

Olimpiada de Programación 2023

Proyecto de Mejora del Código Azul

Escuela de Educación Técnica N°1

Localidad: Monte Grande

Provincia: Buenos Aires

Profesor responsable

Apellido y nombre: González Oscar Angel

Título de grado o profesorado: Ingeniero mecánico con orientación en mecatrónica o analista de sistemas

Cargo docente: Profesor

Correo electrónico: osgonzalez@abc.gob.ar

Teléfono celular (con prefijo jurisdicción): 1165775858

Estudiantes del equipo

Apellido y Nombre: Gauna Lucio

Especialidad que cursa: Técnico en Programación

Ciclo del curso: Ciclo superior

Año de cursada: 7mo 2023

Apellido y Nombre: Elizalde Agustín

Especialidad que cursa: Técnico en Programación

Ciclo del curso: Ciclo superior

Año de cursada: 7mo 2023

Apellido y Nombre: Florenciañez Pamela

Especialidad que cursa: Técnico en Informática Personal y Profesional

Ciclo del curso: Ciclo superior

Año de cursada: 7mo 2023

Apellido y Nombre: Leguizamón Agustín

Especialidad que cursa: Técnico en Informática Personal y Profesional

Ciclo del curso: Ciclo superior

Año de cursada: 7mo 2023

Índice

1. Introducción
2. Planificación y organización
3. Herramientas utilizadas, ventajas y desventajas
4. Base de datos
5. Página web
6. Aplicación móvil
7. Alternativas posibles
8. Componentes y requisitos computacionales
9. Testeos, algoritmos e interacciones
10. Errores y soluciones
11. Bibliografía

1) Introducción

Este informe presenta el proyecto del Sistema de Mejora del Código Azul, diseñado para automatizar y digitalizar los procesos críticos en situaciones de emergencia y cuidado de pacientes dentro de una clínica. Este proyecto aborda la necesidad de mejorar la eficiencia y la calidad de la atención médica en momentos críticos, brindando un enfoque más efectivo y tecnológicamente avanzado para la gestión de emergencias en el entorno clínico.

Este proyecto como objetivo tiene automatizar procesos repetitivos, mantener control de los datos y registros que se van generando y a su vez, poder tener la opción de modificarlos o eliminarlos, controlar el acceso a información frágil y poder ofrecer herramientas simples para problemas complejos.

2) Planificación y organización

Este proyecto incluye programación de aplicaciones móviles, diseño y programación web, documentación, diseño de base de datos, conexiones, entre otros. Podemos dividirlo en 4 áreas:

- Programación de la aplicación móvil
- Diseño web
- Base de datos
- Documentación

Somos 4 integrantes en el grupo, nos dividimos las tareas por área.

- Pamela. Documentación, reglamento del informe.
- Agustín Leguizamón. Registro de experiencia.
- Gauna Lucio. Documentación, desarrollo de la aplicación móvil y diseño de la base de datos.
- Agustín Elizalde. Creación de la base de datos y página web.

Planeamos el desarrollo del proyecto en base a la finalización de tareas.

3) Herramientas utilizadas, ventajas y desventajas


Utilizamos páginas web y softwares gratuitos para el desarrollo del proyecto.

- a) Android Studio. Aunque es un software pesado, es gratuito y fácil en gran medida la creación de la aplicación debido a sus herramientas, como la de poder emular la aplicación, interfaz para el diseño, etc.

1. Ventajas. La primera ventaja que tiene es que es gratuito, la segunda es que tiene diversas herramientas para programar, poder encontrar errores y diseñar, como la posibilidad de ver en tiempo real como funciona nuestro código mientras lo estructuramos.

2. Desventajas. La única desventaja que le encontramos es la exigencia de requisitos computacionales que necesita, nos limitó a programar en determinados momentos, como, por ejemplo, en horario escolar.


3. Alternativas. Tuvimos otras alternativas como IDEA de IntelliJ, o Eclipse (aunque previamente habría que configurarlo). No las elegimos debido a que creemos que no era necesario cambiarnos a otra IDE de código fuente.

