Bomberman Turn-Based Online Multiplayer

Door Niels Vriezer

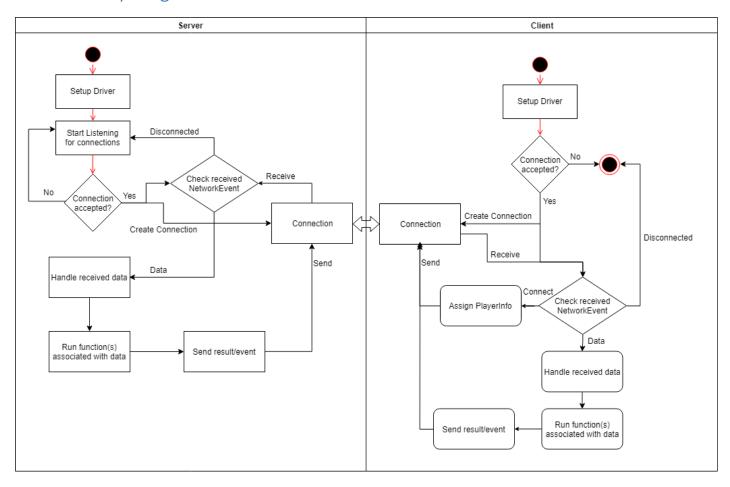
Het design:

Wat gebeurd er als je het concept van bomberman neemt, maar het turn-based maakt? Dit vroeg ik me af toen ik bedacht dat ik graag bomberman wilde maken, maar dat de game per se turn-based moest zijn.

Uiteindelijk is het dus een bomberman kloon geworden genaamd Bombo. Elke speler krijgt om de beurt de kans zijn karakter te bewegen en bommen te plaatsen. De karakter kan tijdens een beurt vrij bewegen binnen een bepaalde radius. Ook kan de speler tijdens zijn of haar beurt oneindig veel bommen plaatsen. Echter zal de speler zeer goed moeten opletten dat hij of zij niet zichzelf raakt en een level verliest.

Als een bom af gaat en een speler wordt geraakt, dan zal die speler een beurt overslaan. Een bom zal afgaan na 2 beurten. Ook zijn er kratten in het level. Deze zullen, net zoals in het origineel, verdwijnen waarna de speler ook op die posities kan komen. Echter zitten er geen power ups verstopt in de kratten.

Activity Diagram:



Class Diagram:

ConnectionEvent Enum

PLAYER, DISCONNECT
REQUEST, OPPONENT
START, GAME
REJORNO, START, GAME
REQUEST, LOADED
REQUEST, LOADED
REQUEST, GAME, READY
REA

IMessageStruct ♠

■ Fields
■ playerID : uint
■ positionX : float
■ positionY : float
■ positionZ : float

O IMessageStruct

CrateDestroyStr...

amountOfData...cratesToDestro...

O IMessageStruct

■ Fields
 ■ matchID: uint

O IMessageStruct TurnUpdateStruct ♠

playerID : uint
totalTurns : uint

○ IMessageStruct PlayerPlaceBom... A

playerID : uint O IMessageStruct GameOverStruct ♠

AssignOpponen... A

PlayerConnectSt... A

RequestOppone...

PlayerMoveStruct Struct

Fields
newPosition: Vector3
playerID: uint

O IMessageStruct

BombSpawnStruct Struct

Fields

bombID: uint
bombPosition: Vector3

■ Fields

■ amountOfPlayers: uint

■ playerLives: List<uint>
■ totalTurns: uint

■ Fields
■ amountOfData: uint
■ bombID: uint
■ flamePositions: List<Vector3>

UIStateUpdateStruct Struct

O IMessageStruct BombExplodeStruct Struct PlayerEndTurnSt... A

PlayerPositionUpdateStruct Struct

Fields

amountOfData: uint

playerID: uint

walkablePath: List < Vector 3 >

■ Fields
■ centerPosition: Vector3
■ listLength: uint
■ walkablePositions: List<Vector3>

amountOfPlayers: uint
 playerInfoCharacter: List < uint>
 playerInfoID: List < uint>
 playingFieldID: uint

O IMessageStruct

StartGameStruct Struct

PlayerNamesStruct Struct

