



Reporte Técnico de Actividades Práctico-Experimentales Nro. 001

1. Datos de Identificación del Estudiante y la Práctica

Nombre del estudiante(s)	Manuel Santiago Tamayo
Asignatura	Desarrollo Basado en Plataformas
Ciclo	5 A
Unidad	1
Resultado de aprendizaje de la unidad	
Práctica Nro.	001
Título de la Práctica	Implementar un servicio REST con Node.js
Nombre del Docente	Edison Leonardo Coronel Romero
Fecha	Viernes 3 de octubre
Horario	07h30 – 10h30
Lugar	Laboratorio Computación aplicada Laboratorio Desarrollo de Software Laboratorio de redes y Sistemas Operativos Laboratorio Virtual EVA Aula
Tiempo planificado en el Sílabo	3 horas

2. Objetivo(s) de la Práctica

- Diseñar y construir un prototipo sencillo de interfaz que represente al menos un flujo completo del proyecto (8 pantallas), utilizando Stitch como herramienta de apoyo al diseño y HTML/CSS para la implementación base.
- Aplicar principios de diseño responsive, maquetación semántica y criterios básicos de accesibilidad y usabilidad.
- Documentar el prototipo mediante capturas de pantalla y una descripción técnica en el repositorio del equipo.

3. Materiales, Reactivos, Equipos y Herramientas

Computador con acceso a Internet.

- Navegador web actualizado.
- Editor de código (VS Code o similar).
- Herramienta de prototipado Stitch (o entorno definido por la asignatura).
- Repositorio del proyecto en GitHub / GitLab.
- Lineamientos de UI/UX del proyecto (si ya existen).



4. Procedimiento / Metodología Ejecutada

Inicio

1. El docente recuerda brevemente los conceptos de:
 - o diseño responsive,
 - o maquetación semántica,
 - o accesibilidad básica (contraste, tamaño de tipografía, estructura).
2. Explica el alcance de la práctica:
 - o Construir **8 pantallas** relacionadas con un flujo del proyecto (por ejemplo: inicio, login, listado, detalle, formulario, resumen, perfil, ayuda).

Desarrollo

Paso 1: Definición rápida del flujo (en equipo)

- Elegir un flujo funcional del proyecto (por ejemplo: “registro y gestión de usuario”, “gestión de equipos”, “registro de resultados”, etc.).
- Definir las **8 pantallas** que formarán parte del flujo y sus relaciones (boceto rápido en pizarra o en Stitch).

Paso 2: Prototipado en Stitch

- Crear en Stitch las 8 pantallas con:
 - o distribución básica de elementos (header, menú, contenido, footer), ubicaciones aproximadas de botones, formularios, tablas/listas, o jerarquía visual (títulos, subtítulos, texto).
- Ajustar la navegación entre pantallas (enlaces o interacciones de Stitch).

Paso 3: Implementación en HTML/CSS

- Crear una carpeta en el repositorio, por ejemplo:
`frontend/prototipos/ui-stitch/`
- Implementar las 8 pantallas en HTML5/CSS3 respetando:
 - o estructura semántica (`<header>`, `<nav>`, `<main>`, `<section>`, `<footer>`, etc.),
 - o diseño responsive básico (uso de flex, grid o media queries), o legibilidad (tipografías, espaciados, tamaños de fuente).



Paso 4: Pruebas y capturas

- Verificar cada pantalla en al menos **dos tamaños de ventana** (ej. escritorio y móvil).
- Tomar **capturas de pantalla** de cada una de las 8 pantallas en su estado principal.

Paso 5: Documentación en el repositorio

En la carpeta `/docs/` o similar, crear un archivo, por ejemplo: `docs/ui_prototipo_stitch.md` con:

- Breve descripción del flujo que cubren las 8 pantallas.
- Una subsección para cada pantalla:
 - nombre de la pantalla,
 - propósito,
 - componentes principales (botones, formularios, listas, etc.), ◦ captura de pantalla (enlazada o referenciada desde `/docs/img/`).

Paso 6: Commit y push

- Hacer commit con un mensaje descriptivo, por ejemplo: `feat(ui): prototipo HTML/CSS basado en Stitch (8 pantallas)`
- Hacer push a la rama correspondiente del equipo. **Cierre**
 - Breve socialización: cada equipo muestra 2-3 pantallas clave.
 - Retroalimentación rápida del docente sobre:
 - coherencia visual,
 - uso de HTML semántico,
 - primeras evidencias de accesibilidad.

