



UNIVERSIDAD  
PRIVADA  
DEL NORTE

## FACULTAD DE INGENIERIA

### CARRERA DE INGENIERIA DE SISTEMAS

**Desarrollo y Documentación Integral del Proyecto de Software:**

“ INFORME ACTUALIZADO - SEMANA 13 ”

**Curso:** TÉCNICAS DE PROGRAM. ORIE. OBJ.

**Ciclo:** V Ciclo

**Docente:** Ing. Jose Carlos Anicama Silva

**Alumnos:** - Percy Aron Benavides Araujo

- Kervyn Rojas Poma

- Jefferson Guizado Mamani

**CLASE:** 11367

**Lima – Perú**

**2025**

# 1. Recolección y documentación de requerimientos ac

## Requerimientos Funcionales

Los **40 requerimientos funcionales** han sido documentados para describir las funcionalidades del sistema. A continuación, se muestra la lista completa, indicando los que **han sido implementados hasta el momento (21)** y los que **no han están pendientes** aún.

Tabla de Requerimientos Funcionales y su Estado.

Nº	Requerimiento Funcional	Estado
1	Registrar un candidato en un ArrayList	Completado
2	Listar todos los candidatos	Completado
3	Eliminar un candidato	Pendiente
4	Modificar un candidato	Pendiente
5	Mostrar detalles de un candidato	Completado
6	Registrar votos de usuarios	Completado
7	Mostrar resultados parciales	Completado
8	Contar votos por candidato	Completado
9	Calcular el porcentaje de votos	Completado
10	Limitar un voto por usuario	Completado
11	Permitir votar solo después de iniciar sesión	Completado
12	Controlar votos por IP	Pendiente
13	Mostrar el estado de la votación	Completado
14	Interfaz gráfica de usuario	Completado
15	Botones de votación	Completado
16	Mostrar resultados finales	Completado
17	Mostrar el ganador	Completado
18	Verificar si hay empate	Completado
19	Botón de "Finalizar votación"	Completado
20	Interfaz de login	Completado
21	Login de usuario	Completado
22	Autenticación de usuario	Completado
23	Registro de nuevos usuarios	Completado
24	Recuperación de contraseña	Pendiente

Nº	Requerimiento Funcional	Estado
25	Limitar intentos de login	Pendiente
26	Historial de votación por usuario	Completado
27	Seguridad de la contraseña	Completado
28	Conectar con base de datos SQL Server	Pendiente
29	Almacenar votos en base de datos	Pendiente
30	Almacenar resultados en la base de datos	Pendiente
31	Validar datos de usuario	Pendiente
32	Obtener votos de la base de datos	Pendiente
33	Manejo de sesiones	Completado
34	Control de acceso	Completado
35	Verificación de estado de la votación	Completado
36	Generar informe de resultados	Completado
37	Crear/Eliminar Candidatos	Pendiente
38	Modificar configuración de votación	Pendiente
39	Exportar resultados a archivo	Pendiente
40	Activar/desactivar votación	Completado

## 2. Historias de Usuario

Nº	Historia de Usuario
1	Como votante, quiero registrar mi voto para un candidato.
2	Como administrador, quiero ver los resultados de la votación en tiempo real.
3	Como sistema, quiero identificar al ganador y verificar si hay empate.
4	Como votante, quiero saber el porcentaje de votos de cada candidato.
5	Como usuario, quiero ingresar con usuario y contraseña para poder votar.
6	Como votante, quiero ver la lista completa de candidatos antes de votar.
7	Como administrador, quiero poder editar o eliminar candidatos antes de iniciar.
8	Como sistema, quiero registrar la hora y fecha de cada voto.
9	Como votante, quiero recibir una confirmación visual después de votar.
10	Como administrador, quiero finalizar manualmente la votación.

## 3. Criterios de Aceptación Verificados

Historia N°	Criterio de Aceptación
1	El voto se almacena al presionar el botón y se refleja en el contador
2	Los resultados se actualizan automáticamente en la interfaz
3	El sistema determina el ganador con base en el mayor número de votos
4	Se muestra una lista previa de candidatos antes de habilitar los botones de votación
5	Solo se accede al sistema si las credenciales coinciden con el usuario registrado
6	El sistema desactiva los botones de votación tras emitir un voto
7	Aparece un mensaje de "voto registrado" después de votar
8	El botón "Finalizar votación" desactiva toda posibilidad de interacción posterior
9	El sistema muestra porcentajes de votos al cerrarse la votación
10	Un letrero indica si la votación está abierta o cerrada

## 4. Evidencias Gráficas (Capturas de Pantalla)

Sistema de Votación Electrónica v1.0

### SISTEMA DE VOTACIÓN ELECTRÓNICA

Simulador para Elecciones Internas

Registro de Candidatos Sistema de Votación

Registro de Candidatos

Número:   
Nombre:   
Partido:

Registrar Candidato  Limpiar Campos

Candidatos Registrados

== CANDIDATOS REGISTRADOS ==

Candidato #1  
Nombre: percy  
Partido: popular  
Votos: 0

Candidato #2  
Nombre: Willy  
Partido: lapiz  
Votos: 0

Candidato #3  
Nombre: Jefferson  
Partido: K  
Votos: 0

Candidato #4  
Nombre: Yetly  
Partido: Fuerzo popular

© 2024 Sistema de Votación Electrónica - Versión 1.0

Use las pestañas para navegar entre las diferentes funciones del sistema.

