

Práctico Integrador – Programación Web 2

Un sistema hospitalario HIS (Hospital Information System, por sus siglas en inglés) es una plataforma de software diseñada específicamente para gestionar y administrar todas las operaciones y datos relacionados con un hospital o clínica. Estos sistemas tienen como objetivo principal mejorar la eficiencia, la calidad de la atención médica y la gestión de recursos en el entorno hospitalario.

Admisión y Recepción

El flujo de atención de pacientes para **internarse** en un hospital puede variar según las políticas y procedimientos específicos de cada institución médica. Sin embargo, a continuación, proporcionamos un flujo generalizado que describe los pasos típicos que un paciente debe seguir para ser admitido e internado en un hospital:

Registro y Preparación Previa:

El paciente puede llegar al hospital por diferentes vías, como una cita programada (paciente quirúrgico), una derivación médica o una situación de emergencia.

En algunos casos, el paciente puede haber sido derivado por un médico de atención primaria o especialista.

Recepción y Registro de Datos:

Al llegar al hospital, el paciente se dirige a la recepción o el área de admisión.

El personal de admisiones solicitará información personal, demográfica y de seguros médicos al paciente. También puede requerir identificación y documentación relevante.

Si es su primera atención en el hospital, se establecerá una cuenta de paciente sino se actualizará la información existente.

En el caso de un paciente derivado por la guardia del hospital se considerará ya registrada la atención del paciente por lo cual solo se registrarán los aspectos relacionados a la internación.

Asignación de Habitación:

Una vez completada la admisión se asignará una habitación al paciente en la unidad apropiada (por ejemplo, medicina interna, cirugía, cuidados intensivos, etc.). Se debe tener en cuenta que el hospital cuenta con una infraestructura de camas agrupadas en Alas y

Habitaciones. Las Habitaciones pueden alojar 1 o 2 camas. Al momento de asignar una cama esta debe estar libre e higienizada, además si es una habitación de 2 camas no podrá asignarse si la otra cama esta ocupada por un paciente de sexo diferente.

Evaluación Inicial por Enfermería

La evaluación inicial de un paciente por parte de un enfermero o enfermera es un paso fundamental en la atención de la salud y puede variar en su alcance y profundidad según la situación y la necesidad del paciente. A continuación, proporcionamos una descripción general de cómo procede una evaluación inicial el personal de enfermería.

Historial Médico y Antecedentes de Salud:

Después de haber verificado la información personal, incluyendo nombre, fecha de nacimiento, dirección y número de teléfono. También se verifica la información de contacto en caso de emergencia.

A continuación, el enfermero pregunta sobre el historial médico del paciente, incluyendo enfermedades previas, cirugías, alergias, medicamentos actuales y antecedentes familiares de enfermedades.

Es muy importante que toda esta información se registre de forma estructurada y sea fácil de hacerlo.

Luego se indaga sobre el motivo de la internación y los síntomas principales que experimenta el paciente. Esto ayuda a determinar la prioridad de la atención y las necesidades inmediatas.

Registro de signos vitales

El enfermero realiza una evaluación física básica, que puede incluir la medición de la presión arterial, la frecuencia cardíaca, la frecuencia respiratoria y la temperatura corporal. También puede observar el aspecto general del paciente, como el color de la piel y la respuesta a estímulos.

Plan de Cuidados Preliminar:

Con base en la evaluación inicial, el enfermero puede elaborar un plan de cuidados preliminar. Esto puede incluir intervenciones inmediatas, administrar medicamentos prescritos o proporcionar tratamiento para el alivio del dolor.

El enfermero se comunica con el equipo médico, si es necesario, para informar sobre el estado del paciente y coordinar el plan de atención.

La evaluación de enfermería es un proceso continuo. El enfermero sigue monitoreando al paciente a lo largo de su estancia y ajusta el plan de cuidados según sea necesario.

Evaluación Médica

El proceso de atención médica en pacientes internados en un hospital implica una serie de pasos y actividades coordinadas por un equipo multidisciplinario de profesionales de la salud. A continuación, se describe el proceso general de atención médica para pacientes internados

El equipo médico desarrolla un plan de atención individualizado para el paciente, que incluye diagnóstico, tratamiento, medicamentos, terapias y cuidados de enfermería.

Evaluación Médica y Diagnóstico:

El médico realiza evaluaciones médicas periódicas y solicita pruebas diagnósticas, como análisis de sangre, radiografías, resonancias magnéticas, etc., para obtener información adicional sobre la condición del paciente.

Tratamiento y Terapias:

El paciente recibe tratamientos médicos, terapias físicas u ocupacionales, y otros procedimientos según sea necesario para su recuperación.

Se administran medicamentos y se ajustan las dosis según las indicaciones médicas.

Se controla y alivia el dolor y otros síntomas del paciente, utilizando analgésicos y medicamentos apropiados.

El equipo médico evalúa continuamente la respuesta del paciente al tratamiento y ajusta el plan de atención según sea necesario.

Alta Hospitalaria:

Una vez que el paciente está lo suficientemente recuperado, se le da de alta del hospital.

Se proporcionan instrucciones de cuidado posterior, recetas médicas y recomendaciones de seguimiento médico.

Algunas consideraciones sobre los requerimientos

Debe tenerse en cuenta que la narrativa anterior no es una narrativa completa y que esta en el alumno consultar sobre las dudas que existieran. Ciertos requerimientos, aunque no están explícitamente detallados deben ser considerados en la implementación. Por ej. En ningún momento se detalla sobre la cancelación de una admisión, pero es totalmente posible que el

paciente se arrepienta de admitir en la clínica o también es posible que existiera algún error de carga por parte del usuario. En estos casos el sistema debe considerar estas situaciones para todos los casos de uso.

