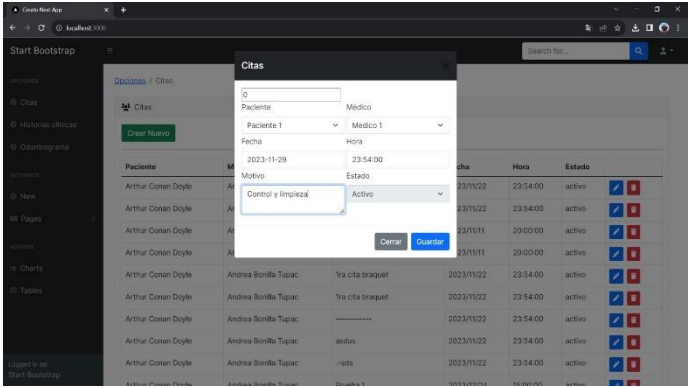
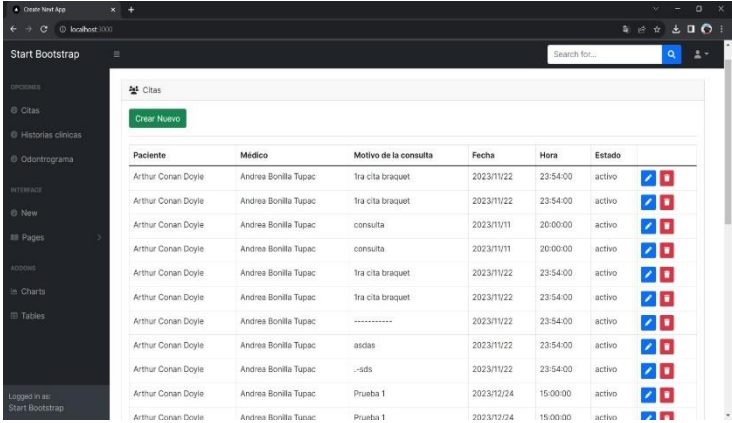


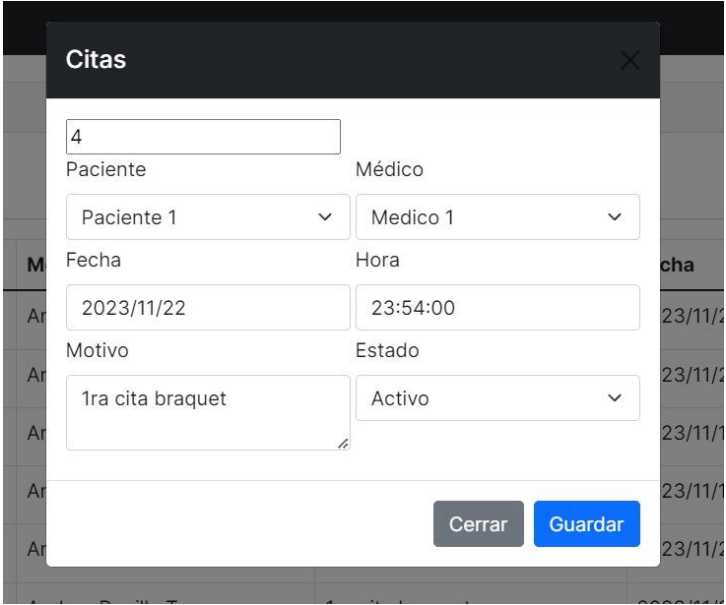
PRUEBAS DE SISTEMA PARA EL SISTEMA ODONTOLÓGICO

Caso de prueba de sistema – Prueba funcional: Registrar citas de los pacientes	
Descripción:	Prueba funcional para la creación de una nueva cita de paciente en el sistema web odontológico.
Precondiciones:	<ul style="list-style-type: none">• Métodos de sistema web desplegados• El tester debe tener acceso al sistema web• Datos de los pacientes disponibles para asociar a las citas• Entorno de prueba configurado y herramienta de prueba operativa
Pasos:	<ul style="list-style-type: none">• Iniciar entorno de desarrollo de Visual Studio Code• Identificar el método para registro de citas• Cambiar los elementos del arreglo de datos, esto implica ingresar los datos del paciente.• Ejecutar la función para mostrar el estado de registro exitoso.• Ingreso de campos: <pre>}, // CREAR CITA async fetchResultCrear(data) { try { const response = await fetch('https://desplieguebackend', method: 'POST', headers: { 'Content-Type': 'application/json' }, body: JSON.stringify({ "IDHistoria": data.IDHistoria, "IDMedico": data.IDMedico, "citFecha": "2023-11-22", //YYYY-MM-DD "citHora": data.citHora, //23:54:00 "citMotivo": data.citMotivo, "citEstado": data.citEstado, })); const jsonResult = await response.json();</pre>

