**3.4.2 Instructor Use Cases**

## Use Case 1

|  |
| --- |
| **Use Case Name:** Create New Lecture |
| **Participant actor instances:** Instructor |
| **Flow of events:**   1. This event starts with Instructor’s click to the Create a new lecture button. 2. Instructor is directed to the create lecture page. 3. Instructor fills the lecture information’s. 4. Instructor clicks done. 5. Lecture is added to the database. |
| **Entry Condition:** User should be logged on as instructor. |
| **Exit Condition:** Lecture is created. Or cancelled if user left page without click done |
| **Alternative flow:**  **4a.** Instructor lefts page without click done.   1. System cancelled function automatically. |

## Use Case 2

|  |
| --- |
| **Use Case Name:** Create Class |
| **Participant actor instances:** Instructor |
| **Flow of events:**   1. This event starts with Instructor’s click to the Create Class button. 2. Instructor is directed to the Create Class page. 3. Instructors clicks add student button and can continue add or delete. 4. Instructor clicks done 5. A class list is constructed and setted to the database. |
| **Entry Condition:** User should be logged on as instructor. And also should already create a lecture. |
| **Exit Condition:** A class is created. Or alternatively empty one is created. |
| **Alternative flow:**  **3a.** Instructor clicks done.  **1.** System created an empty class list. |

## Use Case 3

|  |
| --- |
| **Use Case Name:** Add student |
| **Participant actor instances:** Instructor |
| **Flow of events:**   1. This event starts with Instructor’s click to the add Student button after opened Class page. 2. System shows to Instructor fields to fill with student info. 3. Instructor fills fields with the student information which is taking course. 4. Instructor clicks done. 5. A new student added to class list and setted to the database. |
| **Entry Condition:** User should be logged on as instructor. And also should already create a lecture and class. |
| **Exit Condition:** A student is added to class list. |

## Use Case 4

|  |
| --- |
| **Use Case Name:** Delete student |
| **Participant actor instances:** Instructor |
| **Flow of events:**   1. This event starts with Instructor’s click to the delete Student button after opened Class page. 2. Instructor selects a student. 3. Instructor clicks delete student button. 4. Student is deleted from class list, database updated. |
| **Entry Condition:** User should be logged on as instructor. And also should already created a lecture and class. |
| **Exit Condition:** Selected student(s) is deleted from class list. |

## Use Case 5

|  |
| --- |
| **Use Case Name:** List student |
| **Participant actor instances:** Instructor |
| **Flow of events:**   1. This event starts with Instructor’s click to the list Student button after opened Class page. 2. System shows to Instructor class list. |
| **Entry Condition:** User should be logged on as instructor. And also should already create a lecture and class. |
| **Exit Condition:** A student is added to class list. |

## Use Case 6

|  |
| --- |
| **Use Case Name:** Create Exam |
| **Participant actor instances:** Instructor |
| **Flow of events:**   1. This flow starts with Instructor’s click to the create exam button. 2. System shows to Instructor fields to fill with exam info. 3. Instructor fills fields with the exam information which is date, time, etc. 4. Then Instructor adds questions. 5. Instructor clicks done. |
| **Entry Condition:** User should already created a lecture. |
| **Exit Condition:** An exam is created. |

## Use Case 7

|  |
| --- |
| **Use Case Name:** Add Question |
| **Participant actor instances:** Instructor |
| **Flow of events:**   1. This event starts with Instructor’s click to the add question button. 2. System directs Instructor to question adding page. 3. Instructor first selects a question type. 4. Instructor adds question itself. 5. Instructor adds answer(s). 6. Instructor clicks done. 7. System adds question to the exam and database. |
| **Entry Condition:** User should be created an exam. |
| **Exit Condition:** A question is added to exam. |
| **Alternative flow:**  **6a.** Instructor doesn’t click done and left page.  **1.** System cancelled add student function. |

## Use Case 8

|  |
| --- |
| **Use Case Name:** Delete Question |
| **Participant actor instances:** Instructor |
| **Flow of events:**   1. This event starts with Instructor’s click to the delete question button. 2. System directs Instructor to exam page. 3. Instructor first selects a question. 4. Instructor clicks delete button. 5. System deletes question from the exam and database. |
| **Entry Condition:** User should be created an exam. |
| **Exit Condition:** A question is deleted from exam. |

## Use Case 9

|  |
| --- |
| **Use Case name:** Update Exam |
| **Participant actor instances:** Instructor |
| **Flow of events:**   1. This event starts with Instructor’s click to the update exam button. 2. System directs Instructor to exam page. 3. Instructor updates exam information’s (date, time etc.). 4. Instructor clicks done. 5. System updates exam information and database. |
| **Entry Condition:** User should be created an exam. |
| **Exit Condition:** Exam is updated. |

