**Лабораторна робота № 6**

**Тема:** Основи роботи із бібліотекою React.js . State i Props. Deploy to AWS

**Мета роботи**: Ознайомитись з бібліотекою React.js, навчитись використовувати State i Props та реалізувати deploy to AWS

**Завдання**

Створити додаток на реакт згідно варіанту, в якому необхідно обов’язково використати props, state, створити мінімум дві компоненти, що мають бути вкладеними.

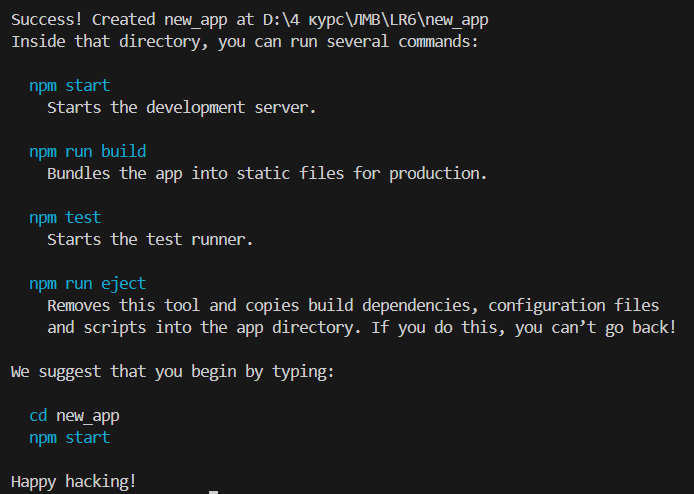
1. Бігущий рядок. Створити програму в якому введена користувачем стрічка перетворюється в рядок компонентів із букв та випадковою заливкою, яка має можливість рухатись за принципом бігущого рядка.