	
Resultados Esperados:	<ul style="list-style-type: none"> ● Mensaje de confirmación de registro.
Criterios de Validación:	<ul style="list-style-type: none"> ● Cambios registrados en la base de datos ● Confirmar que el sistema realice validaciones de datos, asegurándose de que la información ingresada sea coherente y cumpla con los requisitos definidos, como formatos de fechas válidos y datos obligatorios. ● Validar que las citas registradas estén correctamente asociadas con los pacientes correspondientes en la base de datos.
Resultados:	Registro exitoso - pendiente

Caso de prueba de sistema – Prueba funcional: Visualizar un listado de todas las citas de los pacientes.	
Descripción:	Prueba funcional para visualizar un listado de todas las citas de los pacientes. En el sistema web odontológico.
Precondiciones:	<ul style="list-style-type: none"> ● Métodos de sistema web desplegados ● El tester debe tener acceso al sistema web ● Número de registros considerables en la parte de registrar citas de los pacientes ● Entorno de prueba configurado y herramienta de prueba operativa

Pasos:	<ul style="list-style-type: none">● Iniciar entorno de desarrollo de Visual Studio Code● Identificar el método para visualizar un listado de todas las citas de los pacientes● Cambiar los datos en los métodos que trae los datos de registro de citas de pacientes● Ejecutar la función para mostrar la lista de citas de pacientes registrados:
Resultados Esperados:	<ul style="list-style-type: none">● Lista de registro de citas de pacientes actualizado● Número de citas que se muestran igual al número de registros hechos.
Criterios de Validación:	<ul style="list-style-type: none">● Verificar que el sistema muestre de manera precisa y completa el listado de citas de pacientes, incluyendo información relevante como la fecha, hora, nombre del paciente y detalles de la cita● Validar que se pueda acceder a detalles adicionales de cada cita desde el listado, permitiendo una vista más detallada de la información asociada a cada cita.● Validar que las citas registradas estén correctamente asociadas con los pacientes correspondientes en la base de datos.● Asegurar que el listado indique claramente el estado de cada cita, como pendiente, confirmada o cancelada, para facilitar la gestión y comprensión de la información.
Resultados:	

Caso de prueba de sistema – Prueba funcional: Editar citas de los pacientes.	
Descripción:	Prueba funcional para editar citas de los pacientes. En el sistema web odontológico.
Precondiciones:	<ul style="list-style-type: none">• Métodos de sistema web desplegados• El tester debe tener acceso al sistema web• Número de registros considerables en la parte de registrar citas de los pacientes• Entorno de prueba configurado y herramienta de prueba operativa
Pasos:	<ul style="list-style-type: none">• Iniciar entorno de desarrollo de Visual Studio Code• Identificar el método para editar citas de los pacientes• Cambiar los datos en los métodos que modifica los datos de las citas de los pacientes• Ejecutar la función para editar las citas de los pacientes: <div data-bbox="644 1075 1372 1677"></div>
Resultados Esperados:	<ul style="list-style-type: none">• Cita de paciente editada según lo requerido• Cambio reflejado en la base de datos
Criterios de	<ul style="list-style-type: none">• Verificar que el sistema permita editar citas de pacientes de manera exitosa, asegurándose de que los cambios realizados se

Validación:	<p>reflejen correctamente en la base de datos.</p> <ul style="list-style-type: none"> ● Confirmar que el sistema realice validaciones adecuadas para los datos editados, asegurando que la información modificada sea coherente y cumpla con los requisitos definidos.
Resultados:	Notificación de éxito.- pendiente

