**Het inladen van plaatjes in XNA**

Je begint door het aanmaken van een Texture2D object en het plaatje wat wil gebruiken int te laden in de content.(LET OP NIET DE CONTENT REFERENCE)

Voorbeeld:

public class Game1 : Microsoft.Xna.Framework.Game

{

GraphicsDeviceManager graphics;

SpriteBatch spriteBatch;

Texture2D crosshair; 🡨

Hierna zorg je ervoor dat het plaatje wat je wil weergeven ingeladen word in het object

Dit doe je in de LoadContent class

protected override void LoadContent()

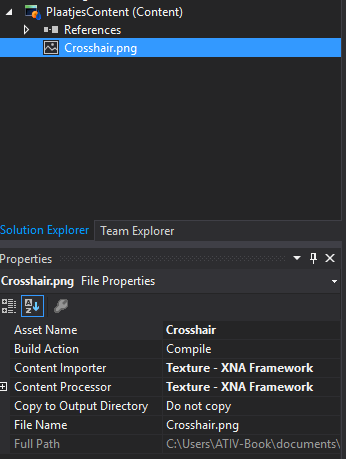
{

// Create a new SpriteBatch, which can be used to draw textures.

spriteBatch = new SpriteBatch(GraphicsDevice);

crosshair = Content.Load<Texture2D>("Crosshair"); 🡨

Crosshair verwijst hier naar de naar de asset naam die je terug kunt vinden in properties.



Nadat je dit allen gedan hebt kun je plaatjes inladen doormiddel van de volgende code(hier ga ik in het volgende blok dieper op in)

spriteBatch.Begin();

spriteBatch.Draw(crosshair, new Rectangle(0, 0, 50, 50), Color.White);

spriteBatch.End();

Zoals je kunt zien spreken regel 1 en 3 voor zichzelf Regel 2 heeft een hoop opties die je hier kunt terugvinden

<https://msdn.microsoft.com/en-us/library/ff433989.aspx>

dit is als volgt opgebouwd

spriteBatch.Draw(Naam van de Texture2D die je wilt inladen , new Rectangle(X coordienaat waar je het wilt hebben, Y coordinaat, Breedte in PX, Hoogte in PX), Color. Gevolgd door gewenste achtegrond kleur);

Breedte en hoogte scalen de grootte van je plaatje(dit vond ik zelf verwarrend als dat c# meestal cropt op deze gegevens)

Breedte en hoogte kunnen ook vervangen worden door Texture2D.Width/Height

Er zijn ook veel andere opties die kun je terug vinden in de link.