

ACTIVIDAD INTEGRADORA

Proyecto II

**Elaborado por :Tammy Karmina
Espinosa Rosete**

BACKLONG Priorización de los elementos

Elaborado por: **Tammy Karmina Espinosa Rosete**

Identificador (ID) de la Historia	Enunciado de la Historia	Alias	Estado	Dimensión/ Esfuerzo	Interacción (Sprint)	Prioridad	Comentarios
01UH-PP	Como residente de la comunidad robles quiero que exista una pagina principal para observar las opciones de la pagina web	Pagina de inicio	Concluido	5 punto s	Sprint 1 (2 de febrero a 9 febrero)	Mediana Alta	Requiere de organización de como se acomodaran para tener visibles todas las opciones
02UH- AC	Como residente de la Comunidad Robles quiero acceder a un apartado en el sitio web donde se publiquen informaciones y actualizaciones sobre los cambios que se realizarán en la residencia, para estar al tanto de las decisiones y mejoras que afecten a la comunidad.	Informes actualizaciones y cambios	Por hacer	8 puntos	Sprint 2 (9 de febrero a 16 febrero)	Mediana Alta	Requiere coordinación con el equipo de administración para la relevancia de la información publicada.
03UH-VC	Como residente de la Comunidad Robles, quiero poder acceder a un formulario en el sitio web para emitir mi voto en las decisiones comunitarias	Votación comunitaria	En proceso	8 puntos	Sprint 3 (16 de febrero a 23 febrero)	Mediana Alta	Requiere integración con base de datos de residentes para verificar la autenticidad de los votos

04UH-CP	Como residente, quiero acceder a un calendario en el sitio web donde estén marcados los límites de pago de las cuotas mensuales, para recordar las fechas de vencimiento	Calendario fechas limites de Pagos	Por hacer	5 puntos	Sprint 4 (23 de febrero a 1 de marzo)	Mediana	Requiere integración con el sistema de gestión financiera para obtener las fechas de vencimiento de las cuotas mensuales.
05UH-CA	Como residente, quiero acceder a un apartado en el sitio web donde pueda consultar mis adeudos pendientes y los pagos realizados previamente, para llevar un registro claro de mis obligaciones financieras con la comunidad.	Consulta de adeudos	Por hacer	3 puntos	Sprint 5(1 de marzo a 8 de marzo)	Mediana Baja	Integración con el sistema de gestión financiera para obtener la información de adeudos y pagos de manera precisa
06UH-MP	Como residente, quiero tener la capacidad de levantar un reporte de emergencia en el sistema web o la aplicación móvil en caso de detectar un problema con algún servicio público o privado, especificando el tipo de circunstancia.	Reporte de emergencia	Por hacer	8 puntos	Sprint 6 (8 de marzo a 15 de marzo)	Mediana Alta	Requiere desarrollo de la funcionalidad para levantar reportes de emergencia, integración con autoridades

FICHAS DE BACKLOG

Ficha de Backlog: Página de Inicio (Alias: 01UH-PP)

Descripción del Elemento: Como residente de la comunidad Robles, quiero que exista una página principal en el sitio web donde pueda observar las opciones disponibles.

Prioridad: Mediana Alta

Estado: Concluido

Estimación de Esfuerzo: 5 puntos

Interacción (Sprint): Sprint 1 (2 de febrero a 9 de febrero)

Requerimientos Adicionales: Requiere de organización para acomodar las opciones de manera que sean visibles y accesibles para los usuarios.

Ficha de Backlog: Informes, Actualizaciones y Cambios (Alias: 02UH-AC)

Descripción del Elemento: Como residente de la Comunidad Robles, quiero acceder a un apartado en el sitio web donde se publiquen informaciones y actualizaciones sobre los cambios que se realizarán en la residencia, para estar al tanto de las decisiones y mejoras que afecten a la comunidad.

Prioridad: Mediana Alta

Estado: Por hacer

Estimación de Esfuerzo: 8 puntos

Interacción (Sprint): Sprint 2 (9 de febrero a 16 de febrero)

Requerimientos Adicionales: Requiere coordinación con el equipo de administración para la relevancia de la información publicada.

Ficha de Backlog: Votación Comunitaria (Alias: 03UH-VC)

Descripción del Elemento: Como residente de la Comunidad Robles, quiero poder acceder a un formulario en el sitio web para emitir mi voto en las decisiones comunitarias.

Prioridad: Mediana Alta

Estado: En proceso

Estimación de Esfuerzo: 8 puntos

Interacción (Sprint): Sprint 3 (16 de febrero a 23 de febrero)

Requerimientos Adicionales: Requiere integración con base de datos de residentes para verificar la autenticidad de los votos.

Ficha de Backlog: Calendario de Fechas Límites de Pagos (Alias: 4UH-CP)

Descripción del Elemento: Como residente, quiero acceder a un calendario en el sitio web donde estén marcados los límites de pago de las cuotas mensuales, para recordar las fechas de vencimiento.

Prioridad: Mediana

Estado: Por hacer

Estimación de Esfuerzo: 5 puntos

Interacción (Sprint): Sprint 4 (23 de febrero a 1 de marzo)

Requerimientos Adicionales: Requiere integración con el sistema de gestión financiera para obtener las fechas de vencimiento de las cuotas mensuales.

Ficha de Backlog: Consulta de Adeudos (Alias: 05UH-CA)

Descripción del Elemento: Como residente, quiero acceder a un apartado en el sitio web donde pueda consultar mis adeudos pendientes y los pagos realizados previamente, para llevar un registro claro de mis obligaciones financieras con la comunidad.

Prioridad: Mediana Baja

Estado: Por hacer

Estimación de Esfuerzo: 3 puntos

Interacción (Sprint): Sprint 5 (1 de marzo a 8 de marzo)Requerimientos

Adicionales:

Integración con el sistema de gestión financiera para obtener la información de adeudos y pagos de manera precisa.

Ficha de Backlog: Reporte de Emergencia (Alias: 06UH-MP)

Descripción del Elemento: Como residente, quiero tener la capacidad de levantar un reporte de emergencia en el sistema web o la aplicación móvil en caso de detectar un problema con algún servicio público o privado, especificando el tipo de circunstancia.

