



**KISI - KISI SOAL
LOMBA KOMPETENSI SISWA
TINGKAT KABUPATEN PEMALANG
TAHUN 2025
BIDANG WEB TECHNOLOGIES**

CLIENT MODULE

CONTENTS

This module has the following files:

1. MODULE_CLIENT.docx
2. MODULE_CLIENT_MEDIA.zip

INTRODUCTION

You are asked to develop a game called **BlindMaze** using HTML and CSS and develop client-side programming using JavaScript. Some media files are available to you in a zip file. You can create more media and modify anything in the media if you want. Your game should be able to be played in a tablet resolution (1000 x 600 pixels). In bigger resolution, the game must be centered in the screen both horizontally and vertically.

DESCRIPTION OF PROJECTS AND TASKS

This is a **3 hours module**. You can create the layout using HTML/CSS and create the functionality of the game using JavaScript that allows the game to work correctly in different web browsers. **EcmaScript 6 / Javascript Modules** not allowed for this module. This module will be marked with directly opening the **index.html** page using **Google Chrome**.

BlindMaze game screen should have meet these requirements below:

1. Username
2. Gameboard
3. Player Lives
4. Maze
5. Timer
6. Stage number

Game Functionalities

1. Game should be on a single page application (index.html). No refresh / reload page and additional html page for any action.
2. **Show game welcome** in the center after pages are loaded.
3. **Players can Start the game** after filling the username field and click the **"Play Game"** button at the bottom of the welcome page.
4. **The "Play Game" button should be disabled** if the user did not input the username.
5. **Game instructions should be shown** when the user clicks the "Show instructions" button.
6. **Leaderboard modal should be open and show saved score descending** when the user clicks the "Leaderboard" button.
7. **Users can close the leaderboard modal** by clicking the **"X"** button.
8. Click the **"Play Game"** button to start the game.
9. **These elements should present** when the game started:
 - a. 10x10 grid
 - b. Stage number
 - c. Time left
 - d. 5 HP
10. The start point is a **single random block on the most left side column**.
11. The finish point is a **single random block on the most right side column**.
12. When the game starts, **generate random walls in the grid**. Walls are represented by **gray color blocks**.
13. You are **not allowed** to generate walls **statically**.
14. Generated walls **must not block ways** from start to finish.
15. There are two phases in a stage:
 - a. **Memorizing time**: player should **memorize the walls** and the route to finish point for 10 seconds. Player cannot move at this time.
 - b. **Move time**: after memorizing time ends, the player has 20 seconds to move to the finish point without hitting the wall.
16. **These walls are only visible in memorizing time**, then disappear when the moving time starts.

17. Players can move with **arrow keys**.
18. **If a player hits a wall**, show an alert, reduce the number of lives by 1 and restart the stage.
19. **If a player runs out of time**, show an alert, reduce the number of lives by 1 and restart the stage.
20. **If a player reaches the finish point**, show a success alert then the player will move to the next stage.
21. Player has **one chance to get a hint** by clicking the "**Hint button**".
22. When the hint button is clicked at the Move Time, the walls **become visible for one second** and then disappear.
23. Every time the user is moving to the next stage, **generate new random walls** and memorizing time starts.
24. Every new stage will generate **new random walls, random starting point, and random finish point**.
25. Game will be over **if the user loses all their lives**.
26. **A confirmation popup appears** when the game is over.
27. The game over popup should show:
 - a. Username
 - b. Stage number
 - c. Button to save score
28. **You can develop your own algorithm** to generate the walls. The generated walls should not prevent the player from moving to the finish point.
29. The game should work correctly on Google Chrome without **console error**.
30. You may see the video provided (**example.mp4**) for more details.

INSTRUCTION FOR COMPETITORS

1. Create a root folder called **XX_CLIENT_MODULE** in your local computer, where **XX** is your **computer number**.
2. Place your works inside the root folder. Make sure your works are working well when directly opening the **index.html** file.
3. Zip your root folder **XX_CLIENT_MODULE** and submit to the submission page.