
Major Project: Zone of Gates

Orlando Mee

Master of Animation, Games and
Interactivity (MAGI),
School of Design,
RMIT University,
Melbourne, Australia
s3690063@rmit.edu.au

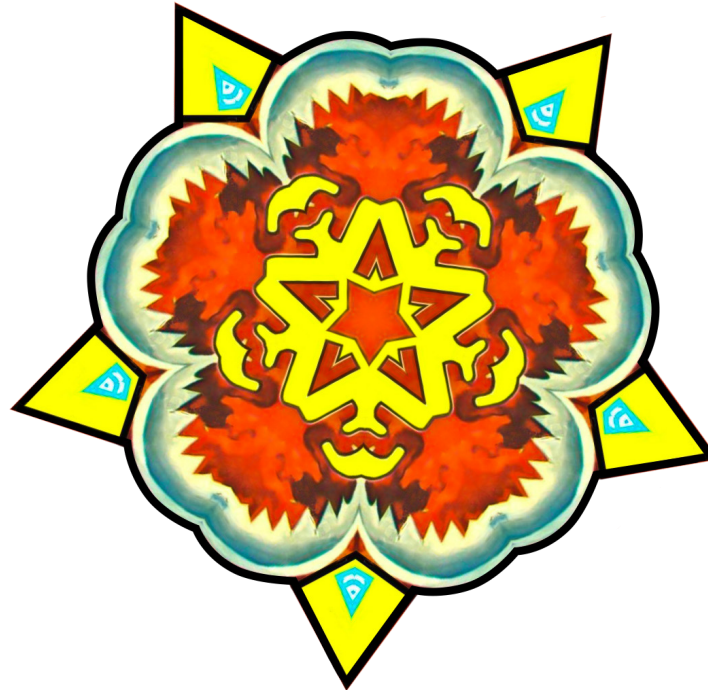
Abstract

A physical board game which uses NFC chips to link to the digital space, enabling novel game mechanics.

Written to function entirely in your smartphone's browser, it represents a simple but effective foray into the Internet of Things.

Authors Keywords

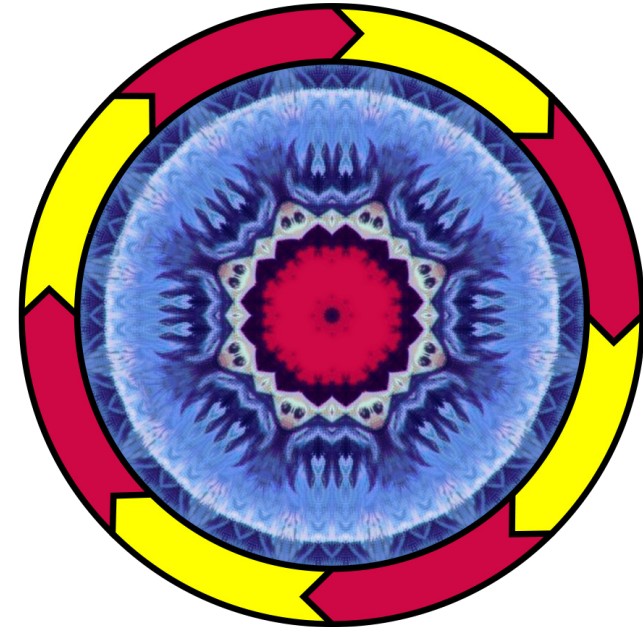
Near Field Communicator (NFC); PHP; Internet of Things; Board games



Introduction

In the first folio cycle of Studio 3, I explored a wide range of possible ideas for my major project. Eventually, I settled on a board game that uses Near Field Communicators (NFCs) to send instructions to the players' smartphones, which function as their playing pieces.

In the course of developing this game into a working product, I faced numerous challenges. I had to devise a system of rules for the game, incentivizing interesting behaviour and crafting a satisfying play experience. I also had to decide what type of software would power my game - whether I should create an app, or make the game entirely browser-based.



Most of all, I had to answer the question, "Why NFC?" I saw this question raised whenever NFC-based board games were discussed online. If a game can work equally well using physical cards and tokens, why involve NFC? On the other hand, if the bulk of the game exists in the digital space, why use a physical board at all?

After analyzing the unique properties of NFCs, I believe I have found a legitimate reason for their inclusion. In fact, they provide a play mechanic that is central to the experience of the game.

Prepare to enter... the *Zone of Gates*.

