**NỘI DUNG GAME SUDOKUONL**

**Nói về Định nghĩa game Sudoku(Lịch sử, luật chơi,…), Python, CodeSandBox, …**

**1. Mục Tiêu Dự Án**

* Mục tiêu chính: Phát triển một trò chơi Sudoku với hai chế độ chơi:
  + Offline: Người chơi giải bảng Sudoku một mình với các mức độ khó (dễ, trung bình, khó).
  + Online: Hai người chơi cùng giải một bảng Sudoku, cạnh tranh để hoàn thành nhanh nhất.
* Tính năng chính:
  + Giao diện thân thiện, hỗ trợ responsive trên nhiều thiết bị.
  + Cơ chế kiểm tra lỗi và highlight ô liên quan.
  + Đếm thời gian và theo dõi số lần nhập sai (tối đa 3 lần sai dẫn đến thua cuộc).
  + Hỗ trợ hoàn tác (undo) và xóa ô trong chế độ offline.
  + Trong chế độ online, hệ thống tự động ghép đôi người chơi và thông báo kết quả.

**2. Cấu Trúc Dự Án**

**Dự án được chia thành các file chính, mỗi file đảm nhiệm một vai trò cụ thể:**

**2.1. Backend (Python - Flask)**

* File: main.py
* Chức năng:
  + Sử dụng Flask và Flask-SocketIO để xử lý các kết nối thời gian thực (real-time) cho chế độ online.
  + Tạo và quản lý phòng chơi (rooms) cho chế độ online, hỗ trợ ghép đôi người chơi.
  + Sinh bảng Sudoku với 3 mức độ khó (dễ: 40 ô trống, trung bình: 50 ô trống, khó: 60 ô trống).
  + API /generate\_sudoku cung cấp bảng Sudoku và lời giải cho chế độ offline.
  + Xử lý các sự kiện SocketIO: join\_game, update\_board, request\_new\_game, disconnect.
  + Kiểm tra điều kiện thắng và thông báo kết quả trong chế độ online.
* Thuật toán:
  + Hàm solve\_sudoku: Sử dụng backtracking để giải bảng Sudoku.
  + Hàm generate\_sudoku: Tạo bảng Sudoku bằng cách xóa ngẫu nhiên các ô từ bảng hoàn chỉnh.

**2.2. Frontend (HTML, JavaScript, CSS)**

* Files HTML:
  + index.html: Trang chính, cung cấp tùy chọn chơi offline hoặc online.
  + practice.html: Giao diện cho chế độ offline.
  + online.html: Giao diện cho chế độ online.
* Files JavaScript:
  + script.js (Offline):
    - Khởi tạo bảng Sudoku, xử lý nhập liệu người chơi, kiểm tra lỗi, và hoàn tác.
    - Gọi API /generate\_sudoku để lấy bảng và lời giải.
    - Hiển thị thời gian, mức độ khó, và thông báo kết quả qua modal.
  + online.js (Online):
    - Kết nối với server qua SocketIO để ghép đôi người chơi và cập nhật bảng.
    - Xử lý các sự kiện như match\_found, board\_updated, game\_over, opponent\_disconnected.
    - Hiển thị thông báo chờ, bảng chơi, và kết quả.
* Files CSS:
  + style.css: Định dạng giao diện cho cả hai chế độ, bao gồm bảng Sudoku, menu, modal, và hiệu ứng highlight.
  + static.css: CSS bổ sung (có một số khác biệt nhỏ so với style.css, cần đồng bộ hóa để tránh trùng lặp).

