# Technical Documentation

# Introduction

The community's mental health services are provided by John Hopkins Hospital. In more than a dozen specialized fields, the Johns Hopkins Hospital Psychiatry Department and Behavioural Sciences serves adults, teenagers, and kids.

To have a flexible medical service record and input the patients' medical histories, it is necessary to improve the inpatient unit's system and the mechanism for storing and managing the data of the physicians, nurses, and patients.

Increasing the number of patients and the medical personnel (doctors and nurses) will result in an increase in the amount of data that has to be managed in order to stay up with contemporary methods for information organization and management.

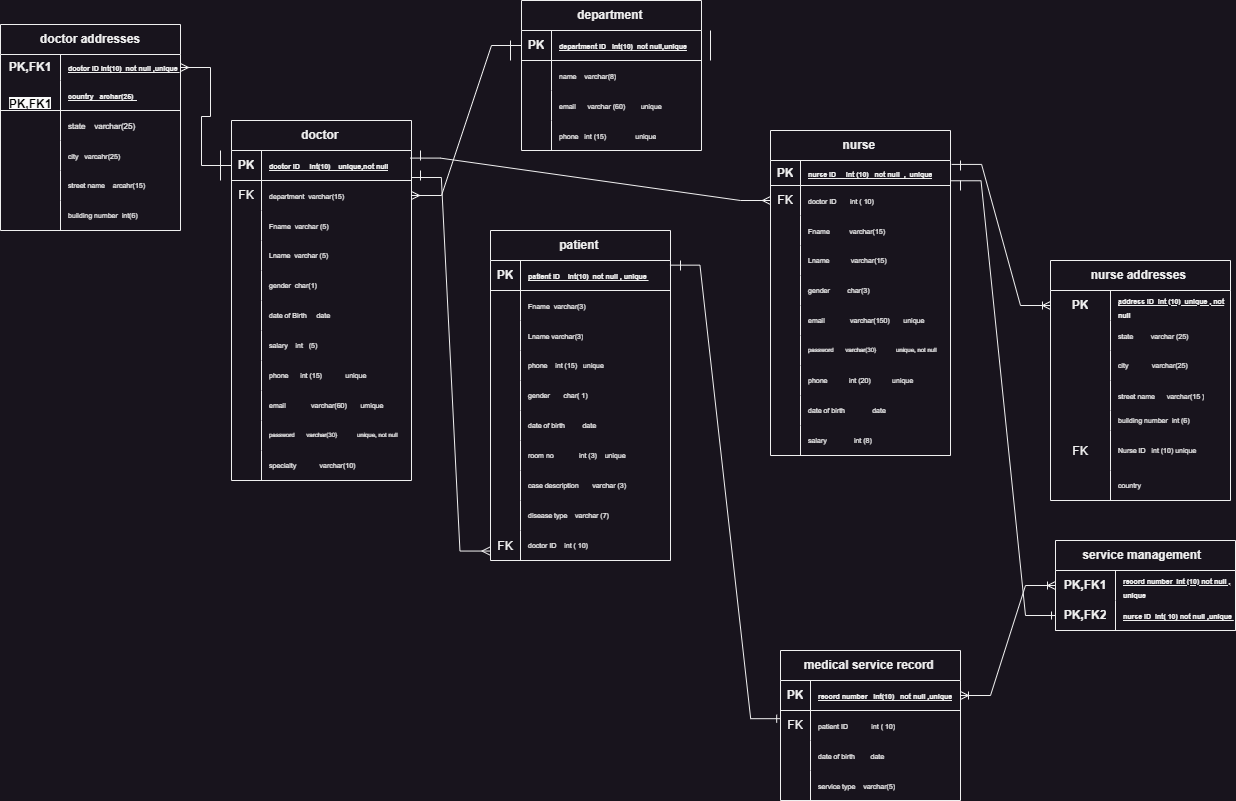
In order to manage this department's enormous volume of data, the hospital's board of directors has agreed to provide funds for database upgrades.

Additionally, it has a graphical user interface for the system's users (nurses, doctors, and patients) to make use of more effectively.

Features of the system: Nurses will deal with the records of medical services, and the system will notify them of the type of service that will be provided (checking, food, medicinal injection, etc.).

Note that the next section will go into more detail after a quick summary of the system's components and functions.

