# Names & IDs:

**Reem Amgad: 202000438**

**Samaa Maged: 202000597**

**Alaa Hussein: 202001728**

**Fatema Gamal: 202002674**

**Gannat Elsayed: 202000750**

**“Momo App”**

NU mapping app

Find your shortest way

Table of Contents

1.0. Introduction

1.1. Purpose

1.2. Scope

1.3. Technology

1.4. Intended Audience

2.0. Overall description

2.1. Our problem

2.2. Our solution (system overview)

2.3. System overview

3.0. Interface

3.1. System Interface

4.0. Functional Requirements

4.1. Choose your destination

4.2. Enter your room number

4.3. Starred places

4.4. Upcoming events

4.5. Student-login

5.0. Non-Functional (Technical) Requirements

5.1. Reliability

5.2. Multi-Lingual support

5.3. Portability

5.4. Recoverability

5.5. Performance

5.6. Application Affinity/Compatibility

6.0. Use cases and scenarios

6.1. Use case diagram

6.2. Login scenario

6.3. Login exception scenario

6.4. Choose destination scenario

6.5. Enter room number scenario

6.6. Enter room number exception scenario

6.7. Starred places scenario

6.8. Upcoming events scenario

6.9. Edit personal information scenario

1. **Introduction:**
   1. **Purpose:**

The purpose of this document is to present a detailed description of the NU Mapping mobile app. It will explain the purpose and features of the system, the interfaces of the system, what the system will do, this app will help us navigate easily through NU and to find a specific place with the shortest path. This document is intended for both the stakeholders and the developers of the system and will be proposed to theuniversity’s media department.

* 1. **Scope of project**

This software will be a mobile app publishing the shortest path inside Nile university. This app will help people go lectures and tutorials in an easy way and to help them find the shortest path to a specific place. Our goal is how to facilitate college experience, in addition, we need to reduce human resources, making a simple yet efficient app easy to use. This app will review the place details that the user selected.

More specifically, this system allows the user to search for a specific room or place on campus then the direction will be available for the user. This System will be designed in order to be easily for the user to use, also it will contain a data base.

* 1. **Used technologies**

This app is designed with kotlin language in addition XMLto design the user interface, also we are using data base (MY SQL) to store the data. There are types of data, for example the name of the buildings, rooms and places categories.

* 1. **Intended Audience**

This document is useful to:-

Customer: To follow up with the project team if there is requirements change.

Developer: To have ease of implementation.

Software Tester: To be able to develop the correct test cases for the system and test it to improve the system's quality.

Software architect: To be able to design and improve the system's architecture.

Project Manager/SCRUM Master: To be able to develop a good plan to work on the project and construct the team.

Project Supervisor: To be able to mentor the teams’ work and assist with any advice and make sure that the goals are met within the time allotted.1.5ReferencesWe are targeting any.

1. **Overall description:**
   1. **Our problem**

Every year there is at least 2000 newcomer to our university. They don’t know anything about the places or rooms or even the building numbers. In addition, there are many freshmen who being very shy to ask anyone and they prefer be confused rather than ask anyone. On the other hand there is new Drs Or new TAs who suffers for a while from this problem