Caso de prueba de sistema – Prueba funcional: Registrar historias clínicas.	
Descripción:	Prueba funcional para registrar historia clínica de los pacientes. En el sistema web odontológico.
Precondiciones:	<ul style="list-style-type: none"> ● Métodos de sistema web desplegados ● El tester debe tener acceso al sistema web ● Datos disponibles del historial del paciente para el registro en el sistema web. ● Entorno de prueba configurado y herramienta de prueba operativa
Pasos:	<ul style="list-style-type: none"> ● Iniciar entorno de desarrollo de Visual Studio Code ● Identificar el método para registrar historias clínicas de los pacientes ● Cambiar los datos en los métodos que registra los datos de las historias clínicas de los pacientes ● Ejecutar la función para registrar las historias clínicas de los pacientes:

	<pre>// CREAR Paciente async fetchResultCrear(data) { try { const response = await fetch('https://desplieguebackend-p method: 'POST', headers: { 'Content-Type': 'application/json' }, body: JSON.stringify({ "NOMBRES": "María", "APELLIDOS": "Paredes", "DNI": "12345678", "SEXO": "FEMENINO", "RELIGION": "BUDISTA", "RAZA": "TRIGUEÑA", "FECHA": "1990-01-01", "LUGAR": "HUANCAYO", "RESIDENCIA": "PSJE. JUNIN 256", "GRADO": "ESTUDIANTE", "OCUPACION": "ESTUDIANTE", "CIVIL": "SOLTERO", "CORREO": "mparedes@gmail.com", "TELEFONO": "963857412", "ESTADO": "1", "ECTOSCOPIA": "JOVEN DELGADO" }) } });</pre>
Resultados Esperados:	<ul style="list-style-type: none"> • Mensaje de registro exitoso • Cambio del registro reflejado en la base de datos
Criterios de Validación:	<ul style="list-style-type: none"> • Verificar que el sistema permita el registro exitoso de nuevas historias clínicas, asegurándose de que los datos ingresados se almacenen correctamente en la base de datos. • Verificar que el sistema permita el registro completo de datos clínicos relevantes, incluyendo antecedentes médicos, tratamientos previos y cualquier información relevante para el diagnóstico y tratamiento.

Resultados:	<p>Datos de la API:</p> <pre> { "result": { "message": "", "affectedRows": 1, "row": { "IDPaciente": 7, "pacNombre": "María", "pacApellido": "Paredes", "pacDNI": "12345678", "pacSexo": "FEMENINO", "pacReligion": "BUDISTA", "pacRaza": "TRIGUEÑA", "pacFechaN": "01/01/1990", "pacEdad": "33", "pacLugar": "HUANCAYO", "pacResidencia": "PSJE. JUNIN 256", "pacGradoI": "ESTUDIANTE", "pacOcupacion": "ESTUDIANTE", "pacEstadoC": "SOLTERO", "pacCorreo": "mparedes@gmail.com", "pacTelef": "963857412", "pacEstado": "1" } } } </pre>
--------------------	--

Caso de prueba de sistema – Prueba funcional: Visualizar historias clínicas	
Descripción:	Prueba funcional para visualizar historias clinicas de los pacientes. En el sistema web odontológico.
Precondiciones:	<ul style="list-style-type: none"> • Métodos de sistema web desplegados • El tester debe tener acceso al sistema web • Número de registros considerables en la parte de registrar historias clínicas de los pacientes • Entorno de prueba configurado y herramienta de prueba operativa
Pasos:	<ul style="list-style-type: none"> • Iniciar entorno de desarrollo de Visual Studio Code • Identificar el método para visualizar un listado de todas las historias clínicas de los pacientes • Cambiar los datos en los métodos que trae los datos de registro de historias clínicas de los pacientes

	<ul style="list-style-type: none"> Ejecutar la función para mostrar la lista de historias clínicas de pacientes registrados:
Resultados Esperados:	<ul style="list-style-type: none"> Lista de registro de historias clínicas de pacientes actualizado Número de historias clínicas que se muestran igual al número de registros hechos.
Criterios de Validación:	<ul style="list-style-type: none"> Verificar que el sistema permita el registro exitoso de nuevas historias clínicas, asegurándose de que los datos ingresados se almacenen correctamente en la base de datos. Confirmar que el sistema realice validaciones adecuadas para los datos de las historias clínicas, asegurando que la información ingresada sea coherente y cumpla con los requisitos definidos. Validar que las historias clínicas registradas estén correctamente asociadas con los pacientes correspondientes en la base de datos, evitando asociaciones incorrectas.
Resultados:	<p>Datos de la API:</p> <pre>[{ "IDPaciente": 1, "pacNombre": "Arthur", "pacApellido": "Conan Doyle", "pacDNI": "76905898", "pacSexo": "Masculino", "pacReligion": "Cristiana", "pacRaza": "Blanca", "pacFechaN": "2022/01/01", "pacEdad": "1", "pacLugar": "Ciudad Ejemplo", "pacResidencia": "Dirección de Residencia", "pacGradoI": "Universitario", "pacOcupacion": "Profesión u Ocupación", "pacEstadoC": "Soltero", "pacCorreo": "correo@example.com", "pacTelef": "123456789", "pacEstado": "2" }, { "IDPaciente": 3, "pacNombre": "Nombre Ejemplo", "pacApellido": "Apellido Ejemplo", "pacDNI": "72113166", "pacSexo": "Masculino",</pre>
Caso de prueba de sistema – Prueba funcional: Editar historias clínicas	
Descripción:	Prueba funcional para editar citas de los pacientes. En el sistema web

