



ESTRUCTURAS DE DATOS 2024-10

Proyecto de aula

Juan Carlos Mariño Morantes

Facultad de ingeniería de sistemas e informática

Universidad Pontificia Bolivariana, Seccional Bucaramanga

Bucaramanga, febrero 5 de 2024

REVISIÓN Y CONTROL DE CAMBIOS

Revisión y versiones

Fecha	Versión Aprobada	Cargo	Nombres y apellidos
2024 – 02 – 05	1.0	Autor	Juan Carlos Mariño Morantes

Control de cambios

Fecha	Versión	Descripción del cambio	Autor
2024 – 02 – 05	1.0	Primera versión	Juan Carlos Mariño Morantes



PROYECTO DE AULA 2024 – 10

Este documento presenta los requisitos para la gestión de pacientes (citas y autorización de órdenes) de un centro de salud. Su objetivo es definir de manera clara y precisa todas las funcionalidades y restricciones del sistema que se planea desarrollar.

Propósito del sistema

En la ciudad de Bucaramanga se desea implementar un nuevo centro de salud privado abierto para el acceso de cualquier persona con el único requisito registrarse en la recepción y solicitar una cita con la especialidad deseada. Este sistema debe estar en capacidad de gestionar las colas de espera para citas y exámenes de distintas especialidades y el proceso de autorización de órdenes para exámenes dentro de las mismas especialidades disponibles.

Requerimientos

El sistema a desarrollar debe contar con los siguientes requisitos:

1. Una funcionalidad para el registro de pacientes tomando sus datos personales (Nombres, Apellidos, Edad, Identificación, número de teléfono), y asociar un ID de historia clínica en caso de tratarse de una primera visita.
2. Un sistema de gestión de citas, el cual recibirá como insumo la siguiente información:
 - Especialidad requerida.
 - Profesional asignado.
 - Motivo de la cita (valoración, examen o control).

Debe tenerse en cuenta que para crear una cita de tipo examen el paciente debe contar con una orden autorizada para este tipo de examen (ver numeral 6).

Una vez registrada la información básica, se debe generar un registro de ticket para el paciente, el cual debe contener un identificador representativo de especialidad, motivo de la cita y una secuencia que indique el turno del paciente (Ej: Valoración de medicina general - GV001. Donde G hace referencia a medicina general y V a valoración), la secuencia de cada especialidad debe ser independiente. Adicionalmente debe asignarse un valor para la consulta (las citas con médicos especialistas tienen un mayor costo que las de medicina general, los exámenes tendrán el valor asignado en el módulo de registro de órdenes (ver numeral 5) y los controles son gratuitos) y un flag asociado al ticket que indique si el valor de la cita ya fue pagado (Por defecto debe indicar No).

3. Una taquilla de pagos donde se ingrese el identificador de la cita y al registrar la transacción se actualice el flag de pago, este proceso deberá añadir cada uno de los IDs a una cola de espera (por especialidad) **en el orden que**

fueron pagados. Todos los tickets independientemente del costo o tipo de visita deben pasar por la caja para ingresar a la cola de espera.

Nota: Asuma que las transacciones se llevan a cabo en la caja, NO es necesario implementar un sistema de pagos.

4. Un sistema de llamado de pacientes a través del cual se notifique, a través del identificador de su ticket, que ya se encuentran habilitados para ser atendidos. Este proceso debe, tras ingresar un comando específico por especialidad, tomar un ID de la cola de espera (respetando el orden de ingreso) y mostrarlo en pantalla. Una vez se haya tomado un ID, este debe abandonar la cola para dar paso al siguiente ticket.
5. Un módulo de registro de órdenes, donde cada paciente se acerque para ingresar todos los exámenes que le hayan sido ordenados y deban autorizarse. Cada registro de orden deberá contener un identificador único, la especialidad a la que pertenece, el tipo de examen, el identificador del paciente al que está asignada, el costo del examen, una breve descripción y un flag que indique su estado (inicialmente nulo).

Nota: Los tipos de examen deben definirse con antelación y debe existir al menos un tipo de examen por especialidad (Ej: Cardiología - ECG).

