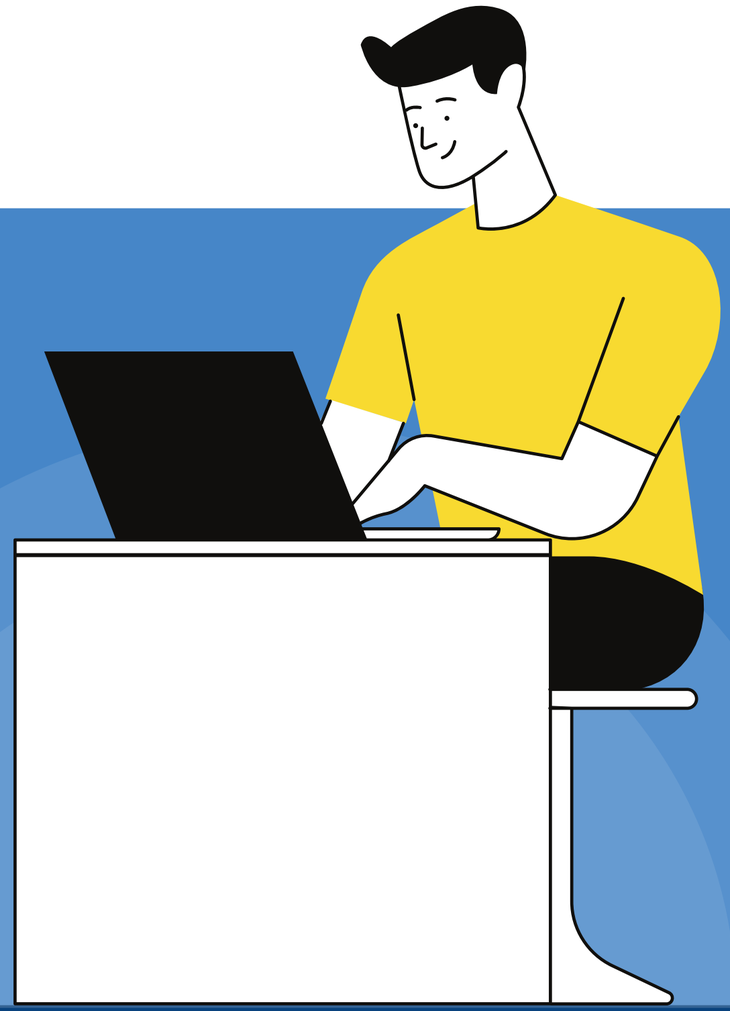
Connect Four



CREATED AND DEVELOPED BY: XANDER ENDRE

Discussion Outline

Topics for discussion

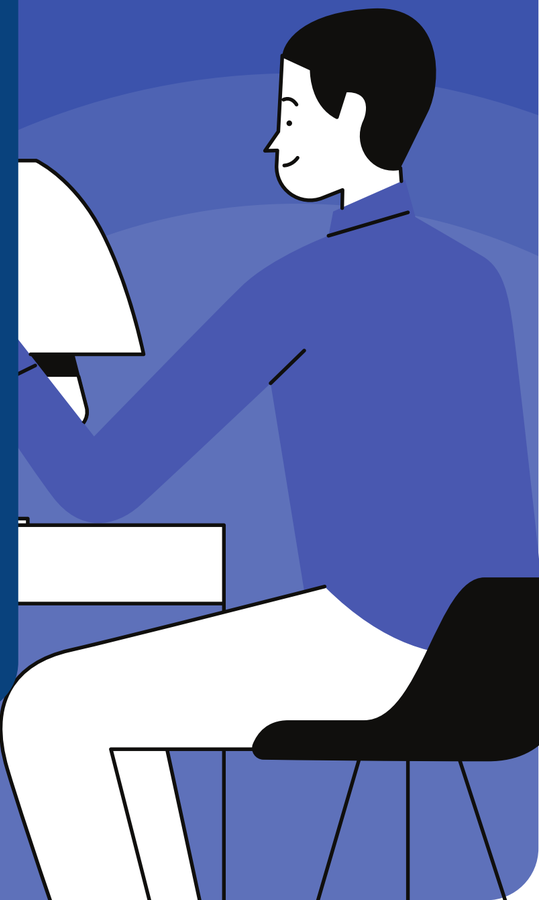
01 What is this project?

02 Why did I choose this project?

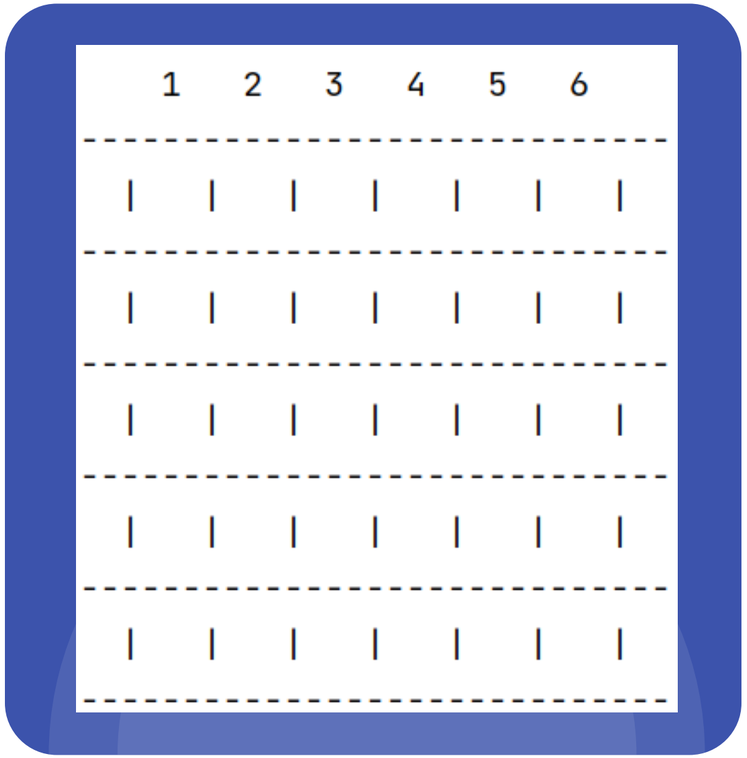
03 What is UML Design?

