Software Design Specification

(SDS)

for

Chashm

Prepared by

Ali Amir Khawaja 344200

Nirmal Rai 337582

Hashim Mahmood 334973

National University of Sciences and Technology (NUST)

December 22, 2022

Contents

[1. Introduction 4](#_Toc89944145)

[1.1 System Purpose 4](#_Toc89944146)

[1.2 System Scope 4](#_Toc89944147)

[1.3 Definitions, Acronyms, and Abbreviations 4](#_Toc89944148)

[2. System Overview 5](#_Toc89944149)

[3. Analysis Model 5](#_Toc89944150)

[3.1 Interface Objects 5](#_Toc89944154)

[3.1.2 Acquaintance Association interface objects 6](#_Toc89944155)

[3.1.3 Containership 6](#_Toc89944156)

[3.1.3.2 Add Interface 7](#_Toc89944157)

[3.1.3.3 Update Interface 7](#_Toc89944158)

[3.1.3.4 Delete Interface 8](#_Toc89944159)

[8](#_Toc89944160)

[3.2 Sub System Diagram 10](#_Toc89944161)

[3.3 Control Objects 11](#_Toc89944162)

[4. System Deatiled design 14](#_Toc89944164)

[5. Design Model 14](#_Toc89944165)

[4.1 Use case diagram 14](#_Toc89944166)

[4.2 Class diagram 14](#_Toc89944167)

[4.3 Sequence Diagrams 14](#_Toc89944168)

[4.3.1 Database Population 14](#_Toc89944171)

[4.3 Communication Diagrams 15](#_Toc89944172)

[4.3.1 Database Population 15](#_Toc89944173)

[4.4 Activity Diagrams 16](#_Toc89944174)

[4.5 State machine diagram 17](#_Toc89944175)

[4.6 Deployment Diagram 17](#_Toc89944176)

[4.7 Component Diagram 17](#_Toc89944177)

[4.8 Package Diagram 17](#_Toc89944178)

# 

# Introduction

## System Purpose

This document is the next step work after the completion of the Software Requirements Specification (SRS). The purpose of this document is to describe the detailed architecture and design specifications for the project entitled “Chasm”. The main purpose behind this project was to facilitate a flexible and powerful educational tool for visually impaired people who are unfamiliar with their native language and do not know basic math, which is important in our daily life, how to write or read it. So that they should be left out of this fast-growing world. It would be a tough task at the start but not as difficult as people get used to it. We will do it with help of physical touch and sounds. Normally devices provided to teach these students is too costly and middle-class family will not purchase them, but this product is.

## System Scope

The scope of this framework is to provide Assistant is to ease understanding of Basic Knowledge and to create a convenient and easy-to-use application for visually Impaired People trying to implement abstract learning concepts. By use of this tool, a Person can intuitively listen to the voice of sound plus there will be directions given on how to write those words. Some animations for the guider and Progress will be mentioned at the end of each test or learning assignment.

# System Overview

For the visually impaired, the process of understanding and learning is very difficult as they are unfortunately deprived of receiving visual information from the brain. Braille language is used for the visually impaired, so they can read and write.

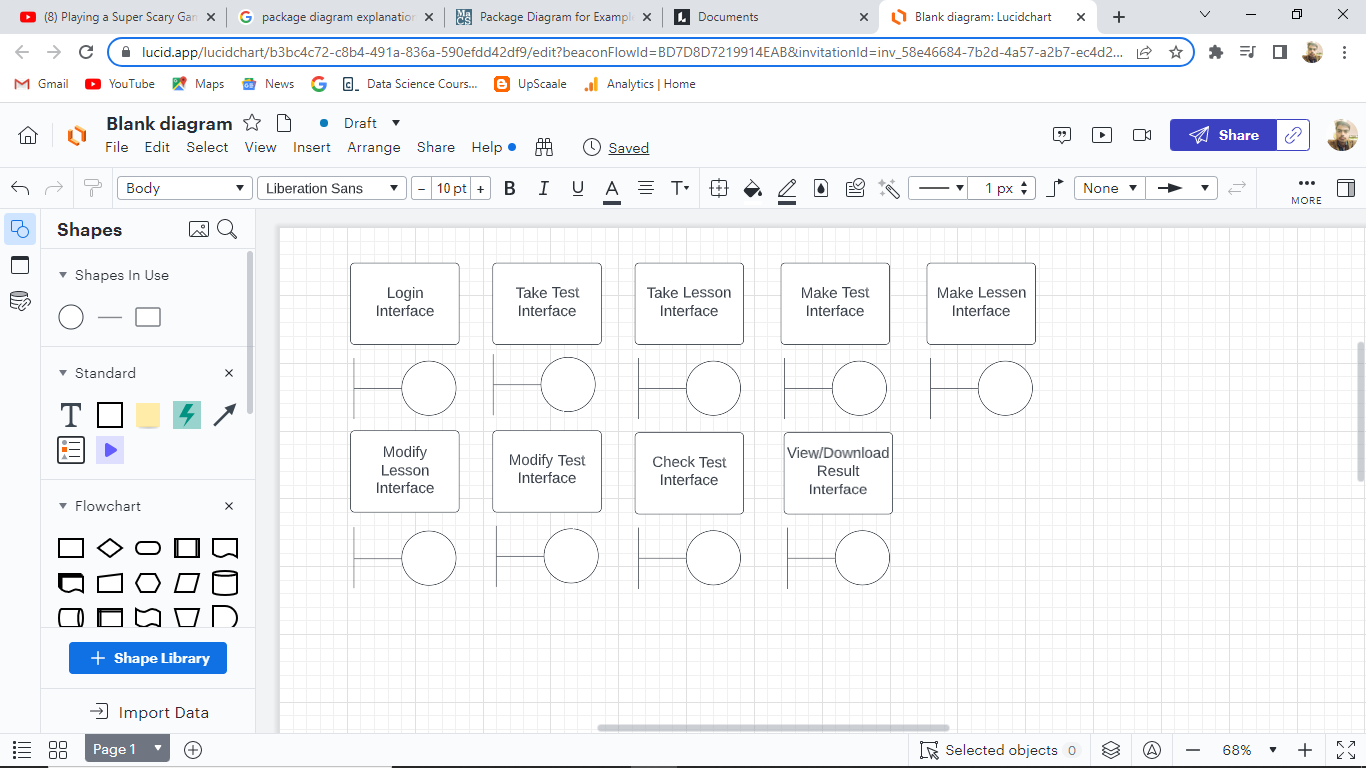
The main purpose behind this project was to facilitate with a flexible and powerful educational tool for visually impaired people who are unfamiliar with native language and do not know basic math which is important in our daily life, how to write or read it. So that they should be left out from this fast growing world. It would be tough task in start but not so difficult as people get used to it. We will do it by help of physically touch and sounds. Normally devices provided to teach these student is too costly and middle class family will not purchase it but this product

# Analysis Model



## Interface Objects

The interface objects represent the main interfaces of the system. Given below is the list of different interface objects used in our system.



