Contents

[1. SFML là gì? 1](#_Toc32373354)

[2. Download và Cài Đặt 1](#_Toc32373355)

[a. Download 1](#_Toc32373356)

[b. Cài đặt 1](#_Toc32373357)

[c. Chú ý: 5](#_Toc32373358)

[3. Hello World với SFML 5](#_Toc32373359)

[a. Demo Code 5](#_Toc32373360)

[b. Giải thích Code Demo 6](#_Toc32373361)

[4. Cách thiết kế và cài đặt bàn cờ chơi 2 người 7](#_Toc32373362)

[a. Ý tưởng 7](#_Toc32373363)

[b. Thực hiện 7](#_Toc32373364)

[5. Demo Code 8](#_Toc32373365)

Tổng quan về SFML

1. SFML là gì?

\_ SFML là viết tắt của Simple and Fast Multimedia Library.

\_ SFML là một thư viện đa phương tiện viết từ C++, nó gần giống với SDL nhưng được viết theo dạng OOP.

\_ SFML gồm 5 modules là: Audio, Graphics, Network, System, Window.

+ Graphics: Bao gôm các class về xử lý đồ họa

+ Audio: Bao gồm các class về xử lý âm thanh.

+ Window: Liên quan tới việc tạo đóng và xử lý các sự kiện cửa sổ.

+ System: Gồm các class liên quan với hệ thống như làm thời gian, xử lý Unicode.

+ Network: Gồm các class về mạng.

1. Download và Cài Đặt
   1. Download

\_ Để có thể sử dụng SFML ta cần tải trình biên dịch C++ và SFML về máy.

\_ Trình biên dịch em dùng là MinGW.

\_ Link download MinGW: <https://sourceforge.net/projects/mingw-w64/>

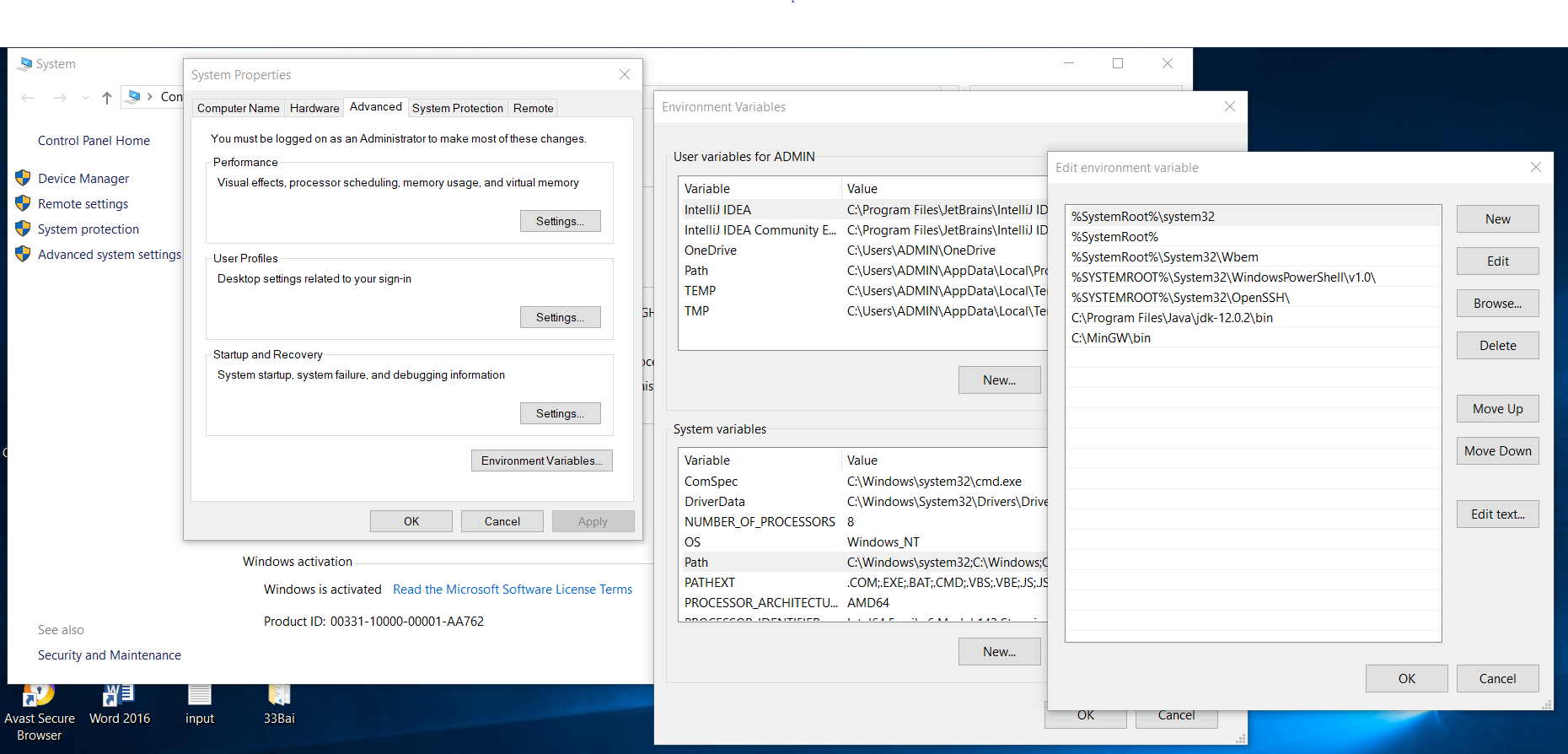
\_ Link download SFML: <https://www.sfml-dev.org/download/sfml/2.5.1/>

