**VotingPRO – La solución de software que automatiza procesos electorales.**

Lopez Alejandro 1; Mena Ian 2; Pino Miguel3

1 Universidad Tecnológica Israel-Departamento de Ciencias de la Ingeniería –Carrera de Sistemas, Quito, Ecuador

**Resumen.** El proyecto presentando a continuación fue desarrollado con unos de los lenguajes de programación más usados, hablamos de Csharp. Csharp es un lenguaje que adopta varias características de otros lenguajes cómo C++ o Java. Su uso se ha convertido en una de las principales tendencias por parte de los desarrolladores ya que posee uno de los compiladores más eficientes y óptimos del mundo de la programación, además de trabajar con uno de los Frameworks más usados hablando de .NET Framework.

El programa posee una conexión de base de datos misma que trabajamos con el motor de MySQL, el sistema está hecho para funcionar en la versión de escritorio y su desarrollo fue concebido en el IDE de programación Visual Studio por medio de la aplicación del Framework .NET con Windows Forms. Además, es importante recalcar que este proyecto fue realizado por medio de la arquitectura MVC, MODELO-VISTA-CONTROLADOR para explotar sus características como poder separar código, reutilizar el mismo y que la aplicación a futuro sea escalable y fácil de mantener.

**Palabras clave:** MySQL, Csharp, .NET Framework Windows Forms, aplicación de escritorio, MVC, Visual Studio,

**Application for registration of candidates, votes and statistical diagrams.**

**Abstract.** The project presented below, was developed with one of the most widely used programming languages, we are talking about Csharp. Csharp is a language that adopts several features from other languages such as C++ or Java. Its use has become one of the main tendencies on the part of the developers since it has one of the most efficient and optimal compilers of the programming world, besides working with one of the most used Frameworks speaking of .NET Framework.

The program has a database connection that we work with the MySQL engine, the system was made to work in the desktop version and its development was conceived in the Visual Studio programming IDE through the application of the .NET Framework with Windows Forms. In addition, it is important to emphasize that this project was made through the MVC architecture, MODEL-VIEW-CONTROLLER to exploit its features such as code separation, code reuse and that the application is scalable and easy to maintain in the future.

**Keywords:** MySQL, Csharp, NET Framework Windows Forms, desktop application, MVC, Visual Studio.

1. **[[1]](#footnote-1)INTRODUCCIÓN**

Para entender el funcionamiento de nuestro programa, es necesario establecer algunos conceptos inmersos en el desarrollo de nuestra aplicación de escritorio. Primero hablaremos de la arquitectura trabaja que es MVC: MODELOS-VISTAS-CONTROLADORES.

* 1. **Patrón MVC:**

La arquitectura MVC propone, independientemente de las tecnologías o entornos en los que se base el sistema a desarrollar, la separación de los componentes de una aplicación en tres grupos (o capas) principales: el modelo, la vista, y el controlador, y describe cómo se relacionarán entre ellos para mantener una estructura organizada, limpia y con un acoplamiento mínimo entre las distintas capas. (Aguilar, 2019)

**El modelo:**

En la capa Modelo encontraremos siempre una representación de los datos del dominio, es decir, aquellas entidades que nos servirán para almacenar información del sistema que estamos desarrollando. Por ejemplo, si estamos desarrollando una aplicación de facturación, en el modelo existirán las clases Factura, Cliente o Proveedor, entre otras. Asimismo, encontraremos la lógica de negocio de la aplicación, es decir, la implementación de las reglas, acciones y restricciones que nos permiten gestionar las entidades del dominio. Será por tanto el responsable de que el sistema se encuentre siempre en un estado consistente e íntegro. (Alicante, 2019)

**La vista:**

Las vistas en el patrón de diseño MVC, son las encargadas de establecer cuál será la interacción que tendrá el usuario con nuestro aplicativo, todo esto se compone en nuestro caso, de los formularios que hemos creado para cada ventana donde se ingresarán distintos tipos de datos por parte del usuario final y permitirá el manejo por medio de listas, botones, campos de texto, menús, etc.

**El controlador:**

El controlador es quizás la parte más importante de nuestro sistema, ya que este posee la lógica de nuestro programa. Los controladores son quienes procesan todas las acciones entre el usuario y nuestro proyecto.

Serán capaces de capturar las acciones de éste sobre la Vista, como puede ser la pulsación de un botón o la selección de una opción de menú, interpretarlas y actuar en función de ellas. Realizarán también tareas de transformación de datos para hacer que los componentes de la Vista y el Modelo se entiendan. Así, traducirán la información enviada desde la interfaz. Podríamos considerar el Controlador como un coordinador general del sistema, que regula la navegación y el flujo de información con el usuario, ejerciendo también como intermediario entre la capa de Vista y el Modelo. (Aguilar, 2019)

* 1. **Csharp**

Csharp es uno de los lenguajes más usados en el mundo, está dentro de los llamados ‘’lenguajes de alto nivel’’ y pertenece al antes mencionado Framework de .NET, mismo que posee otros lenguajes cómo Visual Basic, C++ y C, con este tipo de lenguaje para computadoras, podemos ser capaces de crear aplicaciones tanto para web como para escritorio.

* 1. **.NET Framework**

Para entender lo que es C# es imprescindible decir antes lo que es Microsoft .NET Framework o abreviadamente .NET. Se trata de un entorno de desarrollo multilenguaje diseñado por Microsoft para simplificar la construcción, distribución y ejecución de aplicaciones para Internet. Tiene fundamentalmente tres componentes: una máquina virtual (CLR: Common Language Runtime) que procesa código escrito en un lenguaje intermedio (MSIL: Microsoft Intermediate Language), una biblioteca de clases (biblioteca .NET) y ASP.NET que proporciona los servicios necesarios para crear aplicaciones Web. (Sierra, 2015)

Para la vida de un desarrollador, se pueden utilizar varias herramientas que están a nuestra completa disponibilidad, pero uno de los mejores frameworks que actualmente existen para la programación con C# en el framework .NET. Existen varias ventajas de trabajar con este tipo de framework, por ejemplo, nos permite visualizar una precarga de nuestra aplicación, utiliza el Routing en el ASP.net para Web y Forms, tiene una herramienta para el control/ reducción de ViewState y la utilización del modelo MVC. (Deitel, 2007)

.NET Framework es un componente integrado a Windows que soporta la ejecución y desarrollo de una nueva generación de aplicaciones y del XML web services. Este framework nos provee de un entorno a la programación orientada a objetos de tal forma que el código almacenado y ejecutado localmente, pueda ser ejecutado de igual forma en internet y remotamente. De igual forma, .NET permite una ejecución de código que elimina los problemas de rendimiento generado por lenguajes de programación de script o entornos interpretados. Este Framework tiene dos componentes principales, el CLR mejor conocido como ‘’Common Language Runtine y el .NET Framework class library, que incluye ADO.NET, ASP. NET Y Windows Forms. (Arias, 2014)

* 1. **MySQL**

MySQL es un sistema de administración de bases de datos (Database Management System, DBMS) para bases de datos relacionales. Así, MySQL no es más que una aplicación que permite gestionar archivos llamados de bases de datos. También es muy destacable, la condición de open source de MySQL, que hace que su utilización sea gratuita e incluso se pueda modificar con total libertad, pudiendo descargar su código fuente. Esto ha favorecido muy positivamente en su desarrollo y continuas actualizaciones, para hacer de MySQL una de las herramientas más utilizadas por los programadores orientados a Internet. (kinsta, 2019)

MySQL es una de las herramientas conocidas como SGBD, sus siglas expresan que MySQL es un Sistema Gestor de Base de Datos, mismo que permite la administración de nuestra base y posee una infinidad de recursos para el tratamiento de los datos que lo convierte en uno de los motores más usado actualmente. Las bases de datos son repositorios de información, pero que se encuentran relacionados entre sí y representan gran valor para los dueños de dicha información.

* 1. **Microsoft Visual Studio**

Visual Studio es un entorno de desarrollo integral que nos permite trabajar con varios lenguajes de programación en un solo lugar, permite crear aplicaciones en tiempos muy cortos debido a su sencillez y manejo. Este IDE de programación nos permite desarrollar de una manera más inteligente debido a las sugerencias de código que brinda y otros beneficios que nos pueden ayudar a desarrollar en C# como los siguientes:

Visual Studio nos permite visualizar cambios importantes que ha sufrido nuestro código, de esta manera tenemos una mejor auditoría, de igual forma, la ejecución de código en Visual Studio permite inspeccionar los errores que hayamos tenido y corregirlos conforme se producen. La herramienta nos permite integrar varias cargar de trabajo conforme a los que necesitemos para nuestro desarrollo, y hay que considerar que actualmente la popularidad de Visual Studio es grande, esto es importante a la hora de encontrar información para desarrollar cualquier proyecto. (Sierra, Ceballos Sierra, F. J. (2015). Enciclopedia de Microsoft Visual C#: interfaces gráficas y aplicaciones para Internet con Windows Forms y ASP.NET (4a. ed.), 2015)

* 1. **Aplicación de escritorio**

Para el proyecto que se presentará a continuación, se utilizó la versión para desarrollo de escritorio de Windows Forms, este tipo de aplicaciones solamente se ejecutan en la computadora en dónde fue desarrollado el sistema o bien se puede hacer un ejecutable del programa para instalarlo en otra máquina. Funcionan con bases de datos que de igual forma son creadas en el mismo ordenador donde se desarrolla el aplicativo.

En primer lugar, se procede a crear un proyecto en Visual Studio, creamos un nuevo proyecto dando clic en crear.