Objetivos del TPI

Este trabajo integrador tiene como objetivo principal que los estudiantes puedan desarrollar y poner en práctica los conocimientos adquiridos en la asignatura a partir del desarrollo de un caso de estudio.

Pautas de entrega y presentación

El alumno deberá entregar la aplicación subida a un repositorio github. Este repositorio deberá mantener el histórico tanto de las sesiones de trabajo diarias como así también las nuevas funcionalidades desarrolladas.

Importante!! No se aceptarán entregas cuyos repositorios no refleje el trabajo desarrollado en el proyecto. Por ejemplo, repositorios con pocos commits, commits excesivamente grandes, commits sin cambios significativos, etc

La aplicación deberá ser subida y presentada desde un servidor real en internet. Considere este aspecto desde el comienzo del proyecto. Considere los servidores disponibles para Node y la disponibilidad o no de servicios de gestión de bases de datos mysql o mariadb. Junto con las fuentes del proyecto se debe disponer de un readme que disponga de la información necesaria para desplegar el proyecto en un entorno local como así también un breve informe que cuente con que problemas se encontró y como los soluciono durante el desarrollo del proyecto.

Respecto de la BD, el proyecto debe tener en su raíz un backup sql que permita la creación/restoración de la BD. El backup debe tener los datos necesarios para poder hacer una prueba del sistema entregados (usuarios, registros de tablas auxiliares, registros de configuración, información de ejemplo, etc). De mas esta decir que deben existir tantos usuarios de pruebas como roles existan. La información de estos usuarios debe estar en el readme.

La BD debe estar correctamente configurada y puede ser evaluada durante la presentación del proyecto. Debe ser clara, robusta y bien documentada.

Se evaluará que:

- Las entidades y relaciones estén bien definidas.
- Que este normalizada mínimo 3FN
- Uso adecuado de Claves primarias e índices.

- Correcta integridad referencial
- Uso adecuado de tipos de datos.
- Implementación de restricciones (null, default, check, etc)

Respecto del desarrollo de la aplicación esta debe realizarse usando Node y Express como plataforma principal. La aplicación debe renderizar las vistas totalmente en el lado del servidor utilizando PUG.

La aplicación podrá utilizar cualquier librería que sea necesaria en el lado del servidor y para el front-end podrá reutilizar plantillas CSS, frameworks de CSS (Bootstrap, tailwind, Materialize u otros).

Importante!! No se aceptarán proyectos que utilicen desarrollos con frameworks del front como react, vue, angular o frameworks del back como Next.js, Nuxt.js, gatsby, etc.

Cualquier información necesaria para poder evaluar el trabajo presentado debe estar escrita en el readme del proyecto.

Condiciones para aprobar el proyecto

- El proyecto se defiende en mesa de examen para **Aprobar** la materia. La materia no es promocionable.
- Se valorará especialmente la creatividad y la correcta aplicación de buenas prácticas.
- El cumplimiento de los requisitos funcionales es indispensable para aprobar el trabajo.
- El código debe estar disponible en un repositorio remoto (GitHub o similar) para su revisión.
- La aplicación debe correr en un servidor accesible por internet.
- La documentación debe permitir la instalación y prueba del sistema sin necesidad de intervención del desarrollador.

Recomendaciones

- Realizar tempranamente un análisis del dominio de la aplicación y definir el **alcance** del proyecto.
- Evacuar todas las dudas respecto a los requerimientos funcionales en el comienzo del proyecto.
- Organizar de forma efectiva los archivos fuentes de la aplicación como así también la claridad del código escrito.

Criterios de evaluación del proyecto

Criterio	Descripción	Excelente (9-10)	Bueno (7-8)	Regular (4-6)	Insuficiente (1-3)	Ponderación (%)
1. Diseño y Arquitectura	Estructura organizada, separación clara en modelos, rutas y controladores. Uso de buenas prácticas de programación.	Estructura modular, código limpio y reutilizable.	División clara, pero con algunas redundancias.	Código funcional pero desordenado, con poca reutilización.	Código desordenado, sin modularidad clara.	20%
2. Base de Datos (MySQL)	Diseño relacional, normalización, claves primarias y foráneas, integridad referencial.	Modelo relacional correcto, sin redundancias y con restricciones.	Diseño adecuado, pero con algunas redundancias menores.	Modelo básico, con errores de diseño o falta de normalización.	Modelo deficiente, sin relaciones claras ni restricciones.	10%
3. Autenticación y Seguridad	Login con JWT o sesiones, encriptación de contraseñas, validaciones y protección contra inyecciones SQL.	Autenticación robusta, con validaciones y cifrado adecuado.	Autenticación funcional, pero con algunas carencias de seguridad.	Autenticación básica, con validaciones mínimas.	No implementa autenticación o es insegura.	10%
4. Funcionalidad y Operatividad	Gestión de pacientes, alta, baja, modificación, consulta y validación de datos obligatorios.	Cumple con todos los requisitos y funciona correctamente.	Cumple con la mayoría de los requisitos, con errores menores.	Cumple parcialmente con los requisitos, con errores importantes.	No cumple con los requisitos o la aplicación no funciona.	25%
5. Interfaz de Usuario (Opcional)	Diseño, usabilidad y correcta integración con el backend.	Diseño atractivo, fácil de usar y bien integrado.	Diseño aceptable, con algunas deficiencias de usabilidad.	Diseño básico, con poca integración con el backend.	No presenta interfaz o la integración es deficiente.	25%
6. Documentación	README con instrucciones, documentación de endpoints y comentarios en el código.	Documentación completa, clara y bien estructurada.	Documentación adecuada, aunque con algunos faltantes.	Documentación básica y poco clara.	No presenta documentación o es insuficiente.	10%