Fanmade

2D Puzzle Platformer

Blake Farrugia
Mohammad Rahmani
Nicholas Smith
CIS 487
11/1/2010

1.0 Game Overview

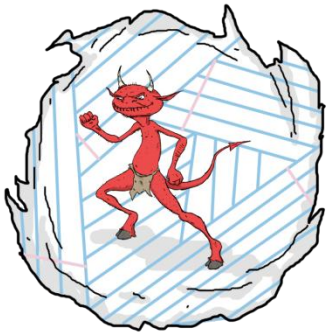
Fanmade is a 2D puzzle platformer created by Blake Farrugia, Mohammad Rahmani, and Nicholas Smith for CIS 487. Art assets were designed and created by Eric Hasiak and Jordan West, colleagues of the team from a previous game design project who attend the College of Creative Studies. The game was developed using C++ and DirectX in the Visual Studio 2010 environment.

1.1 Story

- A piece of Karta-brand paper, eager to have the next big idea drawn to it, is doodled on by a Grad student in class. The paper is carelessly tossed to the trash, where it misses and falls into the ventilation system.
- "Karta" is heartbroken by this. The doodle, a loose representation of a demon, takes on a life of its own and wishes freedom from the dank ventilation system so it may roam freely in the world once more. The demon cannot free itself from the paper it is tied to, though.
- The demon speaks to Karta and calls itself "Ikrah". Ikrah wants to use the paper's lighter-than-air properties to escape. By conjuring whirling fans to help move the paper through the ventilation system, Ikrah will have a greater chance at freedom.
- Fresh air and green trees do not interest Karta though, and fail to lull him from his depression. In order to gain Karta's cooperation, for helping Ikrah escape the school, it promises to grant Karta's wish of being the "next big thing."

1.2 Characters

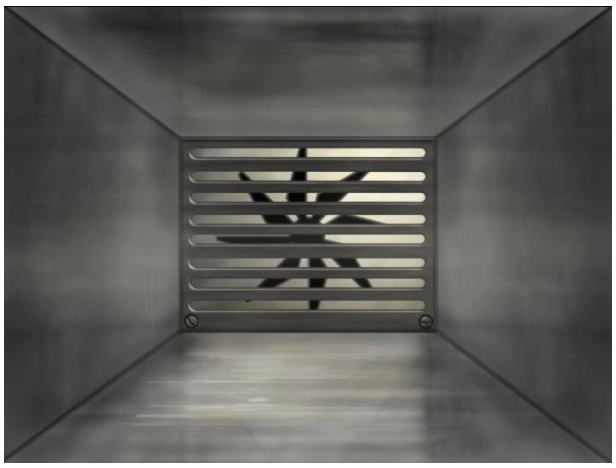
- Ikrah - A paper doodle that resembles a demon. He claims to be over 5,000 years old. Ikrah enjoys quipping about the situation and pointing out the weaknesses of others, though he completely misses the irony that he is chained to Karta for movement.



Karta - A piece of Karta-brand paper that dreams of a worthy idea being written on it. Karta uses its lighter-than-air qualities to help Ikrah, the demon doodle scrawled onto it, but doesn't buy into his claims of granting wishes. Usually silent, though able to speak, it rarely goes along with Ikrah's jabbing, talkative nature.

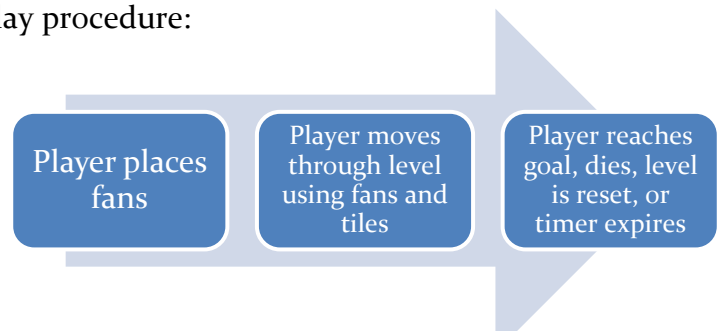
1.3 Setting

- School ventilation system – Dark, gray, and covered in what can only be described as "dust pillows," the school ventilation system does not grant easy access to our somewhat-dynamic duo.