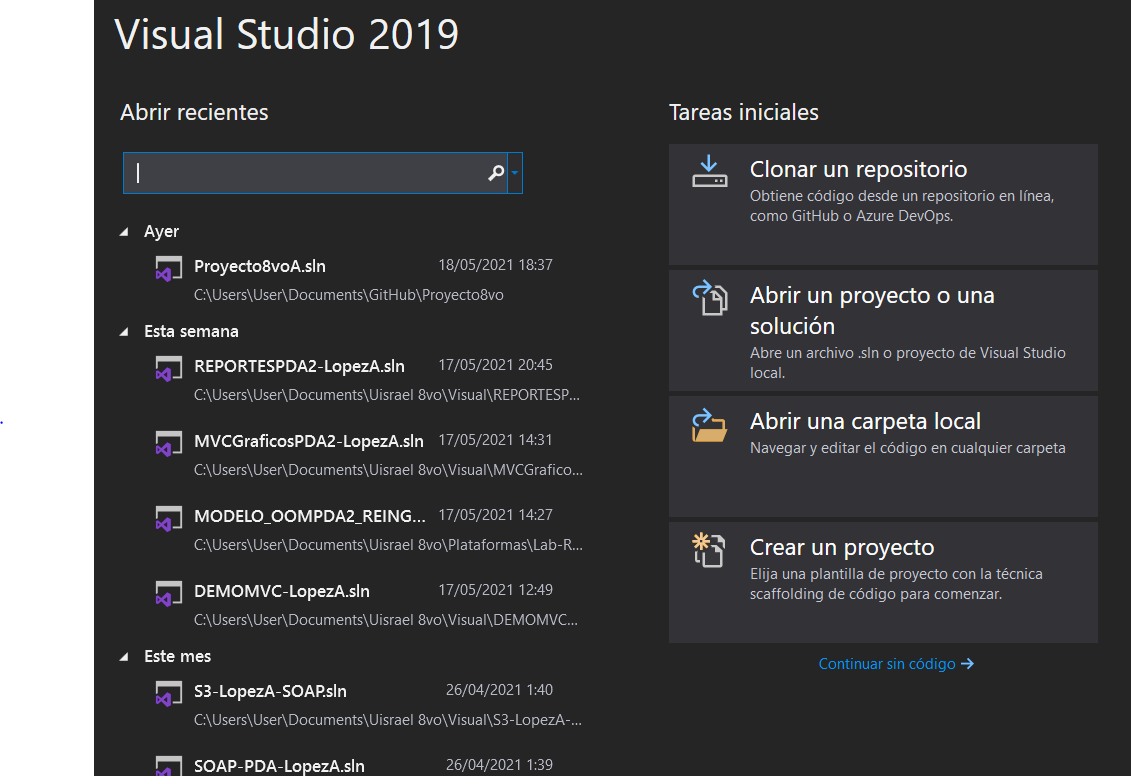


Figure 1: Ventana crear proyecto Visual Studio

*Figure SEQ Figure \\* ARABIC 1: Venta de Visual Studio crear un proyecto.*

Posterior a esto, elegimos la plantilla de nuestro proyecto misma que será Aplicación de Windows Forms y damos clic en siguiente,

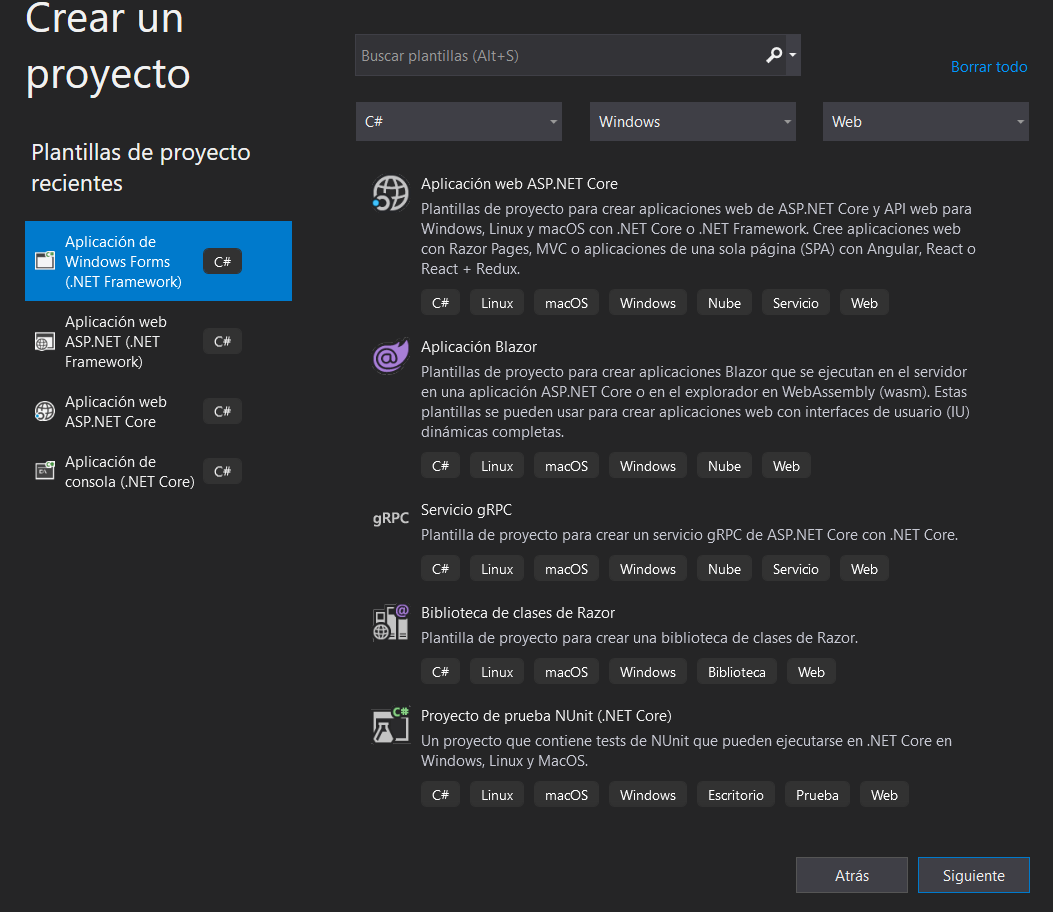


Figure 2: Ventana Aplicación de Windows Forms

*Figure : Aplicación de Windows Forms*

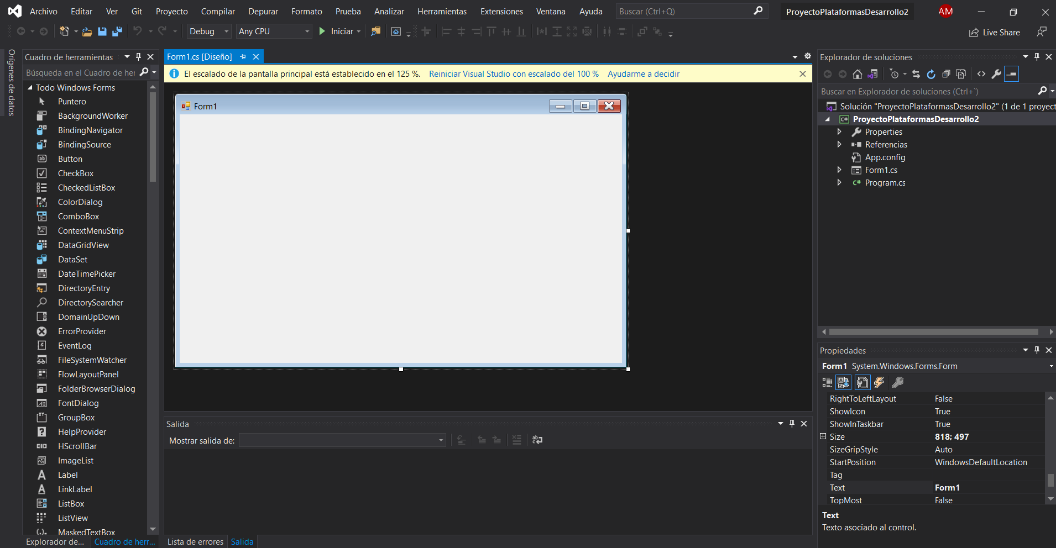
Finalmente escogemos un nombre para nuestro proyecto y damos clic en crear. Así tendremos nuestro proyecto de Windows Forms.

Figure 3: Ventana principal de programación Visual Studio

*Figure : Ventana Visual Studio Windows Forms*

1. **METODOLOGÍA**

El desarrollo de este programa fue realizado en base a un estudio experimental. El fin de la experimentación es identificar las causas por las que se producen determinados resultados. La experimentación a través de la ingeniería de Software nos permite a los desarrolladores entender y comprender todas las variables que componen nuestro sistema, sustentándose en la teoría y en evidencia científica para el desarrollo de software, de esta forma se lograr comprender cómo interactúan los sistemas y construir en base a criterios propios, investigaciones experimentales que avalan nuestro producto final. (Kasiak Tamara, 2012)

Un experimento modela en el laboratorio (es decir, en condiciones controladas) las principales características de una realidad (en nuestro caso el desarrollo de software) lo que permite estudiarla y comprenderla mejor. La fortaleza de la experimentación en laboratorio es que permite variar iterativamente aspectos de la realidad para estudiar el impacto que tienen tales manipulaciones. (Madrid, 2018)

* 1. **METODOLOGÍA XP**

Para nuestro proyecto hemos aplicado la metodología de programación ágil XP, este marco de trabajo es de gran ayuda cuando queremos implementar sistemas medianos en donde los requisitos no varíen mucho durante el desarrollo del programa y cada cierto tiempo, se va obteniendo versiones del sistema para comprobar si las historias de usuario se están cumpliendo o no.