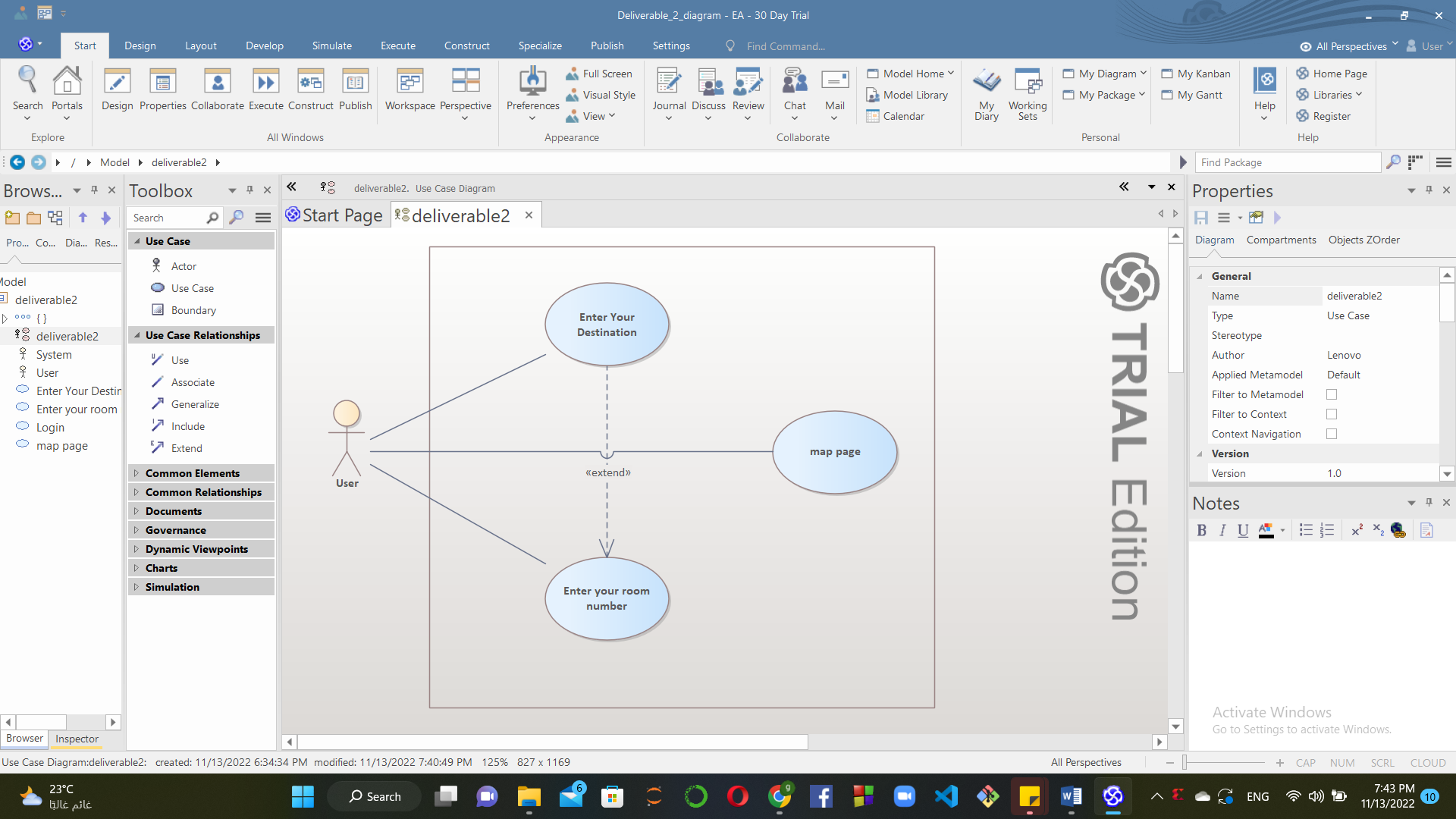
* 1. **User Characteristics**

There is essentially one type of user who can interact with the system . Which is any student with mobile phone access who needs help getting around campus, whether he is a freshman or not.

* 1. **Our solution**

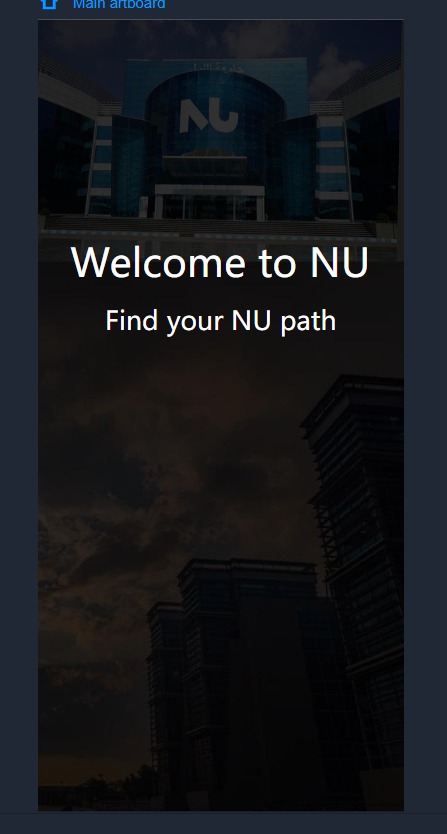
Momo app is the solution we’re trying to offer. This app for NU students, Drs or teaching assistants. So this app is the solution to navigate the road inside the university with the shortest way. You don’t need anything to use the app you just enter the university and start opening the app to go to your destination.

* 1. **System overview**



**3.0. Interface:**

3.1. System Interface



Our Welcome Page

A picture containing text, outdoor

Description automatically generated

Our Home Page

On this page, we choose our destination.

A picture containing graphical user interface

Description automatically generated

Page 1

We look for the rooms in building 1 on this page.

Graphical user interface, application

Description automatically generated

Side page.

To make it simple for users to switch between pages, we created side page.

A picture containing background pattern

Description automatically generated

Map page.

Our map page allows the user to navigate to the location they've chosen.

**4.0. Functional Requirements**

4.1. Choose destination

Aim: To choose the destination you want.

Input: Select a destination from dropdown menu.

Output: navigates to the desired page.

Process: To navigate to the map page or building page to choose specific room

4.2. Enter room number

Aim: To enter the room number in specific building

Input: Room number

Output: Map page of our chosen room

Process: To search for a specific room in the building we choose

4.3. Starred places

Aim: To show your starred places

Input: None

Output: Your starred places

Process: To easily find user’s regularly visited places

4.4. Upcoming events

Aim: To display your upcoming events

Input: None

Output: Your upcoming events

Process: To check your daily events

4.5. Student-login

Aim: Student can log-in to get access to the app with unique username and password

Input: Username and password

Output: Displays a message if the username and password does not match otherwise go to the home page

4.6. Edit personal info

Aim: Student can edit his own personal info

Input: student name, level, major

Output: None

Process: Display a message that the info is updated

**5.0. Non-Functional (Technical) Requirements**

5.1. Reliability

The mean time between failures should be at Maximum 1 time per year.

5.2. Adaptability

The system should be language agnostic since portability is important for users. English and Arabic should be supported by the system. (Multi-Lingual Support)

5.3. Portability

The system should be capable to adapt in different environments without applying action or means other than those provided for this purpose in the product. Since the system works on portable phones, the system therefore is available inside the university or outside it, it’s not restricted to a specific place.

5.4. Recoverability

The system should recover after a breakdown within approximately 5 mins.