	odontológico.
Precondiciones:	<ul style="list-style-type: none"> • Métodos de sistema web desplegados • El tester debe tener acceso al sistema web • Número de registros considerables en la parte de registrar historias clínicas de los pacientes • Entorno de prueba configurado y herramienta de prueba operativa
Pasos:	<ul style="list-style-type: none"> • Iniciar entorno de desarrollo de Visual Studio Code • Identificar el método para editar historia clinica de los pacientes • Cambiar los datos en los métodos que modifica los datos de las historias clínicas de los pacientes • Ejecutar la función para editar las citas de los pacientes: <pre> //ACTUALIZAR Paciente async fetchResultActualizar(data) { debugger try { const response = await fetch(`https://desplieguebacke method: 'PATCH', headers: { 'Content-Type': 'application/json' }, body: JSON.stringify({ "NOMBRES": "PRUEBA 1", "APELLIDOS": "PRUEBA 1", "DNI": "78963215", "SEXO": "FEMENINO", "RELIGION": "ATEO", "RAZA": "MESTIZO", "FECHA": "2000-01-01", "LUGAR": "JAUJA", "RESIDENCIA": "CALLE NUEVA", "GRADO": "DOCTOR", "OCUPACION": "FILOSOFO", "CIVIL": "SOLTERO", "CORREO": "PRUEBA1@GMAIL.COM", "TELEFONO": "963214587", "ESTADO": "1", "ECTOSCOPIA": "JOVEN DELGADO" }) } } </pre>
Resultados Esperados:	<ul style="list-style-type: none"> • Historia clínica de paciente editada según lo requerido • Cambio reflejado en la base de datos
Criterios de Validación:	<ul style="list-style-type: none"> • Verificar que el sistema permita la visualización exitosa de historias clínicas, asegurándose de que la información se muestre de manera clara y completa. • Validar que se muestren de manera clara y comprensible los

	datos clínicos relevantes, como antecedentes médicos, tratamientos previos, resultados de exámenes y cualquier otra información crítica.
Resultados:	<div>Obtener datos</div> <h2>Datos de la API:</h2> <pre>[{ "IDPaciente": 1, "pacNombre": "PRUEBA 1", "pacApellido": "PRUEBA 1", "pacDNI": "78963215", "pacSexo": "FEMENIMO", "pacReligion": "ATEO", "pacRaza": "MESTIZO", "pacFechaN": "2000/01/01", "pacEdad": "23", "pacLugar": "JAUJA", "pacResidencia": "CALLE NUEVA", "pacGradoI": "DOCTOR", "pacOcupacion": "FILOSOSO", "pacEstadoC": "SOLTERO", "pacCorreo": "PRUEBA1@GMAIL.COM", "pacTelef": "963214587", "pacEstado": "1" },]</pre>

Caso de prueba de sistema – Prueba funcional: Registrar nuevo odontograma.	
Descripción:	Prueba funcional para registrar un nuevo odontograma. En el sistema web odontológico.
Precondiciones:	<ul style="list-style-type: none"> • Métodos de sistema web desplegados • El tester debe tener acceso al sistema web • Tener los datos de ingreso para el odontograma, después de realizar el diagnostico médico • Entorno de prueba configurado y herramienta de prueba operativa