La programación extrema (XP por las siglas en inglés) es un proceso ágil de desarrollo de software, enfocada a las buenas prácticas de codificación, una clara comunicación y al trabajo en equipo. Está concebida para proyectos medianos y pequeños donde los requisitos son cambiantes. Por lo tanto, tiene una serie de reglas y recomendaciones que se pueden dividir en planeación y gestión, diseño, codificación, y pruebas para producir un software. (Jovani Alberto Jiménez Builes, 2019)

**HISTORIAS DE USUARIO**

|  |  |
| --- | --- |
| HISTORIA DE USUARIO | |
| **Número:** 001 | **Usuario:** Institución educativa donde se realizarán las votaciones. |
| **Nombre Historia:** Añadir Candidato | |
| **Prioridad en Negocio:** Grado de cumplimiento alto | **Riesgo en el desarrollo:** Complejidad media para el equipo de desarrollo. |
| **Puntos estimados:** Dos semanas | **Iteración Asignada:** 2 |
| **Programador responsable:** Alejandro López, Ian Mena, Miguel Pino | |
| **Descripción:** El sistema debe permitir ingresar los datos básicos de un candidato a las elecciones. | |

*Tabla 1: Historia de Usuario 1 - Añadir Candidato.*

|  |  |
| --- | --- |
| HISTORIA DE USUARIO | |
| **Número:** 002 | **Usuario:** Institución educativa donde se realizarán las votaciones. |
| **Nombre Historia:** Añadir Partido Político | |
| **Prioridad en Negocio:** Grado de cumplimiento alto | **Riesgo en el desarrollo:** Complejidad media para el equipo de desarrollo. |
| **Puntos estimados:** Dos semanas | **Iteración Asignada:** 2 |
| **Programador responsable:** Alejandro López, Ian Mena, Miguel Pino | |
| **Descripción:** El sistema debe permitir el ingreso de los datos del partido político del candidato y adjuntar una fotografía del mismo | |

*Tabla 2: Historia de Usuario 2: Añadir Partido Político.*

|  |  |
| --- | --- |
| HISTORIA DE USUARIO | |
| **Número:** 003 | **Usuario:** Institución educativa donde se realizarán las votaciones. |
| **Nombre Historia:** Añadir Votantes | |
| **Prioridad en Negocio:** Grado de cumplimiento alto | **Riesgo en el desarrollo:** Complejidad media para el equipo de desarrollo. |
| **Puntos estimados:** Dos semanas | **Iteración Asignada:** 3 |
| **Programador responsable:** Alejandro López, Ian Mena, Miguel Pino | |
| **Descripción:** El sistema debe permitir al usuario ingresar los votantes correspondientes a las elecciones con sus respectivos datos personales. | |

*Tabla 3: Historia de Usuario 3: Añadir Votantes.*

|  |  |
| --- | --- |
| HISTORIA DE USUARIO | |
| **Número:** 004 | **Usuario:** Administrador del sistema. |
| **Nombre Historia:** Conexión a una base de datos. | |
| **Prioridad en Negocio:** Grado de cumplimiento alto | **Riesgo en el desarrollo:** Complejidad media para el equipo de desarrollo. |
| **Puntos estimados:** Dos semanas | **Iteración Asignada:** 4 |
| **Programador responsable:** Alejandro López, Ian Mena, Miguel Pino | |
| **Descripción:** Todos los datos ingresados en las historias anteriores, cómo añadir candidatos, partidos políticos y votantes, deben almacenarse en una base de datos SQL. | |

*Tabla 4: Historia de Usuario 4: Conexión a base de datos.*

|  |  |
| --- | --- |
| HISTORIA DE USUARIO | |
| **Número:** 005 | **Usuario:** Institución educativa donde se realizarán las votaciones |
| **Nombre Historia:** Registro de votaciones | |
| **Prioridad en Negocio:** Grado de cumplimiento alto | **Riesgo en el desarrollo:** Complejidad media para el equipo de desarrollo. |
| **Puntos estimados:** Dos semanas | **Iteración Asignada:** 5 |
| **Programador responsable:** Alejandro López, Ian Mena, Miguel Pino | |
| **Descripción:** El usuario podrá realizar su votación por medio del sistema, mismas que se registrarán en la base de datos. | |

*Tabla 5: Historia de Usuario 5: Registro de votaciones*

|  |  |
| --- | --- |
| HISTORIA DE USUARIO | |
| **Número:** 006 | **Usuario:** Administrador del sistemas |
| **Nombre Historia:** Validación de voto. | |
| **Prioridad en Negocio:** Grado de cumplimiento alto | **Riesgo en el desarrollo:** Complejidad media para el equipo de desarrollo. |
| **Puntos estimados:** Dos semanas | **Iteración Asignada:** 6 |
| **Programador responsable:** Alejandro López, Ian Mena, Miguel Pino | |
| **Descripción:** El sistema debe validar que el usuario pueda ingresar un solo voto para la elección. La validación se realizará por medio de la cédula del votante. | |

*Tabla 6: Historia de Usuario 6: Validación de voto.*

|  |  |
| --- | --- |
| HISTORIA DE USUARIO | |
| **Número:** 007 | **Usuario:** Institución educativa donde se realizarán las votaciones |
| **Nombre Historia:** Diagramas estadísticos | |
| **Prioridad en Negocio:** Grado de cumplimiento alto | **Riesgo en el desarrollo:** Complejidad media para el equipo de desarrollo. |
| **Puntos estimados:** Dos semanas | **Iteración Asignada:** 7 |
| **Programador responsable:** Alejandro López, Ian Mena, Miguel Pino | |
| **Descripción:** El sistema debe contar con diagramas estadísticos para mostrar los resultados de las votaciones, mismos que serán modificados automáticamente con cada voto. | |

*Tabla 7: Historia de Usuario 7: Diagramas estadísticos.*

* 1. **MODELOS**

**MODELO ENTIDAD-RELACIÓN**

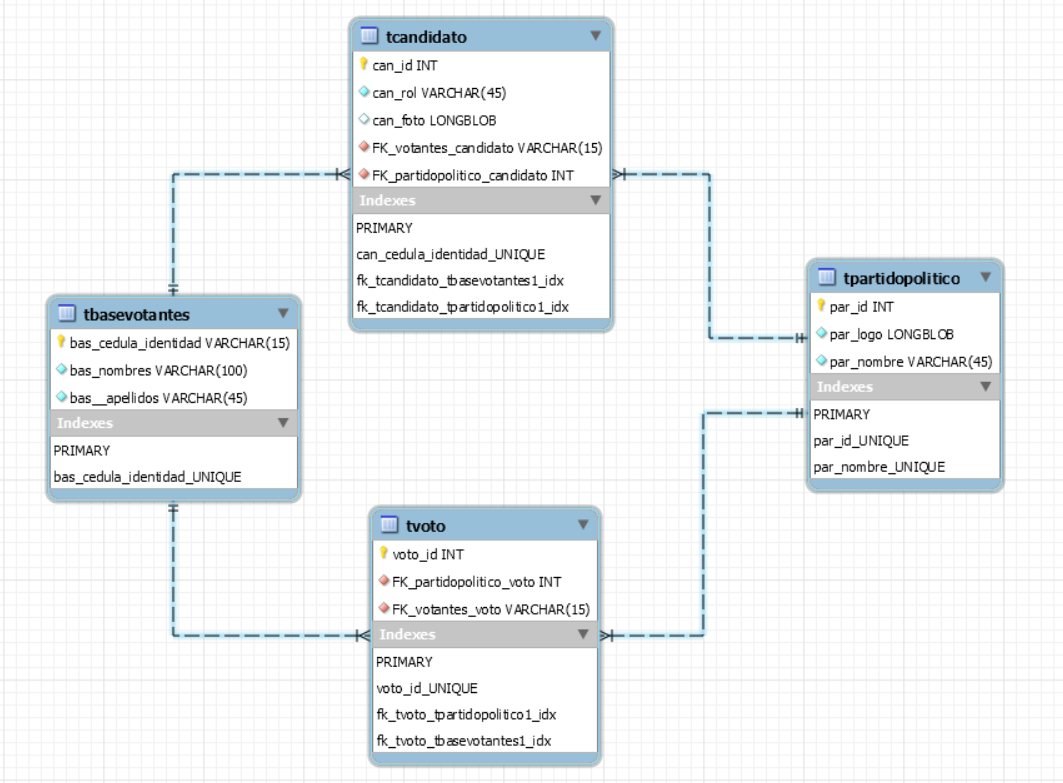
El modelo entidad-relación MER es un modelo de datos que permite representar cualquier abstracción, percepción y conocimiento en un sistema de información formado por un conjunto de objetos denominados entidades y relaciones, incorporando una representación visual conocida como diagrama entidad-relación. (Merchán Manzano, 2017)

Tabla 1: Modelo Entidad-Relación

*Figure : MODELO ENTIDAD RELACIÓN*

**MODELO DE CLASES ORIENTADO A OBJETOS**

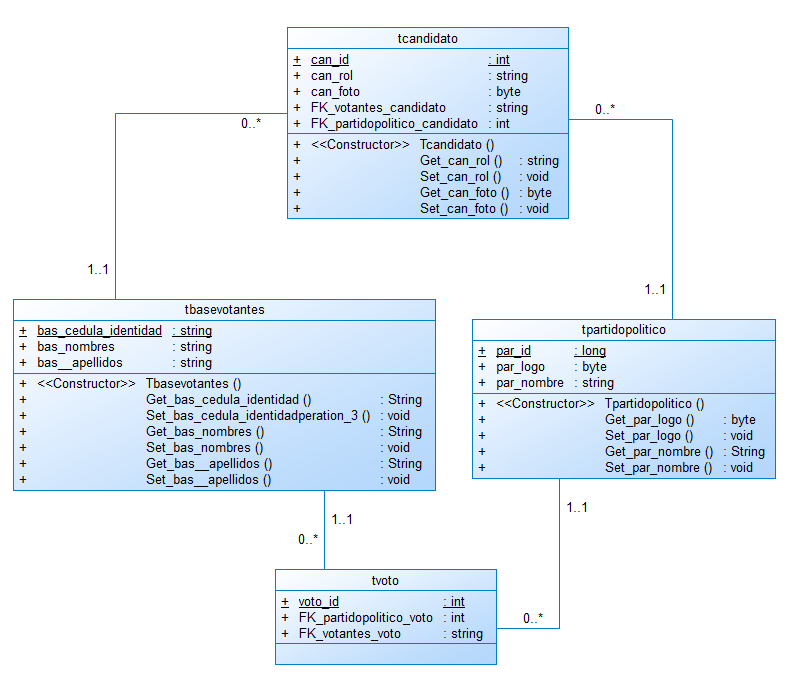
Un Diagrama de Clase es una representación gráfica de una clase, en el que se especifica el nombre de la clase, sus atributos y métodos; básicamente es uno de los elementos de un Diagrama de Clases en el que se muestran además las relaciones entre las clases. Un Diagrama de Clase, facilita en gran manera las acciones al momento de implementar, para a través de la observación reconocer la forma de acceso cada uno de los elementos de la clase, es el diagrama más sencillo que deberá realizarse al incursionar en la Programación Orientada a Objetos. (Norte, s.f.)

