**WIL Report Management System**

Software Requirement Specification

­­­

By

**Ms. Phinthip Samutloiwon 552115050**

**Mr. Veerapat In-ongkarn 562115055**

Department of Software Engineering

College of Arts, Media and Technology

Chiang Mai University

Project Advisor

**Dr. Prompong Sugunnasil**

**Document History**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Document Name** | **Version** | **Status** | | **Date** | **Viewable** | **Reviewer** | **Responsible** |
| **Documents** | | | | | | | |
| WIL project-based report System Proposal\_V.0.1.docx | Created document  - Add Chapter I | Introduction   - Purpose  - Scope  - User Characteristic  - Operation Environment  -Design and Implementation  Constraints  - Dependency Document  - Acronyms  - Document format  - Definition  - Add Chapter II | Overall Description  - Project Perspective  - Project Functions  - Functional Requirement - Add Chapter III | Specific Requirement  - Use Case - Add Chapter IV | Software Requirement Specification | | Draft | 10-02-2017 | PS,VI | PS,VI | PS,VI |
| WIL project-based report System Proposal\_V.0.2.docx | - Edit Chapter I | Introduction   - Purpose  - Scope  - User Characteristic  - Operation Environment  -Design and Implementation  Constraints  - Dependency Document  - Acronyms  - Document format  - Definition  - Edit Chapter II | Overall Description  - Project Perspective  - Project Functions  - Functional Requirement - Edit Chapter III | Specific Requirement  - Use Case - Edit Chapter IV | Software Requirement Specification | | Draft | 17-02-2017 | PS,VI | PS,VI | PS,VI |
| WIL project-based report System Proposal\_V.0.3.docx | - Edit Chapter III | Specific Requirement  - Use Case - Edit Chapter IV | Software Requirement Specification  - Add UC | | Draft | 25-02-2017 | PS,VI | PS,VI | PS,VI |
| WIL project-based report System Proposal\_V.0.4.docx | - Edit Chapter III | Specific Requirement  - Use Case - Edit Chapter IV | Software Requirement Specification  - Edit UC | | Draft | 28-02-2017 | PS,VI | PS,VI | PS,VI |
| WIL project-based report System Proposal\_V.0.5.docx | - Edit Chapter III | Specific Requirement  - Use Case - Edit Chapter IV | Software Requirement Specification  - Edit UC | | Draft | 07-03-2017 | PS,VI | PS,VI | PS,VI |

**Table of Contents**

[Chapter I | Introduction 5](#_Toc476678721)

[1.1. Purpose 5](#_Toc476678722)

[1.2. Project Scope 5](#_Toc476678723)

[1.3. Acronyms and definitions Acronyms 5](#_Toc476678724)

[1.4. Definitions 5](#_Toc476678725)

[Chapter II | Overall Description 7](#_Toc476678726)

[2.1. Product perspective 7](#_Toc476678727)

[2.2. Product Functions 7](#_Toc476678728)

[2.3. User Characteristic 7](#_Toc476678729)

[2.4. Operation Environment 7](#_Toc476678730)

[2.5. Design and Implementation constants 7](#_Toc476678731)

[Chapter III | Software Requirement Specification 8](#_Toc476678732)

[3.1 User requirements 8](#_Toc476678733)

[3.2 System requirements 9](#_Toc476678734)

[3.3 User requirement specification with system requirement specification 13](#_Toc476678735)

[**Chapter IV | Requirement Specification** 22](#_Toc476678736)

[4.1 Use case diagram 22](#_Toc476678737)

[**4.2** **Use case description** 22](#_Toc476678738)

[UC-01 22](#_Toc476678739)

[UC-02 24](#_Toc476678740)

[UC-03 26](#_Toc476678741)

[UC-04 27](#_Toc476678742)

[UC-05 28](#_Toc476678743)

[UC-06 29](#_Toc476678744)

[UC-07 30](#_Toc476678745)

[UC-08 31](#_Toc476678746)

[UC-09 32](#_Toc476678747)

[UC-10 33](#_Toc476678748)

[UC-11 34](#_Toc476678749)

[UC-12 35](#_Toc476678750)

[UC-13 36](#_Toc476678751)

[UC-14 37](#_Toc476678752)

[UC-15 38](#_Toc476678753)

[UC-16 39](#_Toc476678754)

[UC-17 40](#_Toc476678755)

[UC-18 41](#_Toc476678756)

[UC-19 42](#_Toc476678757)

[UC-20 42](#_Toc476678758)

[UC-21 43](#_Toc476678759)

[UC-22 44](#_Toc476678760)

# **Chapter I | Introduction**

## **Purpose**

The purpose of this software requirement specification document (SRS) is to detail overview for WIL report management web application. The document describes how an application will interact with other system and human. It is also contain a detail description of functional and non-functional requirement. In other words, it describe what system can do, limitations, type of users, parameters, and other requirements which support the development. The software requirement specification provides developers and users to understand each other in structure details.

## **Project Scope**

WIL report management system (WRMS) is a web application for managing tasks of students, WIL weekly report, and comments of mentors and supervisors. All users shall receive a real-time notification via the web application and Email. This application was designed to use in College of Art, Media, and Technology. WRMS supports desktop and mobile device on chrome, Safari, and opera.

## **Acronyms and definitions Acronyms**

URS User Requirement Specification  
SRS Software Requirement Specification  
WRMS WIL Report Management System  
UC Use Case  
CD Class diagram  
AD Activity diagram  
SD Sequence diagram  
UI User interface  
UML Unified Modeling Language

## **Definitions**

|  |  |
| --- | --- |
| Use case | In UML, a complete task of a system that provides a measurable result of value for an actor. More formally, a use case defines a set of use case instances or scenarios [1] |
| Feature | Transformation of input parameters to output parameters based on a specified algorithm. It describes the functionality of a product in the language of the product. Used for requirements analysis, design, coding, testing or maintenance. [2] |
| Requirement | (1) A condition or capability needed by a user to solve a problem or achieve an objective. (2) A condition or capability that must be met or possessed by a system or system component to satisfy a contract, standard, specification, or other formally imposed document. (3) A documented representation of a condition or capability as in definition (1) or (2). [2] |
| Specification | Precise description of an activity or work product which serves as basis or input for further activities or work product. A specification can comprise requirements to a product and how they will be solved. Different parts of a specification (e.g., what is to be done, how it will be done) must not be mixed. [2] |
| UML | Unified Modeling Languages. Standardized notation for modeling design descriptions, architectures or scenarios. Not depending on a specific method. Issued and maintained by the object Management Group (OMG). [2] |
| System | A conceptual entity defined by its boundaries. Examples include companies, divisions, sets of software applications, components, machines, and devices. [1] |

# **Chapter II | Overall Description**

## **Product perspective**

WIL report management system (WRMS) consist of 5 parts which are User Management system, Tasks management system, Comments management system, Report export system, Notification system. When students generate report, the system shall change all list tasks into sentences and ordered by weekly or monthly. Mentor and Supervisor are able to see the students’ progress and also leave some comments. After leaving a comment, the system will send a real-time notification to all involved users.

## **Product Functions**

WRMS is a web application and mobile application which helps students to create WIL reports from lists of tasks to sentences. It also helps mentors and supervisor to see the progress of students. Furthermore, Mentors and supervisor are able to leave their comments on tasks of students.

