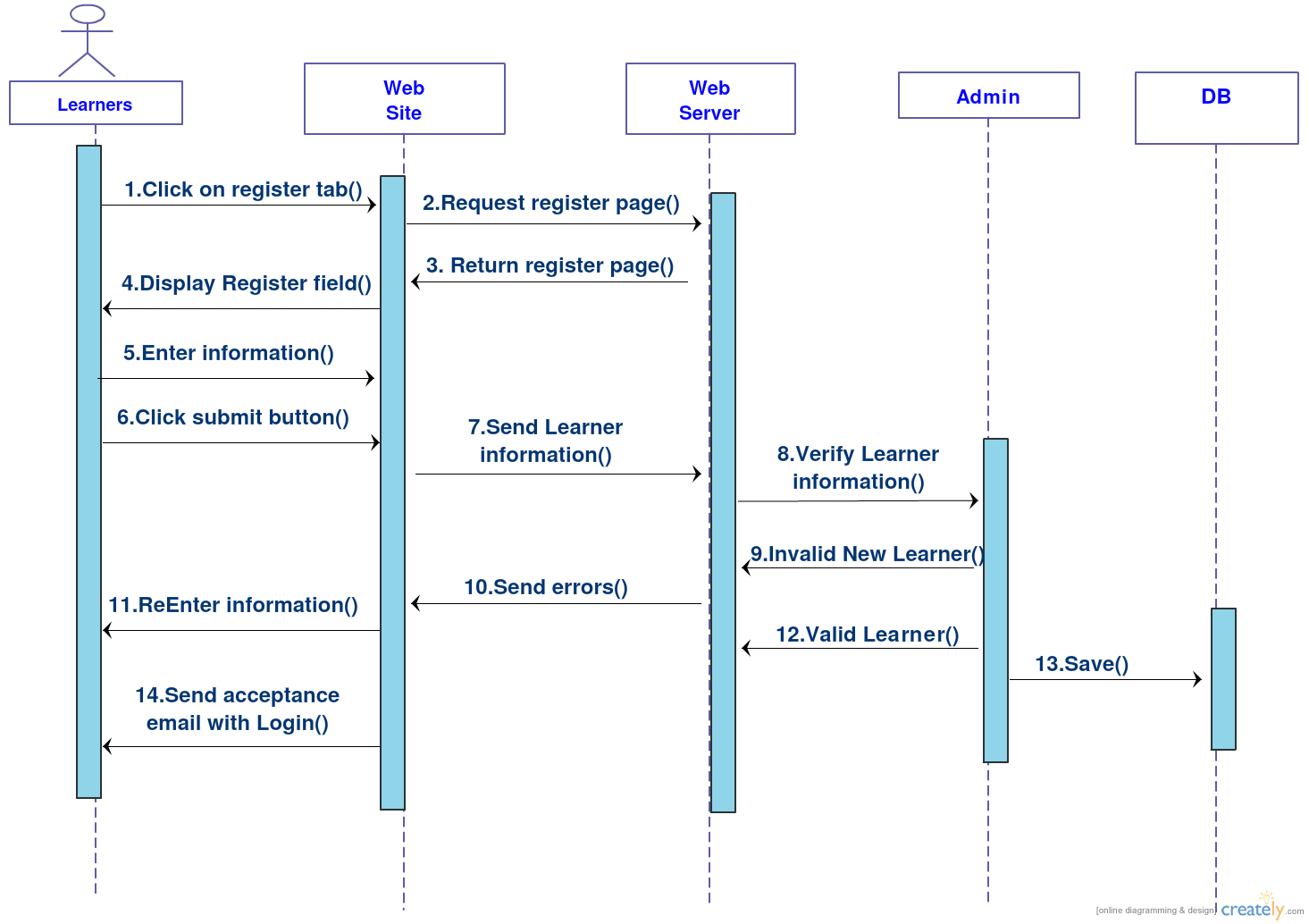
****

Fig 3.7 Sequence Diagram Learners Registration