Tabla 2: Modelo de clases

*Figure : DIAGRAMA DE CLASES*

1. **RESULTADOS Y DISCUSIÓN**

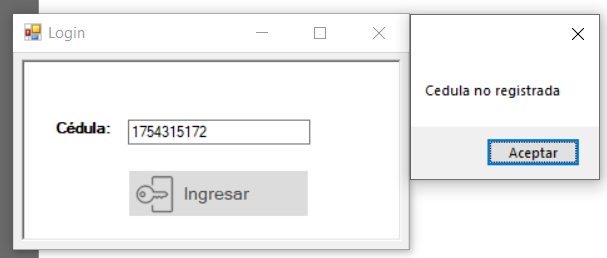
El primer paso a realizar es verificar que nuestro programa tenga dos tipos de acceso, el primero se lo realiza por medio de una cédula Master que nos servirá para poder ingresar a los votantes, candidatos, partidos políticos y visualizar el resultado de las votaciones.

Figure 4: Validación cédula Master

Como podemos ver en la imagen anterior, ingresamos una cédula cualquiera que no corresponde al usuario Master y tampoco está registrada, por lo tanto, no permite ingresar al sistema a realizar una votación y tampoco a realizar acciones de administración.

Para controlar este ingreso, se debe crear un usuario master. Lo que hacemos es ingresar con un número de cédula Master que nuestro caso es 999999999, una vez hecho esto, podremos ingresar al sistema para ingresar un nuevo usuario administrador.

Para este paso, es necesario crear el Master haciendo los siguientes campos:

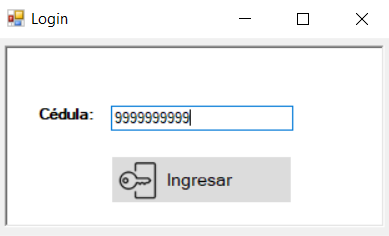
Ejecutamos el programa y colocamos la cédula Master.

Figure 5: Cédula ingreso Master

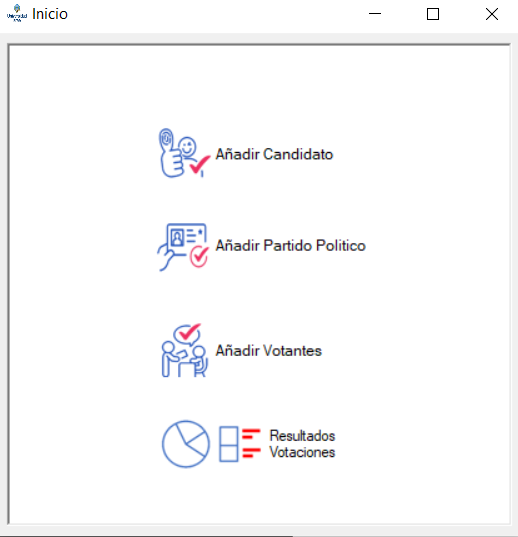
Una vez Ingresada esta cédula, se despliega la ventana principal del sistema para administradores:

Figure 6: Ventana principal para administrador del sistema.

Continuando, se da clic en ‘’Agregar votantes’’ para poder ingresar un nuevo usuario master con los siguientes datos:

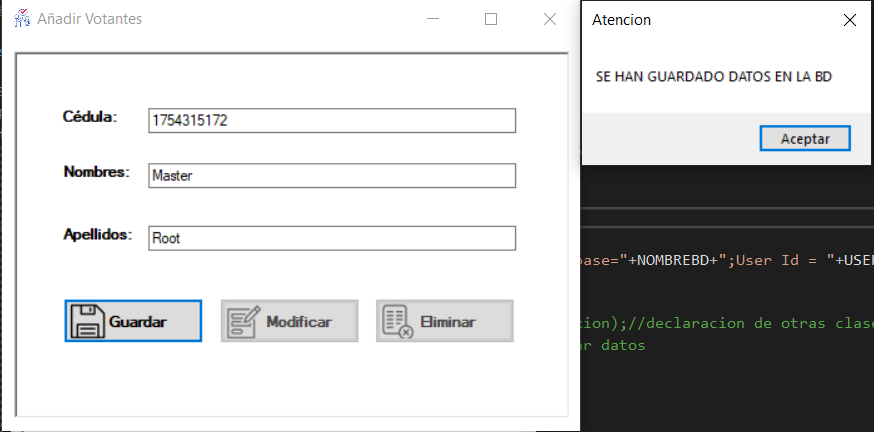
**Nombre: Master, apellido: Root, cédula: ‘’cédula usuario Master’’** Se nos abrirá la siguiente ventana para poder ingresar el usuario, aquí tenemos un campo de validación importante, ya que se implementó un algoritmo de validación de cédula ecuatoriana, si la cédula cumple con esas condiciones, se crea el registro y se guarda en la base de datos.

Figure 7: Ventana agregar usuario administrador.

Con este paso, estará creado un nuevo usuario Master y podrá ingresar al sistema con ese nivel.

Ingresamos nuevamente la cédula creada en Login:

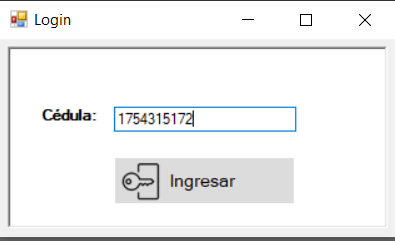


Figure 8: Usuario administrador creado

Y como veremos a continuación, podemos ingresar al sistema sin problema.

Las pruebas realizadas se realizaron con datos de listas, candidatos y votantes ficticios.

Empezamos agregando votantes para las elecciones. Mismo que pueden ser solo votantes o también candidatos de partidos políticos, por esta razón se agregaron primero los votantes, ya que es una de las claves foráneas de la tabla Candidatos. Para poder agregarlos damos clic en la opción ‘’Añadir Votantes’’

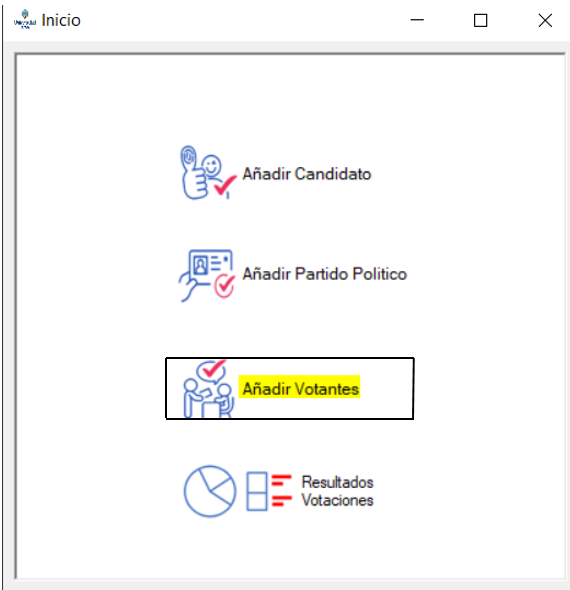


Figure 9: Añadir votantes

Agregamos los datos:

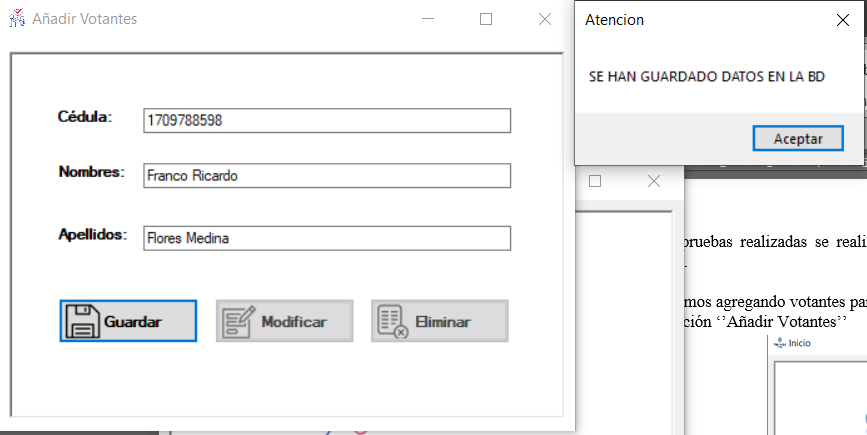


Figure 10: Datos de un votante.

A continuación, agregamos un partido político. En la ventana principal de administración damos clic en ‘’Agregar Partido Político’’ Primero se deben crear los partidos políticos y luego sus candidatos ya que la tabla Candidatos tiene la clave foránea de la tabla Partidos Políticos en la base de datos.

