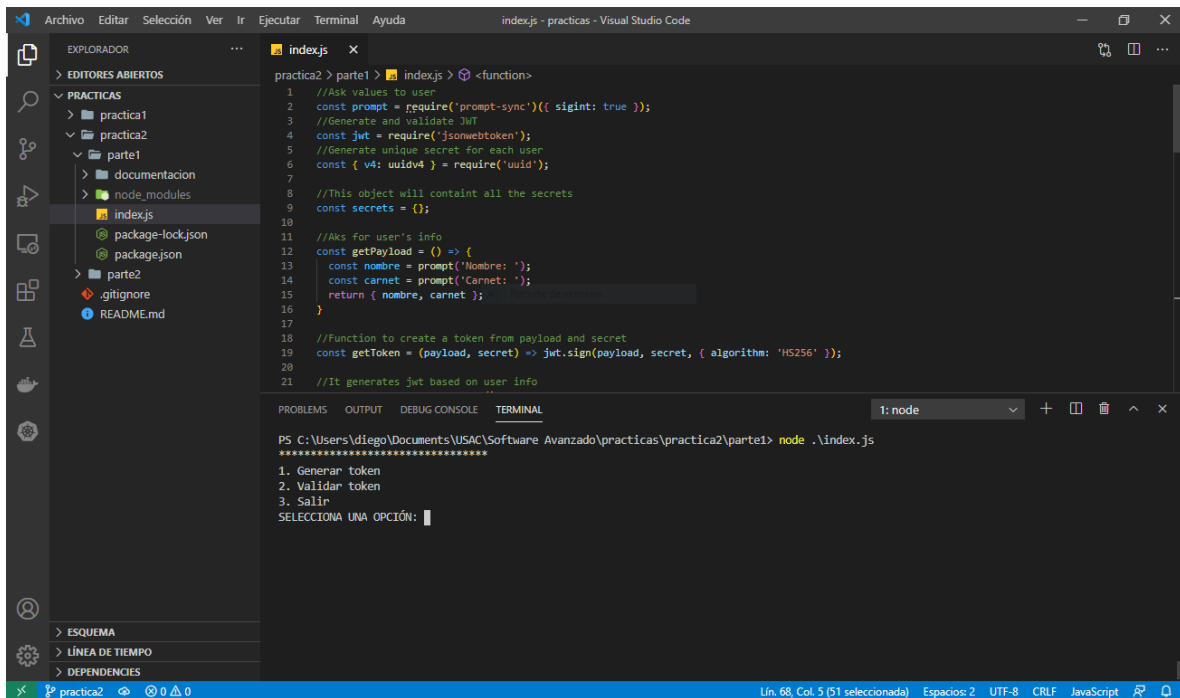
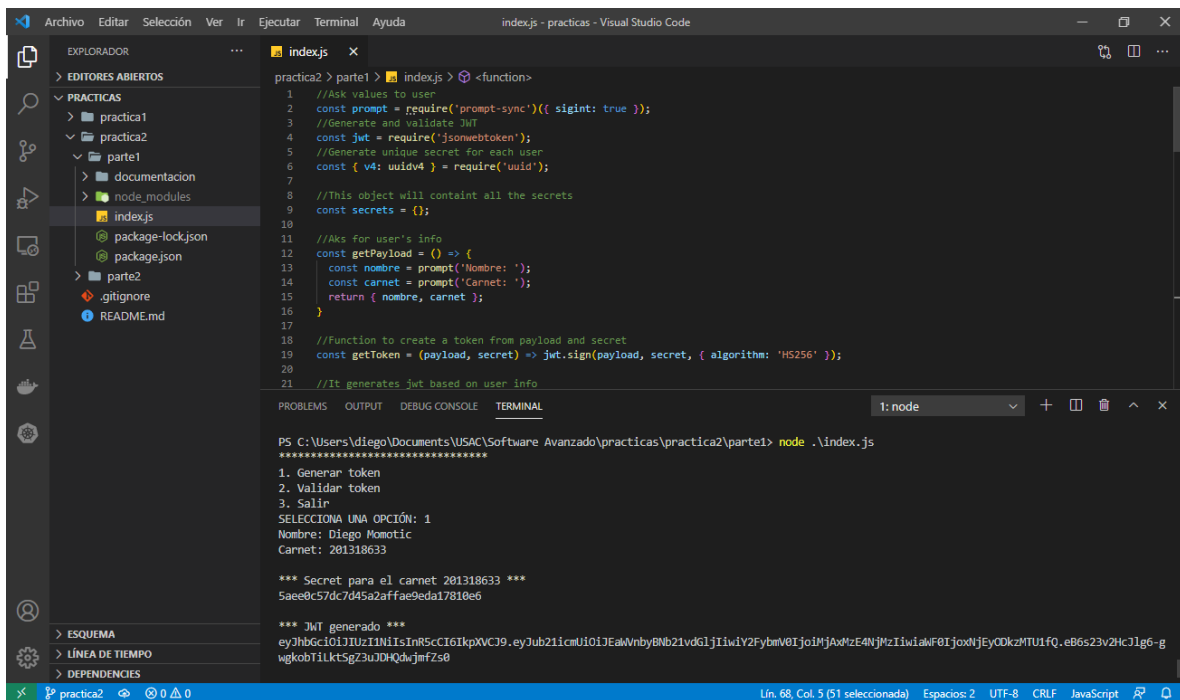