1.4 Game Play Details

- Tile-  Fan-  Goal- 
- Player must complete each level by placing fans with the mouse and moving through the level with the keyboard
- Fans will be placed freely across the level in legal, unblocked positions
- Fans can be placed with different orientations and power
- Player strategy evolves based on puzzle/ level layout
- You will achieve more points for using less fans and beating levels in less time
- More difficult levels provide a better opportunity for more points.
- Game play procedure:



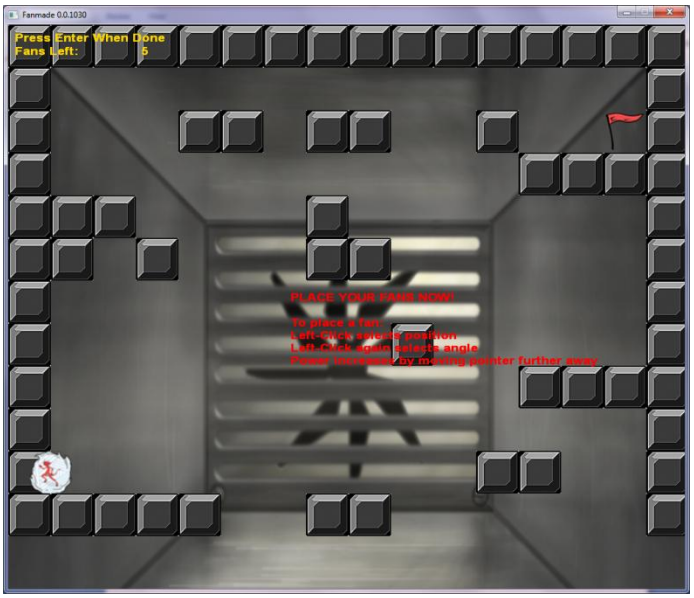
1.5 Input/ Controls

- DirectInput
  - A- Move left
  - D- Move right
  - Space/ W- Jump

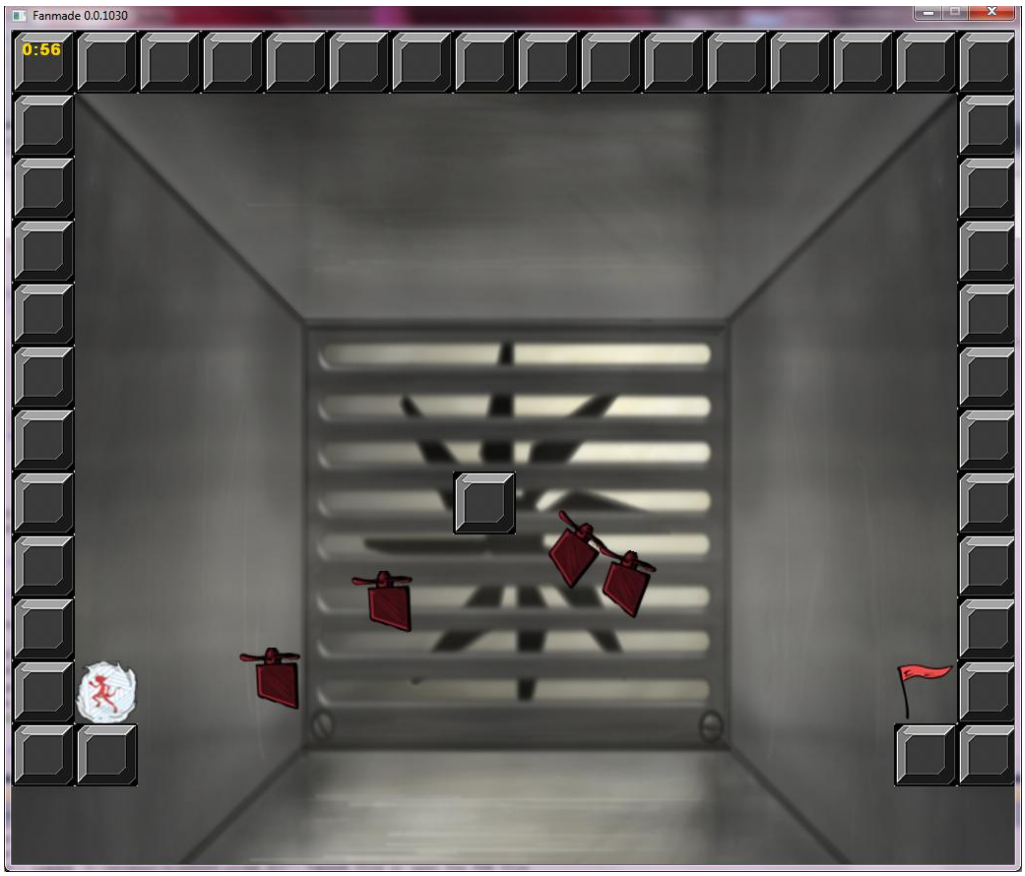
- Mouse- Fan placement
  - Left-click to place fan
  - Left-click again to select angle and power relative to mouse distance