5. Resultados

- Prototipo de interfaz con 8 pantallas implementadas en



UNL

Universidad
Nacional
de Loja

1859

FEIRNNR - Carrera de Computación

The screenshot displays the CODIUM application interface across four main sections:

- Perfil de Usuario - Te...**: Shows the user profile of Ana Torres (@anaftorres), including her name, profile picture, and activity status ("Racha más larga 42 días"). It also lists her work experience ("Trabajo Estudiante") and education ("Institución Universidad Nacional").
- Mis Retos - Resueltos...**: Shows a list of solved challenges (Resueltos) with details like title, points, and solution language (e.g., "Solución en Python", "Solución en JavaScript").
- Mis Retos - Resuelto...**: Shows a list of solved challenges (Resueltos) with details like title, points, and solution language (e.g., "Python Validación de Palíndromos", "JavaScript Inversión de Cadena de Texto").
- Detalle del Reto - Te...**: Provides detailed information about a challenge titled "Suma de dos números". It includes a description, examples of input and output, restrictions, and a "Resolver Ahora" button.

El flujo de trabajo que sigue el prototipo desarrollado corresponde a distintos casos de uso que puede tener el usuario el flujo completo permitirá al usuario conocer el proceso de iniciar sesión, observar el muro de interacción donde los usuarios pueden compartir ideas de solución o dificultades que han tenido al solucionar los retos.

Adicional podrá acceder al panel de los retos para conocer retos nuevos, soluciones enviadas y calificaciones obtenidas por cada uno de los retos que el usuario ha logrado solucionar

Por último, el cliente podrá acceder a los datos de su perfil si accede al botón de “perfil” presente en el panel de navegación ubicado en la parte inferior.



UNL

Universidad
Nacional
de Loja

1859

FEIRNNR - Carrera de Computación

1. Inicio de sesión:

Esta pantalla está diseñada para proporcionar al usuario un formulario de datos necesarios para identificar a cada uno de los usuarios dentro de la aplicación. Por ser prototipo no tiene interacción de entrada de los datos pero al presionar el botón iniciar sesión de color azul dará paso al panel de inicio.

El prototipo muestra dos versiones de la pantalla de inicio de sesión:

Versión 1 (izquierda): Una interfaz sencilla con un cuadro de texto para "Correo electrónico" (tu@email.com) y un cuadro para "Contraseña" (Ingrresa tu contraseña). Hay un enlace "¿Olvidaste tu contraseña?". Abajo se encuentran los botones "Iniciar Sesión" (en azul) y "Continuar con Google".

Versión 2 (derecha): Una interfaz más avanzada que incluye un cuadro para "Contraseña" con un icono para alternar visibilidad. Hay un enlace "¿Olvidaste tu contraseña?". Los botones principales son "Iniciar Sesión" y "Continuar con Google".

2. Muro de interacción social:

Esta pantalla está construida para dar al usuario la funcionalidad de compartir sus ideas con el resto de los usuarios de la aplicación compartiendo ayudas para resolver un reto o comentando las dificultades que han tenido en otros determinados retos.

El prototipo muestra una lista de publicaciones en el muro:

- Maria Rodriguez:** "Resolviendo el problema de la mochila". La programación dinámica fue clave para encontrar la solución óptima. Comparto mi código y el proceso de pensamiento que seguí para llegar a él... [Ver más](#). (35 likes, 12 comments)
- Juan Perez:** "Mi experiencia con el reto FizzBuzz". Fue un reto interesante, al principio intenté resolvélo con un bucle for, pero luego exploré una solución recursiva que resultó ser más elegante... [Ver más](#). (12 likes, 5 comments)

Abajo se muestra un cuadro vacío con el mensaje "El muro está vacío".

A la parte inferior hay un menú con íconos para "Inicio", "Retos", "Ranking" y "Perfil".



UNL

Universidad
Nacional
de Loja

1859

FEIRNNR - Carrera de Computación

3. Panel de nuevos retos:

Esta pantalla permite a los usuarios entrar a una bandeja de mensajería donde podrá encontrar cada uno de los nuevos retos a los cuales puede dar solución.

