Fetch the first Question (UC-C1)

•Actor: MVC Client.

• Brief description: This use case describes how the system can request for Start new exam and get first question from the web API server.

• Pre-conditions: System request for start new exam and get question the first from web API.

• Post-condition: web API save exam information in the database and Apple question algorithm.

• Flow of event:

|  |  |
| --- | --- |
| System | OES Web API |
| 1. Request for new exam”. |  |
|  | 2 Select exam information |
|  | 3. Save the exam information in the database. {student\_id, subject\_id} |
|  | 4.Apply Question Algorithm to select first question. |
|  | 5.send the first question  Refer to the start of the exam and register its information. |

Critical scenario:

1.The Web API server stopped.

2.No connection to Web API server

No Interface Prototype.

Correct and fetch the next question(UC-C2)

• Actor: MVC Client.

•Brief description: This use case describes how the system can request for next question from the web API server.

• Pre-conditions: System request for the next question from web API.

• Post-condition: Web API server correct student answer, save it in the database and get the next question.

Flow of event:

|  |  |
| --- | --- |
| System | Web API |
| 1. Pass student answer with question id. |  |
|  | 2.Connet to database and get the question true answer. |
|  | 3.Apply auto correction algorithm to Correct this answer only |
|  | 4.Save the information in the database |
| 5.Request for next question. |  |
|  | 6.Randomly selects from the Student Questions list, appropriate question Depending on question algorithm. |

Critical scenario:

1.The Web API server stopped.

2.No connection to Web API server.

No Interface Prototype.

Fetch the final Result (UC-C3)

 Actor: MVC Client.

Brief description: This use case describes how the system get student final result from the web API server and stop the exam.

Pre-conditions: System request for the final result from web API.

Post-condition: web API server end the exam and calculate the final result and save it in the database.

Flow of event:

|  |  |
| --- | --- |
| System | Web API |
| 1. Request for final result”. |  |
|  | 2. Server connect to database to get student individually answers. |
|  | 3. Apply Auto correction algorithm to calculate the final result. |
|  | 4.Save the student final result in the Database. |
|  | 5.Send the final result with details to the system. |

Figure 27: Fetch the final Result flow of events

Critical scenario:

The web API server stopped.

No connection to web API server.

No User Interface Prototype.

Get Available view (UC-C4)

 Actor: MVC Client.

Brief description: This use case describes how the system can request for the view that Customized for each user.

Pre-conditions: System request for View from web API.

Post-condition: Web API return view Determined according to User login information.

Flow of event:

|  |  |
| --- | --- |
| System | Web API |
| 1. Request for View”. |  |
|  | 2. Check login information |
|  | 3. According to User Role  The API select Appropriate view |
|  | 4.Return view |

Critical scenario:

Error with login information

No User Interface Prototype