* Admin
  + Command line driven
    - Parameters for access codes and hunts
      * Access codes (are dependant upon user and hunt)
        + One user may have multiple access codes meaning an access code is generated with each new hunt for each new user in the hunt.
        + Access codes have a status that changes based on which step the user is at (active/pending) or if the game is active (active or disabled)
      * Hunt (new game)
    - Admin calls a function to start a new game
      * All information required for a new game is taken as user input and stored server side (need to check with tech on the best way to do this)
        + All game input in BuchuntUserStories file on github
    - A function for editing will be available to the admin
      * It will be accessible based on the parameter that defines game status as only available if pending or active
      * Creation date and status are immutable characteristics
    - Ability to pull from the server to create accounts for list
      * Pulling user info from etsu server (copy/move) can be possibly done with SQL or bash depending on the VM that is stood up.
        + Parameters on a user account will be automatically copied from existing data
      * (could use grep) Include search functionality with parameters to search by as
        + Email
        + Phone number
        + Access code
* Player
  + Interface (web/url driven)
    - Joining a hunt
      * A user receives their access code and opens url browser
        + The browser needs to have an ability for user input for just the access code
        + That access code will correlate with the user and hunt that is associated with it (check server/admin)

Once the user enters the access code the hunt associated with that access code becomes active (server) unless otherwise disabled by the admin.

If the game is not active then the user will not be allowed to join (check on hunt status based on user access code,)

* + - * Hunt started (status active)



