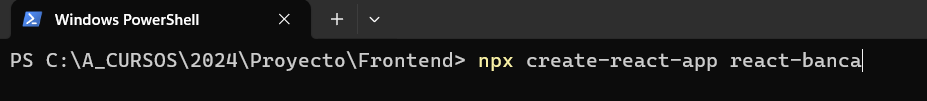
**PROYECTO EJERCICIOS PRACTICOS CON REACT**

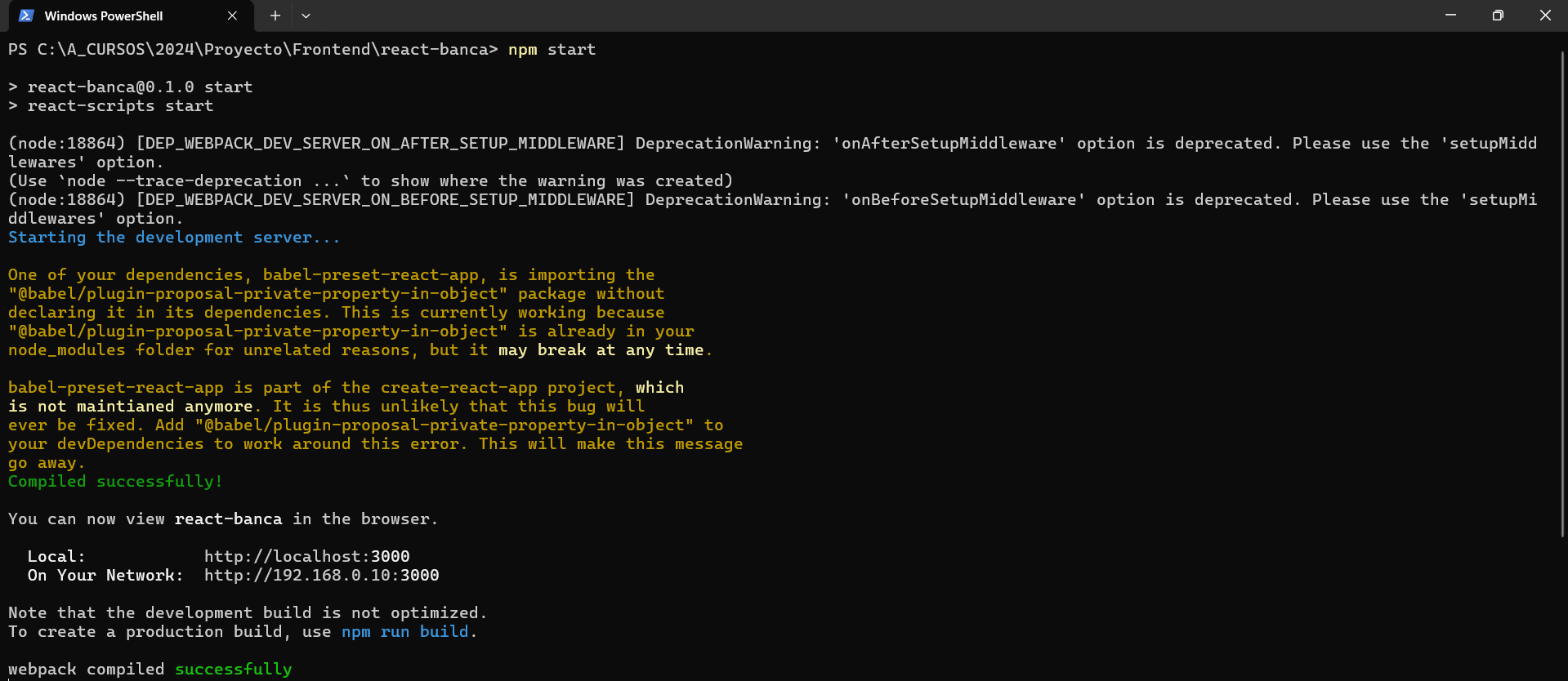
Proyecto de estudio de ejercicios para react utilizando librerías básicas de React