**2.3. Giao Diện Người Dùng**

* Thành phần chính:
  + Header: Hiển thị tiêu đề, mức độ khó, số sao (★), và bộ đếm thời gian.
  + Menu độ khó: Nút chọn mức độ (Dễ, Trung bình, Khó).
  + Bảng Sudoku: Bảng 9x9 với các ô nhập liệu, ô cố định (disabled), và hiệu ứng highlight.
  + Toolbar: Nút xóa ô (eraser) và hoàn tác (undo).
  + Modal: Hiển thị kết quả (thắng/thua) và lý do (ví dụ: quá 3 lần sai, đối thủ ngắt kết nối).
  + Thông báo chờ: Hiển thị khi chờ ghép đôi trong chế độ online.
* Responsive Design:
  + Sử dụng rem cho kích thước, đảm bảo giao diện phù hợp trên cả desktop và mobile.
  + Media query trong style.css điều chỉnh kích thước ô và font cho màn hình nhỏ (<600px).

**3. Công Nghệ Sử Dụng**

* Backend:
  + Python: Ngôn ngữ chính cho server.
  + Flask: Framework web để xử lý các route và API.
  + Flask-SocketIO: Hỗ trợ giao tiếp thời gian thực cho chế độ online.
  + Eventlet: Xử lý các kết nối bất đồng bộ.
* Frontend:
  + HTML5/CSS3: Xây dựng giao diện và responsive design.
  + JavaScript: Xử lý logic client-side và tương tác với server.
  + Socket.IO: Kết nối thời gian thực với server.
  + Font Awesome: Cung cấp biểu tượng (ví dụ: nút xóa).
* Môi trường phát triển:
  + CodeSandbox: Nền tảng để chạy và kiểm thử toàn bộ ứng dụng.

**4. Các Chức Năng Nổi Bật**

* Chế độ Offline:
  + Người chơi tự do chọn mức độ khó và giải bảng Sudoku.
  + Hỗ trợ hoàn tác (undo) thông qua mảng history.
  + Kiểm tra điều kiện thắng và thông báo thời gian hoàn thành.
* Chế độ Online:
  + Tự động ghép đôi người chơi vào cùng một phòng.
  + Hai người chơi cạnh tranh trên cùng một bảng, điểm số dựa trên số ô điền đúng.
  + Hỗ trợ xử lý ngắt kết nối (opponent\_disconnected) và thông báo người thắng.
* Tính năng hỗ trợ:
  + Highlight hàng, cột, và các ô có cùng giá trị khi chọn một ô.
  + Hiệu ứng lỗi (màu đỏ) khi nhập sai, xóa lỗi khi sửa đúng.
  + Bộ đếm thời gian định dạng MM:SS.
  + Modal thông báo kết quả rõ ràng, dễ hiểu.

**5. Các Vấn Đề và Đề Xuất Cải Thiện**

* Vấn đề hiện tại:
  + File static.css và style.css có nội dung tương tự, dễ gây nhầm lẫn. Cần hợp nhất thành một file duy nhất.
  + Chức năng undo trong chế độ online chưa được triển khai (hiển thị alert "not implemented").
  + Không có cơ chế lưu trạng thái trò chơi khi người chơi thoát và quay lại.
  + Thiếu tính năng lưu điểm số cao (high score) hoặc lịch sử chơi.
* Đề xuất cải thiện:
  + Triển khai chức năng undo cho chế độ online bằng cách lưu lịch sử bước đi trên server.
  + Thêm cơ chế lưu trữ phiên chơi (session) để hỗ trợ reconnect.
  + Thêm bảng xếp hạng hoặc lưu điểm số cao sử dụng localStorage hoặc cơ sở dữ liệu.
  + Tối ưu hóa giao diện cho các màn hình lớn hơn (ví dụ: tablet).
  + Thêm hiệu ứng âm thanh hoặc hình ảnh động để tăng trải nghiệm người dùng.