5.5. Performance

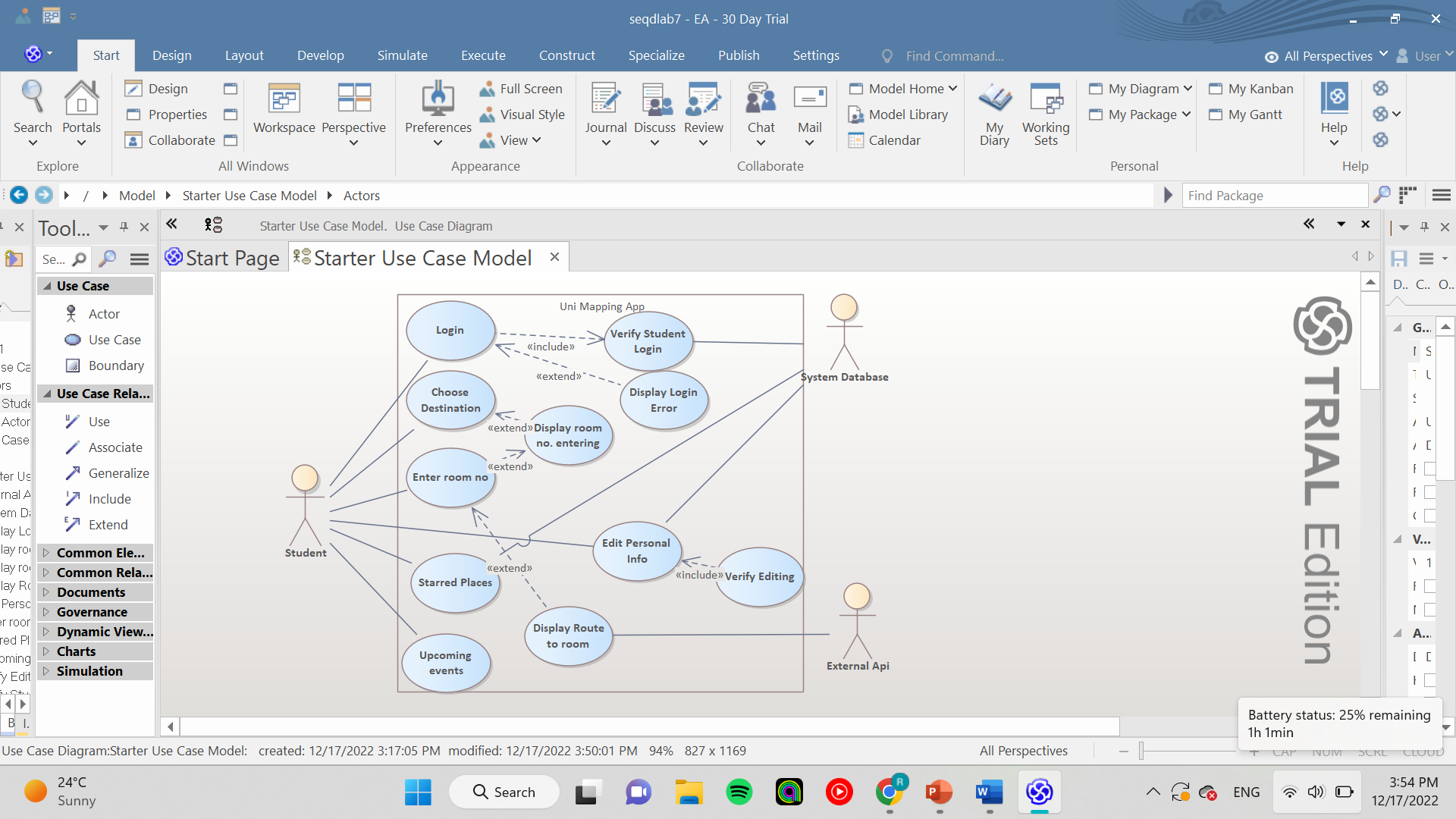
Loading pages shouldn’t exceed 2 seconds (Response Time)

5.6. Application Affinity/ Compatibility

The system shall be compatible with android version 5 and up to the latest android version (Android 13)

**6.0. Use cases and scenarios**

6.1. Use case diagram



Our use case diagram.

6.2. Login scenario

A screenshot of a computer

Description automatically generated

6.3. Login exception scenario

A screenshot of a computer

Description automatically generated

|  |  |
| --- | --- |
| ***Use case Name*** | ***Login*** |
| ***Actors*** | ***Student and University system*** |
| ***Main success scenario:*** | ***University system prompt the student for username and password.***  ***Student enters a valid username and password.***  ***University system verify student’s username and password.***  ***Student is logged in.*** |
| ***Exceptions*** | ***Student enters wrong username or password.***  ***University system displays “wrong username or password, please try again”.***  ***Student has 3 trails.***  ***University system will block the student’s account for 1 minute after the 3 trails.*** |
| ***Pre-condition*** | ***Student must have an account and enters valid username and password.*** |
| ***Post condition*** | ***Student is successfully logged in.*** |