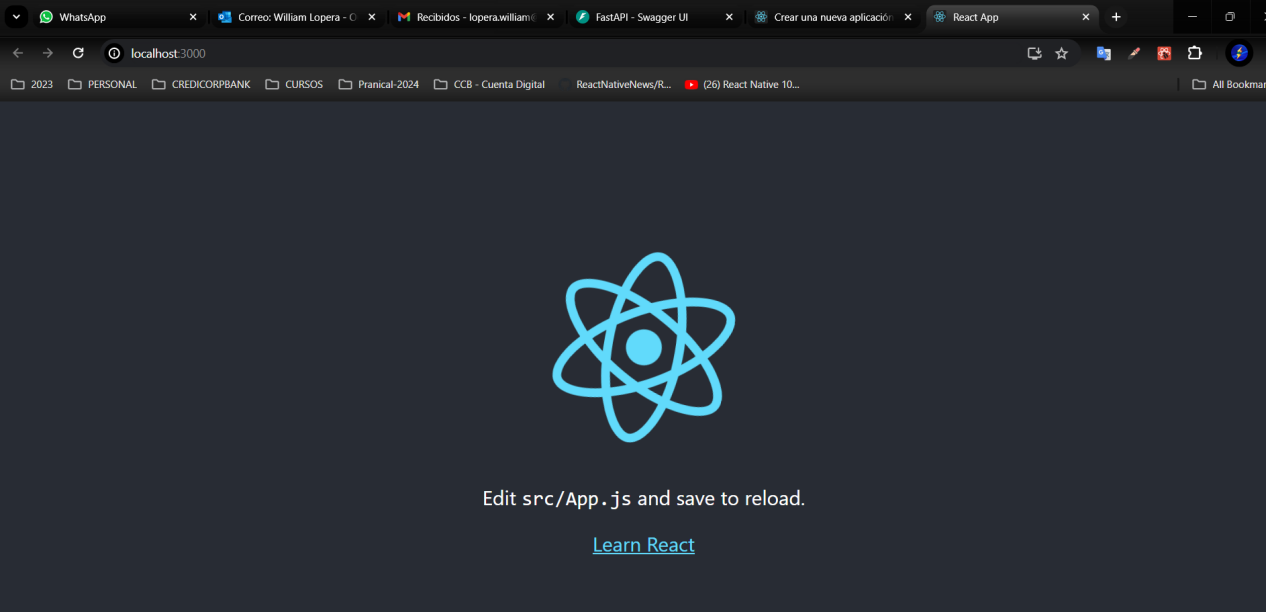
Se mostrara un ejercicio y su solución utilizando lo menos posible internet. Para practicar React, Javascript, Html y CSS.

