

Use-case specification and activity diagram (Activity Diagram)

1. Login (For all users)

| | |
|-------------------------|---|
| Use-case name | Login |
| Description | User must log in to the system with the account provided by the Admin in order to use the system's features. |
| Trigger | Users click to button "Log in" |
| Pre-condition | User must have a valid account in the system. |
| Post-condition | <ul style="list-style-type: none">– User receives a notification about the successful login.– User is granted access to the system and can perform actions according to their role and permissions. |
| Basic flow | <ol style="list-style-type: none">1. The System displays the login page with input fields for the user's username and password.2. The user enters their username/email and password and clicks on the "Log In" button.3. The system validates email & password in correct format.4. The system verifies the user's credentials in database and grants access to the system.5. The system displays successful login notification and redirects the user to homepage. |
| Alternative flow | <p>3a. If the user enters incorrect credentials:</p> <ul style="list-style-type: none">– System displays an error message and prompts the user to re-enter their credentials. <p>Use-case continue at step 2.</p> |

| | |
|-----------------------|--|
| | <p>1a. If the user does not have an account:</p> <ul style="list-style-type: none"> – Users click “Contact” button then system display some basic contact information of organization for the user. – User must contact the Manager to have the issue resolve. <p>Use-case continue at step 1.</p> |
| Exception flow | |

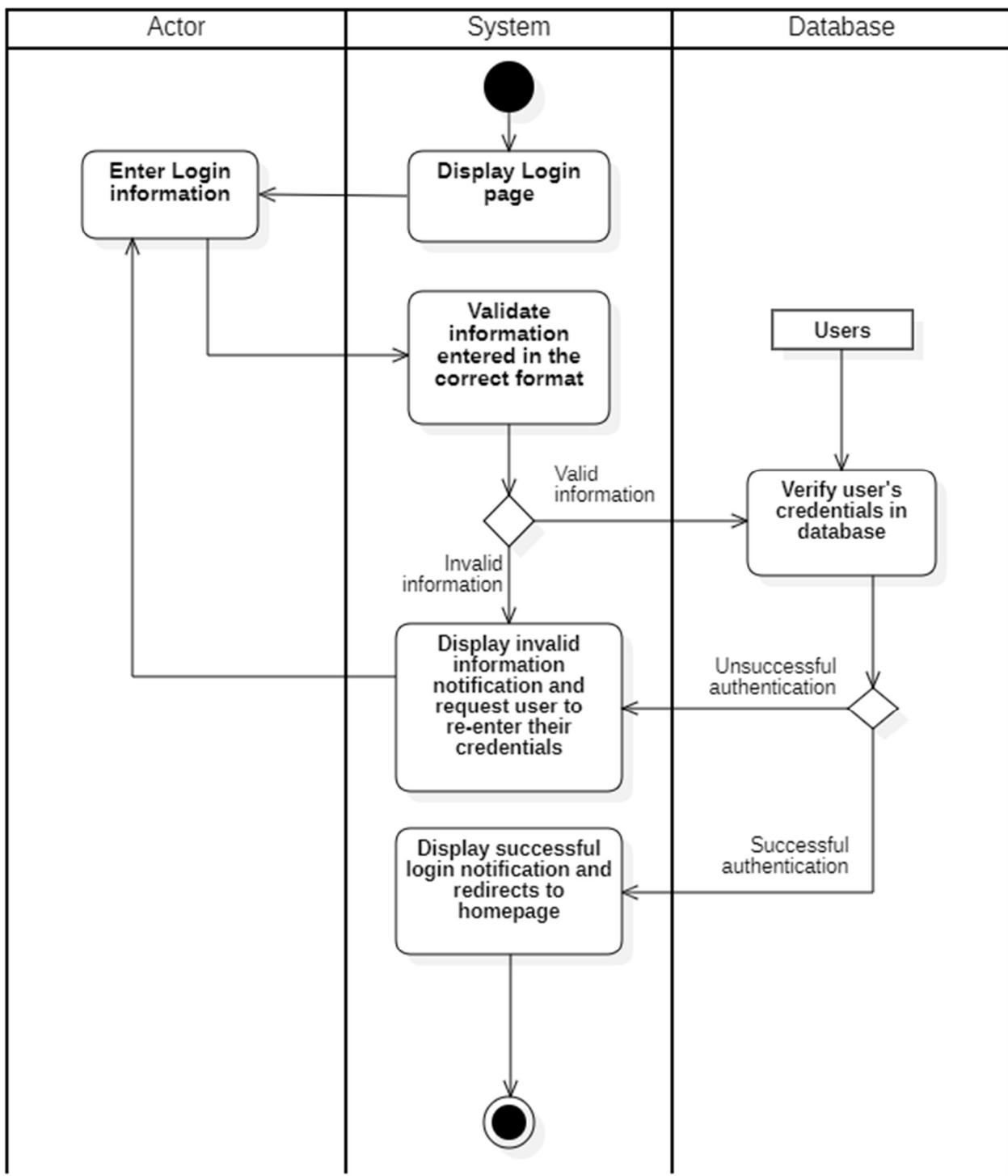


Figure 3. 5. Activity Diagram for Use case “Login”

2. Manage Courses

| | |
|-----------------------|--|
| Use-case name | Manage Courses |
| Description | Manager can insert, update, delete and search Course information |
| Trigger | Manager clicks to “Courses” menu at homepage |
| Pre-condition | Manager’s status is signed in |
| Post-condition | <ul style="list-style-type: none"> – Course’s information successfully updated in database. – Data displayed on the Course management page must be updated with the latest data. |
| Basic flow | <ol style="list-style-type: none"> 1. System displays courses management page 2. Manager chooses one function: insert/update/delete/search course’s information <ul style="list-style-type: none"> • <i>Insert</i> 3. System displays course information page 4. Manager enter course informations & submit 5. System validates course information in correct format 6. System validates course information with existed data in database 7. System stores course information in database 8. System displays successful notification and displays Courses management page <ul style="list-style-type: none"> • <i>Update</i> 9. Manager chooses one course to be updated 10. System displays course information page 11. Manager enter course informations & submit 12. System validates course information in correct format 13. System validates course information with existed data in database 14. System stores course information in database |

| | |
|-------------------------|--|
| | <p>15. System display successful notification and displays Courses management page</p> <ul style="list-style-type: none"> • <i>Delete</i> <p>16. Choose course need to be delete</p> <p>17. Display confirm dialog then manager Confirm delete action</p> <p>18. If Manager confirm delete, System process delete request. Else, cancel action.</p> <p>19. System validates no external data required course information and selected item existed in database</p> <p>20. System delete course information from database</p> <p>21. System displays successful notification and display course management page with the latest data.</p> <ul style="list-style-type: none"> • <i>Search</i> <p>22. Manager select search function and enter keywords.</p> <p>23. System processes searching data</p> <p>24. System queries data base on given keywords</p> <p>25. System filtered data at course management page (data related to keyword is retained)</p> |
| Alternative flow | <p>5a. System display invalid info due to wrong format. Use-case continue at step 4.</p> <p>6a. System display invalid info because data had been existed in the database. Use-case continue at step 4.</p> <p>12a. System display invalid info due to wrong format. Use-case continue at step 11.</p> <p>13a. System display invalid info because data had been existed in the database. Use-case continue at step 11.</p> |
| Exception flow | |

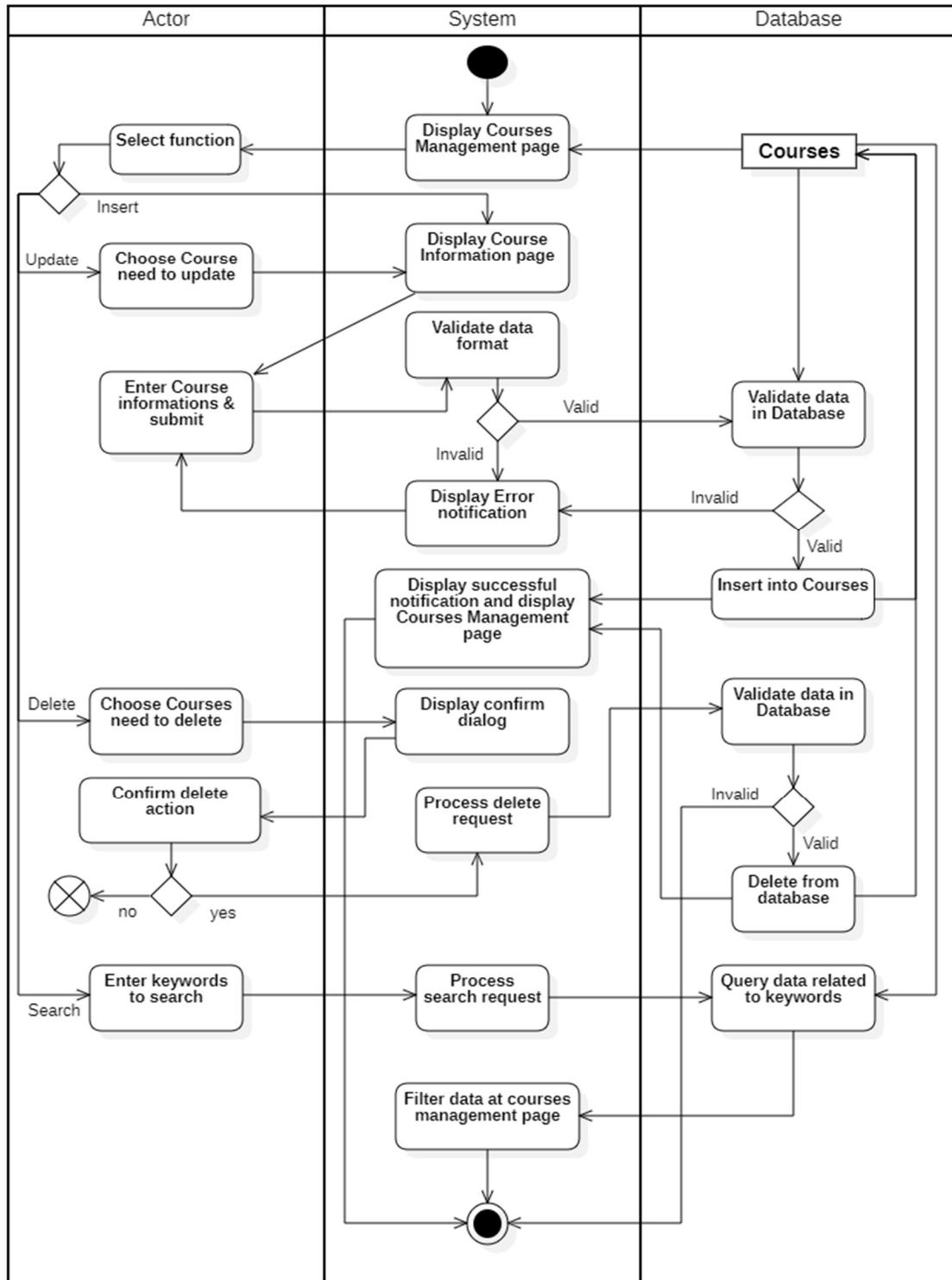


Figure 3. 6. Activity Diagram for Use case “Manage Courses”

