

Procedures/Process involved in the project

Overview of the Online Counseling App:

- The online counseling app is designed to facilitate seamless communication between users seeking mental health support and professional counselors. The app's architecture ensures a secure and efficient user experience, utilizing database principles to manage user information, counselor details, and interactions. Here's an in-depth look at how the app functions, divided into sections for counselors and users, with considerations for both new and returning users.

Front Page and Initial Navigation:

- Upon launching the app, users are greeted with a front page that acts as a gateway, directing them to either the counselor or user login pages based on their role. This initial step is crucial for establishing the correct user flow and ensuring that individuals are routed to the appropriate interface for their needs.

FOR COUNSELORS:

- **Login Process:**

Counselors access the app through a dedicated login page where they enter their unique username and password. This authentication process is essential for verifying the identity of the counselor and ensuring that only authorized personnel can access sensitive information.

- **Profile Page:**

Once logged in, counselors are directed to their profile page. This profile page serves as a dashboard displaying both the counselor's personal details and an overview of their assigned patients. The profile page is dynamically populated with data from the database, ensuring that counselors have up-to-date information about their patients at all times.

- **Chat Inbox:**

From the profile page, counselors can navigate to their chat inbox. The chat inbox is a central communication hub where counselors can engage in real-time messaging with their patients. The database logs all chat interactions, providing a secure and confidential platform for counseling sessions.

FOR USERS:

- **Returning Users:**

Returning users can log in using their anonymous username and password. This login process verifies their identity against the stored credentials in the database. Once authenticated, they are directed to their profile page, which includes their username, assigned counselor, and the specific mental health issue they are addressing. This information is fetched from the database and presented in a user-friendly format.

- **Chat Inbox Access:**

From their profile page, returning users can access their chat inbox to communicate with their assigned counselor. This consistent access point ensures that users can easily continue their ongoing counseling sessions.

NEW USERS:

- **Registration Process:**

New users begin by selecting the sign-in option to register their details. The registration form requires an anonymous username, a preferred email, and a password. This data is securely stored in the database, ensuring user anonymity and data protection.

- **Selecting Assistance Needs:**

After registering, new users are directed to a page where they specify the issue they need assistance with. The app restricts users to selecting one issue during registration, a design choice that simplifies the initial matching process with a counselor. This selected issue is stored in the database and used to personalize the user's experience.

- **Assignment to a Counselor:**

Once the user specifies their issue, the app utilizes a load-balancing algorithm to assign them to a counselor. This algorithm checks the number of patients each counselor

currently has and assigns the new user to the counselor with the least number of patients. This ensures an equitable distribution of workload among counselors and timely attention to new users. The user and counselor details are displayed on the screen during this initial registration period, sourced from the database.

- **Access to Chat Inbox:**

After viewing the counselor details, the new user can proceed to their chat inbox to initiate communication with their assigned counselor. This step is critical in establishing the first point of contact and beginning the therapeutic process.

1. Formulation of Business Rules

Business Rule 1: User Profile Requirements

- A user must complete their profile, including specifying the types of problems they need counseling for, before they can be matched with a counsellor.
- Rationale: This ensures that the system has enough information to make an appropriate match.

Business Rule 2: Counselor Availability

- A counselor should not be assigned more than a predefined maximum number of users at any given time.
- Rationale: This prevents any single counselor from being overloaded, ensuring they can provide adequate attention to each user.

Business Rule 3: Matching Criteria

- A user can only be matched with a counselor who has indicated they are comfortable dealing with the user's specific type of problem.
- Rationale: This ensures that users receive appropriate support from counselors who are qualified to address their issues.

Business Rule 4: Load Balancing

- Among the eligible counselors, the system should assign the user to the counselor who currently has the least number of assigned users.
- Rationale: This promotes an even distribution of users among

counselors, preventing any counselor from being overburdened while others are underutilized.

Business Rule 5: Anonymity

- The system must not display any personally identifiable information (PII) of users or counselors during the matching process or within the chat system.
- **Rationale:** Maintaining anonymity is crucial to the integrity and safety of the counseling platform.

6. Implementation of tables/functions/procedures/views/triggers

Database Design and Table Structure

Counselors Table:

- **Attributes:** Practitioner_No, Counsellor_Name (PK), Email, Password, Qualifications, Availability, Ratings
- **Purpose:** Store details of counselors.
- **Initial Data:** 10 counselors are pre-registered with their details.

Illnesses Table:

- **Attributes:** Illness_Name (PK), Illness
- **Purpose:** Store common mental health issues.

Users Table:

- **Attributes:** Username (PK), Email, Password, Illness_Name (FK)
- **Purpose:** Store user details.
- **Function:** Users register and choose a mental health issue they need assistance with.

Assignments Table:

- **Attributes:** Assignment_ID (PK), Username, Counsellor_Name (FK), Illness_Name (FK)
- **Purpose:** Store assignments of users to counselors based on the illness.

