



Lập trình Python

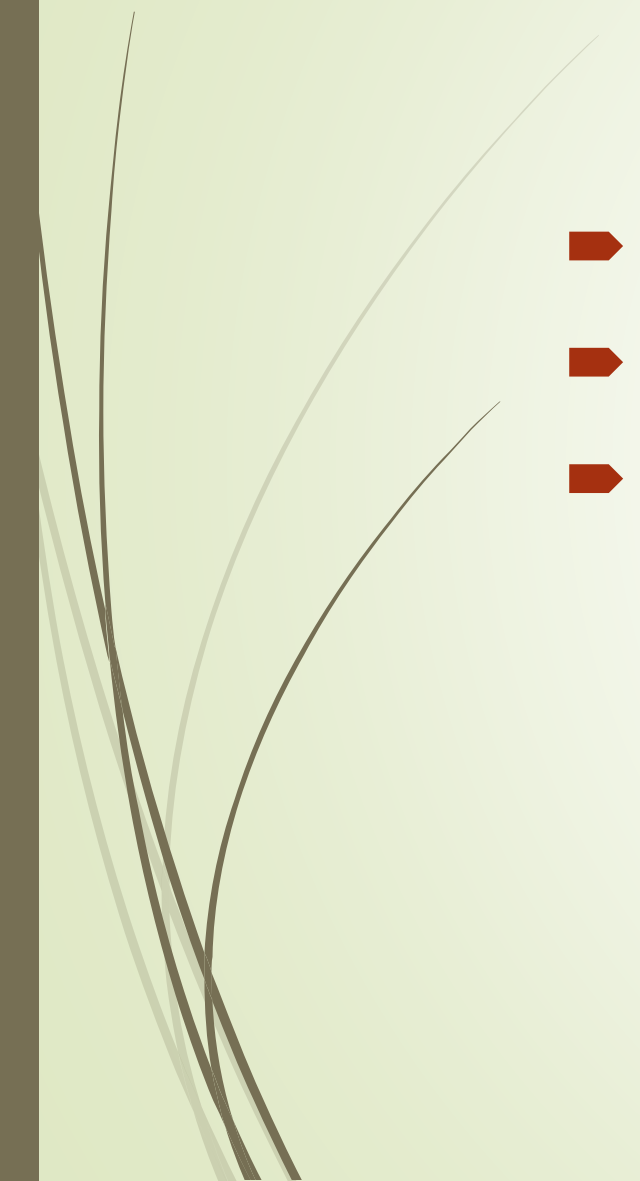
Đề tài : Tic-Tac-Toe GUI với OOP

Sinh viên thực hiện: Lăng Nguyễn Minh Lượng

GVHD: TS Nguyễn Văn Huy



Giới thiệu đề tài

- Xây dựng trò chơi caro 3x3 với Python
 - Giao diện Tkinter + âm thanh Pygame
 - Hỗ trợ chơi với người hoặc máy (AI)
- 