6. Una funcionalidad de autorización de exámenes desde la cual accediendo por paciente, se pueda evaluar si se autoriza o no los exámenes que tenga pendientes. El resultado de este proceso debe ser la actualización del flag de estado de cada orden según la decisión tomada. **Tenga en cuenta que: solo se puede autorizar un examen a la vez, el acceso a las órdenes registradas debe realizarse usando un esquema LIFO.**

Requisitos del producto

- El sistema debe incluir un módulo de gestión de usuarios y permitir la autenticación mediante contraseña. Cada acción debe permitir la trazabilidad asociada al usuario de manera que se pueda garantizar el no repudio de sus acciones.
- La aplicación debe funcionar para el sistema operativo Ubuntu 23.04 LTS o Debian 12 o superior.
- Las vistas del sistema deben tener una distribución suficiente que se adapte al tipo de pantallas que hay en el negocio (I 202).
- La base de datos debe ser centralizada, se recomienda utilizar archivos de texto, XML o JSON o en lo posible un gestor de base de datos.

Requisitos del proyecto

- El lenguaje de programación es **JAVA**.
- El desarrollo debe respetar un patrón de diseño modelo vista controlador o **MVC** para cada módulo.

- Todo el proceso de diseño e implementación de las vistas de la aplicación debe contar con la opinión del cliente y este será quien autorice su puesta en marcha.
- Las únicas estructuras de datos autorizadas y la implementación de los algoritmos deberán ser de autoría de los miembros del equipo.
- El proyecto consta de los siguientes entregables: Especificación de requerimientos, diseño, código fuente, validación del software y evidencia de la puesta en marcha o implementación.
- Los equipos tendrán **máximo 3 estudiantes**.
- La metodología de desarrollo es libre, pero deberá sustentarse con evidencia suficiente para validar su implementación.
- El desarrollo del proyecto deberá estar enmarcado en el cronograma de las 11 semanas.
- El alcance de la propuesta estará en el marco de las competencias del proyecto descrito en este documento.
- Los equipos deben presentar un informe de seguimiento o bitácora semanal de avance del proyecto. La bitácora debe exponer en detalle la actividad desarrollada por cada miembro del equipo y el porcentaje de avance. Para la bitácora se recomienda utilizar alguna herramienta como GitHub o Trello.
- **La primera entrega** será la propuesta de proyecto con formato IEE o APA que incluye: portadas, introducción, planteamiento del problema, justificación, objetivos, marcos referenciales (conceptual, tecnológico y/o legal), metodología, resultados (Especificación de requerimientos y diseños) cronograma de actividades, referencias bibliográficas. Solo será aceptado si se evidencia el seguimiento mencionado en el ítem anterior.
- La sustentación del proyecto y **segunda entrega** será derivada de un documento con formato IEE o APA con todas las secciones que este requiera tales como: introducción, planteamiento del problema, justificación, objetivos, marcos referenciales (conceptual, tecnológico y/o legal), metodología, resultados (Especificación de requerimientos, diseños e implementaciones) , cronograma de actividades, referencia bibliográfica, etc.
- Las referencias bibliográficas deben manejar un porcentaje de 60% en inglés.

Equipos de trabajo

Equipo	NOMBRE
EQUIPO 1	SANTIS SILVA SOFIA
	PARRA FUENTES JUAN PABLO
	ALMEIDA RINCON JOHAN SEBASTIAN
EQUIPO 2	JAIMES CALA ROGGER SANTIAGO

	SALAZAR TRUJILLO CARLOS ALBERTO
	SERRANO PRADA JOSE DANIEL
EQUIPO 3	URIBE SANCHEZ TOMAS
	GOMEZ BUENO JUAN ANGEL
	GUTIERREZ MEJIA DANIEL FELIPE
EQUIPO 4	ARANGO DIAZ SAMUEL
	FLOREZ JULIO JEISON STEVEN
	BRAVO SANCHEZ ORLANDO ENRIQUE
EQUIPO 5	MALDONADO SAAVEDRA SANTIAGO
	TAPIAS FONSECA ANDERSSON JONEYDER
	PINZON CAMACHO ANDRES FELIPE
EQUIPO 6	PABON FLOREZ NESTOR STIVEN
	NIÑO OSORIO JUAN CARLOS
	FUENTES PATIÑO JUAN DAVID
EQUIPO 7	VEGA FERNANDEZ SAMIR ENRIQUE
	BENITEZ PAJARO JUAN EDUARDO
	CARVAJAL CALDAS MARIA ANGELICA
EQUIPO 8	ANTOLINEZ GUZMAN SANTIAGO
	ARENALES PEÑA RONALD EDUARDO
	GOMEZ MANOTAS JUAN CAMILO
EQUIPO 9	LOZANO SUAREZ MARIA ALEJANDRA
	ARIAS SIERRA DANIEL ORLANDO
	NEIRA RAMIREZ SEBASTIAN
EQUIPO 10	BARRAGAN SANCHEZ SILVIA JULIANA
	LAZARO GONZALEZ EDINSON FELIPE
	PICO GARCIA ANDRES FELIPE
EQUIPO 11	MURALLAS RUEDA JHON MARIO
	GUERRERO RUEDA JUAN MANUEL
	URQUIJO BAUTISTA JOSEPH CAMILO