### Acquaintance Association interface objects

**System Access**: this includes the login interface that provides access to the entire system.

**System interaction**: take test, take lesson, make test, make lesson, modify lesson, modify test, and check test interface. These all are the interfaces that help in system interaction.

**Output generation:** The view download result interface is used to generate output from the system.

### Containership

* + - 1. Login Interface

Diagram

Description automatically generated with medium confidence

# Test Interface

Table

Description automatically generated

# Lesson Interface

Table

Description automatically generated

# Result Interface

# Diagram Description automatically generated with medium confidence

3.1.3.5. System Diagram

Diagram

Description automatically generated

Diagram

Description automatically generated

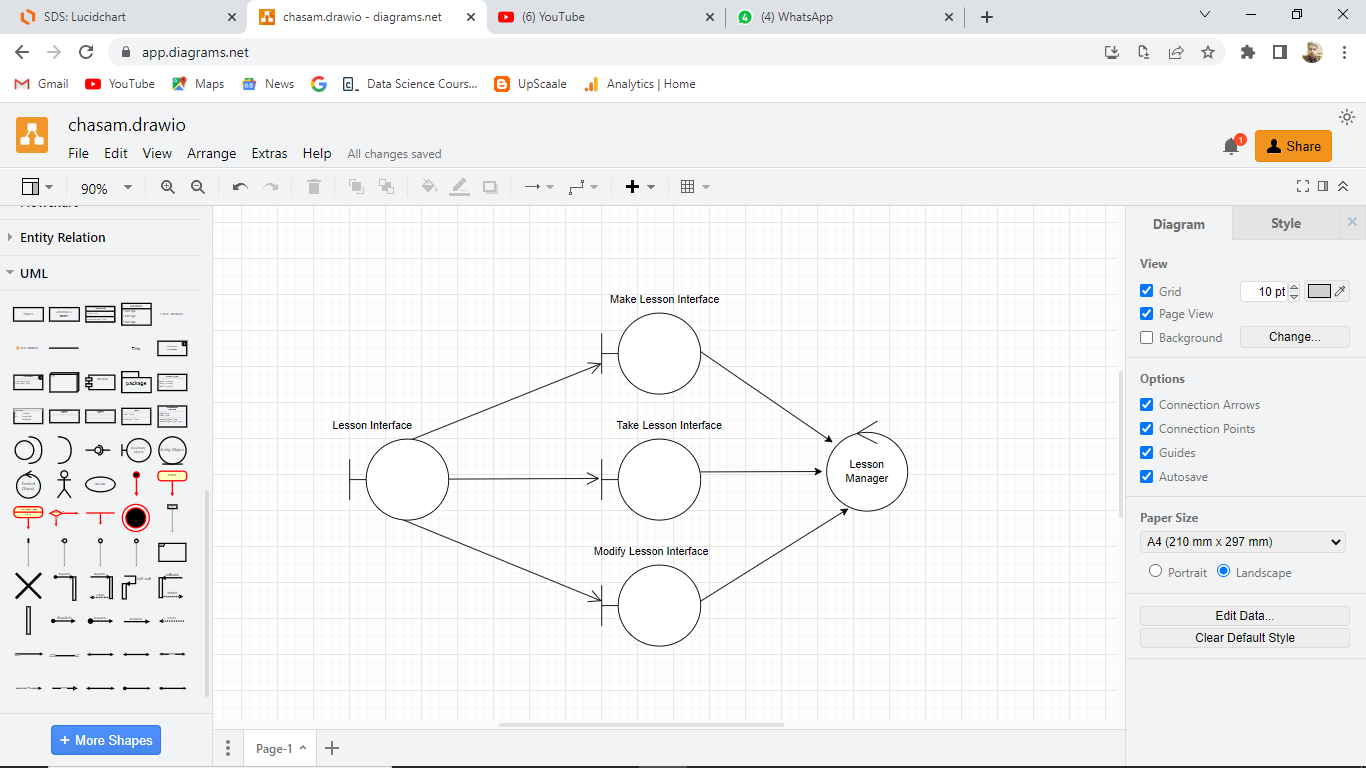
## Sub System Diagram

Test Sub-Diagram:

