**Assassin Admin - Software Requirements Document**

**Introduction / Purpose:**

The Assassin Admin application is used by the Administrator to create and run games of Assassin. Only the Administrator will use this application. Players use the Assassin Player application. The Administrator can use this application to manage multiple games and perform all tasks within a game.

**Terminology:**

* Administrator – Generally doesn’t play in the game. Recruits players, handles disputes, pays out players.
* Player – Pays to play in the game. Can earn bounties.
* Game – Single game with a defined beginning and end.
* Rules – Specific rules for a single game.
* Chain – Primary data structure within the game. The Chain is a closed loop where each Player has a single Target and each Player is a single Target.
* Waiting Queue – List of Players that are awaiting assignment into the Game.
* Target – Each player needs to find and assassinate their Target
* Assassination or Kill – To successfully <squirt> your target according to the rules
* Bounty – The earnings for an Assassination
* Witness – Person (Player or Non-Player) that witnessed the Assassination. A person that witnessed the Assassination may choose to decline to be a Witness.
* Re-buy – If a Player has at least 1 Bounty to be paid, they can use that Bounty to rejoin the game after they have been Assassinated.
* Stun – Player A has a Target of Player B. If Player B <squirts> Player A, then Player B has immunity from Assassinations for a period of time defined by the Rules. Player B Stunned Player A.
* Admin payout – Configurable methods for Admins to earn.
* Bomb – Full shuffle of all active and waiting players.

**Scope:**

* Release 1 - Minimum Required Functionality to create and run a single game.
* Release 2+ - Optional Features – Operating multiple games, all optional features within a game.

**User Profiles:**

* Platform Administrator – LSA (Us)
* Game Administrator – Us during PorcFest, and some customers down the road. Going forward all reference to Admin is to this Game Administrator.

**Use Cases:**

* Outside Game:
  + Login to Platform
  + Logout of Platform
  + Create Game
  + Find Game
  + Load Game
  + Edit common rules\*
    - Minimum Break Length
  + Player Management (If players info is stored outside the game)\*
  + Run Reports\*
* Within Game:
  + Player related:
    - Accept payment for pre-registered player – Field in player document? Or just wait to approve player until Admin gets the cash.
      * Accept extra bounties up front
    - Approve registered player\* – Player registered on their own and uploaded a pic. Admin approves picture, name, accepts payment, and moves player into waiting status.
    - Add Player\* – Walk around recruiting process. Think through the new process, paper slips needed? Old process, paper slip with 2 numbers, I keep one with name and description of person so I can activate later.
    - Find Player\*
    - Edit Player data – Same question, store globally and in game?
      * Upload Picture\*
      * Update data – name?
    - Re-activate Player (Approve Feature)\*
    - Pay Bounty to Player\*
    - Remove Player\*
    - Ping Player – New – GPS is an alternative
    - Message Player – New
    - Message Admins – New
  + Game - Level
    - Edit Game Data – start date, end date, hours, rules, etc.
    - Create Admins
    - Establish Rules – Pick from a common collection of rules for each game style, then update rules for the specific game.
    - Start Game (Make Chain)\*
    - Clear Chain – Feature existed in Japhet’s version. Not sure if its needed going forward?
    - Drop Bomb\*
    - Force break Scheduled Scenario – Consider Message to Players
    - End Game\*
    - Run Reports\*

**Functional Requirements:**

* App Notifications – Ability to alert players on key events including bomb, forced break impact, game start, game over, start day and end day reminders (configurable?)
* Enable timed starts – Release of waiting players at X:00 PM. Need to break chain, volunteer approach.

**Data Requirements:**

* Games
* Rules
* Administrators
* Players
* Chain
* Waiting Queue
* Events
* Reports
* Team

**State Requirements:**

* Matrix
  + Admin Status vs. Allowable Actions
  + Game Status vs. Allowable Actions.
* Incorrect or Error States

**Technical:**

* Security
  + Admin id
  + Game id
  + Player id
  + Access rights to data
* Usability
* Performance

gsutil cors set cors.json gs://assassinfirestoretest1.appspot.com