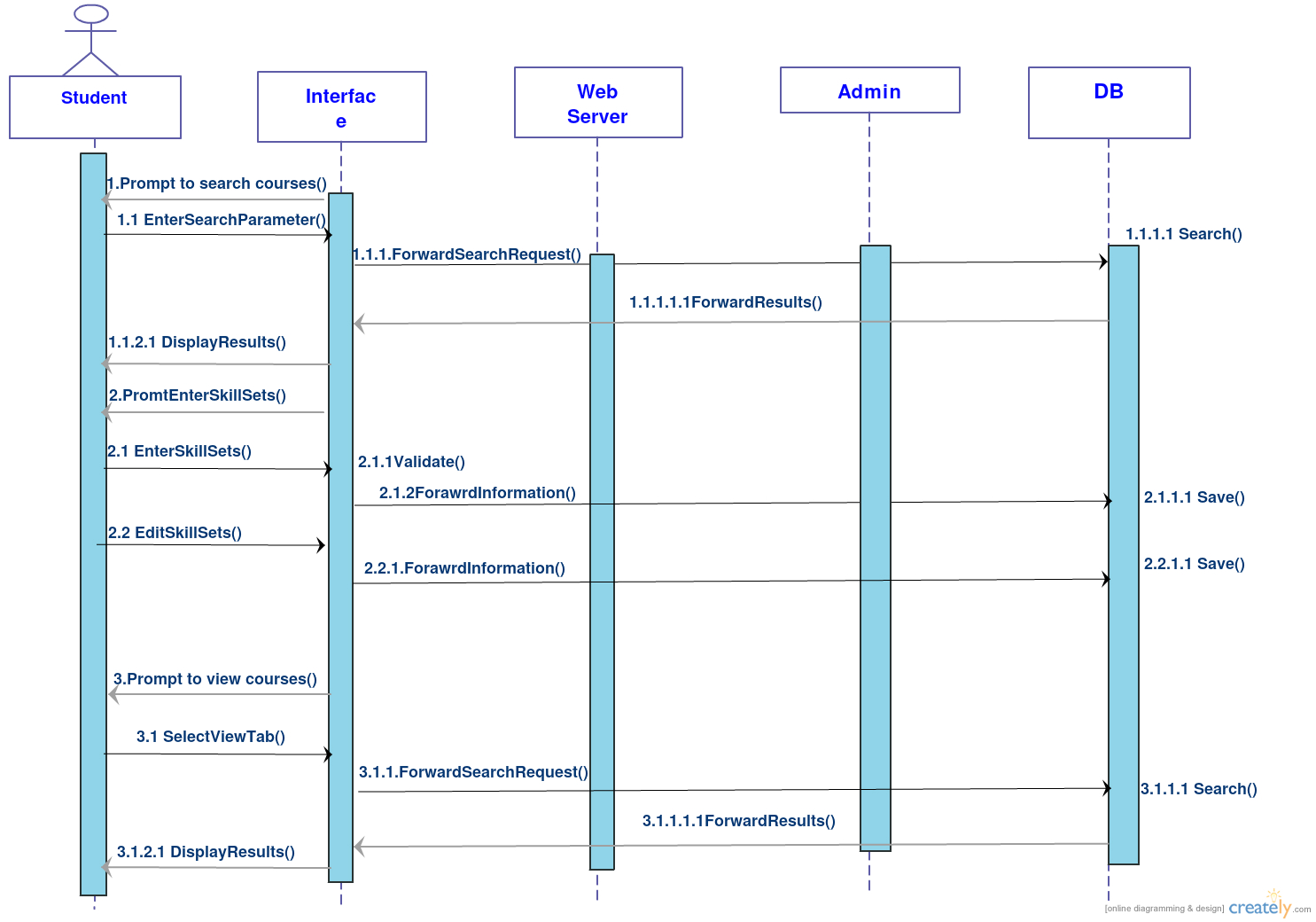
**3.3.4 Student Search and Enter Skill Sets Sequence Diagram**