2.1. Manage Chapters

| | |
|-----------------------|---|
| Use-case name | Manage Chapters |
| Description | Manager can insert, update, delete and search Chapter information In a specific course |
| Trigger | Manager select a Course in Courses management page then clicks to “Manage Chapters” |
| Pre-condition | <ul style="list-style-type: none"> – Manager's status is signed in – Manager has previously accessed Course management menu and selected a specific course. |
| Post-condition | <ul style="list-style-type: none"> – Chapters information successful updated in database. – Data displayed on the Chapters management page must be updated with the latest data |
| Basic flow | <ol style="list-style-type: none"> 1. Manager Select a Course and choose manage chapters 2. System displays chapters management page 3. Manager chooses one function: insert/update/delete/search chapter's information <ul style="list-style-type: none"> • Insert 4. System displays chapter information page 5. Manager enter chapter informations & submit 6. System validates chapter information in correct format 7. System validates chapter information with existed data in database 8. System stores chapter information in database 9. System display successful notification and displays Chapters management page <ul style="list-style-type: none"> • Update 10. Manager chooses one chapter to be updated |

| | |
|-------------------------|--|
| | <p>11. System displays chapter information page</p> <p>12. Manager enter chapter informations & submit</p> <p>13. System validates chapter information in correct format</p> <p>14. System validates chapter information with existed data in database</p> <p>15. System stores chapter information in database</p> <p>16. System display successful notification and displays chapters management page</p> <ul style="list-style-type: none"> • Delete <p>17. Choose chapter need to be delete</p> <p>18. Display confirm dialog then manager Confirm delete action</p> <p>19. If Manager confirm delete, System process delete request. Else, cancel action.</p> <p>20. System validates no external data required chapter information and selected item existed in database</p> <p>21. System delete chapter information from database</p> <p>22. System displays successful notification and display chapters management page with the latest data.</p> <ul style="list-style-type: none"> • Search <p>23. Manager select search function and enter keywords.</p> <p>24. System processes searching data</p> <p>25. System queries data base on given keywords</p> <p>26. System filtered data at course management page (data related to keyword is retained)</p> |
| Alternative flow | <p>6a. System display invalid info due to wrong format. Use-case continue at step 5.</p> <p>7a. System display invalid info because data had been existed in the</p> |

| | |
|-----------------------|---|
| | <p>database. Use-case continue at step 5.</p> <p>13a. System display invalid info due to wrong format. Use-case continue at step 12.</p> <p>14a. System display invalid info because data had been existed in the database. Use-case continue at step 12.</p> |
| Exception flow | |

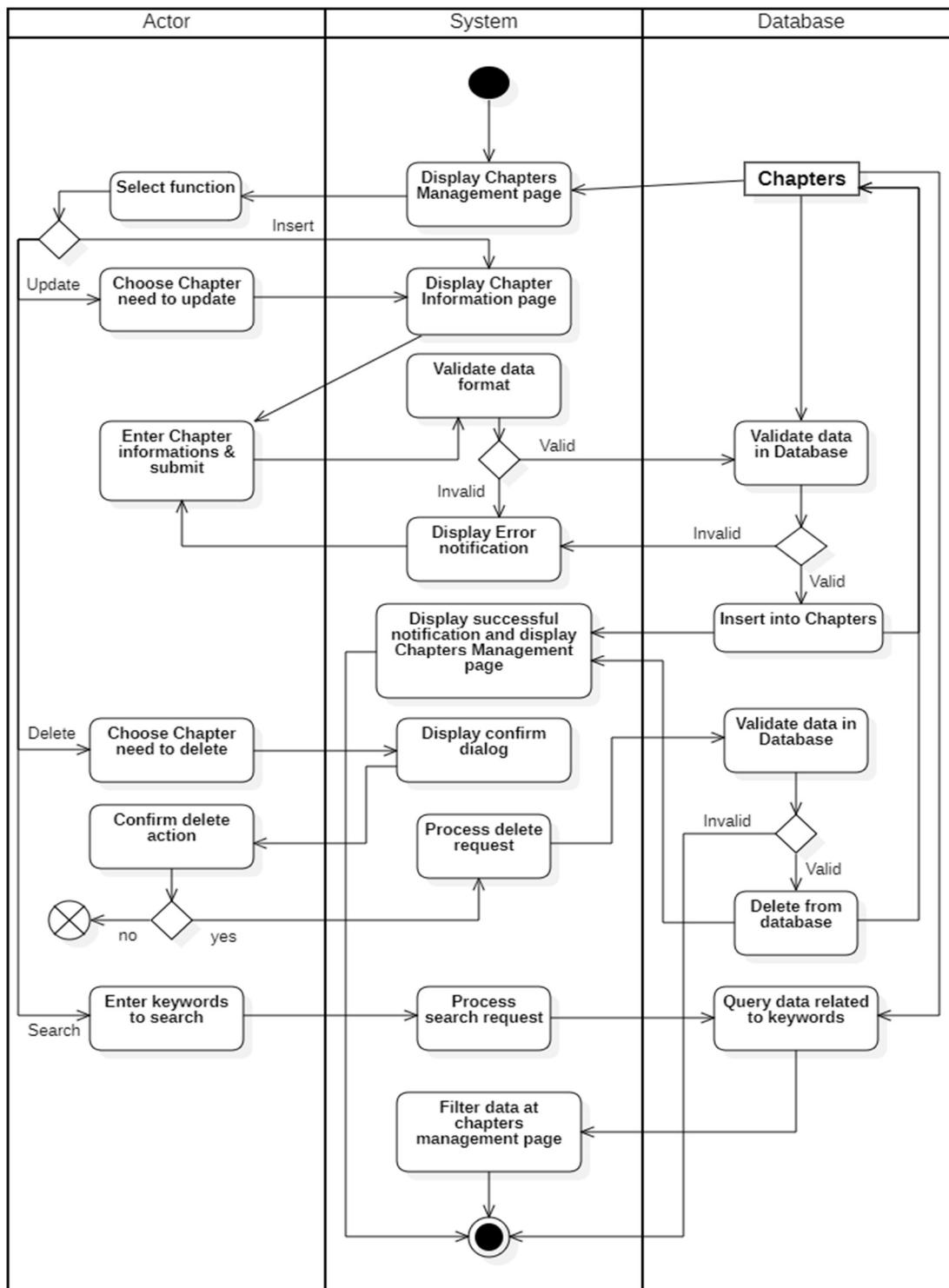


Figure 3. 7. Activity Diagram for Use case “**Manage Chapters**”

3. Manage Classes

| | |
|-----------------------|--|
| Use-case name | Manage Classes |
| Description | Manager can insert, update, delete and search Classes information |
| Trigger | Manager clicks to “Classes” menu at homepage |
| Pre-condition | Manager’s status is signed in |
| Post-condition | <ul style="list-style-type: none"> – Classes information successfully updated in database. – Data displayed on the Classes management page must be updated with the latest data |
| Basic flow | <ol style="list-style-type: none"> 1. System displays classes management page 2. Manager chooses one function: insert/update/delete/search class’s information <ul style="list-style-type: none"> • <i>Insert</i> 3. System displays class information page 4. Manager enter class informations & submit 5. System validates class information in correct format 6. System validates class information with existed data in database 7. System stores class information in database 8. System displays successful notification and displays Classes management page <ul style="list-style-type: none"> • <i>Update</i> 9. Manager chooses one class to be updated 10. System displays class information page 11. Manager enter class informations & submit 12. System validates class information in correct format 13. System validates class information with existed data in database 14. System stores class information in database |

| | |
|-------------------------|--|
| | <p>15. System display successful notification and displays Classes management page</p> <ul style="list-style-type: none"> • <i>Delete</i> <p>16. Choose class need to be delete</p> <p>17. Display confirm dialog then manager Confirm delete action</p> <p>18. If Manager confirm delete, System process delete request. Else, cancel action.</p> <p>19. System validates no external data required class information and selected item existed in database</p> <p>20. System delete class information from database</p> <p>21. System displays successful notification and display Classes management page with the latest data.</p> <ul style="list-style-type: none"> • <i>Search</i> <p>22. Manager select search function and enter keywords.</p> <p>23. System processes searching data</p> <p>24. System queries data base on given keywords</p> <p>25. System filtered data at course management page (data related to keyword is retained)</p> |
| Alternative flow | <p>5a. System display invalid info due to wrong format. Use-case continue at step 4.</p> <p>6a. System display invalid info because data had been existed in the database. Use-case continue at step 4.</p> <p>12a. System display invalid info due to wrong format. Use-case continue at step 11.</p> <p>13a. System display invalid info because data had been existed in the database. Use-case continue at step 11.</p> |

| | |
|-----------------------|--|
| Exception flow | |
|-----------------------|--|

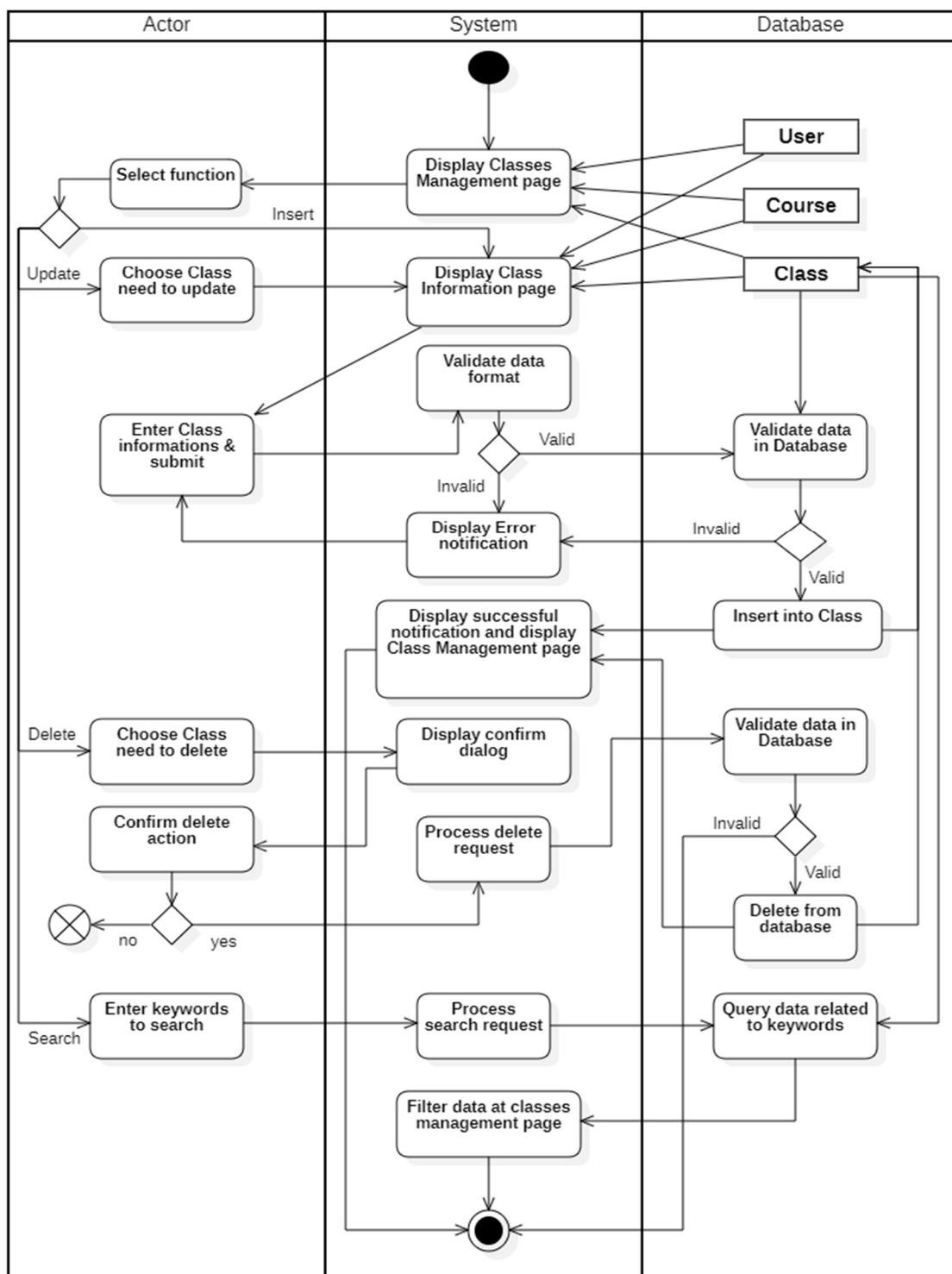


Figure 3. 8. Activity Diagram for Use case “Manage Classes”