 FIGMA. Un sitio web para la creación de pantallas de nuestra aplicación móvil y del sitio web.

1. Ventajas. Es un sitio web ligero y gratuito, tiene un amplio catálogo de herramientas y plugin para crear el diseño de las pantallas.

2. Desventajas. La versión gratuita es más limitada que la versión de paga.


3. Alternativas. Una alternativa puede ser Canva, pero Figma es un programa más adecuado debido a que fue hecho con ese propósito.

 Lucid chart. Un sitio web gratuito que lo utilizamos para la creación del diagrama entidad-relación de la base de datos, es una página web que tiene amplia variedad de plantillas de diagramas y herramientas para hacerlos.

1. Ventajas. Es gratis, es un sitio web ligero y además contiene herramientas que son fáciles de entender.

2. Desventajas. La versión gratuita tiene muchas limitaciones frente a la versión de paga.


3. Alternativas. Pudimos haber utilizado Enterprise Architect para los diagramas, pero no lo conocemos suficiente.

 XAMPP y PHPMyAdmin. Para la estructura de la base de datos, para administrarla y ver los datos que contienen

1. Ventajas. Son gratis, con variadas herramientas y ligeros. Simplifican en gran parte, la creación de un servidor.

2. Desventajas. No tienen la capacidad para soportar grandes sistemas.

3. Alternativas. El WAMPP es una alternativa que pudimos haber utilizado, pero no lo conocemos lo suficiente.

 Visual Studio Code. Un editor de código fuente gratuito que tiene varias herramientas útiles para el desarrollo del sitio web.

1. Ventajas. Es gratuito, tiene una gran cantidad de opciones para programar de manera eficiente, amplio catálogo de plugins y detección de errores.
 2. Desventajas. Puede ser muy exigente en ciertas computadoras, lo cual limita la velocidad de programar.
 3. Alternativas. Existen varias alternativas, NotePad++, SublimeText, entre otros. No elegimos otra debido a que estamos acostumbrados a trabajar con Visual Studio Code
- 4) Base de datos

La base de datos SQL que construimos está pensada para guardar los registros de los pacientes que se atienden, donde se atienden y tener un registro de ellos, además de poder registrar los reportes de estos y tener un control de los empleados y los usuarios autorizados al sistema, es completamente imprescindible para el sistema, sin ella no podemos mantener

