Use Cases

for

Hydra-TowerDefence

Version 3.0 approved

**Prepared by Seenivasan Sashwath Ravilla
Mohamed Shafiq Bin Peer Mohamed
Lim Wi Teow
Heather Chew
Koo Jian Yang
Yeong Wei Xian
Tan Wen Jie
Lim Si Yi**

Team Hydra

13/4/2021

Revision History

Name	Date	Reason For Changes	Version
Lim Wi Teow	14/2/2021	Creation of Document	1.0
Seenivasan Sashwath Ravilla	28/2/2021	Adding of Use Case Description	1.1
Lim Wi Teow			
Seenivasan Sashwath Ravilla	28/2/2021	Adding of Use Case Diagram	1.2
Seenivasan Sashwath Ravilla	24/3/2021	Editing both Use Case Diagram and Description	2.0
Lim Wi Teow	13/4/2021	Finalizing both Use Case Diagram and Description	3.0

Guidance for Use Case Template

Document each use case using the template shown in the Appendix. This section provides a description of each section in the use case template.

Use Case Identification

Use Case ID

Give each use case a unique numeric identifier, in hierarchical form: X.Y. Related use cases can be grouped in the hierarchy. Functional requirements can be traced back to a labeled use case.

Use Case Name

State a concise, results-oriented name for the use case. These reflect the tasks the user needs to be able to accomplish using the system. Include an action verb and a noun. Some examples:

- View part number information.
- Manually mark hypertext source and establish link to target.
- Place an order for a CD with the updated software version.

Use Case History

Created By

Supply the name of the person who initially documented this use case.

Date Created

Enter the date on which the use case was initially documented.

Last Updated By

Supply the name of the person who performed the most recent update to the use case description.

Date Last Updated

Enter the date on which the use case was most recently updated.

Use Case Definition

Actor

An actor is a person or other entity external to the software system being specified who interacts with the system and performs use cases to accomplish tasks. Different actors often correspond to different user classes, or roles, identified from the customer community that will use the product. Name the actor(s) that will be performing this use case.

Description

Provide a brief description of the reason for and outcome of this use case, or a high-level description of the sequence of actions and the outcome of executing the use case.

Preconditions

List any activities that must take place, or any conditions that must be true, before the use case can be started. Number each precondition. Examples:

1. User's identity has been authenticated.
2. User's computer has sufficient free memory available to launch task.

Postconditions

Describe the state of the system at the conclusion of the use case execution. Number each postcondition. Examples:

1. Document contains only valid SGML tags.
2. Price of item in database has been updated with new value.

Priority

Indicate the relative priority of implementing the functionality required to allow this use case to be executed. The priority scheme used must be the same as that used in the software requirements specification.

Frequency of Use

Estimate the number of times this use case will be performed by the actors per some appropriate unit of time.

Flow of Events

Provide a detailed description of the user actions and system responses that will take place during execution of the use case under normal, expected conditions. This dialog sequence will ultimately lead to accomplishing the goal stated in the use case name and description. This description may be written as an answer to the hypothetical question, "How do I <accomplish the task stated in the use case name>?" This is best done as a numbered list of actions performed by the actor, alternating with responses provided by the system.

Alternative Flows

Document other, legitimate usage scenarios that can take place within this use case separately in this section. State the alternative course, and describe any differences in the sequence of steps that take place. Number each alternative course using the Use Case ID as a prefix, followed by "AC" to indicate "Alternative Course". Example: X.Y.AC.1.

Exceptions

Describe any anticipated error conditions that could occur during execution of the use case, and define how the system is to respond to those conditions. Also, describe how the system is to respond if the use case execution fails for some unanticipated reason. Number each exception using the Use Case ID as a prefix, followed by "EX" to indicate "Exception". Example: X.Y.EX.1.

Includes

List any other use cases that are included ("called") by this use case. Common functionality that appears in multiple use cases can be split out into a separate use case that is included by the ones that need that common functionality.

Special Requirements

Identify any additional requirements, such as nonfunctional requirements, for the use case that may need to be addressed during design or implementation. These may include performance requirements or other quality attributes.