Prioridad: Mediana Alta

Estado: Por hacer

Estimación de Esfuerzo: 8 puntos

Interacción (Sprint): Sprint 6 (8 de marzo a 15 de marzo)

Requerimientos Adicionales:

Requiere desarrollo de la funcionalidad para levantar reportes de emergencia.

Integración con autoridades para la gestión adecuada de los reportes.

SPRINT

Elemento de trabajo pendiente	Puntos de historia	Responsable	Estado	Estimado original	Día 1	Día 2	Día 3	Día 4	Día 5	Revisión del Sprint
01UH-PP	5			4 días						
Diseño de la organización apartado (frontend)	2 puntos	Tammy	Concluido	2 días	Definir organización	/	Pasar a código	/	/	Se obtuvo un diseño agradable y acorde
Desarrollo de las organización de los apartados (backend)	3 puntos	Tammy	Concluido	2 días	/	Desarrollar estructura	/	Definir detalles de funcionalidad	/	Tienes funcionalidad la pagina
02UH-AC	8			5 días						
Diseño del apartado de actualizaciones (frontend)	3 puntos	Tammy	En progreso	2 día	Realizar diseño general	Detallar diseño notificaciones	/	/	/	Se obtuvo un diseño acorde.
Desarrollo de sistema de publicación de actualizaciones (backend)	3puntos	Tammy	En progreso	2 días	/	/	Determinar sistema de info	Programar sistema de info	/	Sistema funcional para hacer publicaciones constantes
Configuración de notificaciones	1 puntos	Tammy	Por hacer	23 horas	/	/	/	/	Programar configuración de notificaciones	Se agregaron las notificaciones

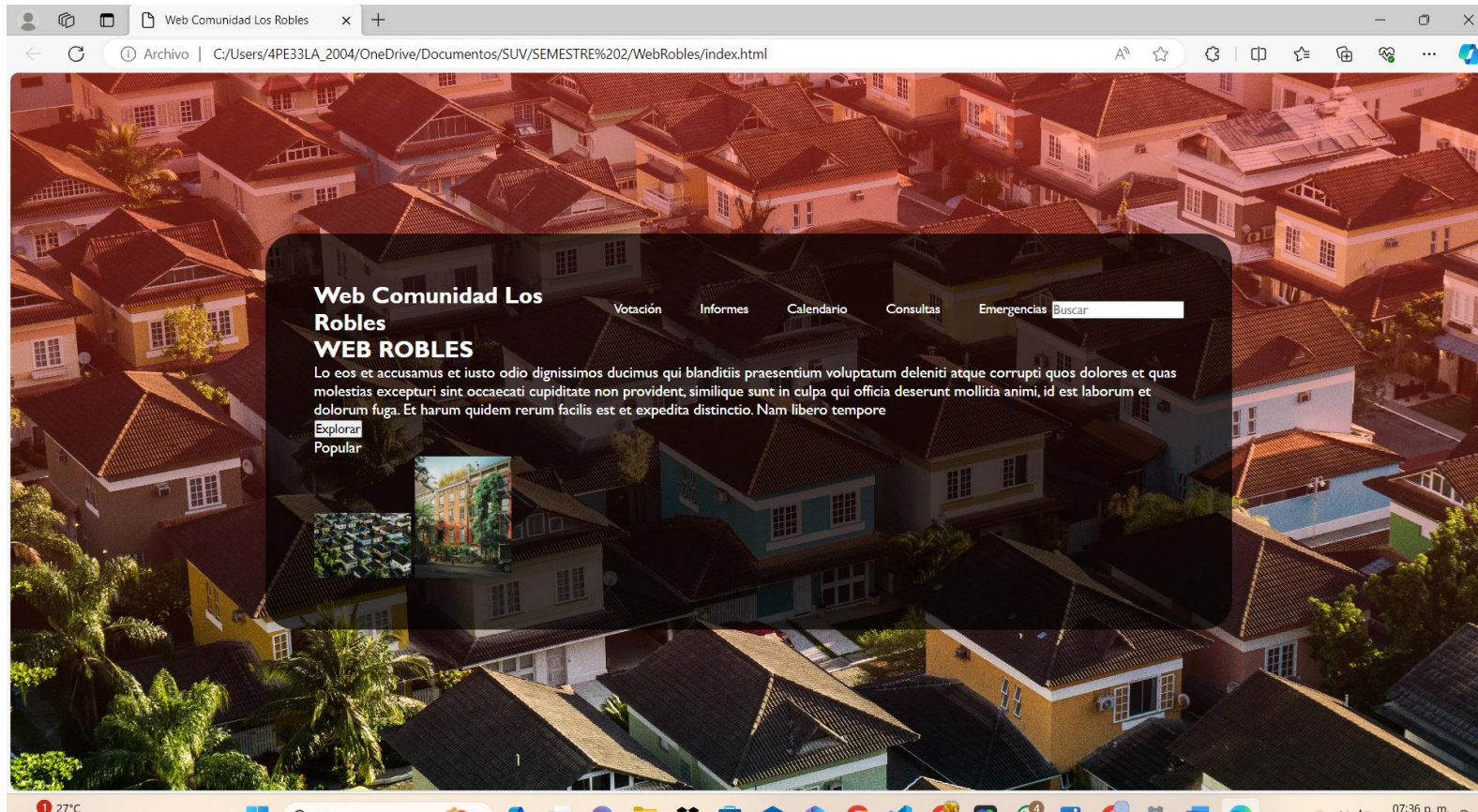
Pruebas de el notificaciones	1 puntos	Tammy	Por hacer	22 horas	/	/	/	/	Realizar pruebas con notificaciones	Las pruebas ayudan a la funcionalidad
03UH- VC	8			5 días						
Diseño del formulario de votación.	2 puntos	Tammy	En proceso	1 día	Realizar diseño	/	/	/	/	Se obtuvo un diseño minimalista por el tiempo reducido
Implementación del backend del formulario.	3 puntos	Tammy	En proceso	2 días	/	Determinar secciones	Programar secciones	/	/	Se desarrollo un formulario
Base de datos autenticación de usuarios	2 punto	Tammy	Por hacer	1 día	/	/	/	Subir base de datos	/	Se necesita actualización de la base de datos
Pruebas del formulario	1 punto	Tammy	Por hacer	20 horas	/	/	/	/	Realizar pruebas	Las pruebas ayudan a la funcionalidad
04UH-CP	5			4 días						
Diseño del calendario de pagos (frontend)	2 puntos	Tammy	Por hacer	1 día	Diseño de calendario	/	/	/	/	El diseño fue minimalista y comprensible
Desarrollo de sistema del calendario de pagos (backend)	1 puntos	Tammy	En proceso	22 horas	/	Programacion del calendario	/	/	/	Se desarrollo la pagina con el calendario

