CS 4670 – Work Log

Contents

[January 6, 2014 3](#_Toc386569502)

[Space Trader 3](#_Toc386569503)

[Overview 3](#_Toc386569504)

[Features 3](#_Toc386569505)

[Overview 4](#_Toc386569506)

[Features 4](#_Toc386569507)

[Endless Space 5](#_Toc386569508)

[Overview 5](#_Toc386569509)

[January 7, 2014 5](#_Toc386569510)

[Dwarf Fortress 6](#_Toc386569511)

[Overview 6](#_Toc386569512)

[Meeting with Professor Durney 6](#_Toc386569513)

[January 8, 2014 6](#_Toc386569514)

[January 9, 2014 6](#_Toc386569515)

[January 14, 2014 6](#_Toc386569516)

[January 15, 2014 6](#_Toc386569517)

[January 20, 2014 6](#_Toc386569518)

[January 27, 2014 6](#_Toc386569519)

[February 5, 2014 6](#_Toc386569520)

[February 7, 2014 7](#_Toc386569521)

[February 8, 2014 7](#_Toc386569522)

[February 12, 2014 7](#_Toc386569523)

[February 14, 2014 7](#_Toc386569524)

[February 19, 2014 7](#_Toc386569525)

[February 20, 2014 7](#_Toc386569526)

[February 27, 2014 7](#_Toc386569527)

[March 6, 2014 7](#_Toc386569528)

[March 13, 2014 8](#_Toc386569529)

[March 16, 2014 8](#_Toc386569530)

[March 17, 2014 8](#_Toc386569531)

[March 20, 2014 8](#_Toc386569532)

[March 24, 2014 8](#_Toc386569533)

[March 31, 2014 8](#_Toc386569534)

[April 4, 2014 8](#_Toc386569535)

[April 5, 2014 8](#_Toc386569536)

[April 8, 2014 8](#_Toc386569537)

[April 14, 2014 8](#_Toc386569538)

[April 20, 2014 8](#_Toc386569539)

[April 21, 2014 8](#_Toc386569540)

[April 22, 2014 9](#_Toc386569541)

[April 24, 2014 9](#_Toc386569542)

[April 25, 2014 9](#_Toc386569543)

[April 26, 2014 9](#_Toc386569544)

[April 28, 2014 9](#_Toc386569545)

[April 29, 2014 9](#_Toc386569546)

[April 30, 2014 9](#_Toc386569547)

[May 1, 2014 9](#_Toc386569548)

# January 6, 2014

Created git repository on github to host project. Added Apache 2.0 License for project. Added approved project proposal to repository. Investigated potential game styles for space-themed game prototype.

## Space Trader

### Overview

Space Trader is a complex game, in which the player's aim is to amass enough money to be able to buy a moon to retire to. The player starts out with a small space ship, armed with one simple laser, and 1000 credits in cash. The safest and easiest way to earn money is to trade goods between neighboring solar systems. If the player chooses the goods to trade wisely, it isn't too difficult to sell them with a profit. There are other ways to get rich, though. You might become a bounty hunter and hunt down pirates. It is also possible to become a pirate yourself and rob honest traders of their cargo. Beware, though: pirating is a way to get rich quickly, but the police force will go after you.

### Features

Some of the features of Space Trader are:

Ten different trade goods are available, two of which are considered to be illegal but can bring great profits.

Ten different ship types are available, some of which are especially suitable for trading, some for pirating, and others for both. Ships differ in size, the distance they can travel, their hull strength, the number of weapons, shields and gadgets they can carry, the number of cargo bays and the number of crew quarters that are available.

Ships can be equipped with different selections of equipment, among which are several types of lasers, several types of shields, an escape pod and certain special gadgets like a cloaking device.

You can distribute skill points over your character at the start of the game, to allow you to function well in your chosen role. For the skills your character lacks, mercenaries can be hired which may have different skills.

There are more than a hundred solar systems in the galaxy, with different sizes, tech levels, governments, resources and special situations. These are not just for background color, but play a vital role in the game.

