**Avalon Online**

SOFTWARE REQUIREMENTS SPECIFICATION

Ryan Austin Fernandez

**Table of Contents**

|  |  |
| --- | --- |
| 1. Executive Summary | 1-1 |
| 1. Overview | 2-1 |
| * 1. Community Flow and Features | 2-1 |
| * + 1. Game Flow and Features | 2-1 |
| * + 1. Statistics | 2-2 |
| * 1. *Data Requirements for Storage of Community Records* | 2-2 |
| * 1. Roles in the Process | 2-3 |
| 1. Problem Analysis | 3-1 |
| 1. Software Solution | 4-1 |
| * 1. Objectives | 4-1 |
| * 1. Characteristics | 4-1 |
| * 1. Scope and Assumptions | 4-1 |
| 1. User Stories | 5-1 |
| * 1. <User Story 1> | 5-1 |
| Appendix A - Improved Business Process | A-1 |
| Appendix B - Sample Forms and Reports | B-1 |
| Appendix C - References and Acknowledgements | C-1 |

1. **Executive Summary**

Avalon is a board game created by Don Eskridge as an improvement over his previous board game, The Resistance, with gameplay similar to it, but with additional features.

The game involves the Knights of the Round Table attempting to succeed in three quests ordered by King Arthur. However, there are treacherous minions of Mordred amongst the loyal servants of Arthur attempting to fail the missions. It is the loyal servants’ mission to discover the identity of the traitors

The additional features of Avalon with respect to The Resistance are its additional character cards: Merlin, Assassin, Perceval, Morgana, Mordred, Oberon, and the Lady of the Lake adding more social dynamics to the gameplay. (The Resistance, n.d.)

1. **Overview of the Business Process**

*This chapter presents an overview of a simple Avalon gaming community’s regular operations flow. Included in this chapter are the following items:*

* *Community Flow and Features*
* *The game’s flow and features*
* *Data requirements for storage of game records*
* *Different kinds of users in the community*
* *Proposed site’s user privileges*

**2.1 Community Flow and Features**

In the community, at most one game may be hosted by a member. That game can be joined or left by the host’s friends in the community. Once the maximum number of players (set by the host) or less, depending on whether the host wants to begin or not, have been met, the game begins. A game may also be cancelled before it begins.

A member may flag another member for misdemeanor, to be reviewed by a moderator or an administrator. An administrator may grant a member the rank of moderator or administrator.

2.1.1. Game Flow and Features

Each game has five to ten players and has a role distribution phase, a quest phase, and an end phase.

Given n players, there are minions of Mordred present. At the beginning of the game, the roles, whether a player is a servant of Arthur (good) or a minion of Mordred (evil), are randomly distributed. The minions of Mordred then secretly discover each other’s identities (in physical games, this is done by everyone closing their eyes then the minions open their eyes).

The quest phase comprises of five quests, each with sub phases, namely the selection, discussion, approval, quest proper, and reveal phases. A team captain is randomly designated at the beginning of the game. This team captain selects the appropriate number of people for the quest depending on how many players and which quest it is in the game. (see Appendix B-1). Once a team is selected, the players then discuss the composition of the team. Once a discussion has been raised, they will now vote on whether they approve or reject the team. If the majority (for even number of players, ties are not majority) approves, the selected players go on a quest and may choose to play a success or fail card. If the vote fails, the team captain token is passed to the left and the selection, discussion, and approval sub phases repeat. If the approval phase fails five times for a particular quest, evil automatically wins. Loyal servants of Arthur have to play a success card. Minions of Mordred may choose to play success or fail. Once they secretly give the team captain their decisions, the team captain shuffles the cards and reveals them one by one. If at least one fail is present, the mission fails and the appropriate mission marker is placed on the game board (see Appendix B-2). An exception is made for the fourth quest in games of more than seven players. It will take two fails for such quests. After each quest, team captain passes to the left.

Before the explanation of the end phase, it is necessary to discuss two optional special characters: Merlin and Assassin. When these two are in play, at the beginning, the evil players raise their thumbs and Merlin opens his eyes. Therefore, evil is revealed to Merlin. He may then influence the flow of the game. In the end phase, if one team has succeeded or failed in three quests, that team wins. However, if the good team has won, the assassin may now guess who among the good players Merlin is. If he is successful, the evil team still wins. A game with Merlin cannot be played without Assassin.

As for the other special characters, Perceval knows who Merlin is; Morgana appears as Merlin to Perceval; Mordred is hidden from Merlin; and Oberon is hidden from the evil players, but the evil players are also hidden from him. Oberon may choose to fail a mission if he pleases.