Intregar base de datos de fechas de vencimiento	1 puntos	Tammy	En proceso	20 horas	/	/	Subir base de datos de fechas	/	/	Se necesitara la actualización de la base de datos
Pruebas de interfaz de usuario	1 puntos	Tammy	Por hacer	12 horas	/	/	/	Pruebas de interfaz	/	Las pruebas ayudan a la funcionalidad
05UH-CA	3			3días						
Diseño del apartado de finanzas (frontend)	1	Tammy	Por hacer	22 horas	Diseño del apartado	/	/	/	/	Diseño minimalista por el tiempo reducido
Desarrollo del contenido de finanzas (backend)	1	Tammy	Por hacer	23 horas	/	Programacion del apartado	/	/	/	Actualización constante de base de datos
Gestión de base de datos	1	Tammy	Por hacer	12 horas	/	/	Subir base de datos	/	/	Las pruebas ayudan a una mejor funcionalidad
06UH-MP	8			5 días						
Diseño de interfaz para el reporte de emergencia (frontend)	3 puntos	Tammy	Por hacer	2 días	Diseño de interfaz de la pagina de emergencia	Diseño en especifico de caja de reportes	/	/	/	Diseño minimalista con detalle
Desarrollo de la funcionalidad de reporte de	3 puntos	Tammy	Por hacer	2 día	/	/	Desarrollo de	Desarrollo especifico	/	Desarrollo de caja reporte de emergencias

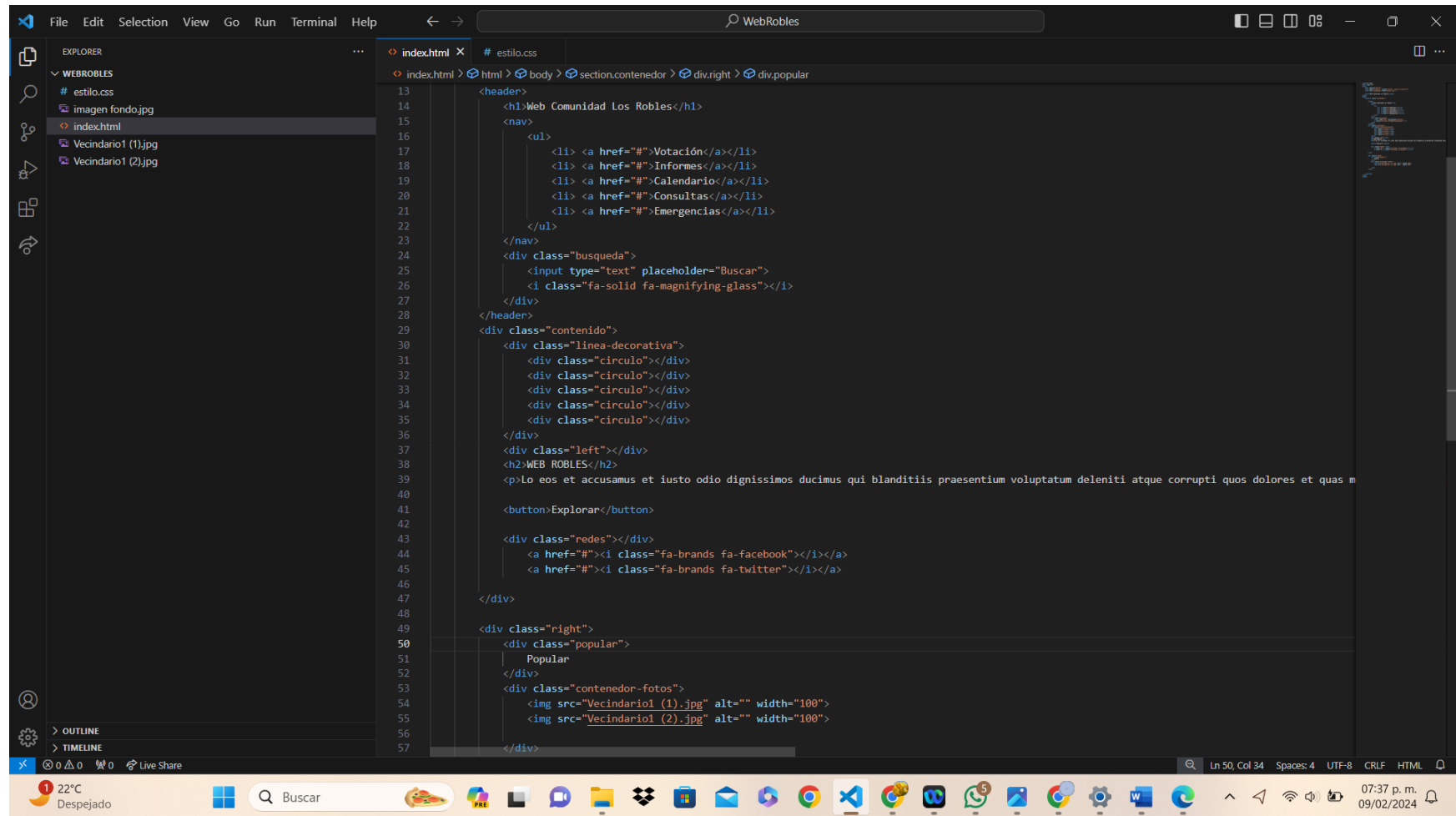
emergencia (backend)							toda la pagina	o de la caja de reportes		
Desarrollo de notificaciones y alertas.	1 puntos	Tammy	Por hacer	12 horas	/	/	/	/	Program ar notificaci ones	Implementación de notificaciones
Pruebas del funcionamiento	1 puntos	Tammy	Por hacer	12 horas	/	/	/	/	Pruebas de funciona miento de la caja de reportes	La pruebas ayudaron a corregir errores.
Total	37 puntos	/	/	28 días aprox	/	/	/	/		

