

How to develop Pong

GLOW PLANT-E
TADRAŁA, PIOTR P.P.

O-PP-CMK

Inhoudsopgave

Versiebeheer.....	2
Inleiding	3
Wat is Pong	3
Plant-e integratie	4
PONG ontwikkelen.....	4
Platform	4
Spellogica	4
Gebruikersinteractie.....	5
Scalability / Extensions	5
Conclusie.....	5

VERSIEBEHEER

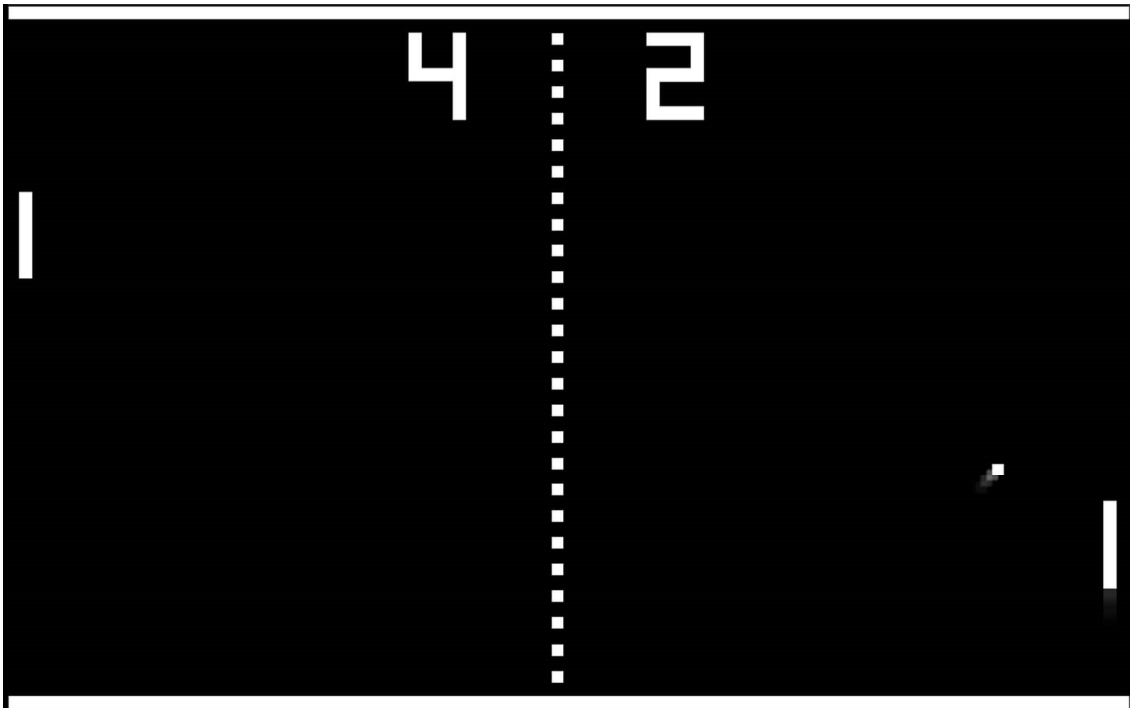
Versie	Datum	Aanpassing
0.1	17-03-2023	Eerste opzet van het document

INLEIDING

In ons project, gericht op het demonstreren en visualiseren van de mogelijkheden van Plant-e technologie, hebben we gekozen voor de simulatie van het spel Pong met behulp van e-planten. Dit willen we bereiken door een grid van planten te gebruiken, waarbij elke plant een of meerdere pixels voorstelt. Op deze manier creëren we een lage resolutie scherm waarop Pong gespeeld kan worden. Dit document is bedoeld als een onderzoek naar de ontwikkeling van de software die nodig is voor dit spel.

WAT IS PONG

Pong, een klassiek arcadespel ontwikkeld door Atari in 1972, heeft een eenvoudige opzet. Aan beide kanten van het scherm bevindt zich een platform, bestuurd door een speler, met in het midden een bal. Het doel is om de bal zo te stuiten met je eigen platform dat deze de tegenstander passeert om te scoren. De speler met de meeste punten wint.



PLANT-E INTEGRATIE

Om het spel met behulp van e-planten te simuleren, hebben we het volgende concept bedacht. We recreëren een lage resolutie scherm door een x-aantal planten in een grid te plaatsen, waarbij elke plant als een of indien mogelijk meerdere pixels functioneert. Via een controller, waarop we het spel zullen runnen, communiceren we met de planten-LEDs om deze aan en uit te schakelen. Zo visualiseren we het spel.

PONG ONTWIKKELEN

PLATFORM

Als we ervoor kiezen en het mogelijk is om de controller, die het spel draait, ook te voeden met energie van e-planten, wordt het cruciaal een programmeertaal te kiezen die zo efficiënt mogelijk is in stroomverbruik. In dat geval kunnen we kiezen uit bijvoorbeeld C of C++. Mochten de planten niet genoeg energie leveren om de controller te voeden, en we besluiten het spel op een laptop te draaien, dan kunnen we in principe uit bijna alle programmeertalen kiezen, bijvoorbeeld JS, Python, C#, etc.

SPELLOGICA

BASISREGELS

Aangezien Pong een ouderwets arcade game is zijn de regels ook relatief simpel. Het spel simuleert een tafeltennis game waarbij 2 spelers een paddle besturen. Doel van het spel is om de bal, die in het midden begint, en in eerste instantie een random kant op gaat, naar de overkant te stuiteren zodat deze langs de tegenstaander gaat.

BALBEWEGING

In Pong beweegt de bal in een rechte lijn tot dat die een paddle of schermrand raakt, waarna die van richting verandert. De terugkatshoek wordt bepaald door de plek waar de bal de paddle raakt. Raakt de bal het midden van de paddle, dan wordt hij horizontaal teruggestuurd; raakt hij de rand, dan vliegt hij weg onder een scherpe hoek, tot 75 graden. De snelheid van de bal blijft altijd gelijk.

Bron: Ricket: [Pong, how to calculate ball's direction.](#)

GEBRUIKERSINTERACTIE

Gebruikers besturen hun paddle in de verticale axis met een joystick, waarbij ze deze naar de juiste positie moeten bewegen om de bal te onderscheppen en terug te kaatsen naar de tegenstander.

SCALABILITY / EXTENSIONS

We kunnen het originele spel natuurlijk spannender maken met extra toevoegingen, zoals meerdere ballen, tegenstander debuffs, teamspellen zoals 2v2 of ieder-voor-zich in een 1v1v1v1 opstelling, of maps met obstakels.

CONCLUSIE

In dit document zijn de kernaspecten en spellogica van het spel Pong toegelicht. Verder is er gekeken naar de potentiële uitbreidingen en hoe we het spel binnen de scope van de Plant-e technologie kunnen realiseren en welke platform we daarvoor zouden kunnen gebruiken.