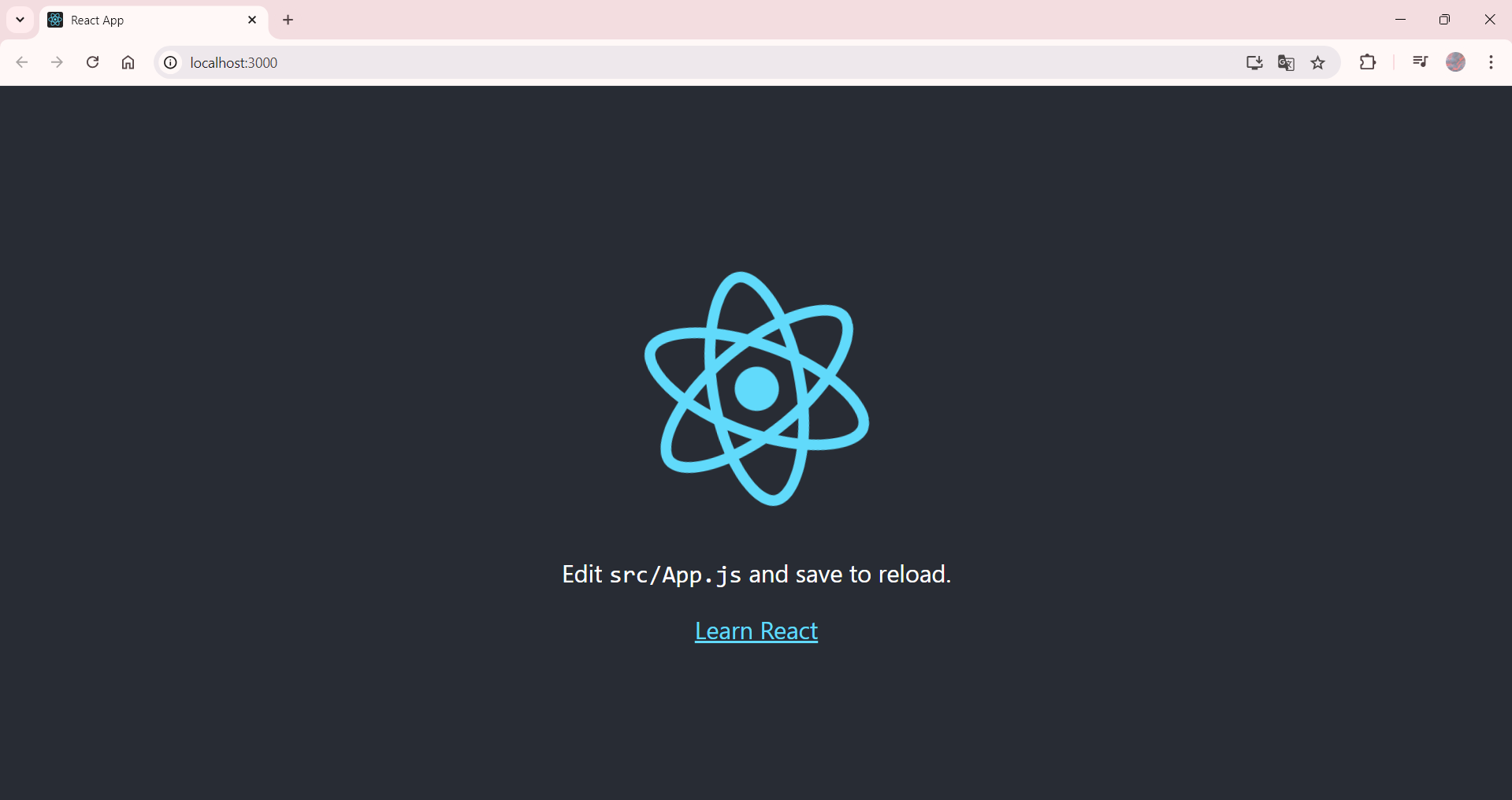
**Хід роботи**

Створила реакт проект за допомогою команди:



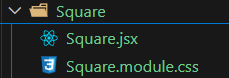


Проект працює:



Далі у структурі проекту в папці src створила папку components/UI, де буду створювати необхідні компоненти.

Спочатку створила компонент та стилі для Square (для відображення квадрату та букви на ньому):



Square.jsx

import React from 'react'

import classes from './Square.module.css'

export default function Square(props) {

  return (

    <div className={classes.square} style={{backgroundColor: props.color}}>

      <h3 className={classes.letter}>{props.letter}</h3>

    </div>

  )

}

Square.module.css

.square {

    width: 50px;

    height: 50px;

    background-color: gray;

    display: flex;

    align-items: center;

    text-align: center;

    justify-content: center;

    margin: 10px;

    border-radius: 5px;

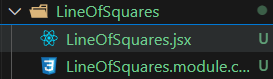
}

.letter {

    color: white;

}

Далі створила компонент LineOfSquares, який буде містити квадрати з буквами, та генерувати і рухати кольори по лінії.



LineOfSquares.jsx

import React, { useEffect, useState } from 'react'

import classes from './LineOfSquares.module.css'

import Square from '../Square/Square';

function LineOfSquares(props) {

    const [colors, setColors] = useState([]);

    // при першому завантаженні компонента

    useEffect(() => {

        const initialColors = props.text.split('').map(() => randomColor());

        setColors(initialColors);

    }, [props.text]);

   // випадковий колір

    const randomColor = () => {

        const colors = ['#7890BF', '#3C8A67', '#c4a701']; // синій, зелений, жовтий

        const randomIndex = Math.floor(Math.random() \* colors.length);

        return colors[randomIndex];

    };

    // оновлення кольорів

    const updateColors = () => {

        setColors((prevColors) => {

        const newColors = [...prevColors];

        // останній колір на першу позицію

        newColors.unshift(newColors.pop());

        return newColors;

        });

    };

    useEffect(() => {

        const interval = setInterval(updateColors, 500);

        return () => clearInterval(interval); // чистка інтервалу при розмонтуванні

      }, [colors]);

      const squares = props.text.split('').map((char, index) => (

        <Square key={index} letter={char.toUpperCase()} color={colors[index]} />

      ));

      return <div className={classes.line\_of\_squares}>{squares}</div>;

}

export default LineOfSquares

LineOfSquares.module.css

.line\_of\_squares{

    display: flex;

    flex-direction: row;

    justify-content: center;

    align-items: center;

    flex-wrap: wrap

}

У цьому компоненті використано хуки:

const [colors, setColors] = useState([]); - для збереження та зміни масиву кольорів для квадратів

Щоб при першому завантаженні компонента задавались випадкові кольори:

useEffect(() => {

        const initialColors = props.text.split('').map(() => randomColor());

        setColors(initialColors);

    }, [props.text]);

При кожній зміні масиву кольорів:

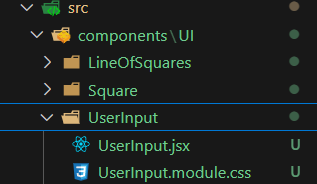
  useEffect(() => {

        const interval = setInterval(updateColors, 500);

        return () => clearInterval(interval); // чистка інтервалу при розмонтуванні

      }, [colors]);

Компонент UserInput створений щоб можна було вводити текст для відображення.