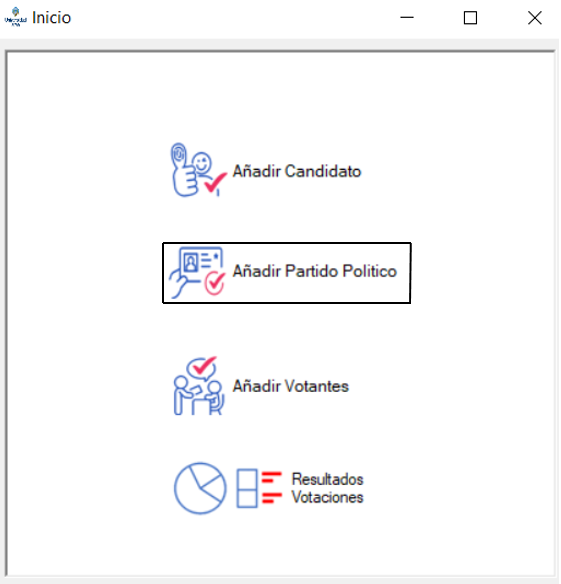


Figure 11: Añadir Partido Político

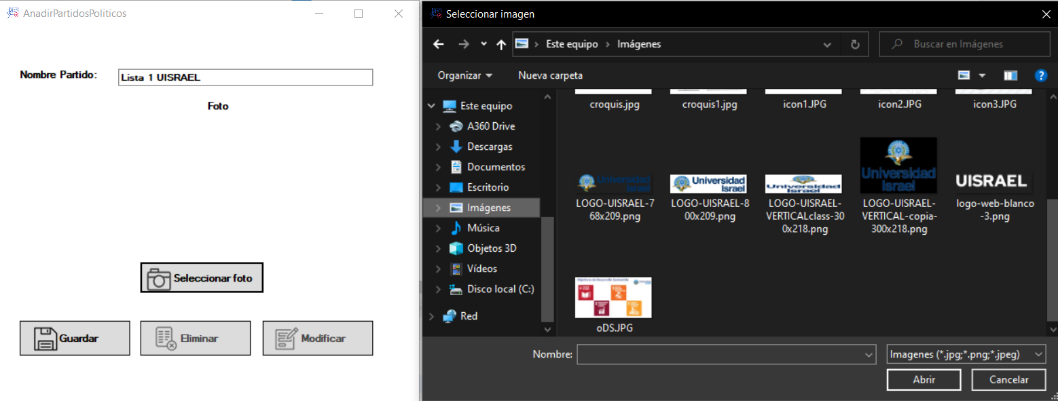
Ingresamos los datos del partido político: Colocamos el nombre del partido y tenemos la posibilidad de cargar una imagen del partido:

Figure 12: Cargar partido político

Abrimos la imagen que deseemos y damos clic en guardar. Quedando nuestro registro de la siguiente forma:

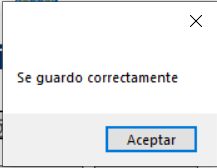


Figure 13: Partido Político Creado

Posterior a esto, procedemos a crear los candidatos de este partido político.

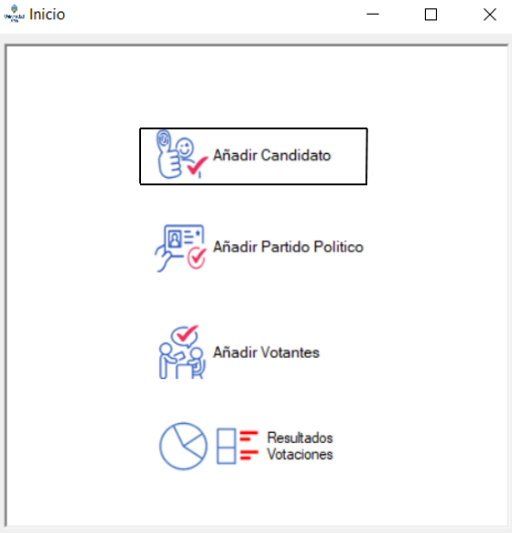
Creamos el candidato, clic en ‘’Añadir Candidato’’:

Figure 14: Añadir Candidato

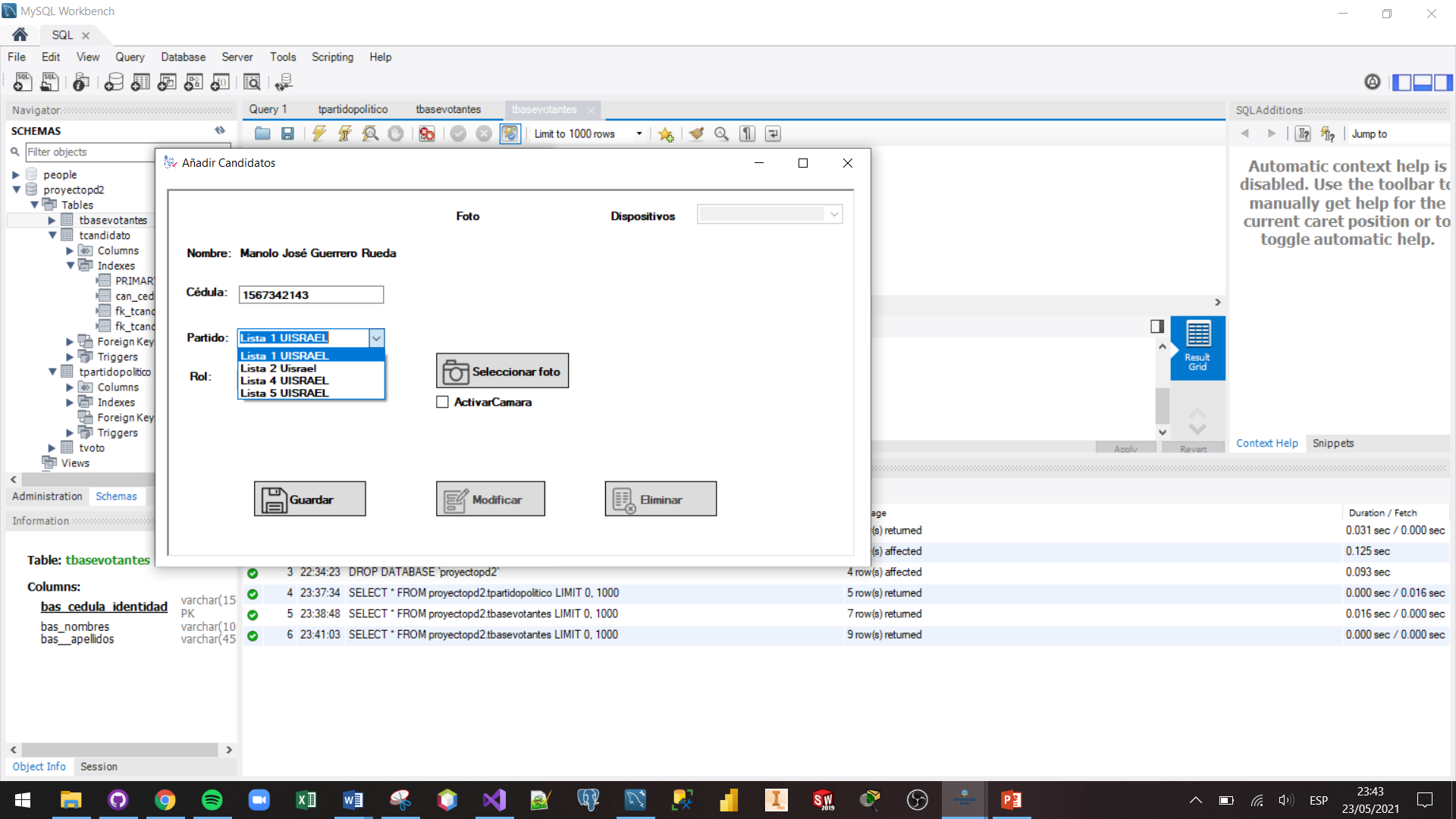
Ingresamos una cédula de la base de votantes y automáticamente nos obtiene los nombre como podemos en la Figura 15. En la opción Partido, elegimos la correspondiente al candidato:

Figure 15: Añadir candidato - Listas

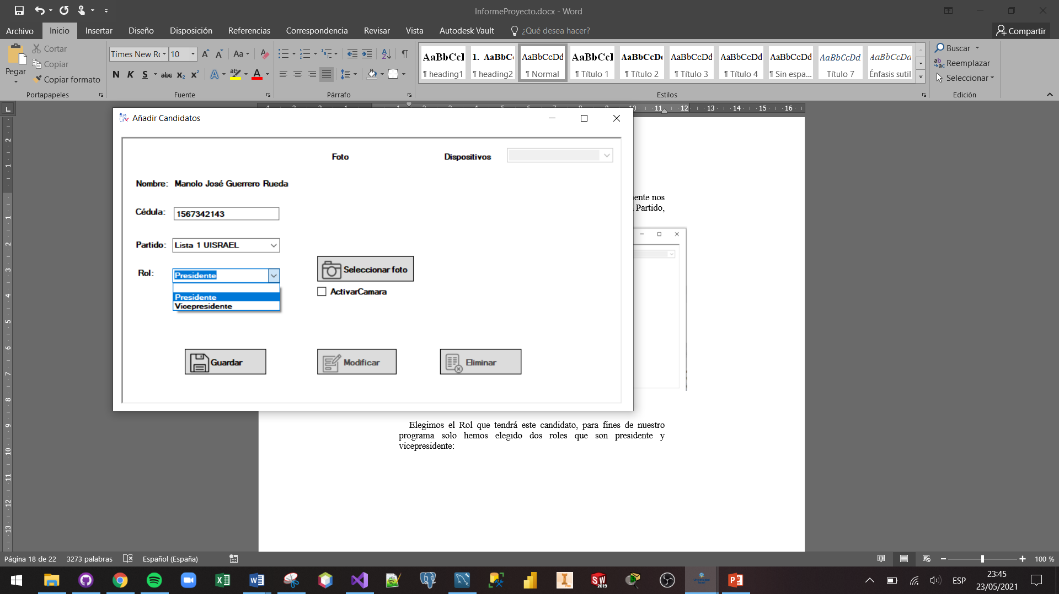
Elegimos el Rol que tendrá este candidato, para fines de nuestro programa solo hemos elegido dos roles que son presidente y vicepresidente:

Figure 16: Roles candidato.

