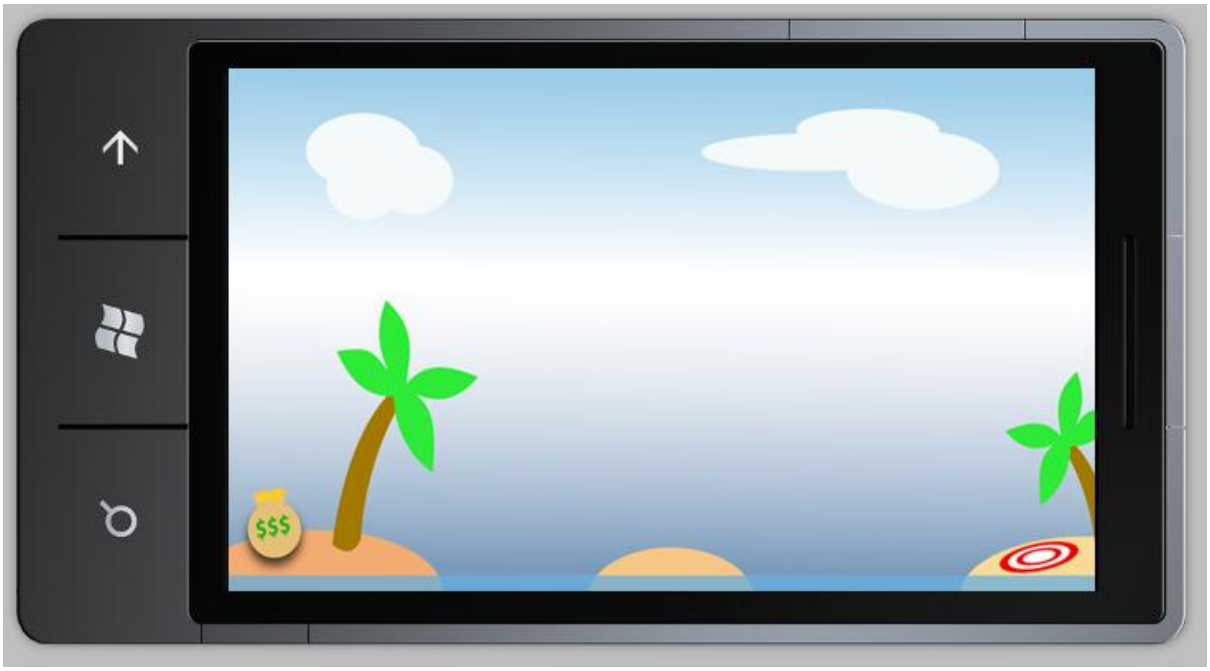
GREEDY BANKER

DESIGN DOCUMENT
VERSION 0.4
JUNE 7, 2011



CREATED BY TEAM FATCAT
DARREN KORMAN
TONY MARTINEZ
GARET ANDERSON

INDEX

VERSION HISTORY

Version 0.5 – June 7, 2011
Updated class diagram, screenshots.

Version 0.4 – May 26, 2011
Updated class diagram and object/component section. Added more art.

Version 0.3 – May 12, 2011
Third submission of doc. Adding screenshots for sense of scale and 'look and feel'; added newly discussed concepts, objects and other team decisions.

Version 0.2 – May 5, 2011
Second submission of doc. Folding in all design notes for week of Apr 28 – May 5.

Version 0.2 – April 28, 2011
First submission of doc to Kelvin.

Version 0.1 – April 27, 2011
- First version of the document.

