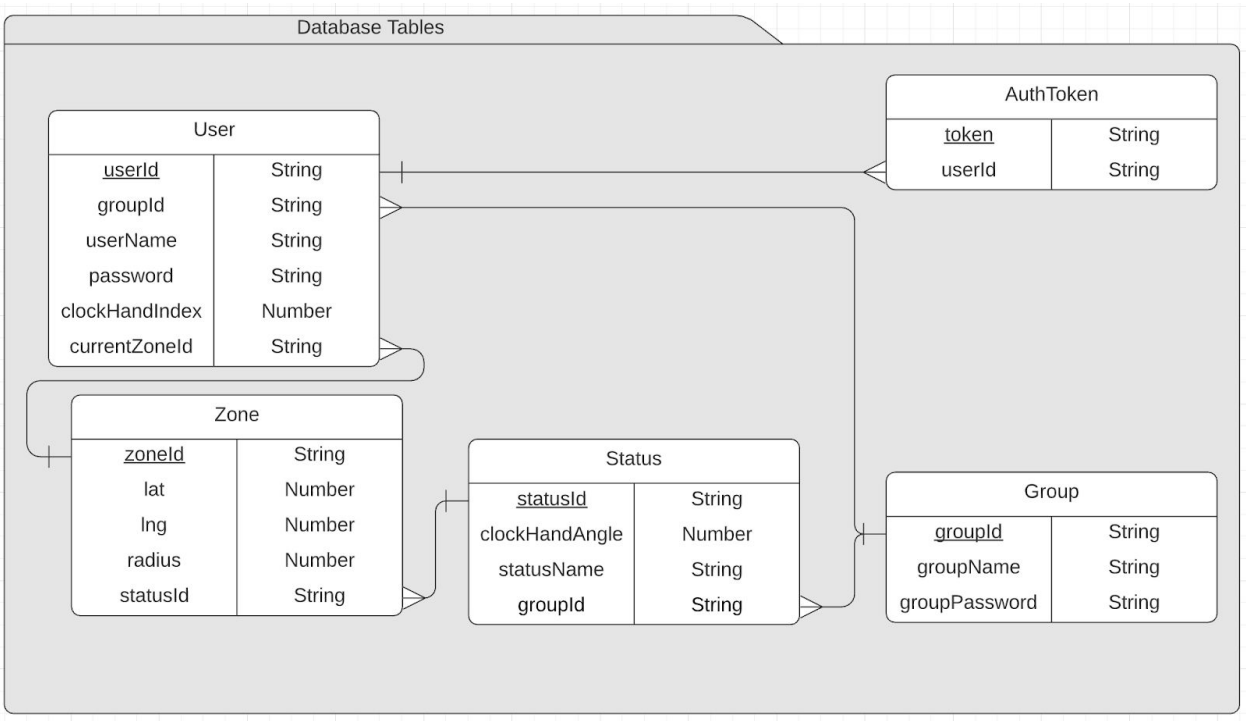
Schema Report

Ryan West, Alec Petersen, Xin Zhao



[Github Link]

My Contribution: As the project leader/creator, I already had a strong idea for what the schema would look like for this project so I made most of the schema. Since most of it was done already, my teammates helped me refine the schema to fix errors and adjust names and relationships to be more appropriate.

Questions/Doubts: After we completed our schema, I realized that both the User and the Clock will need to log in and authenticate with the server. So, perhaps the AuthToken should be changed to represent a polymorphic type so that it can be an AuthToken for a User or a Clock. AuthToken could have two columns, userID and groupID, whichever is not null is the type of AuthToken for that row (refer to how to represent inheritance and polymorphism in a database from the readings).
Another question: Perhaps if Group is used to represent, not just the group of users, but also the statuses belonging to that group, and the login credentials for that group's clock, the name for the table should be Clock (or ClockGroup) instead. That would make sense from a modeling perspective: multiple users have their whereabouts tracked by a single clock, multiple statuses can be displayed on a single clock, and the login details for the clock are located in the Clock table.

Evidence of normalization: There is no duplicated data in our schema. When updating the database, a change needs to be made in only a single place. We use multiple tables to separate different entities.

User: A single person utilizing the features of this app connected to a single group and sharing their position in a single zone.
User (<u>userID</u>, groupID, zoneID, username, password, clockHandIndex)
Foreign Key groupID references Group
Foreign Key zoneID references Zone

userID - Unique identifier for each user. Primary Key.
groupID - The group to which the user belongs.
zoneID - The zone (geofence) that the user is currently within.
username - Name (unique) chosen by each user. Candidate Key.
password - Password chosen by user.
clockHandIndex - Index to consistently assign each user to a specific servo operating the clock hands.

AuthToken: A token sent with each request to verify the identity of the sender.
AuthToken (<u>token</u>, userID)
Foreign Key userID references User

Token - An unique id that is signed to a user once he/she logs in, and it will expire when a user logs out. Primary Key.
userID - Unique identifier for each user that sent the request.

Zone: Representing a geofence around a single location in the physical world.
Zone (<u>zoneID</u>, lat, lng, radius, statusId)
Foreign Key statusId references Status

zoneID - Unique identifier for each zone. Primary Key.
lat - The latitude of the center of the geofence.
lng - The longitude of the center of the geofence.
radius - The radius of the geofence, measured in meters.
statusId - The status to which the zone belongs.

Status: Represents a class of places that a user could be located, indicating the type of activity they are currently performing. For example, there could be multiple zones, one at Walmart, another at the mall, and another at Smith's. Each of these zones would be in a single Status called Shopping.
Status (<u>statusID</u>, clockHandAngle, statusName, groupId)
Foreign Key groupId references Group

statusID - An unique id that identifies each statue in the table. Primary Key.

clockHandAngle - The angle of the clock pin that displays on the user interface and on the physical clock.

statusName - the name of the status. Candidate Key.

groupID - The group to which the status belongs.

Group: A group of users that share their location with the same clock.

Group (groupID, groupName, groupPassword)

groupID - Unique identifier for each group. Primary Key.

groupName - Name (unique) chosen by group creator. Candidate Key.

groupPassword - Password chosen by group creator.