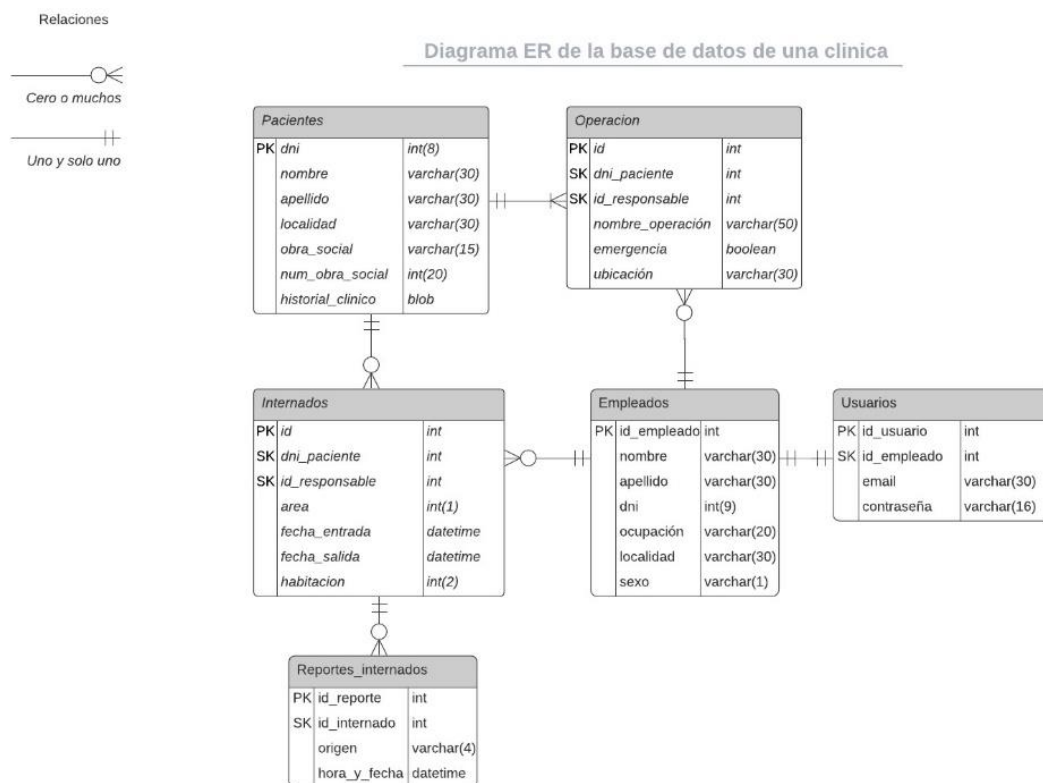


Ilustración 1

el sistema.

Es una realidad que al principio tuvimos una considerable cantidad de confusiones acerca de que debería incluir la base de datos y para que publique era dirigido, no es lo mismo una dirigida a controlar y registrar clientes, que una que forme el sistema de una clínica. Existieron dudas acerca de las tablas y campos que íbamos a incluir, pero que con el paso del tiempo logramos ponernos de acuerdo.

Esta base de datos se puede dividir en dos áreas, una es los pilares de la página web, donde sucede todo el tráfico de datos, la otra, que solo incluye las tablas de 'Usuarios' y 'Reportes_Internados' se utiliza para la aplicación móvil.

A continuación, una breve descripción de cada tabla.

- Pacientes. Es crucial para tener un registro de quien se atiende, se puede cargar un historial clínico en formato PDF.
- Operación. Registra las operaciones que se hacen, quien fue el enfermero responsable, donde ocurrió y si fue con turno previo o emergencia.
- Internados. Como su nombre lo indica, registra quien, cuando y donde estuvo internado un paciente luego de la cirugía.
- Reportes internados. Registra cuando y donde un paciente llamo por atención.
- Empleados. Registra los empleados de la institución, que cargo ocupan y datos necesarios.
- Usuarios. Guarda todas las cuentas de los empleados, es necesario para la aplicación móvil y la página web.

Todas las operaciones que conectan entre la base de datos, la página web y la aplicación móvil, fueron realizadas en PHP.

5) Pagina web

Las credenciales para acceder son:

- Para el administrador el usuario es 'admin', y la contraseña '123'
- Para el usuario comun es usuario 'usu', y la contraseña '123'

Si bien terminamos la página web, y es funcional, agrega, edita y elimina registros, no pudimos diseñarla completamente debido al poco tiempo que tuvimos. Es por ello por lo que tiene ciertas carencias en el apartado visual y navegable.

Seleccionamos los colores, tipografías e imágenes de acuerdo con el contexto de la institución. Colores claros de tonalidad azul y celeste, además de las imágenes y una tipografía general sencilla.