Research

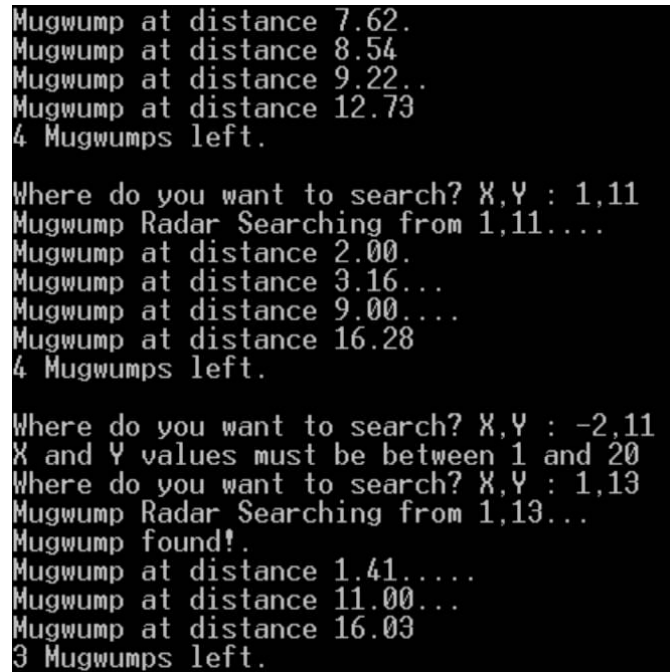
I first began looking into grid-based games, having moved on from my original idea of the player following a single path.

A Wikipedia search on the game Battleship led me to two text-based computer games of the 1970s, 'Mugwump' and 'Hunt the Wumpus'.

The former has the player search for 4 'mugwumps' by choosing coordinates on a 10x10 grid. Each wrong guess will tell the player how far away the closest mugwump is hiding.

Hunt the Wumpus has the player navigate a series of interconnected caves, aiming to hit the wumpus with an arrow without being eaten by it.

A set of gamebooks called Duel Masters: Challenge of the Magi was another influence. In this pair of choose-your-own-adventure books, two players navigate a series of locations (see map, top right), reading out keywords to determine if they are ever in the same place at the same time.

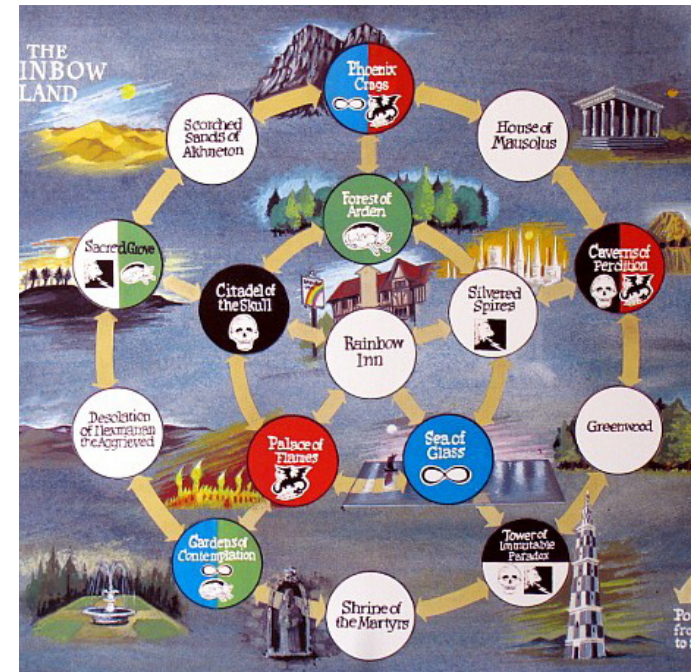
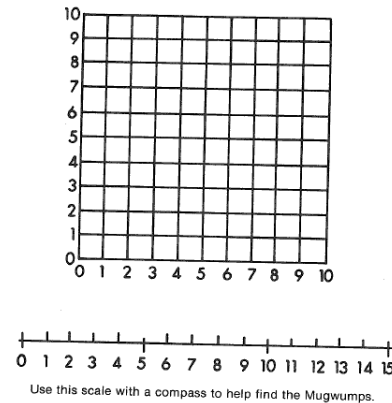


Your objective in this game is to find the four Mugwumps hiding on various squares of a 10 by 10 grid. Homebase (lower left) is position (0,0) and a guess is a pair of whole numbers (0 to 9), separated by commas. The first number is the number of units to the right of homebase and the second number is the distance above homebase.

You get ten guesses to locate the four Mugwumps; after each guess, the computer tells you how close you are to each Mugwump. Playing the game with the aid of graph paper and a compass should allow you to find all the Mugwumps in six or seven moves using triangulation similar to Loran radio navigation.

If you want to make the game somewhat more difficult, you can print the distance to each Mugwump either rounded or truncated to the nearest integer. Statement 390 would print either INT (D+.5) or INT (D).

This program was modified slightly by Bob Albrecht of People's Computer Company. It was originally written by students of Bud Valenti of Project SOLO in Pittsburgh, Pennsylvania.



Developing the Concept

I started thinking of the project as a two-player game taking place on a grid (and later, on separate grids to avoid having two phones on the one space).