CODIGO Y FUNCIONAMIENTO:

Página de Inicio (Alias: 01UH-PP)



CODIGO HTML

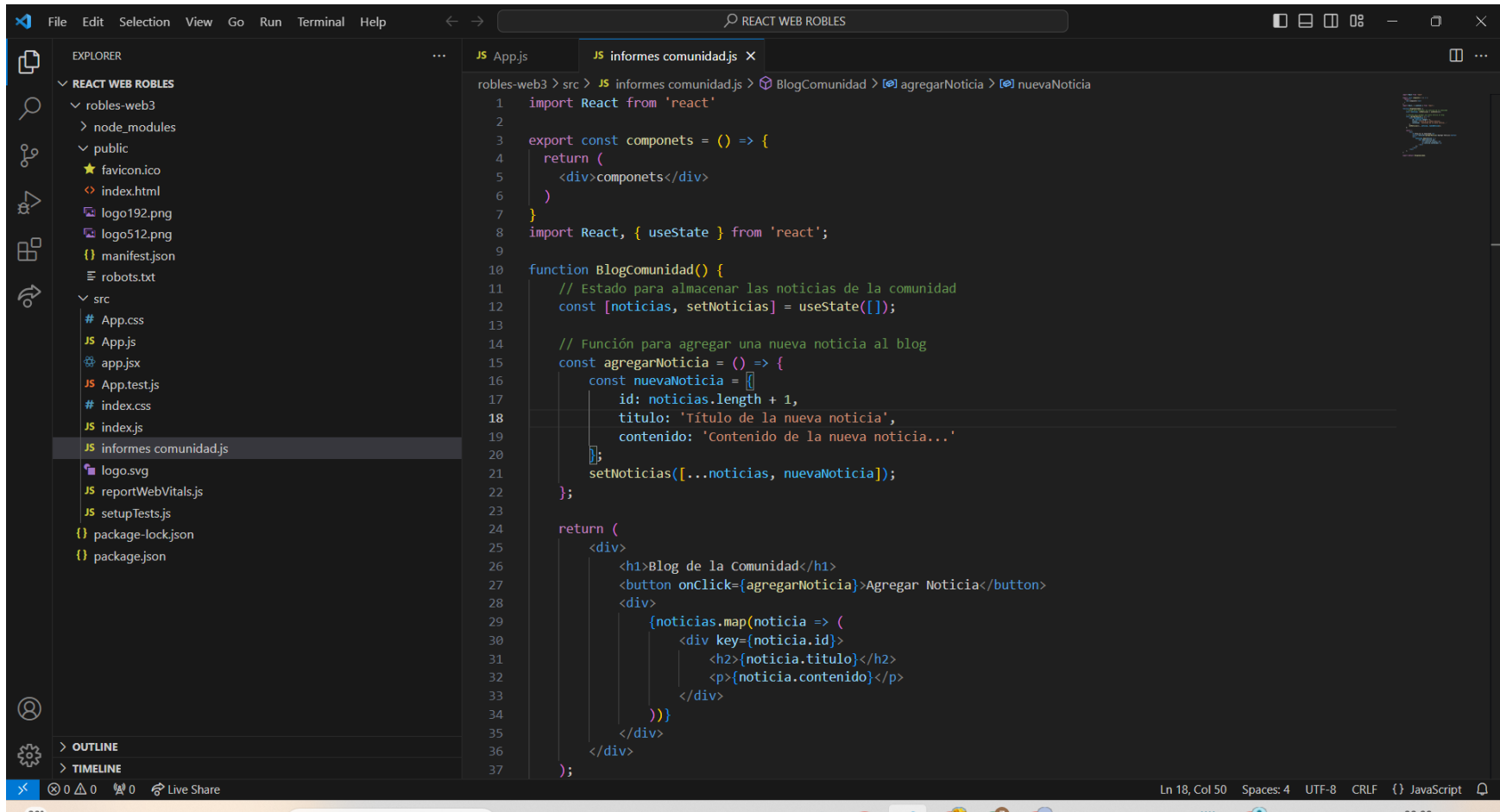


The image shows a Visual Studio Code editor window with a dark theme. The Explorer panel on the left shows a project named 'WEBROBLES' with files: 'estilo.css', 'imagen.fondo.jpg', 'index.html', 'Vecindario1 (1).jpg', and 'Vecindario1 (2).jpg'. The main editor area displays the 'index.html' file, which contains HTML code for a web page. The code includes a header with a title 'Web Comunidad Los Robles', a navigation menu with links to 'Votación', 'Informes', 'Calendario', 'Consultas', and 'Emergencias', a search bar, a decorative line with five circles, a left sidebar, a main content area with a paragraph of placeholder text and an 'Explorar' button, a 'redes' section with Facebook and Twitter links, and a right sidebar with a 'Popular' section and two image placeholders. The status bar at the bottom indicates 'Ln 50, Col 34', 'Spaces: 4', 'UTF-8', 'CRLF', and 'HTML'. The Windows taskbar at the very bottom shows the date '09/02/2024' and time '07:37 p. m.'.

```
13 <header>
14   <h1>Web Comunidad Los Robles</h1>
15   <nav>
16     <ul>
17       <li><a href="#">Votación</a></li>
18       <li><a href="#">Informes</a></li>
19       <li><a href="#">Calendario</a></li>
20       <li><a href="#">Consultas</a></li>
21       <li><a href="#">Emergencias</a></li>
22     </ul>
23   </nav>
24   <div class="busqueda">
25     <input type="text" placeholder="Buscar">
26     <i class="fa-solid fa-magnifying-glass"></i>
27   </div>
28 </header>
29 <div class="contenido">
30   <div class="linea-decorativa">
31     <div class="circulo"></div>
32     <div class="circulo"></div>
33     <div class="circulo"></div>
34     <div class="circulo"></div>
35     <div class="circulo"></div>
36   </div>
37   <div class="left"></div>
38   <h2>WEB ROBLES</h2>
39   <p>Lo eos et accusamus et iusto odio dignissimos ducimus qui blanditiis praesentium voluptatum deleniti atque corrupti quos dolores et quas m
40
41   <button>Explorar</button>
42
43   <div class="redes"></div>
44   <a href="#"><i class="fa-brands fa-facebook"></i></a>
45   <a href="#"><i class="fa-brands fa-twitter"></i></a>
46
47 </div>
48
49 <div class="right">
50   <div class="popular">
51     Popular
52   </div>
53   <div class="contenedor-fotos">
54     
55     
56
57   </div>
```