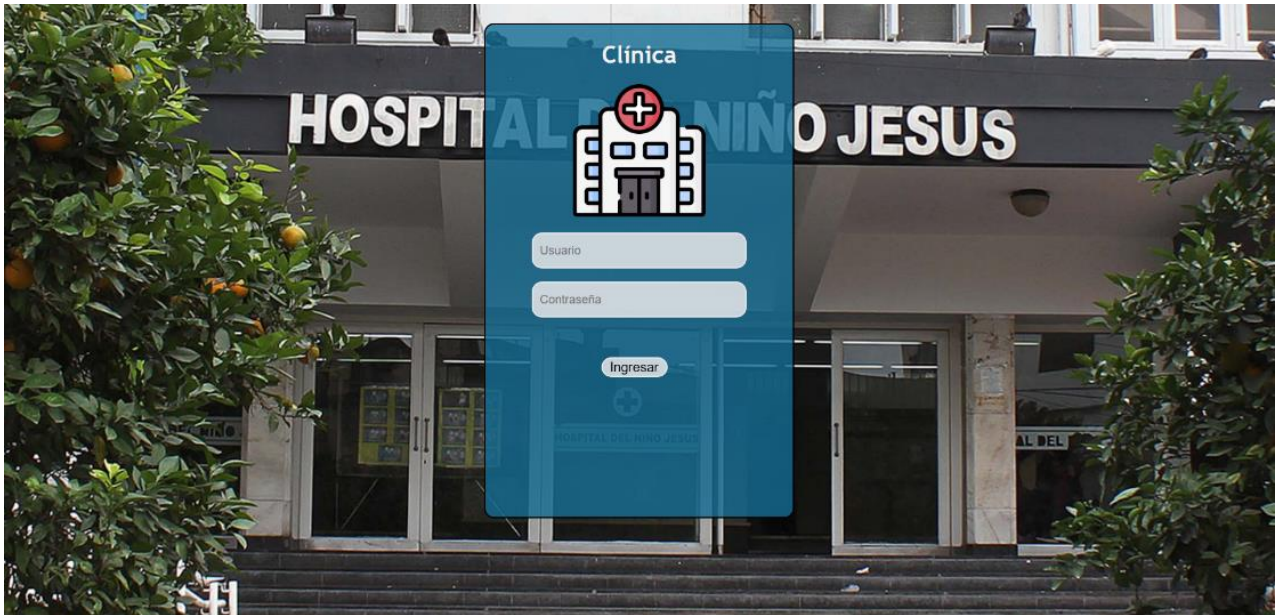


Ilustración 2

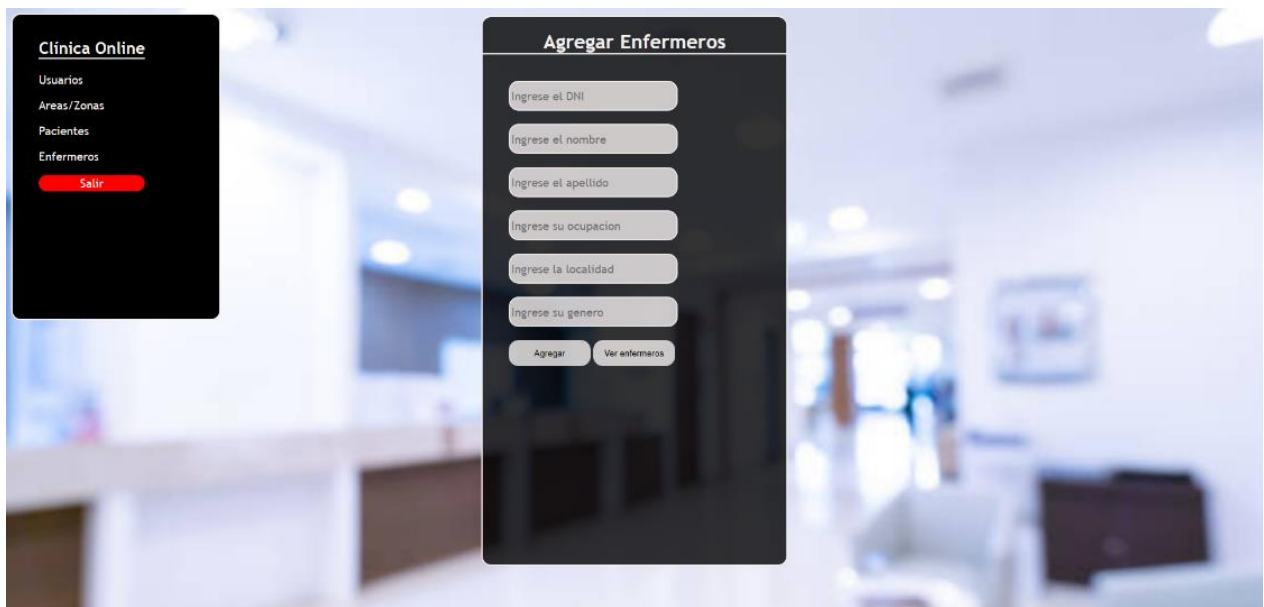


Ilustración 3

La página web es completamente navegable, esto significa que podemos recorrer todo el sitio web sin problemas, no hay pantallas que queden ocultas al entrar otras, excepto la pantalla de inicio de sesión que, por motivos obvios, no debe mostrarse más de una vez.