GAME OVERVIEW

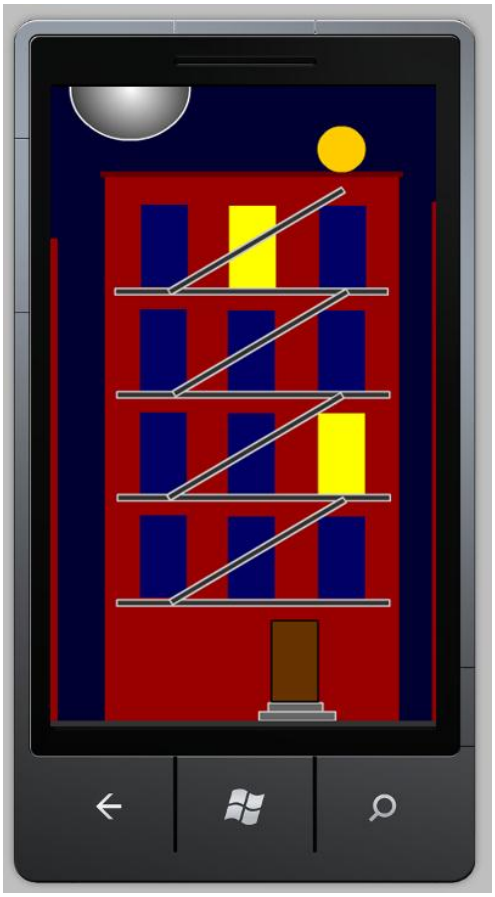
Game logline

Player is a Ultra-rich Tycoon who wants to get their enormous pile of money to a "Safe Haven".

Gameplay synopsis

This game is a 2-d side perspective, level-based, interactive puzzle game.

For each level, the player begins with a "pile of money" in a starting location in the level. The goal for the player is to bring their money to a target location in the level.



To accomplish this goal, the player is given a collection of "components" that can be assembled into a "machine" that can transport money. The general idea is that these machines will be in the style of a Rube Goldberg machine. Wikipedia provides a good definition which is:

A Rube Goldberg machine, device, or apparatus is a deliberately over-engineered machine that performs a very simple task in a very complex fashion.

By giving the player a set of "machine-components" in each level, this allows our player to develop their own creative solutions to overcome the challenges of each level.

[Describe how your game plays and what the user experiences. Ideas to discuss: Uniqueness, Mechanics, Setting, Look and Feel.]

GAME DETAILS

Description
Player is an Evil Banker with a horde of money that they want to get to a "safe" location. To move the money, the player will have to build a level crossing "machine" which will bring money to the "safe" location.

Game genre?
Puzzle

Is this single-player or multiplayer game?
Single

Is this 2D or 3D?
2d Side view

Where does the game take place? (world)
Cartoony and humorous caricature of modern world. Level locations include a back-alley of an Urban city, A Laundromat, Caribbean island, In orbit above the Moon. Each level type will have its own unique challenge.

What do I control? How many characters? Can I use vehicles?
Player controls the assembly and activation of a transport system which moves "money" through the level. There is not a "hero" on the screen that can be controlled by the player.

What is the main focus?
Get at least 50% of your money to the "safe" point of the level. Getting more than 50% of your money to the safe point results in a higher level score.

How long the game lasts? Or how long game matches last?
1-3 minutes for early simple levels. More complex problems may take 3-5 minutes per level.

Comparison
The Incredible Machine (circa 1995) – This machine building game was largely focused on mechanics without a perceivable plot for continuity; ours will be tied together using

an economic theme with money centric goals. Ours will be built for mobile gaming instead of Windows 95.