EQUIPO 12	JAIMES MUÑOZ JUAN PABLO
	ORTIZ JAIMES LEYDER STEVEN
	CONTRERAS HERNANDEZ JUAN ANDRES
EQUIPO 13	BLANDON ARDILA RUBEN SANTIAGO
	PARRA GRAZT MARIA ANGELICA
	FUENTES BOTTIA ZARHET VALENTINA

Objetivos de aprendizaje

- Desarrollar habilidades de expresión oral.
- Desarrollar habilidades para la documentación técnica y para la elaboración de informes.
- Desarrollar habilidades para el liderazgo, planeación y ejecución de proyectos.
- Distingue los componentes generales de un sistema de información.
- Aplica metodologías de desarrollo de software.

Habilidades a evaluar

- Habilidades lectoras, de exploración sobre los temas requeridos y habilidades para escribir y citar fuentes académicas.
- Indagación de las temáticas involucradas para el desarrollo del proyecto en diversas fuentes e información.
- Elaboración de documentos.
- Conceptos de definición de arquitecturas tecnológicas.
- Identifica las redes, software y hardware, necesarias para el desarrollo del sistema.
- Comprende las diferentes estructuras computacionales existentes y la manera como se implementa en un lenguaje de programación establecido.
- Identificar cuál es la estructura de datos más pertinente respecto a la situación a abordar.
- Trabajo en equipo.
- Aprendizaje autónomo.
- Capacidad de diseñar e implementar aplicaciones.
- Capacidad de validar el funcionamiento de software.
- Elaboración de documentos y elaboración de artículos.
- Determina la manera como se comportan los sistemas en la vida real basado en la interacción de los componentes que hacen parte de este para representarlos mediante un algoritmo computacional.

- Modela situaciones y problemas de la vida real en ingeniería mediante la aplicación de las diversas estructuras de datos, y las implementa en un lenguaje de programación.

Fuentes de información

BELL Douglas y PARR Mike. Java para Estudiantes. Tercera Edición. México: Prentice Hall, 2003, 634 pág.

BOOCH, GRADY y RUMBAUGH, James, JACOBSON, Ivar. El Lenguaje unificado de modelado, Madrid: Addison-Wesley Longman, 1999, 432 pág.

CADENHEAD, Rogers. LEMAY, Laura. Sams Teach yourself Java 6. Fifth edition. 2007. Sams editorial.

CAIRÓ, OSVALDO. GUARDATI, SILVIA. Estructuras de datos. -- 3a ed. -- México: McGraw-Hill, 2006, 467 p.

DEITEL, Harvey M., DEITEL, Paul J., Cómo programar en java, Introducción al DOO con UML y lo patrones de diseño. Quita Edición, México: Pearson Prentice Hall. 2004, 1325 pag. - Libro Guía

DROZDEK, Adam. Estructuras de Datos y Algoritmos con Java. -- 2a.ed. -- Australia: Thomson, 2007.

ECKEL Bruce. Piensa en Java. Segunda Edición. Madrid: Pearson Education, 2002, 906 pág, Contiene CD ROM.

HERNÁNDEZ LAZARO, ROBERTO, [et al.] Estructuras de datos y algoritmos. -- Madrid: Prentice Hall, 2001. JAMSA

Kris. Aprenda y Practique Java. Oxford University Press. México: Oxford University press, 2000, 474 pág.

JOYANES AGUILAR, Luis, Programación orientada a objetos, Segunda Edición, Madrid: McGraw-Hill, 1998, 895 pág.

JOYANES AGUILAR, LUIS. Fundamentos de programación: Algoritmos estructuras de datos y objetos. -- 3a.ed -- Madrid: McGraw-Hill, 2003.

JOYANES AGUILAR, LUIS. Programación en C++, Java y UML. Primera Edición. México: McGrawHill, 2010.

MARTIN Rober C. UML para Programadores Java, Madrid: Pearson Education, 2004, 251 pág.

SCHACH Stephen. Análisis y Diseño Orientado a Objetos con UML. El proceso Unificado. McGraw-Hill. 2005.



VILLALOBOS, Jorge. Introducción a las estructuras de datos. Aprendizaje activo basado en casos. Pearson Prentice Hall, 2008.

Bases de Datos: <https://login.consultaremota.upb.edu.co/login>.