1.6 User Interface Design

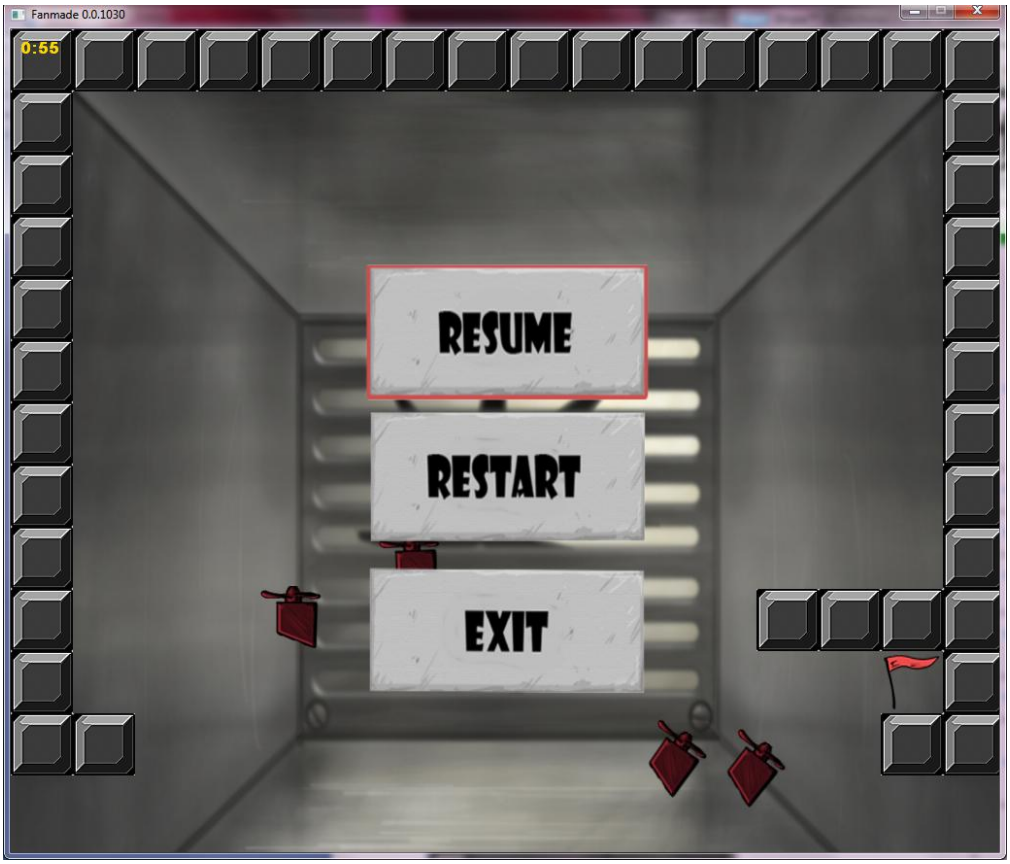
- Start Menu- displays at the start of game and gives options of Start and Exit



