

PROYECTO DE AULA.

Alumnos: Sebastián José Guerra Martínez, Héctor Luis Ríos Cuadro.

Universidad Popular del Cesar

Facultad de Ingeniería

Programación de Computadores II

Ing. Roberto Fernández

Valledupar, Colombia

05/06/2024

Tabla de contenido

Fase No 1: Identificación, formulación, comprensión y especificación del problema.

Identificación y formulación del problema o necesidad.

- Formulación del problema
- Descripción del problema o necesidad identificada.
- Causas y consecuencias del problema.
- Área de aplicación del problema.
- Fuentes: Incluir referencias bibliográficas o web que sustenten la información presentada.

Estado del arte / Vigilancia tecnológica.

- Revisión de soluciones existentes
- Enumeración de soluciones tecnológicas que abordan el problema identificado.
- Descripción breve de las funciones de cada solución.
- Fuentes: Incluir referencias bibliográficas o web que sustenten la información presentada.

Diseño preliminar

- Modelo computacional basado en UML.
- Identificación de entidades.
- Listado de las clases que conforman la solución.
- Descripción breve de cada clase.
- Diseño UML.
- Diagrama UML para cada clase, incluyendo nombre y atributos.
- Métodos (opcionales).
- Diagrama de clases preliminar.
- Identificación de relaciones entre clases (agregación y composición).
- Diagrama general de clases que incluya todas las clases y sus relaciones.
- Notación UML adecuada.
- Implementación de código.
- Avance en el desarrollo del programa.
- Implementación de las clases del diagrama de clases.
- Prueba y funcionamiento de objetos de cada clase.

Fase No 2: Definición de la arquitectura del proyecto.

- Diagrama de clases mejorado.
- Redefinición de clases y atributos (si es necesario).

- Inclusión de herencia, clases abstractas e interfaces.
- Definición de la arquitectura (Diagrama de paquetes).
- Diseño preliminar de interfaces gráficas de usuario (mockup o wireframe).
- Funcionalidad completa del aplicativo en modo consola y colecciones de objetos (ArrayList).

Fase No 3: Desarrollo del producto final de software.

- Software final funcional.
- Proyecto implementado con todas las funcionalidades.
- Interfaces gráficas de usuario (GUI).
- Persistencia en archivos.

Fase (1): Identificación, formulación, comprensión y especificación del problema.

1. Identificación y formulación del problema o necesidad.

PROBLEMA O NECESIDA	CAUSAS	CONSECUENCIAS	ÁREA DE APLICACIÓN
<p>Con el fin de dar soporte al proceso de inscripción de cédulas de la Registraduría General de la Nación, se desea desarrollar una aplicación para las próximas elecciones presidenciales. Cada cédula debe ser registrada en un puesto de votación existente. Al momento de registrar una cédula, se debe ingresar la siguiente información del ciudadano:</p> <p>Número de cédula, Fecha de expedición de la cédula, Lugar de expedición de la cédula, Apellidos, Nombres, Fecha de nacimiento, Lugar de nacimiento, Sexo.</p> <p>El sistema de inscripción debe contener la información de los departamentos. Cada departamento tiene</p>	<p>La razón principal de la necesidad de este programa es optimizar y asegurar el proceso de registro de votantes, que es esencial para elecciones transparentes, eficientes y seguras. La adopción de un sistema automatizado y centralizado no sólo mejorará la gestión y la seguridad de los datos, sino que también facilitará el acceso de los ciudadanos al proceso electoral, aumentando la confianza y la participación pública.</p>	<p>Las consecuencias positivas incluyen un proceso de inscripción más eficiente, una mejor gestión y protección de la información, mayor accesibilidad para los ciudadanos, mayor transparencia y confianza pública, y un sistema escalable y adaptable para futuras necesidades. Estas mejoras garantizarán elecciones más transparentes, eficientes y seguras.</p>	<p>El área de aplicación de este programa es amplia y abarca aspectos críticos del sistema electoral, administración pública, servicios al ciudadano, seguridad informática, auditoría y control, así como planificación y logística. Su implementación no solo mejorará la eficiencia y transparencia del proceso de inscripción de cédulas, sino que también contribuirá al fortalecimiento de la democracia y la confianza pública en el proceso electoral.</p>

sus municipios y cada municipio tiene los puestos de votación disponibles con sus ciudadanos inscritos.			
---	--	--	--

2. Estado del arte / Vigilancia tecnológica.

Nº	FUNCIONALIDADES	TotalVote: KNOWiNK	Plataforma de Civix: Civix	PowerProfile: Election Systems & Software
1	Destaca por su enfoque en la noche electoral y su portal informativo para votantes.	X	X	X
2	Ofrece capacidades avanzadas de GIS y herramientas analíticas robustas.	X	X	X
3	Fuerte en la integración con agencias externas para la verificación en tiempo real y en su interfaz GIS para la precisión de direcciones.	X	X	X
4	Permiten el registro de cédulas con información detallada del ciudadano.	X	X	X
5				

