Project Design Phase

Solution Architecture

Date	30 October 2023
Team ID	NM2023TMID03325
Project Name	Food Tracking System
Maximum Marks	-

Solution Architecture:

Core Features of Customer App
Searching menu: Allow your users to search for different restaurants, cafes by location, and cuisines. Using the search filter, users can easily find their favorite eating places, list menu, offers, etc.
Order placement: The user can place an order of selected dishes and food. They just need to cross-verify their preferred dish, delivery time, and proceed check-out.
Tracking Delivery Partners: With real-time tracking features, it becomes easy for users to track delivery drivers and know their real-time location information. Users can check the time taken by the food delivery executive to deliver their parcel.
Payment gateway integration: You provide the users with multiple payment options like credit/debit cards, different wallets like Google Pay, Paytm, Phonepe, UPI, etc
Core Features of Delivery Partner
Delivery Partner’s profile: Through this feature, a driver can keep his profile update. It contains his full name, address, email, contact number, photo, and other personal information.
Notification for orders: Through push notifications, drivers can get constant updates & alerts for new orders. It will help in the accurate delivery service of your restaurant.
Map for the delivery route: Integrate Google Map or other providers and allow drivers to choose the shortest and fastest routes to reach the location.
Core Features of Food Partners/ Restaurants
Restaurant Profile/Menu: Through this feature, a restaurant owner can add their restaurant details, menu and its availability, price, preparation times, etc
Notification for orders: Through push notifications, Restaurants can get constant updates

& alerts for new orders. It will help in the accurate delivery service of your restaurant.

Notifications for Pickup Partners: They will get alerts about delivery partners, their location when they will pick up, etc.

Payment Details: Information about the payment received from the food delivery system for their orders

Core Features for Food Ordering System Admin

Restaurant management: Being on the admin panel, one can directly manage all the restaurants by adding, updating, and removing any eating joint from the list. He can also check active restaurant status.

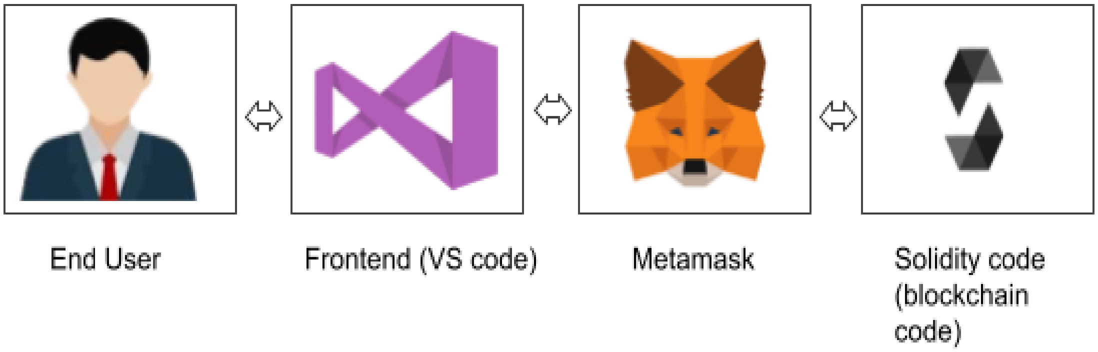
Analytics & report generation: Using analysis and report features, you can get real-time insights of reports and other accounting information which helps you to identify the growth and opportunities to expand reach.

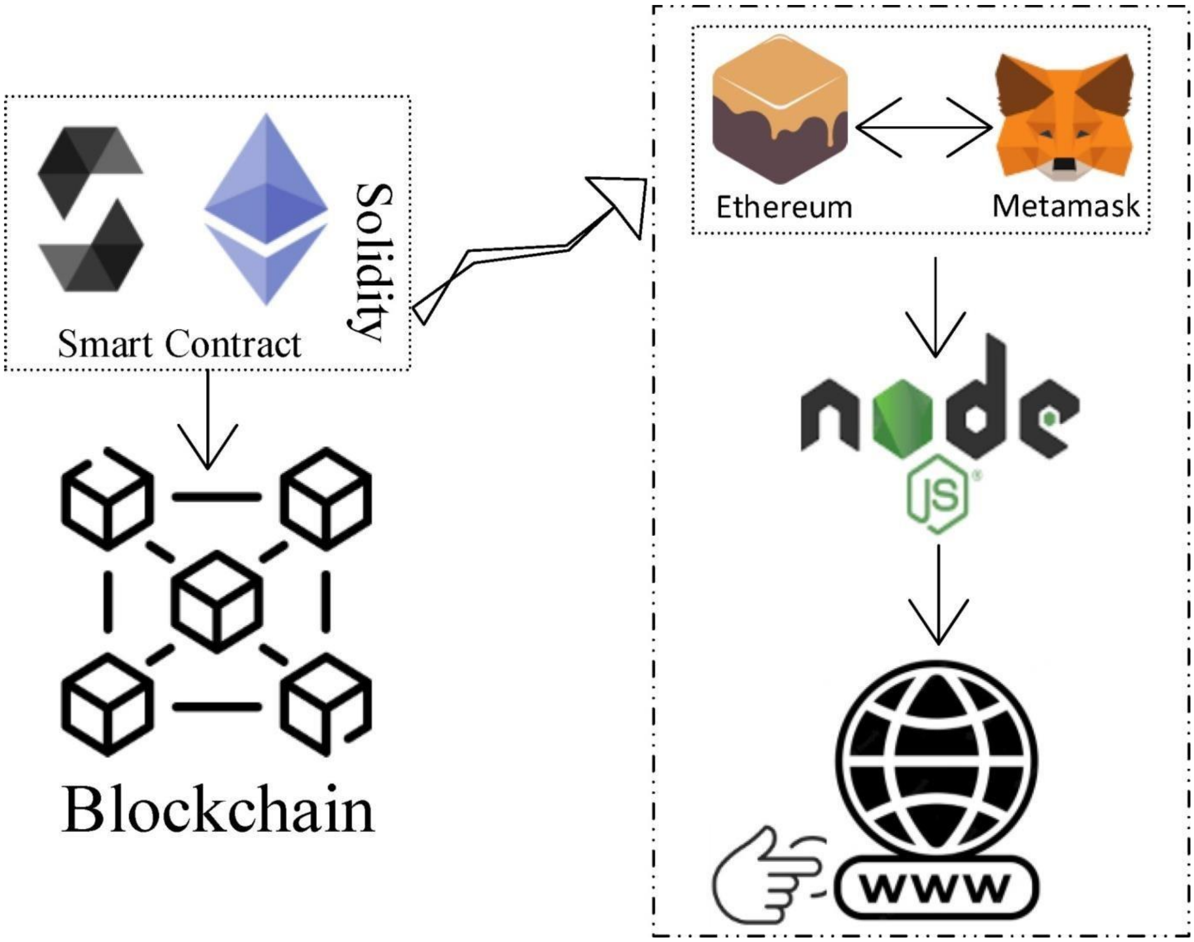
Monitoring every action: Monitor all the food orders, delivery partners, deliveries, reviews & ratings of delivery partners, canceled orders, and other important data related to the driver’s performance.