Nota: La portada se realice en HTML, todo el avance de las paginas aparte lo realice en REACT

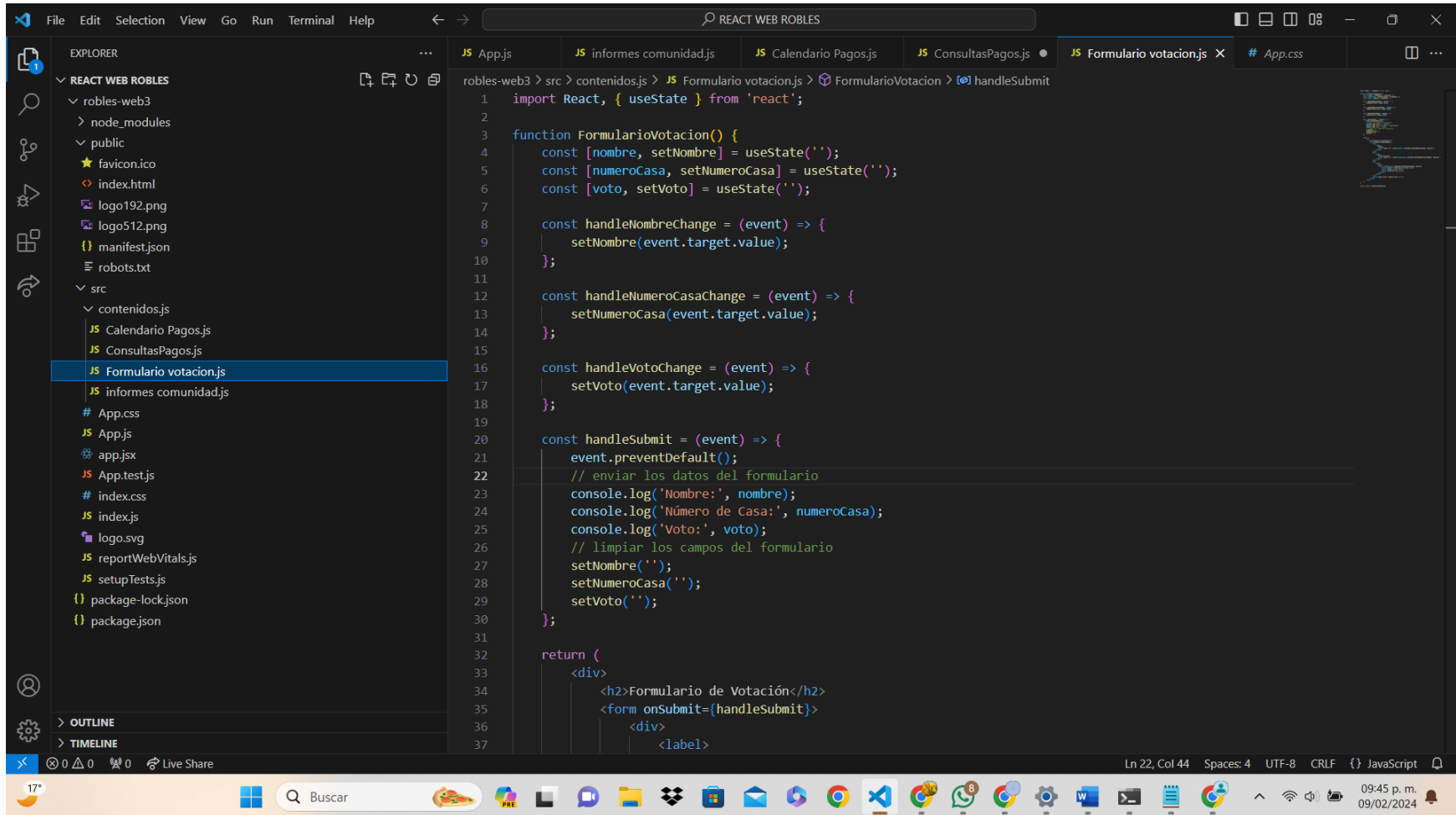
Informes, Actualizaciones y Cambios (Alias: 02UH-AC)



```
1 import React from 'react'
2
3 export const componets = () => {
4   return (
5     <div>componets</div>
6   )
7 }
8 import React, { useState } from 'react';
9
10 function BlogComunidad() {
11   // Estado para almacenar las noticias de la comunidad
12   const [noticias, setNoticias] = useState([]);
13
14   // Función para agregar una nueva noticia al blog
15   const agregarNoticia = () => {
16     const nuevaNoticia = {
17       id: noticias.length + 1,
18       titulo: 'Título de la nueva noticia',
19       contenido: 'Contenido de la nueva noticia...'
20     };
21     setNoticias([...noticias, nuevaNoticia]);
22   };
23
24   return (
25     <div>
26       <h1>Blog de la Comunidad</h1>
27       <button onClick={agregarNoticia}>Agregar Noticia</button>
28       <div>
29         {noticias.map(noticia => (
30           <div key={noticia.id}>
31             <h2>{noticia.titulo}</h2>
32             <p>{noticia.contenido}</p>
33           </div>
34         ))}
35       </div>
36     </div>
37   );
38 }
```

Nota: se desarrollo como si fuera un blog de noticas donde puedan comentar los comunitarios