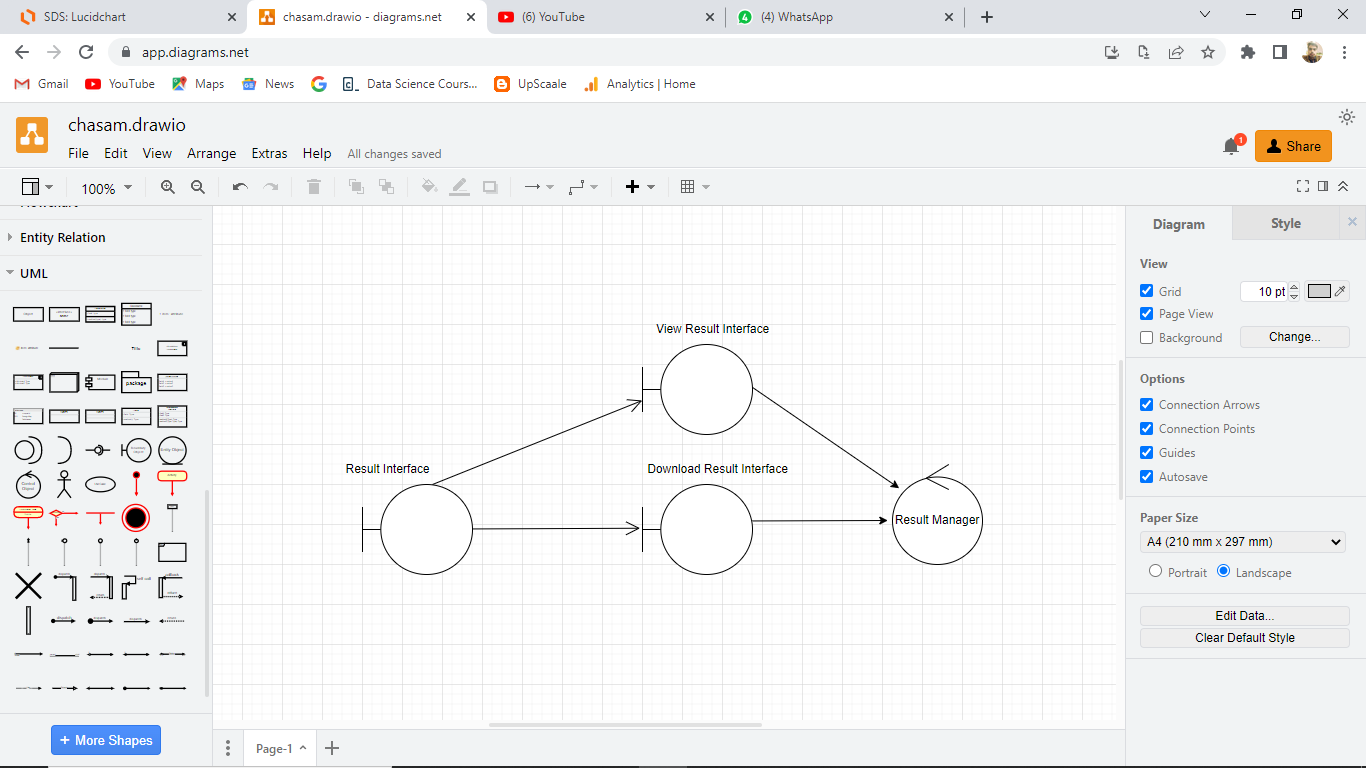
Diagram

Description automatically generated

Lesson Sub-Diagram:



Result Sub-Diagram:



Login Sub-Diagram:

A screenshot of a computer

Description automatically generated with medium confidence

## Control Objects

Control objects are responsible for computation or processing tasks. Following are the main control objects in the proposed framework.

**Description of Control Objects**

1. **Navigation Controller:**

This control object is responsible for the navigation among interfaces.

1. **Action Controller:**

This control object is used to manage policy *Action* parameter. It handles all the Action Add test, Update test, and Add lesson and change Lesson operations as per user request.

1. **Response Receive controller :**

This Controller is used to receive all the responds sent by User

1. **Response sender Controller:**

This Controller is used to send the information to system

1. **Policy Controller:**

This Controller is used to define the rules of how to use hardware

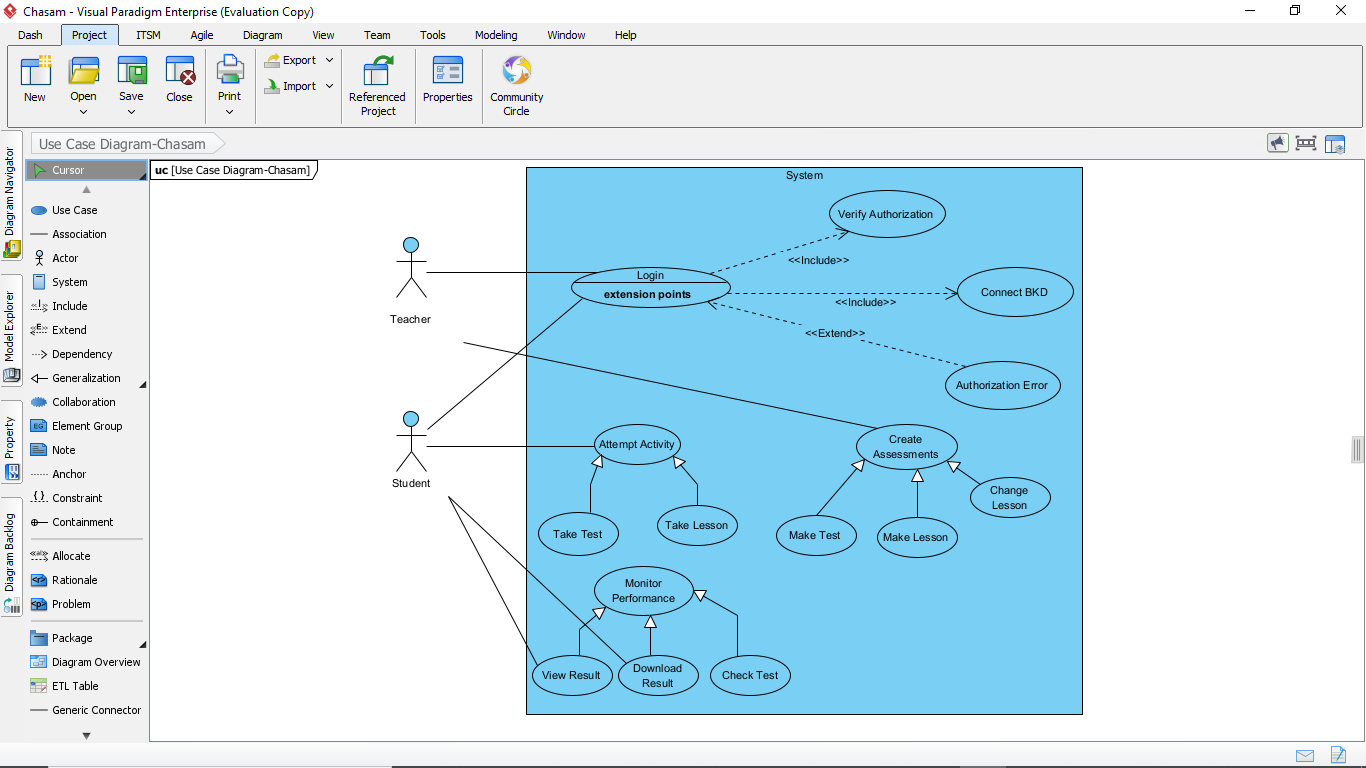
1. **Authentication Controller:**

This is used to control any irrelevant login and give them proper message of invalid login.



