|  |
| --- |
| detail of persons hands with scissors, markers, workingGiới Thiệu  Sản Phẩm Caro |

|  |  |  |
| --- | --- | --- |
| Nhóm: Alpha |  |  |
|  |  |  |

Mục Lục

[Giới thiệu 3](#_Toc501116467)

[Thông tin nhóm 3](#_Toc501116468)

[Tổng quan vấn đề 3](#_Toc501116469)

Chi tiết về sản phẩm và quá trình làm\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 5

[Yêu cầu 1 5](#_Toc501116475)

[Yêu cầu 2 6](#_Toc501116475)

Yêu cầu 3  [7](#_Toc501116475)

Yêu cầu 4  [7](#_Toc501116475)

# Giới thiệu:

## Thông tin nhóm:

* Cơ sở: Trường Đại học FPT Cần Thơ. Lưu trữ
* Tên nhóm: Aplha.
* Số lượng thành viên 3.

## Tổng quan về sản phẩm:

• Tên sản phẩm: Phần mềm chơi caro.

Chức năng chính:

+ Tạo ván đấu 2 người, có thể chơi qua 1 hay 2 thiết bị

+ Tạo giải đấu

Chức năng lưu trữ:

+ Lưu trạng thái ván đấu trên blockchain

+ Lưu điểm ELO của từng cá nhân

Chức năng giao dịch tiền thưởng:

+ Giao dịch qua hệ thống tiền điện tử

A close up of a map

Description automatically generated

*Use case diagram*

# Chi tiết CÁCH LÀM

* 1. **Yêu cầu 1:**

► Front-end:

• Ngôn ngữ lập trình giao diện: Javascript + React

* 1. **Yêu cầu 2:**

Công cụ:

Truffle

Các bước thực hiện:

* + 1. Viết Smart Contract: gồm hàm lưu, load game
    2. Phía Front-End gọi tới hàm tương ứng trong Smart Contract khi người dùng nhấn save/load

+ Người dùng ấn save: gọi hàm saveGame() và truyền

vào vị trí các nước đã đi cùng với trạng thái hiện tại của bàn chơi. Chương trình thông báo ID của bàn vừa được lưu.

+ Người dung ấn load: gọi hàm loadGame() và truyền vào ID của bàn chơi muốn load. Chương trình trả về bàn chơi đó.

* 1. **Yêu cầu 3:**

Bước thực hiện:

1. Viết Smart Contract để lưu thông tin người dùng. Trong Smart Contract có hàm cập nhật chỉ số ELO người dùng.
2. Khi kết thúc một bàn chơi, chương trình sẽ dùng Javascript tính lại chỉ số ELO người dùng. Sau đó, chương trình sẽ gọi tới hàm cập nhật chỉ số ELO với tham số truyền là ID người dùng và chỉ số ELO mới.

**Cách tính ELO:**

*! Những người lần đầu chơi sẽ có điểm ELO là 1200*

|  |
| --- |
| Công thức chung: Rn = Ro + \* (S - ) |

Trong đó:

+ Rn: Số ELO mới sau trận của người chơi cần tính

+ Ro: Số ELO trước khi chơi trận người chơi cần tính

+ S: Kết quả thật của trận đấu của người chơi cần tính

- Thắng : S=1

- Hòa: S=0.5

- Thua: S=0

+ E: kết quả dự đoán của người chơi cần tính

E =

n: nếu A giỏi hơn B gấp 10 lần => Ra-Rb=n

\* n ở đây được quy định là 400

+ K: hệ số K của người chơi cần tính

= 25 dành cho kỳ thủ mới có cường số dưới 1600

= 20 dành cho kỳ thủ mới có cường số dưới 2000

= 15 dành cho kỳ thủ có cường số dưới 2400.

= 10 dành cho kỳ thủ có cường số trên 2400

1. **Cách dùng chương trình**
2. Clone project về
3. Cài đặt truffle và các dependencies cần thiết:

Danh sách các dependencies:

truffle

babel-register

babel-polyfill

babel-preset-es2015

babel-preset-stage-2

babel-preset-stage-3

chai

chai-as-promised

react-scripts

react

react-dom

bootstrap, jquery, popper

[web3@1.0.0-beta.37](mailto:web3@1.0.0-beta.37)

1. chạy lệnh npm run start để chạy chương trình

1. **Reference:**

Cách tính ELO:

<https://vi.wikipedia.org/wiki/H%E1%BB%87_s%E1%BB%91_Elo>

<https://blog.mackie.io/the-elo-algorithm>

Game Caro:

<https://github.com/vietanhdev/van-caro-online>