



Algoritmos y lógica computacional

Johan Felipe molina Aguirre

Facultad de Ingenierías 2024-1

Ingeniería de software

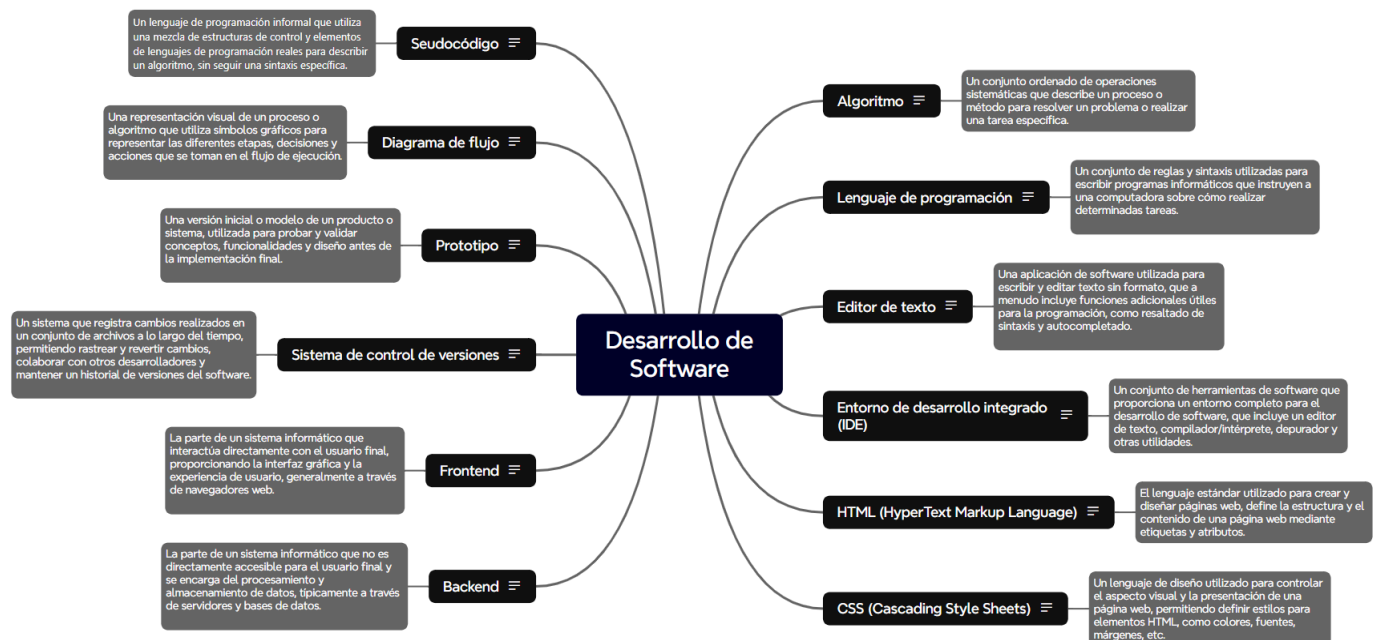
Armenia, Quindio

2024

SOLUCIÓN

1) Haga un mapa conceptual de los siguientes conceptos:

Algoritmo
 Lenguaje de programación
 Editor de texto
 Entorno de desarrollo integrado
 HTML
 CSS
 Backend
 Frontend
 Sistema control de versiones
 Prototipo
 Diagrama de flujo
 Pseudocódigo



(En caso de querer leer mas detalladamente comunicarse con Johan Molina para pasar una copia del archivo original)

2) Describir una posible solución.

Nombre de la Aplicación: SanitasVicus - Armenia

La aplicación SanitasVicus - Armenia se propone como una solución innovadora para abordar el problema de acumulación de basura en las calles de Armenia. Con el objetivo de mejorar la gestión de residuos, SanitasVicus - Armenia ofrece una serie de funcionalidades clave para ayudar a los ciudadanos a contribuir activamente a un entorno más limpio y saludable.