* Crear proyecto

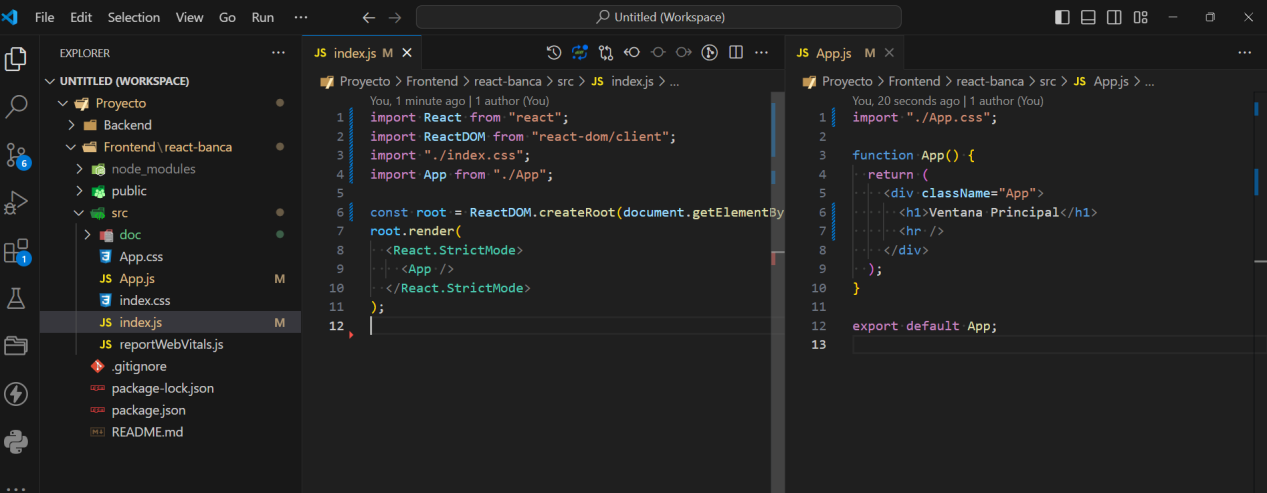


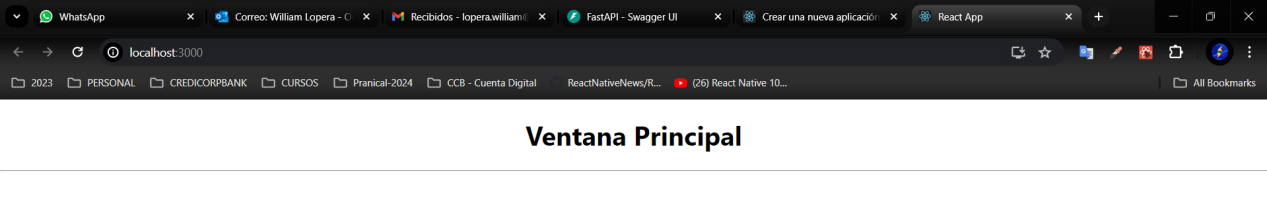
* Subir servidor





* VSCODE: Depurar el código





**Código**

Se crearon cuatro componentes importantes:

* Header: Cabecera de la Pantalla
* Footer: Mensajes de información al usuario y del sistema
* Menu: Area para agregar valores verticales de Menú (dinámico)
* Body: Cuerpo del programa. Este área varia a medida que se seleccione componentes del menú

Además se crearon componentes para el manejo de:

* mensaje: Control de mensajes
* HeaderProcess: Control de titulo, descripción y problema a resolver

Uso de mobx para Tener un almacén común de datos (por ahora mensajes al usuario)

* Librerías:

    "@testing-library/jest-dom": "^5.17.0",

    "@testing-library/react": "^13.4.0",

    "@testing-library/user-event": "^13.5.0",

    "axios": "^1.7.2",

    "mobx": "^6.12.4",

    "mobx-react": "^9.1.1",

    "react": "^18.3.1",

    "react-dom": "^18.3.1",

"react-scripts": "5.0.1",

 "styled-components": "^6.1.11",

    "web-vitals": "^2.1.4"

**Ejercicio 1**

Disponer dos controles de formulario HTML input='number' y un botón. Al presionar el botón mostrar en un alert su suma."

import React from "react";

import { HeaderProcess } from "../../components/headerProcess/HeaderProcess";

import styled from "styled-components";

const title = "Captura de eventos";

const description =

  "Los nombres de eventos en React comienzan con 'on' y luego el primer caracter de cada palabra en mayúsculas: [onClick, onDoubleClick, onMouseEnter, onMouseMove, onKeyPress, onSubmit, etc]";

const exercise =

  "Ejercicio: Disponer dos controles de formulario HTML input='number' y un botón. Al presionar el botón mostrar en un alert su suma.";

export const EventCapture = () => {

  const sum = (e) => {

    e.preventDefault();

    const x = parseFloat(e.target.x.value);

    const y = parseFloat(e.target.y.value);

    return alert(`${x} + ${y} = ${x + y}`);

  };

  return (

    <Container>

      <HeaderProcess

        title={title}

        description={description}

        exercise={exercise}

      />

      <Form onSubmit={sum}>

        <Label>

          Ingrese primer valor:

          <input type="number" name="x" />

        </Label>

        <Label>

          Ingrese segundo valor:

          <input type="number" name="y" />

        </Label>

        <Button>Sumar</Button>

      </Form>

    </Container>

  );

};

const Container = styled.div`

  display: flex;

  flex-direction: column;

  justify-content: center;

  align-items: center;

`;

const Form = styled.form`

  display: flex;

  background-color: lightgreen;

  flex-direction: column;

  align-items: center;

  width: 40vw;

  padding: 20px;

`;

const Label = styled.label`

  padding: 10px;

`;

const Button = styled.button`

  margin-top: 10px;

  background-color: rgb(164, 164, 241);

  width: 10vw;

  height: 4vh;

`;