**Cấu Trúc Thư Mục và Chức Năng Các File trong Dự Án Sudoku**

**1. Cấu Trúc Thư Mục**

/ (Thư mục gốc)

├── .codesandbox/ # Cấu hình CodeSandbox (tự động tạo)

├── .devcontainer/ # Cấu hình môi trường phát triển (tự động tạo)

├── node\_modules/ # Thư mục chứa các thư viện Node.js (tự động tạo)

├── static/ # Thư mục chứa các file tĩnh (CSS, JS)

│ ├── css/ # Thư mục chứa các file CSS

│ │ ├── style.css # CSS chính cho giao diện (dùng chung cho cả offline và online)

│ │ └── static.css # CSS bổ sung (ẩn, có nội dung tương tự style.css, cần hợp nhất)

│ ├── js/ # Thư mục chứa các file JavaScript

│ │ ├── online.js # JS xử lý logic chế độ online (SocketIO, cập nhật bảng, ghép đôi)

│ │ └── script.js # JS xử lý logic chế độ offline (khởi tạo bảng, hoàn tác, kiểm tra lỗi)

│ └── templates/ # Thư mục chứa các file HTML

│ ├── index.html # Trang chính, chọn chế độ chơi (offline hoặc online)

│ ├── online.html # Giao diện chế độ online, kết nối với online.js

│ └── practice.html # Giao diện chế độ offline, kết nối với script.js

├── main.py # File backend chính (Flask, SocketIO, sinh bảng Sudoku)

├── README.md # File mô tả dự án (thường chứa hướng dẫn cài đặt và mô tả)

├── requirements.txt # File liệt kê các thư viện Python cần thiết (Flask, Flask-SocketIO, eventlet)

└── yarn.lock # File quản lý phiên bản thư viện (tự động tạo bởi Yarn)

**2. Chức Năng Từng File**

**2.1. Backend**

* **main.py**:
  + **Chức năng**: File chính cho backend, sử dụng Flask và Flask-SocketIO.
  + Xử lý các route (/, /practice, /online, /generate\_sudoku) để render giao diện và cung cấp API.
  + Quản lý kết nối thời gian thực qua SocketIO cho chế độ online (ghép đôi, cập nhật bảng, kiểm tra thắng thua).
  + Sinh bảng Sudoku với các mức độ khó (dễ, trung bình, khó) bằng thuật toán backtracking.
  + Quản lý phòng chơi và điểm số người chơi trong chế độ online.

**2.2. Frontend - Giao Diện (HTML)**

* **templates/index.html**:
  + **Chức năng**: Trang chính của ứng dụng.
  + Hiển thị tiêu đề "Sudoku" và hai nút để chọn chế độ chơi: "Chơi Offline" (dẫn đến /practice) và "Chơi Online" (dẫn đến /online).
* **templates/online.html**:
  + **Chức năng**: Giao diện cho chế độ online.
  + Kết nối với online.js để xử lý ghép đôi người chơi và cập nhật bảng.
  + Hiển thị bảng Sudoku, menu chọn độ khó, bộ đếm thời gian, và modal kết quả.
  + Có thông báo chờ khi ghép đôi ("Đang chờ người chơi khác...").
* **templates/practice.html**:
  + **Chức năng**: Giao diện cho chế độ offline.
  + Kết nối với script.js để khởi tạo và quản lý bảng Sudoku.
  + Hiển thị bảng Sudoku, menu chọn độ khó, bộ đếm thời gian, toolbar (xóa, hoàn tác), và modal kết quả.

**2.3. Frontend - Logic (JavaScript)**

* **static/js/online.js**:
  + **Chức năng**: Xử lý logic cho chế độ online.
  + Kết nối với server qua SocketIO, xử lý các sự kiện như match\_found, board\_updated, game\_over, opponent\_disconnected.
  + Cập nhật bảng Sudoku theo thời gian thực dựa trên dữ liệu từ server.
  + Quản lý giao diện (hiển thị/ẩn menu độ khó, thông báo chờ, bảng chơi).
* **static/js/script.js**:
  + **Chức năng**: Xử lý logic cho chế độ offline.
  + Gọi API /generate\_sudoku để lấy bảng và lời giải.
  + Xử lý nhập liệu người chơi, kiểm tra lỗi (so sánh với lời giải), và hỗ trợ hoàn tác (undo) thông qua mảng history.
  + Quản lý bộ đếm thời gian, hiển thị modal kết quả khi thắng hoặc thua.