UserInput.jsx

import React from 'react';

import classes from './UserInput.module.css'

function UserInput({onButtonClick, inputRef}) {

    return (

      <div className={classes.userInput}>

        <p>Enter the text: </p>

        <input placeholder='Text' ref={inputRef} />

        <button onClick={onButtonClick}>Click</button>

      </div>

    );

  }

export default UserInput;

У App.js змінила структуру сторінки, та додала хуки:

import './App.css';

import LineOfSquares from './components/UI/LineOfSquares/LineOfSquares';

import UserInput from './components/UI/UserInput/UserInput';

import { useRef, useState } from 'react';

function App() {

  const [text, setText] = useState('');

  const inputRef = useRef(null); // реф для інпуту

  const handleButtonClick = () => {

    if (inputRef.current) {

      setText(inputRef.current.value);

    }

  };

  return (

    <div className="main">

      <UserInput inputRef={inputRef} onButtonClick={handleButtonClick} />

      <LineOfSquares text={text} />

    </div>

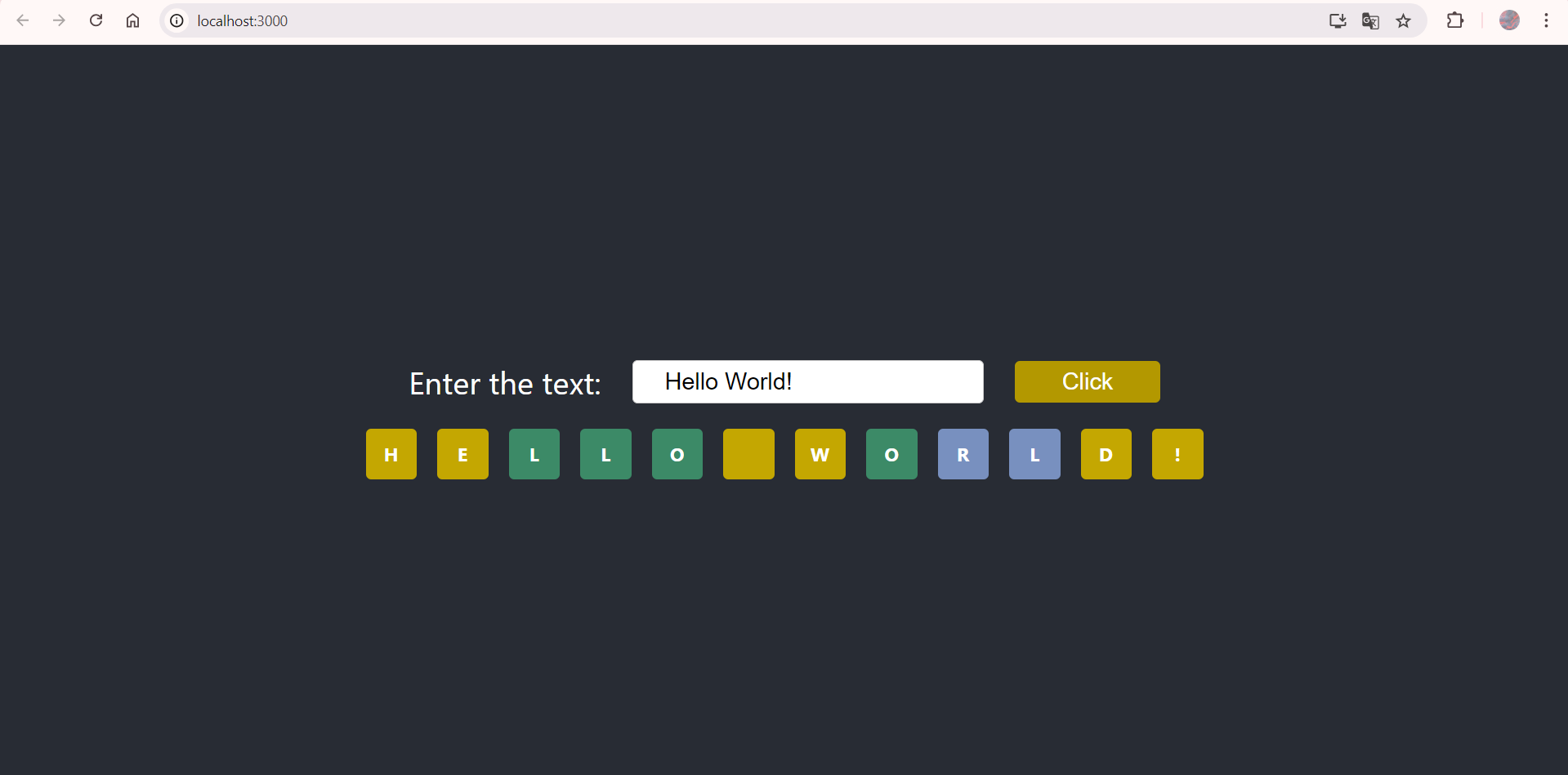
  );