## **User Characteristic**

WIL report management web application includes 4 types of users:

1. Administrator – the user who can initial all functions in the application which are login, logout, register to the system, edit profile of users, edit password of users.
2. Students – the user who is able to login, edit their own profile, logout, view and manage tasks, view task statistics, export reports, and receive notifications.
3. Mentor – the user who is able to login, edit their own profile, logout manage comments, view reports, and receive notifications.
4. Supervisor – the user who is able to login, edit their own profile, logout manage comments, view reports, and receive notifications.
5. Users – Student, Mentor, and supervisor

## **Operation Environment**

WIL report management system is a web application which requires internet connection. The users have to access to the website though internet browser including Internet Explorer 10 (or above), Safari version 9, and opera. The system support windows 8.1 and 10, Mac OS sierra, android 4.0, and iOS 10.

## **Design and Implementation constants**

2.5.1. The application requires internet connection.  
2.5.2. The prototype is available in English.  
2.5.3. Administrator shall create an account for a user.  
2.5.4. Mentor and Supervisor are not able to manage tasks of students.   
2.5.5. Overload data could affect to the performance of web application.

# **Chapter III | Software Requirement Specification**

## **User requirements**

**Feature #1: Users management**

1. Visitor can register to the system.
2. Users can edit their own profile.
3. Users can login into the system.
4. Users can logout from the system.
5. Student can add registration code of supervisor and mentor.

**Feature #2: Task management**

1. Students can view tasks overview of a project.
2. Students can view the statistics of tasks.
3. Students can add a project.
4. Students can delete a project.
5. Students can add tasks of a project.
6. Students can edit tasks of a project.
7. Students can delete tasks of a project.
8. Students can move tasks of a project.

**Feature #3: Progress tracking**

1. Student can view comments from Mentor and Supervisor.
2. Students can reply a comment of Mentor and Supervisor.
3. Student can delete their own comment.
4. Mentors and Supervisor can view tasks overview of a project.
5. Mentors and Supervisor can view the statistics of tasks.
6. Mentor can view comments in student’s report.
7. Mentor can add comments in student’s report.
8. Mentor can edit comments in student’s report.
9. Mentor can delete comments in student’s report.
10. Supervisor can view comments in student’s report.
11. Supervisor can add comments in student’s report.
12. Supervisor can edit comments in student’s report.
13. Supervisor can delete comments in student’s report.

**Feature #4: Report export**

1. Users can view weekly report.
2. Users can generate a printable weekly report file.

**Feature #5: Web and Email notification**

1. Student can receive comment notification message on Web application.
2. Mentor and supervisor can receive activities notification message on Web application.
3. Student can receive comment notification message by Email.
4. Mentor and supervisor can receive activities notification message by Email.

## **System requirements**

1. System shall provide UI for the visitor to select the role of user.
2. System shall provide UI for the visitor to receive name, surname, email, password, student\_id, company name.
3. S System shall provide UI to display an error message if an input email is not matched with constraint.
4. System shall provide UI
5. System shall provide UI to display that the visitor registered successful.
6. System shall provide registration code for supervisor and mentor.
7. System shall provide UI to display an error message if an input name is not matched with constraint.
8. System shall provide UI to display an error message if an input surname is not matched with constraint.
9. System shall provide UI to display an error message if an input email is not matched with constraint.
10. System shall provide UI to display an error message if an input password is not matched with constraint
11. System shall provide UI to display an error message if a company name is not matched with constraint.
12. System shall validate name, surname, email, password, student\_id, company name.
13. System shall save username, password, name, surname, email, student\_id, as a new user on the system with the default status for the visitor.
14. System shall provide UI for the student to edit name, surname, email, password, student\_id, company name, and signature.
15. System shall provide UI for the mentor to edit name, surname, email, password, company name.
16. System shall provide UI for the supervisor to edit name, surname, email, password, and signature.
17. System shall provide UI to display an error message if the student edit their own information which the “name” is not matched with constraint.
18. System shall provide UI to display an error message if the student edit their own information which the “surname” is not matched with constraint.
19. System shall provide UI to display an error message if the student edit their own information which the “email” is not matched with constraint.
20. System shall provide UI to display an error message if the student edit their own information which the “password” is not matched with constraint.
21. System shall provide UI to display an error message if the student edit their own information which the “student\_id” is not matched with constraint.
22. System shall provide UI to display an error message if the student edit their own information which the “company name” is not matched with constraint.
23. System shall provide UI to display an error message if the mentor edit their own information which the “name” is not matched with constraint.
24. System shall provide UI to display an error message if the mentor edit their own information which the “surname” is not matched with constraint.
25. System shall provide UI to display an error message if the mentor edit their own information which the “email” is not matched with constraint.
26. System shall provide UI to display an error message if the mentor edit their own information which the “password” is not matched with constraint.
27. System shall provide UI to display an error message if the mentor edit their own information which the “company name” is not matched with constraint.
28. System shall provide UI to display an error message if the supervisor edit their own information which the “name” is not matched with constraint.
29. System shall provide UI to display an error message if the supervisor edit their own information which the “surname” is not matched with constraint.
30. System shall provide UI to display an error message if the supervisor edit their own information which the “email” is not matched with constraint.
31. System shall provide UI to display an error message if the supervisor edit their own information which the “password” is not matched with constraint.
32. System shall provide “add signature” button to add their signature.
33. System shall save data to database.
34. System shall provide UI for the student to login.
35. System shall provide UI for the mentor to login.
36. System shall provide UI for the supervisor to login.
37. System shall receive email and password.
38. System shall verify email and password.
39. System shall provide UI to display the student login successful.
40. System shall provide UI to display the mentor login successful.
41. System shall provide UI to display the supervisor login successful.
42. System shall provide UI to display an error message that email or password are wrong.
43. System shall navigates the student to access the student dashboard.
44. System shall navigates the mentor to access the mentor dashboard.
45. System shall navigate the supervisor to access the supervisor dashboard.
46. System shall provide UI for the administrator to logout.
47. System shall provide UI for the student to logout.
48. System shall provide UI for the mentor to logout.
49. System shall provide UI for the supervisor to logout.
50. System shall provide UI to display that the administrator logout successful.
51. System shall provide UI to display that the student logout successful.
52. System shall provide UI to display that the mentor logout successful.
53. System shall provide UI to display that the supervisor logout successful.
54. System shall provide UI to receive a code.
55. System shall provide an add button for adding a code.
56. System shall connect to database.
57. System shall provide UI to display student dashboard.
58. System shall provide UI to display projects detail page.
59. System shall provide UI to display lists and cards of tasks.
60. System shall provide UI to display the statistics of tasks.
61. System shall provide UI which receive project name, project description, and due date.
62. System shall provide delete button interface for deleting a project.
63. System shall delete a project in database.
64. System shall provide UI which receive tasks name, task description, and task due date.
65. System shall provide a button to delete a task.
66. System shall provide a delete conformation message.
67. System shall provide a move button to move a task.
68. System shall provide a conformation message “Do you want to move this task?”.
69. System shall display a successful message “Move a task success”
70. System shall provide an error message if the move destination is not match with constraints “You cannot move this task to xxxxx”.
71. System shall provide UI to display a comment of mentor and supervisor in each task.
72. System shall provide a reply button for replying comments of mentor and supervisor.
73. System shall provide a submit button for saving data.
74. System shall display a successful message “comment is added.”
75. System shall provide a delete comment button.
76. System shall display a confirmation message “Do you want to delete this comment?”
77. System shall provide a cancel button.
78. System shall cancel to delete a comment.
79. System shall display a successful message “Comment is deleted.”.
80. System shall provide UI to display tasks of students who add registration code of mentor and supervisor.
81. System shall provide UI to display list of students.
82. System shall provide UI to display specific student’s dash board.
83. System shall provide UI to display statistics of student’s tasks.
84. System shall provide “show report” button to display a student’s weekly report.
85. System shall provide UI to display a student’s weekly report.
86. System shall provide UI which receive comment of mentor in weekly form.
87. System shall display a confirmation message “Do you want to save this comment?”.
88. System shall display a successful message “Add comment success”.
89. System shall provide UI to edit a comment of mentor in daily form.
90. System shall provide UI to edit a comment of mentor in weekly form.
91. System shall display a confirmation message “Do you want to edit this comment?”.
92. System shall display a successful message “Edit comment success”.
93. System shall provide UI to display a list of mentor’s comments.
94. System shall provide delete button for deleting a comment.
95. System shall display a cancel button.
96. System shall display a confirmation message “Do you want to delete this comment?”.
97. System shall display a successful message “comment is deleted”
98. System shall provide UI to display comment options.
99. System shall display comments of supervisor in daily or weekly form.
100. System shall provide UI of adding comment options.
101. System shall provide UI to add a comment of supervisor in daily form.
102. System shall provide UI to add a comment of supervisor in weekly form.
103. System shall provide a save and cancel button.
104. System shall cancel a comment.
105. System shall display a confirmation message “Do you want to add this comment?”.
106. System shall display a successful message “add comment success”.
107. System shall provide UI of editing comment options.
108. System shall provide UI to edit a comment of supervisor in daily form.
109. System shall provide UI to edit a comment of supervisor in weekly form.
110. System shall display a confirmation message “Do you want to edit this comment?”.
111. System shall display a successful message “Edit comment success”.
112. System shall provide UI to display a list of supervisor’s comments.
113. System shall arrange all tasks based on time-stamp.
114. System shall separate tasks in weekly.
115. System shall merge keywords in a task with a sentence template.
116. System shall provide UI to display student’s weekly report content, comments, Mentor’s signature, student’s signature.
117. System shall convert html weekly report to pdf report.
118. System shall display printable weekly report.
119. System shall provide a download button to download a pdf report.
120. System shall detect comments of mentor and supervisor.
121. System shall arrange comments based on time-stamp.
122. System shall provide UI on the right to display comment activities of mentor and supervisor from the earliest to the latest comment.
123. System shall detect task activities of students who under their guidance.
124. System shall arrange task activities based on time-stamp.
125. System shall provide UI on the right to display task activities of mentor and supervisor from the earliest to the latest activity.
126. System shall detect comments of mentor and supervisor.
127. System shall provide email template of comment notification.
128. System shall merge the content of comment to an email template.
129. System shall send a notification via email based on time-stamp of comment.
130. System shall provide email template of task activities notification.
131. System shall merge the content of task to an email template.
132. System shall send a notification via email based on time-stamp of task.

