**Instituto Tecnológico de Costa Rica**

Sede San Carlos

**Escuela de Ingeniería en Computación**

Programación Orientada a Objetos

**Proyecto I**

Voto Tec

Randald Villegas B.

Fabián Zamora R.

Santa Clara, I semestre 2018

**Introducción**

La programación orientada a objetos (POO), quiere decir que se organizaran los archivos que utiliza el programa de tal manera que se representan como objetos, los cuales poseen atributos y métodos que son obtenidos al crearse una instancia de una clase.

Algunas de las ventajas que tenemos al programar de esta manera es que se parece al mundo real, crea modelos muy exactos y complejos, entre otras.

En el siguiente proyecto será elaborado un proyecto en java con el intérprete de Netbeans, en el cual se nos solicita crear una aplicación la cual permita crear elecciones costarricenses, la aplicación está basada en hacer todos los trámites para hacer las elecciones, algunas de las funciones son empadronar personas las cuales cumplan los requisitos que solicita otras de las funciones son crear partidos políticos y candidaturas para presidentes y diputados, además el sistema le permite modificar usuarios, eliminar información , votar y llevar un control, entre otros.

**Análisis del problema**

Se desarrolló una aplicación de escritorio en java, la cual les permite a diferentes usuarios registrados como administradores ingresar al registro civil y empadronar nuevas personas, establecer las mesas por cada distrito electoral, además la acreditación para los participantes fiscales de cada uno de los partidos políticos registrados.

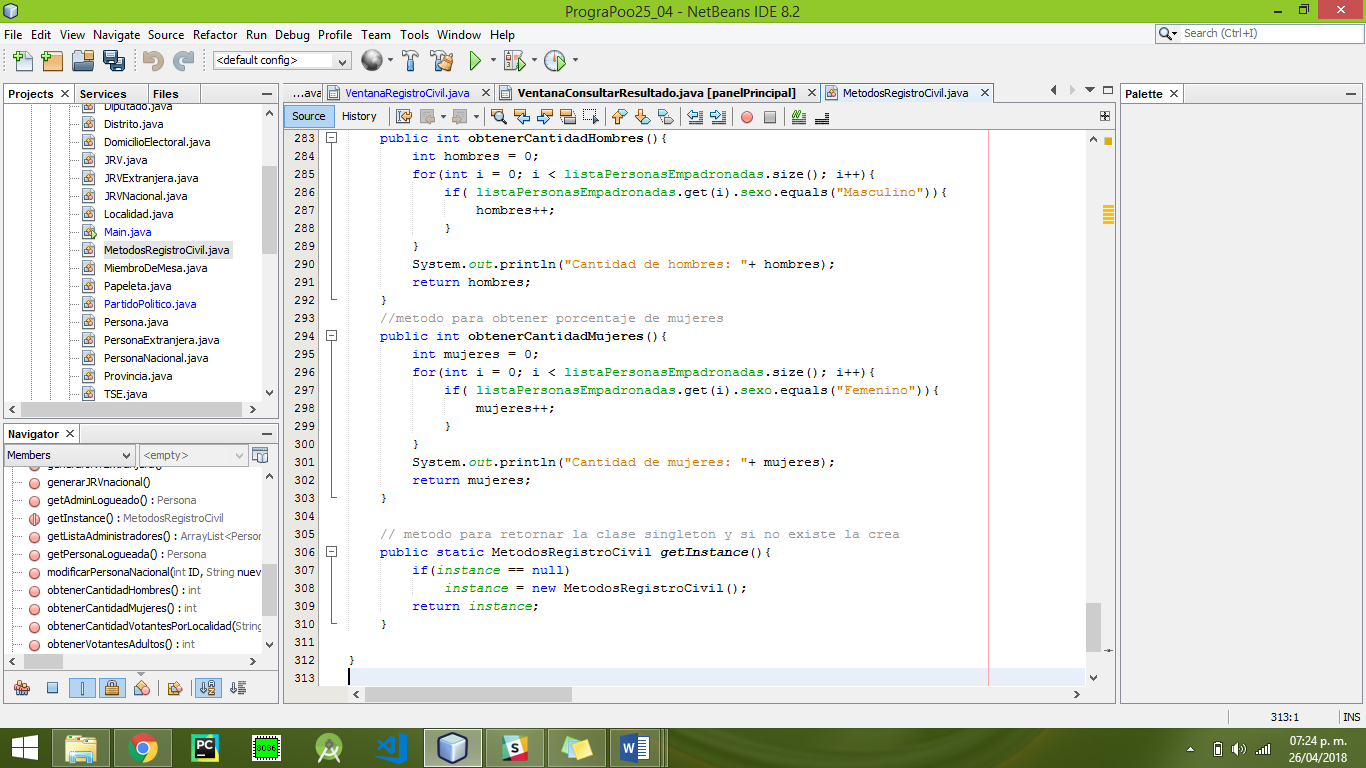
El registro civil guarda la información de cada persona, información como Nombre, Apellidos cedula, estado civil, fecha de nacimiento, sexo, lugar de nacimiento, nacionalidad, si esta fallecido o no, si está empadronado o no y un domicilio electoral.