	Manejan la jerarquía de departamentos, municipios y puestos de votación.	X	X	X
6	Ofrecen interfaces fáciles de usar para ciudadanos y funcionarios.	X	X	X
7	Proveen herramientas para generar reportes detallados y asegurar la transparencia en el proceso electoral.	X	X	X
8	Incluyen mecanismos para asegurar la integridad de los datos y prevenir inscripciones fraudulentas.	X	X	X

3. Diseño preliminar

Entidades identificadas:

- **Votante:** Representa a un votante con atributos como nombre completo, cédula, fecha y lugar de expedición, fecha y lugar de nacimiento, sexo, departamento, municipio, lugar de votación y dirección.
- **SistemaInscripcion:** Representa el sistema de inscripción que maneja una lista de votantes. Utiliza el patrón singleton para asegurar que solo haya una instancia de esta clase.
- **Departamento:** Representa un departamento con atributos como código y nombre del departamento.
- **Municipio:** Representa un municipio y extiende de la clase Departamento. Tiene atributos adicionales como código y nombre del municipio y el nombre del departamento al que pertenece.

- LugarVotacion: Representa un lugar de votación y extiende de la clase Municipio. Tiene atributos adicionales como código del lugar, nombre del lugar de votación, dirección y el nombre del municipio al que pertenece.