~~What is unique?~~

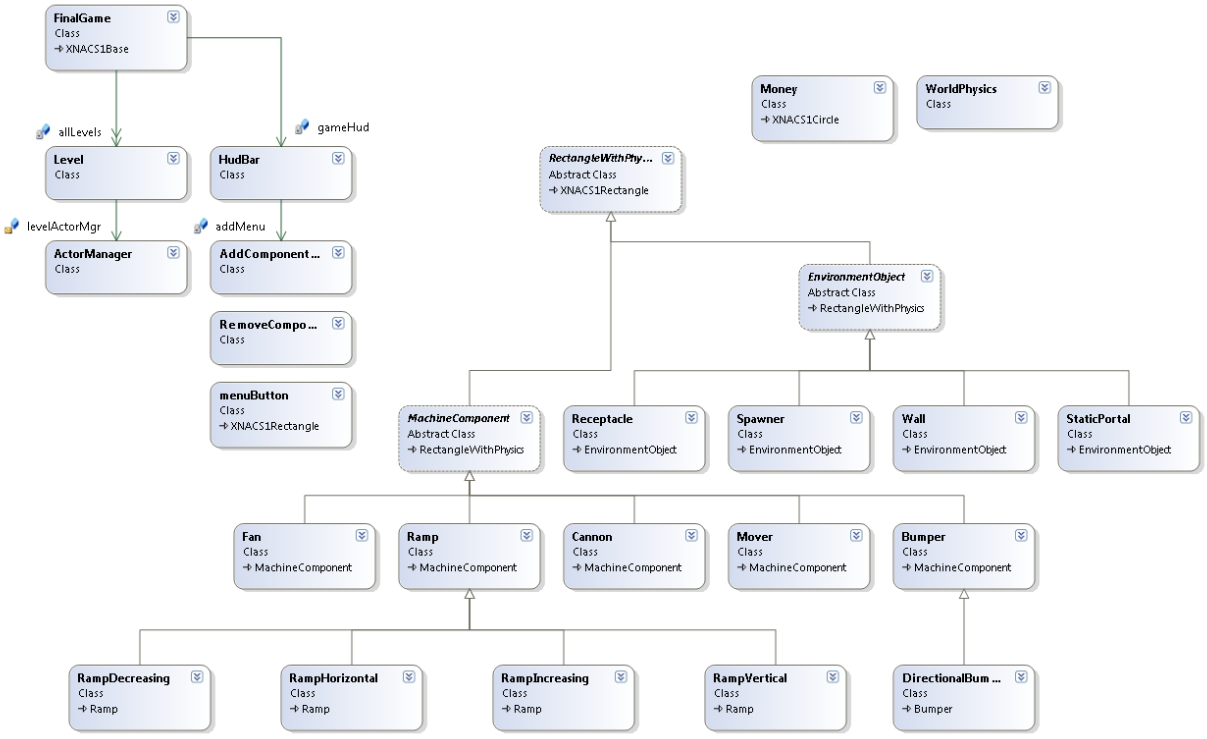
[what has never been seen in any other game before this]

Why create this game?

We wanted to challenge ourselves with a type of game somewhat different than what had been discussed in class.

As developers, we like the idea of giving our player the feeling that they have creatively solved a puzzle presented in a level.

Class Design



AUDIENCE, PLATFORM, AND MARKETING

Target Audience
All ages who use Windows Phone 7. Anybody who likes to build solutions and solve puzzles.

Platform
Windows Phone 7

Top Performers
Can't find any Rube Goldberg style games in WP7 app store. Top games are:

- Super Monkey Ball – Uses accelerometer to roll a ball hero down tracks to goals
- Need For Speed – 3d car racing
- Harbor Master – manipulate boats in a harbor

Feature comparison
Unaware of any competing mobile phone games where the user builds machines to accomplish goals.

FEATURE SET

General features

Multiplayer features
No direct player vs. Player planned. Could have global scoreboards.

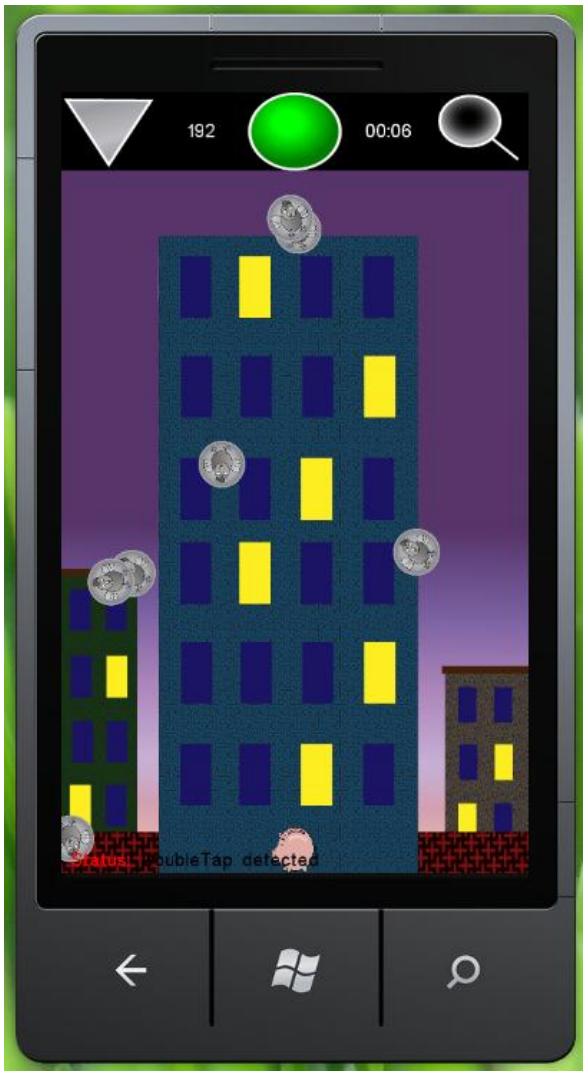
Level Editor
Nice to have for project development needs. Not released for public use.