Votación Comunitaria (Alias: 02UH-VC)



```
1 import React, { useState } from 'react';
2
3 function FormularioVotacion() {
4   const [nombre, setNombre] = useState('');
5   const [numeroCasa, setNumeroCasa] = useState('');
6   const [voto, setVoto] = useState('');
7
8   const handleNombreChange = (event) => {
9     setNombre(event.target.value);
10  };
11
12  const handleNumeroCasaChange = (event) => {
13    setNumeroCasa(event.target.value);
14  };
15
16  const handleVotoChange = (event) => {
17    setVoto(event.target.value);
18  };
19
20  const handleSubmit = (event) => {
21    event.preventDefault();
22    // enviar los datos del formulario
23    console.log('Nombre:', nombre);
24    console.log('Número de Casa:', numeroCasa);
25    console.log('Voto:', voto);
26    // limpiar los campos del formulario
27    setNombre('');
28    setNumeroCasa('');
29    setVoto('');
30  };
31
32  return (
33    <div>
34      <h2>Formulario de Votación</h2>
35      <form onSubmit={handleSubmit}>
36        <div>
37          <label>
```

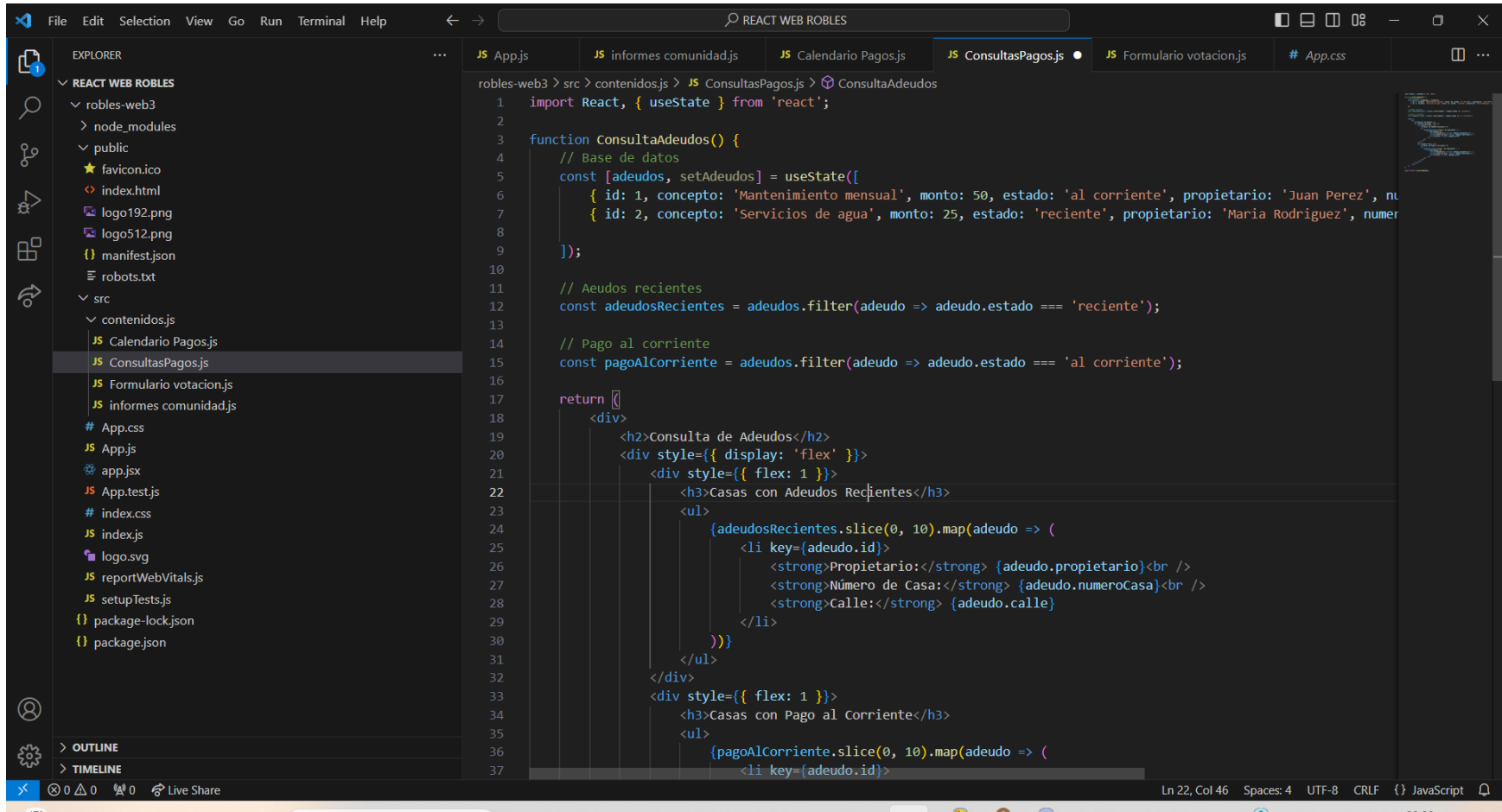
Nota: se realizo un formulario donde te pregunta nombre, numero de casa y un espacio de voto donde se espera en futuro agregar más detalles. (Realice la estructura)

Calendario de Fechas Límites de Pagos (Alias: 4UH-CP)

```
1 import React from 'react';
2
3 function CalendarPagos() {
4
5   const fechasPagos = [
6     { id: 1, fecha: '2024-02-10', concepto: 'Pago mensual de mantenimiento' },
7     { id: 2, fecha: '2024-03-10', concepto: 'Pago mensual de mantenimiento' },
8     { id: 3, fecha: '2024-04-10', concepto: 'Pago mensual de mantenimiento' },
9   ];
10
11   return (
12     <div>
13       <h2>Calendario de Fechas Límites de Pagos - Comunidad Robles</h2>
14       <table>
15         <thead>
16           <tr>
17             <th>Fecha Límite</th>
18             <th>Concepto</th>
19           </tr>
20         </thead>
21         <tbody>
22           {fechasPagos.map(fechaPago => (
23             <tr key={fechaPago.id}>
24               <td>{fechaPago.fecha}</td>
25               <td>{fechaPago.concepto}</td>
26             </tr>
27           ))}
28         </tbody>
29       </table>
30     </div>
31   );
32
33   export default CalendarPagos;
34
35
36
```

Nota: Agregue el calendario como si fuera una tabla, agregue 3 fechas limite como ejemplo sin embargo deje el espacio para subir toda la información base de datos.