# System Detailed design

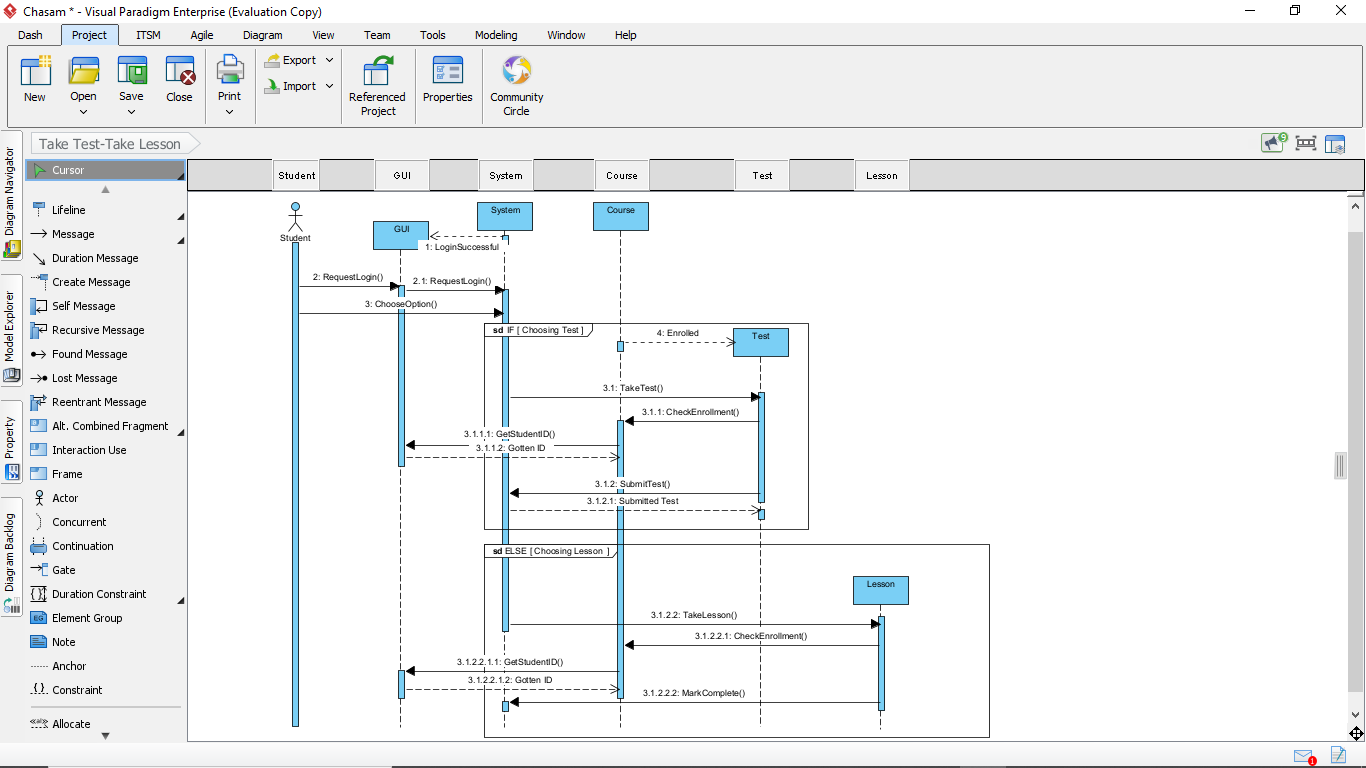
## Use case diagram



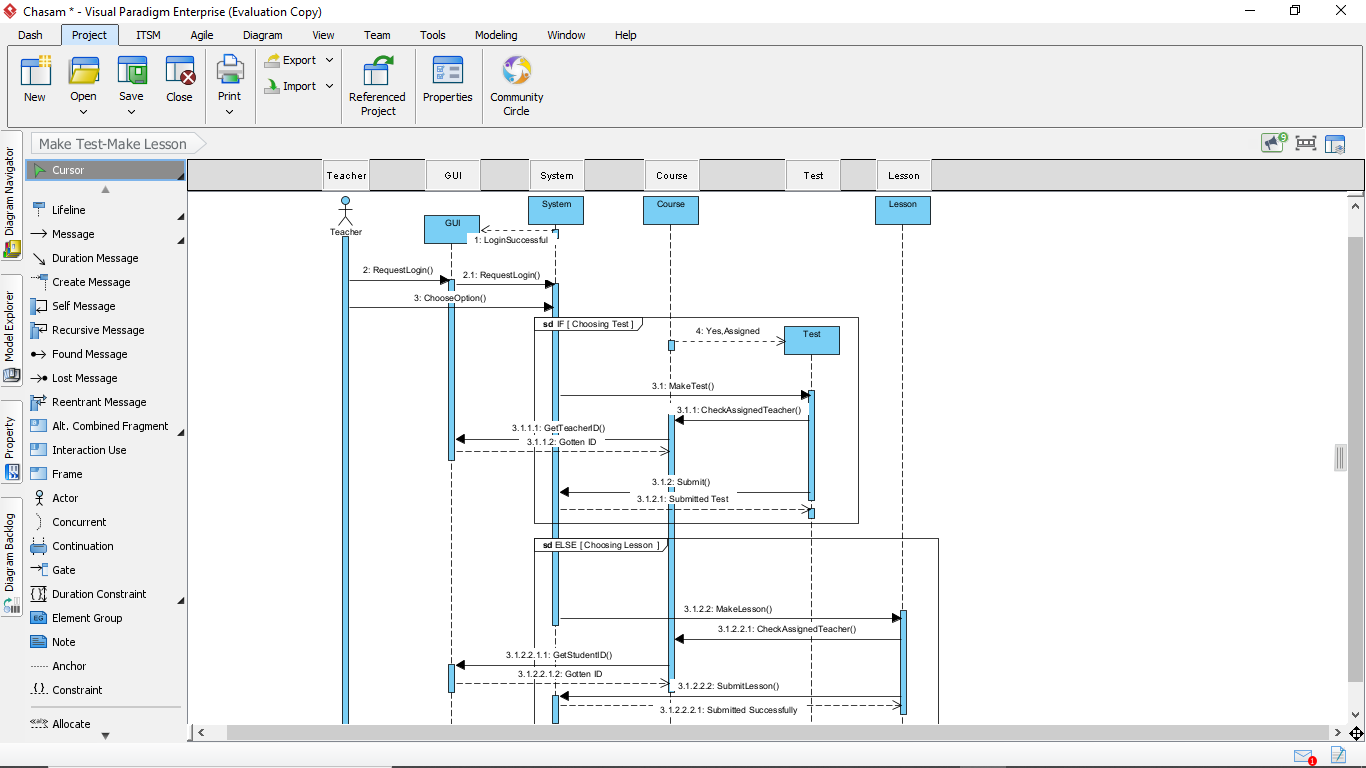
## 

## Class diagram

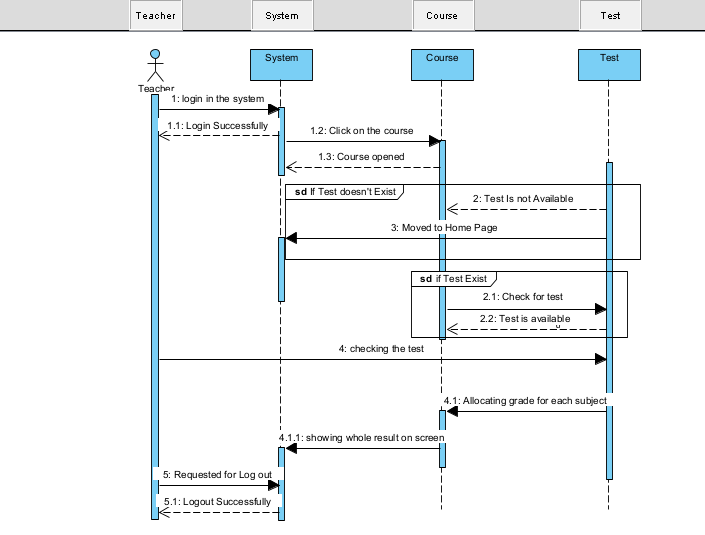
## Sequence Diagrams

**Take Test/Lesson**

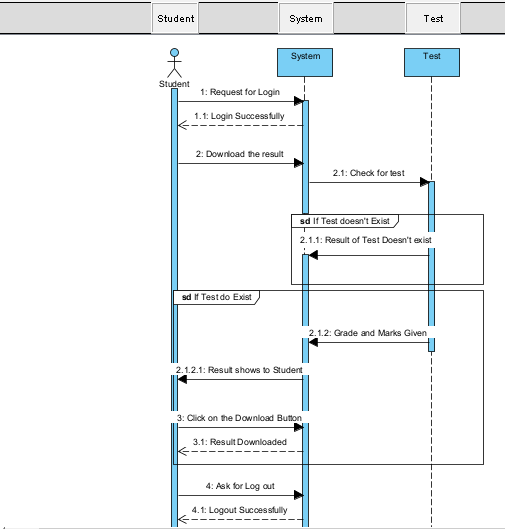