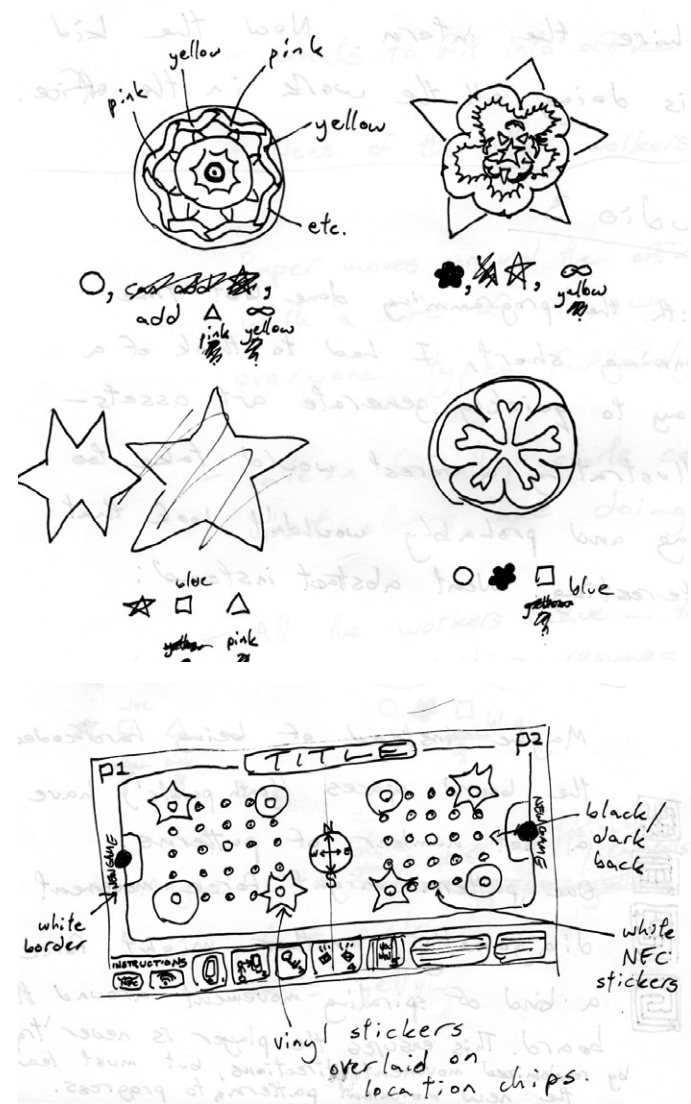
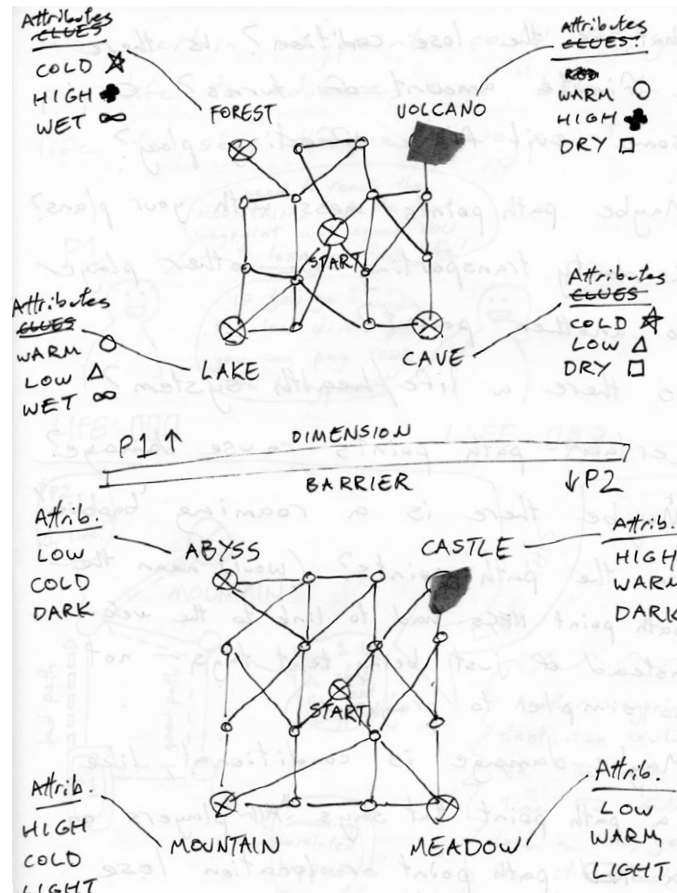
Some versions of the design were for a cooperative game, and others competitive.

The most important idea at this point was that the various NFC 'locations' visited could give the player contextual information about the other locations.

I decided to create the game so that it would run in a smart-phone's web browser, rather than building it as a mobile app in Unity.

Writing the game on the web would ensure it was instantly accessible by most devices, with no downloading required.

After some research, I determined that I should focus on learning the 'PHP: Hypertext Preprocessor' language, originally known as 'Personal Home Page' or PHP.



