**SUMMARIES OF SCENARIO**



Figure 1. Use Case Diagram Summary representing the activities of the Track Student Performance System.

|  |  |
| --- | --- |
| **Use case name:** | Summary of Track Student Performance System |
| **Area:** | Track Student Performance System |
| **Actors:** | Users: Faculty, Administrative Support Staff |
| **Description:** | Use Case Diagram Summary representing the activities of the Track Student Performance System:   * Administrative support staff: can upload (via web) assessment documents every term, a standard report can be generated every term. * Faculty: can easily run new queries on items of interest. |
| **Stakeholder:** | Institution: Student, Faculty, Administrative Support Staff |
| **Steps Performed**   1. Administrative support staff uploads (via web) assessment documents every term. 2. Administrative support staff generates standard report every term. | |
| **Preconditions:** | Users have logged on to the system. |
| **Post conditions:** | Users have successfully logged in. |
| **Success Guarantee:** | Users have successfully processed on the system. |
| **Minimum Guarantee:** | Users can access to the system |

* **Use case “Generate Standard Report”:**



Figure . Use Case shown to Generate Standard Report in the system

|  |  |
| --- | --- |
| **Use case name:** | Generate Standard Report |
| **Area:** | Track Student Performance System |
| **Actors:** | Administrative Support Staff |
| **Description:** | Administrative support staff: Wants to access the system and Generate Standard Report for every term. |
| **Stakeholder:** | Institution: Administrative Support Staff |
| **Steps Performed**   1. An Administrative Support Staff logs in to the system. 2. The user selects Generate Standard Report function. 3. The system redirects to the Generate Standard Report function. 4. The user selects a term or selects a section. 5. The user selects Generate Standard Report. 6. System checks for input condition validity. 7. If valid then generate the report. 8. If invalid then prompt user with an error. | |
| **Preconditions:** | The user is logged in the system. |
| **Post conditions:** | The user has successfully valid conditions. |
| **Success Guarantee:** | The user has successfully generated stand report. |
| **Minimum Guarantee:** | The user has successfully logged in to the system. |

* **Use case “Upload Assessment Documents”:**



Figure . Use Case shown to Upload Assessment Documents in the system

|  |  |
| --- | --- |
| **Use case name:** | Upload Assessment Documents. |
| **Area:** | Track Student Performance System. |
| **Actors:** | Administrative Support Staff. |
| **Description:** | Administrative support staff: Wants to access the system and Upload Assessment Documents for every term. |
| **Stakeholder:** | Institution: Administrative Support Staff. |
| **Steps Performed**   1. Administrative Support Staff logs in to the system. 2. The user selects Upload Assessment Documents function. 3. The system redirects to the Upload Assessment Documents function. 4. The user selects a term or selects a section. 5. The user selects Upload Assessment Documents. 6. System checks input condition validity. 7. If valid then the system uploads the file and reports status. 8. If invalid then prompt user with an error. | |
| **Preconditions:** | The user is logged in the system. |
| **Post conditions:** | The user has successfully valid conditions. |
| **Success Guarantee:** | The user has successfully Upload Assessment Documents. |
| **Minimum Guarantee:** | The user has successfully logged in to the system. |

* **Use case “Run New Queries”:**



Figure . Use Case shown to Run New Queries in the system

|  |  |
| --- | --- |
| **Use case name:** | Run New Queries |
| **Area:** | Track Student Performance System |
| **Actors:** | Faculty |
| **Description:** | Faculty: Wants to access the system and run new queries for every section. |
| **Stakeholder:** | Institution: Faculty |
| **Steps Performed**   1. Faculty loges in to the system. 2. The user selects Run New Queries function. 3. The system redirects to the Run New Queries function. 4. The user selects function to create a question. 5. The user selects function to update a query and sumit it. 6. System checks input condition validity. 7. If valid then the system process to the query and show the result. 8. If invalid then show the errors. | |
| **Preconditions:** | The user is logged in the system. |
| **Post conditions:** | The user has successfully valid conditions. |
| **Success Guarantee:** | The user has successfully run the query. |
| **Minimum Guarantee:** | The user has successfully logged in to the system. |

* **Use case “Make Program Educational Objective”:**



Figure . Use Case shown to Make Program Educational Objective in the system

|  |  |
| --- | --- |
| **Use case name:** | Make Program Educational Objective |
| **Area:** | Track Student Performance System |
| **Actors:** | Faculty |
| **Description:** | Faculty: Wants to access the system and Make Program Educational Objective for every section. |
| **Stakeholder:** | Institution: Faculty |
| **Steps Performed**   1. Faculty logs in to the system. 2. The user selects Make Program Educational Objective function. 3. The system redirects to the Make Program Educational Objective function. 4. The user selects function to create a knowledge topic. 5. The user selects function to create a Student Learning Outcome. 6. The user selects function to create a Program Educational Objective. 7. System checks input condition validity. 8. If valid then the system process to create Program Educational Objective. 9. If invalid then show the errors. | |
| **Preconditions:** | The user is logged in the system. |
| **Post conditions:** | The user has successfully valid conditions. |
| **Success Guarantee:** | The user has successfully created Program Educational Objective. |
| **Minimum Guarantee:** | The user has successfully logged in to the system. |