Detailed Workflow

Procedures:

- Could be implemented for registration processes and assignments to ensure consistency and reusability.

User Registration and Login (Procedures)

User Registration:

- **Form Input:** Username, Email, Password
- **SQL Query:** INSERT INTO Users (Username, Email, Password) VALUES (?, ?, ?);
- **Process:** User details are inserted into the Users table.

User Login:

- **Form Input:** Username, Password
- **SQL Query:** SELECT * FROM Users WHERE Username = ? AND Password = ?;
- **Process:** Verify user credentials for login.

Counselor Registration and Login(Procedure)

Counselor Registration (initially preregistered):

- **Form Input:** Practitioner_No, Counselor_Name, Email, Password, Qualifications, Availability, Ratings
- **SQL Query:** INSERT INTO Counsellors (Practitioner_No, Counselor_Name, Email, Password, Qualifications, Availability, Ratings) VALUES (?, ?, ?, ?, ?, ?, ?);
- **Process:** Counselor details are inserted into the Counselors table.

Counselor Login:

- **Form Input:** Counselor_Name, Password
- **SQL Query:** SELECT * FROM Counsellors WHERE Counsellor_Name = ? AND Password = ?;
- **Process:** Verify counselor credentials for login.

Implementations of procedures, functions and views for “Chats” table

Procedures:

Sending message from sender to receiver:

- **Form Input:** Sender, Receiver, Message, Timestamp, Unread
- **SQL Query:** INSERT INTO Chats (Sender, Receiver, Message, Timestamp, Unread) VALUES (?, ?, ?, NOW(), 1);
- **Process:** Message details are inserted into the Chats table.

Read/Unread messages specifier:

- **Form Input:** Sender, Receiver
- **SQL Query:** UPDATE Chats SET Unread = 0 WHERE Sender = ? AND Receiver = ? AND Unread = 1;
- **Process:** Update chat messages to mark them as read (Unread = 0) for messages sent from the specified Sender to the specified Receiver where they are currently marked as unread (Unread = 1).

Views:

Retrieve messages between sender and receiver:

- **Form Input:** Sender, Receiver
- **SQL Query:** "SELECT * FROM Chats WHERE (Sender = ? AND Receiver = ?) OR (Sender = ? AND Receiver = ?) ORDER BY Timestamp
- **Process:** Retrieve all messages from the specified Sender and Receiver, ordered by Timestamp.

Functions:

- Could be used to encapsulate complex SQL queries, such as finding the least busy counselor.

Assigning Counselors to Users(Functions)

Choosing Mental Health Issue:

- **Form Input:** Illness_Name
- **SQL Query:** UPDATE Users SET Illness_Name = ? WHERE Username = ?;
- **Process:** User selects a mental health issue which updates the Users table.

Finding Least Busy Counselor(:

- **SQL Query:** SELECT Counselor_Name, COUNT(*) AS Total_Booking_Count FROM Assignments WHERE Counsellor_Name IN (SELECT DISTINCT Counsellor_Name FROM Assignments WHERE Illness_Name = ?) GROUP BY Counsellor_Name ORDER BY Total_Booking_Count ASC LIMIT 1;
- **Process:** Identify the counselor with the least number of assignments for the selected Illness_Name.

Assigning User to Counselor:

- **Form Input:** Username, Counsellor_Name, Illness_Name
- **SQL Query:** INSERT INTO Assignments (Username, Counsellor_Name, Illness_Name) VALUES (?, ?, ?);
- **Process:** Create a new assignment in the Assignments table.

Views:

- Could be created to simplify complex joins and aggregations, such as a view to show counselor assignments with user details.

Displaying Profiles and Assignments(Views)

Displaying User Profile:

- **Form Input:** Username
- **SQL Query:** SELECT * FROM Users WHERE Username = ?;
- **Process:** Fetch and display user details on the user profile page.

