SE Group 8 – Meeting 1 Summary & API Design

Group 8

April 30, 2025

1 Final Decision

Project: Maze + DFS/BFS Visualization
Frontend: HTML + CSS + JavaScript
Backend: Python (Flask maybe?)

2 Project Roadmap

2.1 Stage 1

1. Build a basic 2D map (matrix)
2. Let user edit the map (Resetable?)
3. Choose DFS to run
4. Backend simulates and returns visit history with (x, y, color)
5. Frontend renders step-by-step animation

2.2 Stage 2

1. Add BFS
2. Adjustable animation interval
3. Custom start/end points
4. Other visual effects

Color legend proposal:

- White (0): unvisited

- Black (1): obstacle
- Green (2): visiting
- Red (3): visited

3 Role Assignment (Stage 1)

3.1 Frontend (FE)

Tech: HTML, CSS, JavaScript

- Build interactive 2D maze/grid interface
- Let users place obstacles and return matrix
- Fetch map data from BE and show animation
- Animate color changes based on returned array
- Map reset function

3.2 Backend (BE)

Tech: Python (+ Flask)

- Receive map JSON from frontend
- Implement DFS to process pathfinding
- Return visited steps as array: `[{x, y, color}, ...]`

3.3 QA

Tech: Any

- Design test cases (and ideally edge cases)
- Test FE responsiveness and animation correctness
- Test BE output validity
- Check integration of FE-BE

4 API Design Proposal (for Discussion)

Request JSON:

```
{
  "map": [[0, 1, 0], [0, 0, 0], [1, 0, 0]],
  "start": [0, 0],
  "end": [2, 2],
  "algorithm": "BFS"
}
```

Response JSON:

```
{
  "steps": [
    {"x": 0, "y": 0, "color": "green"},
    {"x": 0, "y": 1, "color": "red"},
    {"x": 1, "y": 1, "color": "green"},
    {"x": 2, "y": 2, "color": "red"}
  ],
  "path": [[0, 0], [1, 1], [2, 2]],
  "status": "success"
}
```

Field Descriptions:

Field	**Type**	**Description**
`map`	2D int	0 = path, 1 = wall
`start/end`	[int, int]	Coordinates in row-column format
`algorithm`	string	”DFS” or ”BFS”
`steps`	list	Animation sequence with color transitions
`path`	list	Final path if reachable (optional)
`status`	string	”success”, ”no path”, ”invalid input” etc.

The above is a proposed format and may be discussed/modified based on implementation preferences.

Suggested Schedule

Day	**Task**
1–2	FE builds UI grid / BE sets up Flask API
3–4	BE completes BFS / FE fetches & renders
5–6	Add animations / QA tryout test cases
7+	Refactor, polish UI, add extra features, go for stage 2

Notes

As always, this is a team project so feel free to give advice and suggestions, and correct me at anytime!