Consulta de Adeudos (Alias: 05UH-CA)

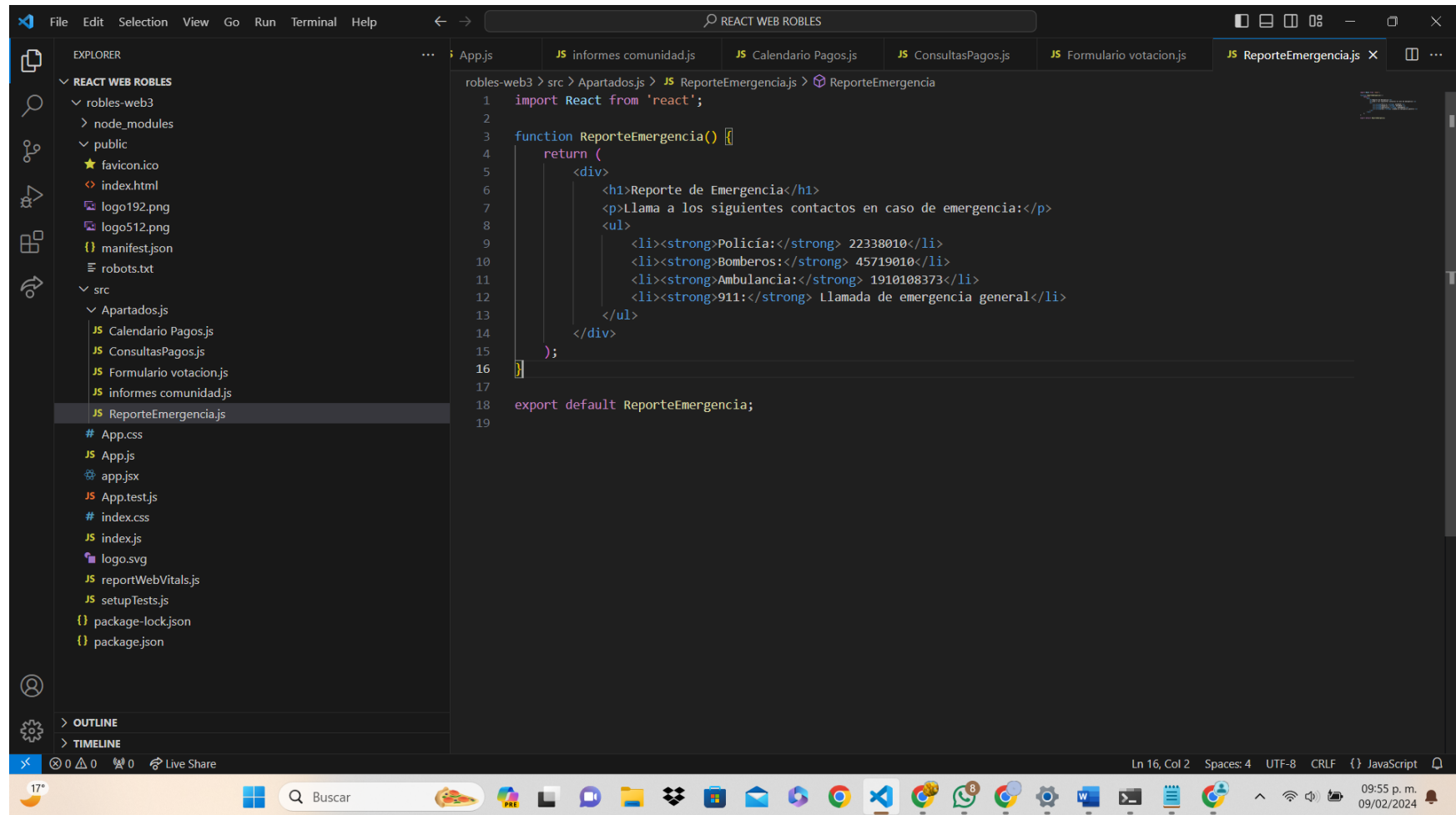


```
1 import React, { useState } from 'react';
2
3 function ConsultaAdeudos() {
4   // Base de datos
5   const [adeudos, setAdeudos] = useState([
6     { id: 1, concepto: 'Mantenimiento mensual', monto: 50, estado: 'al corriente', propietario: 'Juan Perez', nu
7     { id: 2, concepto: 'Servicios de agua', monto: 25, estado: 'reciente', propietario: 'Maria Rodriguez', numer
8   ]);
9
10
11   // Adeudos recientes
12   const adeudosRecientes = adeudos.filter(adeudo => adeudo.estado === 'reciente');
13
14   // Pago al corriente
15   const pagoAlCorriente = adeudos.filter(adeudo => adeudo.estado === 'al corriente');
16
17   return (
18     <div>
19       <h2>Consulta de Adeudos</h2>
20       <div style={{ display: 'flex' }}>
21         <div style={{ flex: 1 }}>
22           <h3>Casas con Adeudos Recientes</h3>
23           <ul>
24             {adeudosRecientes.slice(0, 10).map(adeudo => (
25               <li key={adeudo.id}>
26                 <strong>Propietario:</strong> {adeudo.propietario}<br />
27                 <strong>Número de Casa:</strong> {adeudo.numeroCasa}<br />
28                 <strong>Calle:</strong> {adeudo.calle}
29               </li>
30             ))}
31           </ul>
32         </div>
33         <div style={{ flex: 1 }}>
34           <h3>Casas con Pago al Corriente</h3>
35           <ul>
36             {pagoAlCorriente.slice(0, 10).map(adeudo => (
37               <li key={adeudo.id}>
```

```
15 const pagoAlCorriente = adeudos.filter(adeudo => adeudo.estado === 'al corriente');
16
17 return (
18   <div>
19     <h2>Consulta de Adeudos</h2>
20     <div style={{ display: 'flex' }}>
21       <div style={{ flex: 1 }}>
22         <h3>Casas con Adeudos Recientes</h3>
23         <ul>
24           {adeudosRecientes.slice(0, 10).map(adeudo => (
25             <li key={adeudo.id}>
26               <strong>Propietario:</strong> {adeudo.propietario}<br />
27               <strong>Número de Casa:</strong> {adeudo.numeroCasa}<br />
28               <strong>Calle:</strong> {adeudo.calle}
29             </li>
30           ))}
31         </ul>
32       </div>
33       <div style={{ flex: 1 }}>
34         <h3>Casas con Pago al Corriente</h3>
35         <ul>
36           {pagoAlCorriente.slice(0, 10).map(adeudo => (
37             <li key={adeudo.id}>
38               <strong>Propietario:</strong> {adeudo.propietario}<br />
39               <strong>Número de Casa:</strong> {adeudo.numeroCasa}<br />
40               <strong>Calle:</strong> {adeudo.calle}
41             </li>
42           ))}
43         </ul>
44       </div>
45     </div>
46   </div>
47 );
48
49
50 export default ConsultaAdeudos;
51
```