La estructura de la página web es en HTML, el estilo fue hecho en CSS y las conexiones a la base de datos fueron hechas con PHP.

A continuación, una lista de las operaciones que existen:

- Para agregar un enfermero al sistema 'agregar_enfermero'.
- Para agregar un paciente al sistema 'agregar_paciente'.
- Para agregar una zona al sistema 'agregar_zona'.
- Para asignar un enfermo a un usuario 'agregarEnfermerosUsuario'.
- Para agregar como paciente a un usuario 'agregarPacienteUsuario'.
- Para la validación del inicio de sesión 'login'.
- Para registrar un usuario al sistema 'registrar_usuario'.
- Para asignar un enfermero del sistema 'ver_enfermeros'.
- Para validar un paciente del sistema 'ver_pacientes'.
- Para asignar un usuario del sistema 'ver_usuarios'.
- Para asignar una zona del sistema 'ver_zonas'.
- Para ver los enfermos del sistema 'ver_Enfermeros'.
- Para pedir enfermero desde la vista del usuario 'verEnfermerosUsuario'.
- Para ver los pacientes del sistema 'verPacientes'.
- Para validar paciente desde el lado del usuario 'verPacienteUsuario'.
- Para ver los usuarios del sistema 'verUsuarios'.
- Para ver las zonas cargadas al sistema 'zonas'.

6) Aplicación móvil

Fue lo primero que definimos porque era el trabajo más claro que teníamos, hacer una aplicación solamente para el personal autorizado, el propósito era mostrar todas las llamadas que habían hecho los pacientes en la sala de internación. Empezamos diseñando la aplicación en FIGMA, no hubo grandes confusiones ni varios temas en los cuales debamos establecer lo que debíamos hacer.

Nuevamente, diseñamos de acuerdo con el contexto institucional que se presentaba, colores claros, contraste de colores, formas amigables, tipografías



Ilustración 4

sencillas y pantallas claras.

Los colores que usamos son los siguientes:

- Azul oscuro #104384
- Azul claro #597EB3
- Celeste #C2DBFF
- Naranja claro #FAC35A
- Marrón #AD7507

Tipografía:

Archivo

La aplicación la desarrollamos en Android Studio con el lenguaje Java.

Tuvimos que enfrentar muchos problemas debido a que programar una aplicación móvil, aunque sea una simple, es complicado por la complejidad del código y como todo se conecta entre sí, además de lo visual, lo lógico fue bastante más complicado.

Para acceder a la aplicación móvil, es necesario entrar con el email y contraseña de un usuario ya registrado en la base de datos.

7) Alternativas posibles

Tuvimos alternativas según la tarea que teníamos, podemos dividirlo por ítem:

- Sistema general

En vez de hacer una página web pudimos haber hecho un programa en java. No decidimos hacer esto por el tiempo que teníamos y por la experiencia que tenemos con el desarrollo con las páginas web y las conexiones a las bases de datos, además de que hacer un programa lleva más tiempo.

- Aplicación móvil

En vez de hacer una aplicación móvil, pudimos hacer una aplicación web para ingresar desde cualquier lugar, el problema aquí es que no tenemos experiencia creando aplicaciones web complejas como la que hicimos actualmente. Si bien hicimos sitios web responsive como la página actual, no tenemos experiencia en optimizar una página web para un celular o todo tipo de celulares.

