**Pathfinding**

A Unity navigációs rendszer alapvetően nem támogatja a 2d-s patfinding-ot, ezért egy packeg-et kellett letölteni a <https://arongranberg.com/astar/download> oldalról. Már az ingyenes verzió is tartalmazott mindent, ami nekünk kellett.

Az *A\** nevű objektum a pathfinding közepe. Ennek a *Pathfinder* componens-e határozza meg melyik területen engedélyezett a mozgás. Ehhez szüksége van egy *Obstacle Layr Mask*-ra, ami az akadályokat és azok *collider*-ét tartalmazza, a mi esetünkben ez az *npc\_path*. Lényegében ezután végignézi minden csomóponton, hogy keletkezne e átfedés, és ez alapján létrehoz egy *navigation gride*-et, ami a képen látható. A kék részeken engedélyezett, a piros négyzettel jelölt részeken nem engedélyezett a mozgása.

A *grafika* és a *logika* el van különítve két objektumba. Ez azért hasznos, mivel így könnyen ki lehet cserélni a grafikát anélkül hogy a logikába bele kéne nyúlni.

A logikához hozzá van adva az *AIPath* és a *Seeker* componens. Ez a kettő olyan scriptek, amik minden AI objektum működéséhez szükségesek. A *Seeker* felelős a célhoz vezető útgenerálásért. Az *AIPath* pedig a *Seekert* fogja kontrollálni és mozgatni a grafikát.

Cél megadásához hozzá kell csatolni az *AI Destination Setter -* *Target*-jához a kívánt objektumot. Ahhoz, hogy a grafika mindig a cél felé nézzen az x sebesség alapján kell tükröznünk. Ha ez pozitív, akkor jobbra, ha negatív, akkor balra halad a karakter. Ez az alábbi kódrészletben látható:

void Update()

{

// flipping based on X speed

if(aiPath.desiredVelocity.x >= 0.01f) // moving to the right

{

transform.localScale = new Vector3(-1f, 1f, 1f);

}

else if (aiPath.desiredVelocity.x <= -0.01f) // moving to the left

{

transform.localScale = new Vector3(1f, 1f, 1f);

}

}