Pasos:	<ul style="list-style-type: none"> • Iniciar entorno de desarrollo de Visual Studio Code • Identificar el método para un nuevo odontograma • Cambiar los datos en los métodos que registra los datos del odontograma • Ejecutar la función para registrar odontograma:
Resultados Esperados:	<ul style="list-style-type: none"> • Mensaje de registro exitoso • Cambio del registro reflejado en la base de datos
Criterios de Validación:	<ul style="list-style-type: none"> • Verificar que el sistema permita el registro exitoso de nuevas odontogramas, asegurándose de que los datos ingresados se almacenen correctamente en la base de datos. • Verificar que el sistema permita el registro completo de datos dentales relevantes en la odontograma, incluyendo detalles sobre dientes, encías y cualquier otra información clínica necesaria.
Resultados:	Odontograma generado en el sistema visualmente - pendiente

Caso de prueba de sistema – Prueba funcional: Editar odontograma.	
Descripción:	Prueba funcional para editar un odontograma. En el sistema web odontológico.
Precondiciones:	<ul style="list-style-type: none"> • Métodos de sistema web desplegados • El tester debe tener acceso al sistema web • Número de registros considerables en la parte de registro de odontogramas. • Entorno de prueba configurado y herramienta de prueba operativa
Pasos:	<ul style="list-style-type: none"> • Iniciar entorno de desarrollo de Visual Studio Code • Identificar el método para editar odontograma. • Cambiar los datos en los métodos que modifica los datos de

	<p>odontograma.</p> <ul style="list-style-type: none"> • Ejecutar la función para editar las citas de los pacientes:
Resultados Esperados:	<ul style="list-style-type: none"> • Odontograma editado según lo requerido • Cambio reflejado en la base de datos
Criterios de Validación:	<ul style="list-style-type: none"> • Verificar que el sistema permita la edición exitosa de odontogramas, asegurándose de que los cambios realizados se reflejen correctamente en la base de datos. • Validar que los odontogramas editados estén correctamente asociados con los pacientes correspondientes en la base de datos, evitando asociaciones incorrectas.
Resultados:	Odontograma editado y resultados reflejados en la base de datos - pendiente

Caso de prueba de sistema – Prueba funcional: Visualizar odontogramas	
Descripción:	Prueba funcional para visualizar odontogramas por pacientes. En el sistema web odontológico.
Precondiciones:	<ul style="list-style-type: none"> • Métodos de sistema web desplegados • El tester debe tener acceso al sistema web • Registro de odontogramas existentes en el sistema web. • Entorno de prueba configurado y herramienta de prueba operativa
Pasos:	<ul style="list-style-type: none"> • Iniciar entorno de desarrollo de Visual Studio Code • Identificar el método que permita generar la vista de odontograma • Cambiar los datos en los métodos que trae los datos de registro de odontograma • Ejecutar la función para mostrar la lista de historias clínicas de pacientes registrados: <p>Entrada:</p> <pre>// LISTAR odontograma async fetchResultListar() { console.log("odontograma") try { const response = await fetch('https://desplieguebackend-production.up.railway.app/odontograma/listodontograma/76905898', { method: 'GET' }); const jsonResult = await response.json(); return jsonResult; } catch (error) { console.error('Error al obtener respuesta de la API:', error); throw error; } };</pre>
Resultados Esperados:	<ul style="list-style-type: none"> • Información de odontograma por paciente.
Criterios de Validación:	<ul style="list-style-type: none"> • Verificar que el sistema permita la visualización exitosa de odontogramas asociados a un paciente específico, asegurándose de que la información se muestre de manera clara y completa. • Validar que se muestren de manera clara y comprensible los datos dentales relevantes, incluyendo detalles sobre dientes, encías y cualquier otra información clínica necesaria. • Validar que las historias clínicas registradas estén correctamente

	asociadas con los pacientes correspondientes en la base de datos, evitando asociaciones incorrectas.
Resultados:	<div>Obtener datos</div> <h2>Datos de la API:</h2> <pre>[{ "IDOdontograma": 4, "IDHistoria": 1, "odoFase": "EVOLUCIÓN 2", "odoEspecificacion": "Especificacion", "odoObservacion": "observacion", "odoFecha": "2023/11/25 13:19" }, { "IDOdontograma": 5, "IDHistoria": 1, "odoFase": "Prueba", "odoEspecificacion": "Prueba", "odoObservacion": "Prueba", "odoFecha": "2023/11/25 20:34" }, { "IDOdontograma": 6, "IDHistoria": 1, "odoFase": "Prueba", "odoEspecificacion": "Prueba", "odoObservacion": "Prueba", "odoFecha": "2023/11/26 04:02" },]</pre>