Gameplay
User will have access to various components with which they must accomplish some stated goal. Early stages will serve as introductions to the gameplay and components available.

For example, user may need to move their bag of money from one land mass across some water way to a tropical island goal. They have at their disposal various parts including lifts, ramps, trampolines or similar item. The user places their components and when satisfied starts their machine which will either move their money successfully or may fail to do so at which time the user must reset and reconfigure their machine. Some components or interactions may also cause some money to be lost which could affect success, failure or the final score.

THE GAME WORLD

Overview
2d side view with possible zooming and scrolling where needed.



Key locations (within each level)
Start location for the money

A 'goal' area where the money should be delivered

Hazardous zones and traps may need to be avoided

Travel
The user will not directly manipulate their money, instead they must use the available components to build a machine which will move the money for them.

User can move the camera to change their view of the level.

Scale
"Normal" scaled levels with overall size varying based on level design. Start at 100yards in width and later levels might be 500-1000 yards in width.

Objects
Collide-able obstacles which the player must direct their money around.

Weather
Clear. Might be some animated weather in the backgrounds of levels but nothing that would impact gameplay.

Day and night
Levels can take place at any "time". However, in a particular level, time will NOT transition from day->night or night->day.

Time

Water

Other elements

CAMERA

Overview

Camera will have 2 different levels of zoom. Configurable or automatically adjusts based on the size of the level map.

A full-level view, which will pull the camera out far enough so the bounds of the level can be completely seen on the screen. In this view, the player cannot add/remove/modify any placed machine objects.

A "zoomed-in" view will bring the camera in so that only a portion of the game level is visible. In this view, the player will be able to add/remove/modify a machine object to the world. While "zoomed-in" the player can still scroll around the world by swiping.

GAME CHARACTERS

Overview
There is only one character which is the player who is a "rich Tycoon".

This character may or may not even be shown to the player (depending on time for art assets).

~~Character creation~~
None.

Enemies and monsters
Environmental obstacles (such as walls).

Probably nothing animated.

USER INTERFACE

Overview

We intend to write this for WP7 and will use conventional controls for the mobile platform.

There will be static information at the top of the screen to inform the player about their progress. Information displayed will include:

- Money Remaining ($ value)
- Time elapsed in level
- Money "Saved" ($ value)

Details

We currently expect to see:

- Double tap for zoom in/out
- Swipe to scroll around board when zoomed in
- Zoom is toggled by a button on the HUD
- Touch and drag machine components around the screen.
- Double tap on a component to bring up a "remove component" menu.
- Double tap on an empty are of game world to bring up the "add component" menu.

Single tap on inventory button (on HUD) to bring up "add component" menu

- Perhaps using rapid flicks to charge certain components or other gestures on machine components to cause some action

OBJECTS

Overview

The main gameplay of the game is to construct a "machine" which can transport the player's money to the "safe point". To do this, for each level, the player will be given a set of machine components they can use as they see fit to build their machine.

Machine Components

Cannon (shoots money, ~~but shoot too hard and might lose some $$$~~ can redirect money; loads like a hopper, fires when hopper full?; might fire money in multiple arcs)
~~Trampoline~~ Bumper (money bounces)
Fan (pushes money or other objects)
Lift (raises money vertically in world)
Balloon (floats in air with money attached)
Ramp (money rolls across ramps)
Repulsor (pushes money away in all directions)
Catcher? (a net that "catches" money that is shooting by)
Gravity Control? (accelerometer/tilt influenced)
Zip-line? (money travels down length of line)
Turtle Raft? (money floats on the back of a raft)
Portals (speedy thing goes in, speedy thing comes out)
Coin Spawning devices (A dude shaking his piggybank)
Goal objects or areas (A piggybank)

Detailed descriptions of Game classes and objects are kept on our project wiki. Here is an example:

Fan

Fan is a MachineComponent used to blow Money around. The fan is directional, blowing only along its FrontDirection.