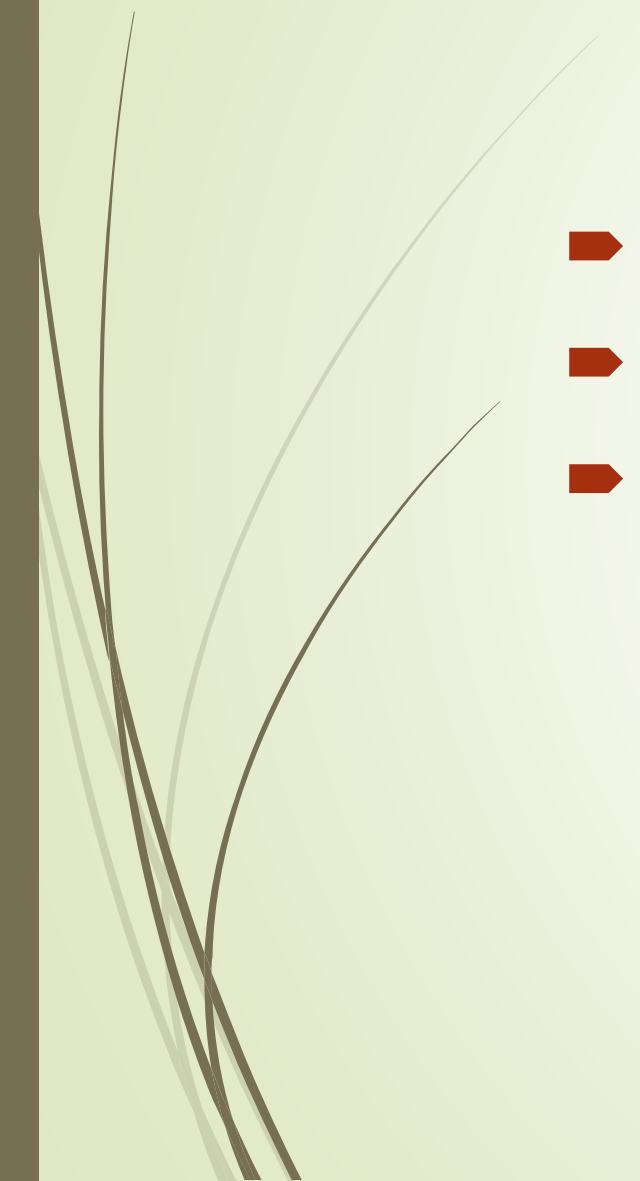


Cách người dùng tương tác

- Click vào ô để đánh dấu X/O
- Tự kiểm tra thắng/hòa sau mỗi lượt
- Nút Reset để chơi lại




Cấu trúc hệ thống

- TTTBoard: xử lý logic trò chơi
 - TicTacToeGUI: tạo và xử lý giao diện
 - Pygame: phát âm thanh hiệu ứng
- 



Các tính năng chính đã làm

- Theo dõi lượt chơi (turn)
- Kiểm tra thắng/hòa
- Chặn nước đi sai
- Reset và chọn lại chế độ



Âm thanh hiệu ứng

- click.mp3 khi đánh dấu
- win.mp3 khi thắng
- tie.mp3 khi hòa
- Sử dụng pygame.mixer