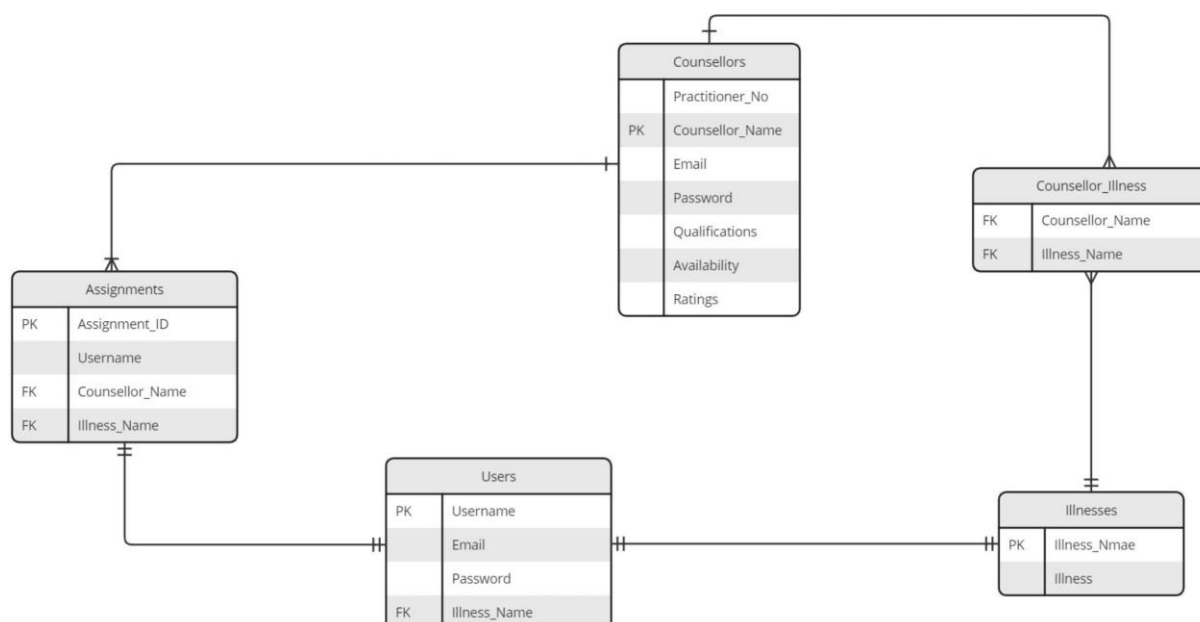
Displaying Counselor Profile:

- **Form Input:** Counsellor_Name
- **SQL Query:** SELECT Username FROM Assignments WHERE Counsellor_Name = ?;
- **Process:** Fetch assigned user details for the counselor profile page.
- **Additional Query:** For each username, fetch user details: SELECT * FROM Users WHERE Username = ?;

Summary of our implementation

This implementation involves creating structured tables to store counselors, users, illnesses, and assignments. We use functions to encapsulate logic like finding the least busy counselor. We use procedures to standardize registration and assignment processes, WE use views to simplify data retrieval, and finally triggers ensure auditing of assignments. This approach ensures a clear, maintainable, and efficient database structure for our online counseling app.

3. Initial ERD



Counsellors to Assignments:

- A counsellor can be assigned to one or many users through the "Assignments" table. This is a one-to-many relationship from Counsellors to Assignments.
- Represented by "Counsellor_Name" in "Assignments" (FK referencing Counsellor_Name in Counsellors).

Users to Assignments:

- Exactly one user can be assigned to exactly one counselor through the "Assignments" table. This is a one-to-one relationship from "Users" to "Assignments".
- Represented by "Username" in "Assignments" (FK referencing "Username" in "Users").

Users to Illnesses:

- Each user is associated with one illness. This is a one-to-one relationship from "Users" to "Illnesses".
- Represented by "Illness_Name" in Users (FK referencing "Illness_Name" in "Illnesses").

Counsellors to Illnesses (Through Counsellor_Illness):

- Counsellors can manage multiple illnesses, and each illness can be managed by multiple counsellors. This many-to-many relationship is resolved through the "Counsellor_Illness" bridge entity.
- "Counsellor_Illness" has two foreign keys: "Counsellor_Name" (referencing "Counsellor_Name" in Counsellors) and "Illness_Name" (referencing "Illness_Name" in "Illnesses").

4. Recognition of Issues

Many-to-Many Relationships:

- Counsellors and Illnesses: There is a many-to-many relationship between Counsellors and Illnesses, resolved through the associative entity `Counsellor_Illness`. Each counselor can handle multiple illnesses, and each illness can be handled by multiple counselors.
- Users and Counsellors/Illnesses: The entity `Assignments` manages the relationship between Users, Counsellors, and Illnesses,

potentially acting as a resolution for a many-to-many relationship if users can have multiple counselors for multiple illnesses.

NULL Values

- **Email, Password, Availability, Ratings in Counsellors:** These fields may not have NULL values as every counselor is required to provide this information.
- **Email, Password in Users:** Similar to “Counselors”, Users' emails and passwords may not be NULL as they are required to register user.
- **Illness_Name in Users:** This could be NULL if the user does not currently have an associated illness.

Multi-valued Attributes

- The diagram does not explicitly show any multi-valued attributes, as these would typically be resolved into separate entities or associative entities. For example, “Counsellor_Illness” resolves the potential multi-valued nature of counselors having multiple illnesses.

5. Final ERD

Chats:

The “Chats” table cannot be directly linked to the “Users” or “Counsellors” tables due to the presence of both “Sender” and “Receiver” attributes in the “Chats” table. This design allows either a user or a counsellor to act as the sender or receiver, creating a many-to-many relationship that requires an intermediary linking table to properly establish these associations in a relational database.

Final ERD