Caso de prueba de sistema – Prueba funcional: Registrar nuevo paciente.	
Descripción:	Prueba funcional para registrar un nuevo paciente. En el sistema web odontológico.
Precondiciones:	<ul style="list-style-type: none"> • Métodos de sistema web desplegados • El tester debe tener acceso al sistema web • Tener los datos del paciente para considerarlo en la entrada al sistema web • Entorno de prueba configurado y herramienta de prueba operativa

Pasos:	<ul style="list-style-type: none"> • Iniciar entorno de desarrollo de Visual Studio Code • Identificar el método para registrar un nuevo paciente • Cambiar los datos en los métodos que registra los datos de un nuevo paciente • Ejecutar la función para registrar nuevo paciente: <pre>// CREAR Paciente async fetchResultCrear(data) { try { const response = await fetch('https://desplieguebackend-p method: 'POST', headers: { 'Content-Type': 'application/json' }, body: JSON.stringify({ "NOMBRES": "María", "APELLIDOS": "Paredes", "DNI": "12345678", "SEXO": "FEMENINO", "RELIGION": "BUDISTA", "RAZA": "TRIGUEÑA", "FECHA": "1990-01-01", "LUGAR": "HUANCAYO", "RESIDENCIA": "PSJE. JUNIN 256", "GRADO": "ESTUDIANTE", "OCUPACION": "ESTUDIANTE", "CIVIL": "SOLTERO", "CORREO": "mparedes@gmail.com", "TELEFONO": "963857412", "ESTADO": "1", "ECTOSCOPIA": "JOVEN DELGADO" }); } }</pre>
Resultados Esperados:	<ul style="list-style-type: none"> • Mensaje de registro exitoso • Cambio del registro reflejado en la base de datos
Criterios de Validación:	<ul style="list-style-type: none"> • Verificar que el sistema permita la visualización exitosa de odontogramas asociados a un paciente específico, asegurándose de que la información se muestre de manera clara y completa. • Validar que se muestren de manera clara y comprensible los datos dentales relevantes, incluyendo detalles sobre dientes, encías y cualquier otra información clínica necesaria.
Resultados:	Cambios guardados en la base de datos - pendiente

Caso de prueba de sistema – Prueba funcional: Editar datos del paciente.	
Descripción:	Prueba funcional para editar los datos del paciente por estado. En el

	sistema web odontológico.
Precondiciones:	<ul style="list-style-type: none"> • Métodos de sistema web desplegados • El tester debe tener acceso al sistema web • Número de registros considerables en la parte de registro de pacientes. • Entorno de prueba configurado y herramienta de prueba operativa
Pasos:	<ul style="list-style-type: none"> • Iniciar entorno de desarrollo de Visual Studio Code • Identificar el método para editar paciente. • Cambiar los datos en los métodos que modifica los datos del paciente. • Ejecutar la función para editar las citas de los pacientes: <pre> //ACTUALIZAR Paciente async fetchResultActualizar(data) { debugger try { const response = await fetch(`https://desplieguebacke method: 'PATCH', headers: { 'Content-Type': 'application/json' }, body: JSON.stringify({ "NOMBRES": "PRUEBA 1", "APELLIDOS": "PRUEBA 1", "DNI": "78963215", "SEXO": "FEMENINO", "RELIGION": "ATEO", "RAZA": "MESTIZO", "FECHA": "2000-01-01", "LUGAR": "JAUJA", "RESIDENCIA": "CALLE NUEVA", "GRADO": "DOCTOR", "OCUPACION": "FILOSOFO", "CIVIL": "SOLTERO", "CORREO": "PRUEBA1@GMAIL.COM", "TELEFONO": "963214587", "ESTADO": "1", "ECTOSCOPIA": "JOVEN DELGADO" }) }) }); </pre>
Resultados Esperados:	<ul style="list-style-type: none"> • Datos del paciente actualizado según lo requerido • Cambio reflejado en la base de datos
Criterios de Validación:	<ul style="list-style-type: none"> • Verificar que el sistema permita la edición exitosa de datos del paciente, asegurándose de que los cambios realizados se reflejen correctamente en la base de datos. • Verificar que cualquier cambio realizado en la edición de datos del paciente se refleje en tiempo real en otras áreas del sistema que muestren información actualizada.

