|  |  |
| --- | --- |
| CaptainCPA Documentation  Final Assignment | Abstract  Simple platformer game created for a Final Assignment that implements basic and advanced XNA functionality, including animations, physics, and more.  Kendall Roth, Doug Epp  Object Oriented Game Programming |

Captain CPA Documentation

Table of Contents

[Components 1](#_Toc437935179)

[Display String 1](#_Toc437935180)

[FixedTile 1](#_Toc437935181)

[HealthDisplay 1](#_Toc437935182)

[HighScoreComponent 1](#_Toc437935183)

[MenuComponent 1](#_Toc437935184)

[MoveableTile 1](#_Toc437935185)

[Pickup 1](#_Toc437935186)

[ScoreDisplay 1](#_Toc437935187)

[Tile 1](#_Toc437935188)

[Managers 1](#_Toc437935189)

[CharacterCollisionManager 1](#_Toc437935190)

[CharacterStateManager 1](#_Toc437935191)

[CollisionManager 2](#_Toc437935192)

[PhysicsManager 2](#_Toc437935193)

[TileCollisionPositioningManager 2](#_Toc437935194)

[Scenes 2](#_Toc437935195)

[AboutScene 2](#_Toc437935196)

[ActionScene 2](#_Toc437935197)

[GameOverMenuScene 2](#_Toc437935198)

[GameScene 2](#_Toc437935199)

[HelpScene 2](#_Toc437935200)

[HighScoreScene 2](#_Toc437935201)

[HowToPlayScene 2](#_Toc437935202)

[LevelOverScene 2](#_Toc437935203)

[LevelSelectScene 2](#_Toc437935204)

[PauseMenuScene 2](#_Toc437935205)

[StartScene 2](#_Toc437935206)

[Tiles 2](#_Toc437935207)

[Block 2](#_Toc437935208)

[Boulder 3](#_Toc437935209)

[Character 3](#_Toc437935210)

[Disc 3](#_Toc437935211)

[Enemy 3](#_Toc437935212)

[Exit 3](#_Toc437935213)

[FloppyDisc 3](#_Toc437935214)

[Gem 3](#_Toc437935215)

[LevelBarrier 3](#_Toc437935216)

[LevelEnd 3](#_Toc437935217)

[Mimic 3](#_Toc437935218)

[Monstar 3](#_Toc437935219)

[MovingPlatform 3](#_Toc437935220)

[Platform 3](#_Toc437935221)

[PursuingEnemy 3](#_Toc437935222)

[Spike 3](#_Toc437935223)

[Utilities 4](#_Toc437935224)

[HighScore 4](#_Toc437935225)

[ColorConverter 4](#_Toc437935226)

[Directions 4](#_Toc437935227)

[LevelLoader 4](#_Toc437935228)

[Settings 4](#_Toc437935229)

[TileType 4](#_Toc437935230)

[Utilities 4](#_Toc437935231)

[Class Diagram 5](#_Toc437935232)

[Images 6](#_Toc437935233)

[Code 6](#_Toc437935234)

# Description and Features

## Description

Captian CPA is a platformer game created in XNA Game Studio by two second year Conestoga Computer Programmer/Analyst students for a final project in Object Orientated Game Programming. The in-game description reads as follows:

You are Captain CPA, a renowned hero whose strength, speed, and jumping ability are known the world over! Traverse a hostile world, collect discs, and avoid enemies to reach your goal!

## Features

Features include 2D Graphics, a full menu system, basic movement animations, high score system, level progression, side-scrolilng, physics and artificial intelligence, and game saving and loading from XML files.

# Class Namespaces

## Components

### Display String

Displays a message (string) to the player and is used in the menu systems and player alerts. Contains methods to update the bounds of the string and draw it to the screen.

### FixedTile

Base class for fixed tiles that inherits from Tile and includes additional properties to determine the X and Y coordinates of the tile in relation to the tile map list.

### HealthDisplay

Inherits from DrawableGameComponent and displays the player’s remaining lives in the top centre of the ActionScene

### HighScoreComponent

Displays player high scores and allows the player to enter their name if they get a high score. Contains methods for retrieving player input (name) when players get a high score and drawing the high score menu with a list of previous high scores.

### MenuComponent

Displays a menu with several options to the user and allows them to select a menu item. The Game Manager then handles user selections and scene transitions. Contains methods for returning the selected menu index and drawing the menu with user selection.

### MoveableTile

Base class for moveable tiles that inherits from Tile and includes additional properties to enable physics. Contains method to determine which way the tile is facing, which is used to flip the tile horizontally during animation.

### Pickup

Base class for items which can be collectedy by the player for points. Inherits from FixedTile and includes FloppyDisc and Disc; is ignored by the collision manager.