Nota: Se dejo el espacio para la base de datos, con 10 espacios en cada columna

Reporte de Emergencia (Alias: 06UH-MP)

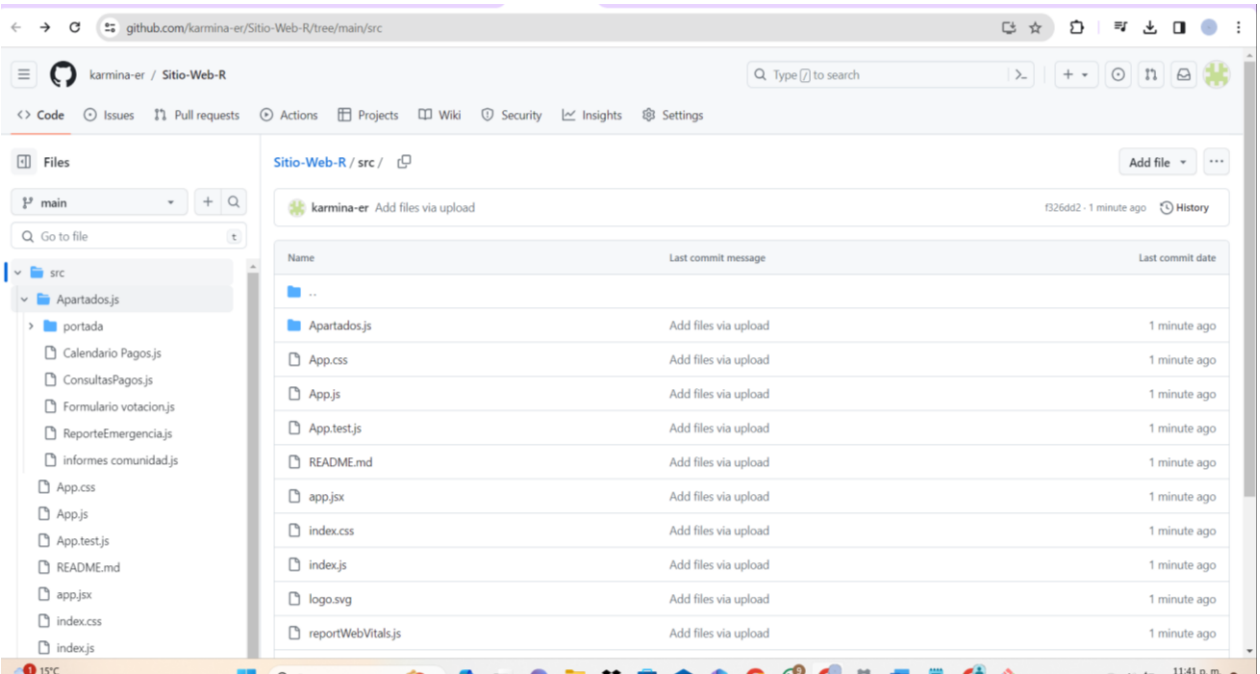


```
1 import React from 'react';
2
3 function ReporteEmergencia() {
4   return (
5     <div>
6       <h1>Reporte de Emergencia</h1>
7       <p>Llama a los siguientes contactos en caso de emergencia:</p>
8       <ul>
9         <li><strong>Policía:</strong> 22338010</li>
10        <li><strong>Bomberos:</strong> 45719010</li>
11        <li><strong>Ambulancia:</strong> 1910108373</li>
12        <li><strong>911:</strong> Llamada de emergencia general</li>
13      </ul>
14    </div>
15  );
16
17
18 export default ReporteEmergencia;
```

Nota: Agregue contactos de emergencia importantes

ACCESO A GITHUB

<https://github.com/karina-er/Sitio-Web-R/tree/f326dd27d33b31da884147cb3cac2e3339438130/src>



Name	Last commit message	Last commit date
..		
portada	Add files via upload	3 minutes ago
Calendario Pagos.js	Add files via upload	3 minutes ago
ConsultasPagos.js	Add files via upload	3 minutes ago
Formulario votacion.js	Add files via upload	3 minutes ago
ReporteEmergencia.js	Add files via upload	3 minutes ago
informes comunidad.js	Add files via upload	3 minutes ago

REFERENCIAS:

- Primeros pasos en React - Aprende desarrollo web | MDN. (s. f.). MDN Web Docs.
https://developer.mozilla.org/es/docs/Learn/Tools_and_testing/Client-side_JavaScript_frameworks/React_getting_started
- García N., Ernesto. (18 de 02 de 2019). ¡Construye tu primera aplicación web con ReactJS! Recuperado de
<https://medium.com/@ernestognw/construye-tu-primera-aplicaci%C3%B3n-web-con-reactjs-parte-1-4c8bcca3b9db>

Luque, Andres. (15 may. 2016). Ejercicio Product Backlog y Sprint Backlog. <https://youtu.be/ySJMHCi-8Uo>