**Actividad integradora. Sprint, el corazón de SCRUM**

Materia: Proyecto II

Alumno: Orlando Joaquín Pizano Prado

Backlog actualizado con calendarización se encuentra en el archivo Excel adjunto.

**FICHAS DE BACKLOG**

**Historia de Usuario: US-001 - Registro e inicio de sesión de un usuario**

**Descripción:** Como usuario, quiero poder registrarme e iniciar sesión para acceder a mi cuenta. **Criterios de Aceptación:**

* El usuario debe poder registrarse con correo y contraseña.
* Debe existir validación de datos.
* El sistema debe permitir el inicio de sesión con credenciales válidas.

**Tareas:**

1. Diseñar base de datos para usuarios (4 horas)
2. Implementar formulario de registro (6 horas)
3. Desarrollar API para el registro de usuarios (5 horas)

**Historia de Usuario: US-002 - Perfil de usuario con datos personales**

**Descripción:** Como usuario, quiero un perfil donde pueda ver y editar mis datos personales. **Criterios de Aceptación:**

* El usuario debe poder visualizar su información.
* Debe poder modificar sus datos personales.
* Los cambios deben guardarse correctamente en la base de datos.

**Tareas:**

1. Crear interfaz de perfil de usuario (5 horas)
2. Conectar la interfaz con la API de usuario (4 horas)
3. Implementar función de edición y guardado (6 horas)

**Historia de Usuario: US-003 - Panel de administración para la mesa directiva**

**Descripción:** Como miembro de la mesa directiva, quiero acceder a un panel de administración para gestionar residentes, noticias y votaciones de manera eficiente.

**Criterios de Aceptación:**

* La mesa directiva debe poder acceder a un panel exclusivo.
* Desde el panel, deben poder gestionar usuarios, publicar noticias y administrar encuestas.
* Debe haber restricciones de acceso según el rol del usuario.

**Tareas:**

1. Diseñar la interfaz del panel de administración (6 horas)
2. Implementar gestión de usuarios desde el panel (7 horas)
3. Crear módulo de administración de encuestas y votaciones (8 horas)

**Historia de Usuario: US-004 - Módulo de noticias y boletines informativos**

**Descripción:** Como residente, quiero acceder a noticias y comunicados de la mesa directiva desde la plataforma.

**Criterios de Aceptación:**

* La mesa directiva debe poder publicar noticias y boletines.
* Los residentes deben poder visualizar las noticias.
* Las noticias deben mostrarse en orden cronológico.

**Tareas:**

1. Diseñar la interfaz de noticias y boletines (5 horas)
2. Implementar la funcionalidad de publicación de noticias (6 horas)
3. Crear un sistema de organización de noticias por fecha (4 horas)

**Historia de Usuario: US-005 - Funcionalidad de encuestas para residentes**

**Descripción:** Como residente, quiero participar en votaciones de la mesa directiva para expresar mi opinión sobre decisiones importantes.

**Criterios de Aceptación:**

* Los residentes deben poder visualizar encuestas activas.
* Deben poder emitir un voto por encuesta.
* Los resultados deben actualizarse en tiempo real.

**Tareas:**

1. Diseñar la interfaz de encuestas para residentes (5 horas)
2. Implementar sistema de votación con restricciones (7 horas)
3. Crear visualización de resultados en tiempo real (6 horas)

**Historia de Usuario: US-006 - Integración con pagos en línea**

**Descripción:** Como residente, quiero realizar pagos en línea de mis cuotas de mantenimiento de manera segura y rápida.

**Criterios de Aceptación:**

* Los residentes deben poder ingresar el monto y método de pago.
* Debe haber validación de pago exitoso o fallido.
* La información de pago debe guardarse en el sistema.

**Tareas:**

1. Implementar pasarela de pagos en la plataforma (8 horas)
2. Crear validaciones de pago exitoso/fallido (6 horas)
3. Registrar pagos en la base de datos (5 horas)

**Historia de Usuario: US-007 - Sistema de notificaciones por correo y en la web**

**Descripción:** Como residente, quiero recibir notificaciones sobre eventos importantes en el residencial para mantenerme informado.

