

# ***Kernvak 3: AI***

Max Kersten  
3020282

## Inleiding

Voor dit kernvak was de opdracht om een AI te maken die in staat is om tussen waypoints te navigeren. Verder moet deze AI de speler moeten detecteren. Als de AI de speler detecteerd dient deze een aantal stappen te ondernemen. Deze stappen zijn:

- Een wapen zoeken/detecteren.
- Het wapen ophalen en equippen.
- Navigeren naar de speler / laast geziene locatie.
- Het aanvallen van de speler.
- Het wapen terug brengen.
- Verder navigeren tussen waypoints.

Ninja AI:

Deze AI dient zich te verstoppen als de enemy AI de speler ziet. Als de Ninja AI zich met succes heeft verstoppt, gooit deze een rookbom tussen de speler en de enemy AI.

## Gekozen systeem

Om dit systeem te bouwen heb ik gekozen voor een redelijk traditionele oplossing. Een behaviour tree met behavioural-nodes, Sequences en Selectors. Ik heb hier voor gekozen omdat ik met dit systeem duidelijke fail en succes conditions kan hangen aan de behavioural-nodes.

## PMI

**Plus:** Dit systeem is makkelijk op te zetten en uit te breiden voor verschillende doeleinde. Online is er veel informatie te vinden over dit soort systemen en wordt vaak toegepast in echte projecten.

**Min:** Omdat dit systeem zo simpel is, is het ook heel snel om te vormen tot een draak van een project. Het opsplitsen van gedrag is soms moeilijk en naderhand gedrag aanpassen kan betekenen dat heel de tree er anders uit komt te zien. Je moet goed van tevoren uitwerken welk gedrag een AI moet hebben.

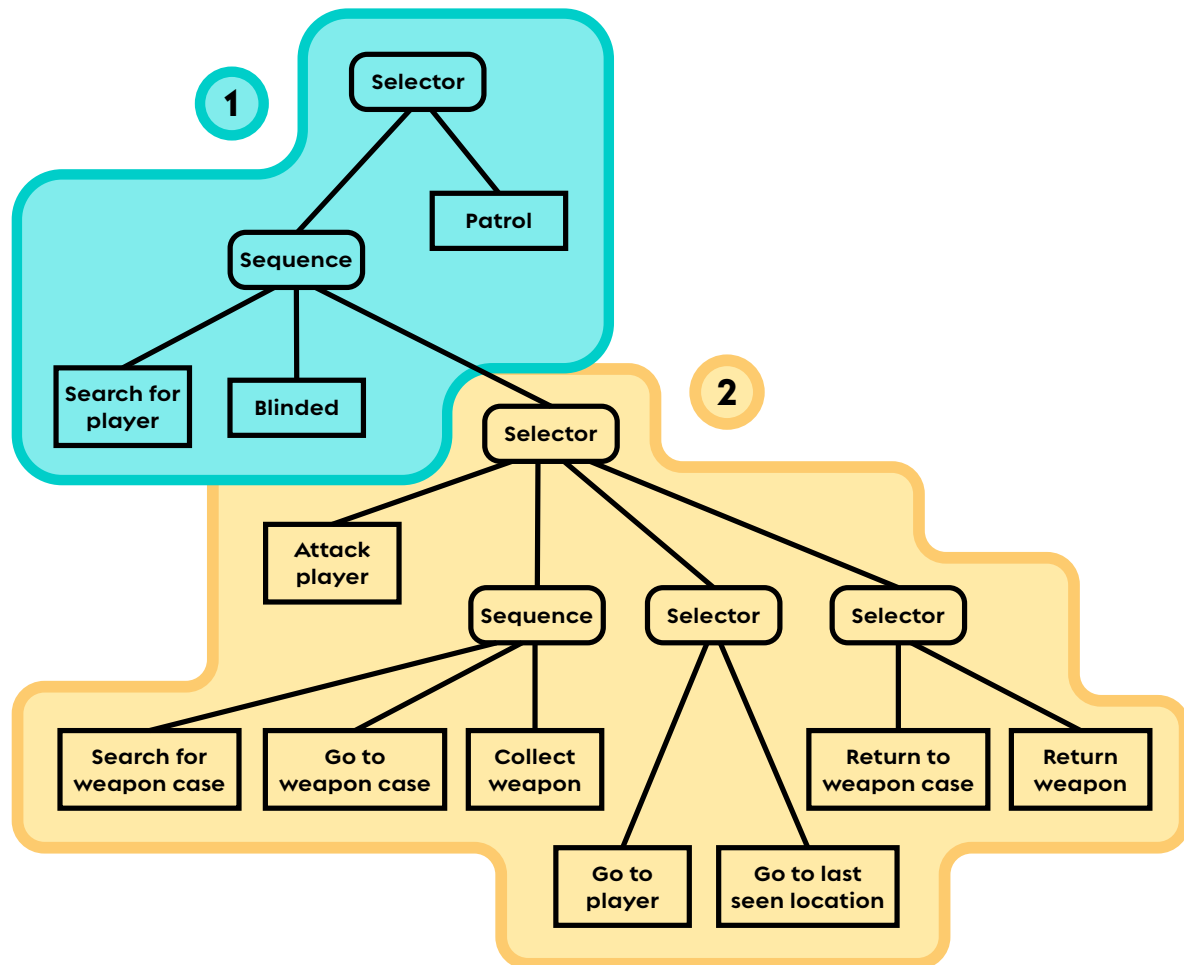
**Interessant :** Omdat dit type systeem al vaak is toegepast is er online veel informatie te vinden om je in de juiste richting te sturen. Dit kan ook een valkuil zijn. Soms is de meest gebruikte oplossing niet de beste.