## **User requirement specification with system requirement specification**

1. **Visitor can register to the system.**
2. System shall provide UI for the visitor to select the role of user.
3. System shall provide UI for the visitor to receive name, surname, email, password, student\_id, company name.
4. System shall provide UI to display an error message if an input email is not matched with constraint.
5. System shall provide UI to display that the visitor registered successful.
6. System shall provide registration code for supervisor and mentor.
7. System shall provide UI to display an error message if an input name is not matched with constraint.
8. System shall provide UI to display an error message if an input surname is not matched with constraint.
9. System shall provide UI to display an error message if an input email is not matched with constraint.
10. System shall provide UI to display an error message if an input password is not matched with constraint
11. System shall provide UI to display an error message if a company name is not matched with constraint.
12. System shall display a confirmation message “Do you want to save your profile information?”
13. System shall validate name, surname, email, password, student\_id, company name.
14. System shall save username, password, name, surname, email, student\_id, as a new user on the system with the default status for the visitor.
15. System shall save data to database.
16. System shall go back to previous UI.

**URS-02: Users can edit their own profile.**

1. System shall provide UI for the student to edit name, surname, email, password, student\_id, company name, Mentor and Supervisor code, and signature.
2. System shall provide UI for the mentor to edit name, surname, email, password, company name.
3. System shall provide UI for the supervisor to edit name, surname, email, password, and signature.
4. System shall provide UI to display an error message if the student edit their own information which the “name” is not matched with constraint.
5. System shall provide UI to display an error message if the student edit their own information which the “surname” is not matched with constraint.
6. System shall provide UI to display an error message if the student edit their own information which the “email” is not matched with constraint.
7. System shall provide UI to display an error message if the student edit their own information which the “password” is not matched with constraint.
8. System shall provide UI to display an error message if the student edit their own information which the “student\_id” is not matched with constraint.
9. System shall provide UI to display an error message if the student edit their own information which the “Mentor and supervisor code” does not exist in database.
10. System shall provide UI to display an error message if the student edit their own information which the “company name” is not matched with constraint.
11. System shall provide UI to display an error message if the mentor edit their own information which the “name” is not matched with constraint.
12. System shall provide UI to display an error message if the mentor edit their own information which the “surname” is not matched with constraint.
13. System shall provide UI to display an error message if the mentor edit their own information which the “email” is not matched with constraint.
14. System shall provide UI to display an error message if the mentor edit their own information which the “password” is not matched with constraint.
15. System shall provide UI to display an error message if the mentor edit their own information which the “company name” is not matched with constraint.
16. System shall provide UI to display an error message if the supervisor edit their own information which the “name” is not matched with constraint.
17. System shall provide UI to display an error message if the supervisor edit their own information which the “surname” is not matched with constraint.
18. System shall provide UI to display an error message if the supervisor edit their own information which the “email” is not matched with constraint.
19. System shall provide UI to display an error message if the supervisor edit their own information which the “password” is not matched with constraint.
20. System shall provide “add signature” button to add their signature.
21. System shall save data to database.

**URS-03: Users can login into the system.**

1. System shall provide UI login.
2. System shall receive email and password.
3. System shall verify email and password.
4. System shall provide UI to display the login successful.
5. System shall display an error message that email or password are wrong.
6. System shall navigates the student to access the student dashboard.
7. System shall navigates the mentor to access the mentor dashboard.
8. System shall navigate the supervisor to access the supervisor dashboard.

**URS-04: Users can logout from the system.**

1. System shall provide UI to logout.
2. System shall display a conformation message.
3. System shall logout from the system.
4. System shall provide UI to display that the logout successful.
5. System shall redirect to login page.
6. System go back to previous UI.

**URS-05: Student can add mentor and supervisor code.**

1. System shall provide UI to receive a code.
2. System shall provide an add button for adding another code.
3. System display successful message.
4. System go back to previous UI.

**URS-06: Students can view tasks overview of a project.**

1. System shall connect to database.
2. System shall provide UI to display student dashboard.
3. System shall provide UI to display projects detail page.
4. System shall provide UI to display lists and cards of tasks.

**URS-07: Students can view the statistics of tasks.**

1. System shall connect to database.
2. System shall provide UI to display student dashboard.
3. System shall provide UI to display the statistics of tasks.

**URS-08: Students can add a project.**

1. System shall connect to database.
2. System shall provide UI to display student dashboard.
3. System shall provide “create new project” button.
4. System shall provide UI which receive project name, project description, and due date.
5. System shall display a successful message “A new project is created.”
6. System shall display an error message “Invalid input data. Please recheck it again ”
7. System shall redirect to create a new project detail page with the previous input data.
8. System shall redirect to previous page.
9. System shall save data to database.

**URS-09: Students can delete a project**.

1. System shall connect to database.
2. System shall provide UI to display student’s dashboard.
3. System shall provide UI to display the project detail page.
4. System shall provide delete button interface for deleting a project.
5. System shall display a confirmation message “Are you sure to delete this project?” with “yes” and “no” button.
6. System shall delete a project in database.

