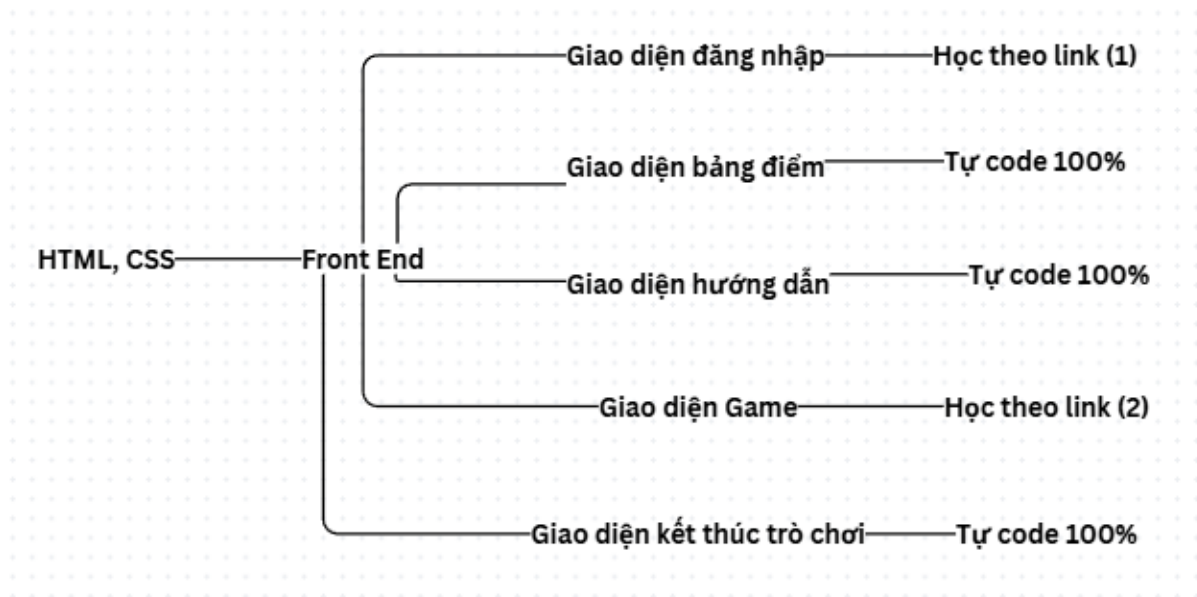


I/ Cách để chơi:

Chạy file User_login.py trong với folder là 25122063

II/ Mô tả tổng quát về Trò chơi:

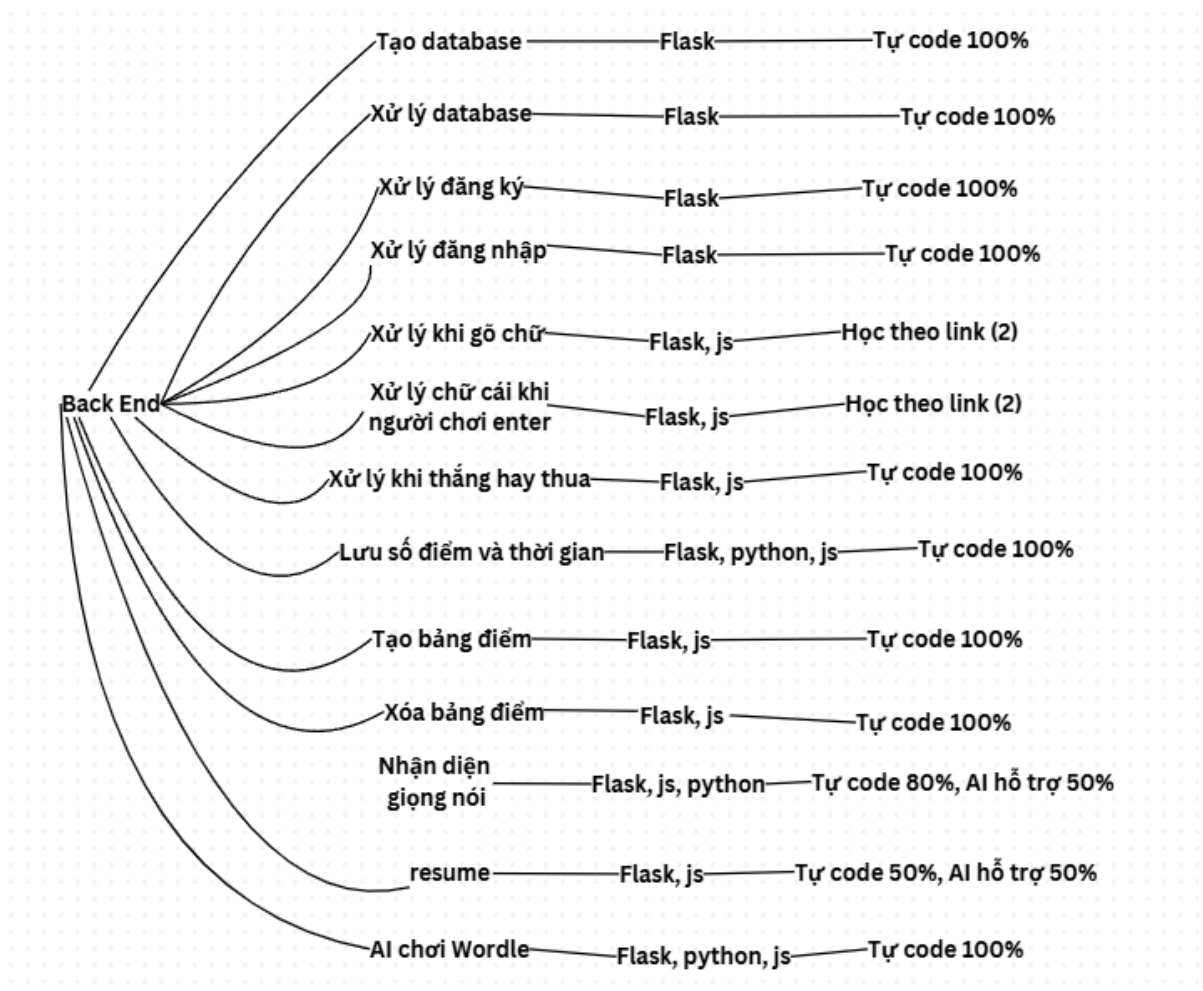
- Với Frontend, em đã thông thạo HTML, CSS nên đã có thể tạo ra trang web với giao diện theo ý muốn, nhưng để làm đẹp thì em đã xem các video (1,2).