Debido a que las personas de diferentes lugares tienen su derecho al voto, se debe generar un cálculo de cada una de las juntas receptoras de votos, las cuales contienen un código, cantidad de electores, un centro de votación, si es nacional o no. Además se debe crear otras clases para localidad, como para los distritos, cantones y provincias de nuestro país. Por otro lado si las juntas receptoras de votos no son nacionales se deben adquirir otros datos como una localidad, un código y un nombre.

Los partidos políticos tienen que ser inscritos en el registro civil, los cuales hay que guarda la siguiente información, nombre, código, escala, fecha que se fundó, además una bandera, un teléfono y su dirección.

Por otra parte para las candidaturas de presidente y diputados hay que cumplir con ciertas características, como por ejemplo un plan de gobierno, su curriculum, foto del candidato, entre otros.

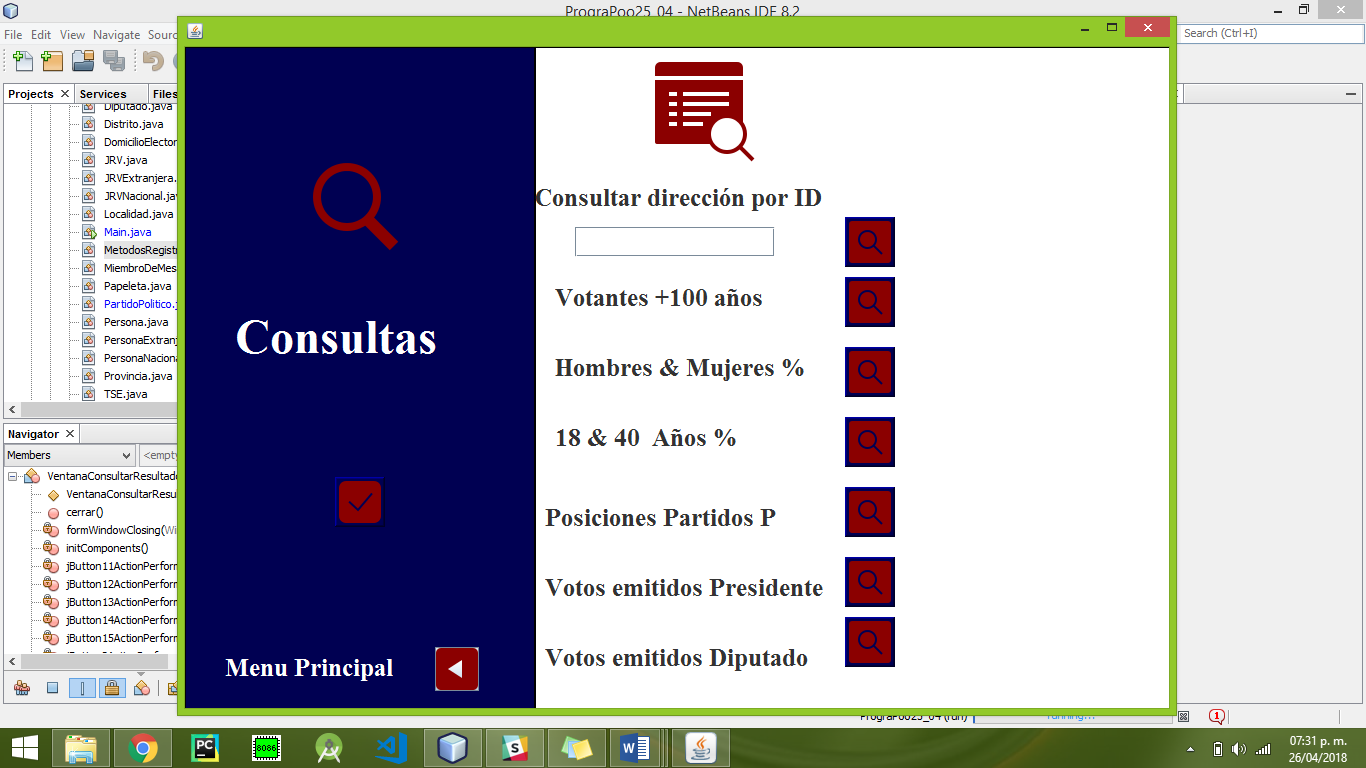
**Solución del problema**

Se analizaron todas las posibles soluciones para desarrollar la aplicación de la manera más eficiente, se creó un singleton para la lógica de registro civil, se creó un menú para la aplicación la cual se encargaba de mostrar diferentes ventanas a las diferentes funciones de la aplicación, en las cuales se puede ir a votar, en donde se muestra una ventana de verificación con su nombre y número de cedula para poder conformar que no pueda una persona votar ambas veces en unas mismas elecciones.