Una de las principales características de SanitasVicus - Armenia es su mapa interactivo de contenedores de basura. Esta función permite a los usuarios encontrar fácilmente el contenedor más cercano a su ubicación actual, contarán con un rastreador en tiempo real para evitar cualquier tipo de pérdida. Además, se proporcionará información útil sobre cada contenedor, como horarios de recolección y su capacidad actual esto por medio de la aplicación.

SanitasVicus - Armenia también incluirá un sistema de reportes que permitirá a los usuarios informar cuando un contenedor esté lleno o en mal estado. La idea es que estos reportes se envíen automáticamente al departamento de gestión de residuos, asegurando una acción inmediata para resolver el problema y evitar desborde de basuras en los contenedores así como los malos olores.

Para recordarle a los usuarios las fechas programadas de recolección de basura en su área, SanitasVicus - Armenia enviará notificaciones un día antes a todos los dispositivos que cuenten con la aplicación. Esta función es fundamental para evitar la acumulación innecesaria de desechos en las calles y así resolver la problemática de los malos olores y la contaminación, así como mantener una excelente higiene.

Además de estas funcionalidades prácticas, SanitasVicus - Armenia también se enfoca en educar a los usuarios sobre la importancia del reciclaje y la correcta disposición de residuos. La aplicación proporcionará consejos prácticos y contenido relevante para fomentar un comportamiento más eco-consciente entre los ciudadanos de la ciudad de Armenia.

Como características adicionales, SanitasVicus - Armenia ofrecerá un seguimiento de reportes para que los usuarios puedan verificar el estado de los problemas que han reportado, así como un sistema de recompensas para incentivar la participación activa.

Desde el punto de vista tecnológico, SanitasVicus - Armenia se desarrollará como una aplicación móvil para plataformas iOS y Android, utilizando tecnologías como React Native (es una tecnología que permite

a los programadores realizar aplicaciones multiplataformas con una buena eficiencia utilizando javascript para garantizar un buen rendimiento) para garantizar una rápida interacción y buen rendimiento. Se integrarán servicios de geolocalización y se utilizarán servicios en la nube para gestionar la base de datos y las notificaciones. Se implementarán medidas de seguridad y privacidad de datos para proteger la información de los usuarios.

En resumen, SanitasVicusus - Armenia tiene como objetivo mejorar la gestión de residuos en Armenia al proporcionar a los ciudadanos herramientas prácticas y educativas para mantener las calles limpias y promover un estilo de vida más eco-consciente.

2.2) Pseudocódigo.

Inicio

Mostrar "¡Hola! Bienvenido a SanitasVicusus - Armenia y contribuir al medio ambiente "

Mostrar "Por favor, sigue las instrucciones para deshacerte de tus residuos."

Mostrar "¿Tienes una bolsa de basura lista? (Sí/No)"

Leer respuesta bolsa

Si respuesta bolsa es igual a "No" entonces:

Mostrar "Por favor, consigue una bolsa de basura y vuelve cuando estés listo."

Fin Si

Mostrar "¿Dónde tienes la basura para desechar?"

Mostrar "1. Cocina"

Mostrar "2. Baño"

Mostrar "3. Habitación"

Mostrar "4. Otro lugar"