Regarding the Lady of the Lake, at the beginning of the game, the Lady token is given to the person to the right of the Team Captain. After the second quest, this person must choose a player to use the Lady of the Lake on. This player is given a red (evil) and a blue (good) loyalty card. They must give their loyalty card to the person who used the Lady of the Lake, who then returns the loyalty card. The Lady of the Lake is passed to the player it was used on and the previous user is now immune to the Lady of the Lake. The former user may now discuss the card that was given to him, even if he lies. This is repeated for the third and fourth quests.

Targeting is also an advanced game option. The team captain may choose to do the quests in any order, except the fifth quest, which may only be attempted after two successes. The fourth quest for games with at least seven players still require two fails. The Lady of the Lake still only functions after the second, third, and fourth quests.

After the game is finished, the statistics of the game are tallied. The members of the community may view these statistics for future reference, either on a per game basis or in total for each member or the entire community.

*The business process can be visualized in the diagram shown in Figure 2-1.*

*<insert your business process diagram here>*

*Figure 2-1. Existing Business Process*

2.1.2. Statistics

The community and member statistics may be viewed by any member at any time. The specific details of which are further discussed in section 2.2.

***2.2 Data Requirements for Storage of Community Records***

Each member has a first name, a last name, a username, a password, a bio, a member level (bronze after playing 50 games, silver after playing 125 games, gold after playing 250 games), and an authorization level (regular, moderator, administrator). A member may only join one game at any given time.

Each game has a game id number, date and time hosted and ended, whether it is ongoing, an optional minimum number of players (which cannot go below five), an optional maximum number of players (which cannot exceed ten), a list of special characters included, whether targeting is enabled, whether the Lady of the Lake is enabled, number of successes, number of fails, winner, and any discussion comments (similar to a chatbox) made during the game.

Each quest has at least one and at most five selections, with the captain who chose the team, each player’s vote for each selection and whether it was approved. It also has the quest results, with each member of the quest’s played card (success or fail) and the result of the quest.

For games with the Lady of the Lake, it is also necessary to mark players who are immune to the Lady of the Lake at any given point in time.

The community statistics comprises total games good won, evil won, good v. evil ratio, total game forfeits, total evil default wins (through five team rejections in one quest), and total games played.

The player statistics are total games played as good; total games played as evil; total games won and lost, as evil, as good, and in total; total games as any of the special roles; total games hosted; and total games played.

***2.3 Roles in the Process***

There are various kinds of members in the Avalon Game Community as shown in Table 2-1.

|  |  |
| --- | --- |
| Role | Description of Tasks |
| Member | * Hosts a game * Cancels a game * Joins a game * Begins a game   + Selects special characters for a game   + Sets special rules for a game   + Selects members to go on a quest   + Votes for a team   + Determines success of a quest   + Discuss the game * Flags a user * View Community Statistics * View User Profile (with statistics) * Edit Own Profile |
| Moderator | * All member privileges * Moderates flagged users |
| Administrator | * All moderator privileges * Grants moderator or administrator privileges |

Table 2-1. Member hierarchy in the Avalon Online

1. **Problem Analysis**

This chapter presents the findings of the investigation on the organization’s needs and problems to be addressed by the software. (Only problems to be ADDRESSED) This section will also provide the reader with a background of the organization which is the primary stakeholder of the system to be developed. The various users and stakeholders of the software are also presented here.

| **ID** | **Description** | **Cause** | **Symptoms** | **Impact** |
| --- | --- | --- | --- | --- |
| # | What’s the problem? | What causes the problem? | How do we know the problem exists? | Why is this important? What are the consequences? |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

***The section ends with the statement of the problem, need or opportunity where the software is the proposed solution (BUSINESS REQUIREMENT).***

1. **Software Solution** 
   1. **Objectives**

This section starts with a statement of the general objective indicating what the development team will do to address the problem(s) identified in the previous section. The general objective should be measurable, attainable, and realistic. The usual format for this statement is “The <software> aims to … “

This statement is followed by an elaboration of the general objective. This is done by enumerating the specific objectives of your software, for example:

*“The specific objectives of the software are as follows:*

* *To provide a facility for managing the records of call center applicants, from resume to training plans;*
* *To provide a facility for tracking the employment history of call center applicants;*
* *To provide a facility for managing the scheduling of training courses;*
* *To provide a facility for call center companies to register with ECTPA;*
* *To provide a facility for managing job openings transactions from request to deployment;*
* *To generate various reports for monitoring call center operators, companies and job openings”* 
  1. **Characteristics**

This section discusses the non-functional requirements that need to be addressed to achieve the business goals stated in Chapter 2.