**Make Test /Lesson**



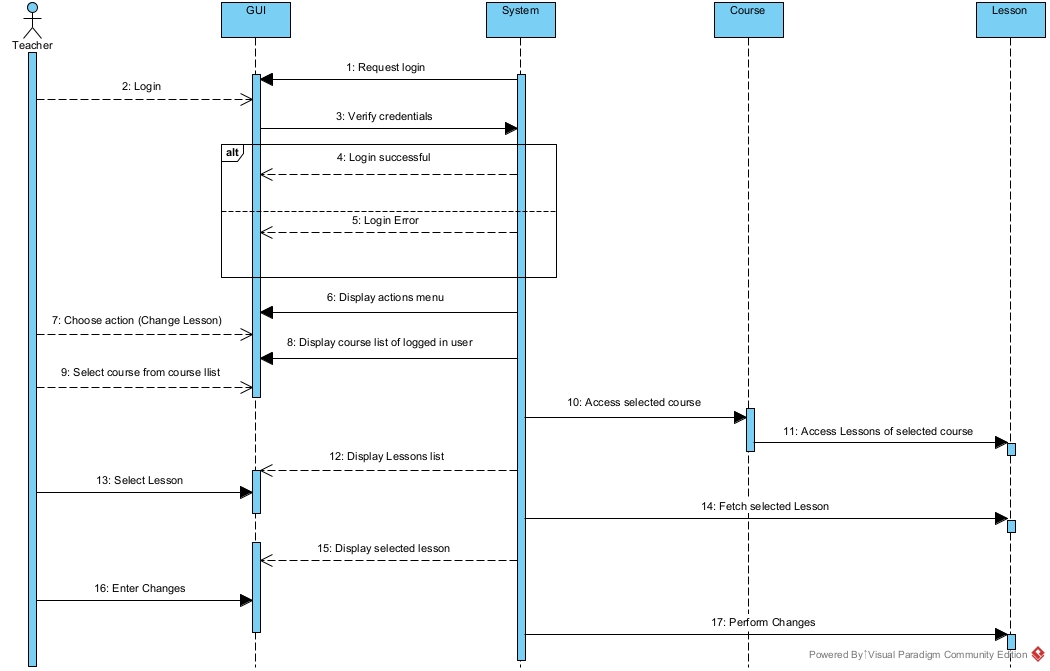
**Check Test**

****

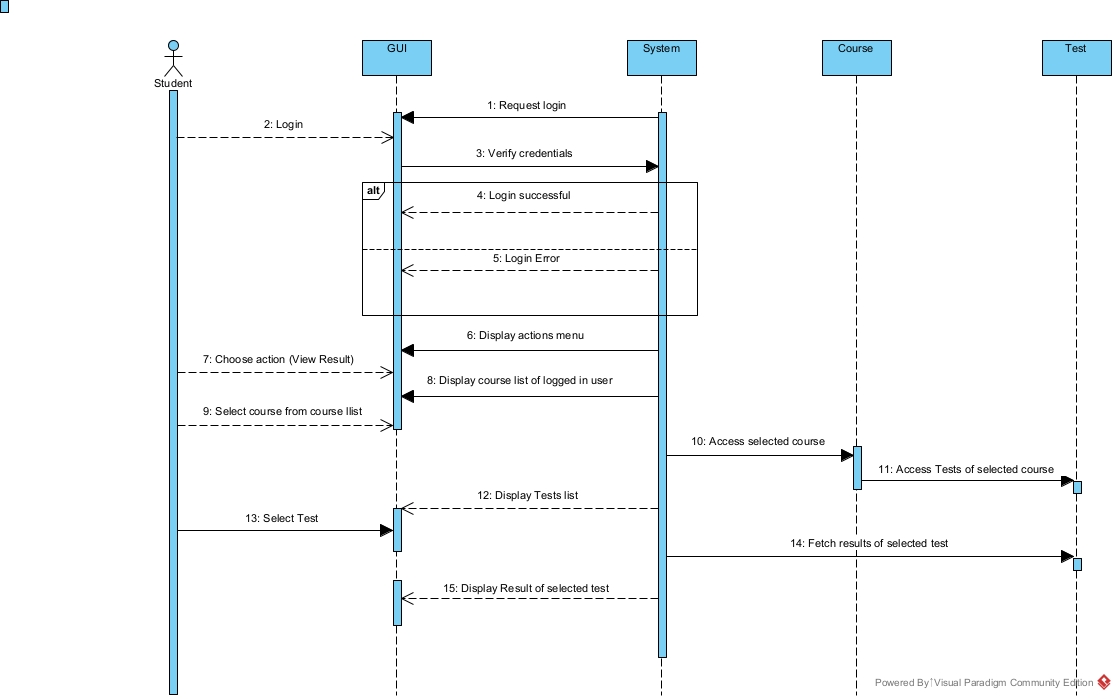
**Download Test**

****

**Change Lesson**

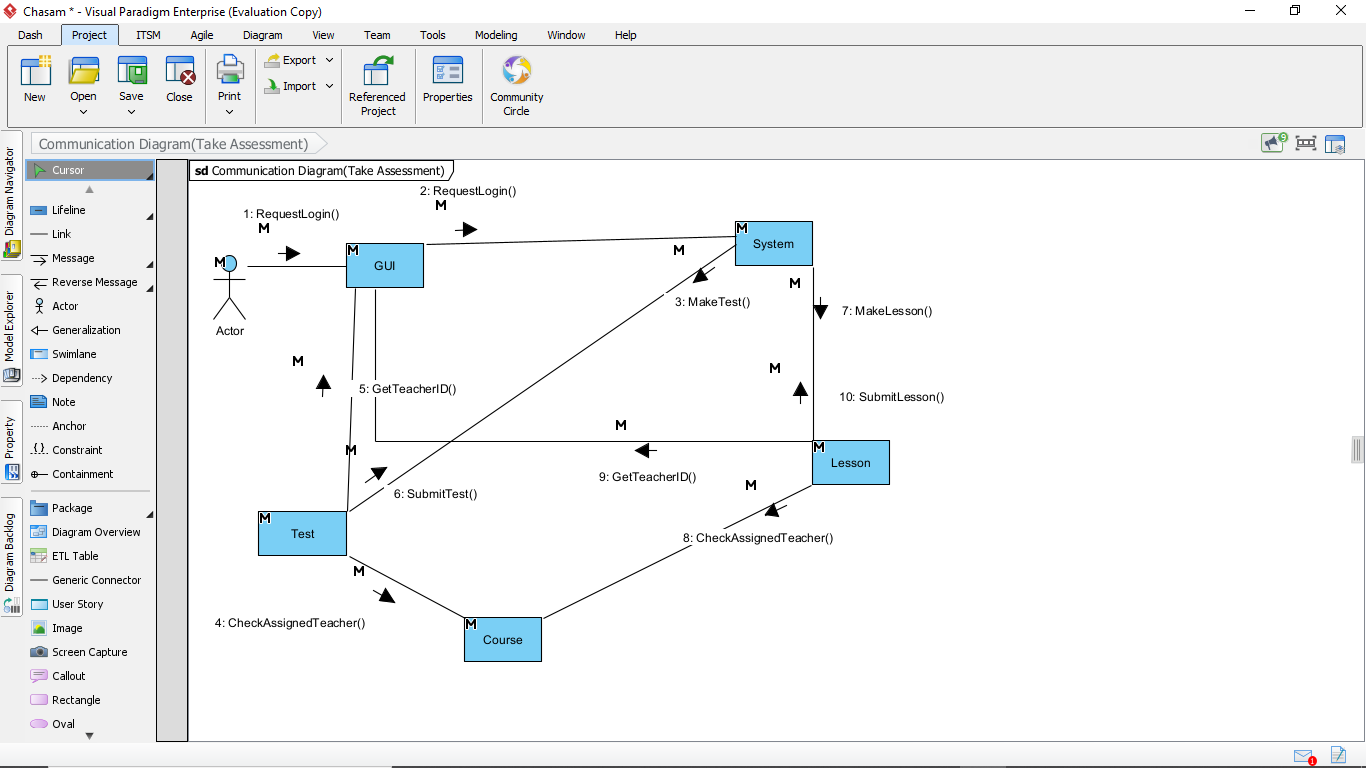
