UTXJ UNIVERSIDAD TECNOLÓGICA DE XICOTEPEC DE JUÁREZ

UTXJ TECNOLOGÍAS DE LA INFORMACIÓN

PapuMedicos

REGISTROS MEDICOS

2024

Integrantes:

Arturo Aguilar Santos M-210716

Justin Martin Muñoz Escorcia M-210659

José Ángel Gómez Ortiz M-210562

Administración de Base de Datos

M.T.I. Marco Antonio Ramírez Hernández

Ingeniería en Desarrollo y Gestión de Software (IDGS)

Enero-Abril 2024

Objetivo General

Desarrollar un sistema integral de gestión hospitalaria que se centre en un módulo de Registro Médicos Electrónicos (EHR) específicamente diseñado para su implementación en hospitales privados. Este sistema tiene como meta primordial optimizar la recopilación, gestión y utilización de datos de pacientes, con el fin de mejorar la calidad de la atención médica, aumentar la eficiencia operativa y garantizar la seguridad de la información en el entorno hospitalario.

Objetivos específicos

Diseñar y desarrollar un módulo de Registro Médicos Electrónicos (EHR) altamente funcional y personalizable, que se integre de manera fluida con los sistemas de gestión hospitalaria existentes y cumpla con los estándares de seguridad y privacidad de datos.

Implementar funcionalidades avanzadas de análisis de datos en el módulo EHR, que permitan a los hospitales aprovechar el potencial de los grandes datos para poder identificar tendencias de salud y mejorar la toma de decisiones médicas.

Garantizar la accesibilidad y disponibilidad de la información del paciente mediante la creación de interfaces de usuario intuitivas y sistemas de búsqueda eficientes, que permitan a los profesionales de la salud acceder rápida y fácilmente a los registros médicos relevantes.

Mejorar la calidad de la atención médica al proporcionar a los proveedores de salud herramientas para personalizar los planes de tratamiento, monitorear la progresión de las enfermedades y prevenir complicaciones, todo ello basado en análisis precisos de datos clínicos.

Asegurar la seguridad y confidencialidad de los datos del paciente mediante la implementación de medidas de protección robustas, como la encriptación de datos, los controles de acceso y las auditorías de seguridad regulares.

Facilitar la interoperabilidad del sistema EHR con otras plataformas de salud electrónica, sistemas de información de laboratorio, sistemas de facturación y registros médicos externos, para garantizar una colaboración efectiva entre diferentes entidades de atención médica.

Capacitar al personal hospitalario en el uso efectivo del sistema EHR, proporcionando capacitación y soporte continuo para garantizar una adopción exitosa y un uso óptimo de la tecnología.

Evaluar periódicamente el rendimiento y la eficacia del sistema EHR mediante la recopilación de retroalimentación de los usuarios, análisis de métricas clave y revisión de indicadores de calidad de la atención médica, con el fin de realizar mejoras continuas y garantizar la satisfacción del cliente.

Promover la innovación y la excelencia en la atención médica mediante la investigación y el desarrollo continuo de nuevas funcionalidades y tecnologías en el ámbito de los registros médicos electrónicos, con el objetivo de mantenernos a la vanguardia de las tendencias y necesidades emergentes en el sector de la salud.

Un rol en MySQL es un conjunto de privilegios que puedes asignar a uno o varios usuarios. A diferencia de los usuarios individuales, que tienen permisos específicos sobre bases de datos y tablas, los roles permiten agrupar conjuntos comunes de privilegios y asignarlos fácilmente a múltiples usuarios.

En una base de datos de un hospital, hay varios roles distintos que podrían tener acceso y responsabilidades diferentes.

Administrador del Sistema: Este rol tiene el control total sobre la base de datos del hospital. Puede configurar permisos, realizar copias de seguridad, actualizar software y gestionar la seguridad de la base de datos.

Personal Médico: Este grupo incluye médicos, enfermeras, terapeutas y otros profesionales de la salud que necesitan acceder a la información del paciente para proporcionar atención médica. Su acceso puede estar restringido según la especialidad y el nivel de autorización.

Personal de Administración: Incluye administradores hospitalarios, personal de facturación y personal de recursos humanos. Tienen acceso a información relacionada con la gestión de pacientes, facturación, recursos humanos y otros aspectos administrativos del hospital.