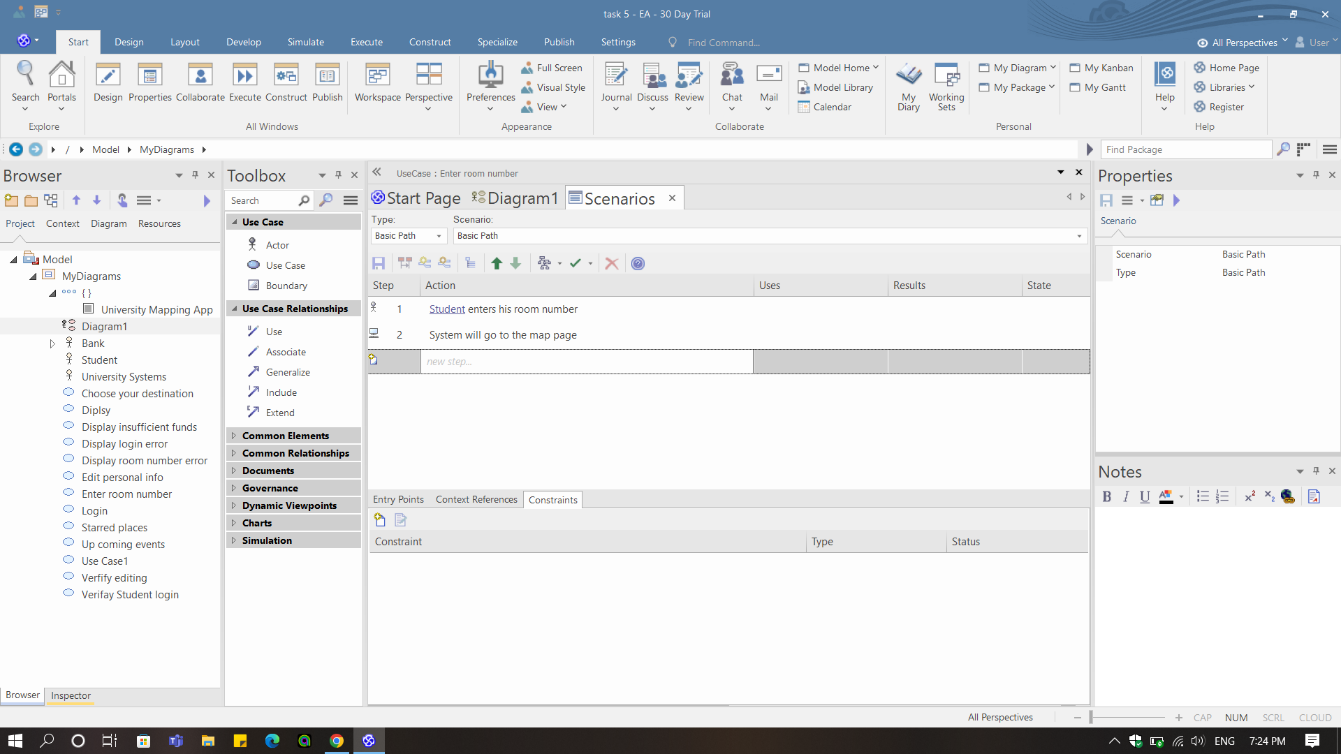
6.4. Choose destination scenario

A screenshot of a computer

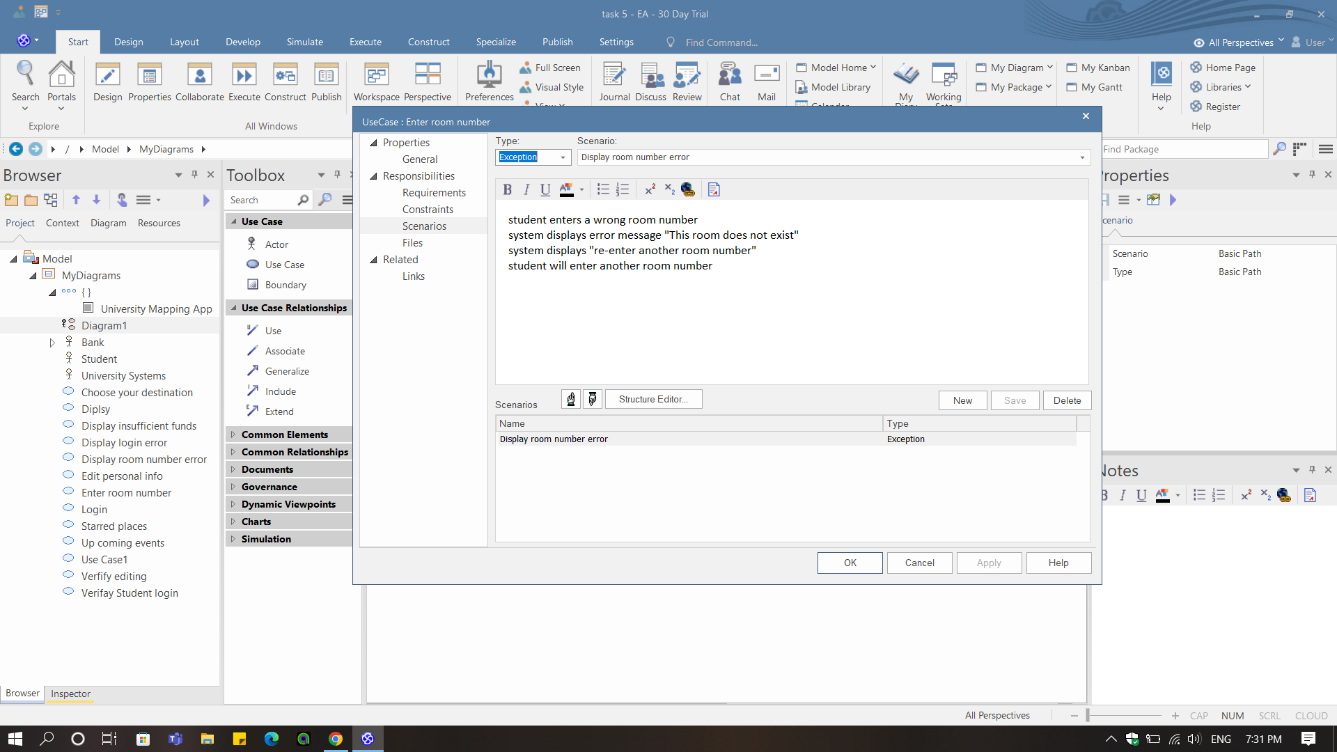
Description automatically generated

|  |  |
| --- | --- |
| ***Use case Name*** | ***Choose destination.*** |
| ***Actors*** | ***Student and University system.*** |
| ***Main success scenario:*** | ***Student will choose from drop down menu his destination.***  ***University system will open the map page.***  ***University system will open the chosen destination page.*** |
| ***Exceptions*** | ***No exceptions.*** |
| ***Pre-condition*** | ***Student must be logged in.*** |
| ***Post condition*** | ***The page of the destination entered is displayed.*** |