Resultados:	Datos de paciente actualizado y evidente en la base de datos - pendiente.
--------------------	---

Caso de prueba de sistema – Prueba funcional: Visualizar pacientes por estado	
Descripción:	Prueba funcional para visualizar datos de pacientes por estado. En el sistema web odontológico.
Precondiciones:	<ul style="list-style-type: none"> • Métodos de sistema web desplegados • El tester debe tener acceso al sistema web • Registro de pacientes existentes en el sistema web. • Entorno de prueba configurado y herramienta de prueba operativa
Pasos:	<ul style="list-style-type: none"> • Iniciar entorno de desarrollo de Visual Studio Code • Identificar el método que permita generar la vista de pacientes con su estado registrados • Cambiar los datos en los métodos que permite visualizar los datos de los pacientes por estado • Ejecutar la función para mostrar la lista de pacientes registrados por estado: <p>Entrada 1:</p> <pre>// LISTAR Paciente async fetchResultlistar() { console.log("Paciente") try { //1 activo 2 inactivo const response = await fetch('https://desplieguebackend-production.up.railway.app/paciente/listPaciente/2', { method: 'GET' }); const jsonResponse = await response.json(); return jsonResponse; } catch (error) { console.error('Error al obtener respuesta de la API:', error); throw error; } };</pre> <p>-con estado 2</p> <p>Entrada 2:</p>

	<pre>// LISTAR Paciente async fetchResultListar() { console.log("Paciente") try { //1 activo 2 inactivo const response = await fetch('https://desplieguebackend-production.up.railway.app/paciente/listPaciente/1', { method: 'GET' }); const jsonResponse = await response.json(); return jsonResponse; } catch (error) { console.error('Error al obtener respuesta de la API:', error); throw error; } };</pre> <p>-con estado 1</p>
Resultados Esperados:	<ul style="list-style-type: none">● Información de odontograma por paciente.
Criterios de Validación:	<ul style="list-style-type: none">● Verificar que el sistema permita la edición exitosa de datos del paciente, asegurándose de que los cambios realizados se reflejen correctamente en la base de datos.● Confirmar que el sistema realice validaciones adecuadas para los datos editados, asegurando que la información modificada sea coherente y cumpla con los requisitos definidos.
Resultados:	<p>Resultado 1:</p> <div>Obtener datos</div> <p>Datos de la API:</p> <pre>[{ "IDPaciente": 3, "pacNombre": "Nombre Ejemplo", "pacApellido": "Apellido Ejemplo", "pacDNI": "72113166", "pacSexo": "Masculino", "pacReligion": "Cristiana", "pacRaza": "Blanca", "pacFechaN": "2000/01/01", "pacEdad": "26", "pacLugar": "Ciudad Ejemplo", "pacResidencia": "Dirección de Residencia", "pacGradoI": "Universitario", "pacOcupacion": "Profesión u Ocupación", "pacEstadoC": "Soltero", "pacCorreo": "correo@example.com", "pacTelef": "123456789", "pacEstado": "2" }, { "IDPaciente": 4, "pacNombre": "Nombre Ejemplo", "pacApellido": "Apellido Ejemplo", "pacDNI": "12345679", "pacSexo": "Masculino".</pre>

Resultado 2:

Datos de la API:

```
[
  {
    "IDPaciente": 1,
    "pacNombre": "PRUEBA 1",
    "pacApellido": "PRUEBA 1",
    "pacDNI": "76905898",
    "pacSexo": "FEMENINO",
    "pacReligion": "ATEO",
    "pacRaza": "MESTIZO",
    "pacFechaN": "2000/01/01",
    "pacEdad": "23",
    "pacLugar": "JAUJA",
    "pacResidencia": "CALLE NUEVA",
    "pacGradoI": "DOCTOR",
    "pacOcupacion": "FILOSOFO",
    "pacEstadoC": "SOLTERO",
    "pacCorreo": "PRUEBA1@GMAIL.COM",
    "pacTelef": "963214587",
    "pacEstado": "1"
  },
  {
    "IDPaciente": 2,
    "pacNombre": "ANA",
    "pacApellido": "RAMIREZ PEREZ",
    "pacDNI": "71458895",
    "pacSexo": "masculino"
```