**URS-10: Students can add tasks of a project.**

1. System shall connect to database.
2. System shall provide UI to display student dashboard.
3. System shall provide UI to display the project detail page.
4. System shall provide UI which receive tasks name, task description, and task due date, submit button, and cancel button.
5. System validate input data with constraint.
6. System show a successful message “A new task is created.”
7. System display an error message “Invalid input data. Please recheck it again”.
8. System shall save data to database.

**URS-11: Students can edit tasks of a project.**

1. System shall provide UI to display student dashboard.
2. System shall provide UI to display the project detail page.
3. System shall provide UI to display lists and cards of tasks.
4. System shall provide UI which receive tasks name, task description, and task due date.
5. System shall save data to database.

**URS-12: Students can delete tasks of a project.**

1. System shall connect to database.
2. System shall provide UI to display student dashboard.
3. System shall provide UI to display the project detail page.
4. System shall provide UI to display lists and cards of tasks.
5. System shall provide a button to delete a task.
6. System shall provide a delete conformation message.
7. System shall save data to database.

**URS-13: Students can move tasks of a project.**

1. System shall provide UI to display student dashboard.
2. System shall provide UI to display the project detail page.
3. System shall provide UI to display lists and cards of tasks.
4. System shall provide a task card to drag and drop a task.
5. System shall display the task in another card column
6. System shall provide a forbidden mouse-cursor while dragging.
7. System shall provide a conformation message “Do you want to move this task?”.
8. System shall display a successful message “Move a task success”
9. System shall provide an error message if the move destination is not match with constraints “You cannot move this task to (destination path)”.
10. System shall save data to database.

**URS-14: Student can view comments from Mentor and Supervisor.**

1. System shall connect to database.
2. System shall provide UI to display a comment of mentor and supervisor in each task.

**URS-15: Students can add a comment.**

1. System shall provide a add comment button for adding comments of mentor and supervisor.
2. System shall provide a submit button for saving data.
3. System shall display a successful message “comment is added.”
4. System shall save data to database.

**URS-16: Student can delete their own comment.**

1. System shall provide a delete comment button.
2. System shall display a confirmation message “Do you want to delete this comment?”
3. System shall provide a cancel button.
4. System shall cancel to delete a comment.
5. System shall display a successful message “Comment is deleted.”.
6. System shall save data to database.

**URS-17: Mentors and Supervisor can view tasks of a student’s project.**

1. System shall connect to database.
2. System shall provide UI to display tasks of students who add registration code of mentor and supervisor.

**URS-18: Mentors and Supervisor can view the statistics of tasks.**

1. System shall connect to database.
2. System shall provide UI to display list of students.
3. System shall provide UI to display specific student’s dash board.
4. System shall provide UI to display statistics of student’s tasks.

**URS-19: Mentor can view comments in student’s report.**

1. System shall connect to database.
2. System shall provide “show report” button to display a student’s weekly report.
3. System shall provide UI to display a student’s weekly report.

**URS-20: Mentor can add comments in student’s report.**

1. System shall provide UI which receive comment of mentor in weekly form.
2. System shall display a confirmation message “Do you want to save this comment?”.
3. System shall display a successful message “Add comment success”.
4. System shall save data to database.

**URS-21: Mentor can edit comments in student’s report.**

1. System shall connect to database.
2. System shall provide UI to edit a comment of mentor in daily form.
3. System shall provide UI to edit a comment of mentor in weekly form.
4. System shall display a confirmation message “Do you want to edit this comment?”.
5. System shall display a successful message “Edit comment success”.
6. System shall save data to database.

**URS-22: Mentor can delete comments in student’s report.**

1. System shall connect to database.
2. System shall provide UI to display a list of mentor’s comments.
3. System shall provide delete button for deleting a comment.
4. System shall display a cancel button.
5. System shall display a confirmation message “Do you want to delete this comment?”.
6. System shall display a successful message “comment is deleted”
7. System shall save data to database.

**URS-23: Supervisor can view comments in student’s report.**

1. System shall connect to database.
2. System shall provide UI to display comment options.
3. System shall display comments of supervisor in daily or weekly form.

**URS-24: Supervisor can add comments in student’s report.**

1. System shall connect to database.
2. System shall provide UI of adding comment options.
3. System shall provide UI to add a comment of supervisor in daily form.
4. System shall provide UI to add a comment of supervisor in weekly form.
5. System shall provide a save and cancel button.
6. System shall cancel a comment.
7. System shall display a confirmation message “Do you want to add this comment?”.
8. System shall display a successful message “add comment success”.
9. System shall save data to database.

**URS-25: Supervisor can edit comments in student’s report.**

1. System shall connect to database.
2. System shall provide UI of editing comment options.
3. System shall provide UI to edit a comment of supervisor in daily form.
4. System shall provide UI to edit a comment of supervisor in weekly form.
5. System shall display a confirmation message “Do you want to edit this comment?”.
6. System shall display a successful message “Edit comment success”.
7. System shall save data to database.

**URS-26: Supervisor can delete comments in student’s report.**

1. System shall connect to database.
2. System shall provide UI to display a list of supervisor’s comments.
3. System shall provide delete button for deleting a comment.
4. System shall display a cancel button.
5. System shall display a confirmation message “Do you want to delete this comment?”.
6. System shall display a successful message “comment is deleted”
7. System shall save data to database.

**URS-27: Users can view weekly report.**

1. System shall connect to database.
2. System shall arrange all tasks based on time-stamp.
3. System shall separate tasks in weekly.
4. System shall merge keywords in a task with a sentence template.
5. System shall provide UI to display student’s weekly report content, comments, Mentor’s signature, student’s signature.

**URS-28: Users can generate a pdf file of weekly report.**

1. System shall convert html weekly report to pdf report.
2. System shall display a pdf file of weekly report.
3. System shall provide a download button to download a pdf file.
4. System shall provide a close button.
5. System shall provide a print button.

**URS-29: Users can download a pdf file of weekly report.**

1. System shall provide a download button.
2. System shall download a file to a local computer.

**URS-30: Student can receive comment notification message on Web application.**

1. System shall detect comments of mentor and supervisor.
2. System shall arrange comments based on time-stamp.
3. System shall display comment activities from the earliest to the latest comment in activities box (in dashboard).

**URS-31: Mentor and supervisor can receive activities notification message on Web application.**

1. System shall detect task activities of students who under their guidance.
2. System shall arrange task activities based on time-stamp.
3. System shall display task activities from the earliest to the latest activity in activities box (in dashboard).

**URS-32: Student can receive comment notification message by Email.**

1. System shall detect comments of mentor and supervisor.
2. System shall provide email template of comment notification.
3. System shall merge the content of comment to an email template.
4. System shall send a notification via email based on time-stamp of comment.

**URS-33: Mentor and supervisor can receive activities notification message by Email.**

1. System shall detect task activities of students who under their guidance.
2. System shall provide email template of task activities notification.
3. System shall merge the content of task to an email template.
4. System shall send a notification via email based on time-stamp of task.