Ảnh 1: Mô tả Frontend

- Với Backend, Em đã học thêm về Flask với video của freeCodeCamp (3). Với việc thông thạo python và javascript, em đã có thể tạo cho mình các chức năng cần thiết để chơi Wordle:
 - + Có sẵn các file txt riêng dành cho tiếng anh (4) và tiếng việt (5).
 - + Người chơi có thể đăng ký khi chưa có tài khoản, nếu người chơi đã có tài khoản mà đăng nhập sai mật khẩu thì sẽ không cho vào game. Các tài khoản sẽ được lưu trong database của flask.
 - + Có 3 chế độ chơi là easy, medium, hard cũng như chế độ chơi là tiếng việt hay tiếng anh.
 - + Thiết kế Backend của game được em tìm hiểu thêm với video của bitSEEK (2).
 - + Lưu bảng điểm bằng cách dựa vào database của flask, sắp xếp thứ tự từ trên xuống theo “point” và “time” của từng người chơi.

- Ngoài ra, người chơi có thể lựa chọn xóa bảng điểm để có thể làm mới bảng điểm. Bảng điểm không bị mất khi restart game.
- + Có resume đối với từng tài khoản người chơi, người chơi có thể lựa chọn chơi lại game cũ hoặc chơi tiếp tùy theo ý thích.
 - + Xử lý khi chơi thắng hay thua bao gồm: Quay lại trang chủ hoặc chơi tiếp để cải thiện điểm.
 - + Nhận diện giọng nói, thay vì gõ phím hay click từng chữ, người chơi có thể bấm vào mic để record và game sẽ tự động gõ chữ. Mô hình được chọn là openAI Whisper (6)
 - + Có sử dụng cấu trúc dữ liệu là list để lọc từ vựng khi xây dựng AI.
 - + Điều đặc biệt của game là khi người chơi chỉ chơi một mình, có thể nhận đối thủ của mình là một con AI (có trong giao diện game).



Ảnh 2: Mô tả Backend

- Tổ chức code dễ đọc và theo trật tự:

- + Trong Web_FrontEnd sẽ có:
 - AI_Playing: Giao diện cho AI chơi
 - Finish_AI_UI: Giao diện khi AI kết thúc trò chơi
 - Finish_UI: Giao diện khi người chơi kết thúc trò chơi
 - Game: Giao diện cho người dùng chơi
 - Instruction: Bảng hướng dẫn
 - Score_board: Bảng điểm
 - User_login: Giao diện đăng nhập, đăng ký