* 1. Cài đặt

\_ IDE em sử dụng ở đây là CodeBlock.

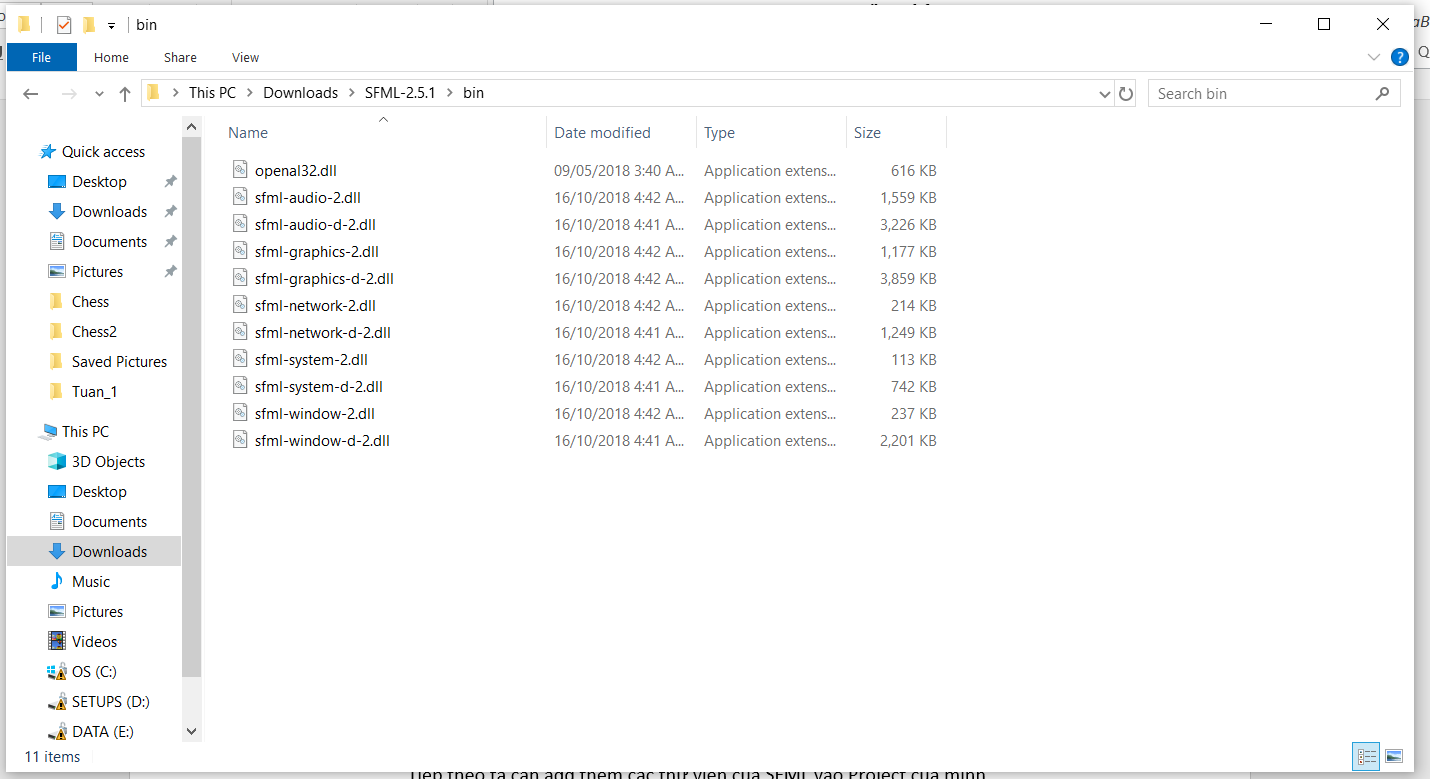
\_ Trước hết ta cần cài đặt MinGW và cài nó vào biến môi trường:

Click chuột phải vào This PC 🡪 Chọn Properties 🡪 Chọn Advanced System Settings 🡪 Chọn Enviroment Variables 🡪 Chọn Path 🡪 Cuối cùng chọn Browse để thêm MinGW vào biến môi trường.



\_ Tiếp theo ta cần add thêm các thư viện của SFML vào Project của mình.

\_ Copy tất cả các file .dll từ folder bin của SFML sau khi giải nén file đã tải về vào Project của mình.



\_ Vào lại CodeBlock chọn Project 🡪 Chọn vào phân Build Options 🡪 Chọn Debug hoặc

Release.

\_ Tiếp theo ta cần add các thư viện mới này CodeBlock:

+ Chọn Complier Settings vào phần Complier Flags chọn trình biên dịch mà bạn muốn

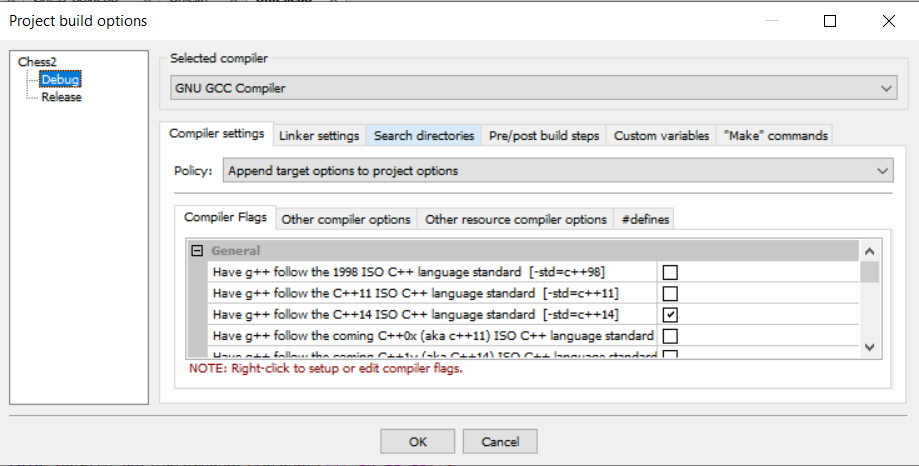
+ Chọn linker settings và add các thư viện mới vào: Chọn add sau đó gõ tên các thư viện

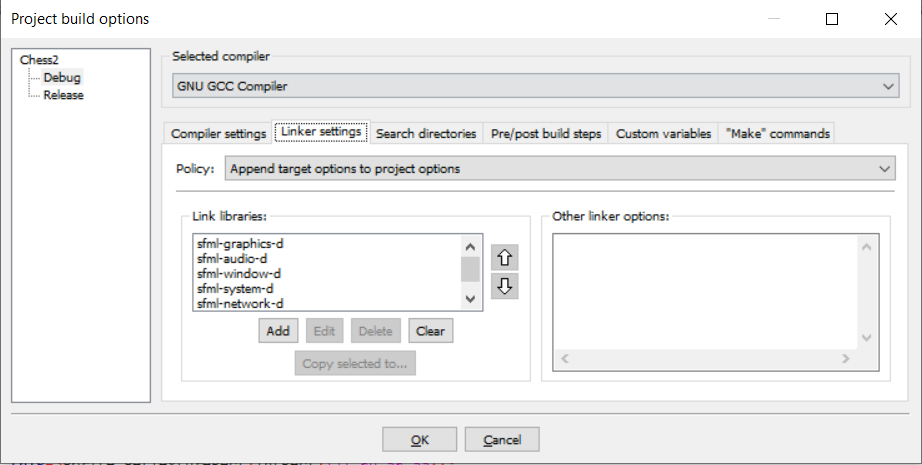
muốn add vào: sfml-graphics-d, sfml-audio-d, sfml-window-d, sfml-system-d,

