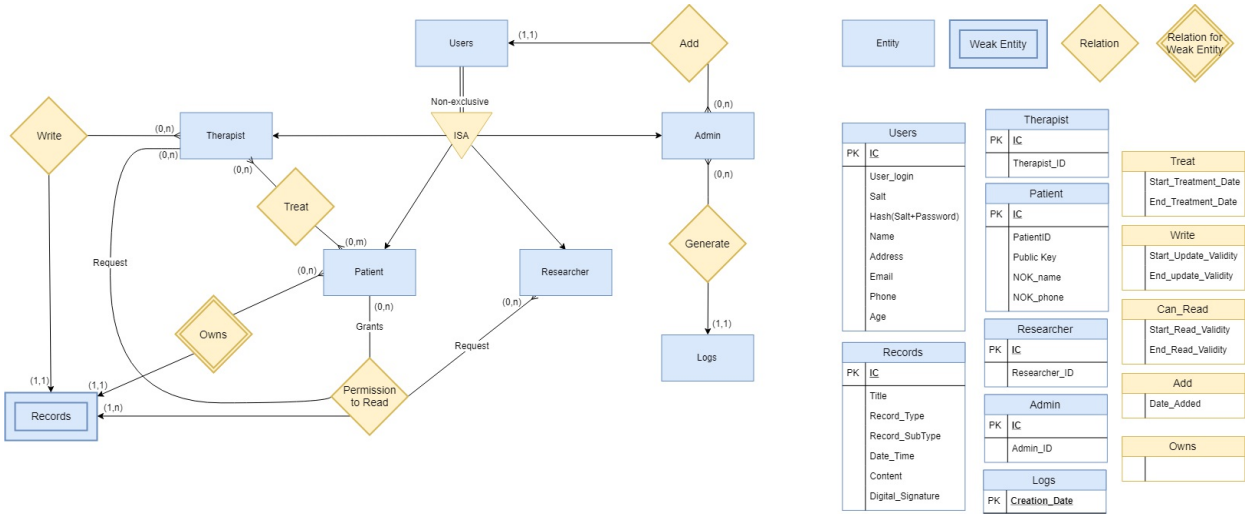
Database Design

ER Diagram

The ER diagram for our database is as follows:



Tables

The tables for our database can be found in the text file here.

Design Description

Users

There is a total of 4 types of `Users`:

1. Therapist
2. Patient
3. Researcher
4. Administrator

A user can be a therapist, a patient, a researcher, an administrator or a combination of any of them. Although a user can have multiple roles, the web application for each role will be separate (i.e. the user has to re-login to another web application if he wishes to assume a different role).

Only the IC number is sufficient to identify each different unique role.

The patient role would have these additional attributes:

- Public key (to verify the signatures of records)
- Private key (stored in the tag instead of the database)
- Next-of-kin's name
- Next-of-kin's phone number (in case of emergency)

Health Data

There are 5 types of `Health Data`:

1. Reading
2. Image
3. Time series data
4. Movie
5. Document

These health data will be categorised as records. Each record has the following attributes:

- Type
- Subtype
- Title
- Date_time
- Owner_ic
- Signature
- Content

The 'type' attribute describes the type of health data (i.e. either reading, image, time series data, movie or document). The 'subtype' attribute describes the type within the specific type of health data. For example, a blood pressure reading record can have a reading type and a blood pressure subtype. The 'title' attribute describes further in detail what the record is about. Using the same example of the blood pressure reading record, the title could be "dizzy spell". The 'date_time' attribute specifies the date and time the record was created. The 'owner_ic' attribute refers to the IC number of the owner of the record (who could either be a therapist or a patient). The 'content' attribute refers to the value of the record. For records typed as readings, the value of the content attribute will be the value of the reading. For records typed as time series data, the value of the content attribute will be the raw time series data. For records typed as documents, the value of the content attribute will be the text of the document. As MySQL is not a good database for storing images or movies (images and movies have to be stored as blobs, and this causes a lot of overhead), images and movies will

be saved to the file system in our database server. Therefore, for records typed as movies and images, the value of the content attribute will be the links to the movies and images in the server's file system respectively. The signature attribute contains the signed content by the owner of the record for verification of ownership. This is necessary for records owned by patients but not by therapists.