While travelling to another solar system, you might encounter police ships, pirates and other traders. There are several ways to handle such encounters. You can even force a trader to allow you to plunder his ship.

There are about a dozen unique missions and offers available, some of which may bring great special rewards.

There is support for color, grayscale and black&white Palms, with OS version 2.0 and higher. OS 5.0 is supported.

The ships are displayed graphically during an encounter. There are also large pictures during key moments of the game.

Summary retrieved from <http://ticc.uvt.nl/~pspronck/spacetrader/STFrames.html>

<http://en.wikipedia.org/wiki/Space_Trader_(Palm_OS)>

Sid Meier's Pirates!

### Overview

Pirates! is a single-player, open world game. The player receives a letter of marque authorizing service as a privateer for the Spanish Empire, the Dutch Republic, the Kingdom of England, or the French colonial empire in the Caribbean. The player's loyalties may change over the course of the game; he may also hold rank with multiple countries and may turn to piracy at any time. Gameplay is open-ended; the player may choose to attack enemy ships or towns, hunt pirates, seek buried treasure, rescue long-lost family members, or even avoid violence altogether and seek to increase his wealth through trade. The game also has no predetermined end, although as time goes on, it becomes more difficult to recruit crew members. As the player character ages, fighting becomes more difficult, and deteriorating health will eventually force the character into retirement. The game ends when the player retires, at which point he is given a position in his future life, from beggar to King's adviser, based on accumulated wealth, land, rank, marital status, and other accomplishments.

### Features

Some of the features of Pirates! are:

The era of play is one of the choices given to a player at game-start. Different eras provide a different challenge, as political and economic power shifts between the four fledgling European empires.

Ship designs are also era-dependent, with some types of ships appearing more frequently in certain eras and less in others, and certain ship types being used near-exclusively by certain nations.

The game tests a wide range of skills:

hand-eye coordination during the fencing sections

tactical ability during the land and sea combat phases

strategic thinking, for everything from choosing a wife to deciding when to divide up the plunder.

Moreover, each game is likely to take a different course, as most events in the game are random, including the economic and political systems, and early in the game these can greatly affect future strategic options

In Pirates!, many of the most important factors which affect player decisions are randomized at the beginning of the game, and continue to shift during gameplay.

Changes happen whenever time passes and they are unrelated to player actions. In fact, in this game in the series, random events do not have any graphical representation, and the player can do nothing to prevent them.

Relations may differ greatly from game to game, and can shift in an instant, creating and removing opportunities, possibly even for long periods of time

Cities are also dynamic, with statistics like wealth and population fluctuating constantly. The player has a list of cities in one of the game's menus, allowing him to see the statistics of any city. Knowing the statistics of a city helps the player plan ahead, especially with regards to trading or any desire to raid or conquer cities.

Summary retrieved from <http://en.wikipedia.org/wiki/Sid_Meier's_Pirates!>

<http://sidmeierspirates.wikia.com/wiki/Sid_Meier%27s_Pirates!_Wiki>

## Endless Space

### Overview

Endless Space is a turn-based strategy game set in 3000 AD, where each player (up to a maximum of eight per game) represents the leader of one of nine unique interstellar empires. A player may also choose to create their own unique civilization by selecting from a number of different traits that correspond to military, science, diplomacy, hero units and so on. Each player is to guide their empire over hundreds, if not thousands, of years to diplomatic, scientific, or military conquest, attempting to meet requirements for several different victory conditions. The game takes place in a randomly generated galaxy, which can change in size and shape, depending on how the host player chooses to generate it. Every player begins with one colonised capital system which is connected to further systems via cosmic links, which act as travel routes for starships. Players can also research new technologies from four different research trees, representing military, science, expansion/exploration and diplomacy. Research unlocks new ship types, planetary improvements, stat modifiers (either for heroes or planets), new travel methods which do not rely on cosmic links and more. As players expand their empire, they will gain access to strategic and luxury resources, which can be used to upgrade ships, build improvements and trade with other players. Strategic resources are primarily used to upgrade components of your empire, while luxury resources are primarily used for trade and maintaining your empire's approval rating.