Assumptions

List any assumptions that were made in the analysis that led to accepting this use case into the product description and writing the use case description.

Notes and Issues

List any additional comments about this use case or any remaining open issues or TBDs (To Be Determined) that must be resolved. Identify who will resolve each issue, the due date, and what the resolution ultimately is.

Use Case Diagram
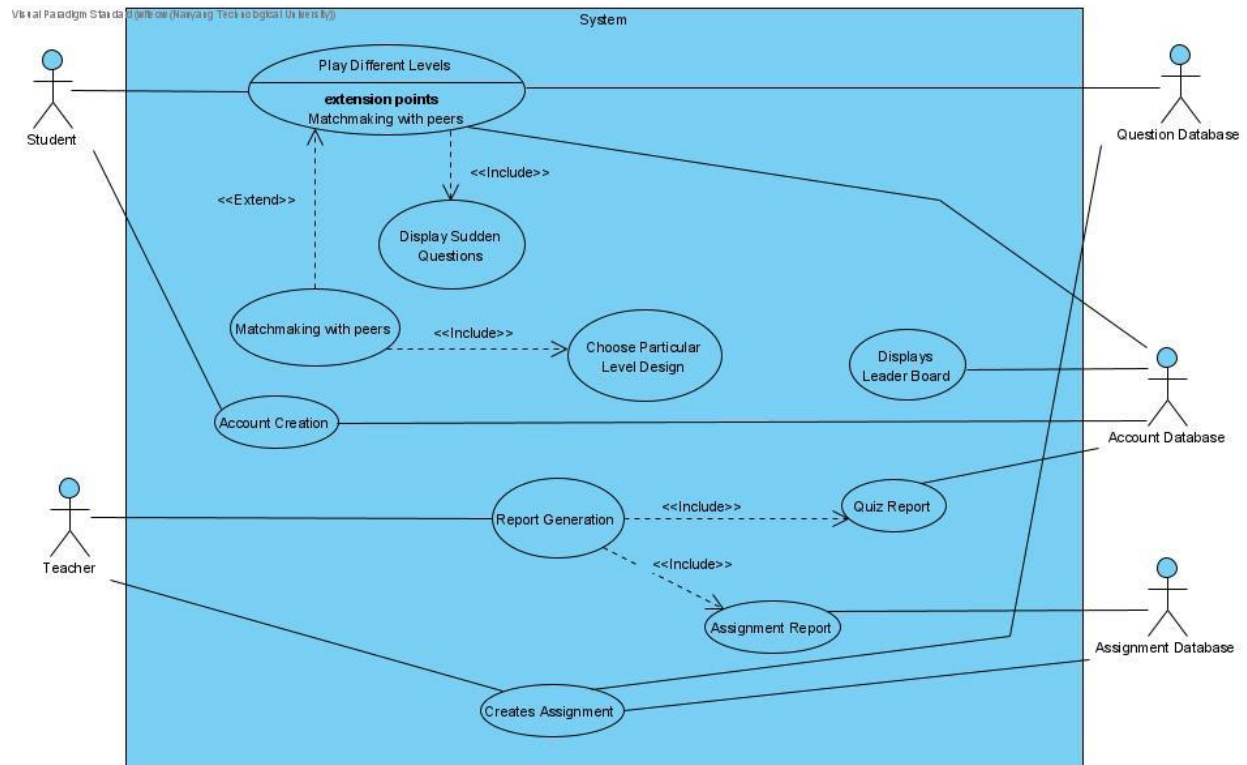
Use Case Description

Use Case ID:	1		
Use Case Name:	Play different levels		
Created By:	Sashwath	Last Updated By:	Lim Wi Teow
Date Created:	14/02/2021	Date Last Updated:	14/02/2021