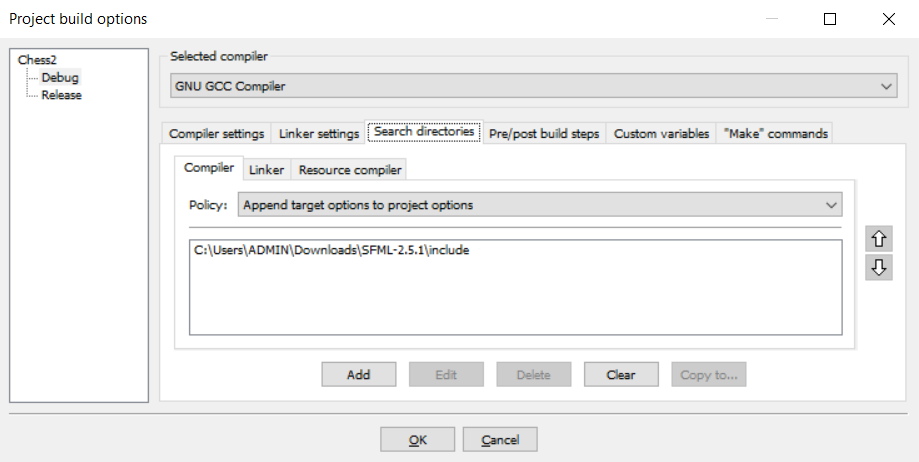
sfml-network-d, với Release thì không cần thêm –d vào đằng sau.

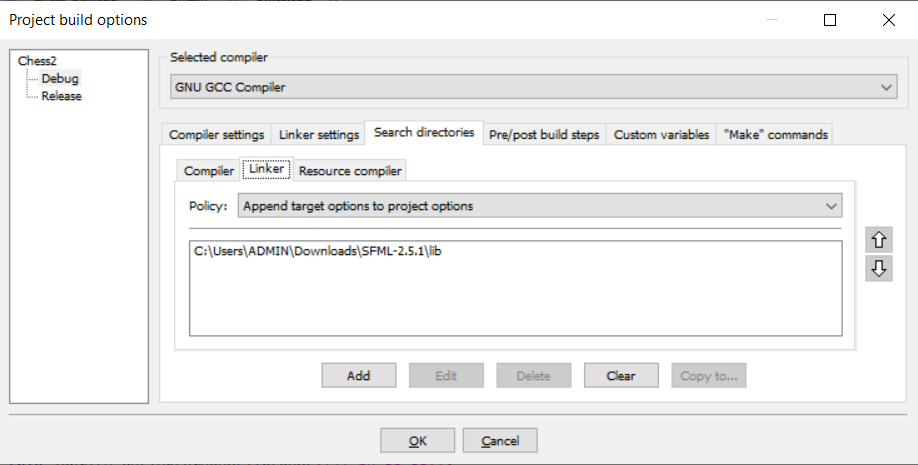
+ Chọn Search Directories vào phần Complier sau đó add folder include của SFML vào

+ Ở phần Search Directories tiếp tục chọn Linker sau đó add folder lib của SFML vào