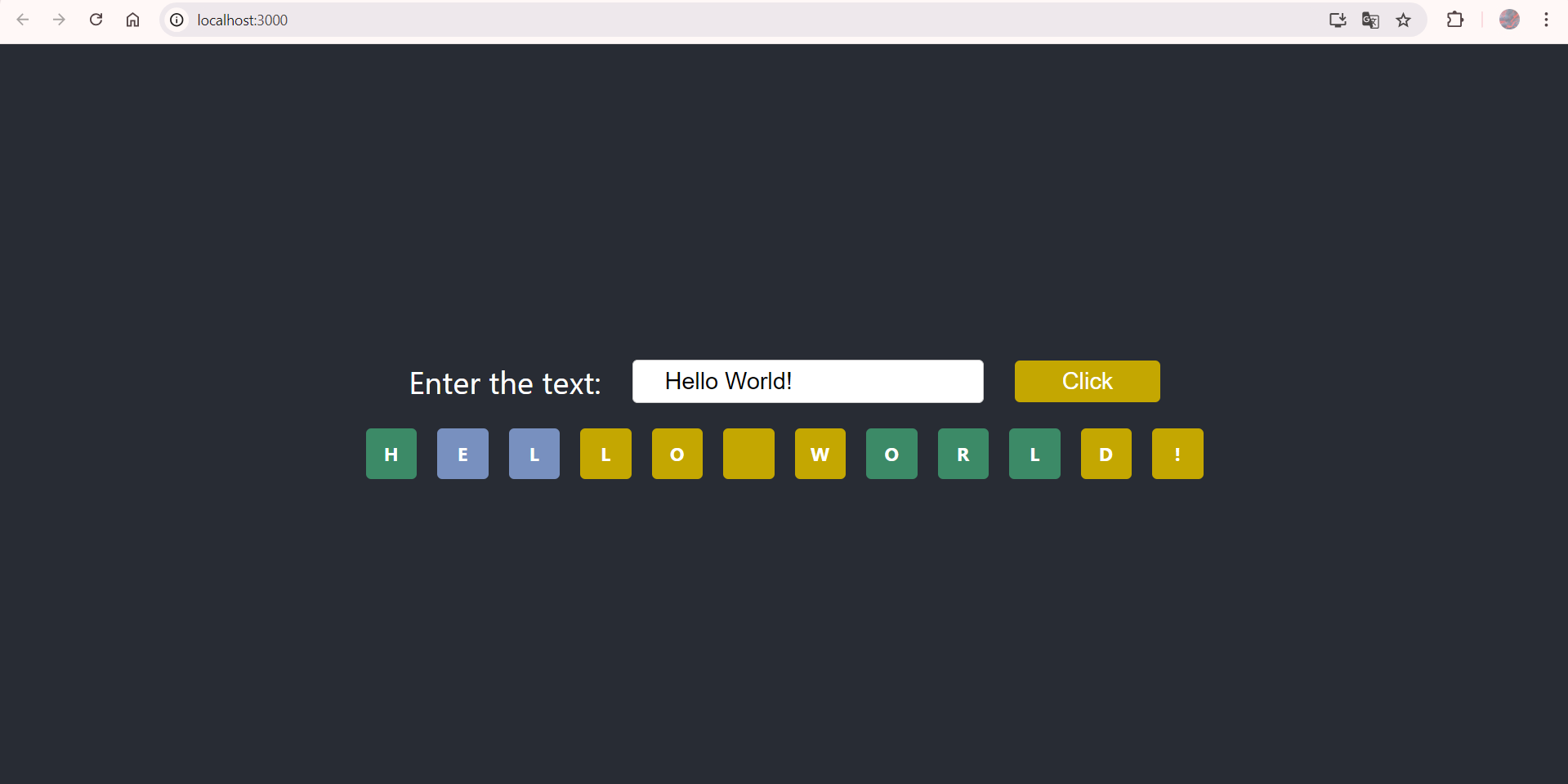
}

export default App;