3.1. Manage Lessons

| | |
|-----------------------|--|
| Use-case name | Manage Lessons |
| Description | Manager can insert, update, delete and search Lessons information In a specific Class |
| Trigger | Manager select a Class in Classes management page then clicks to “Manage Lessons” |
| Pre-condition | <ul style="list-style-type: none"> – Manager's status is signed in – Manager has previously accessed Classes management menu and selected a specific class. |
| Post-condition | <ul style="list-style-type: none"> – Lessons information successful updated in database. – Data displayed on the Lessons management page must be updated with the latest data |
| Basic flow | <ol style="list-style-type: none"> 1. Manager Select a Class and choose manage Lessons 2. System displays Lessons management page 3. Manager chooses one function: insert/update/delete/search Lesson information <ul style="list-style-type: none"> • Insert 4. System displays lesson information page 5. Manager enter lesson informations & submit 6. System validates lesson information in correct format 7. System validates lesson information with existed data in database 8. System stores lesson information in database 9. System display successful notification and displays Lessons management page <ul style="list-style-type: none"> • Update 10. Manager chooses one lesson to be updated |

| | |
|-------------------------|--|
| | <p>11. System displays lesson information page</p> <p>12. Manager enter lesson informations & submit</p> <p>13. System validates lesson information in correct format</p> <p>14. System validates lesson information with existed data in database</p> <p>15. System stores lesson information in database</p> <p>16. System display successful notification and displays Lessons management page</p> <ul style="list-style-type: none"> • Delete <p>17. Choose lesson need to be delete</p> <p>18. Display confirm dialog then manager Confirm delete action</p> <p>19. If Manager confirm delete, System process delete request. Else, cancel action.</p> <p>20. System validates no external data required lesson information and selected item existed in database</p> <p>21. System delete lesson information from database</p> <p>22. System displays successful notification and display Lessons management page with the latest data.</p> <ul style="list-style-type: none"> • Search <p>23. Manager select search function and enter keywords.</p> <p>24. System processes searching data</p> <p>25. System queries data base on given keywords</p> <p>26. System filtered data at course management page (data related to keyword is retained)</p> |
| Alternative flow | <p>6a. System display invalid info due to wrong format. Use-case continue at step 5.</p> <p>7a. System display invalid info because data had been existed in the database. Use-case continue at step 5.</p> |

| | |
|-----------------------|---|
| | <p>13a. System display invalid info due to wrong format. Use-case continue at step 12.</p> <p>14a. System display invalid info because data had been existed in the database. Use-case continue at step 12.</p> |
| Exception flow | |

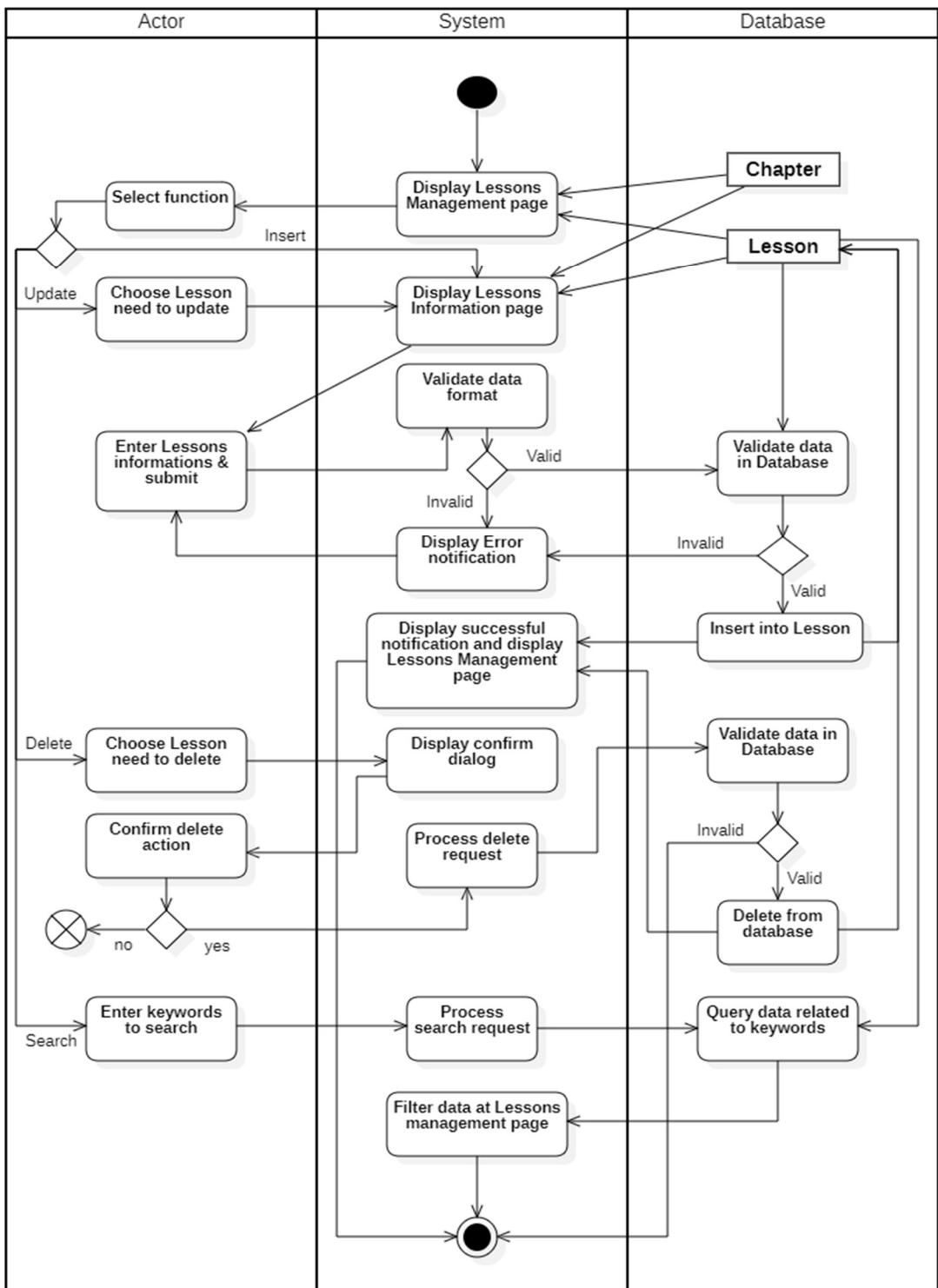


Figure 3. 9. Activity Diagram for Use case “Manage Lessons”

3.1.1. Manage Materials (For Teacher and Manager)

| | |
|-----------------------|--|
| Use-case name | Manage Materials |
| Description | User can upload, download, delete and search Material information In a specific Lesson |
| Trigger | User select “Materials” tab after select a Lesson in Lessons Management page |
| Pre-condition | <ul style="list-style-type: none"> – User’s status is signed in – User has previously accessed Lessons management page and selected a specific lesson. |
| Post-condition | <ul style="list-style-type: none"> – Materials information successful updated in database. – Data displayed on the Materials management page must be updated with the latest data |
| Basic flow | <ol style="list-style-type: none"> 1. User Select a lesson and choose Materials tab. 2. System displays Materials management page 3. User chooses one function: upload/download/delete/search Material <ul style="list-style-type: none"> • Upload 4. System displays Material page 5. User upload files/materials from device 6. System validates files information in correct format 7. System validates material information with existed data in database 8. System stores material information in database 9. System display successful notification and displays Materials management page <ul style="list-style-type: none"> • Delete 10. Choose material need to be delete 11. Display confirm dialog then User Confirm delete action |

| | |
|-------------------------|--|
| | <p>12. If User confirm delete, System process delete request. Else, cancel action.</p> <p>13. System validates no external data required material information and selected item existed in database</p> <p>14. System delete material information from database</p> <p>15. System displays successful notification and display Materials management page with the latest data.</p> <ul style="list-style-type: none"> • Search <p>16. User select search function and enter keywords.</p> <p>17. System processes searching data</p> <p>18. System queries data base on given keywords</p> <p>19. System filtered data at course management page (data related to keyword is retained)</p> <ul style="list-style-type: none"> • Download <p>20. User Choose material need to download</p> <p>21. System Get material path from database</p> <p>22. System Display Save file dialog</p> <p>23. User Choose destination directory</p> <p>24. System Process download request</p> |
| Alternative flow | <p>6a. System display invalid info due to wrong format. Use-case continue at step 5.</p> <p>7a. System display invalid info because data had been existed in the database. Use-case continue at step 5.</p> |
| Exception flow | |

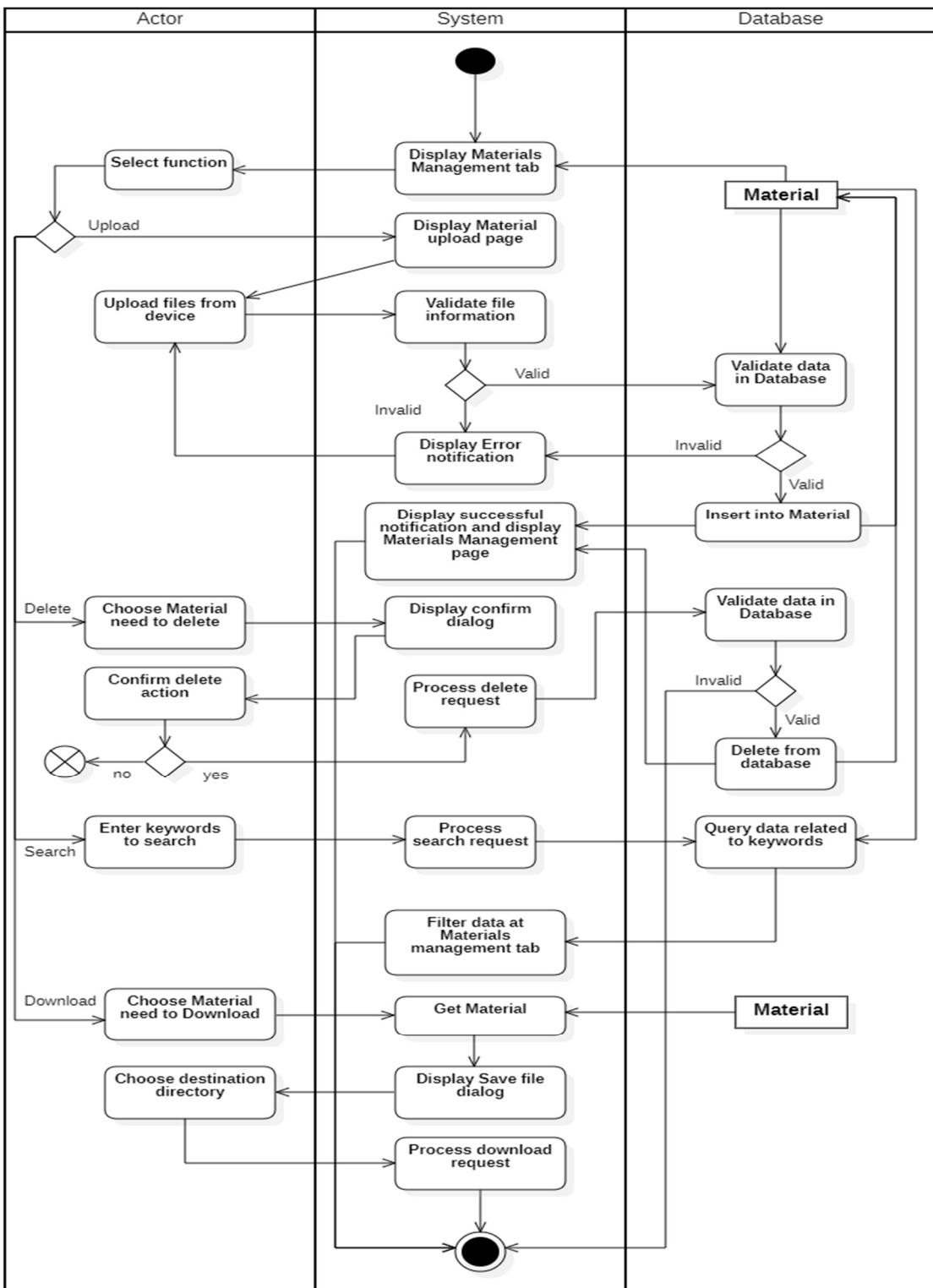


Figure 3. 10. Activity Diagram for Use case “**Manage Materials**”