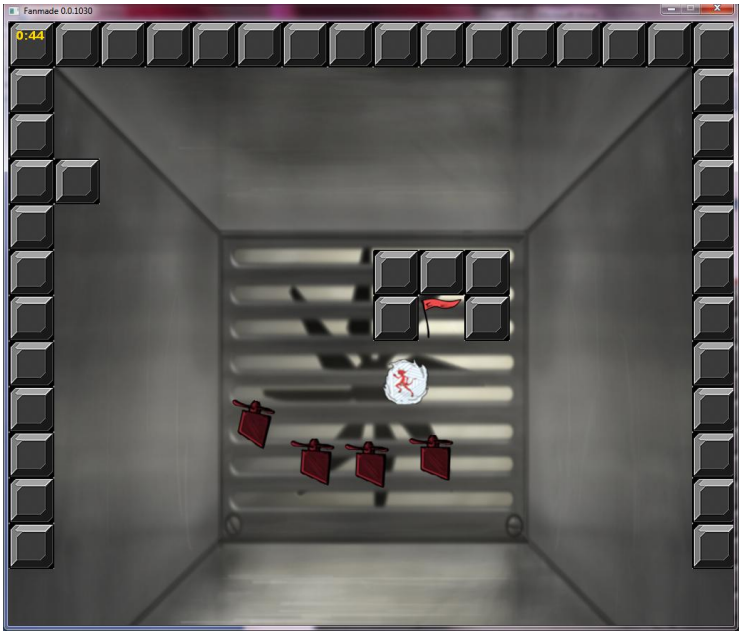
- In Game- displays timer and main game components/loop

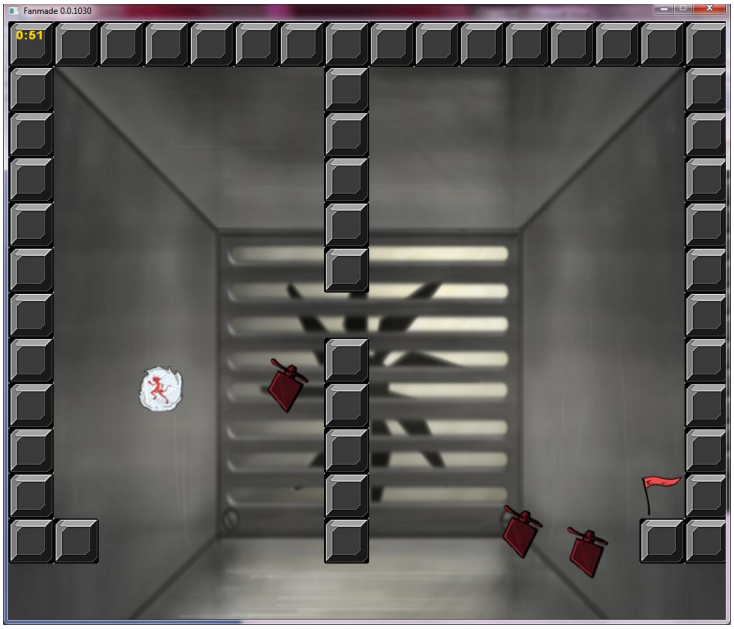


- Pause Menu- displays when player strikes the escape key ingame. Gives three options of Resume, Restart, and Exit



1.7 Levels and Level Creation





- Levels contain tiles, fans and a goal
- Three difficulties of Easy, Medium, and Hard, with chief differences being the number of tiles and fans used as well as positions for both
- Gradual difficulty increase for each level
- Grid system for levels
- Scripts (.txt) will generate levels
- Levels were created by Nicholas Smith, Blake Farrugia, Mohammad Rahmani, and Michael Stamboulian (College for Creative Studies)