Fig 3.8 Sequence Diagram Student Enter Skill Sets

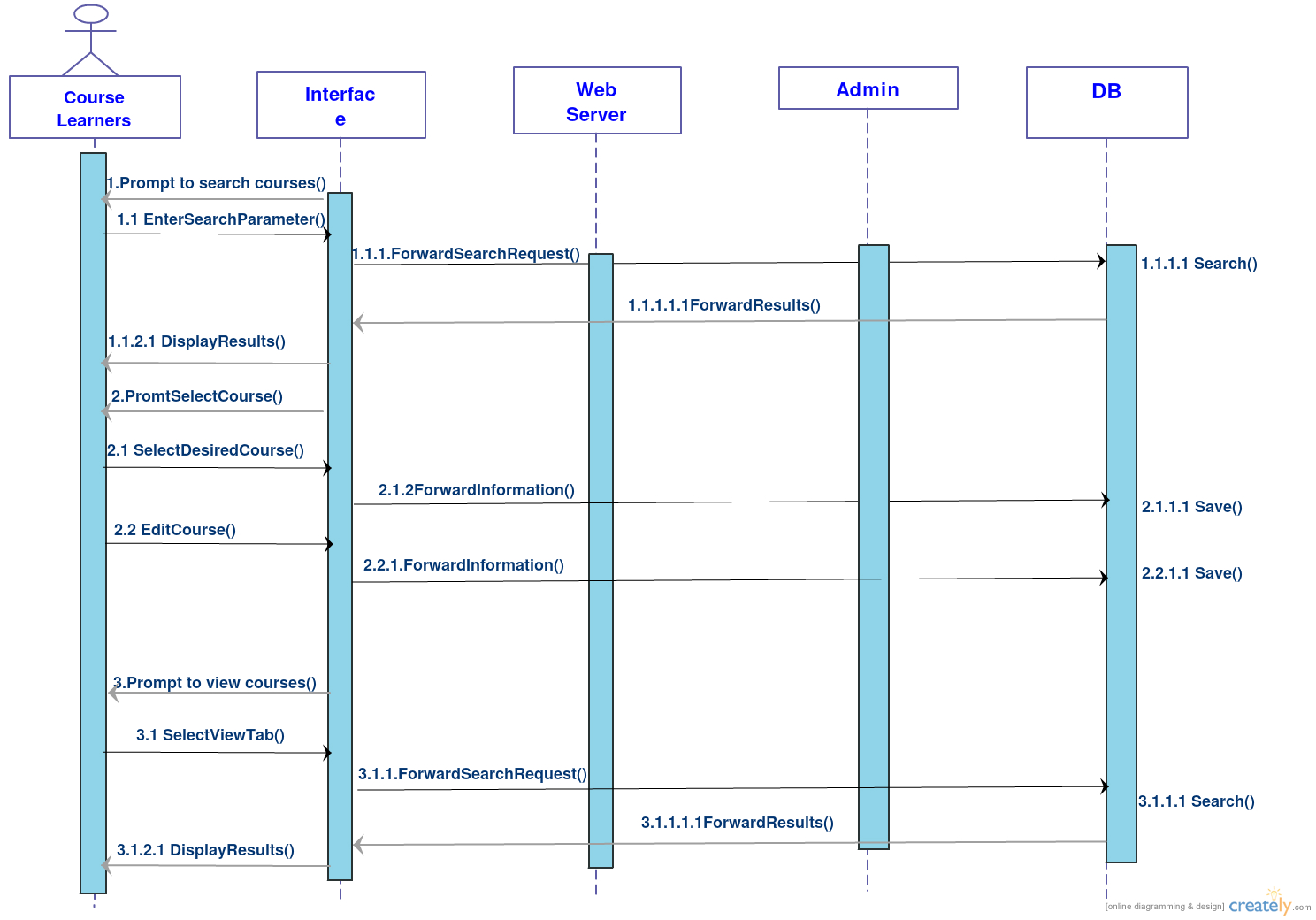
**3.3.5 Learners Search and Course Registration Diagram**