# **Chapter IV | Requirement Specification**

## **Use case diagram**

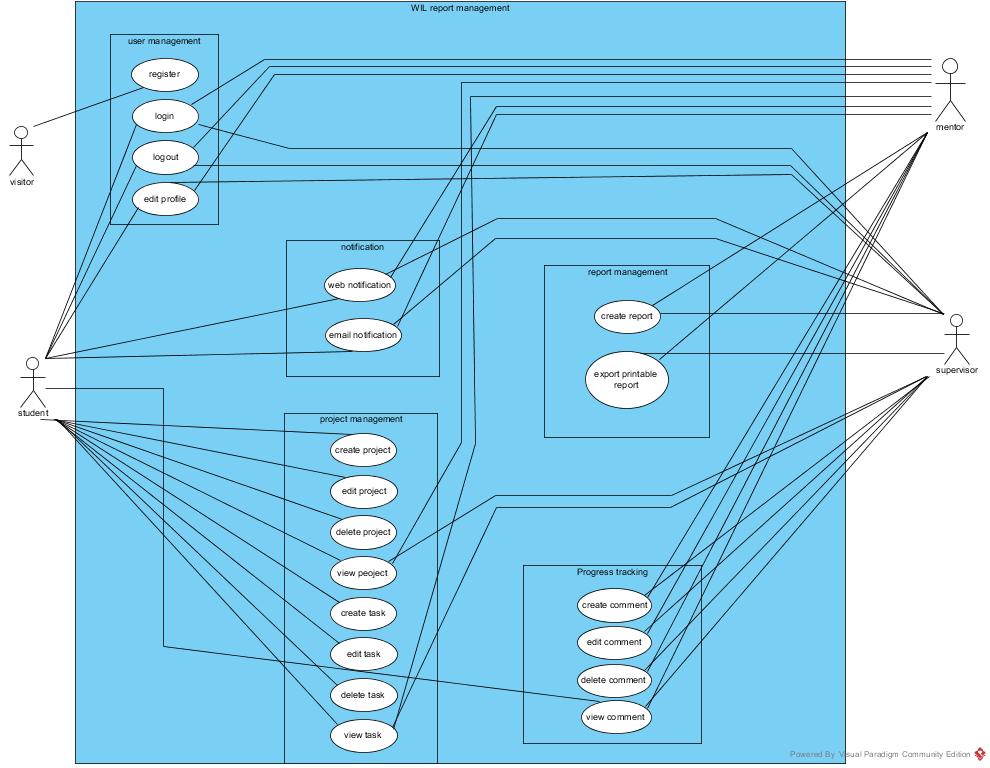


Figure 1 Use case diagram

## **Use case description**

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID:** | UC-01 | | |
| **Use Case Name:** | Visitor can register to the system. | | |
| **Created By:** | Phinthip | **Last Updated By:** | Phinthip |
| **Date Created:** | 26/02/2017 | **Last Revision Date:** | 07/03/2017 |
| **Actors:** | Visitor | | |
| **Description:** | Visitor can register to the WIL report management system. | | |
| **Trigger:** | Visitor enter to WIL report management web application. | | |
| **Preconditions:** | - | | |
| **Post conditions:** | Visitor can create a user account. | | |
| **Normal Flow:** | 1. System provides UI for the visitor to select the role of user. 2. A visitor selects a user role. | | |
| **Alternative Flows:** | a.) If visitor select “student” role:   1. System provides UI for the visitor to receive name, surname, email, password, student\_id, company name. 2. A visitor inputs name, surname, email, password, student\_id, company name. 3. A visitor clicks submit button. 4. System displays a confirmation message “Do you want to save your profile information?” 5. A visitor clicks ok button. 6. System validates name, surname, email, password, student\_id, company name. 7. System provides UI to display that the visitor registered successful. 8. System saves username, password, name, surname, email, student\_id, as a new user on the system with the default status for the visitor. 9. System saves data into the database.   b.) If a visitor selects “Mentor” or “Supervisor” role:   1. System provides UI for the visitor to receive name, surname, email, password, company name. 2. A visitor inputs name, surname, email, password, company name. 3. A visitor clicks submit button. 4. System displays a confirmation message “Do you want to save your profile information?” 5. A visitor clicks ok button. 6. System validates name, surname, email, password, company name. 7. System provides registration code for supervisor and mentor. 8. System provides UI to display that the visitor registered successful. 9. System saves data into the database. | | |
| **Exceptions:** | From a.5) and b.5) If visitor input incorrect format of name, surname, email, password, company name or empty forms.   1. System provides UI to display an error message if an input email is not matched with constraint. 2. System provides UI to display an error message if an input name is not matched with constraint. 3. System provides UI to display an error message if an input surname is not matched with constraint. 4. System provides UI to display an error message if an input email is not matched with constraint. 5. System provides UI to display an error message if an input password is not matched with constraint 6. System provides UI to display an error message if a company name is not matched with constraint.   From a.4) and b.4) If visitor click cancel button:   1. System redirects back to previous UI. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID:** | UC-02 | | |
| **Use Case Name:** | Users can edit their own profile. | | |
| **Created By:** | Phinthip | **Last Updated By:** | Phinthip |
| **Date Created:** | 26/02/2017 | **Last Revision Date:** | 07/03/2017 |
| **Actors:** | Student, Mentor, Supervisor | | |
| **Description:** | Users can edit their own profile. | | |
| **Trigger:** | Users click on edit profile button. | | |
| **Preconditions:** | UC-01, UC-03 System connects to database. | | |
| **Post conditions:** | Profile of users are updated into the database. | | |
| **Normal Flow:** | 1. System checks user\_id. 2. System provides UI for the student to edit information. | | |
| **Alternative Flows:** | From 2)  a.) If a user is student:   1. System provides UI to receive name, surname, email, password, student\_id, company name, and signature. 2. Student inputs name, surname, email, password, student\_id, company name, Mentor and Supervisor code, and signature. 3. Student clicks submit button. 4. System displays a confirmation message “Do you want to edit your profile information?” 5. Student clicks ok button. 6. System validates name, surname, email, password, student\_id, company name. 7. System displays successful message. 8. System saves username, password, name, surname, email, student\_id, as a new user on the system with the default status for the visitor. 9. System saves data into the database.   b.) If a user is “Mentor” or “Supervisor” role:   1. System provides UI to receive name, surname, email, password, company name, and signature. 2. Mentor or Supervisor inputs name, surname, email, password, company name, and signature. 3. Mentor or Supervisor clicks submit button. 4. System displays a confirmation message “Do you want to save your profile information?” 5. Mentor or Supervisor clicks ok button. 6. System validates name, surname, email, password, company name. 7. System provides registration code for supervisor and mentor. 8. System displays successful message. 9. System saves data into the database. | | |
| **Exceptions:** | From a.5) If student input incorrect format of name, surname, email, password, student\_id, company name or empty forms:   1. System provides UI to display an error message if the student edit their own information which the “name” is not matched with constraint. 2. System provides UI to display an error message if the student edit their own information which the “surname” is not matched with constraint. 3. System provides UI to display an error message if the student edit their own information which the “email” is not matched with constraint. 4. System provides UI to display an error message if the student edit their own information which the “password” is not matched with constraint. 5. System provides UI to display an error message if the student edit their own information which the “student\_id” is not matched with constraint. 6. System provides UI to display an error message if the student edit their own information which the “company name” is not matched with constraint. 7. System provides UI to display an error message if the student edit their own information which the “Mentor and supervisor code” does not exist in database.   From b.5) If Mentor or supervisor inputs incorrect format of name, surname, email, password, company name or empty forms:   1. System provides UI to display an error message if the mentor or supervisor edit their own information which the “name” is not matched with constraint. 2. System provides UI to display an error message if the mentor or supervisor edit their own information which the “surname” is not matched with constraint. 3. System provides UI to display an error message if the mentor or supervisor edit their own information which the “email” is not matched with constraint. 4. System s provides UI to display an error message if the mentor or supervisor edit their own information which the “password” is not matched with constraint. 5. System provides UI to display an error message if the mentor or supervisor edit their own information which the “company name” is not matched with constraint.   From a.4) and b.4) If a visitor clicks cancel button:   1. System redirects back to previous UI. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID:** | UC-03 | | |
| **Use Case Name:** | Users can login into the system. | | |
| **Created By:** | Phinthip | **Last Updated By:** | Phinthip |
| **Date Created:** | 26/02/2017 | **Last Revision Date:** | 07/03/2017 |
| **Actors:** | Student, Mentor, Supervisor | | |
| **Description:** | Users can login into the WIL report management system. | | |
| **Trigger:** | Users go to WIL report management web application.  Users click on login button. | | |
| **Preconditions:** | System redirect to dashboard. | | |
| **Post conditions:** | Users are directed to their role dash board.  User’s account session start. | | |