6.5. Enter room number scenario

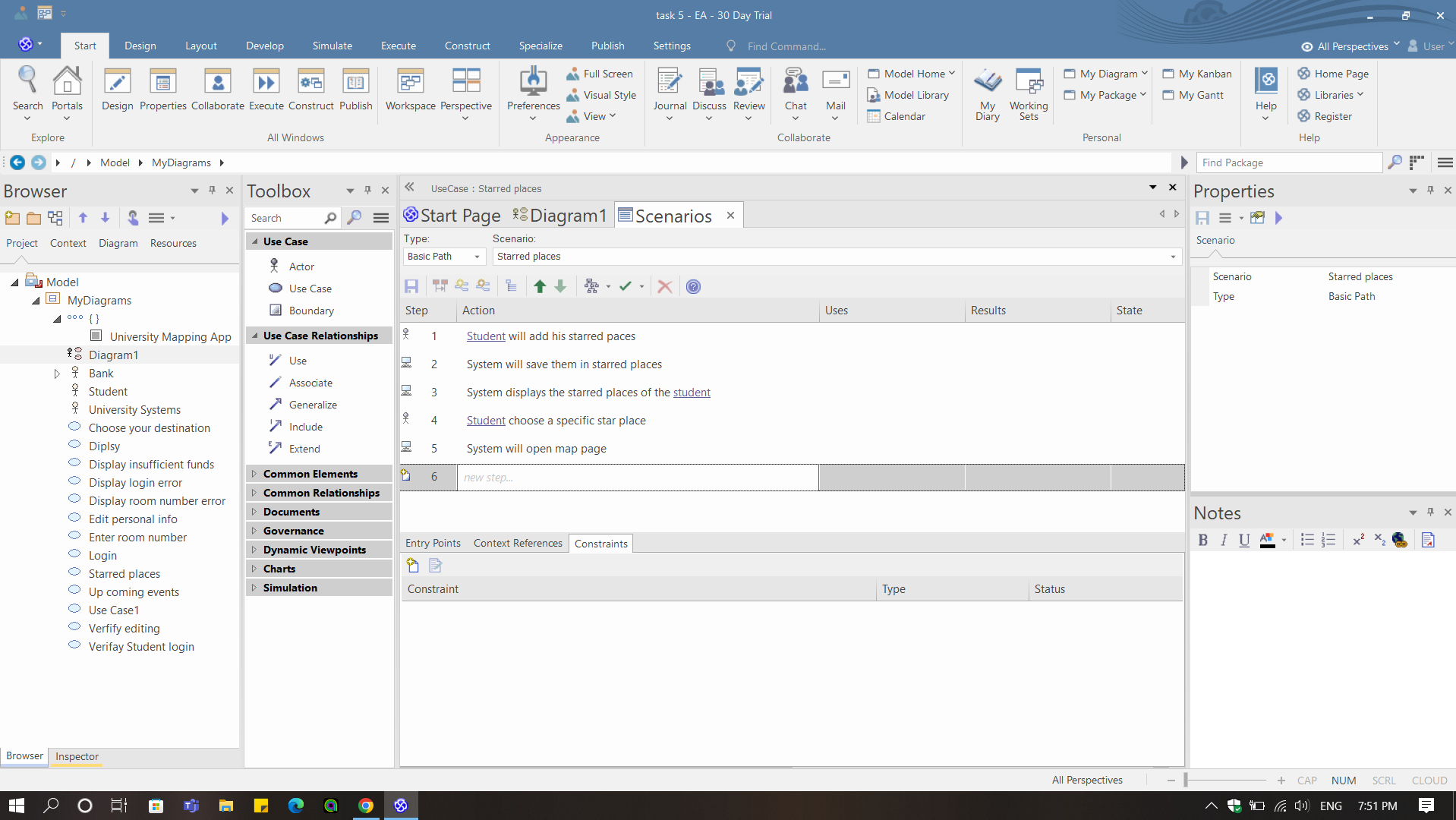


6.6. Enter room number exception scenario



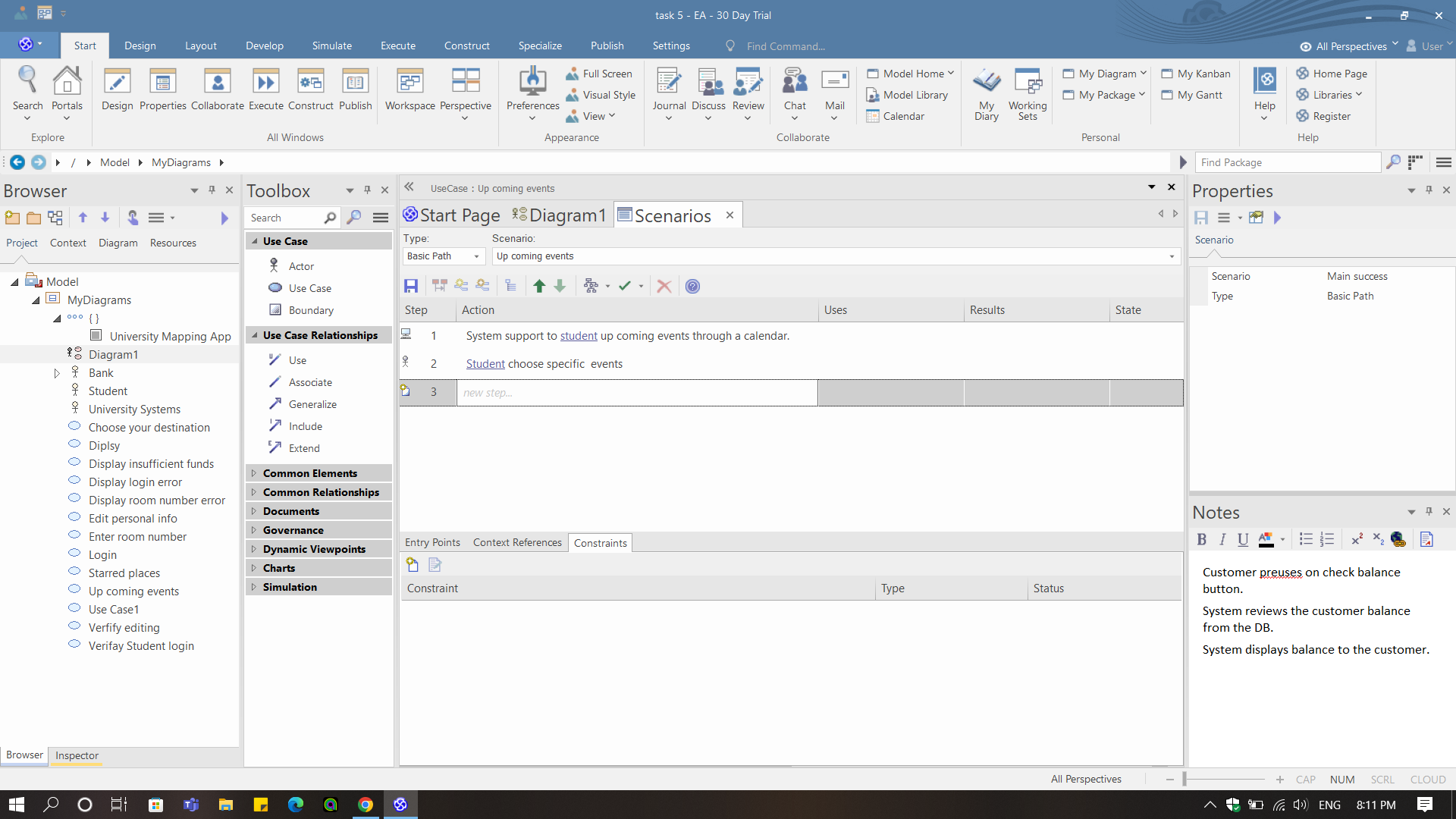
|  |  |
| --- | --- |
| ***Use case Name*** | ***Enter room number.*** |
| ***Actors*** | ***Student and University system.*** |
| ***Main success scenario:*** | ***Student enters his room number.***  ***University system will go to the map page.*** |
| ***Exceptions*** | ***Student enters a wrong room number.***  ***University system displays “This room does not exist”.***  ***University system displays “Re-enter a valid room number”.***  ***Student will enter a valid room number.*** |
| ***Pre-condition*** | ***Specific building is chosen (building 1 or building 2).*** |
