

Clientside Commands

These are a list of commands that can be executed by the client. Each subsection contains a different menu which can be accessed from the client. Additionally, the Basic Commands contains an option to quit the application, and the other menus offer an option to go back to the menu accessed previously. Any inputs from a user are specified using brackets.

Basic Commands

These commands are accessed when first using the application. These commands do not have a specific argument unless otherwise specified.

- READ: list leaderboards
 - shows all leaderboard names on this server
- READ: open leaderboard [leaderboard name]
 - opens "Commands Associated With a Leaderboard" with leaderboard name as the argument.
- ADMIN: create leaderboard [leaderboard name]
 - adds a leaderboard with the specified name
- READ: list users
 - outputs a list of all users with ID and identity
 - Additional client request fields:
 - None
 - Resource server response:
 - A list of users, each a tuple of (id, identity)
- READ: open user [user ID]
 - opens "Commands Associated With a User" with user ID as the argument
- READ: open self
 - opens "Commands Associated With a User" with the logged in user as the argument

Commands Associated With a Leaderboard

These commands are all related to a leaderboard which has been opened by the user.

- READ: list entries
 - shows all verified entries (ID, place, player name, score, date/time) on the leaderboard
- MOD: list unverified
 - shows all unverified entries (ID, place, player name, score, date/time) on the leaderboard
- WRITE: submit entrv

- then prompted to enter
 - score
 - description
- READ: open entry [entry ID]
 - opens "Commands Associated With an Entry" with entry ID as the argument.
- MOD: get access groups
 - gets users and their respective access levels for this leaderboard
- ADMIN: score order [ascending/descending]
 - set the ordering of scores to be ascending or descending
- ADMIN: remove leaderboard
 - deletes leaderboard from the database (after asking to confirm)

Commands Associated With an Entry

These commands are all related to an entry which has been opened by the user.

- READ/MOD: view entry
 - Access Denied if entry is unverified and user is not moderator
 - show the ID, place on the leaderboard, player name, score, date/time of submission, date/time of verification and who verified it (if none, declare it to be so), list of proof files, and number of comments for the current entry
- WRITE: add proof [filename]
 - must be original author of the submission
 - upload specified file as a proof file to this entry
- READ/MOD: download proof [filename] into [local filename]
 - Access Denied if entry is unverified and user is not moderator
 - download the proof file into the local filename specified
- MOD: remove proof [file_id]
 - must be a moderator or the person who submitted the entry
 - removes the file associated with proof of the entry
- READ/MOD: view comments
 - Access Denied if entry is unverified and user is not moderator
 - show all comments, printing poster's name, text content of the message, and a date/time. chronologically ordered.
- WRITE/MOD: post comment [message]
 - Must be owner of submission or moderator
 - add a comment containing your message

- MOD: verify entry
 - mark the entry as verified
- MOD: unverify entry
 - mark the entry as unverified
- WRITE/MOD: remove entry
 - Must be owner of submission or moderator
 - take the entry out of the database

Commands Associated With a User

These commands are all related to a user which has been opened by the logged in user.

- READ: view user [user_id]
 - Access Denied if requester has No Access
 - Gives an error if a user with the id doesn't exist
 - Shows the name and registration date of the user
 - Lists all entries from a user that the requester has access to
- MOD: view permissions [user_id]
 - List all permissions given to this user, as leaderboard:access pairs
- MOD: set user [user_id] permission for [leaderboard_id] to [permission]
 - sets the user's access level for the given leaderboard
- READ: open submission [entry ID]
 - opens "Commands Associated With an Entry" with entry ID as the argument.
- ADMIN: remove user
 - deletes user from database (with confirmation)
 - optionally deletes all content associated with that user
 - this sounds really hard :)
 - not sure what should be deleted and what should be anonymized