****

**View Result**

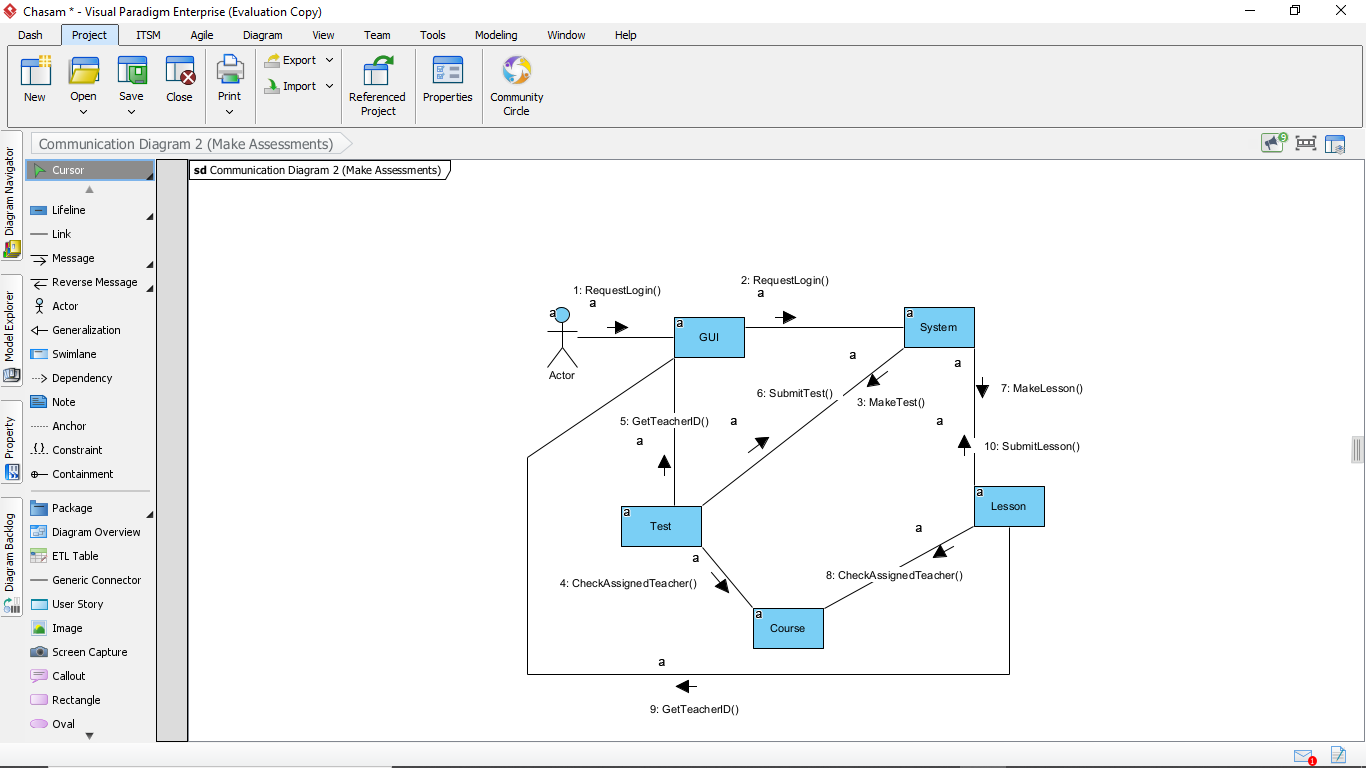
****

## Communication Diagrams

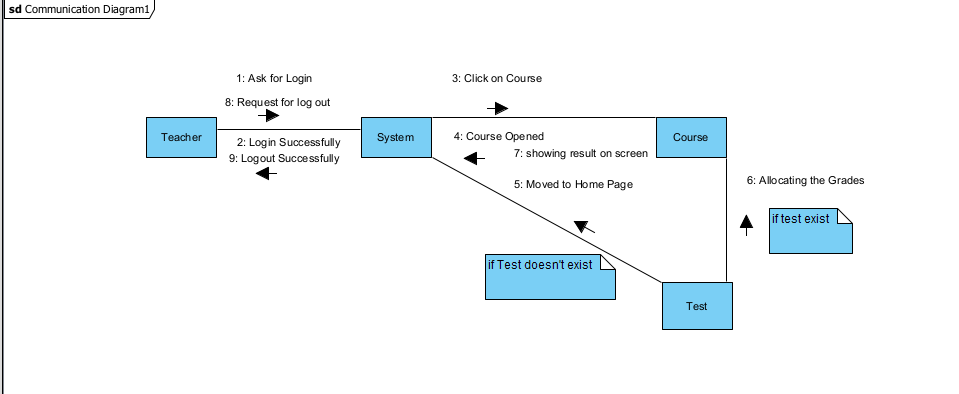
**Take Assessment**



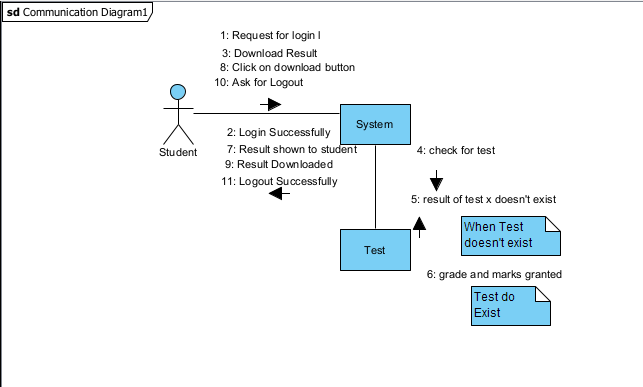
**Make Assessments**



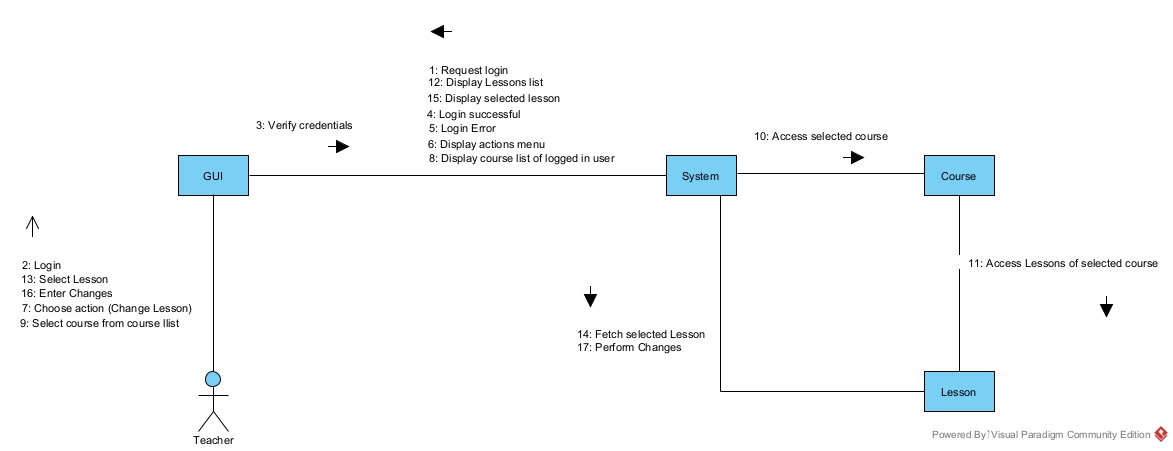