Personal de TI: Este grupo es responsable de mantener la infraestructura tecnológica del hospital, incluida la base de datos. Pueden tener acceso para realizar tareas de mantenimiento y solución de problemas, pero sus privilegios pueden ser limitados en términos de acceso a datos sensibles del paciente.

Investigadores Clínicos: En algunos casos, los investigadores clínicos pueden necesitar acceder a datos de pacientes para realizar estudios médicos y análisis. Su acceso está limitado y generalmente requiere aprobaciones éticas y autorizaciones especiales.

Pacientes: Aunque no son parte del personal del hospital, los pacientes pueden tener acceso limitado a cierta información de su historial médico a través de portales en línea seguros o solicitudes directas al personal médico.

Como crear roles en la base de datos:

```
CREATE ROLE 'desarrolladores', 'lectores';
```

Los privilegios determinan qué acciones pueden realizar los diferentes usuarios o roles sobre los datos almacenados en la base de datos.

Privilegios de Lectura (SELECT): Permite a los usuarios ver datos en la base de datos. Por ejemplo, médicos y enfermeras podrían tener privilegios de lectura para acceder al historial médico de los pacientes.

Privilegios de Escritura (INSERT, UPDATE, DELETE): Permite a los usuarios agregar, modificar o eliminar datos en la base de datos. Este tipo de privilegio generalmente está restringido a roles específicos, como médicos para actualizar registros de pacientes o personal de facturación para ingresar información sobre facturación y seguros.

Privilegios de Administración (CREATE, ALTER, DROP): Permite a los usuarios realizar cambios en la estructura de la base de datos, como crear, modificar o eliminar tablas, índices o vistas. Este tipo de privilegio generalmente está reservado para administradores de bases de datos y personal de TI.

Privilegios de Gestión de Usuarios (GRANT, REVOKE): Permite a los administradores de bases de datos conceder o revocar privilegios a otros usuarios. Esto incluye el control sobre quién tiene acceso a qué datos y qué acciones pueden realizar.

Privilegios de Auditoría y Seguridad: Permite a ciertos usuarios acceder a registros de auditoría y registros de seguridad para monitorear quién accede a la base de datos y qué acciones realizan.

Privilegios de Copia de Seguridad y Restauración: Permite a los usuarios realizar copias de seguridad de la base de datos y restaurarla en caso de fallos o pérdida de datos.

Como dar privilegios a cada rol en la base de datos:

GRANT ALL ON basehospital.* TO 'desarrolladores';

GRANT SELECT ON basehospital.* TO 'lectores';

GRANT INSERT, UPDATE, DELETE ON basehospital.* TO 'desarrolladores'';

En una base de datos de un hospital, hay varios tipos de usuarios que pueden necesitar acceso por razones diferentes, puede haber usuarios que necesiten los mismos privilegios, es por eso que se crean los roles.

Como crear usuarios en la base de datos:

CREATE USER 'registrosmedicos'@'localhost' IDENTIFIED BY '1234';

CREATE USER 'pacientes'@'localhost' IDENTIFIED BY '1234';

CREATE USER 'cirujano'@'localhost' IDENTIFIED BY '1234';

Como dar rol a los usuarios en la base de datos sql:

GRANT 'desarrolladores' TO 'cirujano'@'localhost';

GRANT 'lectores' 'TO 'cirujano'@'localhost', 'pacientes'@'localhost';

GRANT 'lectores', 'desarrolladores' TO 'registrosmedicos'@'localhost';

Para respaldar una base de datos utilice el comando mysqldump:

$ mysqldump -u registrosmedicos -p basehospital > respaldo_db.sql

Como dar rol a los usuarios en la base de datos nosql:

user: Esta propiedad representa el nombre del usuario, en nuestro caso le asignaremos lector

pwd: Acá asignaremos la contraseña para el usuario.

roles: Un arreglo de objetos. Acepta un arreglo para el caso que tu usuario vaya a tener acceso a múltiples bases de datos.

El código quedaría de la siguiente manera:

db.grantRolesToUser("registrosmedicos", [{ role: "customRole", db: " basehospital " }]);

Para respaldar una base de datos nosql:

mongodump --db basehospital --out /ruta/donde/guardar/backup/