- + Trong Web_BackEnd sẽ có:
 - Word: Chứa các từ tiếng việt và tiếng anh cần thiết
 - Store_Micro: Lưu trữ âm thanh dạng webm khi người chơi nói
 - AI.py: Xử lý thông minh để AI chọn từ khi AI chơi
 - Database_resume.py: Cấu trúc lưu trữ khi người chơi cần resume
 - Database.py: Cấu trúc lưu trữ người chơi
 - Finish_AI_UI.py: Liên kết đến giao diện khi AI kết thúc trò chơi
 - Finish_UI.py: Liên kết đến giao diện khi người chơi kết thúc trò chơi
 - Game.py: Liên kết đến giao diện trò chơi, lưu trạng thái khi cần resume, lấy từ cần đoán, lấy các từ vựng gồm 5 chữ cái, xử lý giọng nói và lưu số điểm và thời gian của người chơi.
 - Instruction.py: Liên kết đến bảng hướng dẫn
 - Read_database.py: Hỗ trợ đọc database khi cần
 - Score_board.py: Sắp xếp thứ tự người chơi dựa theo điểm và thời gian chơi và liên kết đến bảng điểm
 - User_login.py: Đây là file chính của trò chơi , để chơi thì chạy file này. File này hỗ trợ tạo database, hỗ trợ đăng nhập, đăng ký và lựa chọn chế độ và độ khó của trò chơi.

III/ Bảng tự đánh giá:

	Mô tả chức năng	Mức độ hoàn thiện
Frontend	HTML, CSS: Hỗ trợ giao diện bao gồm:	100%

	Giao diện đăng nhập, đăng ký; Giao diện bảng điểm; Giao diện hướng dẫn; Giao diện Game; Giao diện kết thúc trò chơi	
Backend	Hỗ trợ đăng ký, đăng nhập bằng database của flask (nếu có người chơi mới thì đăng ký, nếu có người chơi cũ thì nhập đúng mật khẩu mới cho chơi)	100%
	Chọn chế độ chơi (tiếng việt hay tiếng anh) và độ khó (easy, medium, hard)	100%
	Xử lý chữ cái khi người chơi gõ và cũng như hiển thị các màu chữ khi người chơi enter	100%
	Lưu số điểm và thời gian để hiển thị trên bảng điểm. Có thể xóa bảng điểm	100%
	Có thể chơi lại game cũ nếu thoát giữa chừng đối với từng tài khoản	100%
	Xử lý khi người chơi thắng hay thua	100%
	Xử lý giọng nói (mô hình openAI whisper (6))	80%
	Hỗ trợ AI chơi cùng khi người chơi chỉ chơi một mình	100%

IV/ Sử dụng AI

Em có sử dụng AI trong việc học thêm về flask (ví dụ trong flask có redirect, return_template và url_for thường hay gây nhầm lẫn) và xin lời khuyên (ví dụ cách lọc từ để AI dự đoán từ). Ngoài ra, em còn sử dụng AI trong 2 mục đích hỗ trợ code sau:

- Hỗ trợ code trong viết code resume trò chơi ở javascript (dòng 342 ở Game_script.js)
- Hỗ trợ code trong viết code cho phép sử dụng mic ở browser và trả giọng nói về [Game.py](#) để nhận dạng giọng nói (ở Game_mic.js)

Ngoài các AI hỗ trợ đã nêu ở trên, em cam kết không sử dụng AI để copy & paste code hay bảo AI code giúp trong các phần khác.

V/ Một số lưu ý khi sử dụng code

- Code chỉ chạy hoàn hảo khi thư mục được mở là 25122063.
- Phiên bản em nộp trên moodle là phiên bản không có mic, tại vì phiên bản này cần phải tải các thư viện là openai-whisper và ffmpeg (cách tải ở (6)). Nếu cần phiên bản có mic, em đã có để code trên github (7).
- Whisper của openAI hoạt động rất tốt khi nhận diện tiếng anh, nhưng tiếng việt thì hơi tệ nên em đánh giá mức độ hoàn thành là 80%. Có model là PhoWhisper nhưng bị lỗi token, API hugging face nên em không truy cập được.

V/ Link:

1. https://www.youtube.com/watch?v=so_FzCUMlf0
2. <https://www.youtube.com/watch?v=M-uKfV8a1C8>
3. https://www.youtube.com/watch?v=Z1RJmh_OqeA&t=1221s
4. https://github.com/Hugo0/wordle/blob/main/webapp/data/languages/en/en_5words.txt
5. https://github.com/Hugo0/wordle/blob/main/webapp/data/languages/vi/vi_5words.txt
6. <https://github.com/openai/whisper>
7. <https://github.com/NguyenThanhCong170/-n-HCMUS/tree/main/%C4%90%E1%BB%93%20%C3%A1n%20n%C4%83m%20nh%E1%BA%A5t%20-%20Game%20Wordle%20-%20c%C3%B3%20mic>