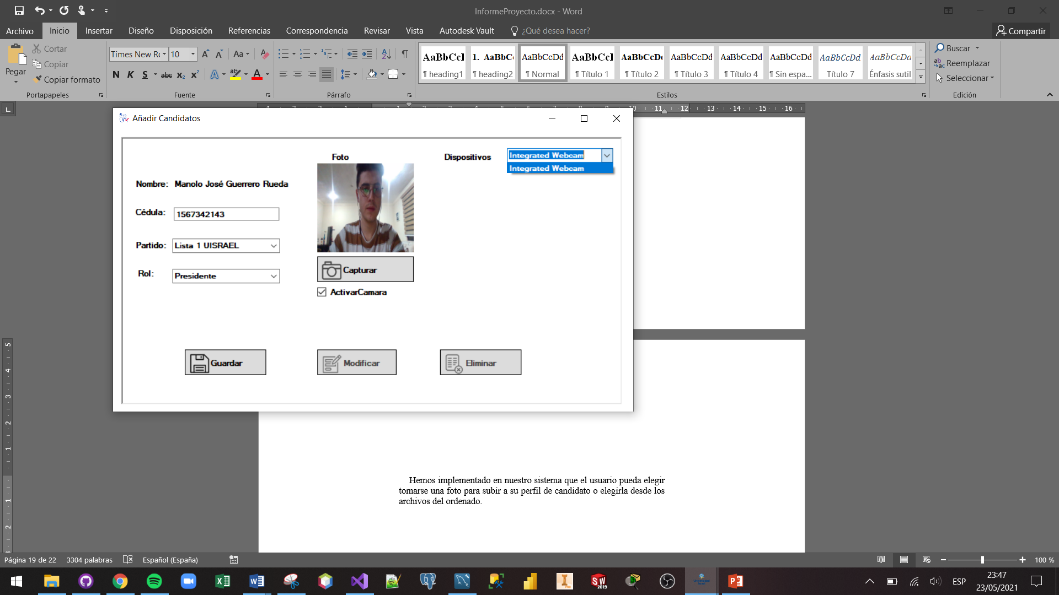
Hemos implementado en nuestro sistema que el usuario pueda elegir tomarse una foto para subir a su perfil de candidato o elegirla desde los archivos del ordenador. Si deseamos tomar una fotografía, activamos la opción ‘’Activar Cámara’’ y en dispositivos, escogemos nuestra cámara de la PC.

Figure 17: Tomar captura del candidato

Damos clic en capturar y si es de nuestro agrado, guardamos la imagen.

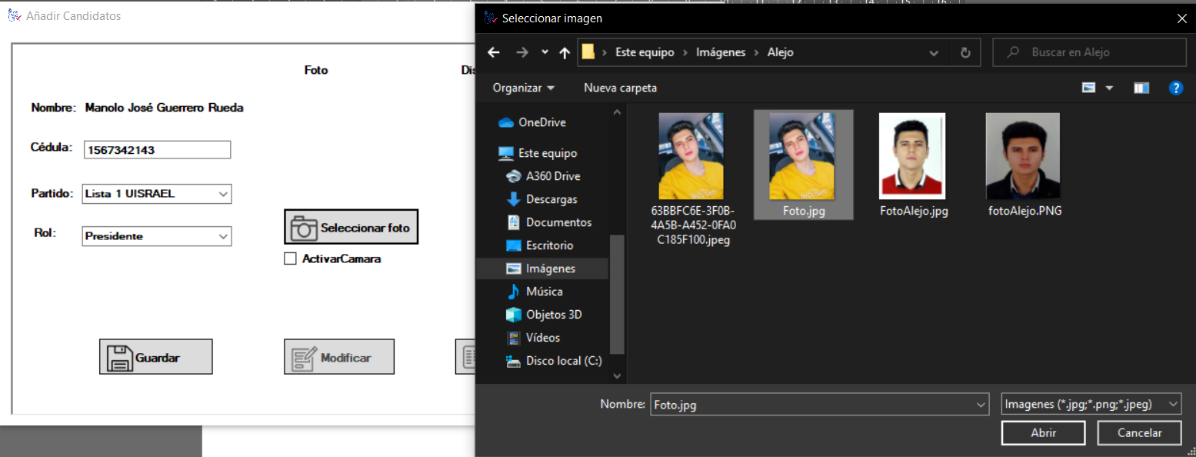
La otra forma es eligiendo una foto de nuestro ordenador, damos clic en ‘’Seleccionar foto’’ y damos clic en abrir.

Figure 18: Subir foto candidato.

Si estamos conformes, guardamos el candidato:



Figure 19: Candidato creado

El mismo proceso lo realizamos para vicepresidente y completar la lista:

Figure 20: Añadir Vicepresidente

Realizaremos las pruebas de votaciones para las elecciones, aquí hay que recalcar que los administradores del sistema no pueden ingresar a la ventana de votaciones ya que se controlo para mantener estándares de seguridad a pesar de ser un prototipo experimental. Para ingresar a votar, es necesario constar en la base de votantes, como candidato o cómo sufragante.

Ingresamos al sistema con una cédula habilitada para sufragar:

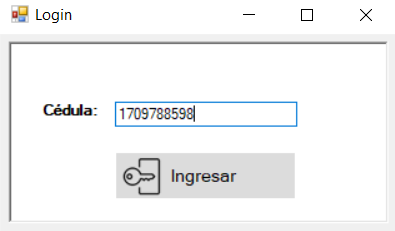


Figure 21: Ingresar login para sufragar.

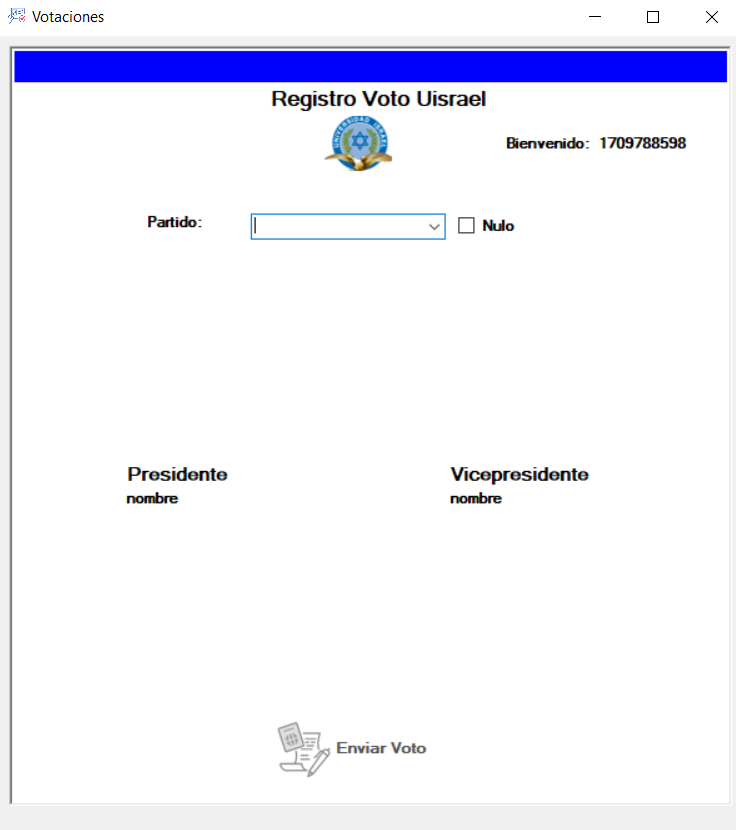
Se habilita la ventana para poder realizar la votación:

Figure 22: Ventana Votaciones

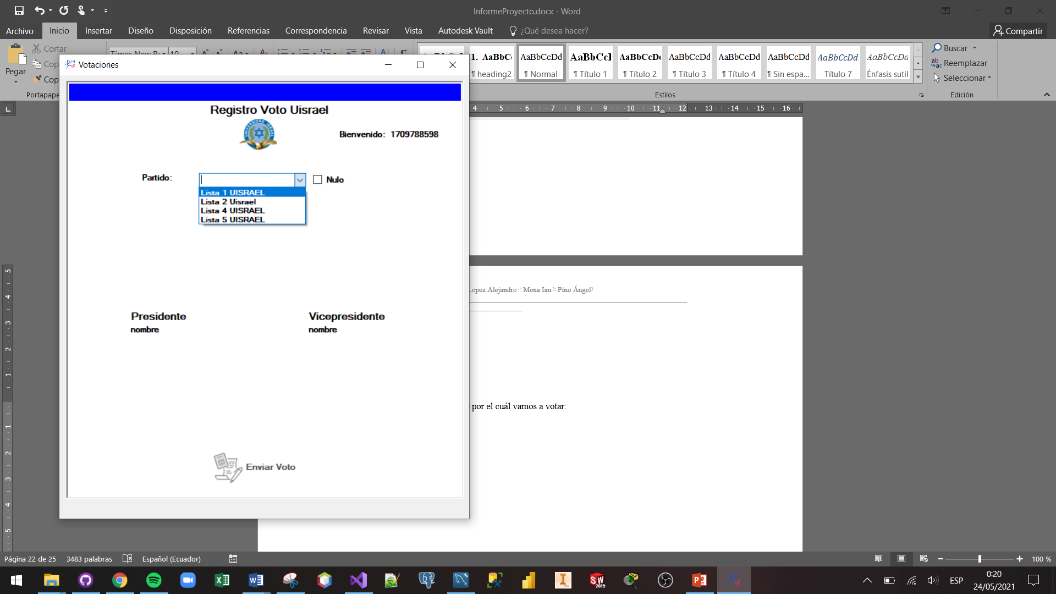
Elegimos el partido por el cuál vamos a votar:

Figure 23: Elegir lista a votar.