3.1.2. Manage Attendances (For Teacher and Manager)

| | |
|-----------------------|---|
| Use-case name | Manage Attendances |
| Description | User can insert, update, delete and search Attendances information in a specific Lesson |
| Trigger | User select “Attendances” tab after select a Lesson in Lessons Management page |
| Pre-condition | <ul style="list-style-type: none"> – User’s status is signed in – User has previously accessed Lessons management page and selected a specific lesson. |
| Post-condition | <ul style="list-style-type: none"> – Attendances information successful updated in database. – Data displayed on the Materials management page must be updated with the latest data |
| Basic flow | <ol style="list-style-type: none"> 1. User Select a lesson and choose Attendances tab. 2. System displays Attendances management tab with absentee table and presentee table 3. User Select a student from absentee table or presentee table <ul style="list-style-type: none"> • Mark present 4. System move selected student from absentee table to presentee table 5. System insert selected student into database <ul style="list-style-type: none"> • Mark absent 6. System move selected student from presentee table to absentee table 7. System delete selected student from database <ul style="list-style-type: none"> • Search 8. User select search function and enter keywords. 9. System processes searching data 10. System queries data base on given keywords |

| | |
|-------------------------|---|
| | 11. System filtered data at course management page (data related to keyword is retained) |
| Alternative flow | <p>2a. The page will initialize by default the first time you access this tab, the absent students table will be the list of students for that class and the students present table will be empty.</p> <p>2b. On subsequent visits, the absent and present tables will show data edited since the last visit.</p> |
| Exception flow | |

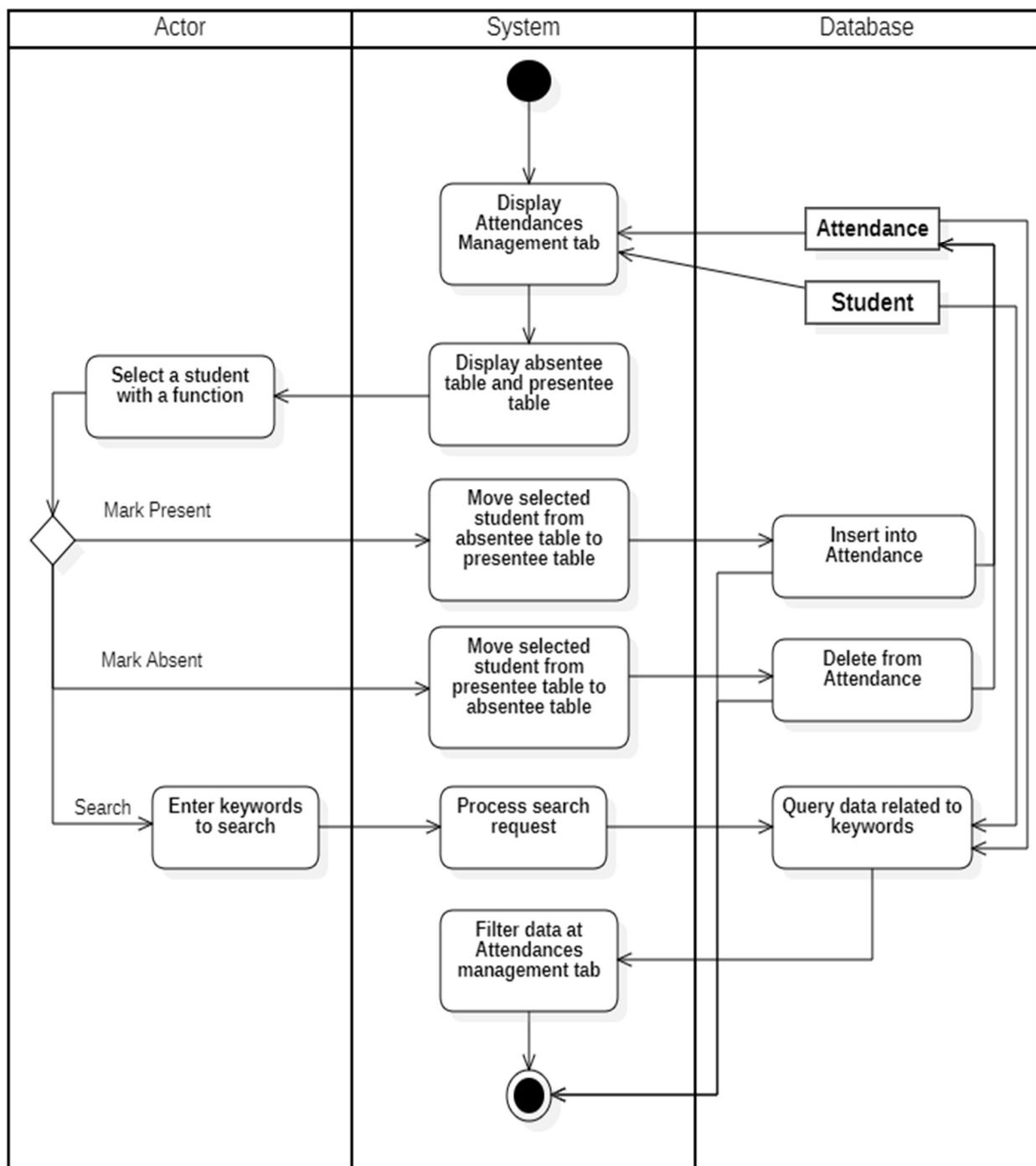


Figure 3. 11. Activity Diagram for Use case “**Manage Attendances**”

4. Manage Classrooms

| | |
|-----------------------|--|
| Use-case name | Manage Classrooms |
| Description | Manager can insert, update, delete, search and view schedule in a specific Classrooms |
| Trigger | Manager clicks to “Rooms” menu at homepage |
| Pre-condition | Manager’s status is signed in |
| Post-condition | <ul style="list-style-type: none"> – Classrooms information successfully updated in database. – Data displayed on the Classrooms management page must be updated with the latest data |
| Basic flow | <ol style="list-style-type: none"> 1. System displays Classrooms management page 3. Manager chooses one function: insert/update/delete/search/view schedule classrooms <ul style="list-style-type: none"> • Insert 4. System displays classroom information page 5. Manager enters classroom information & submit 6. System validates classroom information in correct format 7. System validates classroom information with existed data in database 8. System stores classroom information in database 9. System displays successful notification and displays classrooms management page <ul style="list-style-type: none"> • Update 10. Manager chooses one classroom to be updated 11. System displays classroom information page 12. Manager enters classroom information & submit 13. System validates classroom information in correct format 14. System validates classroom information with existed data in |

| | |
|-------------------------|---|
| | <p>database</p> <p>15. System stores classroom information in database</p> <p>16. System display successful notification and displays classrooms management page</p> <ul style="list-style-type: none"> • Delete <p>17. Choose classroom need to be delete</p> <p>18. Display confirm dialog then manager Confirm delete action</p> <p>19. If Manager confirm delete, System process delete request. Else, cancel action.</p> <p>20. System validates no external data required classroom information and selected item existed in database</p> <p>21. System delete classroom information from database</p> <p>22. System displays successful notification and display classrooms management page with the latest data.</p> <ul style="list-style-type: none"> • Search <p>23. Manager select search function and enter keywords.</p> <p>24. System processes searching data</p> <p>25. System queries data base on given keywords</p> <p>26. System filtered data at classroom management page (data related to keyword is retained)</p> <ul style="list-style-type: none"> • View Schedule <p>27. Manager choose a classroom want to view schedule</p> <p>28. System get data from database and generate a schedule for selected room</p> <p>29. System display schedule as timetable with class information</p> |
| Alternative flow | 6a. System display invalid info due to wrong format. Use-case continue at step 5. |

| | |
|-----------------------|--|
| | <p>7a. System display invalid info because invalid in database. Use-case continue at step 5.</p> <p>13a. System display invalid info due to wrong format. Use-case continue at step 12.</p> <p>14a. System display invalid info because data had been existed in the database. Use-case continue at step 12.</p> |
| Exception flow | |

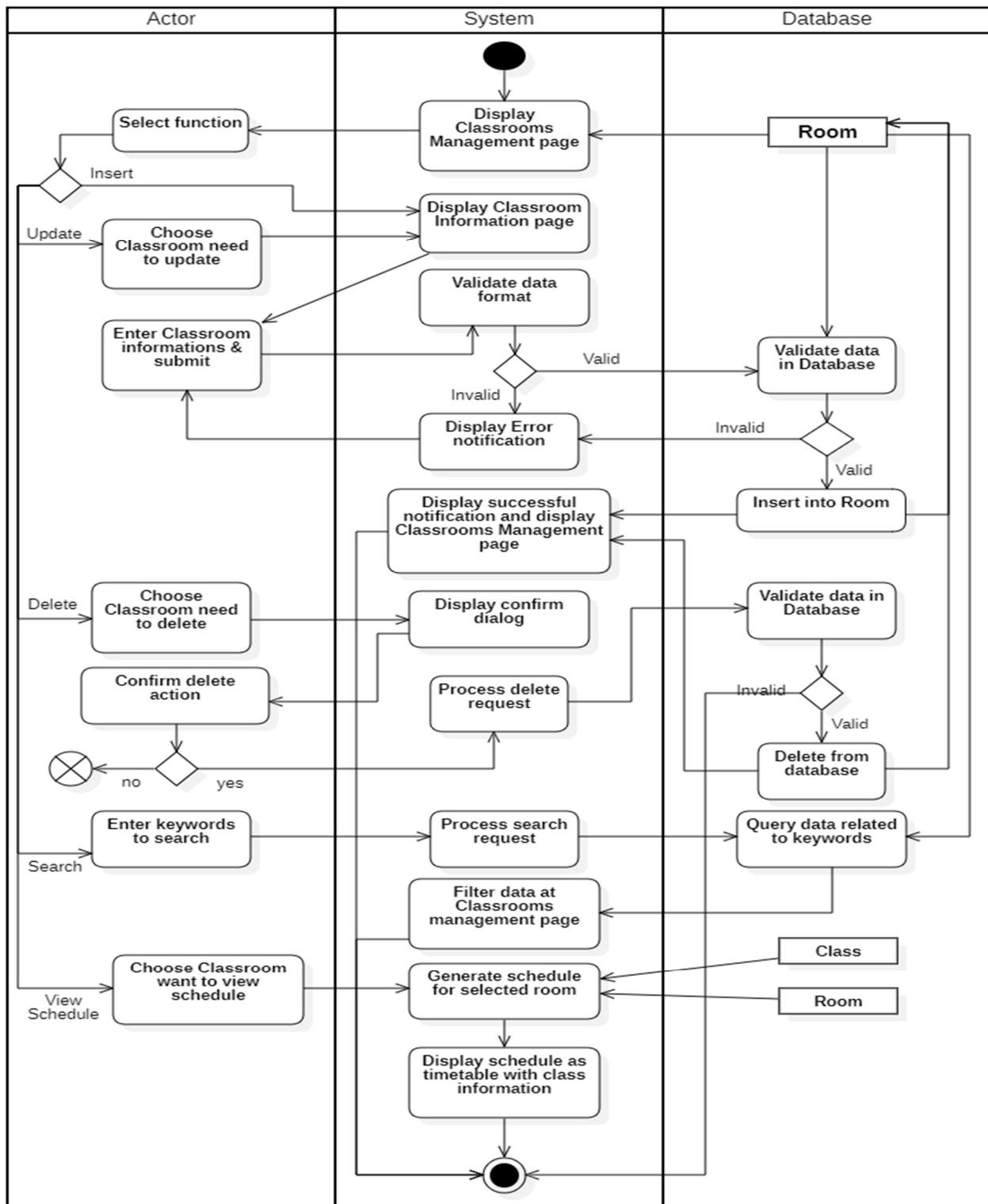


Figure 3. 12. Activity Diagram for Use case “Manage Classrooms”