Using PHP

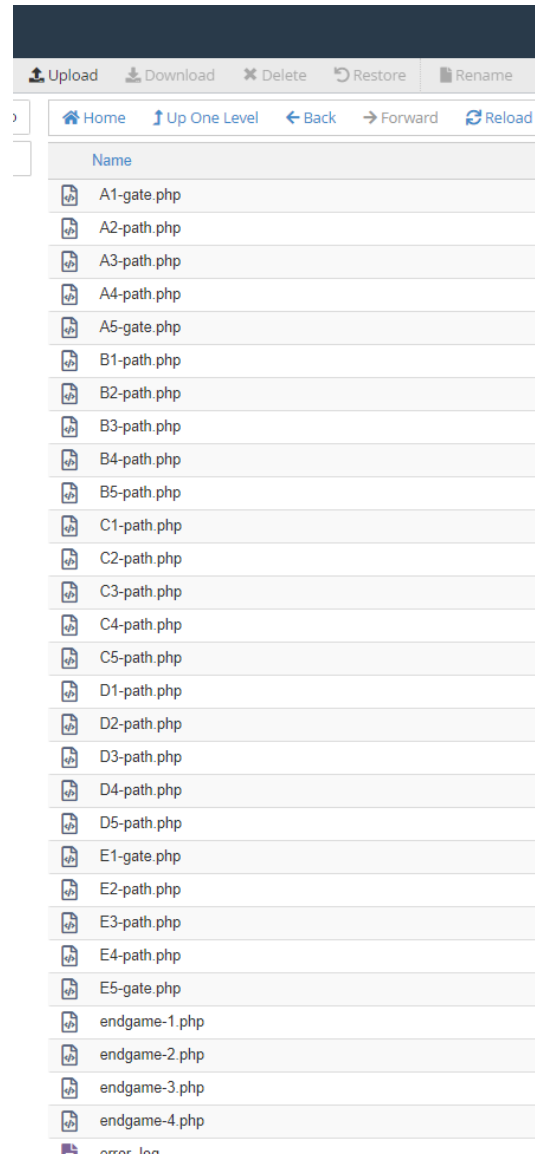
I found PHP surprisingly easy to pick up, and very similar to coding for a video game.

Once I figured out how to isolate variables to a particular user's web session, I was able to read and change those variables depending on the web pages visited by the user.

This effectively means that scanning different NFCs with your smartphone can have repercussions in the digital space. The act of moving the playing piece will alter variables within the game.

At the beginning of each user session, the goal variable (`$_SESSION['goal']`) is given a random value between 1 and 4. This value determines whether the goal is gate A1 (top left), A5 (top right), E1 (bottom left) or E5 (bottom right).

Each gate visited will send the user a clue associated with the relevant goal. All of these variables are set up in the home page, which is the first page visited.



```

1 <?php
2     session_start();
3 ?>
4 <html>
5 <head>
6 </head>
7 <body>
8     <font size="7">
9     <?php
10 //the session variable determines the end goal.
11     $_SESSION['goal'] = mt_rand(1,4);
12     $_SESSION['key'] = 0;
13     $_SESSION['pathset'] = mt_rand(0,2);
14     require('functions.php');
15     echo "<h1>YOU STARTED A NEW GAME!";
16     <br> </br>
17     Wait until your opponent is ready.
18     <br> </br>
19     On the count of 3, place your phone in the centre space of your playing
20     area and start playing!</h1>;
21     if ($_SESSION['goal'] == 1) //goal is TOPLEFT
22         {$_SESSION['clues'] = array("The goal has petals like a flower.",
23         "The goal has points like a star.", "The goal shines with a
24         harsh yellow.", "You found the key!");}
25     if ($_SESSION['goal'] == 2) //goal is TOPRIGHT
26         {$_SESSION['clues'] = array("The goal is round like a ball.", "The
27         goal shines with a harsh yellow.", "The goal glows with a dull
28         pink.", "You found the key!");}
29     if ($_SESSION['goal'] == 3) //goal is BOTTOMLEFT
30         {$_SESSION['clues'] = array("The goal has points like a star.", "The
31         goal glows with a dull pink.", "The goal has six sections.",
32         "You found the key!");}
33     if ($_SESSION['goal'] == 4) //goal is BOTTOMRIGHT
34         {$_SESSION['clues'] = array("The goal is round like a ball.", "The
35         goal has petals like a flower.", "The goal has six sections.",
36         "You found the key!");}
37     shuffle($_SESSION['clues']);
38 ?>
39 </font>
40 </body>
41 </html>

```

The code to start the game (home page)

```

<html>
<head>
</head>
<body>
    <font size="7">
    <?php
    echo "<h1>".$_SESSION['clues'][0]. "</h1>";
    if ($_SESSION['clues'][0] == "You found the key!") {$_SESSION['key'] = 1;}
    if ($_SESSION['pathset'] == 0) {echo "<h1>Move RIGHT or DOWN.</h1>";}
    if ($_SESSION['pathset'] == 1) {echo "<h1>Move RIGHT.</h1>";}
    if ($_SESSION['pathset'] == 2) {echo "<h1>Move RIGHT.</h1>";}
    $_SESSION['pathset'] = mt_rand(0,2);
    ?>
    <?php if($_SESSION['key'] == 1) :?>
    <input type="submit" onClick="location.href='endgame-1.php'" value="USE YOUR
    KEY TO OPEN THIS GATE" style="font-size : 40px; width: 100%; height:
    100px;">
    <?php endif; ?>
    </font>
</body>