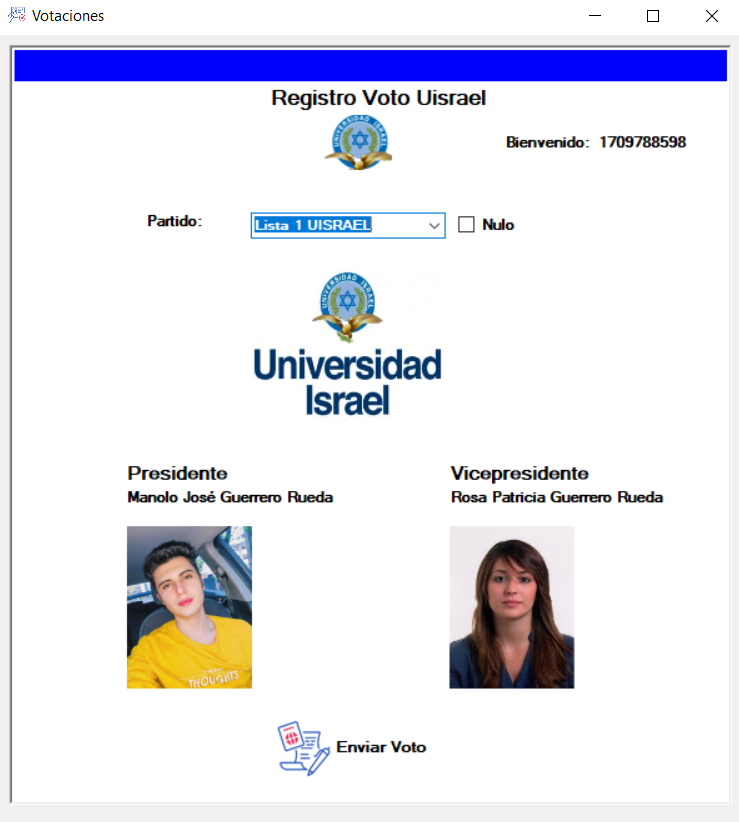
Se cargarán los datos del partido:

Figure 24: Partido cargado.

Si damos clic en el botón ‘’Enviar voto’’ nos pedirá una confirmación para sufragar correctamente:

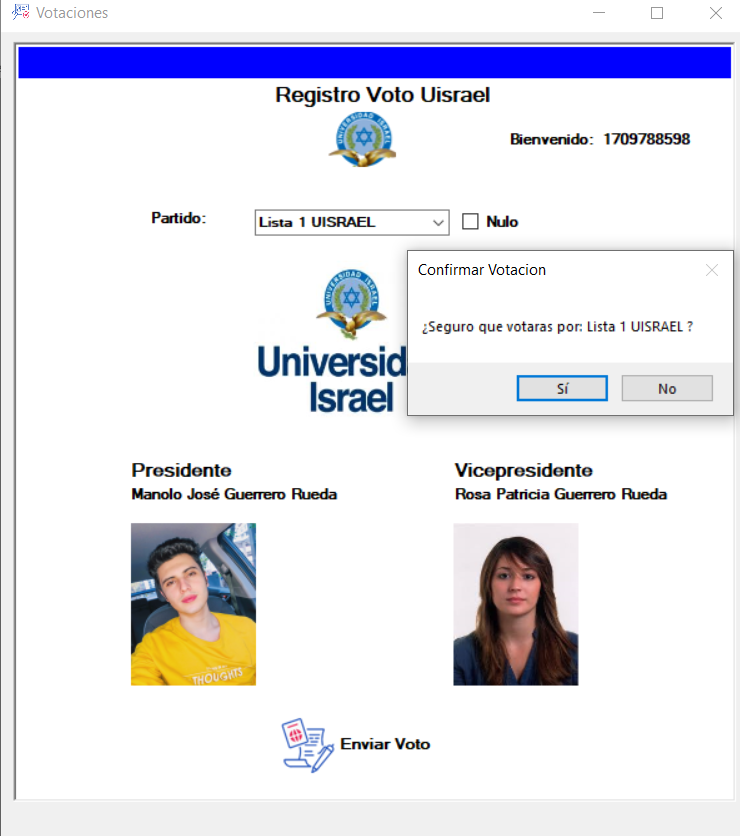


Figure 25: Confirmar voto.

El voto se guarda en la base de datos

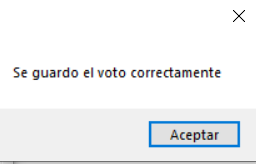


Figure 26: Voto ingresado

Los votos se validan una sola vez para cada cédula, para evitar problemas en el registro y en los posteriores resultados:

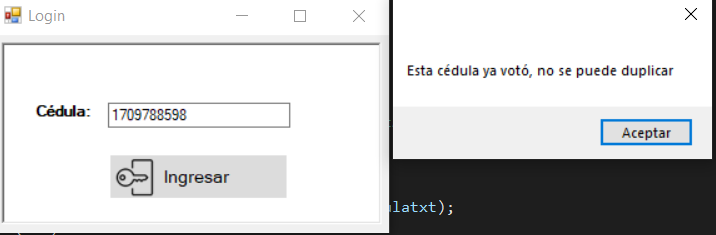


Figure 27: Voto registrado, no permite ingreso a votar nuevamente.

De la misma forma, se pueden realizar votos nulos, ingresamos de igual forma con una cédula habilitada para sufragar:

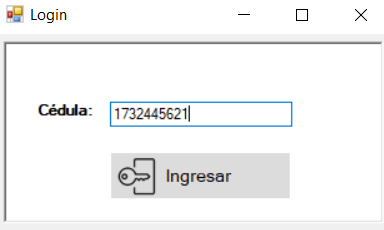


Figure 28: Cédula habilitada, prueba voto nulo.

Seleccionamos nulo y se bloquean todas las opciones para elegir candidatos.

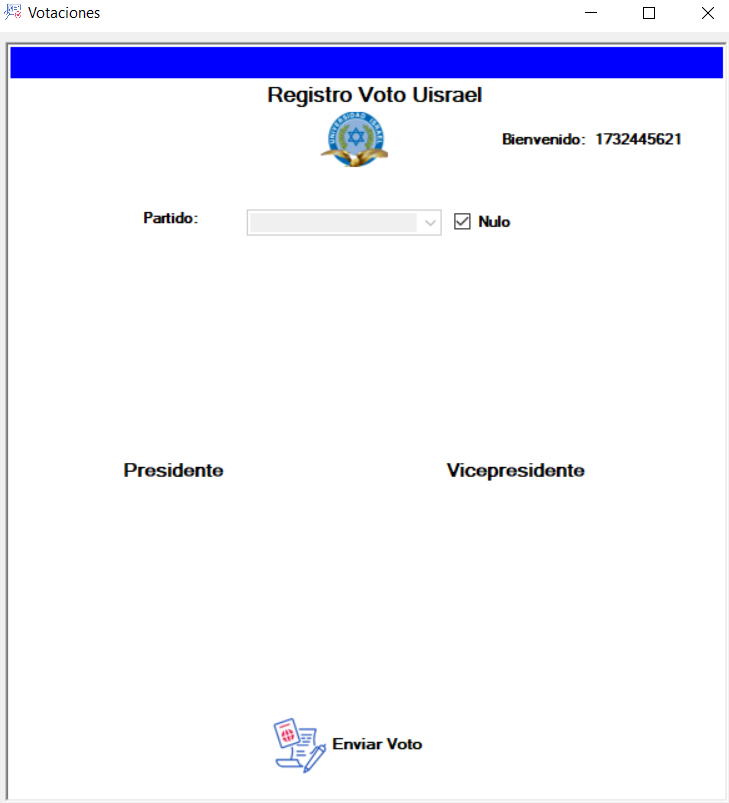


Figure 29: Sufragio nulo

El programa de igual forma pide confirmar el voto:

Figure 30: Confirmación Voto Nulo.

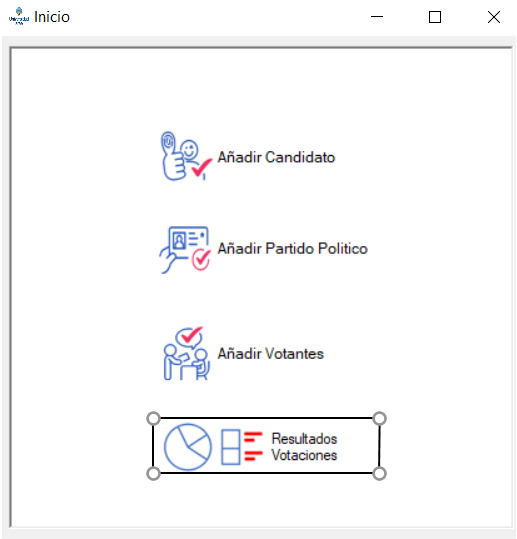
Por último, ingresamos como usuario habilitado para administrador para poder visualizar los resultados de las votaciones. Ingresamos a la ventana principal del sistema y damos clic en la opción ‘’Resultados Votaciones’’

Figure 31: Resultados Votaciones

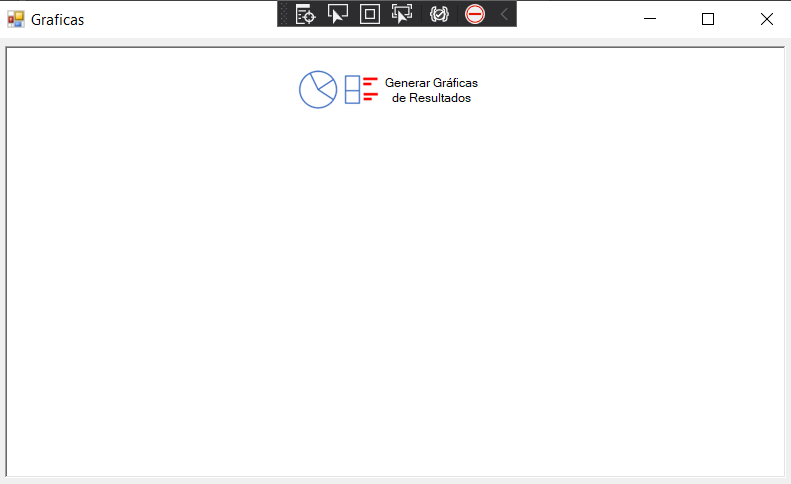
Se desplegará la ventana para obtener la gráfica de los resultados de la siguiente forma:

Figure 32: Gráfica de resultados.