04 How does the game work?

05 What does the game look like?



XANDER ENDRE | **NEUMONT COLLEGE OF COMPUTER SCIENCE**



This project is a game of Connect Four that was designed and developed for Command Line in IntelliJ Idea. In this project you can play **player versus player**, **player versus computer**, and **computer versus computer**

What is this project?

Why choose Connect Four

After speaking with the tutor Robert he gave me the instructions for creating a Connect Four Game from the Object Orientated Project class.

XANDER ENDRE | **NEUMONT COLLEGE OF COMPUTER SCIENCE**

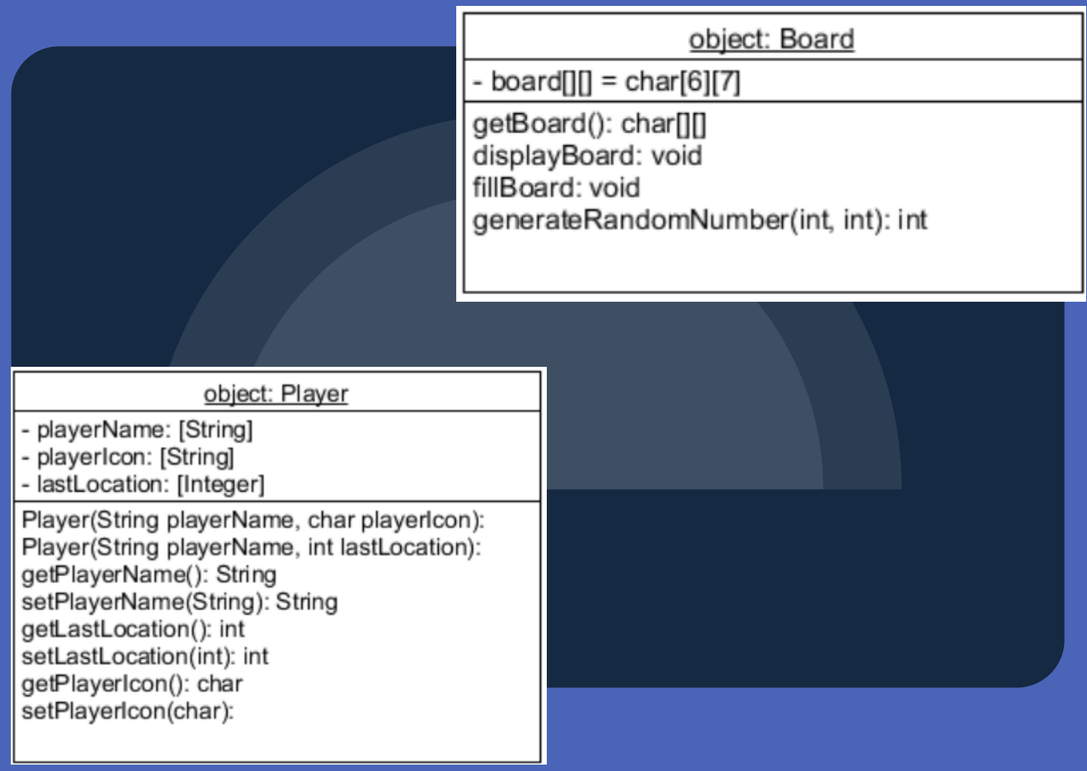
Overview of the UML

This is an example of how I split the game of Connect Four into separate objects.

The concept behind the UML of a game of Connect Four can be both simple but complex. In this project there are two separate objects that we will be focusing on. The **Player Object** and the **Board Object.**

The player object deals with everything in relation to the player with their name, icon, and last location.

While the board deals with fetching the board, displaying the board, filling the board and generating a random number for the board.



How does this game work?

```
          CONNECT FOUR
  ----------------------------
        Choose a game type!

  - Player versus Player (PVP)
  - Player versus Computer (PVC)
  - Computer versus Computer (CVC)

  ----------------------------
Enter a game type:
```

01 Choose Game Type

The game starts by the user choosing a player type of game. There are three different types of games: player versus player, player versus computer, and computer versus computer.

02 Play the Game

Once the game starts the first and second player will be asked to place their icon in a specific column. From there the game will start stacking the icons and allow for regular gameplay.

03 Win the Game

After every player turn the checkWin() method is run which will check if a player has won in any direction: vertical, horizontally, upward diagonally, and downward diagonally.

Visibility of Knowledge

A list of all the requirements that my code has met.

Code Quality	Multiple Classes	If Statements	Methods with parameters and return types	Getters and Setters
Constructors	Method Overloading	Arrays	Text Input & Outputs	For and While Loops

XANDER ENDRE | **NEUMONT COLLEGE OF COMPUTER SCIENCE**