```

The code at Gate A1 (top left)

```

index.php (PHP script text)

<?php
session_start();
?>
<html>
<head>
</head>
<body>
<?php
//the session variable determines the end goal.
$_SESSION['goal'] = mt_rand(1,4);
require('functions.php');
echo "<h1>index.php</h1>";
report();
if ($_SESSION['goal'] == 1) //goal is volcano
{$_SESSION['clues'] = array("WARM", "HIGH", "DRY");}
if ($_SESSION['goal'] == 2) //goal is lake
{$_SESSION['clues'] = array("WARM", "LOW", "WET");}
if ($_SESSION['goal'] == 3) //goal is forest
{$_SESSION['clues'] = array("COLD", "HIGH", "WET");}
if ($_SESSION['goal'] == 4) //goal is cave
{$_SESSION['clues'] = array("COLD", "LOW", "DRY");}
shuffle($_SESSION['clues']);
?>
<br><br>
<input type="button" onClick="location.href='volcano.php'"
value="Volcano">
</body>

```

```

volcano.php (PHP script text)

<?php
session_start();
?>
<html>
<head>
</head>
<body>
<?php
require('functions.php');
echo "<h1>Volcano</h1>";
report();
?>
<br> <br>
<?php
echo "The goal is " .$_SESSION['clues'][0]. " ";
?>
</body>
</html>

```

index.php

Goal = 3

omnfc.website/volcano.php

Not secure | omnfc.website/volcano.php

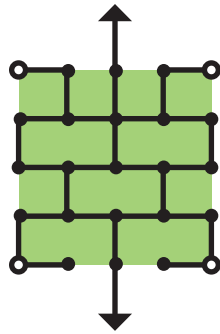
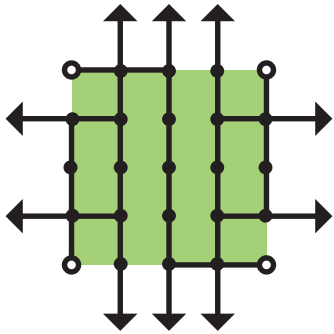
Apps

Volcano

Goal = 3

The goal is COLD

These early tests use PHP to link an array of clues (`$_SESSION['clues']`) to a particular goal (`$_SESSION['goal']`). The array is then shuffled so that the clues will always appear in a different order. You can see that a particular gate (here called 'Volcano') will always show the first clue in the array (`$_SESSION['clues'][0]`), whatever that may be.



Creating 'Pathsets'

Each NFC space on the board that isn't a goal is a path. Players navigate these paths by following the NFC instructions received by their phones.

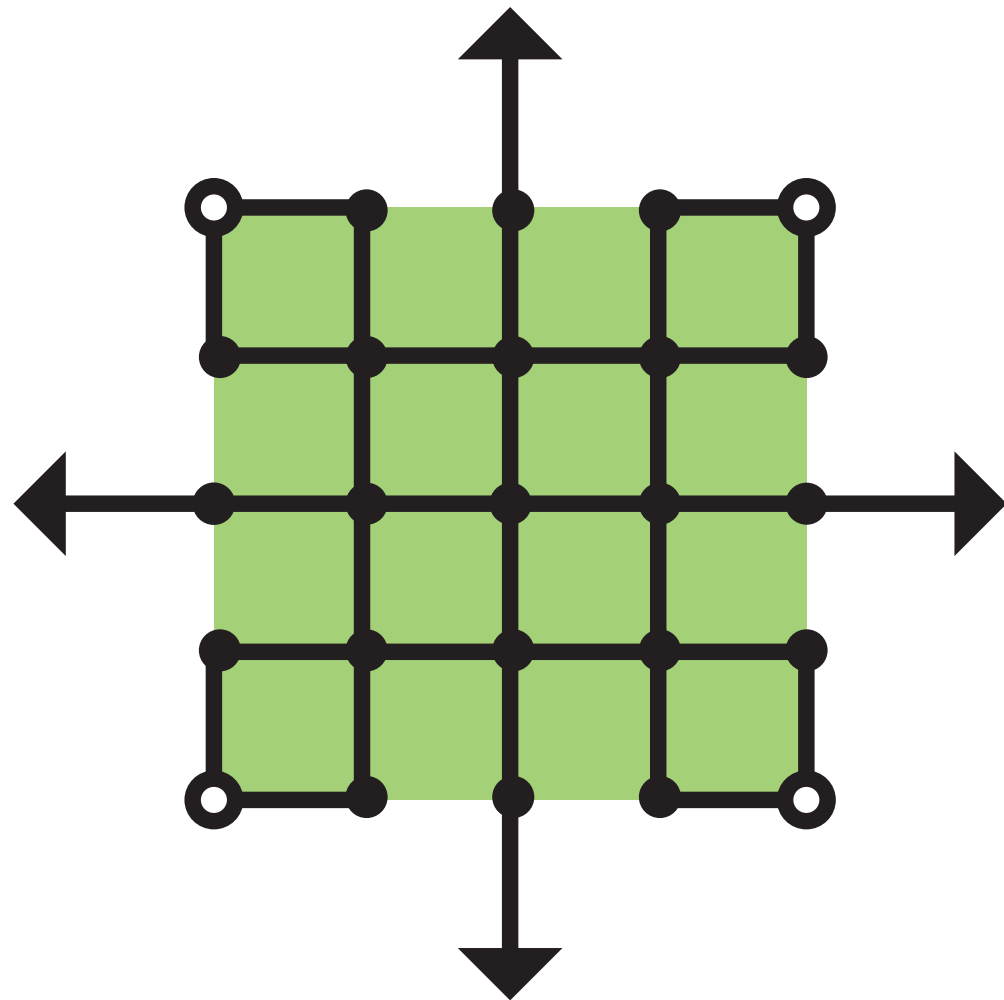
In addition to making the goal random in each different play-through, I decided the paths would also be randomized after each gate is visited.