Diseño UML

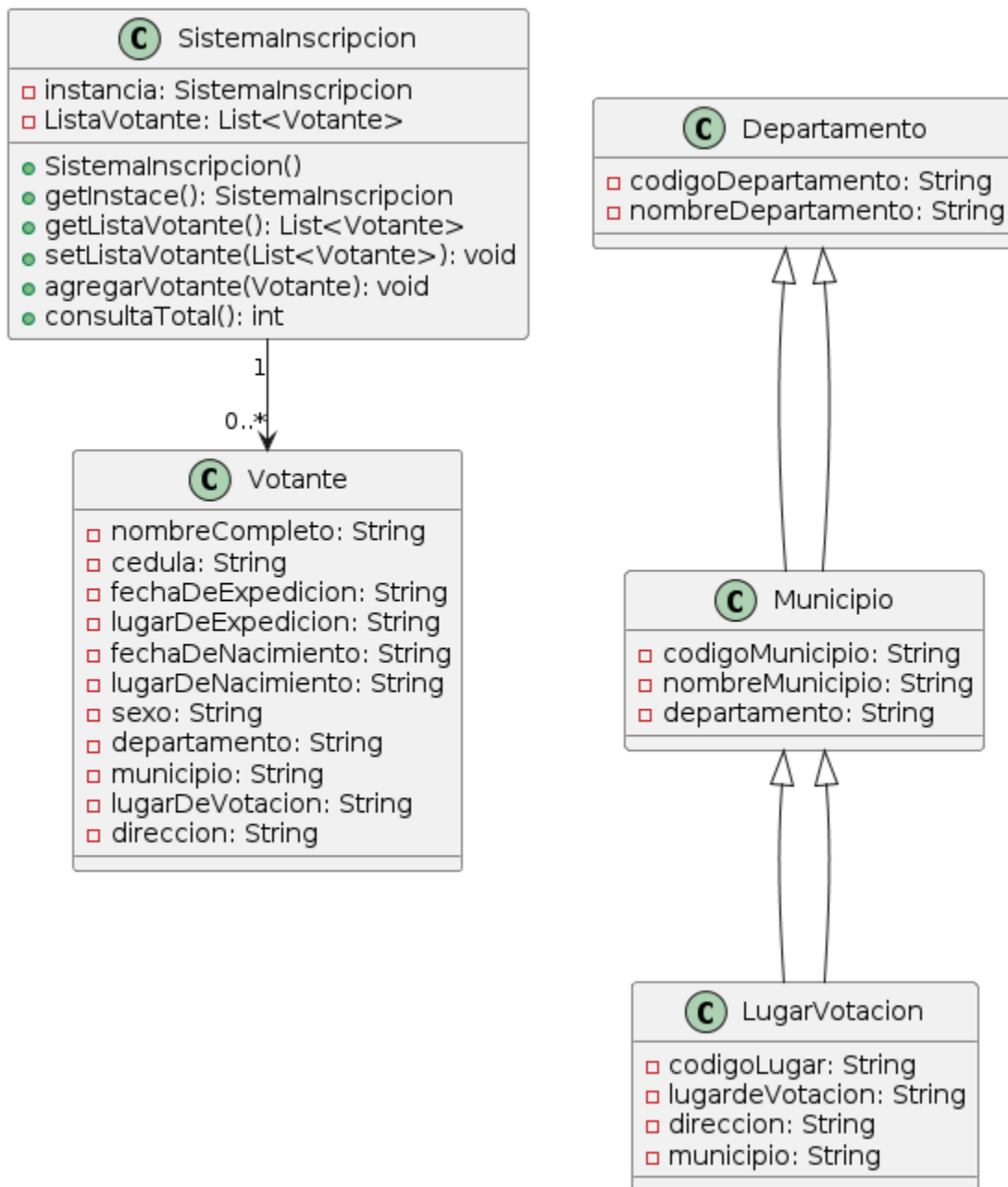


Diagrama de clases

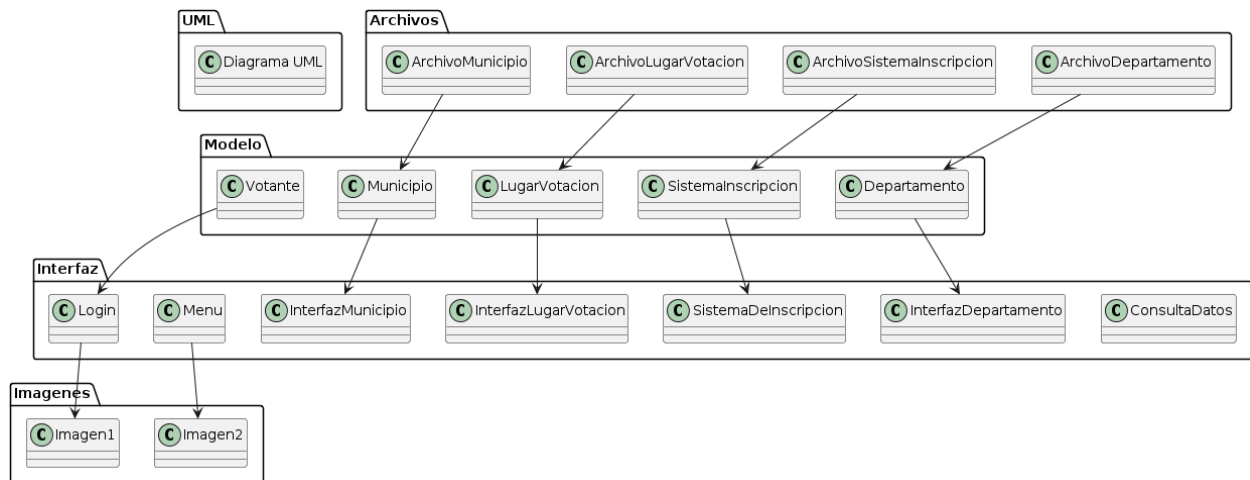


Fase (2): Definición de la arquitectura del proyecto.

Diagrama de clases




Diagrama de paquetes



Diseño preliminar de interfaces gráficas de usuario

X

INSCRIPCIÓN DE CEDULA




INICIAR SESIÓN

USUARIO:

CONTRASEÑA:

INGRESAR

X



Menú

Registrar Persona

Registrar Departamento

Registrar Municipio

Registrar Lugar de Votacion

Consultar

X



INGRESE LOS DATOS

Nombre completo

Cedula de ciudadanía

Fecha y lugar de expedición

Fecha y lugar de nacimiento

Sexo

...

Departamento

...

Municipio

...

Lugar de votación

...

Dirección

Registrar

Salir

X

REGISTRO DEPARTAMENTO

Codigo Departamento

Nombre Departamento

Codigo	Nombre

Registrar

Salir

REGISTRO MUNICIPIO

Codigo Municipio

Nombre Municipio

Nombre Departamento

...



Codigo

Nombre

Departamento

Registrar

Salir

REGISTRO LUGAR DE VOTACIÓN

Codigo Votación

Lugar Votación

Dirección

Nombre Municipio

...



Codigo Votación

Lugar

Dirección

Municipio

Registrar

Salir

X

CONSULTA DE REGISTROS

Nombre Completo	Cedula	Departamento	Municipio	Lugar de Votación

Salir

Fase (3): Desarrollo del producto final de software.

https://drive.google.com/file/d/1U6Z09uCyDAYxa0UDct6jlJGueWOfZyzD/view?usp=drive_link

Referencias de los programas

TotalVote. (2021, February 1). KNOWiNK. <https://knowink.com/products/totalvote/>

Election management. (2023, February 17). Civix. <https://gocivix.com/resources/guides/election-platform/>

PowerProfile - election systems & software. (2019, January 30). Election Systems & Software. <https://www.essvote.com/products/powerprofile/>