|  |
| --- |
| Project Plan |
| Family Jewels |
| Peter Brill Jr.  Paul Fahey  **Revision 1.0**  **Wednesday, February 29, 2012**  **Revision 1.1**  **Wednesday, March 07, 2012** |

Table of Contents

[Introduction 2](#_Toc318898022)

[Project Organization 2](#_Toc318898023)

[Hardware and Software Resource Requirements 3](#_Toc318898024)

[Hardware 3](#_Toc318898025)

[Software 3](#_Toc318898026)

[Work Breakdown 4](#_Toc318898027)

[Project Schedule 4](#_Toc318898028)

[Risk Analysis 5](#_Toc318898029)

[Monitoring and Reporting Mechanisms 6](#_Toc318898030)

[Project Post Mortem 6](#_Toc318898031)

# Introduction

This project is primarily focused on the development and release of the application ‘Family Jewels’. Family Jewels (Known herein as FJ), will be a simple economy and buy/trade simulator, borrowing directly from such works as ‘Drug Wars’ or the simple economy systems found in many RPG and Puzzle games. Simply put, the only real strategy for FJ is to buy low and sell high until a win condition is met. Above all else our focus with FJ is to make it both fun and rewarding to the player; Attempting to be at times both challenging and fun.

# Project Organization

## Paul ‘Bird with Arms’ Fahey

Paul is primarily focused on the front end or menu-structure of the application. He will be tasked with designing the economies interaction with the player, as well as all the required structures to interact with the main focus of the program: The Economy. The player will primarily interact with the interface to in turn interact with the economy.

## Peter ‘Petey’ Brill Jr.

Peter is primarily focused on designing the economy, implementing all the member functions and capabilities so the interface only needs to interact with the well-designed Economy object to complete all of its tasks. The economy object itself will use a queue structure to randomize the prices day by day of the economy items.

## The ‘Family Jewels’ Team

Together in tandem many of the interconnection design problems will be handled by the team at large. The connection between the economy and the player-facing interface is paramount; and both members of the team will be investing significant time to make sure that this is okay.

# Hardware and Software Resource Requirements

## Hardware

For the development of FJ, both members of the team will need access to some significant systems to develop and test the application. The hardware used for said development will at the very least need to host Microsoft Visual Studio 2010 (What we primarily use for development) while still having the raw computational power to compile and run the program very rapidly, simply to avoid an extra bottleneck in the process.

## Software

The primary setup used to develop this application will be as follows:

*Windows 7*

This will be developed in a Windows 7 environment, using it as a platform to support and work with Visual Studio 2010.

*Microsoft Visual Studio 2010*

The primary tool for developing the application will be Microsoft Visual Studio 2010 (Herein referred to as VS2010.) We will be using VS2010 because of its capability to debug, implement, develop and track then placement of all the application source code files. Although VS2010 is a hog of resources, our hardware platforms (Custom built systems for personal use, built within the last year) have no trouble giving it everything it requires.

# Work Breakdown

## Paul Fahey

* Gant Chart/Project Schedule
* Test Plan
* Interface Code/Design

## Peter Brill

* Project Plan
* Project Requirements
* Class Code/Design
* User Guide

## Team Effort Items

* Design Document
* Application Code Implementation

All of the team members will be involved in the compilation of the Design Document and the final combination of the applications two parts into a meaningful implementation. Though multiple team members will work on some of the requirements, it is always the responsibility of the person the requirement is listed under to ensure a high level of quality in the completed work.

# Project Schedule

This is often a Gantt chart or a reference to an MPP file.

# Risk Analysis

Some of the main things considered to be risks are as follows:

* Loss of documentation data through hardware/software failure.
* Loss of source data through hardware/software failure.
* Team Disconnection, where work is being done that will not fit into the final product due to lack of communication.

To deal with these three fears succinctly, a very simple methodology solves the first two. To avoid having to recomplete work previously completed, each team member will make constant personal backups of all documentation and source code: Contributing to a communal daily/nightly shared location. We are currently investigating Google Documents cloud capabilities to fulfill this need, and until the need arises that something else is required, it will suit our needs perfectly.

Concerning the last of the three obvious pitfalls, the nightly/daily updates to the shared copy of the files are to be annotated clearly with change logs especially of major changes that may affect the work done by the opposite team member. Large changes should be discussed and communication will be paramount to developing the application without serious difficulties.

# Monitoring and Reporting Mechanisms

The progress within the project will be primarily tracked via a Gantt Chart. As each ‘Work Breakdown’ item is completed, it is the responsibility of the listed team member to make sure it is both proofread and free of any errors to the best of their ability to discern such errors. The primary mechanism of organization is the two man check and balance system, this works as follows:

* Both team members will keep a communal ‘team document book’ with a list of each requirement
* Before a requirement is considered ‘Complete’ it must be looked over by both team members and unanimously agreed to be complete
* ‘Finished’ pieces of the puzzle will only be revisited if absolutely required, and if so (especially concerning changes to base documents, such as this one) must be done with both of the FJ team members aware of these changes before, during, and after they have occurred

This system labors to be simple yet elegant in design, allowing the least amount of sweat for the most amount of productivity, at a quick estimation.

# Project Post Mortem

This section is filled in after the project has been completed. It is a review of went well, what did not and lessons learned.