5. Manage Users

| | |
|-----------------------|---|
| Use-case name | Manage Users |
| Description | Manager can insert, update, delete, search Users and Enroll user in a Class |
| Trigger | Manager clicks to “Users” menu at homepage |
| Pre-condition | Manager’s status is signed in |
| Post-condition | <ul style="list-style-type: none"> – Course’s information successfully updated in database. – Data displayed on the Course management page must be updated with the latest data. |
| Basic flow | <ol style="list-style-type: none"> 1. System displays Users management page 2. Manager chooses one function: insert/update/delete/search users and enroll in a class <ul style="list-style-type: none"> • <i>Insert</i> 3. System displays user information page 4. Manager enter user informations & submit 5. System validates user information in correct format 6. System validates user information with existed data in database 7. System stores user information in database 8. System displays successful notification and displays users management page <ul style="list-style-type: none"> • <i>Update</i> 9. Manager chooses one user to be updated 10. System displays user information page 11. Manager enter user informations & submit 12. System validates user information in correct format 13. System validates user information with existed data in database |

14. System stores user information in database
15. System display successful notification and displays users management page
- ***Delete***
16. Choose user need to be delete
17. Display confirm dialog then manager Confirm delete action
18. If Manager confirm delete, System process delete request. Else, cancel action.
19. System validates no external data required user information and selected item existed in database
20. System delete user information from database
21. System displays successful notification and display users management page with the latest data.
- ***Search***
22. Manager select search function and enter keywords.
23. System processes searching data
24. System queries data base on given keywords
25. System filtered data at user management page (data related to keyword is retained)
- ***Enroll in a Class***
26. Manager select a Student to enroll in a Class
27. System display of available classes
28. Manager choose a Class in the list
29. System display confirmation dialog
30. If Manager confirm enrollment, System process request. Else, cancel action.
31. System validate the enrollment in database
32. System displays successful notification

| | |
|-------------------------|--|
| Alternative flow | <p>5a. System display invalid info due to wrong format. Use-case continue at step 4.</p> <p>6a. System display invalid info because data had been existed in the database. Use-case continue at step 4.</p> <p>12a. System display invalid info due to wrong format. Use-case continue at step 11.</p> <p>13a. System display invalid info because data had been existed in the database. Use-case continue at step 11.</p> <p>31a. System display error notification because invalid in database. Use case continue at step 28.</p> |
| Exception flow | |

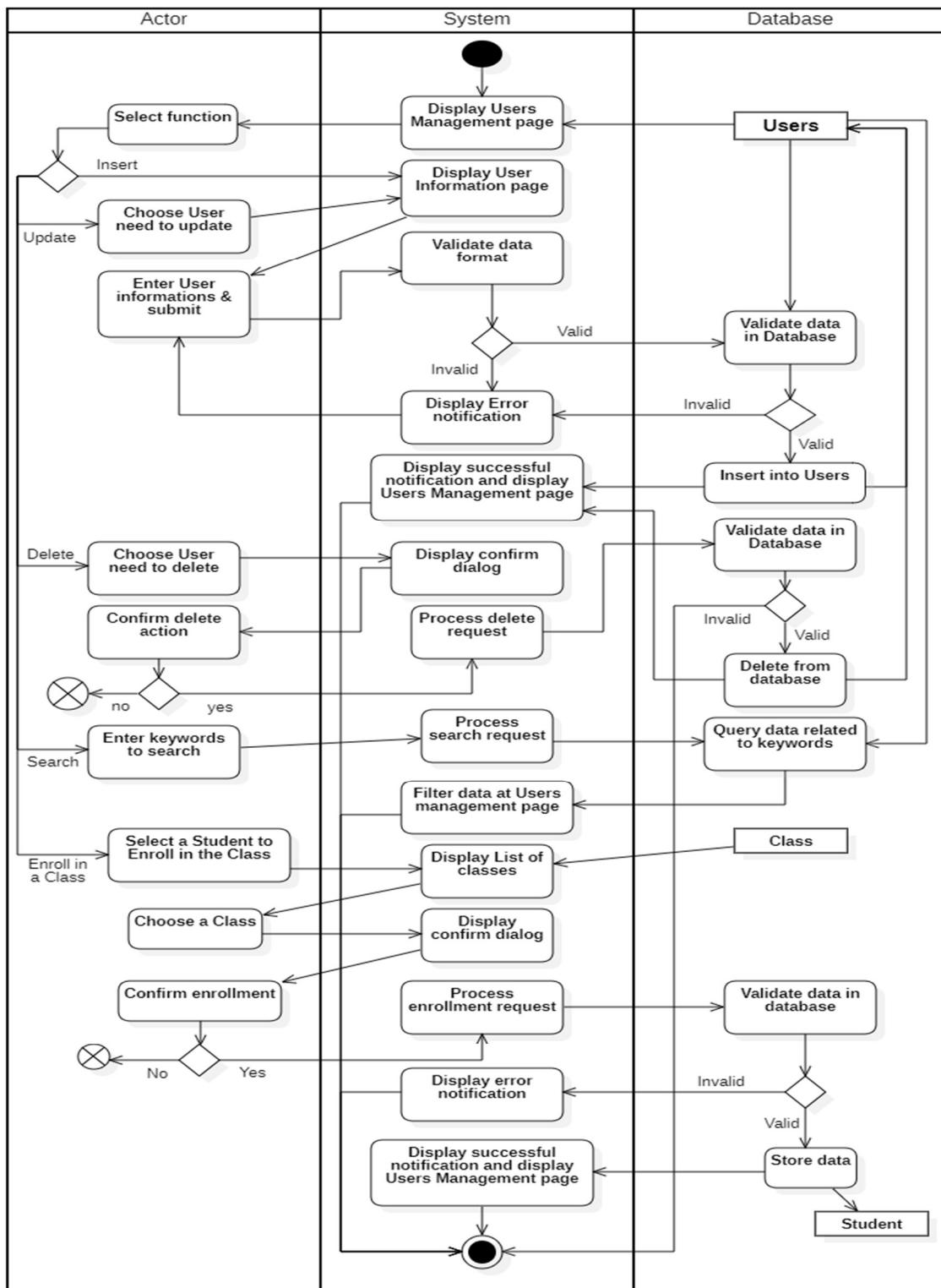


Figure 3. 13. Activity Diagram for Use case “Manage Users”

6. Export Class report

| | |
|-------------------------|---|
| Use-case name | Export class report |
| Description | Manager can export class report about class attendances |
| Trigger | Manager clicks to “Export report” button at Classes Management Page |
| Pre-condition | Manager’s status is signed in Manager was entered Classes Management Page |
| Post-condition | <ul style="list-style-type: none">● Class report export at destination directory successful● Class report contains suitable information |
| Basic flow | <ol style="list-style-type: none">1. System display list of Classes2. Manager choose a class in the list3. System generate Class report for selected Class4. System display Save file dialog for manager5. Manager choose file destination directory for the report6. System export the report to PDF into destination directory7. System display successful notification |
| Alternative flow | |
| Exception flow | |

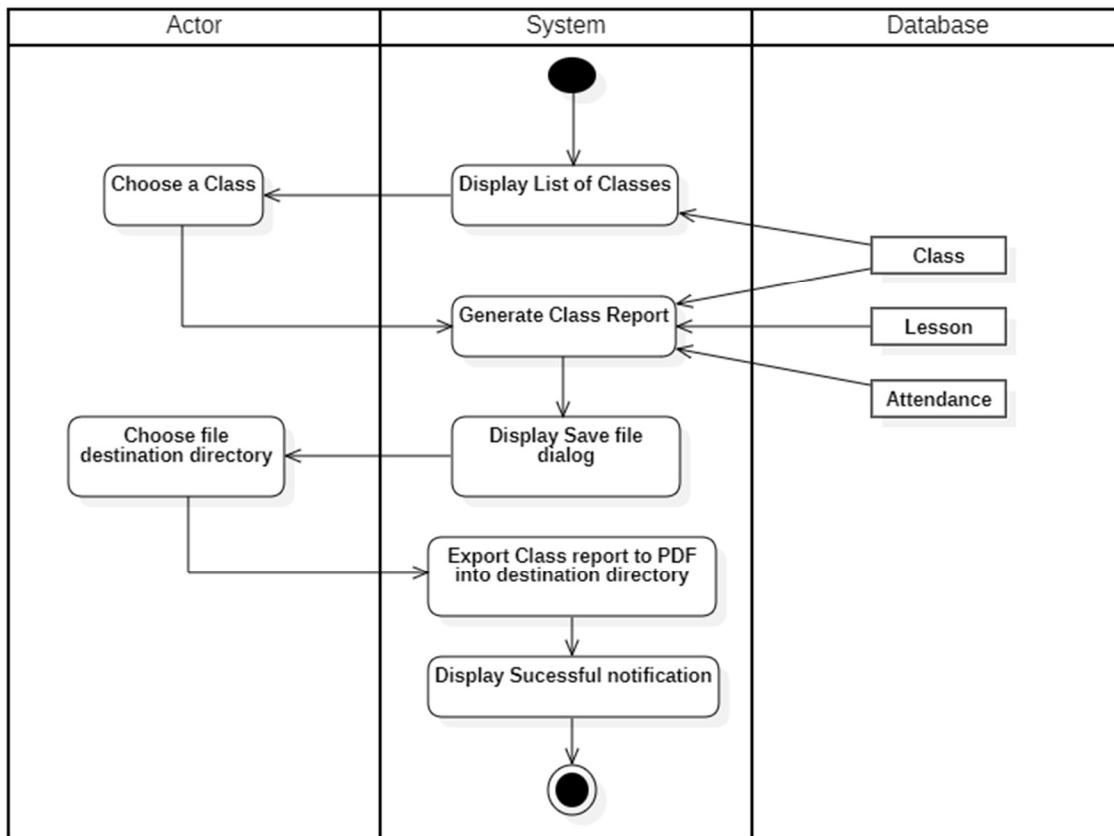


Figure 3. 14. Activity Diagram for Use case “**Export Class Report**”

7. Analyze

| | |
|-----------------------|--|
| Use-case name | Analyze |
| Description | Manager can analyze the performance of courses and classes through visual chart |
| Trigger | Manager clicks to “General” menu at homepage |
| Pre-condition | Manager’s status is signed in |
| Post-condition | Data & Chart display on General page with data queried from database |
| Basic flow | <ol style="list-style-type: none"> 1. System display dashboard page with some basic metric 2. Manager select a analysis category <ul style="list-style-type: none"> • <i>Number of students through lessons in a Class</i> 3. System display list of classes |

| | |
|-------------------------|--|
| | <p>4. Manager choose a class want to analyze</p> <p>5. System query data from database</p> <p>6. System generate and display visual chart base on data queried</p> <ul style="list-style-type: none"> <i>Number of classes of each course</i> <p>7. System query data from database</p> <p>8. System generate and display visual chart base on data queried</p> |
| Alternative flow | |
| Exception flow | |