## Use Case 10

|  |
| --- |
| **Use Case Name:** Update Question |
| **Participant actor instances:** Instructor |
| **Flow of events:**   1. This event starts after Instructor’s selecting a question. 2. System opens Instructor to question page. 3. Instructor clicks selects update question. 4. Instructor updates question or answer(s). 5. Instructor clicks done. 6. System updates question and database. |
| **Entry Condition:** User should be created an exam. |
| **Exit Condition:** Question is updated. |

## Use Case 11

|  |
| --- |
| **Use Case Name:** Remove Exam |
| **Participant actor instances:** Instructor |
| **Flow of events:**   1. This event starts after Instructor clicks remove exam button. 2. System deletes exam from lecture and database. |
| **Entry Condition:** User should be created an exam. |
| **Exit Condition:** Exam is removed. |

## Use Case 12

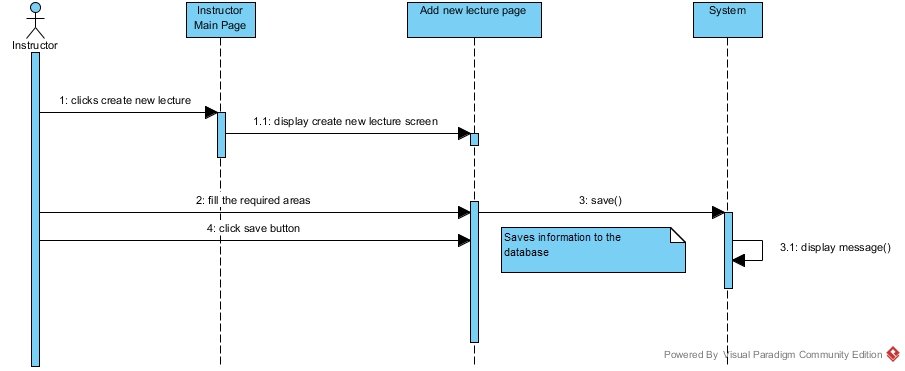
|  |
| --- |
| **Use Case Name:** Remove Lecture |
| **Participant actor instances:** Instructor |
| **Flow of events:**   1. Flow starts when Instructor clicks to the remove lecture button. 2. System removes lecture from Instructor’s account and database. |
| **Entry Condition:** User should be created a lecture. |
| **Exit Condition:** Lecture is removed, database changed. |

## Use Case 13

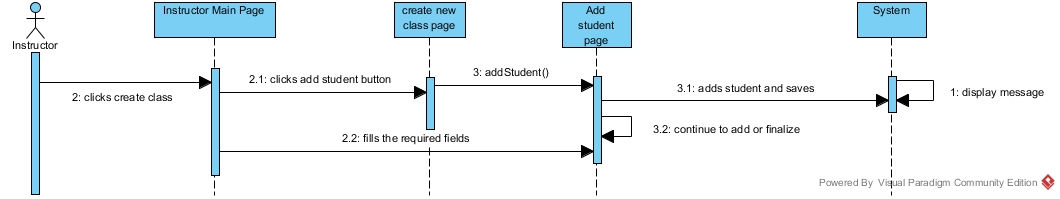
|  |
| --- |
| **Use Case Name:** List students results |
| **Participant actor instances:** Instructor |
| **Flow of events:**   1. Flow starts with Instructor’s click to show results button. 2. System opens the student grade list for the lecture to Instructor. |
| **Entry Condition:** Instructor should be logged in. |
| **Exit Condition:** Instructor has seen grade list of the exam. |

**3.4.4 Dynamic Model**

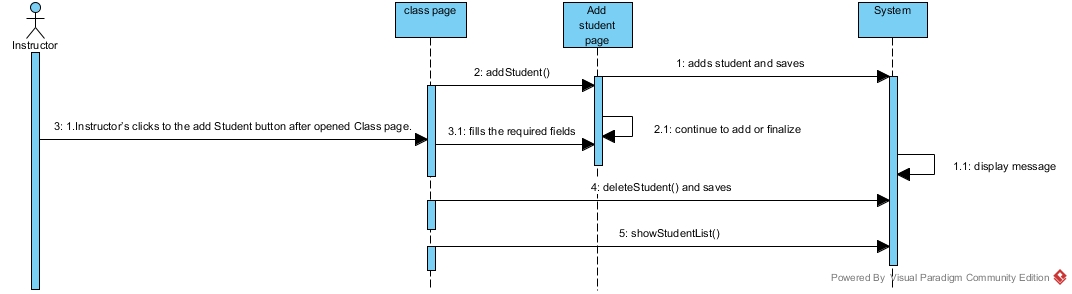
1. **Create New Lecture**
   1. Instructor main page is boundary entity.
   2. Add new lecture page is boundary entity.
   3. System is controller entity.



