**Software Requirements Specification**

**For**

**“Cards Against Telegram”**

|  |  |
| --- | --- |
| **Instructor:** | David Augenblick |
| **Team Members:** | |  | | --- | | Jessica Hoban | | Amir Omidi | | Balaji Lakshmanan | | Sagar Patel | | Tanfe Aderemi | |
| **Cycle:** | 1 |
| **Date Submitted:** | 6 October 2016 |

Document template copyright 2005-2015, CCI Faculty. Version 2.3. Use permitted under Creative Commons license CC-BY-NC-SA. See http://creativecommons.org/licenses/by-nc-sa/3.0/.

**Grading Rubric - Requirements Specification**

This rubric outlines the grading criteria for this document. Note that the criteria represent a plan for grading. Change is possible, especially given the dynamic nature of this course. Any change will be applied consistently for the entire class.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Achievement** | **Minimal** | **Exemplary** | **Pts** | **Score** |
| **Content (80)** | Section(s) missing, not useful, inconsistent, or wrong. | Provides all relevant information correctly and with appropriate detail |  |  |
| Introduction  Scope  Definitions |  |  | 10 |  |
| User Profile |  |  | 20 |  |
| Functional Requirements |  |  | 30 |  |
| Performance & Design Requirements |  |  | 10 |  |
| Data Requirements |  |  | 10 |  |
| **Writing (20)** |  |  |  |  |
| Grammar and Spelling | Many serious mistakes in grammar or spelling | Grammar, punctuation, and spelling all correct | 10 |  |
| Expression | Hard to follow or poor word choices | Clear and concise. A pleasure to read | 5 |  |
| Tone | Tone not appropriate for technical writing | Tone is consistently professional |  |  |
| Organization | Information difficult to locate | All information is easy to find and important points stand out | 5 |  |
| Layout | Layout is inconsistent, visually distracting, or hinders use | Layout is attractive, consistent, and helps guide the reader |  |  |
| **Late Submission** |  |  | -10  -25 |  |
| **Total** |  |  | 100 |  |

# **Introduction**

## **Scope**

Cards Against Telegram is an implementation of the popular game, Cards Against Humanity. We will implement an interface to play Cards Against Humanity in Telegram. The user will be able to select card packs, invite other players and see global statistics online.

The motivation for this project is that, since chatbot games are gaining popularity and momentum, there is a public demand for this project. Our target audience is people in their late teens.

## **Definitions, Acronyms, and Abbreviations**

Card Czar: The user that chooses the best response

Black Card: The prompt card given by the software

White Card: The response card dealt to each user by the software, and chosen by the users. These cards are refilled every round to their original fixed amount.

## **User Profile**

Users will maintain their Telegram profile, and no distinction is made between the users during the game. Card Czars are picked at random in the beginning; this role is rotated sequentially so that each user gets a turn. This way, all users are equal, and fall under the same category of “player”.

# **External Interfaces**

The users are interacting through their laptops or mobile devices to connect with a minimum of two other players. The system will interact with the database that the cards are stored in.

## **User Interface**

The user interface is Telegram. Our software will run on its API, and have the same appearance and functionality as other games on Telegram.

## **Data Interface**

There is one common database that all three users connect to. Through this, they will be dealt cards and submit their responses, which other players will be able to see.

# **Specific Requirements**

## **Functional Requirements**

(All users fall under the same category of “player”. Thus, we will make no distinction between them for the system’s functions.)

The application should function as such:

After starting the game on Telegram…

1.   Users make sure that their opponents are connected.

2.   All users are dealt a fixed amount of cards.

3.   The system will randomly pick a Card Czar at the beginning of the game. Following that, each user gets a turn as Card Czar.

4.   The system deals a black card.

5.   Users (other than the Card Czar) have 10 seconds to pick one of their white cards as a response.

6.   Once submitted, all users get to see each other’s responses.

7.   The Card Czar gets 10 seconds to choose the best response. The user with the best response gets a point for that round.

8.   The next user is chosen as Card Czar. Steps 4 through 7 are repeated until the users decide to end the game. (Option will be provided.)

## **Performance Requirements**

Requirements are as follows:

1.       Hardware: Desktop/Laptop when using Telegram through the computer or smartphone with touch responsiveness for mobile accessibility

2.       Software: Internet browser access to see website. Access to database for scores, user records, etc.

3.       Security: Secure user information and database

4.    Performance: begin game, end game, make move. Response time relative to Telegram’s servers (most cases – nearly instant)

## **Design Constraints**

Our software can run on any platform that Telegram supports- that is, laptop, desktop, or mobile versions such as smartphones.

General user knowledge of using games on Telegram is recommended.

## **Data Requirements**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Type** | **Size** | **Comment** |
| Card | Black | N/A | The prompt card given by the software |
| Card | White | N/A | The response card dealt to each user by the software, and chosen by the users. These cards are refilled every round to their original fixed amount. |