**Ascension**

**Problem Statement**

**Rose-Hulman Institute of Technology – CSSE 376**

Kenneth Faulkner

Gabriel Glenn

Jack Petry

Table of Contents

**Table of Contents** **1**

**1 Executive Summary** **1**

**2 Introduction** **1**

**3 High Level Problem Summary** **4**

3.1 Elevator Statement 5

3.2 Primary Success Criteria 6

3.3 Scope 6

3.3.1 Within Scope 6

3.3.2 Outside Scope 6

**4 Detailed Problem Statement** **4**

4.1 Function 5

4.2 Form 6

4.2.1 Availability 6

4.2.2 Usability 6

4.2.3 Performance 6

4.2.4 Security 6

4.2.5 Maintainability 6

4.3 Economy 6

4.31 Marketability 6

4.4 Time 6

4.4.1 Historical 6

4.4.2 Current 6

4.4.3 Future 6

4.5 Schedule 6

**5 Key Stakeholders** **1**

**6 Version Information**

1 Executive Summary

The purpose of this document is to describe and summarize the problem faced by the players of the game Ascension. This was written in conjunction with the entity-relationship diagram, as part of the first set of deliverables for CSSE 376. This document contains a high level problem statement, analysis of the economical impact of this project, and information regarding key stakeholders in the project.

Ascension players face difficult decisions when choosing a medium with which to play the game. We couldn’t find a version of Ascension for the computer. In order to rectify this problem, we will create a thoroughly tested version of Ascension that is available for the PC.

2 Introduction

This document describes a problem related to the board game Ascension, specifically, that it is only playable as a board game or as a smart-phone application. This document is intended to specify the exact parameters of the problem. It is also the first set of deliverables. Following deliverables will include a final presentation of the finished product, along with periodic status reports.

3 High Level Problem Summary

3.1 Elevator Statement

Our main goal is to create a working software version of the deck building game Ascension: Chronicle of the Godslayer.

3.2 Primary Success Criteria

* Users can view all of the relevant details of the game from the screen
* Users can play cards and witness their effects as they progress through the game
* The game can be played either as a hotseat game or against an AI
* At the end of the game, the winner is declared
* The system is durable and works consistently

3.3 Scope

**3.3.1 Within Scope**

* Players of the game
* Cards in the game
* Keeping track of honor
* AI
* Expansion Packs
* Promo Cards
* Basic Graphical Representation of Cards

**3.3.2 Outside Scope**

* Networking
* Security
* Intricate animation

4 Detailed Problem Statement

4.1 Function

Our final product must be able to simulate the card game Ascension, and accurately represent the game logic in all situations. This program will be used to play games of Ascension on the user’s PC. We will prioritize a basic “hotseat” gameplay before any AI or elaborate graphics.

**4.1.1 Key Business Features**

* Easy to understand visual display
* An easy to use movement interface

**4.1.2 Key Enabling Features**

* We, as admins, must be able to update the cards usable by the player

**4.1.3 Key Concurrency Issues**

* With hotseat play mode, it is very important that the screen shift is timed well
* It is also important that all of the results of a play are viewable simultaneously to avoid user confusion

4.2 Form

**4.2.1 Key Attributes**

* Performance & capacity-The system needs to run quickly enough so as not to appear sluggish to users
* Reliability & availability-The system must be made robustly and be usable on all modern platforms
* Usability-User experience is a high priority for any game. Users must be able to easily understand what is going on within the game make plays.
* Modifiability, maintainability, & customizability-The system will need to have appropriate abstraction and testing suite to enable rapid development and easy maintainability
* Testability-The system should set up in such a way that all of the business is easily testable

**4.2.2 Hardware & Software Constraints**

This project will be created in the Java programming language, designed to be used on Windows machines. This means our tests will be JUnit tests.We will be using the Eclipse IDE and SVN for version control.

**4.2.3 Key Interfaces**

We will not have anynetwork infrastructure or any information going in or out of the system. There will be no required data formats or shared data.

**4.2.4 Required Standards**

The solution will be constrained to functioning on Windows machines, and even further, only working on machines that have support for Java. The standards for the project itself is that it will follow the rules as described in the hard copy of the card game.

**4.2.5 Domain**

This project is similar in scope and implementation to other computerized board games and card games. As such, the problem domain of those designs is very similar to the problem domain of our project. Additionally, there is an iOS version of the Ascension card game, so our project domain has a similar domain to this as well.

4.3 Economy

**4.3.1 Business Context**

* Our project will allow the customer, or end user, to play Ascension on their computer
* Ascension is an increasingly popular game more and more people want to play and our project will offer a solution

**4.3.2 Customer Organization**

* A computer and its necessary input hardware (i.e a mouse) are required
* The customers will probably be local to our locations. Our meaning the developers.
* The player must be able to read

**4.3.3 Development Organization Constraints**

* Rose-Hulman Students
* Software Engineering Students
* Available budget: A twinkie and a couple of pizzas
* Expected cost: No cost aside for time (and sleep)

**4.3.4 Key Risks and Uncertainty**

* Some people prefer to be able to physically hold the cards
* People may be disappointed by the implemented graphics.

4.4 Time

**4.4.1 Historical Context**

The previous version of this solution is a physical deck building time.

**4.4.2 Current Context**

The game needs to be completed within the next 9 weeks to meet organizational requirements.

**4.4.3 Future Context**

The product should be designed such that future development cycles can improve upon the product.

4.5 Schedule

Week 1: Have problem statement complete

Week 2: Plan out the different classes and their interaction. Have components tabbed out. Develop base platform for entire project to allow divide and conquer.

Week 3: Get the game board background displaying and cards displaying. Internally handle cards being played.

Week 4: Have event handlers to allow users to click on cards to play them. Have turns able to be played.

Week 5: Add additional card functionality.

Week 6: Add basic “working” AI that chooses cards to play and makes decisions.

Week 7: Finish out AI to make it good, possibly adding different difficulty levels.

5 Key Stakeholders

|  |  |
| --- | --- |
| **Name** | **Role** |
| **Buffalo** | **Grader** |
| **Kenneth Faulkner** | **Developer** |
| **Gabriel Glenn** | **Developer** |
| **Jack Petry** | **Developer** |

6 Version Information

|  |  |  |
| --- | --- | --- |
| **Version** | **Date** | **Comments** |
| 1.0 | 3/14/2013 | The initial draft of the problem statement paper. |
| 1.1 | 3/19/2013 | Added Schedule |