Members

Constructors

public Fan(Vector2 center, float width, float height, float rotation = 0f, float static_friction = 0.1f, float bounciness = 0.6f) -- Sets basic stats and set its ComponentType to fan.

Methods

public override void Update(Vector2 input_pos, XNACS1Primitive obj = null) -- Overrides the base Update function to allow extra steps for "blowing" a circle object.

protected void Blow(XNACS1Circle ball) -- Pushes Money around when in the Fan's area of effect.

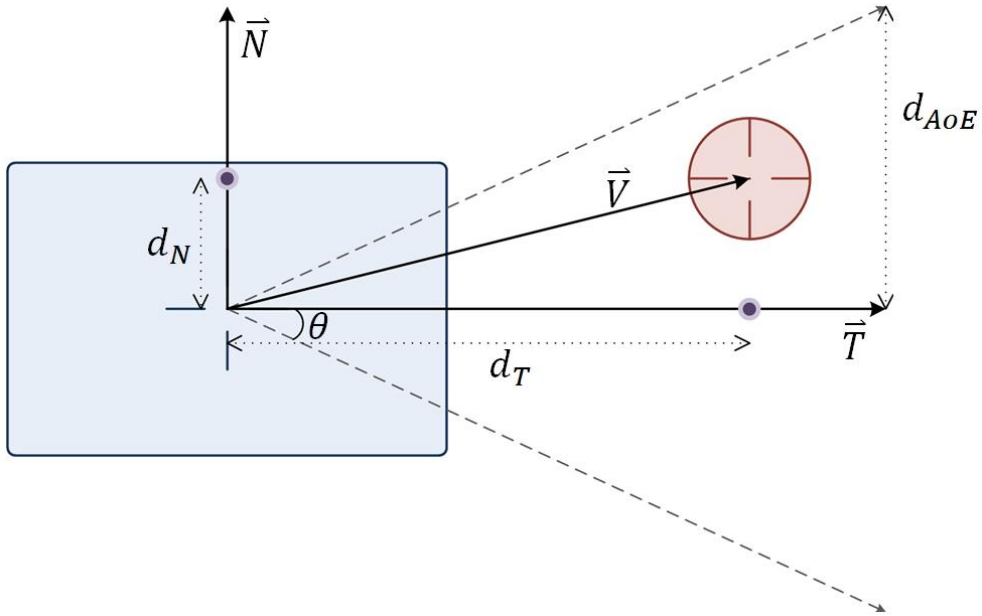
Constants and Enumerations

protected const float kTheta = 50f -- Area of Effect for fan; theta

protected const float kWindForce = 2f -- Base "force" the fan exerts

Mechanics

The Fan is a directional component with a triangular area of effect (AoE). If a circle's center is found to be within the AoE, we push it away in proportion to its distance from the Fan. We first determine whether the circle is within the AoE by checking that the circle's center is lined up with the Fan's tangent direction (since it blows in only *one* direction). $\vec{V} \cdot \vec{T} = d_T$. If the sign of d_T points us in the $\vec{T}$ direction, we check to see if it falls into one of the right triangles formed by $\vec{T}$ and θ. First, calculate $d_{AoE} = \|d_T \hat{T}\| \times \sin\theta$. We then check to see if $|d_N| \leq d_{AoE}$. If so, the circle is in range of the Fan's AoE so we can push it away with the Fan's acceleration (a_F). The circle gets pushed by $a_F \times \|\vec{V}\|^{-1}$ in the $\vec{T}$ direction, scaling a_F by the multiplicative inverse of the length of $\vec{V}$. Given that I am inept when it comes to math, there are likely far better ways to do this.


$\vec{N}$
d_{AoE}
$\vec{V}$
d_N
θ
$\vec{T}$
d_T

MUSIC AND SOUNDS EFFECTS

Overview
Cartoony music and effects.

Details
To be determined…

Music tracks
To be determined…

Sound effects
Coin impacts (“tinks”). Cannon Sounds. Some sort of “tally” sound when coins reach the safe area.

3D sound
None.