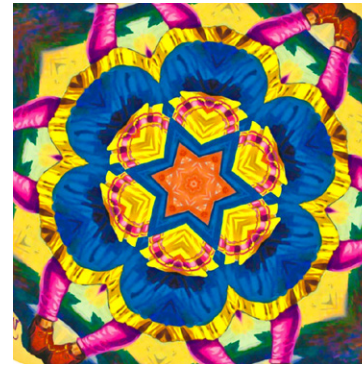
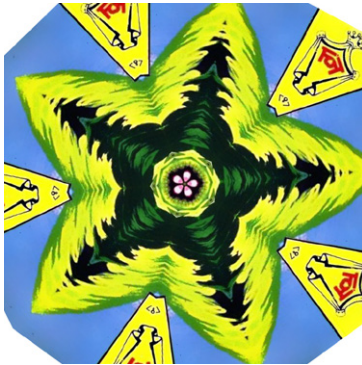
I mapped out several possible pathsets using the directions UP, DOWN, LEFT and RIGHT in various combinations.

Although the instructions received by the NFCs may at first seem random, players will quickly develop a sense of which particular pathset they are on after a few play-throughs.

This awareness enables players to plan their movements, taking the shortest route to their next gate while avoiding cul-de-sacs and diversions.

At the moment there are three possible pathsets programmed, but many more could be added.

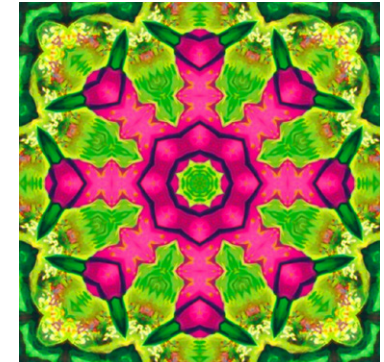
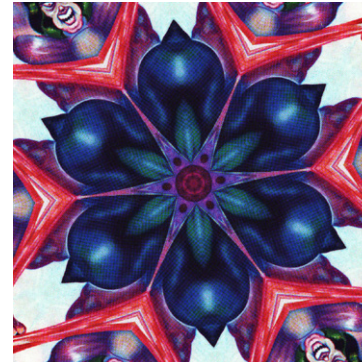
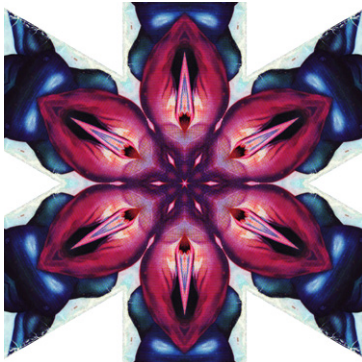




Art Assets

I originally planned to have each gate be associated with a particular location - a volcano, a lake, a castle, etc., each with a series of different associated variables like HOT/COLD, DRY/WET.

However, I realised this probably wouldn't look particularly striking. We've all seen a lake before. Instead, I decided to make the gates abstract, with purely visual qualities.



There are four different gates, each with three associated qualities. Each quality is shared with one other gate, so that a single clue will not be enough to reveal the goal immediately.

I used the program Sumopaint to generate my kaleidoscopes (based on colourful Hindi comic book covers) and edited them in Photoshop to give them their specific mix of qualities.

“So why NFC?”

The aspects of the game where NFC becomes valuable are in the giving of clues and the rearranging of paths.

Each gate visited is like a playing card which, when turned face-up, gives the player one clue about the end goal. However, since the goal will be different in each play-through, the clues will be different each time as well.

If this was all done with physical cards, the gates would have no way of ‘knowing’ where the goal was, and hence no way of knowing the appropriate clue to give the player.

Using NFCs, the game can be played on the board while hidden variables are managed in the digital space. The gates will automatically know which clue to provide.

