DATABASE MongoDB users **BACK-END SERVER** addresses highscores **RESPONSE**: when the REST API has found what the client requested, it sends back a message object, typically in XML or JSON **QUERY**: REST API queries format database for what the Client requested **REST API** Receives HTTP request form Clients and does whatever the request needs, e. g. create users

CRUD OPERATIONS:

Create: creates new records or entities in a database. It typically involves inserting new data into the system. E.g., creating a new user account, adding a new product to an inventory, or inserting a new entry in a blog post.

express.js

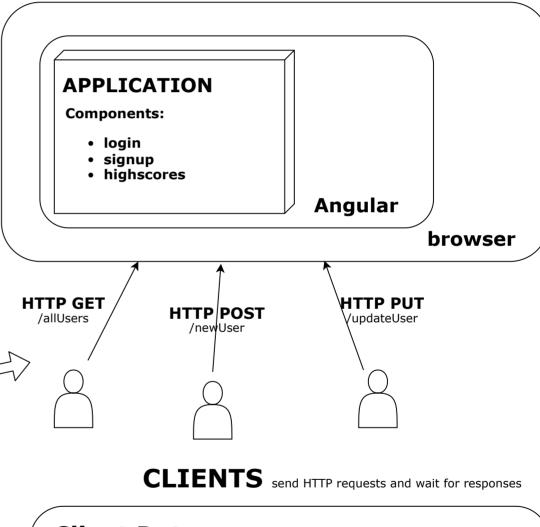
node.js

Read: retrieves existing data from a database. It involves fetching or accessing data that has already been stored. E.g., retrieving user details, displaying a list of products, or accessing specific information from a database.

Update: modifies or updates existing data in a database. It involves making changes or edits to the existing records. E.g., updating a user's profile information, changing the quantity or price of a product, or editing the content of a blog post.

Delete: removes or deletes existing data from a database. It involves eliminating or discarding records that are no longer needed. E.g., deleting a user account, removing a product from the inventory, or deleting a specific entry in a database.

FRONT-END APPLICATION



Client Data

- The client sends his email address, his password and his postal address details to the database when signing up
- The client can request his personal data, when logging into the application
- The client can access other data that belongs to him, such as personal high scores
- Each client can request the server to send the high scores of other users, so that he can see the current leaderboard
- The client can also make a new entry into the high scores table, by sending his new high score data to the database