| **Normal Flow:** | 1. System shall provide UI to login 2. Users input email and password. 3. System shall receive email and password. 4. Users click ok button. 5. System shall verify email and password. 6. System shall provide UI to display the login successful. | | |
| **Alternative Flows:** | From 6.)  a.) If a user is student:   1. System shall navigates the student to access the student dashboard.   b.) If a user is Mentor:   1. System shall navigates the mentor to access the mentor dashboard.   c.) If a user is supervisor:   1. System shall navigate the supervisor to access the supervisor dashboard | | |
| **Exceptions:** | From4.) If a user click cancel button:   1. System shall go back to previous page.   From 5.) If email and password are invalid:   1. System shall display an error message that email or password are wrong. 2. Users click ok button. 3. System shall go back to previous page. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID:** | UC-04 | | |
| **Use Case Name:** | Users can logout from the system. | | |
| **Created By:** | Phinthip | **Last Updated By:** | Phinthip |
| **Date Created:** | 26/02/2017 | **Last Revision Date:** | 07/03/2017 |
| **Actors:** | Student, Mentor, Supervisor | | |
| **Description:** | Users can logout from the WIL report management system. | | |
| **Trigger:** | Users click on logout button. | | |
| **Preconditions:** | UC-03 | | |
| **Post conditions:** | User’s account session is expired. | | |
| **Normal Flow:** | 1. System shall provide UI to logout. 2. Users click logout button. 3. System shall display a conformation message. 4. Users click ok button. 5. System shall logout from the system. 6. System shall display that the logout successful message. 7. System shall redirect to login page. | | |
| **Alternative Flows:** | - | | |
| **Exceptions:** | From 4.) If user click cancel button:   1. System go back to previous UI. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID:** | UC-05 | | |
| **Use Case Name:** | Student can add registration code. | | |
| **Created By:** | Phinthip | **Last Updated By:** | Phinthip |
| **Date Created:** | 26/02/2017 | **Last Revision Date:** | 07/03/2017 |
| **Actors:** | Student | | |
| **Description:** | Student can add registration code in edit profile page. | | |
| **Trigger:** | Users select Mentor and Supervisor code input field. | | |
| **Preconditions:** | UC-01, UC-02, UC-03 Student has to know their mentor and supervisor code. | | |
| **Post conditions:** | Account of student connect to mentor and supervisor account. | | |
| **Normal Flow:** | 1. System shall provide UI to receive a code. 2. Users click submit button. 3. System shall valid (a) code(s). 4. System display successful message. | | |
| **Alternative Flows:** | From 1.) If student would like to add more codes:   1. Users click “+” button. 2. System shall provide an add button for adding another code.   From 4.) If student slick cancel button:   1. System go back to previous UI. | | |
| **Exceptions:** | From 5) If (a) code(s) is/are invalid:   1. System shall display an error message 2. System go back to previous UI. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID:** | UC-06 | | |
| **Use Case Name:** | Users can view tasks of a project. | | |
| **Created By:** | Phinthip | **Last Updated By:** | Phinthip |
| **Date Created:** | 27/02/2017 | **Last Revision Date:** | 07/03/2017 |
| **Actors:** | Student, Mentor, Supervisor | | |
| **Description:** | Student, Mentor, Supervisor can view tasks of each project. | | |
| **Trigger:** | Student clicks on a project name. | | |
| **Preconditions:** | UC-01, UC-02, UC-03, UC-05 System connects to database.  Users are in dashboard. | | |
| **Post conditions:** | - | | |
| **Normal Flow:** | 1. System provide UI to display a dashboard. | | |
| **Alternative Flows:** | From 1.) If a user is students:   1. System provide UI to display a dashboard with list pf project and statistics. 2. Users click to a project list. 3. System links to project detail page. 4. System displays list and card of tasks.   System display list and card of tasks.  From 1.) If a user is mentor or supervisor:   1. System provides list of students in dashboard. 2. Mentor or supervisor clicks on a student list. 3. System links to dashboard of selected student. 4. Users click to a project list. 5. System links to project detail page. 6. System displays list and card of tasks. | | |
| **Exceptions:** | - | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID:** | UC-07 | | |
| **Use Case Name:** | Users can view the statistics of tasks. | | |
| **Created By:** | Phinthip | **Last Updated By:** | Phinthip |
| **Date Created:** | 26/02/2017 | **Last Revision Date:** | 07/03/2017 |
| **Actors:** | Student, Mentor, Supervisor | | |
| **Description:** | Student can view statistic of tasks. Mentor and Supervisor can view statistics of tasks in specific student (who are under their guidance). | | |
| **Trigger:** | Users click to see dashboard. | | |
| **Preconditions:** | UC-01, UC-02, UC-03, UC-05 System connects to database. | | |
| **Post conditions:** | - | | |
| **Normal Flow:** | 1. System provide UI to display a dashboard. | | |
| **Alternative Flows:** | From 1.) If a user is student:   1. System provide UI to display a dashboard. 2. System display statistics box of tasks on the top of dashboard.   From 1.) If a user is mentor or supervisor:   1. System provide list of students in dashboard. 2. Mentor or supervisor click on a student list. 3. System links to dashboard of selected student. 4. System display statistics box of tasks on the top of dashboard. | | |
| **Exceptions:** | - | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID:** | UC-08 | | |
| **Use Case Name:** | Students can add a project. | | |
| **Created By:** | Phinthip | **Last Updated By:** | Phinthip |
| **Date Created:** | 28/02/2017 | **Last Revision Date:** | 07/03/2017 |
| **Actors:** | Student | | |
| **Description:** | Students can create a new project. | | |
| **Trigger:** | Student click on “create new project” button. | | |
| **Preconditions:** | UC-03 Student is in a dashboard. | | |
| **Post conditions:** | System save data into the database.  System redirect project detail page. | | |
| **Normal Flow:** | 1. System provide “create new project” button in a student’s dashboard. 2. Student click on “add new project” button. 3. System shall provide UI which receive project name, project description, and due date. 4. System provide “submit” and “cancel” button. 5. Users click on submit button. 6. System valid input data constraint. 7. System display a successful message. | | |
| **Alternative Flows:** | From 4.) If s student clicks on “cancel” button:   1. System redirect to previous page. | | |
| **Exceptions:** | From 6.) If the input data is invalid:   1. System display an error message “Invalid input data. Please recheck it again ” 2. User click “Ok” button. 3. System redirect to create a new project detail page with the previous input data. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID:** | UC-09 | | |
| **Use Case Name:** | Student can delete a project. | | |
| **Created By:** | Phinthip | **Last Updated By:** | Phinthip |
| **Date Created:** | 28/02/2017 | **Last Revision Date:** | 07/03/2017 |
| **Actors:** | Student | | |
| **Description:** | Student can delete a project. | | |
| **Trigger:** | Student click on “delete” button. | | |
| **Preconditions:** | UC-03  System connect to database.  Student has at least 1 project in their project list.  Student is in a dashboard. | | |
| **Post conditions:** | A project is removed from the database. | | |
| **Normal Flow:** | 1. System provide UI to display student’s dashboard. 2. Student click on projects list. 3. System display the project detail page. 4. System provide delete button for deleting a project. 5. Student click on delete a project button. 6. System display a confirmation message “Are you sure to delete this project?” with “yes” and “no” button 7. Student click “yes” button. | | |