1.8 Scoring

- Score is based on a mathematical equation that takes into account time remaining, number of fans used, and level difficulty.
- Equation: (points per sec * time remaining) * ((fan multiplier - num of fans used)/ level difficulty)
    - points per sec (constant)- 3
    - fan multiplier (decreases with num fans used)- 10
    - time remaining- time left on timer
    - num of fans used- number of fans placed
    - level difficulty- 1: Hard, 2: Medium, 3: Easy

1.9 Timer

- The timer will start a countdown at the beginning of the level. When the timer reaches zero, the level will end. If the player has reached the goal before the timer ends, the level is complete.
- When the level is complete, the time remaining is used to calculate the level's score

2.0 Obstacles

- Enemy Fans will be placed in levels to affect the player's strategy blocking paths, manipulating routes, and shifting players position.
- Tiles may be placed in a hard to solve manner (example: player has to squeeze through one tile space) to increase difficulty

2.1 Triggers

- Fan Effect Area- When player enters this trigger area, the fan will exert a force on the player.
- Victory Flag - When player enters this trigger area, the level will be completed.

2.2 Graphics

- DirectX
- 1024 x 768 windowed screen
- 16x12 Map grid (number of tiles)
- 64x64 pixel Tiles
- Tile-based system
- Each class contains a Render() function for drawing to the screen.

2.4 Physics

- Gravitational Force- When the player is in the air, the force of gravity will act on the player. This will cause the player to fall to their death, fall into a fan force, or land on a tile. The gravitational force will always be acting straight down with a constant magnitude.

- Fan Forces – Fans use a rotated rectangular bounding box to determine if the player is within effective range of the fan. Fans produce force on the player proportionally to the inverse of the square of the distance between the player and the fan. These forces are represented in the game by two-dimensional vectors. The larger the x and y values of the vectors, the more force is applied in the respective direction. To prevent a single fan from being able to blow the player excessively far, there is a cap on the amount of power the fan can produce.

- Friction – Air and ground friction are both present in the game. In the air, the player loses 2% of its velocity with every player position update; on the ground, that rises to 25%. This allows the player to drift from side to side as needed over the fans, while still being able to stop quickly upon landing on a tile.

2.5 Collision

- Box Collision - Tiles will use box collision, so that the player can run across and interact with the level. The fans will use rotated box collision. Given that the fans can be set up in any arbitrary direction, they utilize rotated boxes to determine their area of effect.
- Screen Bounds - Screen bounds use box collision. If the player, is not within the bounds (out of the background/ level) the player is dead.

2.6 State Management

- Game states
    - StartMenu
    - InGame
    - FanPlace
    - PauseMenu
    - LevelComplete
    - Death
    - EndGame
    - Exit
- Game states are declared as a C++ enumeration
- Game state transition is handled by switch statements within main

2.7 Programming

- Nicholas Smith, University of Michigan- Dearborn, Computer Information Science, Senior
- Blake Farrugia, University of Michigan- Dearborn, Computer Information Science, Senior
- Mohammad Rahmani, University of Michigan- Dearborn, Computer Information Science, Senior

2.8 Art

- Eric Hasiak- College for Creative Studies, Entertainment Arts, Senior
- Jordan West- College for Creative Studies, Entertainment Arts, Senior

2.9 Software

- Microsoft Windows® operating system
- C++ (http://www2.research.att.com/~bs/homepage.html)
- Microsoft DirectX® (http://www.microsoft.com/games/en-us/aboutgfw/pages/directx.aspx)
- TortoiseSVN ®
- Google Docs®
- Adobe Photoshop®

3.0 Hardware

- Mouse and keyboard
- Mild System Requirements

3.1 References

- Direct X 9 Programming Guides
- Tricks of the Windows Game Programming Gurus
- XNA Platformer Starter Kit
- Internet (Blake's side only)

3.2 Future Work

- Drag and drop level editor
- Larger, scrolling levels
- User created levels
- More art content including animation
- Ports (i.e. I Phone, android, face book, Mac, Xbox 360, directly interface to human brain for full game immersion!!!)
- Immerse story into game play

- Dialog
- Score bonus power ups
- More sound and better sound quality
- Shifting level design for multiple goals