The interface consists of a header with the title 'Mis Retos', total points (7500 pts), and a notification bell. Below the header are tabs for 'Nuevos', 'Resueltos', and 'Envíos'. The 'Nuevos' tab is currently active, showing two challenges: 'Suma de dos números' (Difficulty: Fácil) and 'Invertir Árbol Binario' (Difficulty: Media). Each challenge has a 'Detalle' button. The 'Resueltos' tab shows the same challenges, indicating they have been solved. The bottom navigation bar includes 'Inicio', 'Retos' (highlighted in blue), 'Ranking', and 'Perfil'.

4. Panel de retos resueltos:

La ventana de retos resueltos está destinada a presentar una lista con el historial de todos los retos que ha podido responder nuestro usuario, detallando la información de número de puntos obtenidos y lenguaje de programación utilizado.

The interface consists of a header with the title 'Mis Retos', total points (7500 pts), and a notification bell. Below the header are tabs for 'Nuevos', 'Resueltos', and 'Envíos'. The 'Resueltos' tab is currently active, showing two solved challenges: 'Python Validação de Palíndromos' (+150 pts) and 'JavaScript Inversión de Cadena de Texto' (+100 pts). Each challenge has an 'Observar envíos' button. The 'Nuevos' tab shows the same challenges, indicating they have been solved. The bottom navigation bar includes 'Inicio', 'Retos' (highlighted in blue), 'Ranking', and 'Perfil'.



5. Descripción de los retos entrantes:

Después de entrar a la bandeja de entrada de los nuevos retos se puede acceder a una subpestana que muestra la información referente al reto

Detalle del Reto

Suma de dos números

Descripción

Dado un array de enteros 'nums' y un entero 'target', retorna los índices de los dos números tal que su suma sea igual a 'target'. Puedes asumir que cada entrada tendrá exactamente una solución, y no puedes usar el mismo elemento dos veces. Puedes retornar la respuesta en cualquier orden.

Ejemplos

```
Input: nums = [2,7,11,15], target = 9
Output: [0,1]

Explicación: Porque nums[0] + nums[1] == 9, retornamos [0,1].
```

```
Input: nums = [3,2,4], target = 6
Output: [1,2]
```

Restricciones

- $2 \leq \text{nums.length} \leq 10^4$

Resolver Ahora

Inicio Retos Ranking Perfil

Detalle del Reto

Suma de dos números

Descripción

Dado un array de enteros 'nums' y un entero 'target', retorna los índices de los dos números tal que su suma sea igual a 'target'. Puedes asumir que cada entrada tendrá exactamente una solución, y no puedes usar el mismo elemento dos veces. Puedes retornar la respuesta en cualquier orden.

Ejemplos

```
Input: nums = [2,7,11,15], target = 9
Output: [0,1]

Explicación: Porque nums[0] + nums[1] == 9, retornamos [0,1].
```

```
Input: nums = [3,2,4], target = 6
Output: [1,2]
```

Restricciones

- $2 \leq \text{nums.length} \leq 10^4$
- $10^9 \leq \text{nums}[1] \leq 10^9$
- $10^9 \leq \text{target} \leq 10^9$

Resolver Ahora

Inicio Retos Ranking Perfil

6. Panel de soluciones enviadas para cada reto:

Después de que el usuario elige un reto para dar solución, puede generar la solución de múltiples formas y usando múltiples lenguajes, esta pestaña le permite ver todas las respuestas que ha generado a un reto en específico.

Mis Retos

Nuevos Resueltos Envíos

- Solución en Python
32 líneas - Tests: 7/7
- Solución en Python
45 líneas - Tests: 4/7
- Solución en JavaScript
28 líneas - Tests: 5/5

Inicio Retos Ranking Perfil

Mis Retos

Nuevos Resueltos Envíos

- Solución en Python
32 líneas - Tests: 7/7
- Solución en Python
45 líneas - Tests: 4/7
- Solución en JavaScript
28 líneas - Tests: 5/5

Inicio Retos Ranking Perfil



UNL

Universidad
Nacional
de Loja

1859

FEIRNNR - Carrera de Computación

7. Código en lenguaje de programación enviado:

Para cada una de las soluciones que puede generar el usuario se puede observar el código fuente, esta pestaña es la encargada de mostrar dichos códigos.