8) Los requisitos mínimos para la aplicación móvil son los siguientes:

- Versión de Android: La versión mínima de Android es la 8.0
- Procesador: Al menos un procesador de 1 GHz o superior.
- Memoria RAM: Al menos 1 GB de RAM.
- Almacenamiento interno: 50MB
- Conexión a Internet: Si, es necesario.
- Permisos: Permiso a conexión internet.

Requisitos mínimos para la página web:

- Sistema operativo. Windows 10 o superior.
- Memoria RAM: 4GB en adelante
- Almacenamiento: 512MB para la página, pero es necesario que tenga mínimo 20GB para el sistema operativo
- Conexión a internet: Si

9) Testeos, algoritmos e interacciones

A continuación, se mencionará las interacciones y algoritmos de la aplicación móvil:

MainActivity:

Interacción con el usuario (Líneas 29-40): Aquí, se interactúa con el usuario para obtener su correo electrónico y contraseña. Cuando el usuario hace clic en el botón "Login", se inicia una acción basada en eventos (evento de clic) que desencadena la validación del usuario.

Validación de Usuario (Líneas 42-75): En esta sección, se envían los datos del usuario (correo electrónico y contraseña) a un servidor web (mediante Volley) para su validación. Esto implica la interacción con una base de datos en el servidor.

Llamadas:

Obtención de Datos (Líneas 23-43): La actividad Llamadas se encarga de obtener datos de un servidor web (mediante Volley) en formato JSON. Esto implica una interacción con una base de datos en el servidor para recuperar datos de llamadas de emergencia.

Procesamiento de Datos (Líneas 45-70): Después de recibir la respuesta del servidor, se procesa el JSON para convertirlo en objetos entidades_reportes. Esto implica algoritmos para analizar y transformar datos JSON en objetos Java.

entidades_reportes:

Definición de Clase (Líneas 3-24): Esta clase define la estructura de datos para representar un informe de llamadas de emergencia. No es un algoritmo en sí, pero define una estructura de datos importante para la aplicación.

ReportesAdapter:

Configuración de la Vista (Líneas 20-39): En esta clase de adaptador, se configura la vista de cada elemento en el RecyclerView para mostrar datos de informes de llamadas. Si bien no es un algoritmo en sí, es importante para la visualización de datos en la interfaz de usuario.

10) Errores y soluciones

Errores que ocurrieron con la aplicación móvil.

- a) En la pantalla de las llamadas no se veía más de un registro, esto ocurría por el tamaño que definimos de los 'TextView', estaba en la medida de DP, cuando lo cambiamos a 'wrap_content' pudimos visualizar los otros registros.

- b) Tuvimos múltiples avisos de posibles errores con el nombramiento de variables por la letra 'ñ'. Preferimos cambiar todas las palabras que contengan esa letra por alguna alternativa. En general, pasamos de tener una variable 'contraseña' a que se llamen 'contraseña'.
- c) La aplicación crasheaba cada vez que se abría, descubrimos que en versiones posteriores a Android 9 se bloquea una aplicación al contener conexiones HTTP no seguras, teníamos dos soluciones, la primera es la más segura, y es habilitar HTTPS en el servidor, para esto necesitábamos un certificado SSL válido para el servidor web, algo que claramente, con el tiempo que teníamos, no podíamos hacer, así que simplemente la línea de código: `android:usesCleartextTraffic="true"`

11) Bibliografía

Guerrerosystem (2021). Listar datos. Recuperado el 19 de septiembre, de https://www.youtube.com/watch?v=scX4KKiQ8q4&ab_channel=guerrerosystem

Developeru (2018). Playlist base de datos MySQL. Recuperado el 20 de septiembre de 2023, de <https://youtu.be/i-eKtws0EnI?si=x0ljXuasd5ESEGa4>

Joanolo (2013). XAMPP error. Recuperado el 22 de septiembre, de <https://stackoverflow.com/questions/15169861/xampp-error-mysql-shutdown-unexpectedly>

Ant Code (2020). Android Request no reconocido. Recuperado el 21 de septiembre, de https://www.youtube.com/watch?v=KI67yYd6m94&t=2s&ab_channel=AntCode