| **Alternative Flows:** | - | | |
| **Exceptions:** | From 6.) If student click “no” button:   1. System redirect to previous page. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID:** | UC-10 | | |
| **Use Case Name:** | Students can add tasks of a project. | | |
| **Created By:** | Phinthip | **Last Updated By:** | Phinthip |
| **Date Created:** | 28/02/2017 | **Last Revision Date:** | 07/03/2017 |
| **Actors:** | Student | | |
| **Description:** | Student can create a task of a project. | | |
| **Trigger:** | Student click on “create new task” button. | | |
| **Preconditions:** | UC-03 System connects to database.  Student is in a dashboard. | | |
| **Post conditions:** | A new task is added into a database. | | |
| **Normal Flow:** | 1. System displays student’s dashboard. 2. Student click on a list item in project list. 3. System displays the project detail page. 4. Student clicks on “add new task” button. 5. System shall provide UI which receive tasks name, task description, task due date, submit button, and cancel button. 6. Student click on submit button. 7. System validate input data with constraint. 8. System show a successful message “A new task is created.” | | |
| **Alternative Flows:** | From 5.) If student click on cancel button:   1. System redirect to previous page. | | |
| **Exceptions:** | From 7.) If input data is invalid:   1. System cancel to add a new data. 2. System display an error message “Invalid input data. Please recheck it again”. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID:** | UC-11 | | |
| **Use Case Name:** | Students can edit tasks of a project. | | |
| **Created By:** | Phinthip | **Last Updated By:** | Phinthip |
| **Date Created:** | 28/02/2017 | **Last Revision Date:** | 07/03/2017 |
| **Actors:** | Student | | |
| **Description:** | Students can edit tasks of each project. | | |
| **Trigger:** | Student click the edit task button. | | |
| **Preconditions:** | UC-03, UC-08, UC-10  System connects to database.  Student is in a project detail page. | | |
| **Post conditions:** | System redirect to project detail page. | | |
| **Normal Flow:** | 1. Student click “edit task” button. 2. System provide UI to receive tasks name, task description, task due date, submit button, and cancel button. 3. Student click on submit button. 4. System validate input data with constraint. 5. System show a successful message “Task was edited.” | | |
| **Alternative Flows:** | From 2.) If student click on cancel button:   1. System redirect to previous page. | | |
| **Exceptions:** | From 4.) If input data is invalid:   1. System cancel to add a new data.   System display an error message “Invalid input data. Please recheck it again”. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID:** | UC-12 | | |
| **Use Case Name:** | Students can delete tasks of a project. | | |
| **Created By:** | Phinthip | **Last Updated By:** | Phinthip |
| **Date Created:** | 03/03/2017 | **Last Revision Date:** | 07/03/2017 |
| **Actors:** | Student | | |
| **Description:** | Students can delete tasks of a project. | | |
| **Trigger:** | Student click “delete task” button. | | |
| **Preconditions:** | UC-03, UC-08, UC-10  System connects to database.  Student is in a project detail page.  There is a task list in selected project. | | |
| **Post conditions:** | System redirect to project detail page. | | |
| **Normal Flow:** | 1. Student select a task in the list of task. 2. Student click “delete task” button. 3. System provide UI to display a confirmation message “Do you want to delete this task?” with OK and Cancel button. 4. Student click “OK” button. 5. The task is deleted from the database. 6. System display a successful message “The task was deleted” | | |
| **Alternative Flows:** | From 4.) If student click “Cancel” button:   1. System redirect to project detail page. | | |
| **Exceptions:** | - | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID:** | UC-13 | | |
| **Use Case Name:** | Students can move tasks of a project. | | |
| **Created By:** | Phinthip | **Last Updated By:** | Phinthip |
| **Date Created:** | 03/03/2017 | **Last Revision Date:** | 07/03/2017 |
| **Actors:** | Student | | |
| **Description:** | Students can move tasks of a project. | | |
| **Trigger:** | Student drag a task card.  Student click “move task” button. | | |
| **Preconditions:** | UC-03, UC-08, UC-10  System connects to database.  Student is in a project detail page.  There is a task list in selected project. | | |
| **Post conditions:** | - | | |
| **Normal Flow:** | Flow 1.) Drag and drop   1. Student drag a task card and drop it to another column. 2. System arrange the task to the last order. 3. System shall save data to database.   Flow 2.) Click “move task” button   1. Student click on “move task” button. 2. System provide UI to display dropdown of column option, OK button, and cancel button. 3. Student select the destination column. 4. Student click OK button. 5. System move the selected task to the destination button and display a successful message. | | |
| **Alternative Flows:** | From 1-1.) If student drag a task card and drop it to the same column   1. The task is still in the current column.   From 2-4.) If student select cancel button   1. Student click Cancel button. 2. System cancel to move the task. 3. System redirect to list of tasks. | | |
| **Exceptions:** | From 1.) If student drag a task card and does not drop in any column.   1. System display forbidden sign as a mouse cursor. 2. The task goes to the current column. | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID:** | UC-14 | | |
| **Use Case Name:** | User can view a comment. | | |
| **Created By:** | Phinthip | **Last Updated By:** | Phinthip |
| **Date Created:** | 03/03/2017 | **Last Revision Date:** | 07/03/2017 |
| **Actors:** | Student | | |
| **Description:** | Student, Mentor, Supervisor can view comments from Mentor and Supervisor. | | |
| **Trigger:** | Users click to see a task detail. | | |
| **Preconditions:** | UC-03, UC-05, UC-08, UC-10  System connects to database.  Student is in a prokect detail page.  There is a comment in a system.  There is a task list in selected project. | | |
| **Post conditions:** | - | | |
| **Normal Flow:** | 1. Users click to see a task detail. 2. System request comments data from database. 3. System provide UI to display (a) comment(s). 4. Users can see the comment(s). | | |
| **Alternative Flows:** | If there is no comment, system will not provide UI for displaying comment. | | |
| **Exceptions:** | - | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID:** | UC-15 | | |
| **Use Case Name:** | Users can add a comment. | | |
| **Created By:** | Phinthip | **Last Updated By:** | Phinthip |
| **Date Created:** | 03/03/2017 | **Last Revision Date:** | 07/03/2017 |
| **Actors:** | Student, Mentor, and Supervisor | | |
| **Description:** | Student, Mentor, and Supervisor can add a comment(s). | | |
| **Trigger:** | Users click “add comment” button. | | |
| **Preconditions:** | UC-03, UC-05, UC-08, UC-10, UC-14  Student is in a task detail page.  There is a task in the system.  There is a comment in a system. | | |
| **Post conditions:** | System links back to a task detail page. | | |
| **Normal Flow:** | 1. System provide “Add comment” button. 2. Users click “Add comment” button. 3. System provide input field, OK button, and close button. 4. Users fill the text(s) in. 5. Users click OK button. 6. System save the comment data into the database. 7. System display the message. | | |
| **Alternative Flows:** | From 5.) If a user select close button   1. The input field is closed. 2. System cancel to add a comment. 3. System redirect to the previous UI. | | |
| **Exceptions:** | - | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID:** | UC-16 | | |
| **Use Case Name:** | Users can delete their own comment. | | |