| ***Post condition*** | ***The map page of the room.*** |

6.7. Starred places scenario



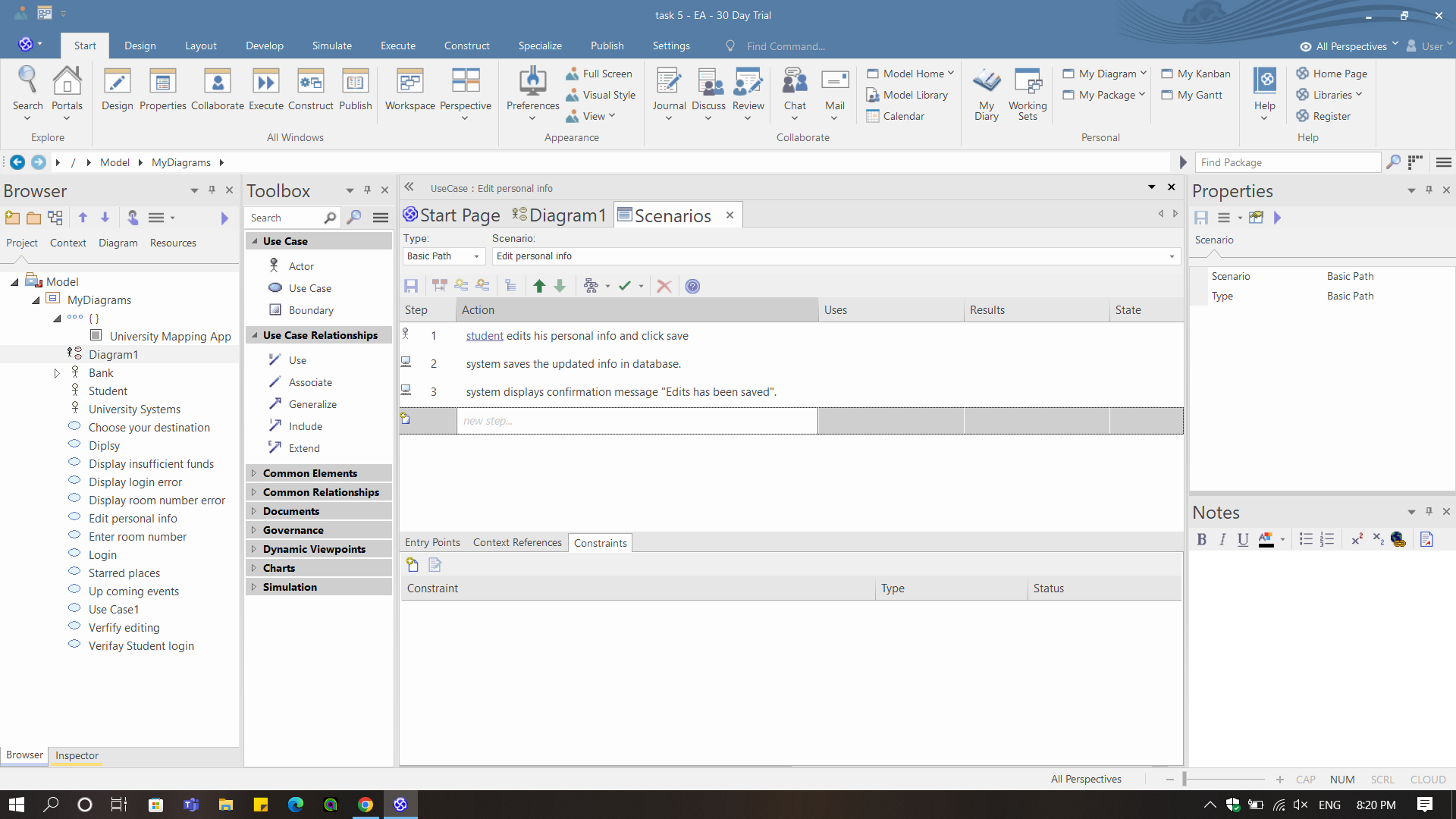
|  |  |
| --- | --- |
| ***Use case Name*** | ***Starred places.*** |
| ***Actors*** | ***Student and University system.*** |
| ***Main success scenario:*** | ***Student will add his starred places.***  ***University system will save them in starred places.***  ***University system displays the starred places of the student.***  ***Student chooses a specific star place.***  ***University system will open map page.*** |
| ***Exceptions*** | ***No exception.*** |
| ***Pre-condition*** | ***Student must have starred places.*** |
| ***Post condition*** | ***The map page is displayed.*** |