As another example, the paths by which the players move between gates are altered whenever a gate is visited. This would be time-consuming and confusing to re-create with physical cards, but in the digital world it can be done instantly.

So the difference is that a paper card always contains the same information. Using NFC, the same ‘card’ can contain different information depending on any number of changing variables.

This enables us to create game mechanics which rely on changing contexts and interrelational structures.

The same is, of course, possible in a video game. Anyone could recreate *Zone of Gates* as a purely digital experience, the same way they could make a digital version of chess.

However, *Zone of Gates* does not have the feel of a computer game. It is highly tactile, requiring connections between physical objects as an analogy for ‘visiting’ the different gates and moving along the paths.

It is fast-paced and mentally engaging, with more thought required to deduce the goal and strategize the route to it than there is PHP code working behind the scenes.

It is also extremely simple to set up and play, with minimal barriers to entry. No app or program needs to be downloaded, and no card decks need to be shuffled or laid out.

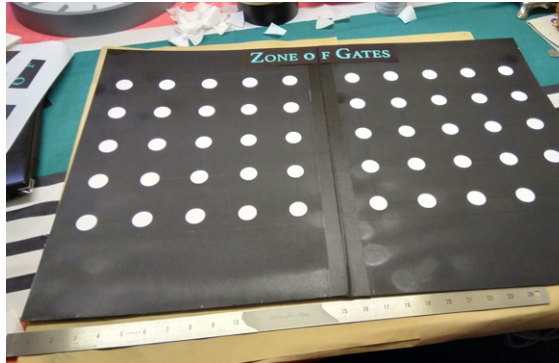
This quality of simplicity (or apparent simplicity) has been an important factor in my developing arts practice.

I have written in previous folios of my tendency to make works overly complex, assuming that this would make them appear clever.

I have now observed many times that complexity can be a barrier to engagement. Instead, a design appears clever when its function is simple.

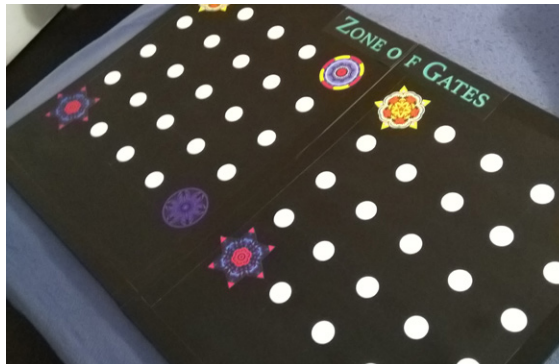
Assembling the board

The first board I made was too thin, and warped when I painted it black. I decided to make the board much more sturdy, using the backs of two art books edged and jointed with bookbinding tape.



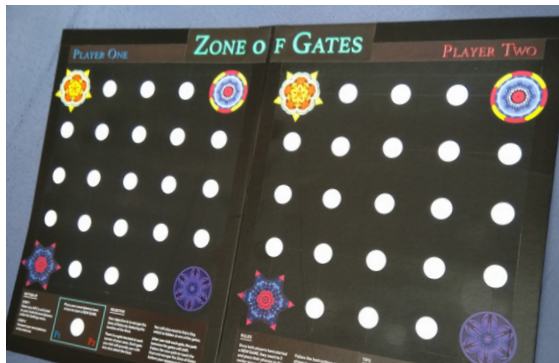
NFC placement

I drew up a grid on each board, leaving adequate distance between each point to avoid overlapping the ranges of the NFC stickers.



Final touches

I glued my four 'gate' images over the corner NFC stickers. Finally, I added the title and written instructions.



Things I would do differently next time

The best suggestion I received was to add an anti-cheating function, which would prevent players from making illegal moves while their opponent is focused on their own board.

I think this would be simple to make in PHP. An illegal move would send the user to a page that flashes red, or play a sound to attract the other player's attention.

As mentioned above, the act of moving the playing piece will alter variables within the game. What if every movement somehow altered the dynamics of the entire game? Just moving from point to point on the path would be like redistributing weight in a net, causing the net to alter its shape.

This proposition seems loaded with potential, for as far as I am aware there isn't much precedent for games which rely on internet servers to manage variables tied to real-world objects.