The game uses four basic resources to manage its economy: food, industry, dust and science, or FIDS. Dust is a substance that was leftover from an extremely powerful, ancient, extinct civilization only known as the Endless. Players must balance FIDS in order to rapidly expand their empire, build ships and research advanced technologies. In addition, a lower or imbalanced FIDS across your empire results in a low approval rating. A high approval rating provides bonuses to production efficiency, while a low approval rating can drastically lower efficiency making it incredibly difficult to advance. A tax rate slider can also be adjusted to change approval rating, but lower taxes result in a lower revenue stream of Dust. Hero units can be recruited using the game's currency, Dust, to act as either fleet commanders or system administrators. Each hero unit is unique and provides two unique bonuses, which can be further leveled up. Three hero units are randomly selected from a pool unique to each empire and new hero units are available for purchase every fifty turns.

# January 7, 2014

Reviewed Dwarf Fortress and met with Professor Durney.

## Dwarf Fortress

### Overview

Every game in Dwarf Fortress starts with the generation of a new world; only one game at a time can be ongoing per world. The exact qualities of a world are randomly generated, but can be influenced quite heavily with input from the player, who determines the map size, natural savagery, mineral occurrence, et cetera. The world generator first uses a fractal algorithm to create a randomized elevation map. This is then further elaborated upon by a temperature map, rainfall projection map, drainage value, vegetation value, and salinity. Each tract of land is then differentiated into a biome based upon a combination of these values. Tracts of land are then sorted into evil, neutral, or good regions, as well as benign, wild, or savage ones. Mountains are then worn away with temporary rivers, followed by permanent ones flowing from high points to low ones. Local animal and plant populations are established, followed by sentient populations.

## Meeting with Professor Durney

Define four or five different games for network architecture to support. These may include SpaceTrader, Risk, Battleship, StarCraft, and others. Choose one game to implement. Design several different network architectures that support these particular games.

# January 8, 2014

Researched network architectures and created diagrams of possible network architectures.

# January 9, 2014

Started defining requirements.

# January 14, 2014

Continued defining requirements.

# January 15, 2014

Continued defining requirements.

# January 20, 2014

API design

# January 27, 2014

Game Design

# February 5, 2014

Researched Python networking and websockets

Implemented websocket chat server

# February 7, 2014

Installed pypi and Tornado

# February 8, 2014

Built simple echo server in tornado. Debugged networking issues in HTTP with Fiddler

Implemented protocol for server

# February 12, 2014

Implemented protocol for game engine

# February 14, 2014

Met with professor Durney and migrated python development from NotePad++ to PyCharm

# February 19, 2014

Tested every request for server and game engine.

Server and game engine are stubbed out for their logic

Server protocol logic complete.

Game engine protocol logic complete.

# February 20, 2014

Began Windows application.

Researched:

Tasks

Await

Async

HTTPClient

PostAsync

JSON libraries

# February 27, 2014

Completed tentative windows application

Began Web Application

# March 6, 2014

Completed tentative web application

Began major refactoring

# March 13, 2014

Continued major refactoring

# March 16, 2014

Everything broken because I tried to move python functions into classes.

# March 17, 2014

Everything working again, moved python functions out of classes. Began MongoDB work.

# March 20, 2014

Database working

# March 24, 2014

Database designed and implemented

# March 31, 2014

Database refactoring

# April 4, 2014

Began comprehensive refactoring

# April 5, 2014

Continued comprehensive refactoring

# April 8, 2014

Continued comprehensive refactoring

# April 14, 2014

Continued comprehensive refactoring

# April 20, 2014

Communication between game engine and server not working.

# April 21, 2014

Communication working again

# April 22, 2014

Completed basic UI for Windows Client and Web Client.

Completed chat functionality and took screenshots of client displays.

# April 24, 2014

Added Pencil files and first draft presentation and report

Added logic diagrams and got HTTP chat working

# April 25, 2014

Added final draft presentation

Continued work on project report

# April 26, 2014

Continued work on project report

# April 28, 2014

Continued work on project report

# April 29, 2014

Continued work on project report

# April 30, 2014

Continued work on project report

# May 1, 2014

Finalized work on project report