6.8. Upcoming events scenario



|  |  |
| --- | --- |
| ***Use case Name*** | ***Upcoming events.*** |
| ***Actors*** | ***Student and University system.*** |
| ***Main success scenario:*** | ***University system supports the student’s upcoming events through a calendar.***  ***Student chooses specific events.*** |
| ***Exceptions*** | ***No exception.*** |
| ***Pre-condition*** | ***University System must be linked with outlook.*** |
| ***Post condition*** | ***Student can fill the form of any event he wants to attend.*** |

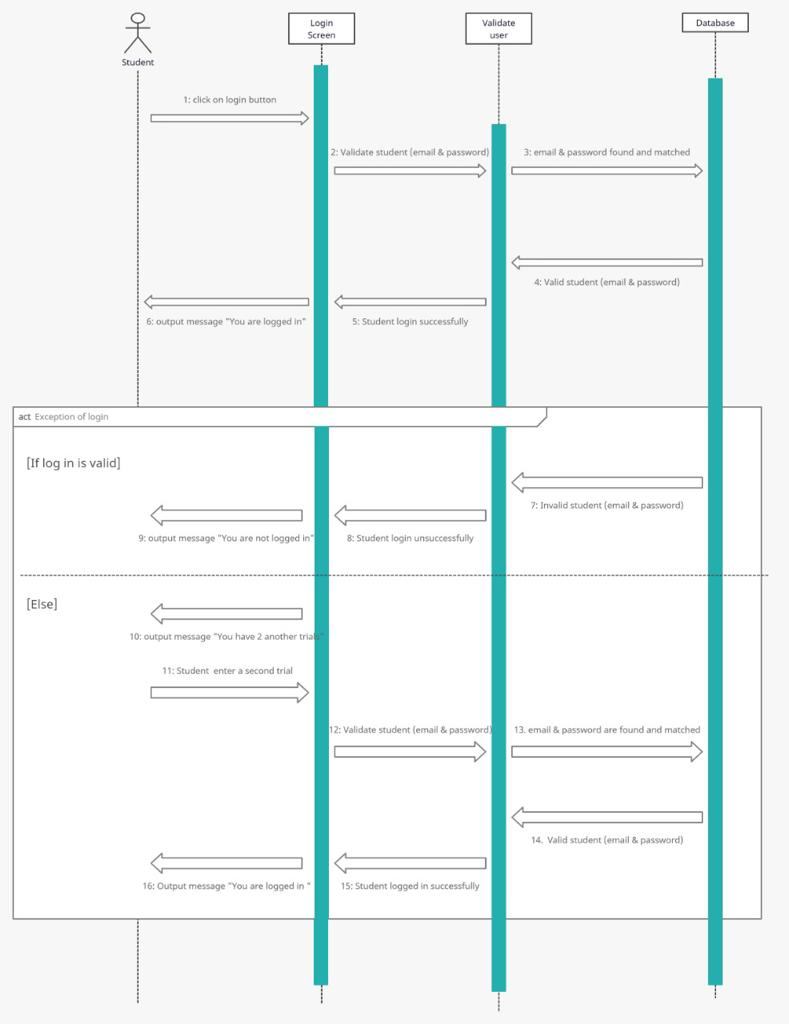
6.9. Edit personal information scenario



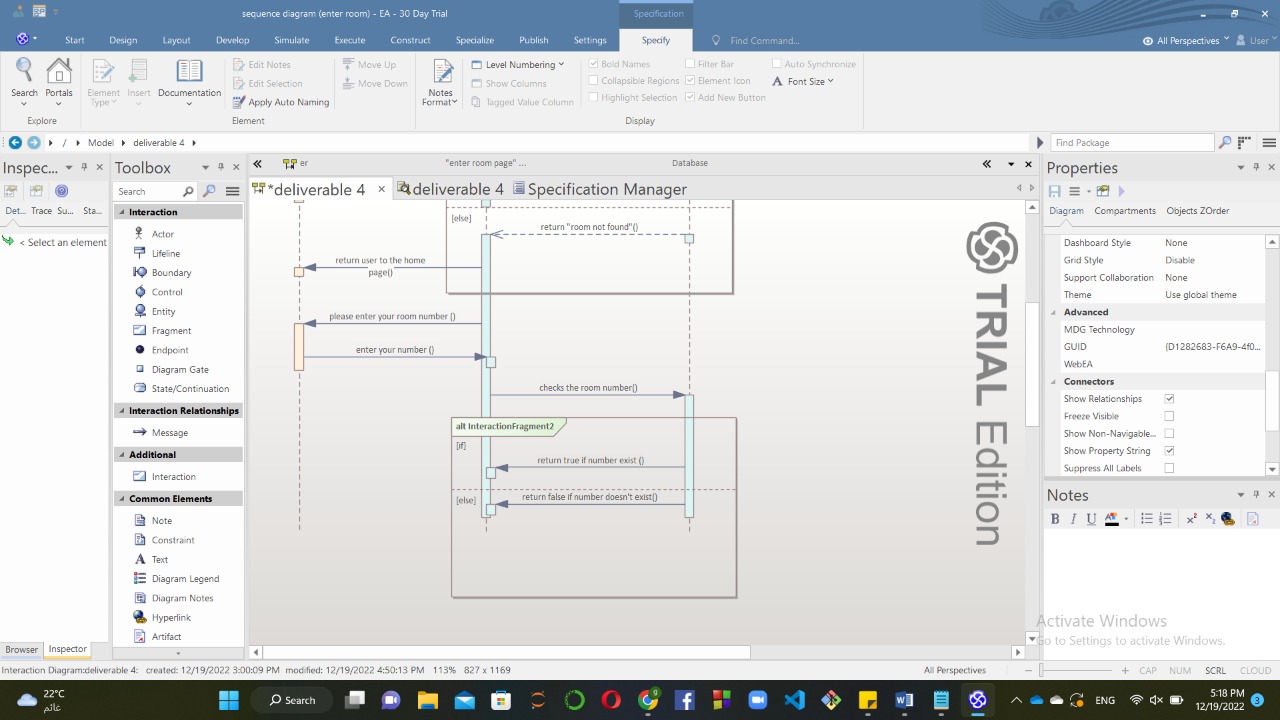
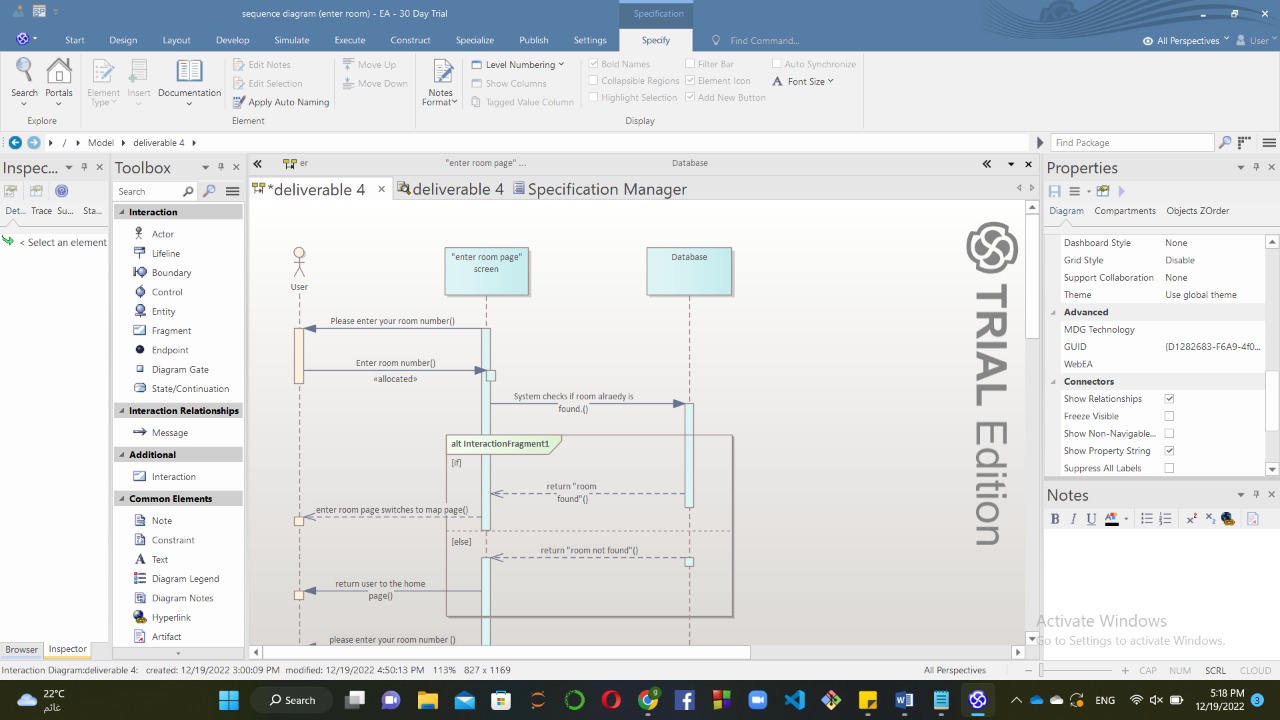
|  |  |
| --- | --- |
| ***Use case Name*** | ***Edit personal information.*** |
| ***Actors*** | ***Student and University system.*** |
| ***Main success scenario:*** | ***Student edits his personal info and click save.***  ***University system saves the updated info in database.***  ***University system displays confirmation message "Edits has been saved".*** |
| ***Exceptions*** | ***No exception.*** |
| ***Pre-condition*** | ***Student logged in.*** |
| ***Post condition*** | ***Student fills his personal information.*** |

**7.0. Sequence Diagrams**

7.1. Login Sequence Diagrams



7.2. Enter room sequence diagram



Scrum video link:

https://nileuniversity-my.sharepoint.com/:v:/g/personal/r\_amgad\_nu\_edu\_eg/EfGrPW7PRKJBukFewi1RHJIBxMku8-8wt\_lMzzzjF7vaxw?e=ha6hU7