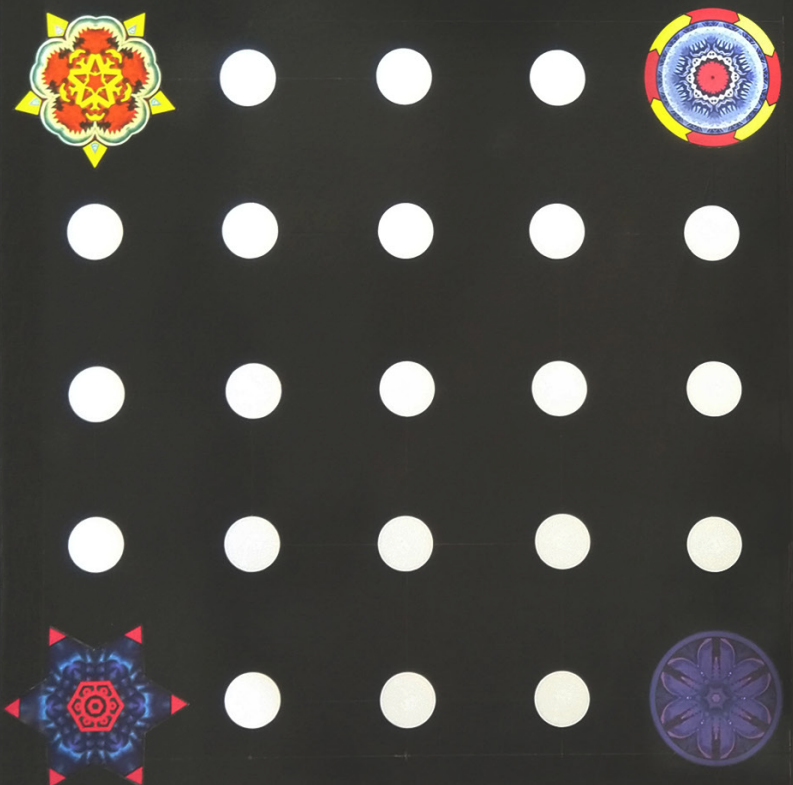
Major Project conclusion

Once I had planned out how my game would work, putting it together proved relatively painless. With just a few tutorials and some simple code, as well as the help of the cheap and plentiful NFC stickers, I was able to create quite a unique artefact.

This was an encouraging foray into the Internet of Things by a person with not much coding experience, and I would encourage any Makers or DIY enthusiasts to come up with similar games using these principles.

ZONE OF GATES

PLAYER ONE



SETTING UP

STEP 1
Make sure NFC is activated on your Android smartphones under the Settings menu.

STEP 2
Connect your smartphones to the internet.

Place your smartphones here in turn to start a NEW GAME.

P₁

P₂

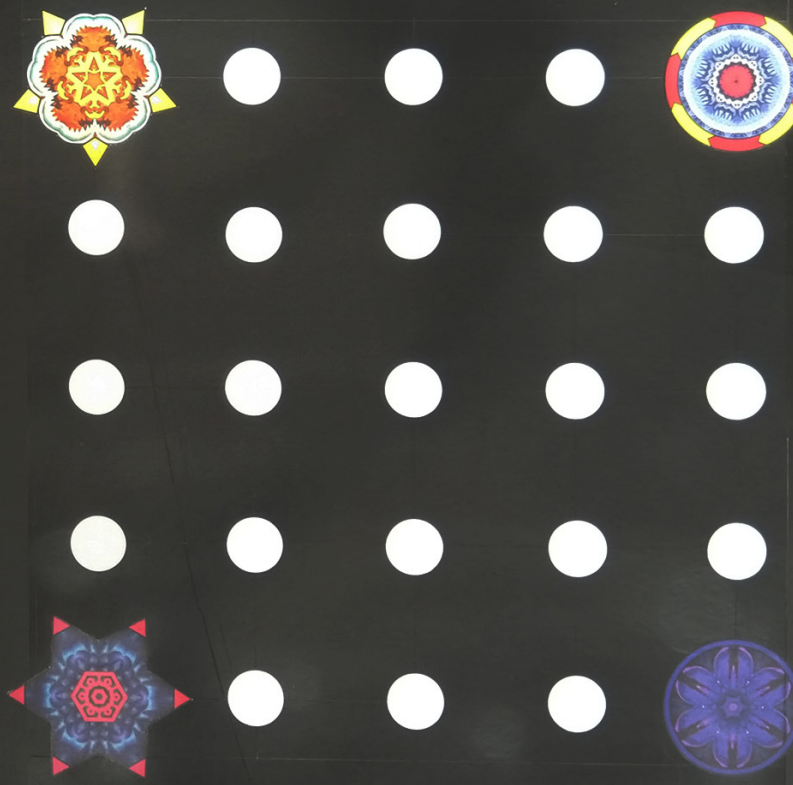
OBJECTIVE

Your objective is to escape the Zone of Gates by deducing the location of the Goal.

There is a gate located at each corner of your zone. Each gate you visit will provide you with one clue about the Goal.

You will also need to find a Key, which is hidden at one of the gates. After you visit each gate, the path between the gates will shift. Follow the new path to reach the Goal and escape the Zone of Gates before your opponent does!

PLAYER TWO



RULES

Once both players have started a NEW GAME, they count to 3 and place their phones on the center space of their zone at the exact same time.

Place your phones gently!

Follow the instructions on your phone to move it around the board, visiting each gate to gather clues.

Play occurs simultaneously until the game is won or lost. There are no turns, so be quick!

TIPS

Both players should keep to their own boards.

If, at any point, a move would take you outside of your zone, simply move to the same row on the opposite side the zone.

You will not be able to open any gate until you have found the Key. Then, a button will appear with the option to open the current gate.

If you try to use the Key on a gate that is not the Goal, the Key will break and you will lose the game!