In our database, there are 2 types of relationship. The first relationship is the "treats" relationship. Therapists treat patients. Each therapist can treat zero or more patients, while each patient must be treated by at least one therapist. The "treats" relationship has the following attributes: therapist_ic (which identifies the therapist in the relationship), patient_ic (which identifies the patient in the relationship), start_date (for record purposes) and end_date (for checking the validity of consent i.e. if the "treats" relationship has ended, the consent given by the patient to the therapist or vice versa in the relationship will be invalid). The second relationship is the "gives_permission_to" relationship. This is a ternary relationship in which a therapist gives permission (consent) to a patient or vice versa to view a particular record that he owns. The "gives_permission_to" relationship has the following attributes:

- Therapist_ic
- Patient_ic
- Owner_ic (specifies the owner who is also the one granting consent)
- Expiry_date (to check the validity of the consent; should also be in line with the end date of the "treats" relationship)
- Record_date_time (date and time when the record was created). By default, when a patient gets assigned to be treated by a therapist, permission (consent) is granted to the therapist. The patient will have to actively remove the permission (consent) if he does not want the therapist to view his records.

Table

There is a total of 8 `Table` in our database:

1. Users
2. Therapists
3. Patients
4. Researchers
5. Administrators
6. Records
7. Treats
8. Gives_permission_to (where the "users" table is necessary to ensure unique IC numbers are keyed in)

Permissions

1. Therapists will be able to list their patients, create and edit own documents, view records that patients have given permission for and print out records which the therapists can view.

2. Patients will be able to create and edit own documents, view but not edit their own records (except documents), view documents (reports) that their therapists have given permission for and print out records which the patients can view.

3. Researchers will only be able to retrieve and view anonymised data stored in a separate table.

4. Administrators will be able to add users (including administrators) and display logs of all transactions in the system (stored in a log file).

5. Documents can only be edited by their owners and can be viewed by users whom the owners have given permission (consent) to. Other records can only be viewed by their owners and other users whom the owners have given permission (consent) to.

6. To determine the capabilities of a user, his IC number will simply be checked against the tables for the different user types and capabilities are determined accordingly. To check the permission to view or edit records, the type and ownership of record and the "gives_permission_to" table will be checked against.

Anonymizing and Data Retrieval

A special table for research data will hold anonymized data. The data will not contain any names or unique identifying information.

A form will be used to retrieve data using prepared statements. This is because we will be dealing mainly with parameter inputs for this segment. It also provides less overhead in terms of building a query.

Security

Server and Database

The server root access will be removed and instead a special Database administrator account will have admin access to the server. All databases will set up according to CIS guidelines for best practices. The servers hosting them will be following these guidelines as well.

Authentication

To authenticate the user, the user has to key in his IC number and his password. The web app will verify the IC number and password by checking against the hash of the user's password and salt in the user table of our database. If it is a match, the user will be able to log in successfully. However, 2-Factor Authentication will be used for patients. In addition to the checking of the password and IC number, patients are required to have their BLE tags nearby. The BLE tags should send a pair of values (a random number and an encrypted value of the random number using the patient's private key) to the web app. The web app then further verifies the identity of the user by decrypting with the patient's public key stored in our database. If successful, the patient will be logged in successfully.

File System

All image and video files will be scanned with an anti-virus scanner before upload. Attempting to upload an unsafe file will result in an account getting locked.

Preventing Attacks

All input information will be escaped to prevent any SQL and Cross-site scripting (XSS) attacks. Executables will not be allowed for upload as they are not required for our purposes.