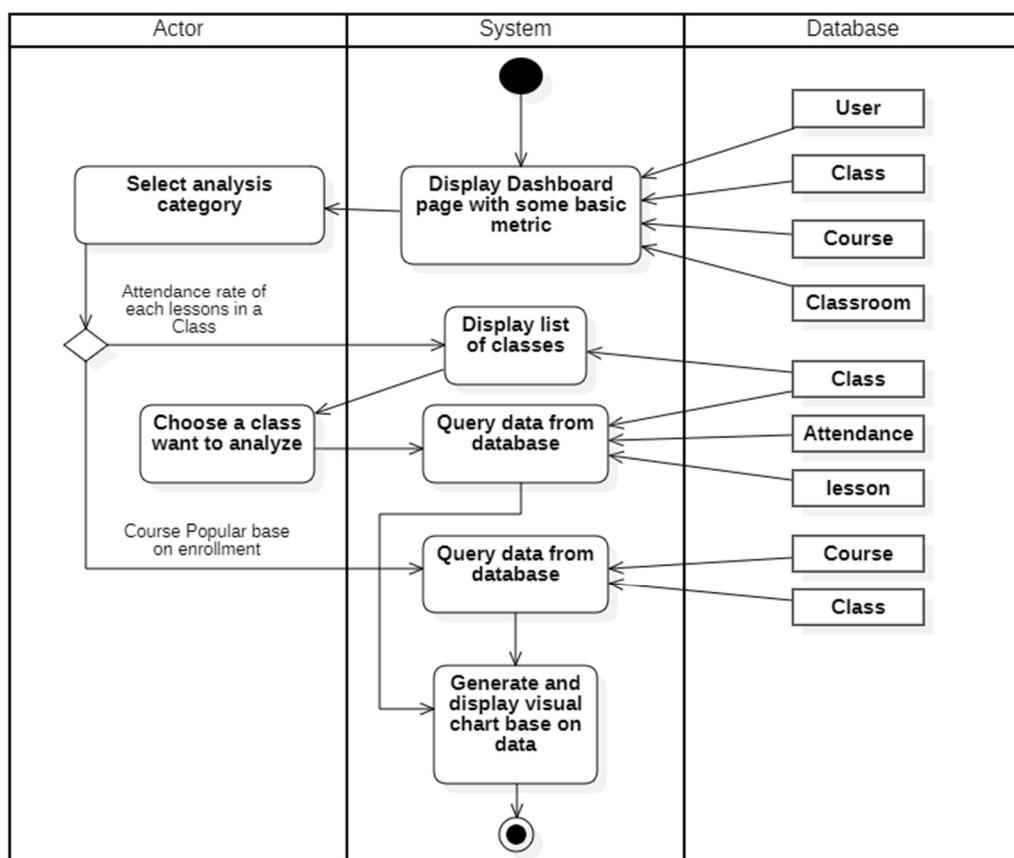


Figure 3. 15.Activity Diagram for Use case “Analyze”

8. Logout (For all users)

| | |
|-------------------------|--|
| Use-case name | Logout |
| Description | User log out from system |
| Trigger | User clicks to “Sign out” button at homepage |
| Pre-condition | User’s status is signed in |
| Post-condition | <ul style="list-style-type: none"> – User logged out from system – System display Login page |
| Basic flow | <ol style="list-style-type: none"> 1. System display Confirm dialog 2. Manager confirm action log out 3. Display Login page |
| Alternative flow | 2a. System cancel action due to Manager no confirm action |
| Exception flow | |

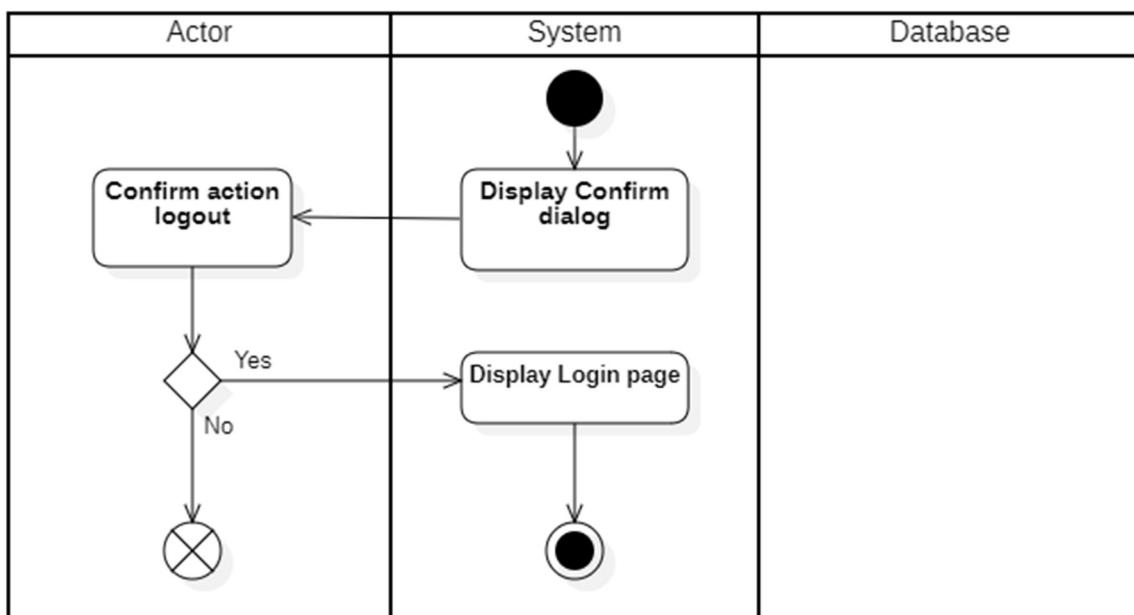


Figure 3. 16. Activity Diagram for Use case “Logout”

9. Explore Courses (For User)

| | |
|-------------------------|--|
| Use-case name | Explore Courses |
| Description | Explore exists Courses in the center |
| Trigger | User clicks to “Courses” menu at homepage |
| Pre-condition | User’s status is signed in |
| Post-condition | Courses menu display Courses queried from database |
| Basic flow | <ol style="list-style-type: none">1. System display List of Courses queried from database2. Manager select search function3. Manager enter keywords to search4. System process search request5. System query data from database base on given keywords6. System filter data at Courses page |
| Alternative flow | |
| Exception flow | |

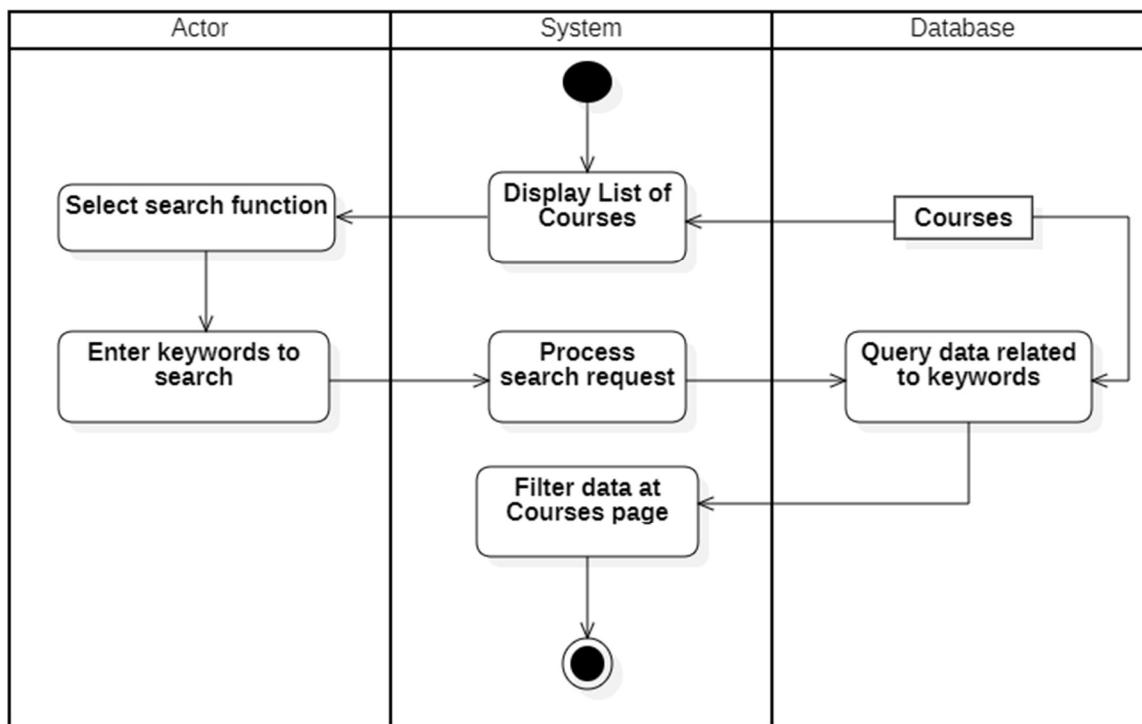


Figure 3. 17. Activity Diagram for Use case “Explore Courses”

9.1. Explore Chapters (For User)

| | |
|-------------------------|---|
| Use-case name | Explore Chapters |
| Description | Explore exists Chapters in a specific Course |
| Trigger | User clicks to “Chapters” after select a Course at courses page |
| Pre-condition | User’s status is signed in User already access Courses page |
| Post-condition | Chapters menu display Chapter in a Course queried from database |
| Basic flow | <ol style="list-style-type: none"> 1. System display List of Chapters in a Course queried from database 2. Manager select search function 3. Manager enter keywords to search 4. System process search request 5. System query data from database base on given keywords 6. System filter data at Chapters page |
| Alternative flow | |
| Exception flow | |

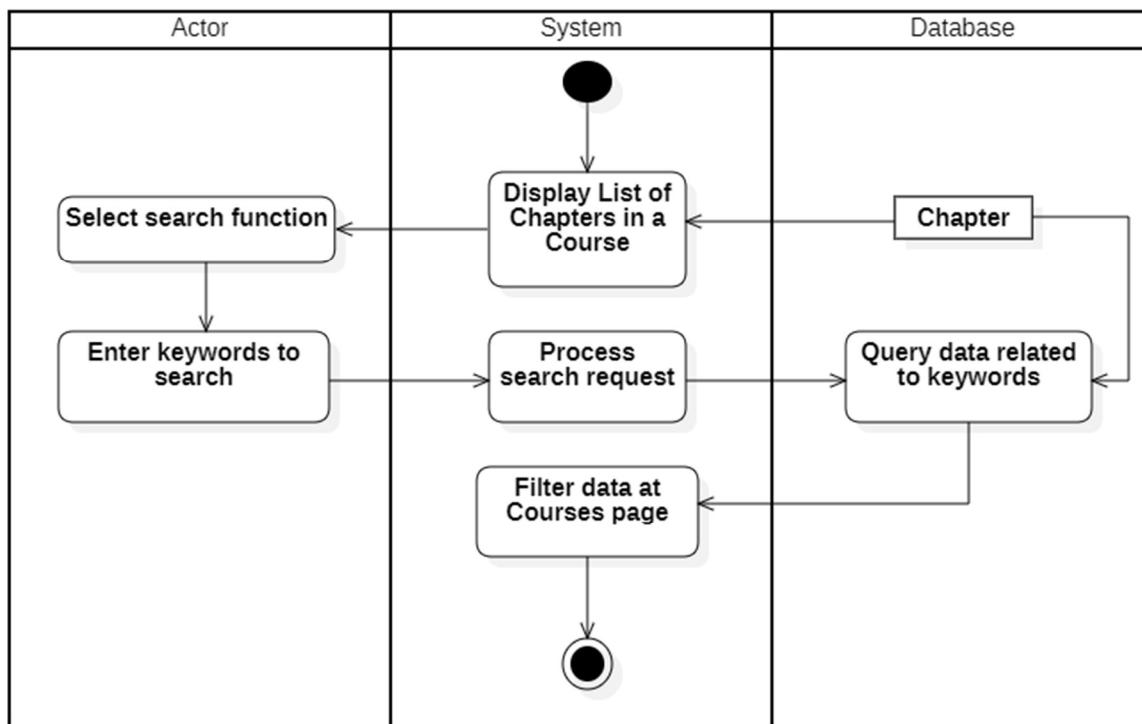


Figure 3. 18. Activity Diagram for Use case “Explore Chapters”

10. View Schedule (For User)

| | |
|-------------------------|---|
| Use-case name | View Schedule |
| Description | View Studying/Teaching Schedule in English Center |
| Trigger | User clicks to “Schedule” menu at homepage |
| Pre-condition | User’s status is signed in |
| Post-condition | Menu generates a clearly schedule for User |
| Basic flow | <ol style="list-style-type: none"> 1. System Query data from database 2. Generate schedule for User 3. System Display schedule as timetable with class information |
| Alternative flow | |
| Exception flow | |