# Physical Schema:



# Database Development

## Database Overview

|  |  |  |
| --- | --- | --- |
| **Table** | **Name** | **Description** |
|  | Department | This table is the main table it's represent the department of Department of Psychiatry at Johns Hopkins Hospital that contain the (ID of the department, the name, email, phone) the unique value of all those is the ID (primary key) the constraint of it is not null and unique the other attributes the phone and the email is unique attributes. Also related with the doctors' table with one to many relationship |
|  | Doctor | This table describe the features that the doctors has in that department. The attributes are (ID, department name, first name, last name, gender, date of birth, salary, phone, email, speciality). The constraint of all those attributes the (primary key) which is the ID of the doctor and the department name as a foreign key from the department table also the email and phone are unique attributes. In addition this table has relation with department as we mention and the doctor addresses one to many relationship, patient's table with one to many relationship and with nurse's table one to many relationship |
|  | Doctor address | This table describe the multi addresses of the doctors and this table has (address ID, doctor id, state, country, city, street name, building number). the constraints are unique and not null for the primary key which is address ID and doctor ID is a foreign key from the doctor's table. There's no relationship between this table except that we mentioned early. |
|  | Patient | This table describe the information about the patient that contain (patient ID, First name, Last name, relatives phone, gender, date of birth, room no, case discerption, disease type, doctor ID). The constraints in this table are the primary key is patient ID, relatives phone is unique, room no is unique also doctor ID is a foreign key. This table is related with the doctor by a one to many relationships additionally there's relationship with the medical service recorded in in by one to many relationship. |
|  | Medical service record | This table describe the record of the services for the patient that they receive in the department this table contain (record number, patient id, date of birth, service type). The constraints are record is a primary key, not null and unique). This table link with the patient in one to one relationship and service management table with many to many. |
|  | Nurse | This table describe the feature that the nurses have it has (ID, doctor ID, last name, first name, gender, email, phone, date of birth, salary). The constraints like nurse which is the primary key not null and unique, the email unique and the email unique. This table has relationship with doctor as we mentioned one to many and the service management on to one also with nurse addresses with in one to many. |
|  | Nurse address | This table is to have the feature's addresses for the nurse it has (address ID, state, city, country, street name, building number, nurse ID). the constraints are the primary key is unique and not null also the nurse ID which is the foreign key. This table related only with the nurse ID one to many relationship. |
|  | Service management | This table is created from the mapping process contain (record number, nurse ID). The constraints are the both attributes primary and foreign key not null and unique. This table related with the nurse and service record by one to many relationship. |

|  |  |  |
| --- | --- | --- |
| **View** | **Name** | **Description** |
|  | Doctor salary | It's a view to display the doctors who's salary higher than 3000$ |
|  | LA nurse | This view display the nurses who live in Los Angeles by using view and join to get the data from two different tables |
|  | Anxiety | It's a view to display the patient's ID that have Anxiety. |
|  | Avg salary | The view count the average salary of the doctors |

|  |  |  |
| --- | --- | --- |
| **Procedure** | **Name** | **Description** |
|  | Adding patients | This Procedure add patient on the table |
|  | Doc\_info | This procedure pick the all info doctor which has less than 6000 |
|  | Food time | This procedure return the time of the service is food |
|  | remove nurse | This procedure remove nurse to the table |

## Security

|  |  |  |  |
| --- | --- | --- | --- |
| **User name** | **Privilege Command** | **Description** | **Screenshot** |
| Ethan | GRANT SELECT, INSERT, DELETE, UPDATE ON nurse.\* TO ethan; | This command give all the authority for ethan to show the table of nurses and do editing (remove, insert) also modify any data on it |  |
| GRANT SELECT, INSERT, DELETE, UPDATE ON nurse\_addresses.\* TO ethan; | This privilege give Ethan full access to the nurse address |  |
| GRANT SELECT, INSERT, DELETE, UPDATE ON doctor.\* TO ethan; | This command |  |
|  | GRANT SELECT, INSERT, DELETE, UPDATE ON patient.\* TO ethan; | This make ethan can make remove and modify also insert and show the data of the patients |  |
|  | GRANT SELECT, INSERT, DELETE, UPDATE ON service\_managment.\* TO ethan; | This table and all those privilege for it to be the controller of the service management and determine the nurse and the record that will organize |  |
|  | GRANT SELECT ON medical\_recored TO ethan; | This command give the right of the doctor to show the table of the medical record and monitor it |  |
| Mia | GRANT SELECT, UPDATE, INSERT, DELETE ON nurse\_addresses.\* TO Mia; | All the operation is given to Mia to insert ,remove ,update and select to the nurse address so Mia can do all the operation. |  |
| GRANT SELECT, UPDATE, INSERT, DELETE ON medical\_recored.\* TO Mia; | All the operation is available to Mia in the medical record and that because it's arrange and follow it |  |
|  | GRANT SELECT, UPDATE, INSERT, DELETE ON patient.\* TO Mia; | The fully access and all the authorize operation from selection, inserting, deleting and updating | D:\Univercity\DBD\DBD final\screenshots\19.png |
|  | GRANT SELECT ON patient TO Mia;  GRANT SELECT ON medical\_recored TO Mia;  GRANT SELECT ON service\_managment TO Mia;  GRANT SELECT ON nurse\_addresses TO Mia;  GRANT SELECT ON Anxiety TO Mia; | This commands to show medical record, service management ,nurse address,nurse and the view anxiety |  |
|  | GRANT SELECT ON medical\_recored TO ethan;  GRANT SELECT ON service\_managment TO ethan;  GRANT SELECT ON patient TO ethan;  GRANT SELECT ON nurse TO ethan;  GRANT SELECT ON nurse\_addresses TO ethan;  GRANT SELECT ON doctor\_addresses TO ethan; | And all those command to show the table to the user |  |
|  | GRANT SELECT ON LA\_nurse TO ethan;  GRANT SELECT ON Anxiety TO ethan; | In these commands the user can show the 2 views |  |
|  |  |  |  |