The image contains two side-by-side screenshots of a mobile application interface. Both screens have a header "Solución en Python".
The left screen shows a large block of Python code for a function named "two_sum". The code uses a dictionary "num_map" to store each number and its index as it iterates through the input list "nums". It returns the indices of the first pair of numbers that add up to the target sum.
The right screen shows another instance of the "Solución en Python" screen, displaying a similar but slightly different version of the same function. This second version includes additional comments and handles edge cases like no solution or very long lists.

8. Tabla de ranking de los usuarios:

Como su nombre ya da una pista, está encargada de mostrar el ranking global de todos los usuarios de la aplicación y así mismo mostrar el puesto del usuario corriente.

The image contains two side-by-side screenshots of a mobile application interface. Both screens have a header "Ranking de Usuarios".
The left screen displays a table of users with their names, points, and programming languages. The columns are: Name, Points, Language, and Retos. The users listed are Sarah L. (11800 pts, JS), Mike D. (12500 pts, Python), Ken T. (11500 pts, C++), John P. (11200 pts, Java), Emily R. (10900 pts, Go), and Lisa K. (10500 pts, Ruby).
The right screen shows a similar ranking table with users Sarah L., Mike D., Ken T., John P., Emily R., and Lisa K. The user "Tú" (the current user) is highlighted with a blue border and labeled "27" at the top left of the row. The bottom navigation bar for both screens includes "Inicio", "Retos", "Ranking", and "Perfil".



UNL

Universidad
Nacional
de Loja

1859

FEIRNNR - Carrera de Computación

9. Perfil de usuario:

Esta última ventana está encargada de mostrar la información completa de los usuarios, esto contiene los nombres de usuario, los nombres completos del usuario su puntaje total.

- HTML/CSS, coherentes con el flujo del proyecto.

```
<!DOCTYPE html>
<html lang="es"><head>
<meta charset="utf-8"/>
<meta content="width=device-width, initial-scale=1.0" name="viewport"/>
<title>Iniciar Sesión / Registrarse - Tema Claro</title>
<link href="https://fonts.googleapis.com" rel="preconnect"/>
<link crossorigin="" href="https://fonts.gstatic.com" rel="preconnect"/>
<link href="https://fonts.googleapis.com/css?family=Space+Grotesk:wght@400;500;700&display=swap" rel="stylesheet"/>
<link href="https://fonts.googleapis.com/css?family=Material+Symbols+Outlined:wght@400" rel="stylesheet"/>
<script src="https://cdn.tailwindcss.com/plugins-forms/container-queries"></script>
<script>
tailwind.config = {
  darkMode: "class",
  theme: {
    extend: {
      colors: {
        "primary": "#113388",
        "background-light": "#f6f6f6",
        "background-dark": "#111111",
        "text-light": "#585858",
        "border-light": "#e0e0e0"
      },
      fontFamily: {
        "display": ["Space Grotesk", "sans-serif"]
      },
      borderRadius: ["DEFAULT", "1rem", "2rem", "3rem", "full: 9999px"],
      },
    }
  }
</script>
<style>
  .material-symbols-outlined {
    font-variation-settings: 'FILL' 0, 'wght' 400, 'GRAD' 0, 'opsz' 24;
    font-size: 24px;
  }
</style>
```