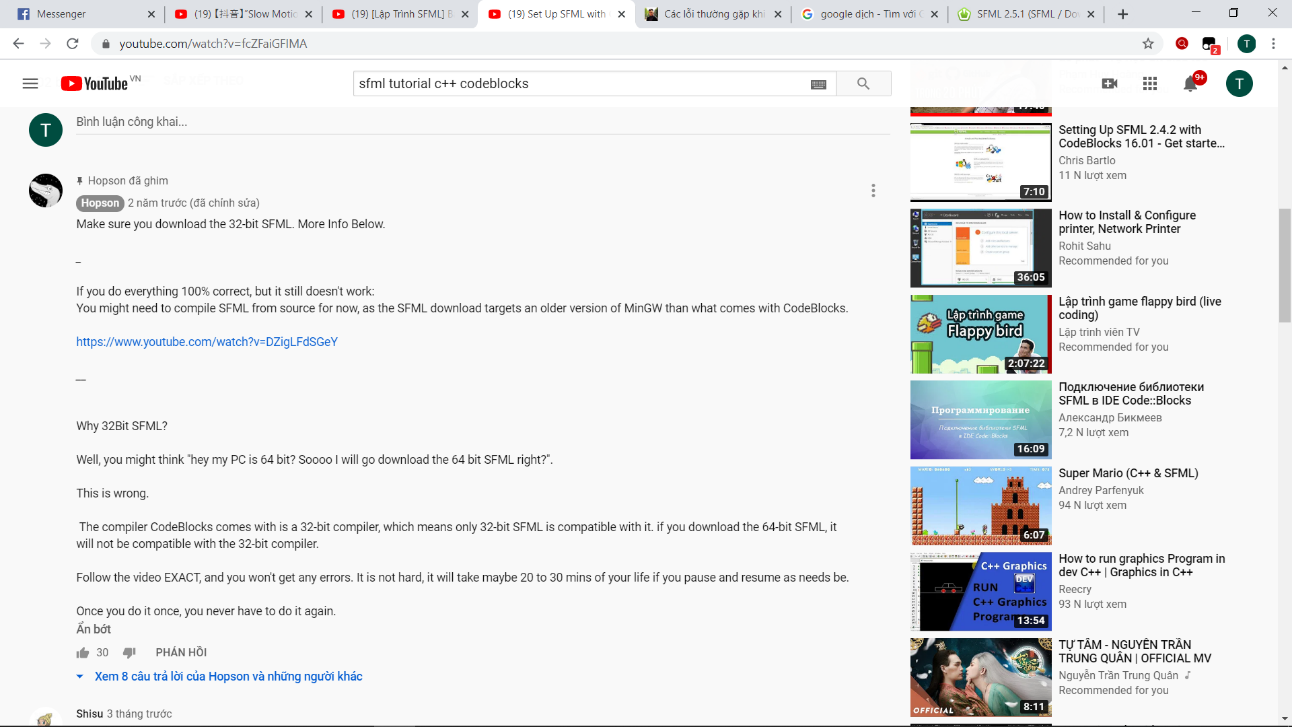


\_ Cuối cùng chọn File 🡪 Chọn Save Project as Template để những lần sau khi tạo Project sẽ không phải làm lại những thao tác trên.

* 1. Chú ý:

\_ Nếu làm đúng như bên trên rất có thể chương trình của bạn sẽ không chạy. Sẽ không có cửa sổ window xuất hiện và trình biên dịch sẽ bào lỗi.

\_ Sau 1 khoảng thời gian tìm hiểu em đã phát hiện ra lý do tại sao trình biên dịch lại báo lỗi



\_ Đa phần các máy hiện giờ đều là 64 bit nên khả năng rất cao là sẽ tải phiên bản SFML 64 bit. Điều này không có gì là sai nhưng do chúng ta đang sử dụng CodeBlock và nó chỉ hỗ trợ phiên bản 32 bit.

1. Hello World với SFML
   1. Demo Code

#include <SFML/Graphics.hpp>

#include <bits/stdc++.h>

using namespace std;

using namespace sf;

int main()

{

RenderWindow window(VideoMode(500, 500),"Hello World", Style::Default);

CircleShape circle;

circle.setFillColor(Color::Blue);

circle.setRadius(250);

while(window.isOpen())

{

Event event;

while(window.pollEvent(event))

{

if(event.type == Event::Closed) window.close();

}

window.clear();

window.draw(circle);

window.display();

}

return 0;

}

\_ Khi chạy đoạn code này chúng ta sẽ thấy có một cửa sổ window hiện lên có tiêu đề Hello World và 1 vòng tròn màu xanh hiện lên ở chính giữa cửa sổ.