Leer ubicación basura

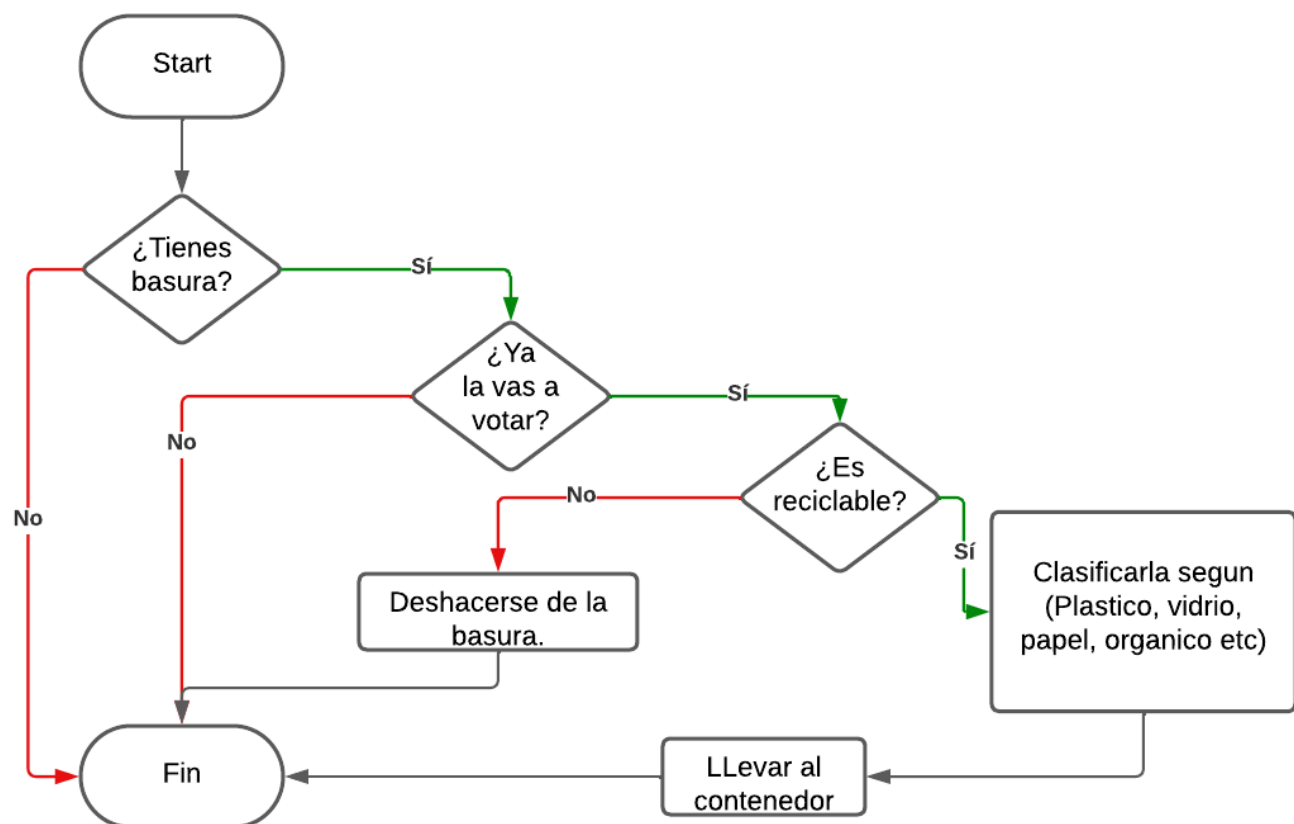
Mostrar "Por favor, saca la bolsa de basura y deposítala en el contenedor principal de residuos fuera de tu hogar."

Mostrar "Recuerda botarla en el contenedor correspondiente:"

Mostrar "- Si tienes residuos orgánicos, deposítelos en el contenedor marrón."
Mostrar "- Si tienes plásticos, deposítalos en el contenedor amarillo."
Mostrar "- Si tienes vidrio, deposítalo en el contenedor verde."
Mostrar "- Si tienes papel o cartón, deposítalo en el contenedor azul."
Mostrar "- Si tienes otros tipos de residuos, colócalos en el contenedor adecuado según las indicaciones locales."