### ScoreDisplay

Inherits from DrawableGameComponent and displays the character’s current score in the top left corner of the screen

### Tile

Base class for all platform tiles that includes basic tile properties that will be applied to all child classes. Contains methods for updating tile bounds (for collision checking) and drawing the tile to the screen.

## Managers

### CharacterCollisionManager

Manages non-positioning collisions for the character; extends CollisionManager

### CharacterStateManager

Holds components of the character’s current state as static variables for universal access by scenes, managers and other components.

### CollisionManager

Base collision manager class for managing collisions between fixed and moveable tiles; contains all fixed and moveable tiles as lists

### PhysicsManager

Inherits from CollisionManager; manages physics like gravity and collision avoidance for tiles.

### TileCollisionPositioningManager

Manages collisions and post-collision positioning for tiles.

## Scenes

### AboutScene

A scene which inherits from GameScene; displays the names of the game's programmers.

### ActionScene

A scene which inherits from GameScene; displays the selected level and is the scene in which players actually play the game.

### GameOverMenuScene

A scene which inherits from GameScene; shows the game-over menu and high scores, and allows the player to go back to the main menu.

### GameScene

Base class for all game scenes; inherits from DrawableGameComponent; draws the scene's content and allows the user to interact with the scene.

### HelpScene

A scene which inherits from GameScene; displays a description of the game

### HighScoreScene

A scene which inherits from GameScene and displays the game's high scores

### HowToPlayScene

A scene which inherits from GameScene, and can be accessed from the main menu or pause menu; shows instructions for playing the game.

### LevelOverScene

Inherits from GameScene and is shown to the player when they complete a level

### LevelSelectScene

Inherits from GameScene and allows the user to begin playing any level which they have already unlocked

### PauseMenuScene

Displays the in-game pause menu; inherits from GameScene

### StartScene

Displays the pre-game main menu; inherits from GameScene

## Tiles

### Block

Inherits from Tile and represents the outer borders of the game level.

### Boulder

Inherits from Enemy; rolls in one direction and resets when it moves off the screen. Does not chase the character but causes him to lose a life if it hits him

### Character

Inherits from Tile and contains various player properties. Contains methods for losing lives, positioning, animation, and drawing the character to the screen.

### Disc

A collectable object worth a variable number of points and shaped like a compact disc; inherits from Pickup

### Enemy

Base class for enemy tiles that inherits from Tile and will end the current level when it reaches the player.

### Exit

Inherits from Tile and represents the end of the level, which will trigger the beginning of the next level.

### FloppyDisc

A collectable object worth a variable number of points and shaped like a floppy disc; inherits from Pickup

### Gem

Inherits from Tile and represents a collectible item that will increase player score by specified property when collected.

### LevelBarrier

Inherits from FixedTile; a translucent barrier that enemies like Boulders can pass but the character cannot.

### LevelEnd

Inherits from FixedTile; a barrier which represents the end of a level once the character has crossed it.

### Mimic

Inherits from PursuingEnemy; an animated hopping monster which chases the character's x position and jumps with every step

### Monstar

Inherits from PursuingEnemy; an animated monster which chases the character's x-position

### MovingPlatform

Inherits from Platform; moves a preset number of tiles back and forth and can be jumped on

### Platform

Inherits from Tile and represents a platform in the game that can be jumped on.

### PursuingEnemy

Inherits from Enemy; moves toward the character's X position and causes the character to lose a life if they collide

### Spike

Inherits from Tile and represents a character obstacle that will decrease the player’s lives if collided with.

## Utilities

### HighScore

Data structure for player highscores that stores player name and high score.

### ColorConverter

Utility class for converting colors from strings to XNA color objects. Contains methods for receiving a string input and returning the matching XNA color.

### Directions

Enumeration containing the various directions of movement in the game. Used by several components to determine direction vectors and perform a corresponding action.

## LevelLoader

Manages the loading of each game level from an XML file. Contains a method to create and return a list of all moveable tiles and fixed tiles in the specified level file.

### Settings

Utility class with several static variables to hold global game properties, including game window size and tile size.