Fig 3.9 Sequence Diagram Learners Course Register

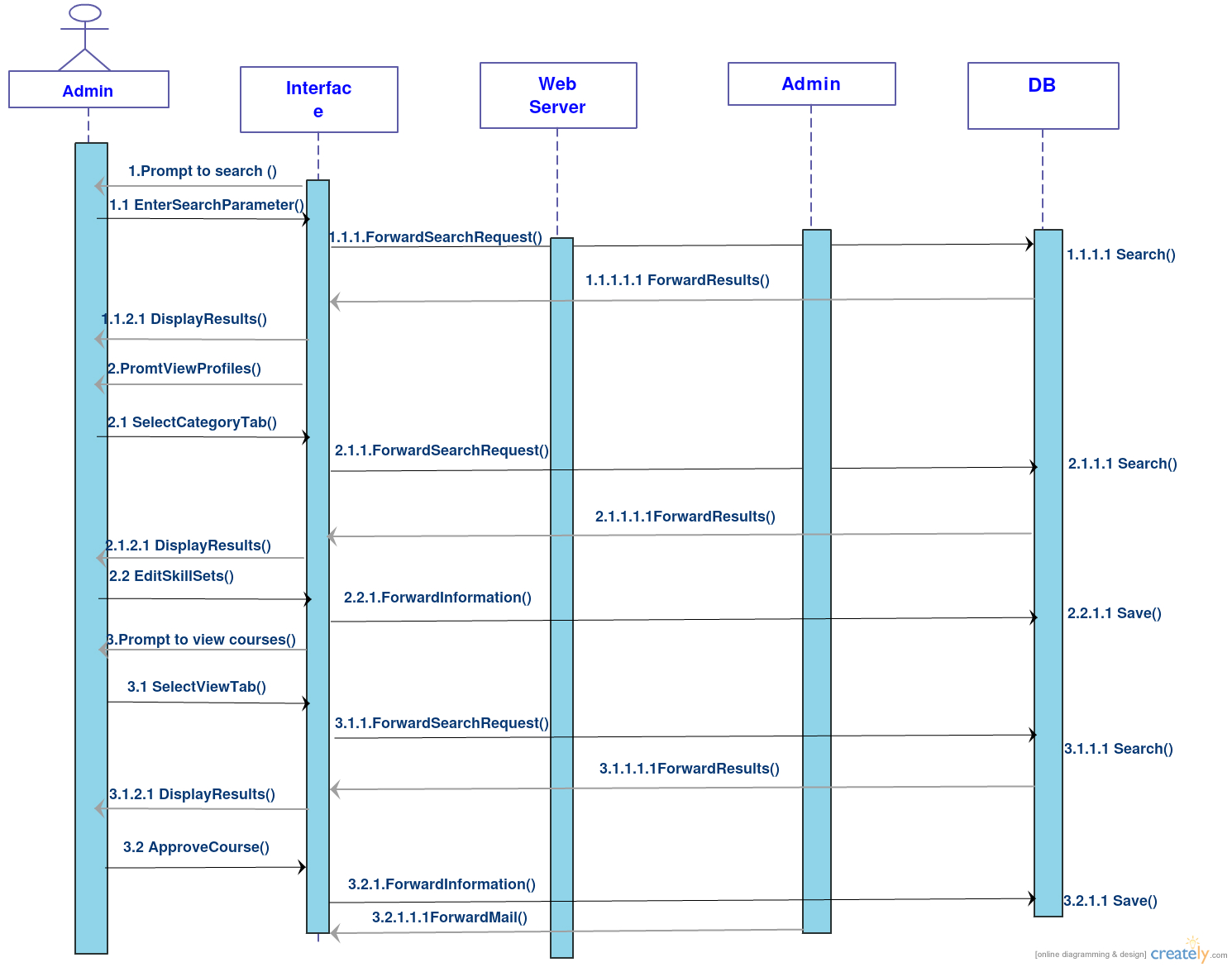
**3.3.6 Admin Registration Diagram**

Fig 3.10 Sequence Diagram Admin Roles

**References.**

[**http://creately.com/**](http://creately.com/)

[**https://github.com/**](https://github.com/)

[**http://www.agilemodeling.com/essays/umlDiagrams.htm**](http://www.agilemodeling.com/essays/umlDiagrams.htm)

[**http://www.tutorialspoint.com/uml/index.htm**](http://www.tutorialspoint.com/uml/index.htm)

[**http://www.oracle.com/index.html**](http://www.oracle.com/index.html)