| **Created By:** | Phinthip | **Last Updated By:** | Phinthip |
| **Date Created:** | 03/03/2017 | **Last Revision Date:** | 07/03/2017 |
| **Actors:** | Student, Mentor, and Supervisor | | |
| **Description:** | Student, Mentor, and Supervisor can delete their own comment. | | |
| **Trigger:** | Users click on the cross button on the top right of the comment box. | | |
| **Preconditions:** | UC-03, UC-05  There is a comment in the system. | | |
| **Post conditions:** | - | | |
| **Normal Flow:** | 1. System provide UI to delete their own comment. 2. Users click on the cross button on the top right of the comment box. 3. System display a confirmation message “Do you want to delete this comment?”. 4. Users click OK button. 5. System request database to delete the comment. 6. System display successful message “Comment is deleted.” | | |
| **Alternative Flows:** | From 4.) If a user click on cancel button.   1. Users click on Cancel button. 2. System cancel to delete a comment. 3. System redirect to the pervious UI. | | |
| **Exceptions:** | - | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID:** | UC-17 | | |
| **Use Case Name:** | Users can edit a comment. | | |
| **Created By:** | Phinthip | **Last Updated By:** | Phinthip |
| **Date Created:** | 04/03/2017 | **Last Revision Date:** | 07/03/2017 |
| **Actors:** | Student, Mentor, Supervisor | | |
| **Description:** | Users click “edit comment” button. | | |
| **Trigger:** | UC-03, UC-05, UC-08, UC-10, UC-14  Student is in a task detail page.  There is a task in the system.  There is a comment in a system. | | |
| **Preconditions:** | System links back to a task detail page. | | |
| **Post conditions:** | - | | |
| **Normal Flow:** | 1. System provide “Edit comment” button. 2. Users click “Edit comment” button. 3. System provide input field with existing comment, OK button, and close button. 4. Users fill the text(s) in. 5. User click OK button. 6. System send a request to save the comment data into the database. 7. System display the user’s comment. | | |
| **Alternative Flows:** | From 5.) If a user select close button   1. The input field is closed. 2. System cancel to edit a comment. 3. System redirect to the previous UI. | | |
| **Exceptions:** | - | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID:** | UC-18 | | |
| **Use Case Name:** | Users can view weekly report. | | |
| **Created By:** | Phinthip | **Last Updated By:** | Phinthip |
| **Date Created:** | 04/03/2017 | **Last Revision Date:** | 07/03/2017 |
| **Actors:** | Student, Mentor, Supervisor | | |
| **Description:** | Student, Mentor, Supervisor can view a weekly report. | | |
| **Trigger:** | Users click “Report” menu.  Users click “view report” button. | | |
| **Preconditions:** | UC-03  There are tasks in a system. | | |
| **Post conditions:** | - | | |
| **Normal Flow:** | Flow 1.) Users click “Report” menu.   1. Users click “Report” on left navigation menu. 2. System arrange all tasks based on time-stamp. 3. System separates tasks in weekly. 4. System merges keywords in a task with a sentence template. 5. System provide UI to display overall weekly report.   Flow 2) Users click “view report” button.  Users click “view report” button on dashboard.   1. System arrange all tasks based on time-stamp. 2. System separates tasks in weekly. 3. System merges keywords in a task with a sentence template. 4. System provide UI to display overall weekly report. | | |
| **Alternative Flows:** | - | | |
| **Exceptions:** | - | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID:** | UC-19 | | |
| **Use Case Name:** | Users can generate a pdf file of weekly report. | | |
| **Created By:** | Phinthip | **Last Updated By:** | Phinthip |
| **Date Created:** | 04/03/2017 | **Last Revision Date:** | 07/03/2017 |
| **Actors:** | Student, Mentor, Supervisor | | |
| **Description:** | Student, Mentor, Supervisor can a pdf file of weekly report. | | |
| **Trigger:** | Users click on “Export report” button. | | |
| **Preconditions:** | UC-03, UC-18  There are tasks in a system. | | |
| **Post conditions:** | - | | |
| **Normal Flow:** | 1. Users click “Export report” button. 2. System converts html weekly report to pdf report. 3. System provide UI to display a pdf file of weekly report. | | |
| **Alternative Flows:** | - | | |
| **Exceptions:** | - | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID:** | UC-20 | | |
| **Use Case Name:** | Users can download a pdf file of weekly report. | | |
| **Created By:** | Phinthip | **Last Updated By:** | Phinthip |
| **Date Created:** | 04/03/2017 | **Last Revision Date:** | 07/03/2017 |
| **Actors:** | Student, Mentor, Supervisor | | |
| **Description:** | Student, Mentor, Supervisor can download a pdf file of weekly report. | | |
| **Trigger:** | Users click on “download” button. | | |
| **Preconditions:** | UC-03, UC-18, UC-19  There are tasks in a system. | | |
| **Post conditions:** | - | | |
| **Normal Flow:** | 1. System provide UI to display a pdf file of weekly report with download and close button. 2. Users click “download” button. 3. System request to download a file from database to a local computer. | | |
| **Alternative Flows:** | From 2.) If a user clicks “cancel” button.   1. System cancel to download a pdf file. 2. System redirect to weekly report UI. | | |
| **Exceptions:** | - | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID:** | UC-21 | | |
| **Use Case Name:** | Users can receive notification via Web application. | | |
| **Created By:** | Phinthip | **Last Updated By:** | Phinthip |
| **Date Created:** | 04/03/2017 | **Last Revision Date:** | 07/03/2017 |
| **Actors:** | Student, Mentor, Supervisor | | |
| **Description:** | Student, Mentor, Supervisor can receive notification via Web application | | |
| **Trigger:** | An activity is occurred. (A new comment is created, A new project is created, A project is edited, A new task is created, A task is edited.) | | |
| **Preconditions:** | UC-03 | | |
| **Post conditions:** | - | | |
| **Normal Flow:** | If a user is a student:   1. System can detect comments of mentor and supervisor. 2. System arrange comments based on time-stamp. 3. System displays comment activities from the earliest to the latest comment in activities box (in dashboard).   If a user is a mentor or supervisor:   1. System shall detect task activities of students who under their guidance. 2. System shall arrange task activities based on time-stamp. 3. System shall display task activities from the earliest to the latest activity in activities box (in dashboard). | | |
| **Alternative Flows:** | - | | |
| **Exceptions:** | - | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **Use Case ID:** | UC-22 | | |
| **Use Case Name:** | Users can receive notification via Email. | | |
| **Created By:** | Phinthip | **Last Updated By:** | Phinthip |
| **Date Created:** | 04/03/2017 | **Last Revision Date:** | 07/03/2017 |
| **Actors:** | Student, Mentor, Supervisor | | |
| **Description:** | Student, Mentor, Supervisor can receive notification via Email. | | |
| **Trigger:** | An activity is occurred. (A new comment is created, A new project is created, A project is edited, A new task is created, A task is edited.) | | |
| **Preconditions:** | UC-03  System have emails of users. | | |
| **Post conditions:** | - | | |
| **Normal Flow:** | If a user is a student:   1. System can detect comments of mentor and supervisor. 2. System provide email template of comment notification. 3. System merge the content of comment to an email template. 4. System send a notification via email based on time-stamp of comment.   If a user is a mentor or supervisor:   1. System can detect task activities of students who under their guidance. 2. System provide email template of task activities notification. 3. System merge the content of task to an email template. 4. System send a notification via email based on time-stamp of task. | | |
| **Alternative Flows:** | - | | |
| **Exceptions:** | - | | |