Actor:	Student, Question Database
Description:	Student can select from a range of levels to play
Preconditions:	1. Student is logged in
Postconditions:	1. Scores earned count towards the leaderboard
Priority:	High
Frequency of Use:	Every time the game is opened
Flow of Events:	1. Student selects a world to go into. 2. Student will then select the relevant section that they want to play. 3. Student is given questions to answer. 4. Student earns additional points if he answers the questions correctly. 5. The points earned gives the student power-up items to use in the game be it buying new turrets or upgrading current turrets. 6. The game starts, the student will be asked questions based on his proficiency level. 7. The student will start with 5 lives and will be required to defend their tower. Every enemy that manages to 'hit' the tower will decrease the lives counter by 1. 8. Ideally when the student cleared all waves, the game will end, and a score will be assigned based on their performance 9. If the game ends when student has 0 lives, it will end showing the number of waves completed and how many waves are supposed to appear.
Alternative Flows:	AF1: Instead of playing in story mode, students can also matchmake with their peers.
Exceptions:	None
Includes:	1. Includes displaying sudden questions during the game. a. Students can choose to click on additional questions to answer during the game to gain more points to buy better turrets.
Special Requirements:	None
Assumptions:	1. For the quiz before the start of the game as well as the sudden questions, the questions will be selected at random. Hence, it is assumed that the database will potentially have a larger questions database as a small question pool will only mean that the same questions will appear every time.
Notes and Issues:	None

Use Case ID:	1.1		
Use Case Name:	Display sudden questions		
Created By:	Sashwath	Last Updated By:	Lim Wi Teow
Date Created:	14/02/2021	Date Last Updated:	14/02/2021

Actor:	Student, Question database
Description:	Additional questions can be answered during the game.
Preconditions:	1. Student is logged in. 2. Student is currently playing a level or a custom game.
Postconditions:	None
Priority:	Medium
Frequency of Use:	Maximum of 5 times for every game
Flow of Events:	1. During the time frame between the end of a wave and the start of a new wave, students can click on the Additional Questions button to attempt more questions. 2. These questions will give the students extra currency to potentially buy more turrets or update the existing turrets. 3. At every time that button is clicked, the student will answer 5 random questions generated from the Question Database. 4. Once the 5 questions are answered, the game will resume and the student will have more currency than they had initially if they answered any of the sudden questions correctly.
Alternative Flows:	AF1: If student gets all the questions wrong 1. Students will get no additional currency to play.
Exceptions:	None
Includes:	None
Special Requirements:	None
Assumptions:	This is assuming that the question database has enough questions to generate different random questions every time. Currently since our database is limited, the same questions will appear.
Notes and Issues:	None

Use Case ID:	2		
Use Case Name:	Matchmaking with peers		
Created By:	Sashwath	Last Updated By:	Lim Wi Teow
Date Created:	14/02/2021	Date Last Updated:	14/02/2021

Actor:	Student, Question database, Account Database
Description:	Student can create a custom game to challenge other students.
Preconditions:	1. Student is logged in. 2. There are 2 students that are ready to play against each other.
Postconditions:	1. Scores earned do not count towards the leaderboard.
Priority:	High
Frequency of Use:	Every time the game is opened
Flow of Events:	1. Student selects the option to create a custom PVP game. 2. Student selects the world and section he wants to compete in. 3. System generates the map and an access code. 4. Other student enters the same access code and joins the custom level. 5. Students will be playing against each other by earning more points based on the number of correct answers to the questions as well as their progress in the game. 6. The game format will be the same as playing in the story mode. 7. After the game ends, the winner will be stated on both players' screens.
Alternative Flows:	AF1: Student has not completed levels 1-3 in the selected world 1. Student will be alerted that he has not completed the previous level. 2. Student creates custom game using a previous level. 3. Continue from step 3 in the normal flow of events.
Exceptions:	EX1: Student inputs the wrong access code 1. Student will be alerted that the wrong access code is entered. 2. Student will be prompted to re-enter the access code.
Includes:	1. Choose particular level design a. Students can select which world and section they will want to compete in with their friends.
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

Use Case ID:	3		
Use Case Name:	Account Creation		
Created By:	Sashwath	Last Updated By:	Lim Wi Teow
Date Created:	14/02/2021	Date Last Updated:	14/04/2021