Sistema de Votación Electrónica v1.0

### SISTEMA DE VOTACIÓN ELECTRÓNICA

Simulador para Elecciones Internas

Registro de Candidatos Sistema de Votación

Sistema de Votación

Estado: Votación no iniciada Total de votos: 0

Número del candidato:

Resultados en Vivo

== RESULTADOS EN TIEMPO REAL ==

Candidato #1: percy (popular)  
Votos: 0 (0,00%)

Candidato #2: Willy (lapiz)  
Votos: 0 (0,00%)

Candidato #3: Jefferson (K)  
Votos: 0 (0,00%)

Candidato #4: Yetly (Fuerzo popular)  
Votos: 0 (0,00%)

## 5. Lecciones Aprendidas

- La planificación en equipo mediante tareas pequeñas (SCRUM) ha sido clave para mantener un ritmo de avance constante y efectivo. Al dividir el proyecto en entregas semanales, logramos distribuir responsabilidades de manera equitativa y mantener una visión clara del progreso. Esta metodología nos permitió adaptarnos mejor a los cambios y corregir errores con rapidez, lo cual fue fundamental para cumplir con los plazos establecidos.
- Evitar adelantarse a requerimientos no solicitados fue una lección valiosa que nos ayudó a mantenernos enfocados en los objetivos concretos de cada etapa. En semanas anteriores, cometimos el error de incluir funcionalidades como la conexión a base de datos antes de que fueran requeridas, lo cual desvió nuestra atención de lo realmente importante en ese momento. Aprendimos que respetar el alcance permite una gestión más eficiente de tiempo y de los recursos del equipo.
- Realizar pruebas desde las primeras fases de desarrollo fue una práctica fundamental para evitar acumulación de errores. Detectar fallos tempranos nos permitió corregirlos de forma oportuna sin comprometer la estabilidad del sistema. Asimismo, probar los módulos conforme se implementaban facilitó la integración progresiva del sistema y fortaleció nuestra confianza en el funcionamiento general del simulador.
- Mantener una estructura de código organizada en capas (modelo, vista, controlador) resultó ser una estrategia esencial para la claridad, escalabilidad y mantenimiento del proyecto. Esta separación de responsabilidades no solo nos permitió trabajar en paralelo como equipo, sino también facilitó la identificación de errores, el refactorizado del código y la implementación de mejoras sin afectar el resto del sistema.

## 6. Conclusión Grupal Reflexiva

Como equipo, reconocemos que este proyecto ha sido una experiencia integral que nos permitió afianzar de manera sólida nuestros conocimientos técnicos en programación orientada a objetos, aplicando conceptos como encapsulamiento, herencia y modularidad en un entorno práctico. La creación del simulador de votación no solo representó un reto técnico, sino también una oportunidad para mejorar nuestras habilidades blandas, tales como la colaboración efectiva, la planificación estructurada y la toma de decisiones en equipo bajo presión.

Durante el desarrollo, aprendimos a analizar e interpretar requerimientos funcionales de forma precisa, priorizando los más relevantes según el alcance semanal y evitando adelantarnos innecesariamente a funcionalidades que aún no se solicitaban. Esto nos enseñó la importancia de trabajar con enfoque iterativo e incremental, alineado con metodologías ágiles como SCRUM, lo cual nos ayudó a dividir tareas complejas en metas alcanzables y bien distribuidas.

Además, enfrentamos desafíos reales de coordinación, interpretación de requerimientos, revisión de código y validación de funcionalidades. Estos desafíos nos llevaron a desarrollar habilidades de autoevaluación y retroalimentación constante entre los miembros del grupo, lo cual fortaleció nuestro compromiso con el cumplimiento de los objetivos y la calidad del trabajo entregado.

Finalmente, nos sentimos satisfechos con el avance logrado y motivados para continuar mejorando el simulador en futuras entregas. Nos comprometemos a seguir fortaleciendo tanto nuestras competencias técnicas como nuestras capacidades de trabajo en equipo, con la convicción de que cada entrega representa una oportunidad de aprendizaje valioso y aplicación real del conocimiento adquirido.