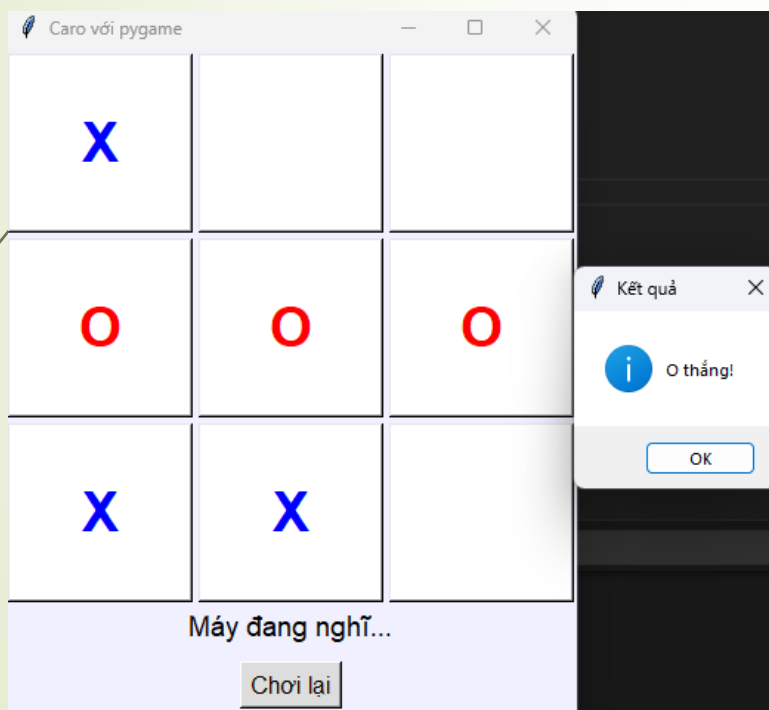


Giao diện người dùng

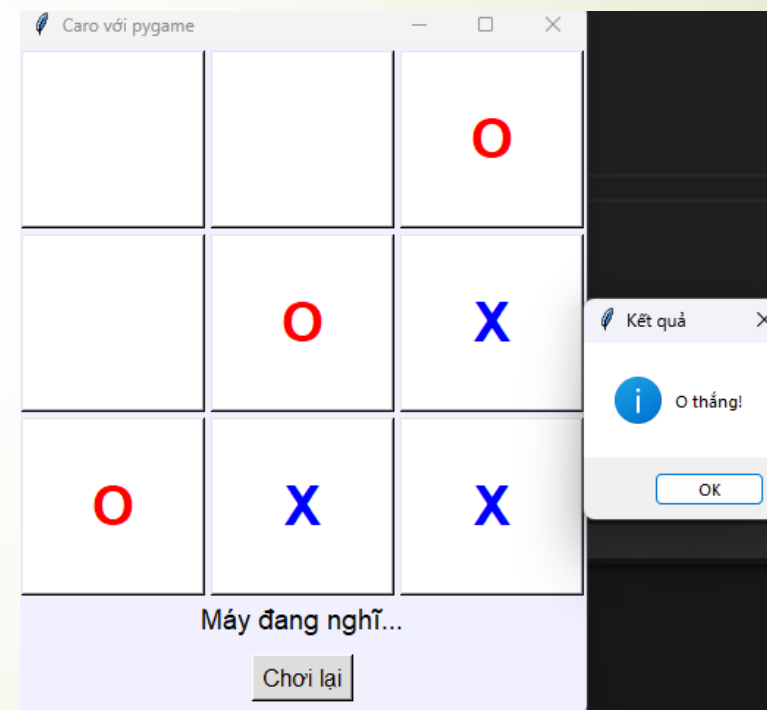
- Bảng 3x3 bằng 9 nút
- Màu X xanh, O đỏ
- Hiển thị lượt và kết quả