Entre las otras funciones que tiene la aplicación podemos mencionar la función de ir al registro civil, donde se solicita que sea una persona administradora quien pueda ingresar, al sistema y ahí poder realizar diferentes funciones como agregar personas nacionales, agregar extranjeros, eliminar personas, editar los datos de las personas, agregar distritos, provincias, cantones, localidades y consulados;

Además se puede agregar partidos políticos, generar Juntas Receptoras de votos, entre otros.

Entre otras de las funciones que cuenta la aplicación es agregar candidaturas para presidente, diputados y consultar los resultados de las votaciones, algunas de las consultas son consultar dirección por id, cantidad de personas que votaron mayores a 100 años, porcentaje entre hombre y mujeres, posiciones de los partidos políticos, votos emitidos para el presidente y para diputado



**Análisis de Resultados**

**Tarea Completado %**

|  |  |
| --- | --- |
| El menú está estructurado de manera adecuada | 100 |
| Utiliza componentes adecuados para la gestión (inserción, modificado y borrado) de los datos | 100 |
| Todas las pantallas están debidamente estandarizadas con respecto a la utilización de: tipo y tamaño de la letra, íconos, etc | 100 |
| El diseño de las pantallas es congruente con el diagrama de clases | 100 |
| Rubro de evaluación (Gestión de datos y consultas) | 100 |
| El sistema permite insertar en todas las clases | 100 |
| El sistema permite modificar los datos de los ciudadanos, los partidos políticos y las candidaturas. | 100 |
| El sistema permite borrar la información completa de los ciudadanos. | 100 |
| El sistema realiza una adecuada asociación de las clases según los diferentes tipos de relación que se definieron en el diagrama de clases | 100 |
| El sistema genera correctamente las consultas solicitadas | 100 |
| El sistema genera correctamente los reportes solicitados | 70 |
| El sistema permite el desarrollo de la aplicación adecuadamente según el enunciado del problema | 100 |
| El sistema cuenta con datos previamente insertados | 100 |
| Rubro de evaluación (Documentación) | 100 |
| El documento contiene todas las partes y está organizado de la manera solicitado | 100 |
| En la introducción se explica con claridad de qué trata el documento, especificando las partes que los componen y una pequeña descripción de cada una de ellas | 100 |
| El problema está perfectamente delimitado y descrito | 90 |
| Se analizan las causas y consecuencias que originan el problema descrito en el documento | 95 |
| Comunica de manera eficiente la solución adoptada a partir del análisis del problema realizado | 100 |
| Es claro y conciso a la hora expresar los resultados del proyecto | 100 |
| Las conclusiones guardan correspondencia con el problema y los resultados encontrados | 100 |

**Conclusiones**

La gran mayoría de los requerimientos pudieron ser implementados en la aplicación java durante el lapso de tiempo disponible para entregar el producto final de la aplicación, sin embargo, hicimos el mejor trabajo posible luchando con el tiempo.

Las clases y sus relaciones fueron las encargadas de ayudarnos a desarrollar el proyecto de la mejor manera posible y de la manera más fácil, tomando en cuenta que se relacionan entre si y de esta manera se puede hacer referencia a las clases y diferentes métodos utilizando estrategias como la del singleton.

Además, se pudo aprender acerca del entorno de java y la programación orientada a objetos. Gracias al trabajo y esfuerzo que nos dimos al investigar diferentes temas.

**Bibliografía**

Random (Java Platform SE 7 ). (n.d.). Retrieved April 24, 2018, from https://docs.oracle.com/javase/7/docs/api/java/util/Random.html

Zukowski, J. (2005). *The definitive guide to Java Swing*. Apress. Retrieved from https://books.google.es/books?hl=en&lr=&id=YPjZNlEgAMcC&oi=fnd&pg=PR19&dq=TransferHandler+java&ots=GQ5snWfYa8&sig=\_Aanzr6YKKA0sIGuD78OyFHvNgA#v=onepage&q=TransferHandler java&f=false