**Check Test**



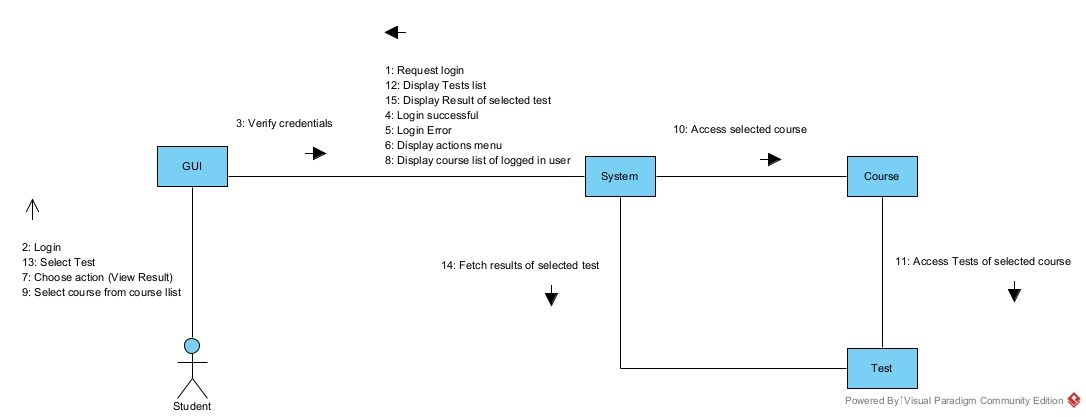
**Download Result**



**Change Lesson**

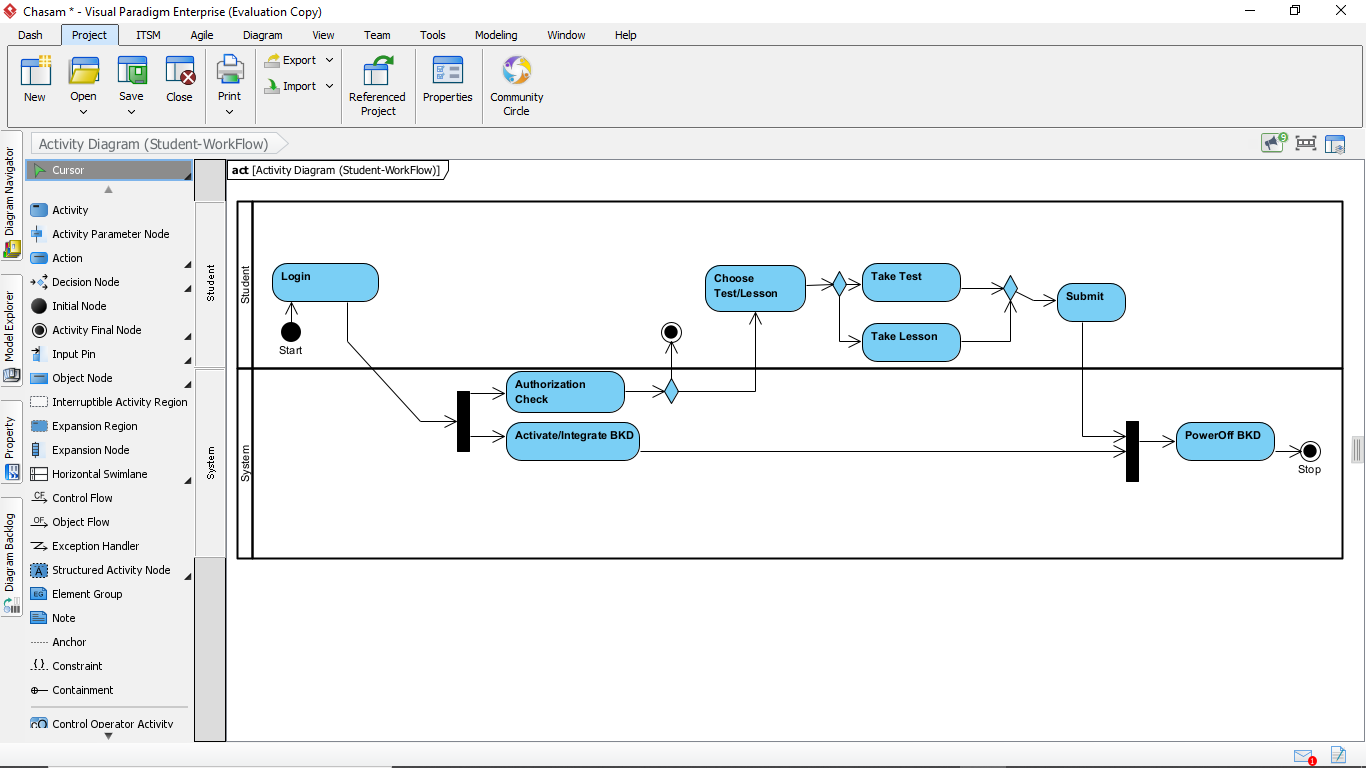
****

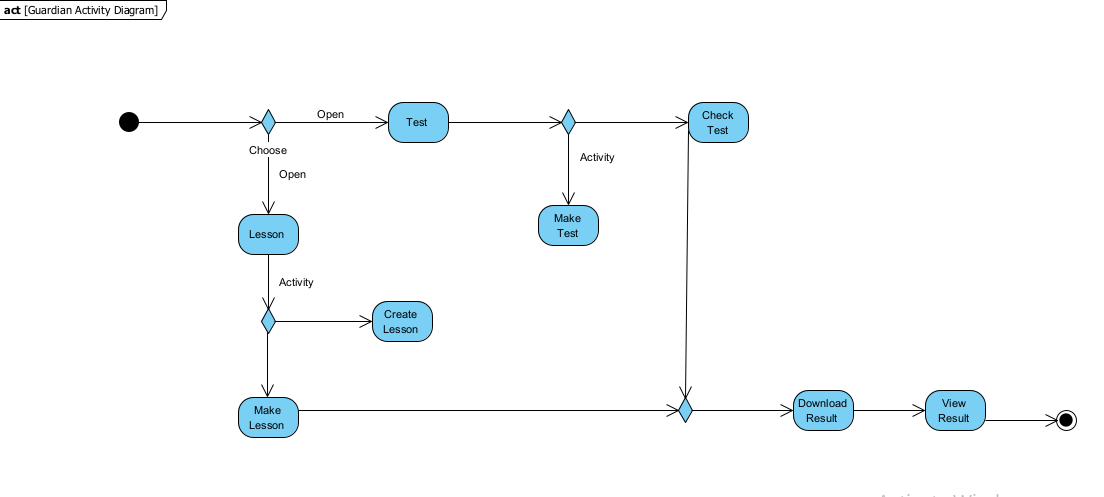
**View Result**

****

## Activity Diagrams

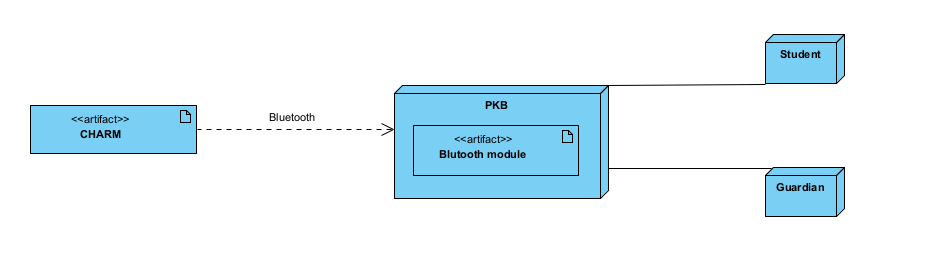
**Student Activity Diagram**



**Guardian Activity Diagram**

## State machine diagram

## Deployment Diagram



## Component Diagram

## Package DiagramChart, box and whisker chart Description automatically generated

## System architecture diagram