Kết quả chạy thực tế

Khi thắng hàng dọc/ngang

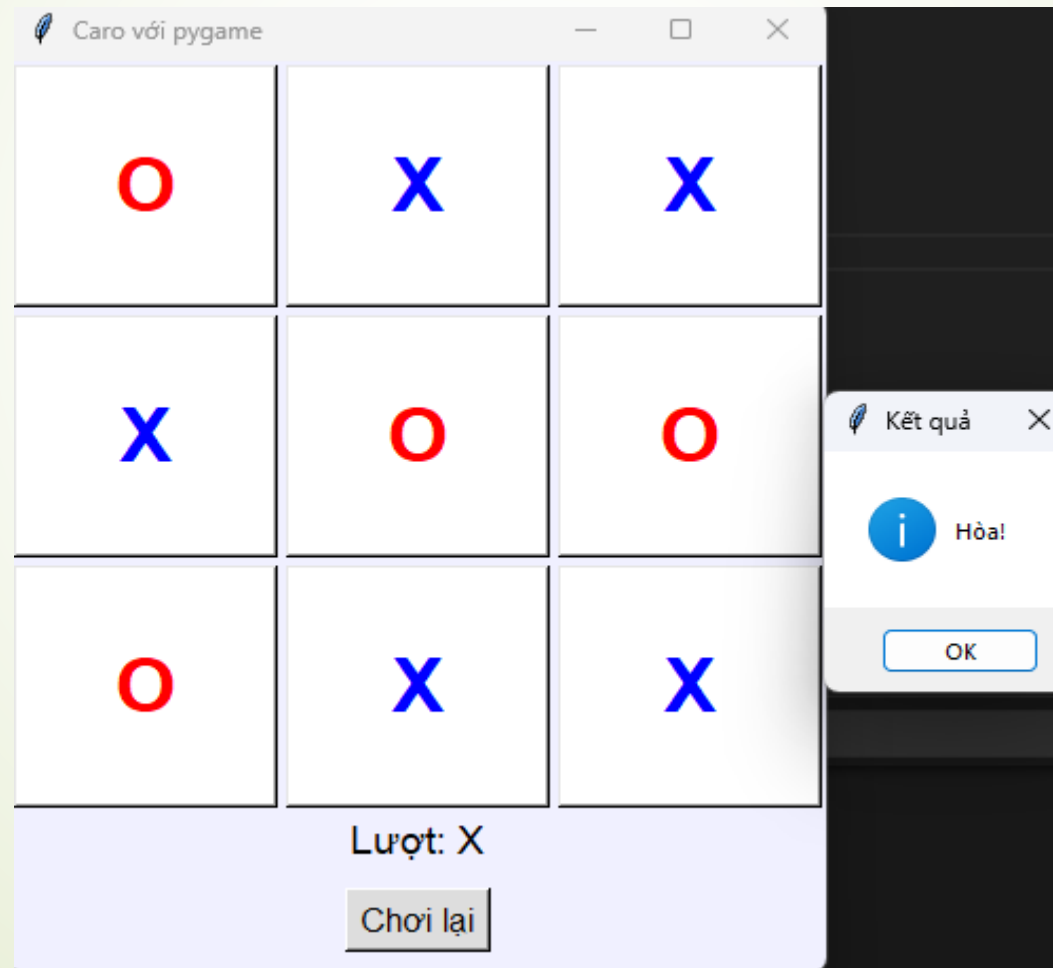


Khi thắng hàng chéo



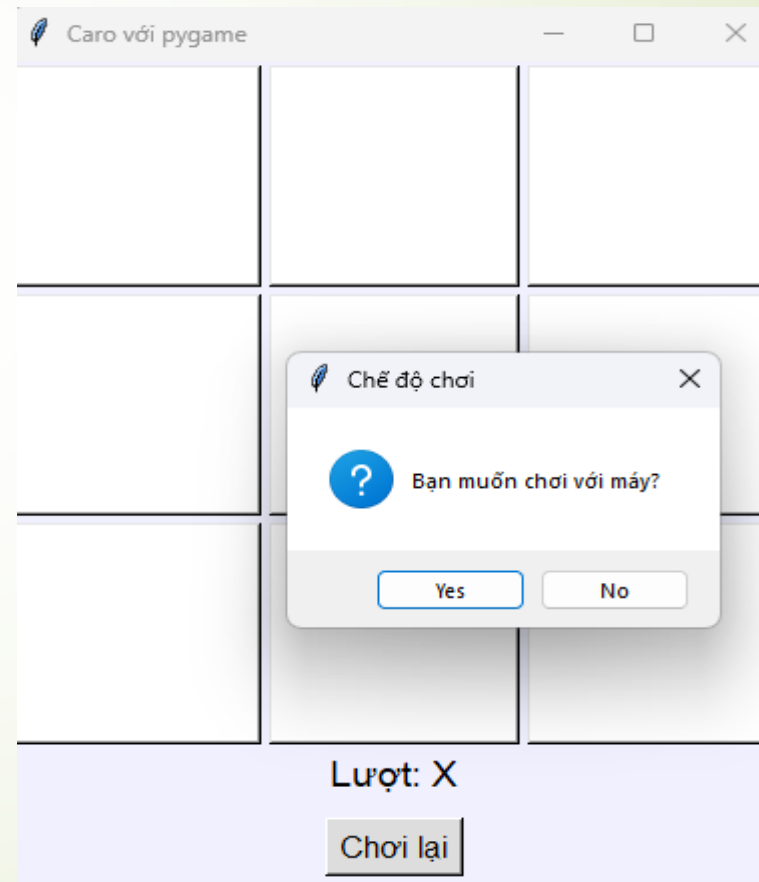
Kết quả chạy thực tế

- ➡ Khi hòa → hiển thị thông báo “Hòa”



Kết quả chạy thực tế

- Khi ấn "chơi lại" hiện cửa sổ thông báo chọn chế độ chơi
- - Yes để chơi với AI
- - No để chơi 2 người

