

Universidad Mariano Gálvez de Guatemala

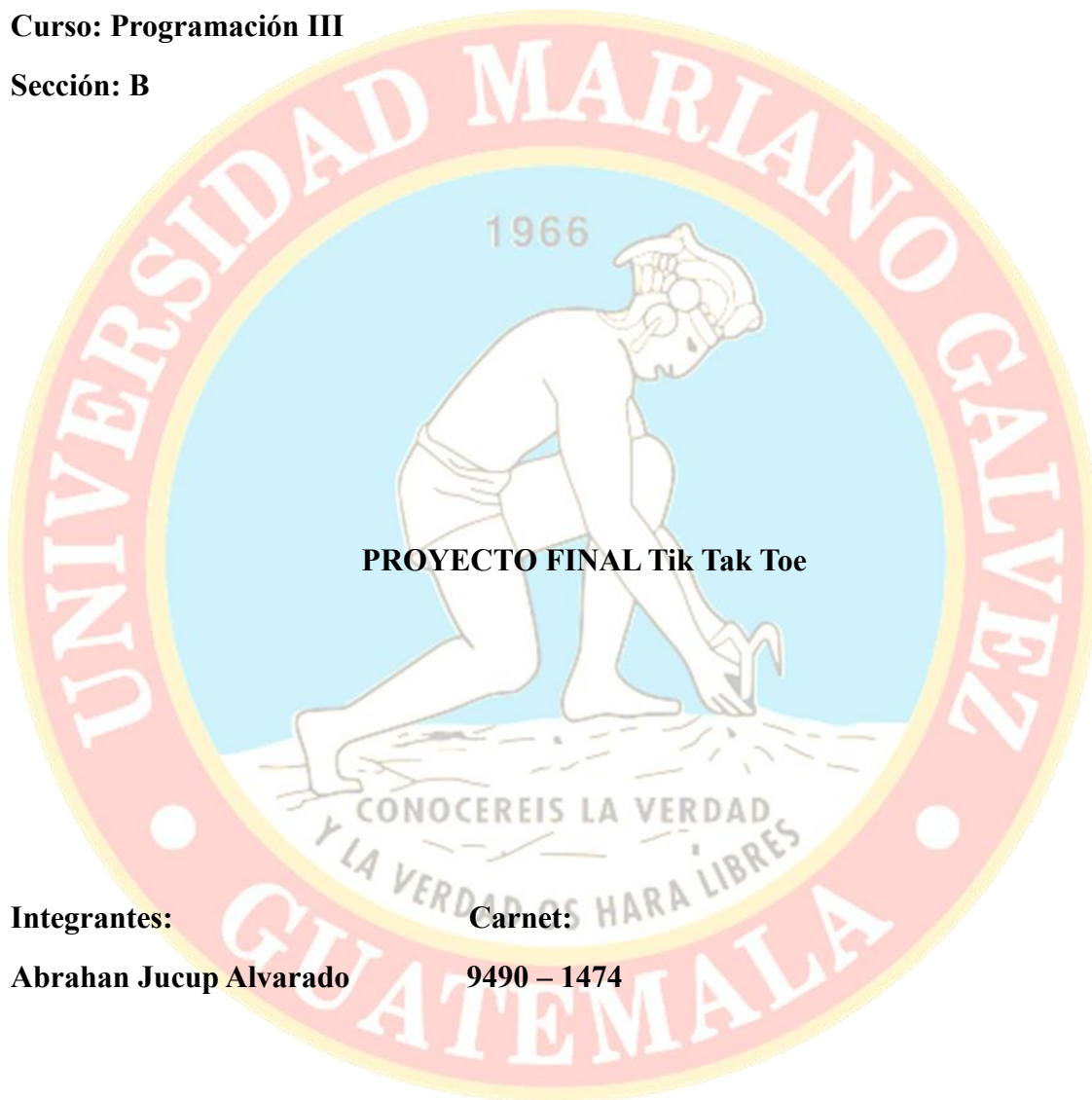
Centro universitario el Naranjo

Facultad de Ingeniería en Sistemas y Ciencias de la Computación

Catedrático: Ing. José Miguel Villatoro Hidalgo

Curso: Programación III

Sección: B



PROYECTO FINAL Tik Tak Toe

Integrantes:

Abrahan Jucup Alvarado

Carnet:

9490 – 1474

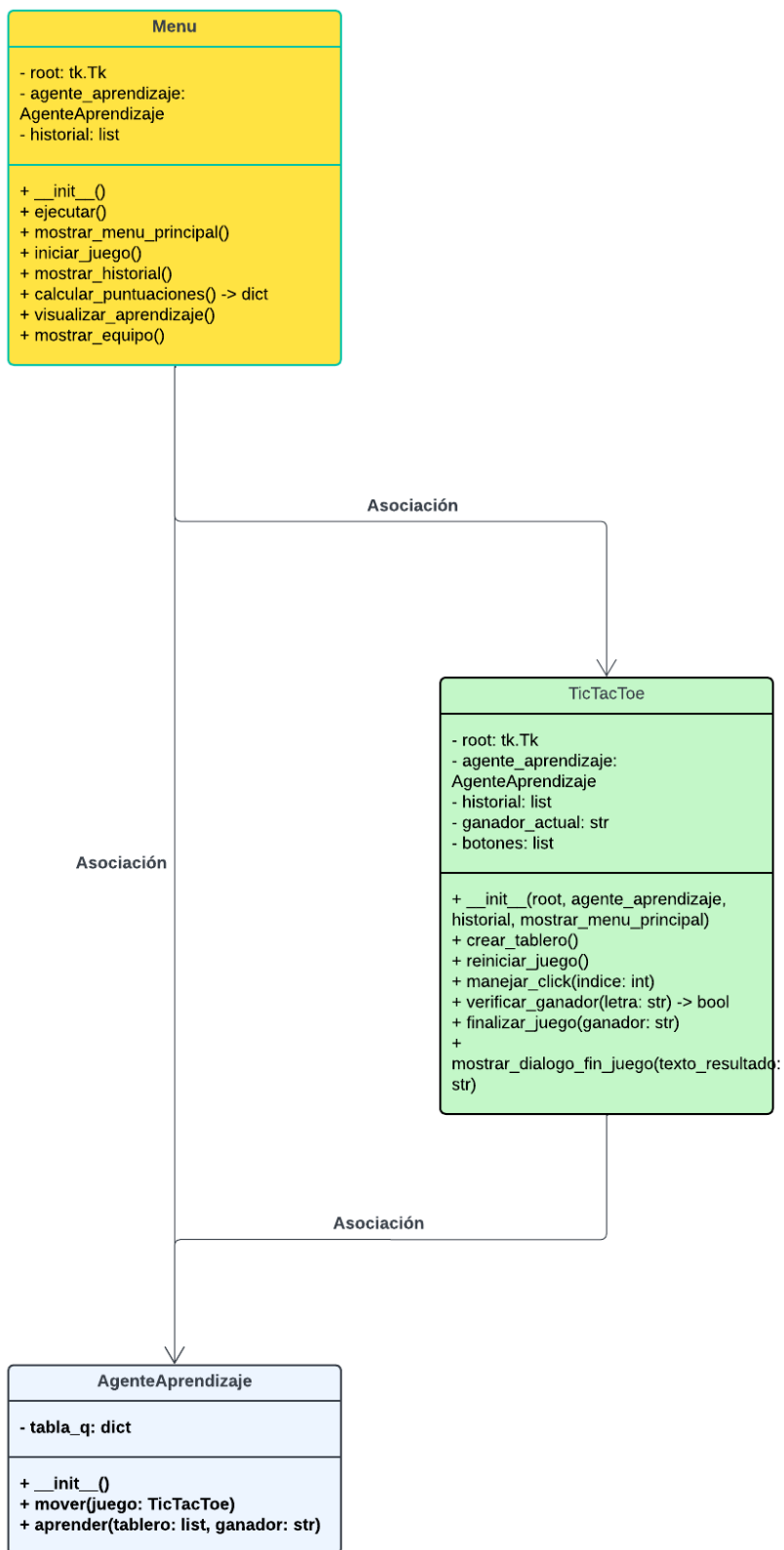
Guatemala, 31 de mayo de 2024.

MANUAL TECNICO

Esquema conceptual: Describe la estructura general del proyecto y las relaciones entre sus componentes principales.



Diagrama de Clases: El diagrama de clases representa la estructura de clases del proyecto y sus relaciones.



Descripción de Clases

Se proporcionará una descripción detallada de cada clase importante del proyecto, incluyendo su propósito, atributos y métodos.

Clase TicTacToe:

Propósito: Esta clase controla el juego del tres en raya, incluyendo la lógica del juego, la gestión del tablero y la interfaz gráfica.

Atributos:

root: La ventana principal de la interfaz gráfica.

learning_agent: Una instancia de la clase LearningAgent que representa al agente de aprendizaje.

history: Una lista que almacena el historial de partidas jugadas.

current_winner: Una cadena que almacena al ganador de la partida actual.

buttons: Una lista de botones que representan las casillas del tablero en la interfaz gráfica.

Métodos:

__init__(root, learning_agent, history, show_main_menu): Constructor de la clase, inicializa los atributos y crea el tablero de juego en la interfaz gráfica.

create_board(): Crea los botones del tablero en la interfaz gráfica.

reset_game(): Reinicia el juego, restaurando el tablero y los botones a su estado inicial.

handle_click(index): Maneja el clic del usuario en un botón del tablero, realizando la jugada correspondiente y verificando si hay un ganador.

check_winner(letter): Verifica si hay un ganador en el juego para la letra especificada (X u O).

end_game(winner): Finaliza el juego, mostrando el resultado y actualizando el historial de partidas.

show_end_game_dialog(result_text): Muestra un cuadro de diálogo al finalizar el juego con el resultado.

Clase LearningAgent:

Propósito: Esta clase implementa un agente de aprendizaje que juega al tres en raya contra el usuario.

Atributos:

q_table: Un diccionario que almacena la tabla Q para el aprendizaje del agente.

Métodos:

__init__(): Constructor de la clase, inicializa la tabla Q.

move(game): Realiza el movimiento del agente en el juego especificado.

learn(board, winner): Actualiza la tabla Q basándose en el resultado de la partida.

Clase Menu:

Propósito: Esta clase gestiona el menú y las opciones del juego, así como la navegación entre las diferentes funcionalidades.

Atributos:

root: La ventana principal de la interfaz gráfica.

learning_agent: Una instancia de la clase LearningAgent que representa al agente de aprendizaje.

history: Una lista que almacena el historial de partidas jugadas.

Métodos:

run(): Inicia la ejecución del menú.

show_main_menu(): Muestra el menú principal en la interfaz gráfica.

start_game(): Inicia una nueva partida del juego.

show_history(): Muestra el historial de partidas jugadas.

calculate_scores(): Calcula y devuelve las puntuaciones de las partidas.

visualize_learning(): Visualiza el progreso de aprendizaje del agente.

show_team(): Muestra la información sobre los integrantes del proyecto.