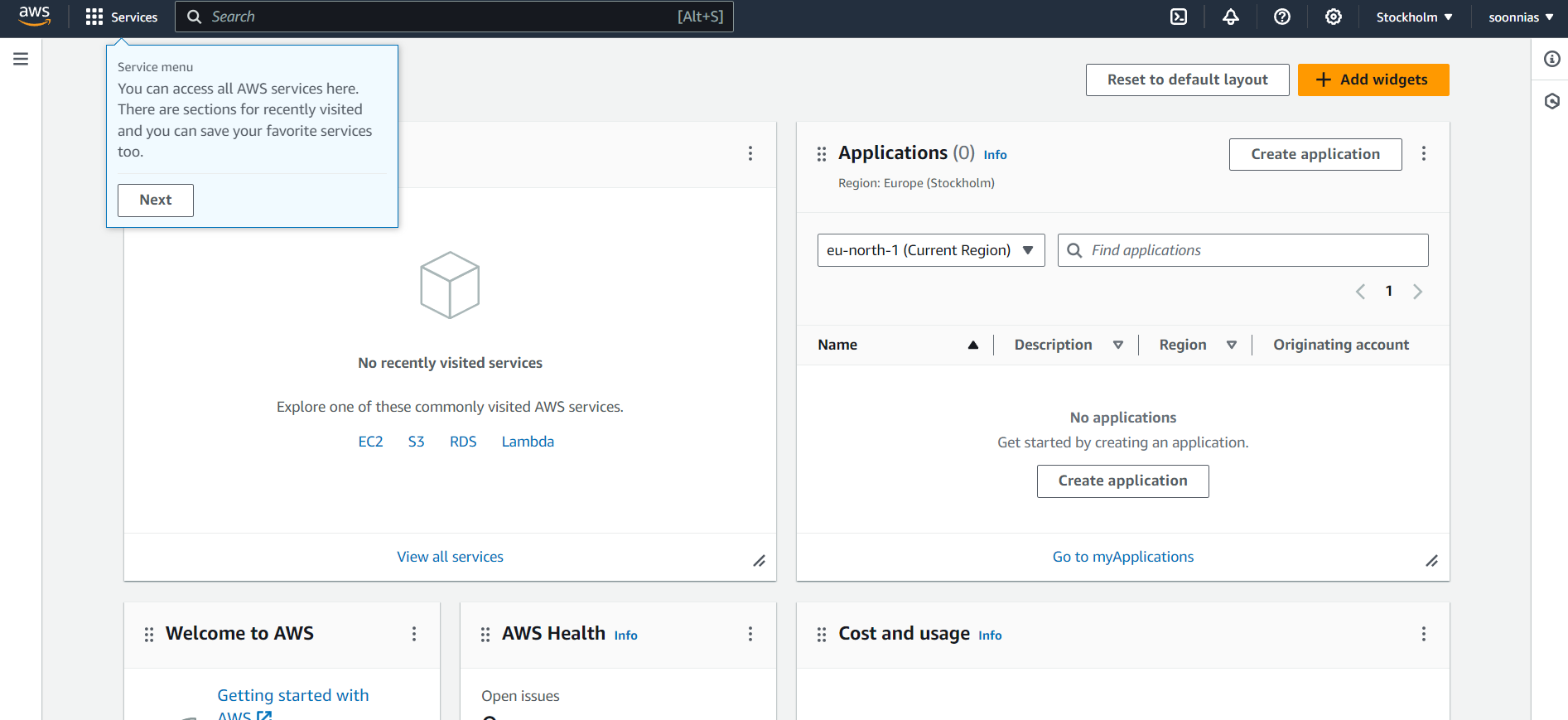
При натиску на кнопку, яка знаходиться в UserInput з використанням useRef береться текст з input та поміщається в state ‘text’, після зміни якого перемальовується елемент LineOfSquares.

Результат роботи:

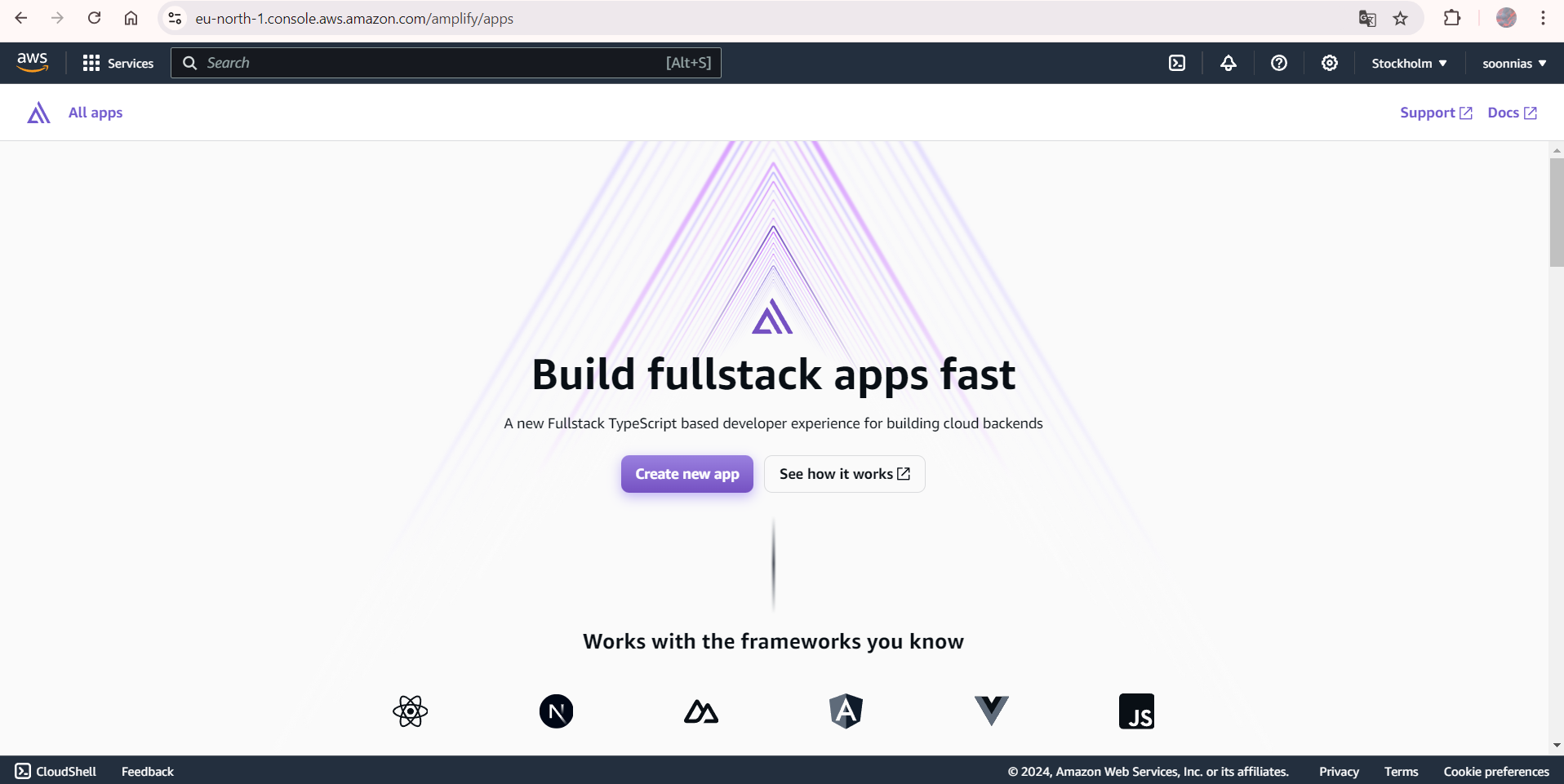




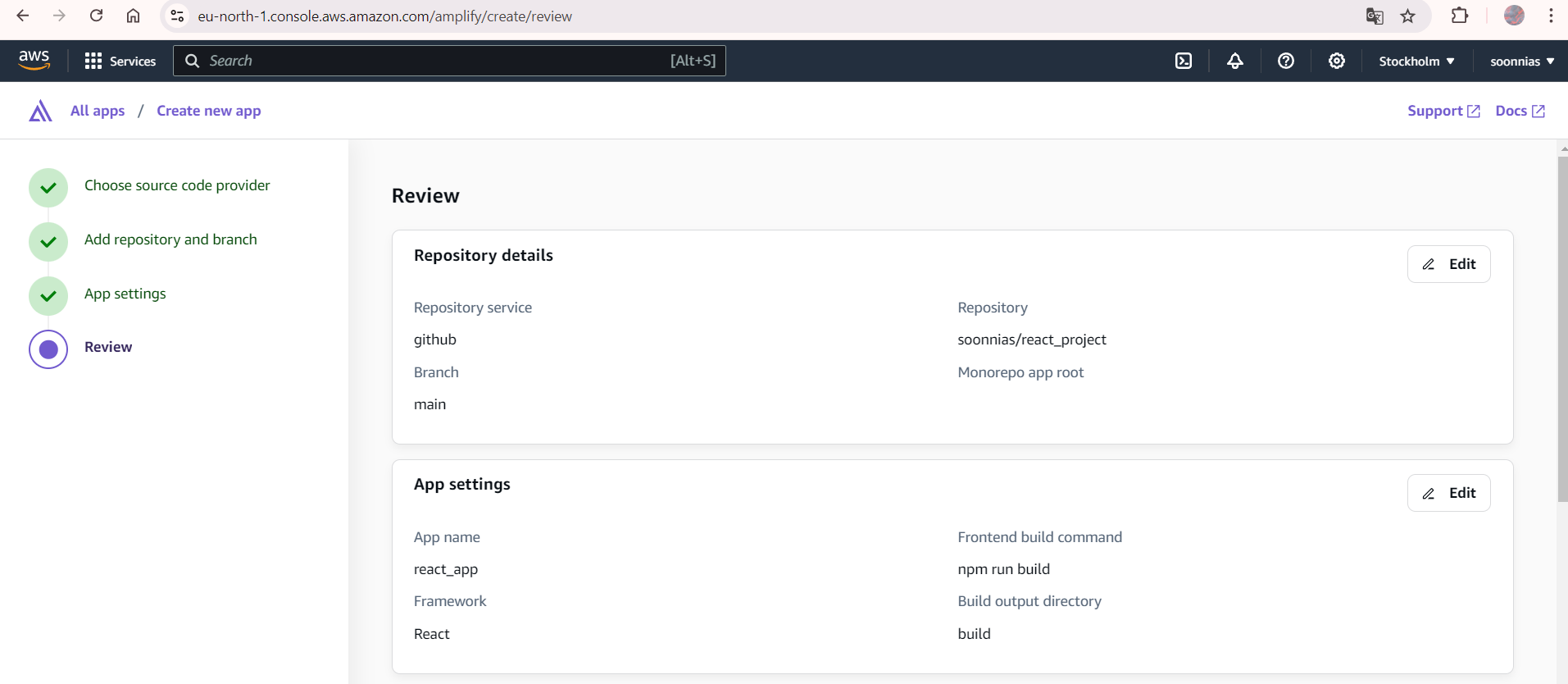
Щоб розмістити проект на AWS сервісі спочатку зареєструвалась на сайті AWS:



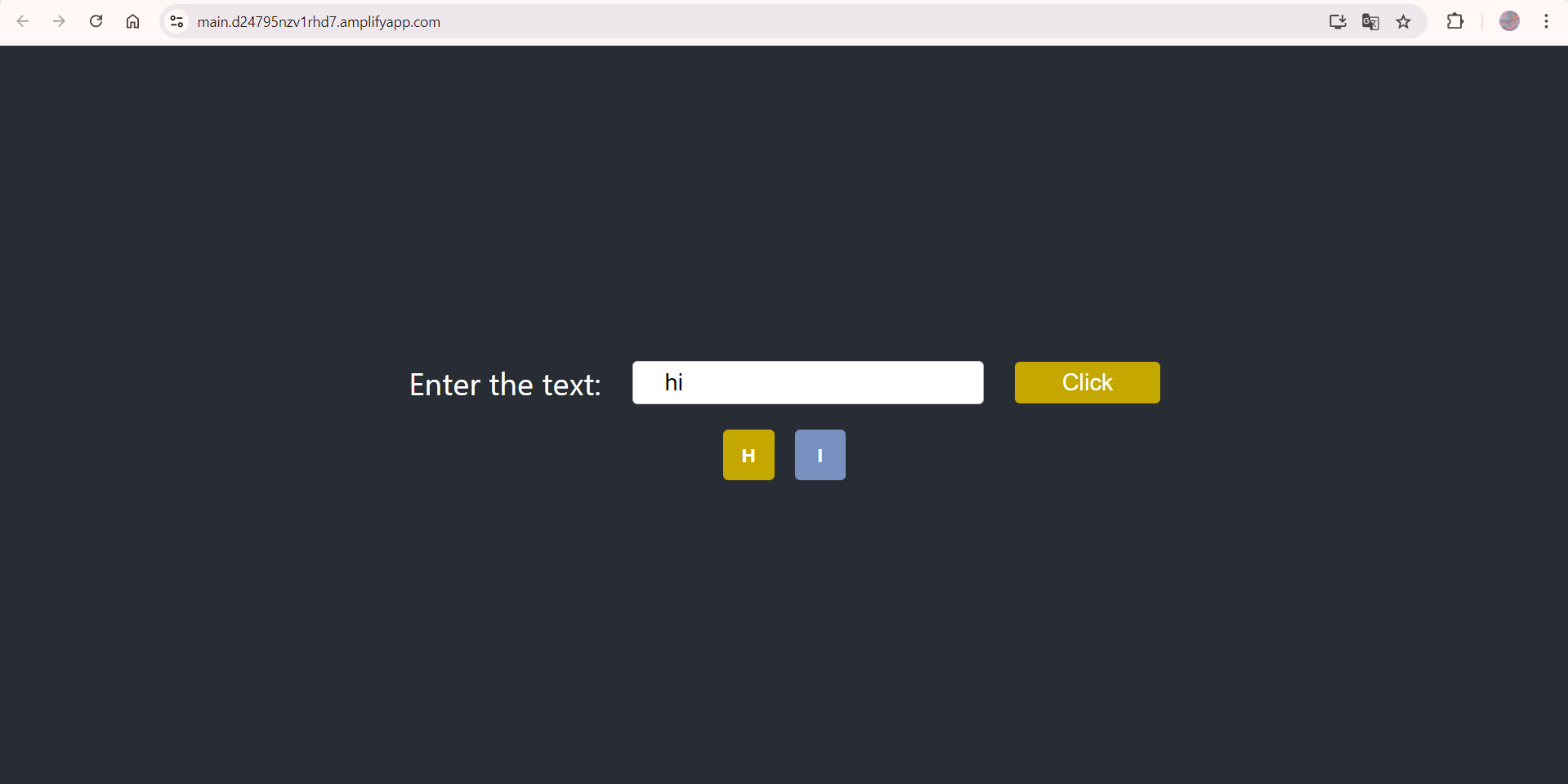
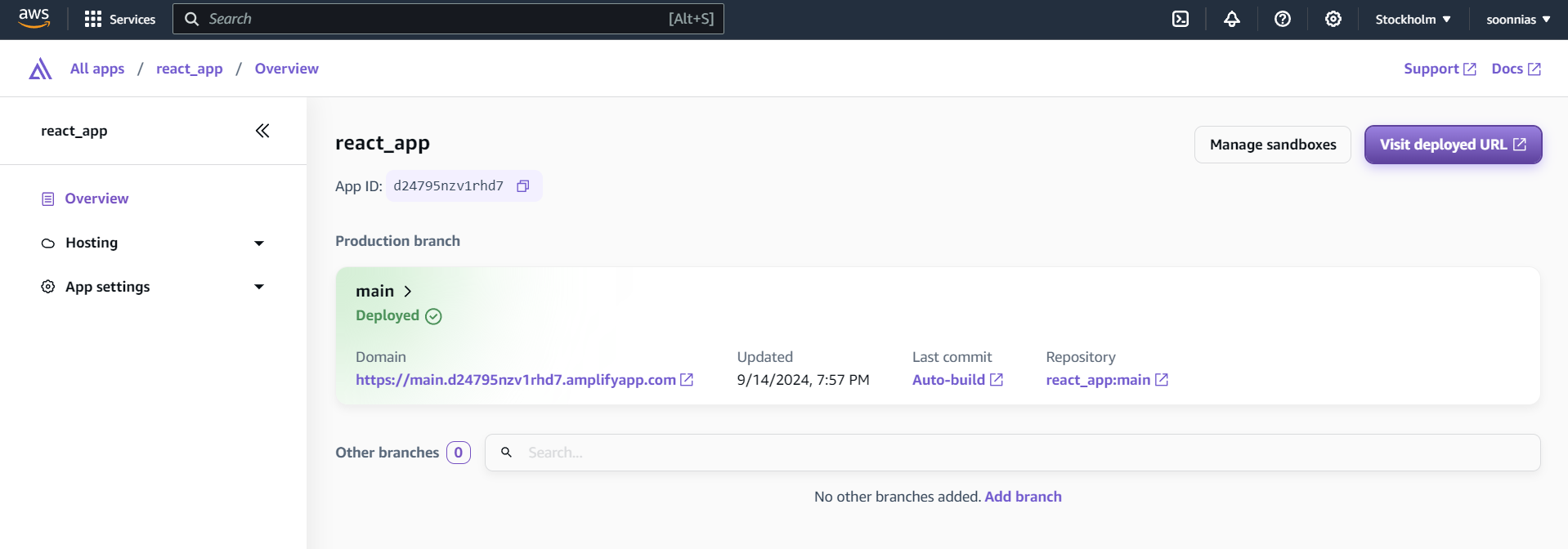
Далі в полі пошуку знайшла сервіс aws amplify та підключила його:



Створила та налаштувала проект натиснувши на відповідну кнопку:



Після успішного створення та розгортання проекту було згенероване посилання на сторінку:



**Висновок**: На цій лабораторній роботі ознайомилась з бібліотекою React.js, навчилась використовувати State i Props та розміщувати проект на AWS.