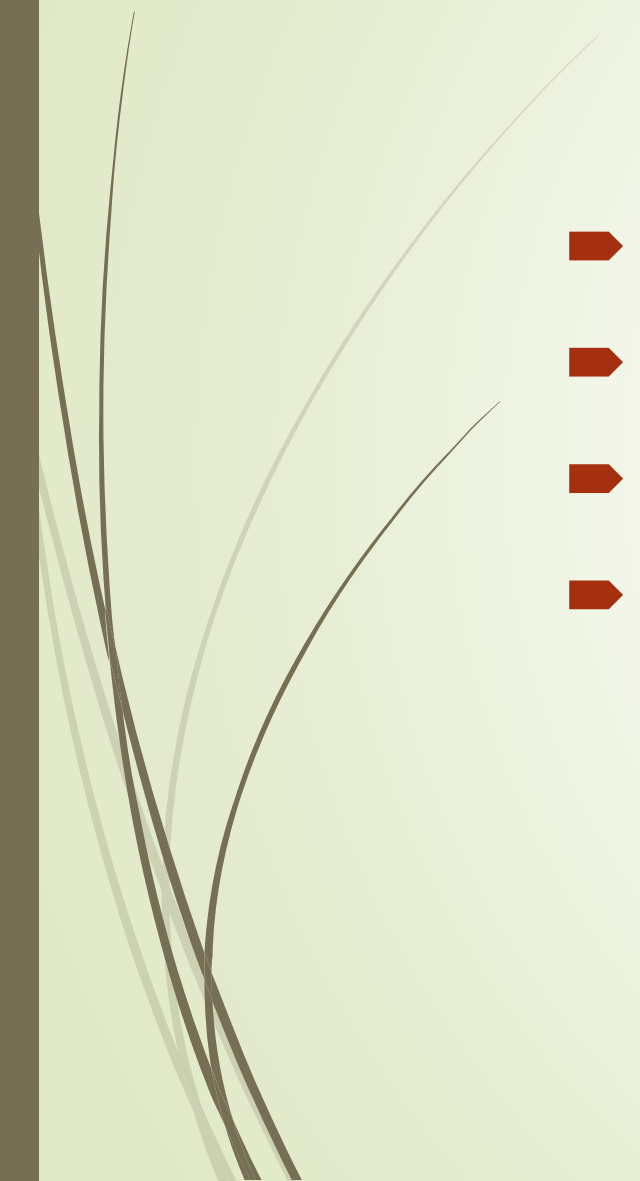


Tổng kết chức năng

- ☒ Theo dõi lượt chơi
- ☒ Kiểm tra hợp lệ
- ☒ Xác định người thắng
- ☒ Reset
- ☒ Vô hiệu hóa nút
- ☒ Âm thanh hiệu ứng

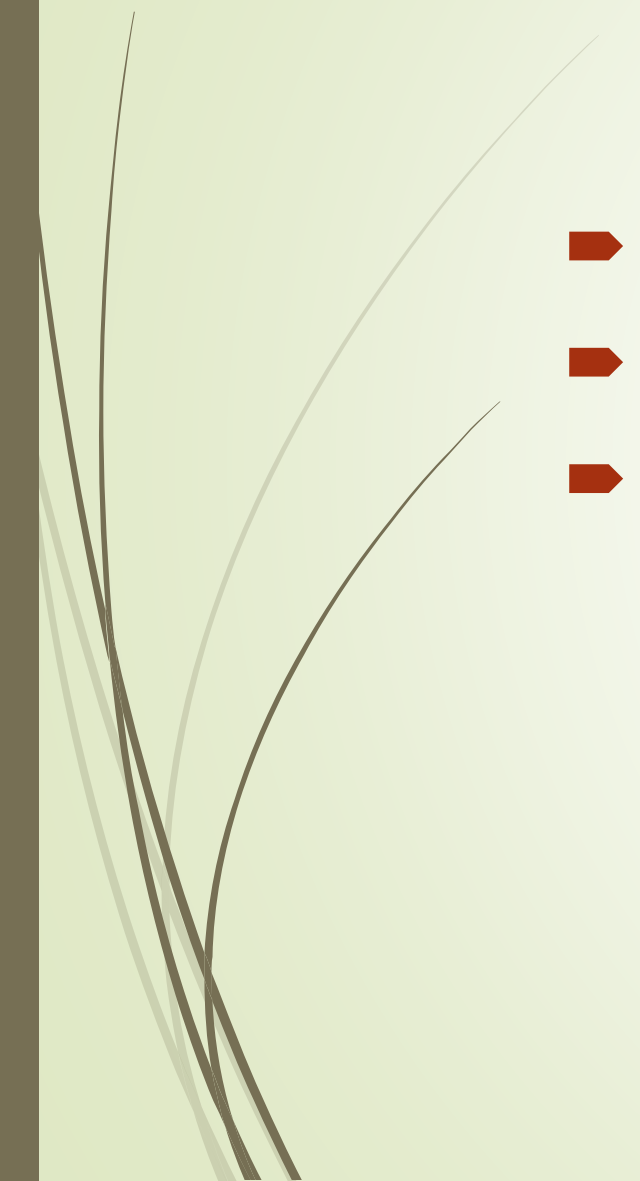



Những gì đã học được

- Lập trình GUI bằng Tkinter
 - Tổ chức mã theo hướng đối tượng
 - Xử lý âm thanh bằng Pygame
 - Tư duy hệ thống và tương tác người dùng
- 



Những khó khăn khi thực hiện

- Làm quen Tkinter
 - Kiểm tra thẳng đúng logic
 - Kết hợp GUI và âm thanh không đơ giao diện
- 



Cải tiến trong tương lai

- Nâng cấp AI (Minimax)
- Giao diện tùy biến
- Lưu điểm và lịch sử
- Xuất bản bản chạy .exe hoặc bản web



Kết luận

- Sản phẩm đáp ứng yêu cầu đề bài
- Giao diện thân thiện, âm thanh sống động
- Có tiềm năng mở rộng và triển khai thực tế

Cảm ơn thầy và các bạn đã xem !