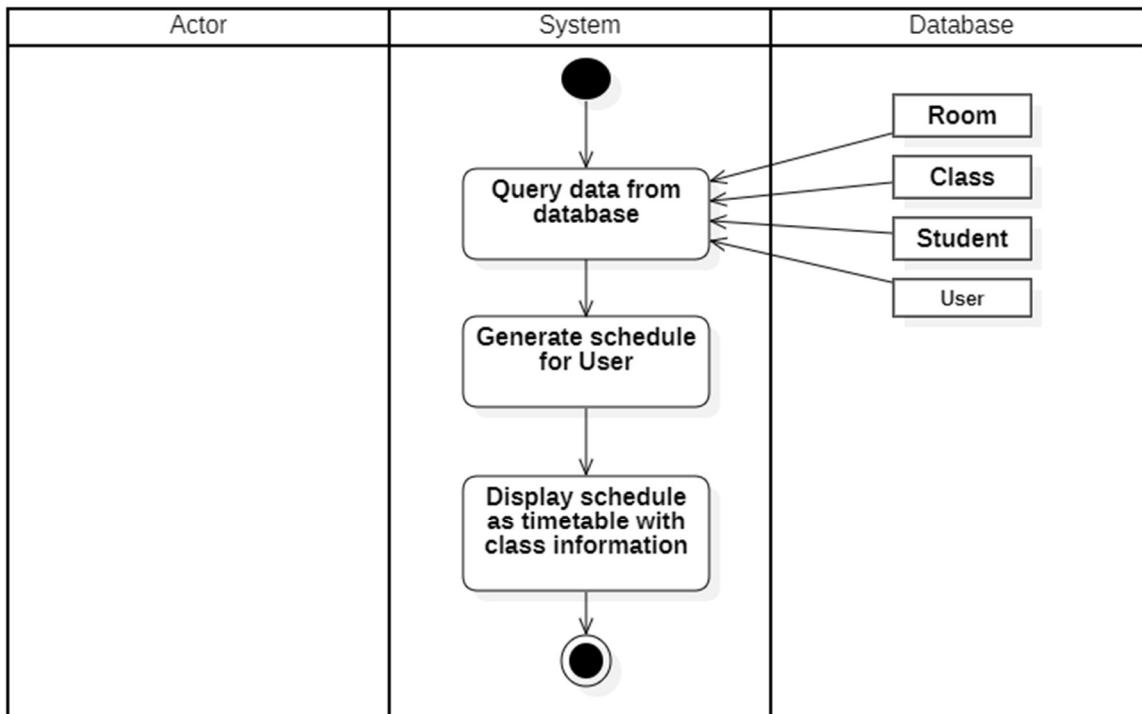


Figure 3. 19. Activity Diagram for Use case “View Schedule”

11. Manage Classes (For Teacher)

| | |
|-------------------------|--|
| Use-case name | Manage Classes |
| Description | Manage Classes that Teacher is currently participating in |
| Trigger | Teacher clicks to “Classes” menu at homepage |
| Pre-condition | Teacher’s status is signed in |
| Post-condition | Classes menu display list of Classes that teacher has participated |
| Basic flow | <ol style="list-style-type: none">1. System display List of Classes queried from database2. Teacher select search function3. Teacher enter keywords to search4. System process search request5. System query data from database base on given keywords6. System filter data at Classes page |
| Alternative flow | |
| Exception flow | |

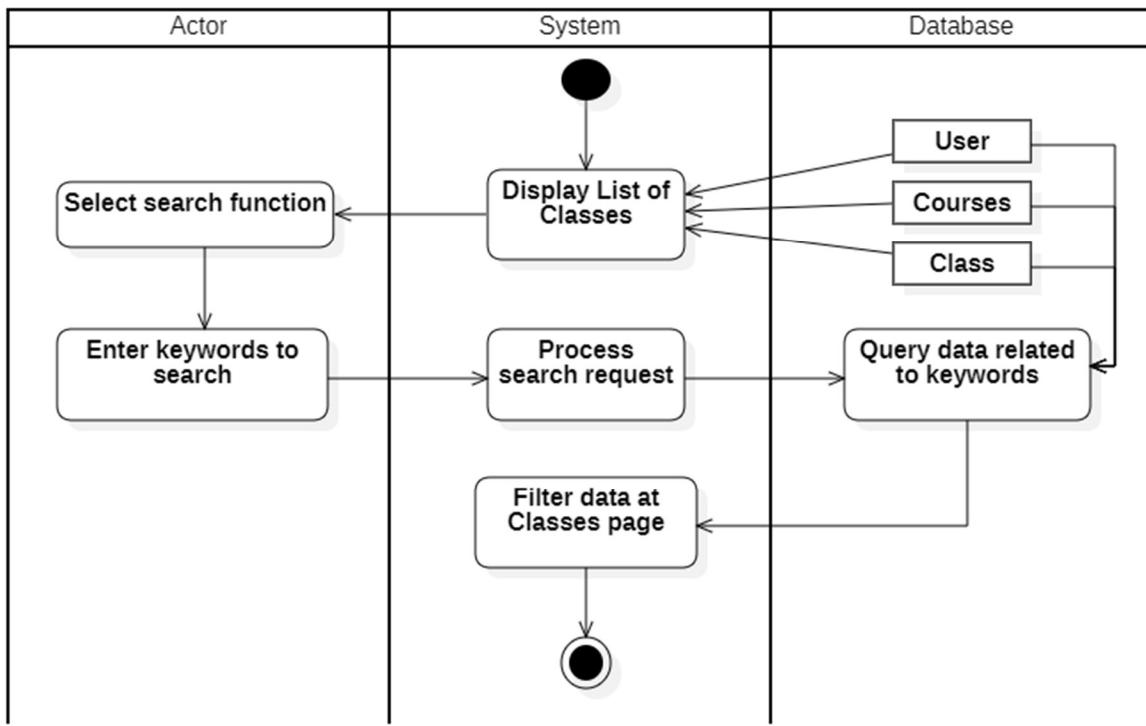


Figure 3. 20. Activity Diagram for Use case “Manage Classes”

11.1. Manage Lessons (For Teacher)

| | |
|-------------------------|--|
| Use-case name | Manage Lessons |
| Description | Manage exists Lessons in a specific Class |
| Trigger | Teacher clicks to “View Lessons” after select a Class at courses page |
| Pre-condition | Teacher’s status is signed in Teacher already access Classes page |
| Post-condition | Lessons menu display Lessons in a Class queried from database |
| Basic flow | <ol style="list-style-type: none"> 1. System display List of Lessons in a Class queried from database 2. Teacher select search function 3. Teacher enter keywords to search 4. System process search request 5. System query data from database base on given keywords 6. System filter data at Lessons page |
| Alternative flow | |
| Exception flow | |

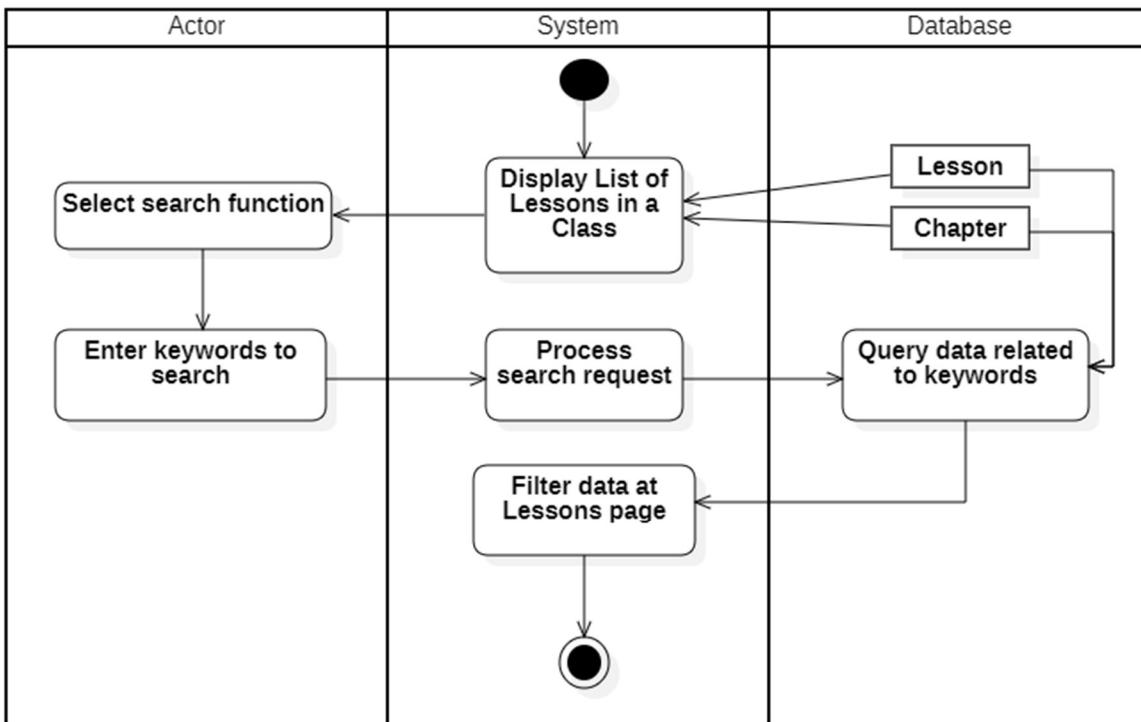


Figure 3. 21Activity Diagram for Use case “Manage Lessons”

12. View Classrooms

| | |
|-----------------------|---|
| Use-case name | View Classrooms |
| Description | Teacher can search and view schedule in a specific Classrooms |
| Trigger | Teacher clicks to “Rooms” menu at homepage |
| Pre-condition | Teacher’s status is signed in |
| Post-condition | Data displayed on the Classrooms management page must be updated with the latest data |
| Basic flow | <ol style="list-style-type: none"> 1. System displays Classrooms management page 2. Teacher chooses one function: search/view schedule classrooms <ul style="list-style-type: none"> • Search 3. Teacher select search function and enter keywords. |

| | |
|-------------------------|--|
| | <p>4. System processes searching data</p> <p>5. System queries data base on given keywords</p> <p>6. System filtered data at classrooms page (data related to keyword is retained)</p> <ul style="list-style-type: none"> • View Schedule <p>7. Teacher choose a classroom want to view schedule</p> <p>8. System get data from database and generate a schedule for selected room</p> <p>9. System display schedule as timetable with class information</p> |
| Alternative flow | |
| Exception flow | |

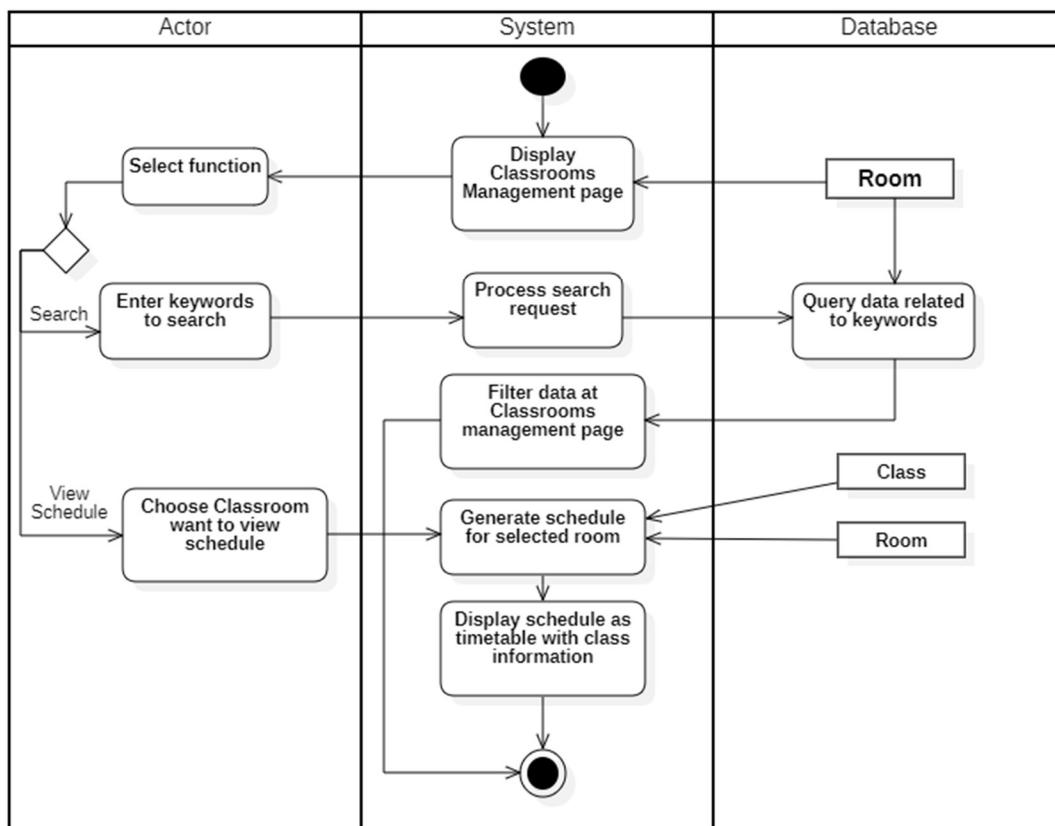


Figure 3. 22. Activity Diagram for Use case “View Classrooms”