**2.4. Frontend - Giao Diện (CSS)**

* **static/css/style.css**:
  + **Chức năng**: Định dạng giao diện chính cho cả hai chế độ (offline và online).
  + Bao gồm kiểu dáng cho bảng Sudoku, menu độ khó, modal, toolbar, và hiệu ứng highlight (hàng, cột, ô cùng giá trị).
  + Hỗ trợ responsive design với media query cho màn hình nhỏ (<600px).
* **static/css/static.css**:
  + **Chức năng**: CSS bổ sung, nhưng có nội dung tương tự style.css.
  + **Lưu ý**: File này bị ẩn trong CodeSandbox (đánh dấu bằng #), có thể không được sử dụng. Nên hợp nhất với style.css để tránh trùng lặp.

**2.5. Cấu Hình và Tài Liệu**

* **requirements.txt**:
  + **Chức năng**: Liệt kê các thư viện Python cần thiết cho backend (flask, flask-socketio, eventlet).
  + Đảm bảo CodeSandbox cài đặt đúng các thư viện này.
* **README.md**:
  + **Chức năng**: Mô tả dự án, hướng dẫn cài đặt, và cách chạy ứng dụng.
  + Thường bao gồm thông tin về cách khởi động server và truy cập trò chơi.
* **yarn.lock**:
  + **Chức năng**: Quản lý phiên bản các thư viện JavaScript (tự động tạo bởi Yarn).
  + Đảm bảo tính nhất quán khi cài đặt các thư viện như SocketIO client.

**2.6. Thư Mục Tự Động Tạo**

* **.codesandbox/**, **.devcontainer/**, **node\_modules/**:
  + **Chức năng**: Các thư mục này được CodeSandbox tự động tạo để cấu hình môi trường phát triển và quản lý thư viện.
  + Không cần chỉnh sửa, nhưng cần đảm bảo node\_modules chứa các thư viện như socket.io-client.

**3. Lưu Ý**

* File static.css và style.css có nội dung tương tự, nên hợp nhất để tránh trùng lặp.
* Đảm bảo CodeSandbox hỗ trợ kết nối SocketIO (có thể cần cấu hình CORS trong main.py).
* File README.md nên được cập nhật để bao gồm hướng dẫn chi tiết cho người dùng cuối.

(Đoạn này tìm hiểu rồi viết về file Style.css thôi nha)

**Khi chạy thử trò chơi:**

* + 1. Chạy lệnh python ./main.py

A screenshot of a computer

AI-generated content may be incorrect.

* + 1. Sang phần port để mở đường link

A screenshot of a computer

AI-generated content may be incorrect.

* + 1. Hiện ra giao diện màn hình để chọn chế độ chơi

A screenshot of a computer

AI-generated content may be incorrect.

* + 1. Chế độ chơi offline:

Chọn cấp độ khó

A screenshot of a computer

AI-generated content may be incorrect.

Khi vào chơi thì đồng hồ bắt đầu bấm giờ

A screenshot of a computer

AI-generated content may be incorrect.

Nếu người chơi nhập sai thì ô sẽ hiện màu đỏ:

A screenshot of a computer

AI-generated content may be incorrect.

Nếu nhập sai quá 3 lần thì sẽ kết thúc trò chơi

A screenshot of a computer

AI-generated content may be incorrect.

* + 1. Chế độ chơi online:

Người chơi thứ nhất chọn chế độ rồi gửi link sang cho người chơi thứ 2

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

Khi người thứ 2 bấm vào thì cả hai sẽ bắt đầu chơi, đồng hồ bắt đầu bấm giờ

A screenshot of a computer

AI-generated content may be incorrect.