## User Interface:

### Flowchart and Data Movement Diagrams

### Interfaces Development

|  |  |  |  |
| --- | --- | --- | --- |
| **Page ID** | **Title** | **Description** | **Screenshot** |
|  | Service management | It's a title that describe the relationship between the nurse and the medical record so the nurse have organize and work according to the case nurse has so that record the service for the rooms with so many kinds of services that the nurse must provide |  |
|  | Nurse | It's a title to describe the information about the nurses with their email and phones also with the doctor that work with |  |
|  | Nurse addresses | It's a title to describe the multi address of the nurses |  |
|  | patient | The title is to represent the patient info |  |

# Maintenance

## Database recovery & backups

Recovery

Importing a Database:

1\_Select the target database from the left sidebar of PHPMyAdmin by opening it.

2\_The "Import" tab may be found on the top navigation menu.

3\_Select the backup file from your computer's local drive or enter the URL for the backup file.

4\_Choose the proper import parameters, including the character set, format-specific settings, etc.

To begin the import procedure, click the "Go" or "Import" button.

5\_The SQL commands in the backup file will be executed by PHPMyAdmin, restoring the database's structure and contents.

Backup

Exporting Database:

1\_Select the source database from the left sidebar of PHPMyAdmin by opening it.

2\_The "Export" tab will appear in the top navigation menu.

3\_Choose whether you want to export only a few tables or the complete database.

4\_The "Go" or "Export" buttons must be clicked to create the backup file.

5\_The chosen database or tables will be exported by PHPMyAdmin, and a backup file will be generated for you to download.

## Database maintenance in general

# Testing

## Data Validation

|  |  |  |  |
| --- | --- | --- | --- |
| **Number** | **Type** | **Description** | **screenshot** |
|  | All cases of PK | In the doctor table we have the nurse ID which is a primary key the value of it can't be null and I insert a null in it |  |
| In the doctor table we have the nurse ID which is a primary key the value can't be duplicated and I insert duplicate |  |
|  | All cases of FK | Cascade: change one of them to make it null which is the foreign key and see it in the another table |  |
|  |  |
|  | Unique | I got an value mustn't be double this example from the nurse table |  |
|  | Default | In this case I identify the gender in nurse table as M which is male I'll not inset a gender and let's see the result |  |
|  | Not null | In patient table we add a patient with value null which is the diseases |  |
|  | Check | This for checking operation in this test we going to have checking if the nurse is M or F |  |

## Output Validation

|  |  |  |  |
| --- | --- | --- | --- |
| **Number** | **Query Description** | **Screenshot (query + result)** | **Result validation** |
|  | This view calculate the avg salary for the doctors |  | We got the medical record that need a food because they're hungry |
|  | Selection of the patient that has anxiety |  | The output is get the whole row and the patient is Emma |
|  | Join the doctor and the department table to get the name of the doc also the ID and the department |  | The output get the doctors with their names and ID |
|  | Procedure that pick the docs that has salary more than 6000 |  | It's pick them with their names and ID and all the info about them |

## Security Validation

**Note**: you need to test the given and not given privileges.

|  |  |  |  |
| --- | --- | --- | --- |
| **Number** | **User Name** | **Description of privilege/no privilege** | **Screenshot (query + result)** |
|  | ethan | There is no privilege for this user to insert, update and delete in the medical record (let's try update) |  |
|  | Ethan | This user has full privileges to do all the operation on the service management |  |
|  | Mia | This user doesn’t have the right to insert, delete or update in the doctor table (it's not even exist) |  |
|  | Mia | This user have privilege to insert delete and update also for sure select from nurse-address(the address id is deleted) |  |

### 

## GUI Validation

|  |  |  |
| --- | --- | --- |
| **Number** | **Description** | **screenshot** |
|  | In this case we add a new patient from the nurse Mia and that should be add at both |  |
|  | We change the last name of one of the nurses called Emma |  |
|  | We can the select from the medical record |  |
|  | Mia just move from her house so we must drop the old address |  |

## Assess whether meaningful data has been extracted

## Assess the effectiveness of testing

# Evaluation of database solution

## Effectiveness of the database solution based on user and system requirement

## Suggested improvements

## Evaluation based on improvements needed