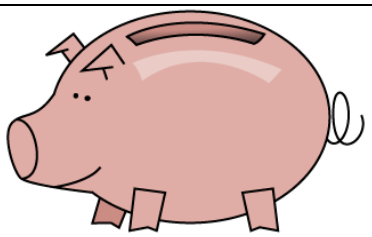
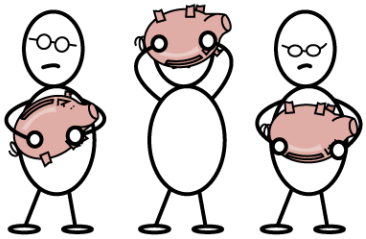
Other

ART

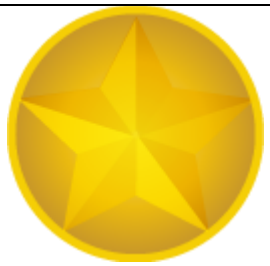
What kind of style will be used in the game?
Cartoon/Humorous

Scenery models
Billboards in background. Show silly art (or possibly real advertisements?)

Component Art

Piggy Bank The "safe-location" where the player is trying to get their money to.	
Spawner This clearly miffed fellow shakes coins out of an overturned piggy bank. This is where coins are first spawned.	
Ramp The most basic component, ramps are useful for herding coins and guiding them to the piggy bank. See *ramp styles* for variations.	
Fan This handy object blows coins around. Useful for getting a boost of speed or traversing vertical challenges.	

“Money” Art

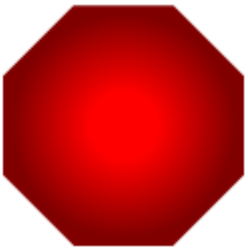
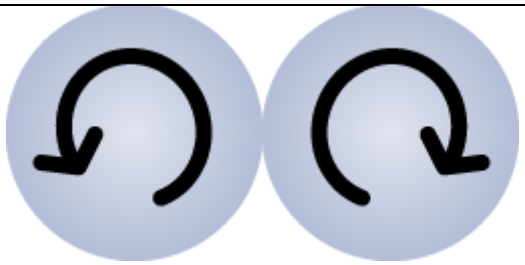
Copper Coin A small value of money.	
Silver Coin A larger value of money.	
Gold Coin The largest value of money.	

Ramp Styles

Wood Ramp	
Metal Ramp	

Particle effects

HUD

Inventory Tapping this button opens up your available inventory of components	
Go button Tapping this button starts the coins falling	
Stop button Tapping this button stops coins from falling	
Cancel button Tapping this button exits out of the inventory menu	
Rotate buttons Tapping these buttons rotates the selected component.	

SINGLE-PLAYER GAME

Overview
A series of puzzle levels with increasingly difficult challenges.

To start a level, the user will select a level from the "level select" screen. To start with, later levels will be "locked". To unlock a level, the user must complete the previous level.

Details
Challenge Level – A "randomly" generated level which is used for testing game objects.

Level 1 – Urban Back Alley. Player is trying to get cash down to a waiting escape vehicle.

Level 2 – Money Laundering. Player is trying to "clean" their cash by sending it through washing machines in a Laundromat.

Level 3 – Tax Evasion. Player is trying to get their cash to a "safe haven" on a distant Caribbean island.

Level 4 – Cash In Space. Player is trying to get their cash to a safe location in outer space. Possibly changing gravity will be a challenge in this level.

Story
Player is a rich business Tycoon who wants to get his money to a safe tax-free haven.

Victory conditions
Get a certain minimum percentage (50%) of your money to the "safe" goal area in a level.

MULTIPLAYER GAME

Overview
Not currently planned.

Max players
N/A

Servers
N/A

Customization
N/A

Is the world persistent or not
N/A

Saving and loading
N/A

Scores
N/A

Chatting
N/A

Clans
N/A

RESOURCES & LINKS

Links to related resources
Art for game is being done by group member Darren Korman

Design document based on this template:

~~HTTP://FORUMS.XNA.COM/FORUMS/T/229.ASPX~~

~~HTTP://INDIEPATH.COM/PUBLIC/DESIGNDOCUMENTTEMPLATE01.DOC~~

A few sections of the document are based on the design template in this book:

Fullerton, Tracy. Game Design Workshop, 2nd Edition: Elesvier Inc, 2008.