Mostrar "¡Gracias por participar en este increíble cambio! ¡Hasta luego!"
Fin

DIAGRAMA DE FLUJO



2.3) MOCKUP



Página de Acceso



Página para Iniciar sesión



Página de Registro



Página de Inicio



Notificaciones



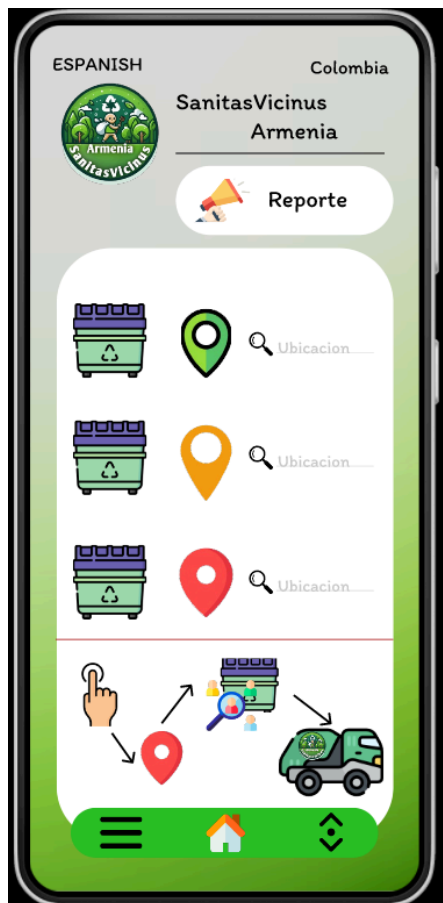
Menú



Clasificación



Aprende sobre el reciclaje y el correcto desecho de residuos



Página de Reportes.

2.4) UN VIDEO ADJUNTADO

2.5)

Cómo subir un repositorio a github.

Configurar el entorno de trabajo.

Trabajamos con visual studio code que sería nuestro editor de textos.

Configurar mi repositorio.

Un sistema de control de versiones: git y en la nube Github

Crear un repositorio:

Primero verificamos tener git y visual studio code con sus respectivas extensiones para mayor facilidad.

Creamos una carpeta desde archivos y la abrimos con “Open git bash here”

1. Ponemos el comando: git init

- 1.1 después: git config --global user.name “usuario”

- 1.2 después: git config --global user.email correo

- 1.3 Ponemos: code .

Ya en visual code studio creamos una carpeta que podemos llamar Archivo.txt o Archivo.html o Archivo.js

2. Volvemos a git y ponemos : git add . _____ (El nombre del archivo)

3. Después: git commit -m _____ (Mensaje, podría ser first commit)

4. Después: git remote add origin _____ (link de github, para esto ya tiene que tener una cuenta en github y haber creado un repositorio nuevo)

5. git push -u origin master

Ya por último solo queda verificar que sí se haya subido el archivo.

En caso de ser requerido un token. Antes de hacer push se debe de poner “git remote set-url origin https://Tu_Token@github.com/USUARIO/REPOSITORIO.git”

2.6) Tablero (Solo)

Trabajo en equipo

¿Qué aprendí? sobre el trabajo en equipo

Del trabajo en equipo aprendí el hecho de que todos deben aportar por igual. Si no nos sentimos bien en el ambiente de trabajo en el que nos encontramos, no podemos forzar nada. Nadie es jefe de nadie y cada quien es responsable de lo que pueda aportar. En un trabajo en "equipo" se debe de poner como prioridad ya que es por el bien común y mas de uno puede salir afectado en caso de no cumplir con lo que le toque aportar. En un equipo de trabajo se pueden adquirir muchas habilidades o se pueden complementar de unos a otros. Con estas habilidades se puede obtener experiencia y lograr un buen trabajo para obtener los resultados mas exquisitos.

¿Qué aprendí? sobre el tema

Del tema aprendí bastante, así como utilizar Inteligencias artificiales correctamente no para que me realicen todo el trabajo, sino para que me complementen y me expliquen paso a paso y detalladamente cómo elaborar todo el trabajo de forma correcta y teniendo como enfoque completamente mis ideas y habilidades. Desarrollé nuevas habilidades con herramientas como Xmind y Figma, así como chat GPT y día a día conociendo muchos más atajos del teclado y páginas que facilitan el desarrollo de todo este proceso.