**Criterios de Aceptación:**

* El sistema debe enviar notificaciones por correo y dentro de la plataforma.
* Los residentes deben poder activar o desactivar notificaciones.
* Debe haber una sección de notificaciones en la plataforma.

**Tareas:**

1. Implementar sistema de notificaciones internas (5 horas)
2. Configurar envíos de correos automáticos (6 horas)
3. Crear opción para activar/desactivar notificaciones (4 horas)

**Historia de Usuario: US-008 - Seguridad y control de accesos**

**Descripción:** Como administrador, quiero garantizar la seguridad de los datos del sistema restringiendo accesos según roles de usuario.

**Criterios de Aceptación:**

* Se deben definir roles con permisos específicos.
* Un usuario no autorizado no debe acceder a funciones restringidas.
* Debe haber control de acceso a datos sensibles.

**Tareas:**

1. Implementar sistema de roles y permisos (6 horas)
2. Restringir accesos a secciones protegidas (5 horas)
3. Crear pruebas de seguridad para verificar accesos (4 horas)

**Historia de Usuario: US-009 - Diseño responsive para móviles**

**Descripción:** Como residente, quiero acceder a la plataforma desde mi celular sin problemas de visualización.

**Criterios de Aceptación:**

* La interfaz debe adaptarse a diferentes tamaños de pantalla.
* Deben aplicarse estilos adecuados para móviles.
* Las funciones deben ser accesibles desde cualquier dispositivo.

**Tareas:**

1. Implementar diseño responsive en toda la plataforma (8 horas)
2. Probar compatibilidad en distintos dispositivos (5 horas)
3. Ajustar estilos CSS para mejorar la experiencia móvil (4 horas)

**Historia de Usuario: US-010 - Pruebas de calidad y rendimiento**

**Descripción:** Como equipo de desarrollo, queremos asegurar la calidad del sistema realizando pruebas antes de su lanzamiento.

**Criterios de Aceptación:**

* Se deben realizar pruebas funcionales en cada módulo.
* Se deben probar cargas altas en la plataforma.
* Deben detectarse y corregirse errores antes del lanzamiento.

**Tareas:**

1. Diseñar casos de prueba para cada funcionalidad (6 horas)
2. Implementar pruebas de carga con usuarios simulados (7 horas)
3. Corregir errores detectados y realizar pruebas finales (8 horas)

**Calendarización de Sprints**

**Sprint 1:** 26 de febrero - 10 de marzo

* Implementación de registro e inicio de sesión
* Creación de perfil de usuario

**Sprint 2:** 11 de marzo - 24 de marzo

* Desarrollo del panel de administración
* Creación del módulo de noticias y boletines

**Sprint 3:** 25 de marzo - 7 de abril

* Funcionalidad de encuestas para residentes
* Integración con pagos en línea

**Sprint 4:** 8 de abril - 21 de abril

* Sistema de notificaciones
* Seguridad y control de accesos

**Sprint 5:** 22 de abril - 5 de mayo

* Diseño responsive
* Pruebas de calidad y rendimiento

**Link del proyecto en GitHub:**

<https://github.com/orlando5353/ProyectoII>

**Bibliografía:**

* García N., E. (2019, febrero 18). ¡Construye tu primera aplicación web con ReactJS! *Medium*. Recuperado de <https://medium.com/@ernestognw/construye-tu-primera-aplicaci%C3%B3n-web-con-reactjs-parte-1-4c8bcca3b9db>
* Academia Join. (2015, mayo 27). *Scrum Ejemplo Práctico de la Planeación del Sprint* [Archivo de video]. YouTube. Recuperado de <https://youtu.be/BNCDcGqYNwk>
* Luque, A. (2016, mayo 15). *Ejercicio Product Backlog y Sprint Backlog* [Archivo de video]. YouTube. Recuperado de <https://www.youtube.com/watch?v=ySJMHCI-8Uo&t=2s>