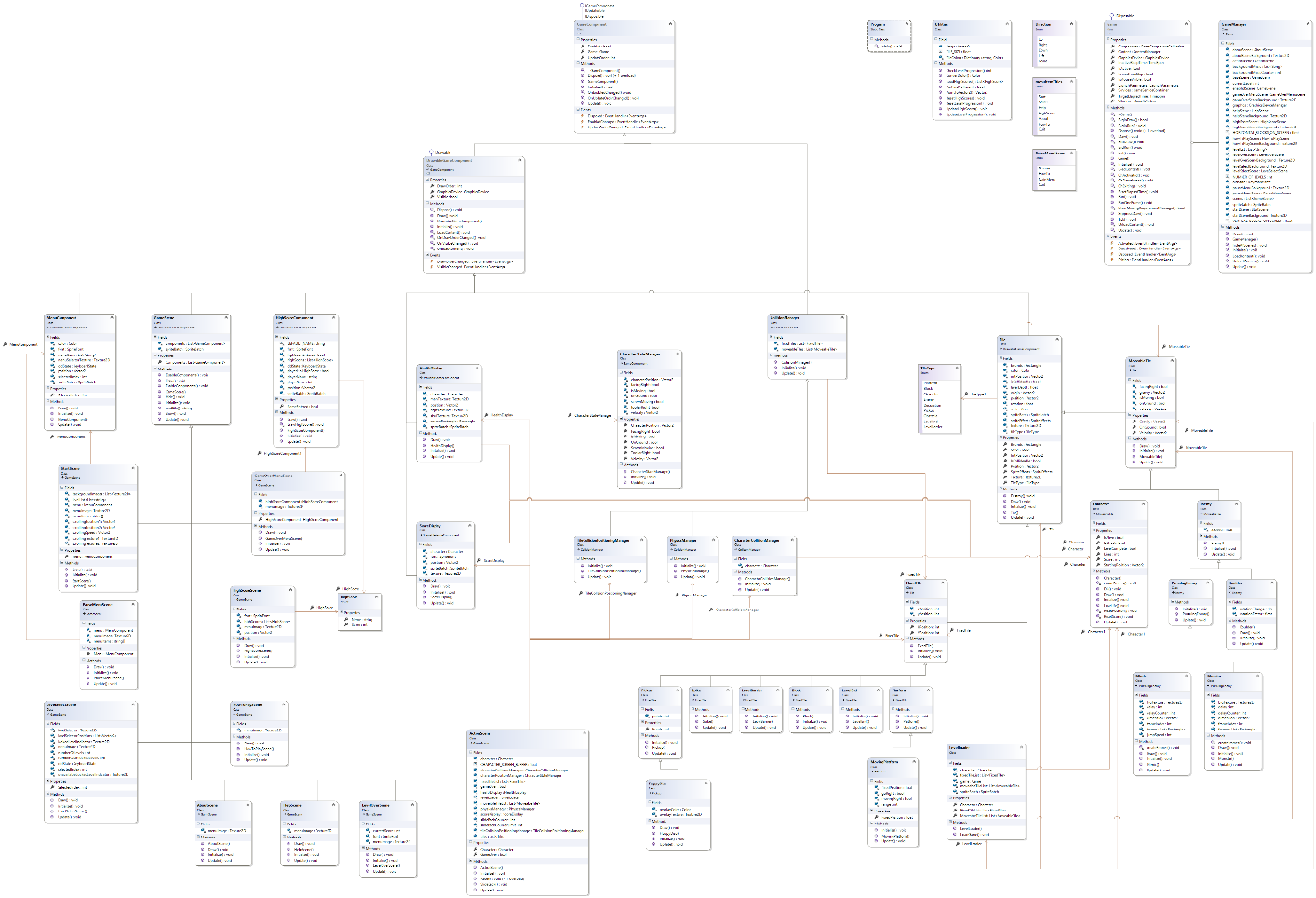
### TileType

Enumeration containing the various types of tiles in the game. Used by several components to determine tile type and perform a corresponding action (rather than checking class type).

### Utilities

Utility class with various miscellaneous methods that didn’t belong anywhere else. Contains methods for saving and loading XML high scores files, determining if pixel collisions have occurred, and converting Point data types to Vector2.

# Class Diagram



# References

## Images

* Tim Sprite
  + <https://pixeltheater.files.wordpress.com/2011/11/tim-from-braid.png>
* Braid Spritesheet
  + <http://www.swingswingsubmarine.com/2010/11/25/seasons-after-fall-spritesheet-animation/>
* Monstar Spritesheet
  + <http://spriters-resource.com/download/40891>
* Mimic Spritesheet
  + <http://spriters-resource.com/download/408910>
* Boulder
  + Class resources (Sabbir Ahmed)
* Dinosaur
  + <https://www.colourbox.com/preview/10522315-cartoon-dinosaur-skeleton.jpg>

## Sounds

* Ding
  + Generic computer ding
* Character Hurt sound
  + Minecraft
* Game Over sound
  + <https://www.freesound.org/people/Ionicsmusic/sounds/196868/>
* Applause sound
  + Class Resources (Sabbir Ahmed)
* Background Music (Cinderella’s Ballgag)
  + <http://opengameart.org/content/cinderellas-ballgag>
* Background Music (Drop Table Bass Final)
  + <http://opengameart.org/content/energetic-platformer-music-drop-table-bass>

## Code

* Pixel Collision
  + <http://stackoverflow.com/questions/7292870/per-pixel-collision-code-explanation>