13. View Classes

| | |
|-------------------------|---|
| Use-case name | View Classes |
| Description | View Classes that Student is currently enrolled |
| Trigger | Student clicks to “Classes” menu at homepage |
| Pre-condition | Student’s status is signed in |
| Post-condition | Classes menu display list of Classes that Student has enrolled |
| Basic flow | <ol style="list-style-type: none"> 1. System display List of Classes queried from database 2. Student select search function 3. Student enter keywords to search 4. System process search request 5. System query data from database base on given keywords 6. System filter data at Classes page |
| Alternative flow | |
| Exception flow | |

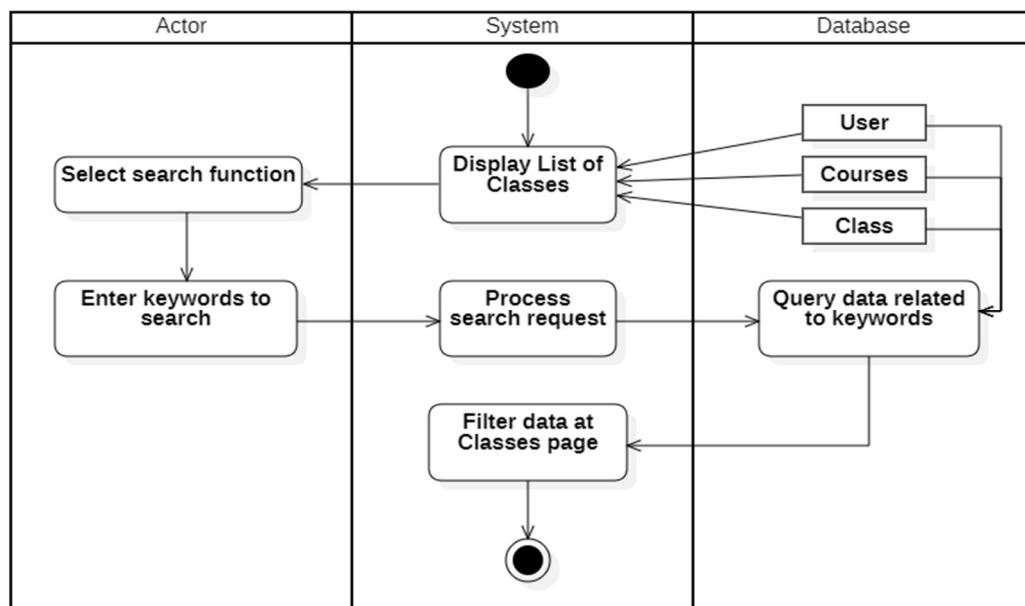


Figure 3. 23. Activity Diagram for Use case “View Classes”

13.1. View Lessons

| | |
|-------------------------|--|
| Use-case name | View Lessons |
| Description | View exists Lessons in a specific Class |
| Trigger | Student clicks to “View Lessons” after select a Class at courses page |
| Pre-condition | Student’s status is signed in Student already access Classes page |
| Post-condition | Lessons menu display Lessons in a Class queried from database |
| Basic flow | <ol style="list-style-type: none"> 1. System display List of Lessons in a Class queried from database 2. Student select search function 3. Student enter keywords to search 4. System process search request 5. System query data from database base on given keywords 6. System filter data at Lessons page |
| Alternative flow | |
| Exception flow | |

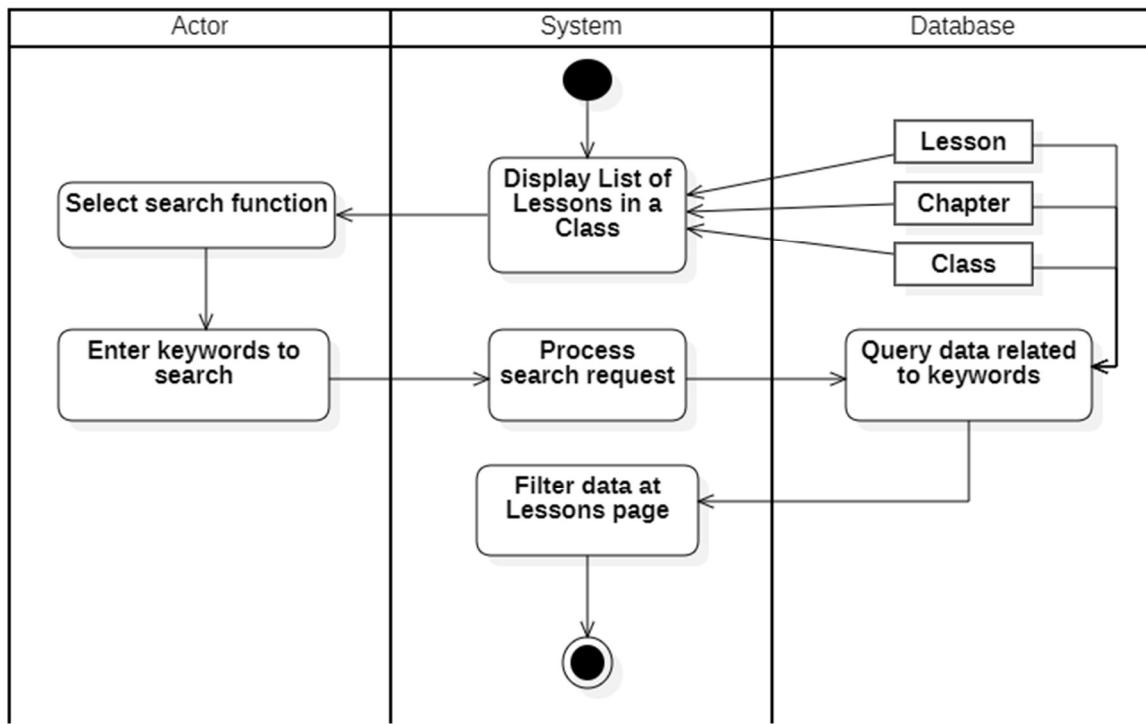


Figure 3. 24. Activity Diagram for Use case “View Lessons”

13.1.1. View Materials

| | |
|-----------------------|--|
| Use-case name | View Materials |
| Description | Student can search and download Material information In a specific Lesson |
| Trigger | Student select “Materials” tab after select a Lesson in Lessons page |
| Pre-condition | <ul style="list-style-type: none"> – Student’s status is signed in – Student has previously accessed Lessons management page and selected a specific lesson. |
| Post-condition | Data displayed on the Materials page must be updated with the latest data |
| Basic flow | <ol style="list-style-type: none"> 1. Student Select a lesson and choose Materials tab. 2. System displays Materials management page 3. Student chooses one function: download/search Material <ul style="list-style-type: none"> • Search 4. Student select search function and enter keywords. |

| | |
|-------------------------|--|
| | <p>5. System processes searching data</p> <p>6. System queries data base on given keywords</p> <p>7. System filtered data at Materials management page (data related to keyword is retained)</p> <ul style="list-style-type: none"> • Download <p>8. Student Choose material need to download</p> <p>9. System Get material path from database</p> <p>10. System Display Save file dialog</p> <p>11. Student Choose destination directory</p> <p>12. System Process download request</p> |
| Alternative flow | |
| Exception flow | |

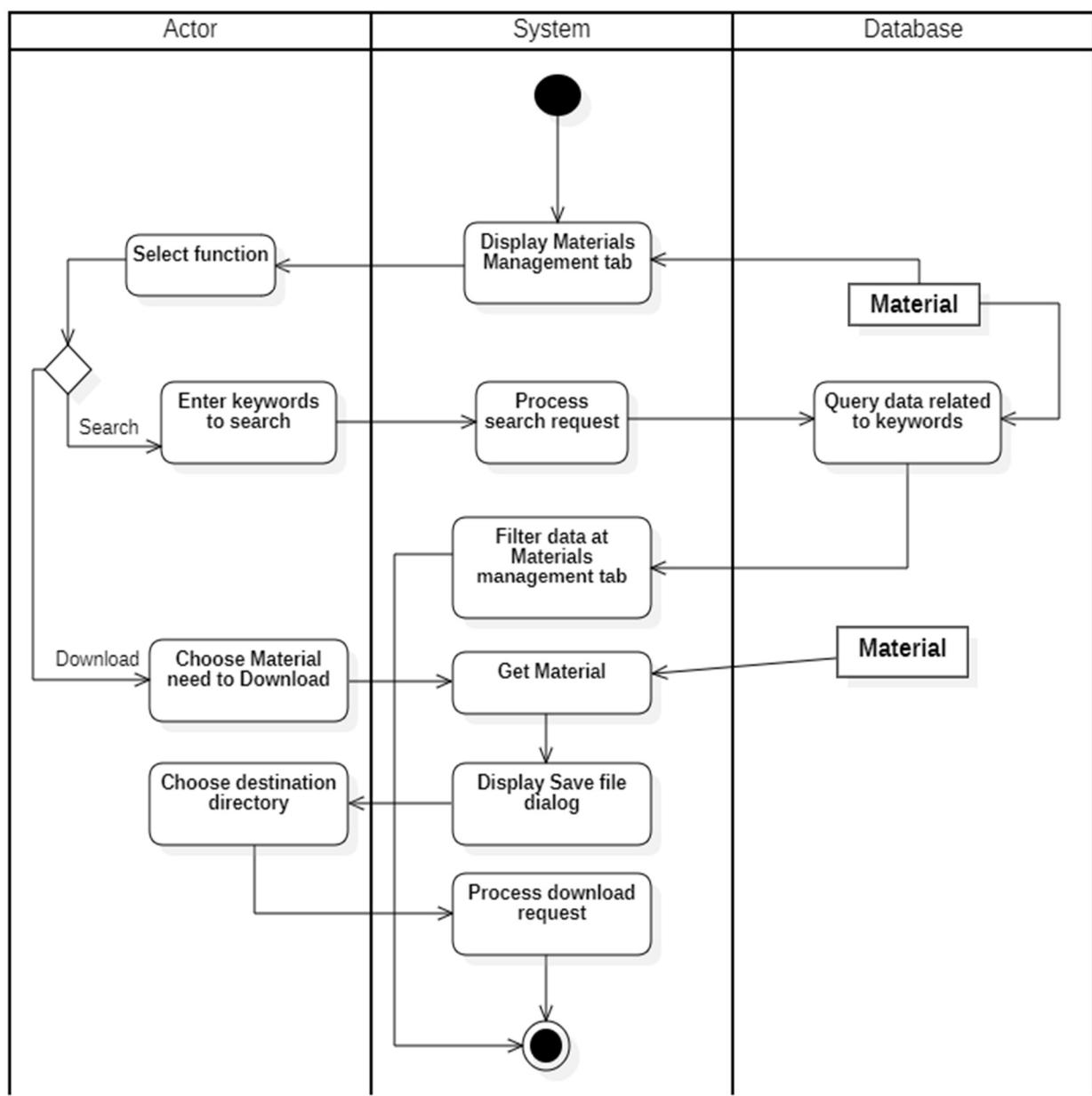


Figure 3. 25. Activity Diagram for Use case “View Materials”