## Guard AI

**Deel een:** Dit gedeelte checkt of de guard de speler kan zien. Als de speler niet zichtbaar is, zal de guard de "Patrol" state in gaan en van punt tot punt lopen.

**Deel twee:** De AI zal dit deel ingaan als de speler zichtbaar is. Als hij geen wapen heeft zal hij op zoek gaan naar een kast met wapens. Als hij een kast heeft gevonden, zal hij hier naar toe navigeren en als hij in range is van de kast, pakt hij het wapen op. Hierna checkt hij hoever hij zich van de speler bevindt. Is dit verder dan de range van het wapen. Zal hij naar de speler navigeren. Is de speler in range, dan zal hij de speler beschieten. Is de speler na het ophalen van het wapen niet meer zichtbaar. Dan zal de guard naar de positie lopen waar de speler voor het laatst is gezien. Als de guard de speler niet heeft gevonden, zal hij het wapen terug brengen en weer in de "patrol" state gaan.

## Guard Tree



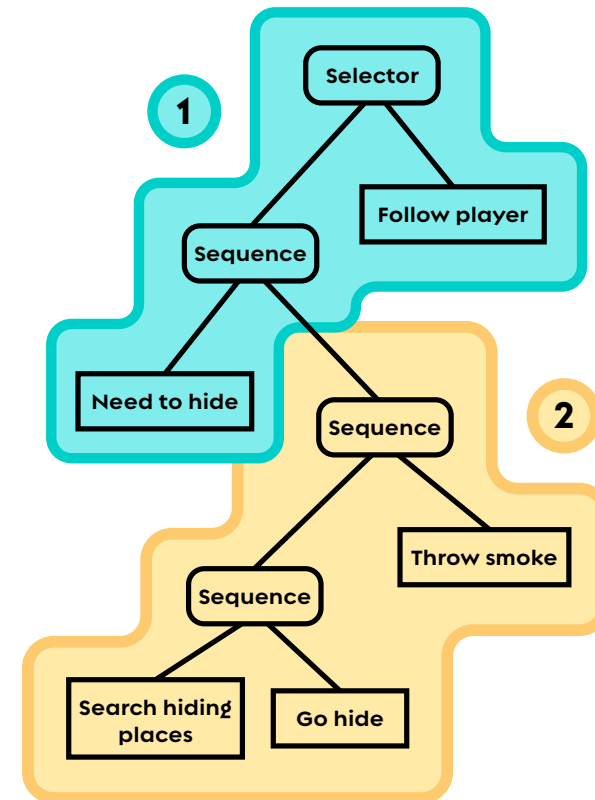
## Ninja AI

**Deel een:** Dit gedeelte checkt of de ninja zich moet gaan verstoppen. Hij zal zich gaan verstoppen als de guard in de "Attack player of Go to player" state bevindt. Anders zal de ninja de speler blijven volgen.

**Deel twee:** De ninja AI zal dit gedeelte ingaan als de guard zich in de correcte state bevindt. De ninja zal op zoek gaan naar een verstopplek die onzichtbaar is voor de guard en het dichtste bij de speler bevindt. Als de ninja zich op de verstopplek bevindt, zal hij een rookbom gooien en de visie van de guard beperken.

Mocht ondertussen de verstopplek van de ninja gecomprimeerd worden, zal hij op zoek gaan naar een andere verstopplek en naar deze toe navigeren.

## Ninja Tree



**Video:** [https://youtu.be/byw\\_\\_cabxiuw](https://youtu.be/byw__cabxiuw)

**Git:** <https://github.com/mcrkersten/AISystem>

**Build:** <https://github.com/mcrkersten/AISystem/blob/main/Documentation/Build/AI-build.zip>