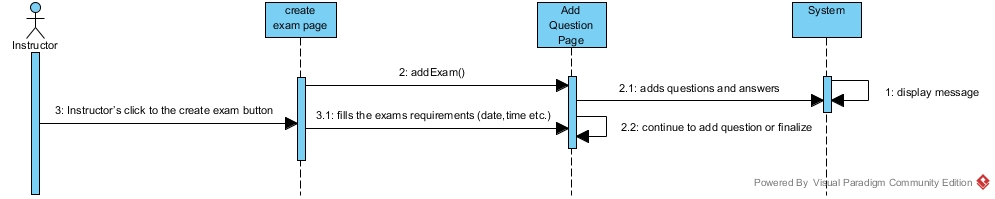
1. **Create Class**
   1. Instructor main page is boundary entity.
   2. Create new class page is boundary entity.
   3. Add student page is boundary entity.
   4. System is controller entity.



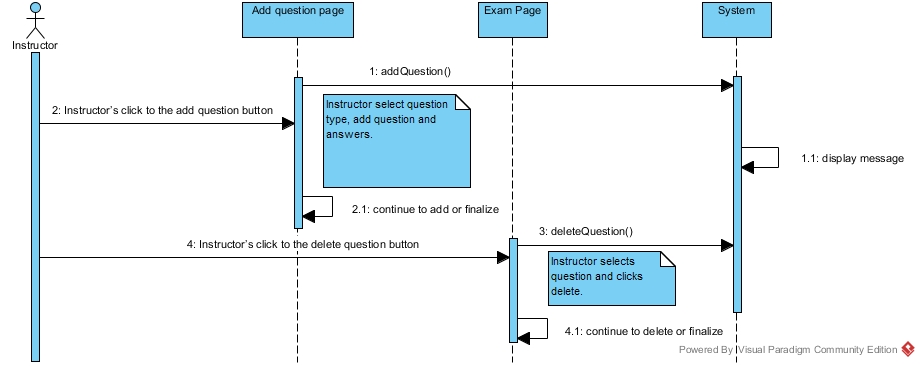
1. **Add, delete and show student**
   1. Class page is boundary entity.
   2. Add student page is boundary entity.
   3. System is controller entity.



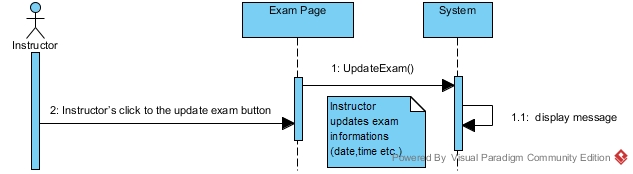
1. **Create Exam**
   1. Create exam page is boundary entity.
   2. Add question page is boundary entity.
   3. System is controller entity.



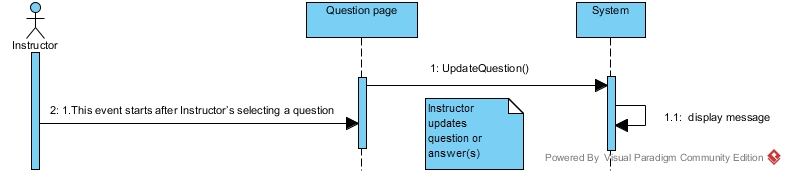
1. **Add and delete question**
   1. Add question page is boundary entity.
   2. Exam page is boundary entity.
   3. System is controller entity.



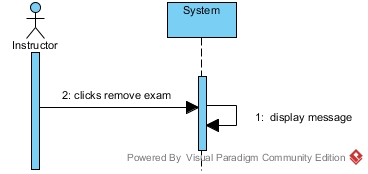
1. **Update Exam**
   1. Exam page is boundary entity.
   2. System is controller entity.



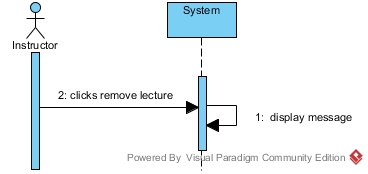
1. **Update Question**
   1. Question page is boundary entity.
   2. System is controller entity.



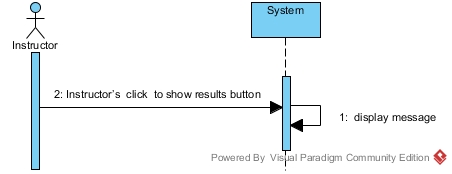
1. **Remove exam**
   1. System is controller entity.



1. **Remove lecture**
   1. System is controller entity**.**



1. **List students results**
   1. System is controller entity.



**3.4.3. Activity Diagram for Instructor**