* 1. Giải thích Code Demo

\_ 2 dòng code đầu tiên dùng để khai báo thư viện đồ họa của SFML và thư viện tiêu chuẩn của STL

\_ 2 dòng code đầu tiên để chúng ta không cần sử dụng std:: và sf::

\_ Tiếp theo chúng ta tạo ra 1 cửa sổ window có tiêu đề là Hello World kích cỡ là 500x500 và có kiểu là mặc định.

\_ Tiếp theo ta tạo ra một hình tròn và Set bán kính cũng như đổ màu cho nó bằng 2 phương thức setRadius và setFillColor.

\_ 1 vòng lặp với điều kiện chạy là cửa sổ window vẫn còn được mở

\_ Trong vòng lặp ta tạo ra một sự kiện, sự kiện này sẽ nhận bất cứ thao tác nào tác động lên cửa sổ window của chúng ta.

\_ Ta tiếp tục có 1 vòng while bên trong dùng để xử lý các thao tác tác động lên cửa sổ window.

\_ Khi người dùng đóng cửa sổ type của event sẽ là Closed khi đó lệnh window.close() sẽ được thực hiện và đóng cửa sổ window lại.

\_ window.clear() dùng để làm sạch cửa sổ window sau mỗi vòng lặp.

\_ window.draw() dùng để biểu diễn vật thể lên cửa sổ window.

\_ window.display() dùng để hiện thị lên màn hình.

1. Cách thiết kế và cài đặt bàn cờ chơi 2 người
   1. Ý tưởng

\_ Rất đơn tạo ra một bàn cờ và các quân cờ nhận lệnh điều khiển từ người dùng và di chuyển các quân cờ trên bàn cờ

* 1. Thực hiện

\_ Đầu tiên em cần chuẩn bị các hình ảnh về bàn cờ cũng như các quân cờ để có thể hiển thị trên cửa sổ window.

\_ Em tạo ra một Class Chess\_Box để làm 1 ô trên bàn cờ. Mỗi ô trên bàn cờ ta sẽ mặc định nó sẽ có một đơn vị chiếm đóng tại đó.

\_ Em tiếp tục tạo ra một Class Units bao gồm các quân cờ, mỗi quân cờ sẽ được load từ một file hình ảnh chứa hình ảnh của nó.

\_ Em tạo ra một bàn cờ bằng một vector 2 chiều với mỗi phần tử là một Chess\_Box và có trạng thái giống như trạng thái ban đầu của một bàn cờ vua.

\_ Em sử dụng vòng while để mở cửa sổ và biến event để xử lý các thao tác của người dùng.

\_ Khi người chơi chọn một quân cờ tức là họ là click vào một ô trên cờ em sử dụng Mouse::isButtonPressed(Mouse::Left) và Mouse::getPosition(window) để kiểm tra xem người đó đã click chuột hay không và lấy vị trí con chuột để biết được họ đã chọn vào ô nào trên bàn cờ sau đó khi người chơi di chuyển em liên tục cập nhật lại vị trí của quân cờ mà người chơi di chuyển trùng với vị trí con chuột để tạo ra hiệu ứng di chuyển của quân cờ.

\_ Khi người chơi tiếp tục Click chuột lần 2 tức là người chơi muốn di chuyển quân cờ đến vị trí đó em sẽ xét xem nước đi có hợp lệ hay không nếu không hợp lệ quân cờ sẽ trở lại vị trí ban đầu và người chơi tiếp tục lượt đi của mình.

\_ Nếu nước đi là hợp lệ thì em sẽ update lại đơn vị hiện tại của 2 ô mà người chơi đã chọn.

\_ Em sử dụng một biến Playing để xác định đang đến lượt chơi của bên nào nếu bên trắng đang chơi thì sẽ không thể chọn và di chuyển được các quân cờ ở bên đỏ và ngược lại.

\_ Khi 1 trong 2 quân vua bị mất trò chơi sẽ kết thúc và cửa sổ sẽ đóng lại ngay lập tức.

1. Demo Code

\_ Link GitHub: <https://github.com/LeThanhLong582000/Chess>