1. **User Stories**

|  |  |
| --- | --- |
| **User Story #1:** The lab personnel (pathologist, secretary, medical technician) will use a password to give him/her access to the system. | |
| **Estimate (Days):** | **Priority:** |
| **Pre-condition:** The system is running. | |
| **Scenario:**   1. The system prompts the lab personnel for a password. 2. The lab personnel inputs a password. 3. The system validates the password. 4. The system displays the main menu. | |
| **Post-condition:**  The lab personnel has successfully logged in. The main menu is displayed. | |
| **Acceptance Criteria:**   1. Test that if the password is correct, the main menu will show up. 2. Check if the password is incorrect, the main menu will not show up and the user will be prompted for the correct password. | |

*This chapter presents the user stories included in the product backlog.*

Notes:

* The scenario should provide the sequence of interaction between the user and the system based on valid inputs.
* There should be no mention of interface details (such as screen, buttons, clickable, presses) or platforms (web) anywhere in the user story (including pre- and post-conditions, scenario and acceptance criteria).
* Pre-conditions must state the constraints (on user roles, data availability) that must hold true before the user story can be performed.
* Post-conditions must state the outcome (on data, process, and user state) that will hold true when the user story has been performed.

**Appendix A – Improved Business Process**

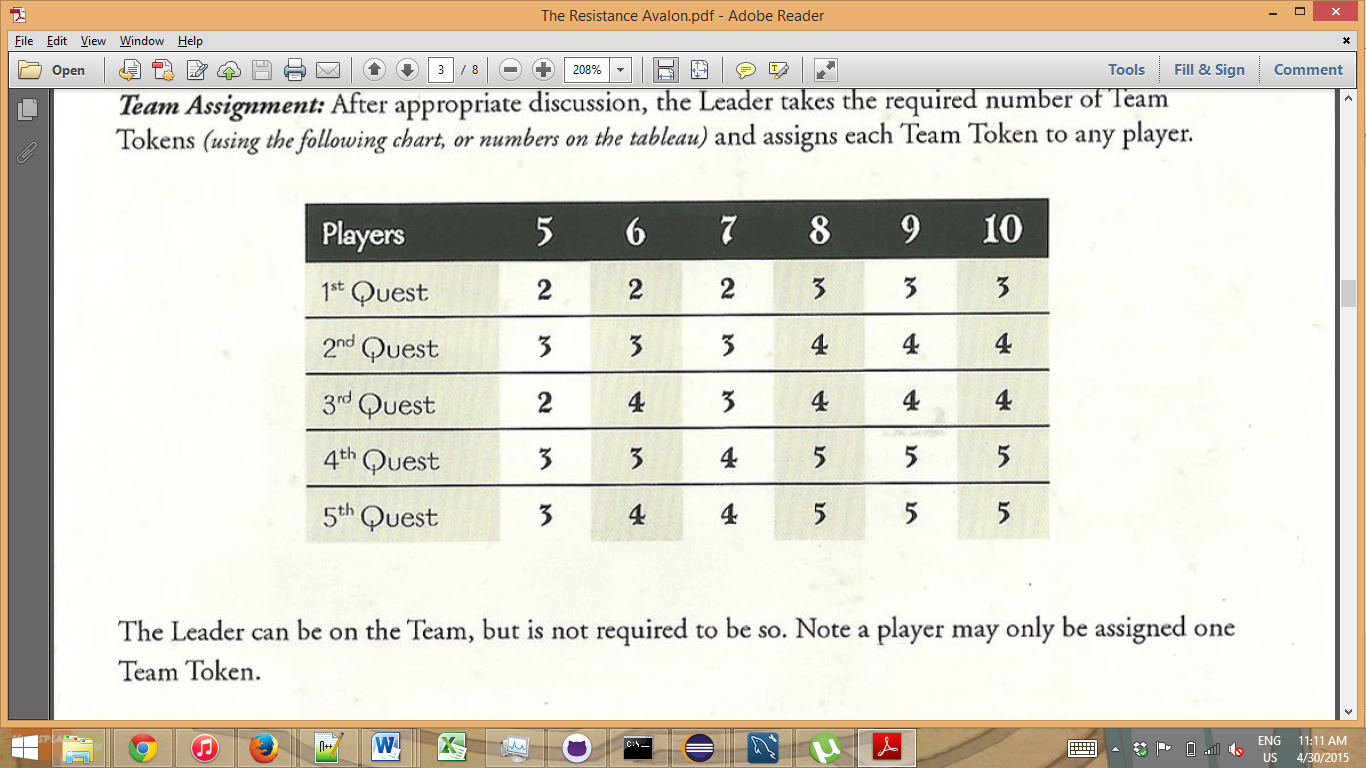
*This chapter presents the improved business process when the proposed software solution is implemented. This visualizes how the software solution benefits or affects the current business process.*

**Appendix B – Sample Forms and Reports**

*This chapter contains the various gameplay materials needed for the site.*

B-1. Player/Quest Table

This table notes the number of players to send on each quest in the game, depending on the total players.



B-2. Score Tableau

This is a sample of a score tableau used for the game.



**Appendix C – References and Acknowledgement**

The Resistance. (n.d.). Retrieved April 30, 2015, In Wikipedia: http://en.wikipedia.org/wiki/The\_Resistance

\_%28game%29