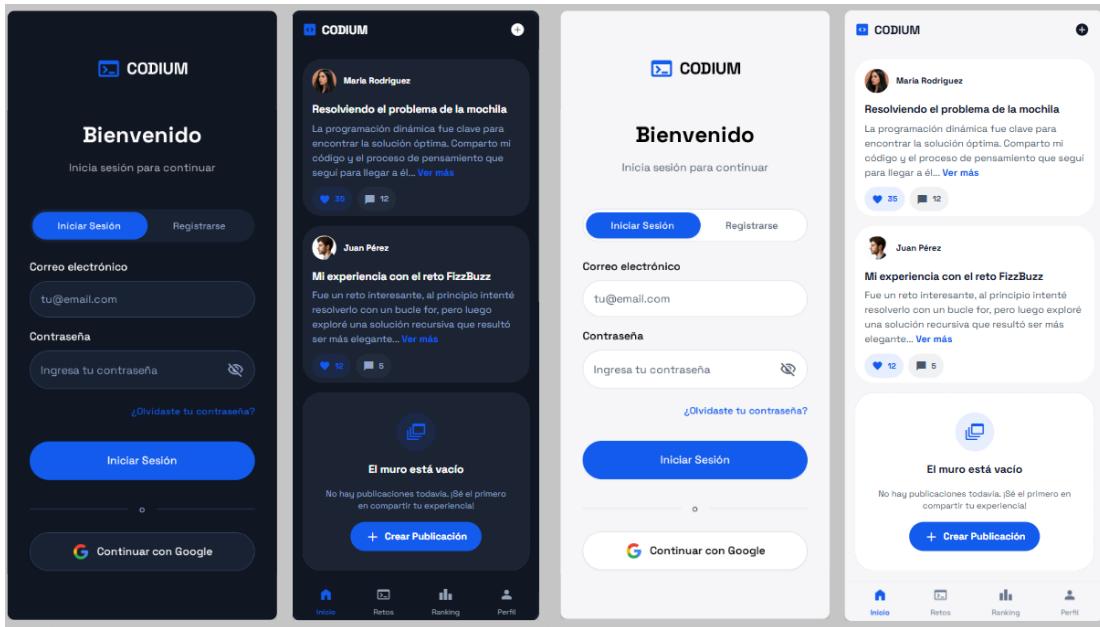


UNL

Universidad
Nacional
de Loja

1859

- Prototipo en Stitch que sirvió de base para la implementación.



6. Preguntas de Control

¿Qué ventajas encuentras al usar primero un prototipo en Stitch antes de codificar en HTML/CSS?

Usar un prototipo en Stitch permite visualizar la estructura y el flujo de las pantallas antes de escribir código. Esto evita retrabajos, facilita validar ideas con el equipo y acelera la toma de decisiones sobre diseño y funcionalidad. Además, permite detectar inconsistencias tempranas y definir mejor la jerarquía visual, reduciendo errores al momento de implementar en HTML/CSS.

¿Por qué es importante que las pantallas estén construidas con etiquetas semánticas?

Las etiquetas semánticas mejoran la accesibilidad, ya que ayudan a lectores de pantalla y otras herramientas a comprender el contenido. También favorecen el posicionamiento SEO, permiten que el código sea más fácil de mantener y facilitan su comprensión por parte de cualquier desarrollador que trabaje en el proyecto. Una estructura semántica también hace que los estilos CSS sean más coherentes y organizados.



¿Qué elementos de diseño responsive aplicaste en, al menos, una de las pantallas?

Se aplicaron principios como el uso de **layouts flexibles (Flexbox/Grid)**, imágenes ajustables al contenedor, **tipografías fluidas**, y puntos de quiebre (breakpoints) para adaptar la distribución del contenido a pantallas pequeñas. También se ajustaron márgenes, tamaños y columnas para que la experiencia sea consistente en móviles, tablets y escritorio.

¿Qué mejoras podrías realizar para mejorar la accesibilidad (colores, tamaños, estructura)?

Algunas mejoras posibles incluyen:

- Aumentar el contraste entre texto y fondo para cumplir con WCAG.
- Incrementar tamaños mínimos de fuente y áreas clicables.
- Asegurar una estructura clara con encabezados ordenados (h1, h2, h3).
- Incluir textos alternativos en imágenes.
- Agregar etiquetas ARIA cuando sea necesario.
- Evitar colores como único medio para comunicar información.

Estas mejoras hacen que las pantallas sean más inclusivas para usuarios con discapacidades visuales, motoras o cognitivas.

¿Cómo ayuda esta práctica al desarrollo posterior del frontend definitivo del proyecto?

Esta práctica agiliza el desarrollo del frontend porque define desde el inicio la estructura, comportamiento y flujo de las pantallas. Disminuye errores de implementación, permite que el equipo trabaje con una visión unificada y facilita la transición a código real. Además, ayuda a crear componentes reutilizables, definir reglas de diseño y asegurar coherencia en todo el proyecto.

7. Conclusiones

- Realizar prototipos de las vistas con las que contará el sistema nos ayuda a tener una vista previa sobre la funcionalidad que debe tener el mismo, con esto logramos obtener un producto final acorde a lo que nos ha pedido nuestro cliente.
- Tener prototipos con navegación entre las distintas pantallas ayuda al usuario a manera de guía para saber como funcionará el producto final. De esta manera puede solicitar al equipo de desarrollo cambios con lo que respecta a la UI y la UX.