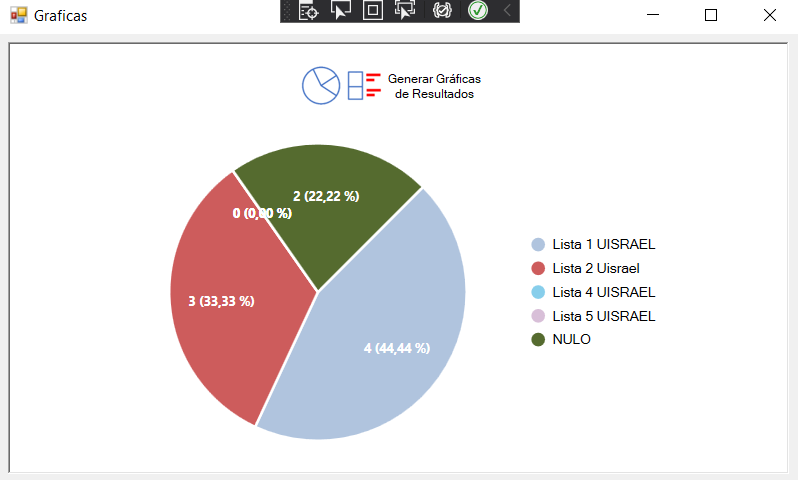
Damos clic en el botón de ‘’Generar Gráficas de Resultados’’ y obtenemos los resultados de las votaciones.

Figure 33: Resultados Votaciones

Podemos colocarnos sobre la gráfica para obtener información más detallada sobre los resultados. Se ha controlado todo aspecto de rendimiento del sistema y se han obtenido resultados satisfactorios en relación al objetivo del programa. La parte experimental ha sido útil para corregir errores dentro de las versiones ejecutadas.

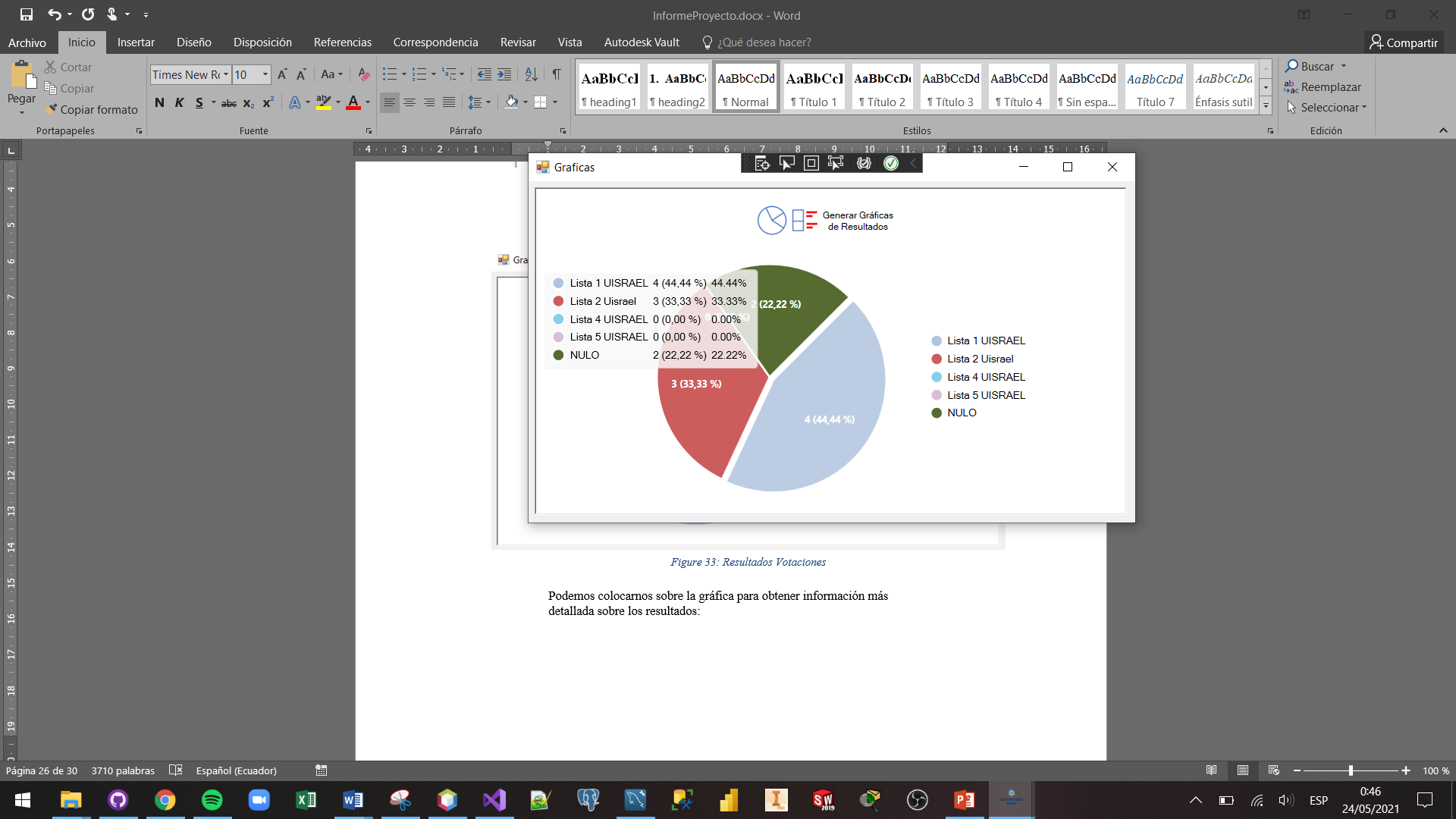


Figure 34: Información resultados elecciones.

1. **CONCLUSIONES**

* El trabajo con formularios es bastante eficiente a la hora de dar un diseño responsive que mantenga un estándar claro y con tendencias actuales de desarrollo.
* En la programación nada está escrito en piedra, y Visual Studio con el lenguaje de programación Csharp, se recomienda aprenderlo ya que es un entorno muy amplio que busca actualizarse constantemente y además existe gran cantidad de información sobre cómo desarrollar con este IDE, lo cual ayuda mucho al trabajo y al aprendizaje.
* Los gráficos estadísticos nos ayudan a interpretar y comprender de mejor manera los datos, aprender a dibujarlos mediante C# es bastante bueno ya que tenemos más conocimiento para desarrollar programas más eficientes.
* Los métodos de una clase u objeto se implementan mediante funciones miembro o métodos. Un método es un conjunto de instrucciones que realizan una determinada tarea.
* Los atributos de una clase son las características individuales que diferencian un objeto de otro y determinan su apariencia, estado u otras cualidades.
* Modelo Vista Controlador (MVC) es un estilo de arquitectura de software que separa los datos de una aplicación, la interfaz de usuario, y la lógica de control en tres componentes distintos.
* El MVC posee múltiples vistas a partir del mismo modelo, pudiendo reaprovechar mucho mejor los desarrollos y asegurando consistencia entre ellas.
* MVC es una propuesta de arquitectura del software utilizada para separar el código por sus distintas responsabilidades, manteniendo distintas capas que se encargan de hacer una tarea muy concreta.
* El programa posee una conexión a base de datos tipo SQL, esta base debe mantener una normalización mínima 3ra forma normal para evitar inconsistencias en los datos.
* Los sistemas que son desarrollados con bases de datos sólidas, son mucho más eficientes en cuestiones como integridad de los datos, menor redundancia, coherencia de resultados y balance para obtener los datos.

1. **REFERENCIAS**
2. Arias, Á. (2014). *Aprende a Programar ASP.NET Y C#.* Publicado por IT Campus Academy. Obtenido de https://books.google.com.ec/books?id=P2rjCgAAQBAJ&printsec=frontcover&source=gbs\_ge\_summary\_r&cad=0#v=onepage&q&f=false
3. Deitel, P. J. (2007). *Cómo programar en C# (2a. ed.).* Pearson Educación. Obtenido de https://elibro.net/es/lc/uisrael/titulos/85342
4. Jovani Alberto Jiménez Builes, D. L. (2019). Metodología de desarrollo de software para plataformas educativas robóticas usando ROS-XP. *REVISTA POLITÉCNICA*, 15(30), 55-69. Obtenido de https://revistas.elpoli.edu.co/index.php/pol/article/view/1620
5. Kasiak Tamara, G. D. (1 de Agosto de 2012). *http://sedici.unlp.edu.ar/.* Obtenido de http://sedici.unlp.edu.ar/handle/10915/18976
6. kinsta. (27 de Mayo de 2019). *https://kinsta.com/*. Obtenido de https://kinsta.com/es/base-de-conocimiento/que-es-mysql/
7. Madrid, U. P. (18 de Junio de 2018). *https://www.fi.upm.es/.* Obtenido de https://www.fi.upm.es/docs/estudios/postgrado-admision/537\_Seminario%20ESE.pdf
8. Merchán Manzano, O. J. (2017). *http://dspace.uazuay.edu.ec/.* Obtenido de http://dspace.uazuay.edu.ec/handle/datos/7036
9. Norte, U. T. (s.f.). *http://www.utn.edu.ec/*. Obtenido de http://www.utn.edu.ec/reduca/programacion/poo/diagrama\_de\_clase.html
10. Sierra, F. J. (2015). *Ceballos Sierra, F. J. (2015). Enciclopedia de Microsoft Visual C#: interfaces gráficas y aplicaciones para Internet con Windows Forms y ASP.NET (4a. ed.).* RA-MA Editorial. Obtenido de https://elibro.net/es/lc/uisrael/titulos/62510
11. Sierra, F. J. (2015). *Microsoft C#: lenguaje y aplicaciones (2a. ed.).* RA-MA Editorial . Obtenido de https://elibro.net/es/ereader/uisrael/62462

1. Estudiante: Alejandro López Medina – Universidad Tecnológica Israel

   Estudiante: Ian Mena – Universidad Tecnológica Israel

   Estudiante: Miguel Pino – Universidad Tecnológica Israel [↑](#footnote-ref-1)