1. Ejecutando programa



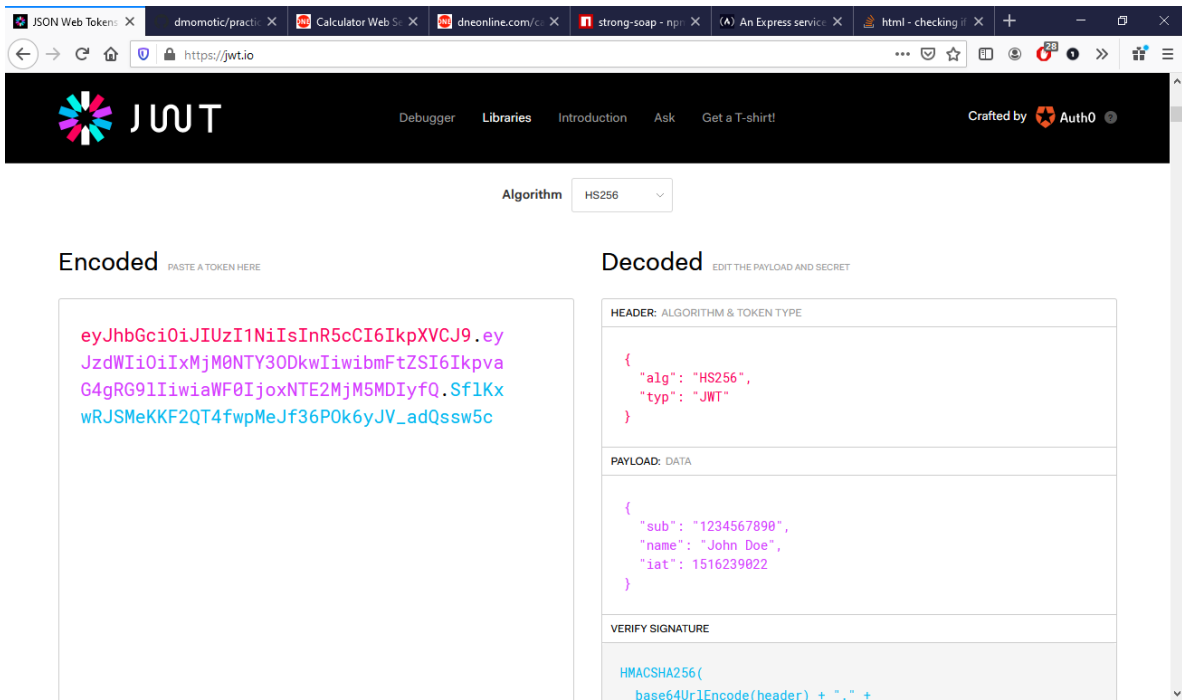
2. Generando token a partir de nombre y carnet solicitado al usuario



3. Validando token generado

[illegible]

4. Validando token invalido obtenido desde <https://jwt.io/>



5. Comprobando token invalido

```
1 //Ask values to user
2 const prompt = require('prompt-sync')({ sigint: true });
3 //Generate and validate JWT
4 const jwt = require('jsonwebtoken');
5 //Generate unique secret for each user
6 const { v4: uuidv4 } = require('uuid');
7
8 //This object will contain all the secrets
9 const secrets = {};
10
11 //Asks for user's info
12 const getPayload = () => {
13   const nombre = prompt('Nombre: ');
14   const carnet = prompt('Carnet: ');
15   return { nombre, carnet };
16 }
17
18 //Function to create a token from payload and secret
19 const getToken = (payload, secret) => jwt.sign(payload, secret, { algorithm: 'HS256' });
20
21 //It generates jwt based on user info
```

Token: eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJzdWIiOiIxMjMNTV3ODkxIiwiaWF0IjoxNTE2MjM5MDIyfQ.SFlKowR3SMekKF2QT4fwp

**** TOKEN INVALIDO ****

1. Generar token
2. Validar token
3. Salir

SELECCIONA UNA OPCIÓN: 3

PS C:\Users\diego\Documents\USAC\Software Avanzado\practicas\practica2\parte1>
PS C:\Users\diego\Documents\USAC\Software Avanzado\practicas\practica2\parte1>
PS C:\Users\diego\Documents\USAC\Software Avanzado\practicas\practica2\parte1>