Actor:	Student, Account Database
Description:	Students will have to create an account to play the game.
Preconditions:	1. Student must be taking Software Engineering for that current semester.
Postconditions:	None
Priority:	High
Frequency of Use:	Once for every user
Flow of Events:	1. Student will click on the 'Create New Account' button on the Main Menu to start this process. 2. Student will be required to enter their desired username. 3. Student will be prompted to key in their desired password twice. 4. Student can then click on 'Register' to proceed. 5. Students then will be allowed to choose an Avatar of their choice: Fire, Air, Earth and Water.
Alternative Flows:	None
Exceptions:	EX.1 When there are blank fields for Username or Password, an error message will display stating that one of the fields are empty. EX.1 When an existing username is given, an error message will display stating that the Username already exist.
Includes:	None
Special Requirements:	None
Assumptions:	Students will only create 1 account.
Notes and Issues:	None

Use Case ID:	4		
Use Case Name:	Displays leaderboard		
Created By:	Lim Wi Teow	Last Updated By:	Lim Wi Teow
Date Created:	14/02/2021	Date Last Updated:	14/02/2021

Actor:	Account Database
Description:	In the game, students can view the leaderboard displayed with the students' name and their corresponding scores.
Preconditions:	1. Students will need to create an account in the game.
Postconditions:	None
Priority:	Low
Frequency of Use:	Anytime whenever the Leaderboard Button is selected
Flow of Events:	1. Student selects the option to view the leaderboard. 2. At the main page, the leaderboard in general for all worlds will be displayed, displaying the top 10 students with the highest scores.
Alternative Flows:	None
Exceptions:	EX1: In the event that there are fewer than 10 students that have played the level, the empty listings will be represented with a dash '- '.
Includes:	None
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

Use Case ID:	5		
Use Case Name:	Report Generation		
Created By:	Lim Wi Teow	Last Updated By:	Lim Wi Teow
Date Created:	14/02/2021	Date Last Updated:	14/02/2021

Actor:	Teacher
Description:	Teachers will be able to extract scores from the Score Database for each student.
Preconditions:	1. Teachers must have an account in order to access the game. 2. Students have at least played the game once to have a score.
Postconditions:	None
Priority:	Medium
Frequency of Use:	Anytime
Flow of Events:	1. When teachers are logged into the game, they will be led to a main menu that allows them to generate a report for either assignments or quiz. 2. Assignments refer to the question set generated by the teachers with an access code for the student to use. 3. Quiz refers to the question set that appears before every game for the story mode.
Alternative Flows:	None
Exceptions:	None
Includes:	1. Generate Quiz Report a. Proficiency for the respective levels in each world. 2. Generate Assignment Report a. Report of the latest 5 Assignments b. Includes details like start and end date, the current average score of the student, as well as the standard deviation.
Special Requirements:	None
Assumptions:	None
Notes and Issues:	None

Use Case ID:	6		
Use Case Name:	Create Assignment		
Created By:	Lim Wi Teow	Last Updated By:	Lim Wi Teow
Date Created:	13/04/2021	Date Last Updated:	13/04/2021

Actor:	Teacher, Question Database, Assignment Database
Description:	The Teacher can create questions for students as an Assignment.
Preconditions:	1. Teacher has access to the system. 2. Teacher has a social media account for minimally either Twitter or Facebook. 3. The students are keeping track of the teacher's social media posts.
Postconditions:	1. Assignment will be created that will automatically stops students from answering two weeks after its creation. 2. Students' Score will be updated in the Assignment Database
Priority:	High
Frequency of Use:	Anytime
Flow of Events:	1. Teachers will have to login with their own account to access this feature. 2. Teachers will then need to state how many questions will there be in the Assignment as well as the access code. 3. Teachers are required to fill in all of the question fields including the Title, Answer and all four options. 4. Teachers have the option to select questions from the current database and edit from there accordingly. 5. Once all the required fields are filled up, the teacher will be brought to the social media sharing page. 6. This is when the teacher can either share through Facebook and Twitter. 7. Assignments created under that specific Access Code will have a lifespan of 2 weeks and will close once the time is up.
Alternative Flows:	None
Exceptions:	None
Includes:	None
Special Requirements:	None
Assumptions:	None
Notes and Issues:	This feature also allows teachers to add questions to existing assignments, however, the time limit will not be extended.