Application Flow

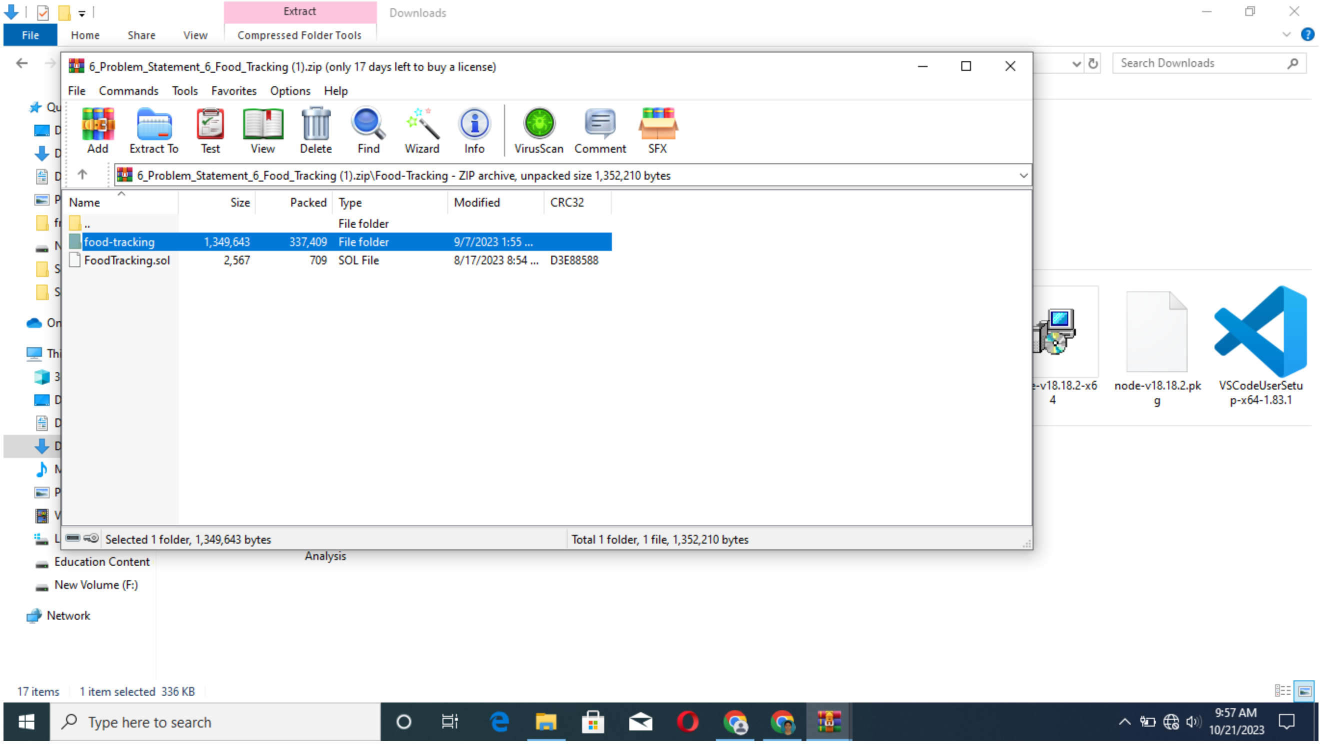
We are considering here microservices-based architecture. Different services are listed in the architecture diagram

Solution Architecture Diagram:





Steps to complete the project



6_Problem_Statement_6_Food_Tracking (1).zip (only 17 days left to buy a license)

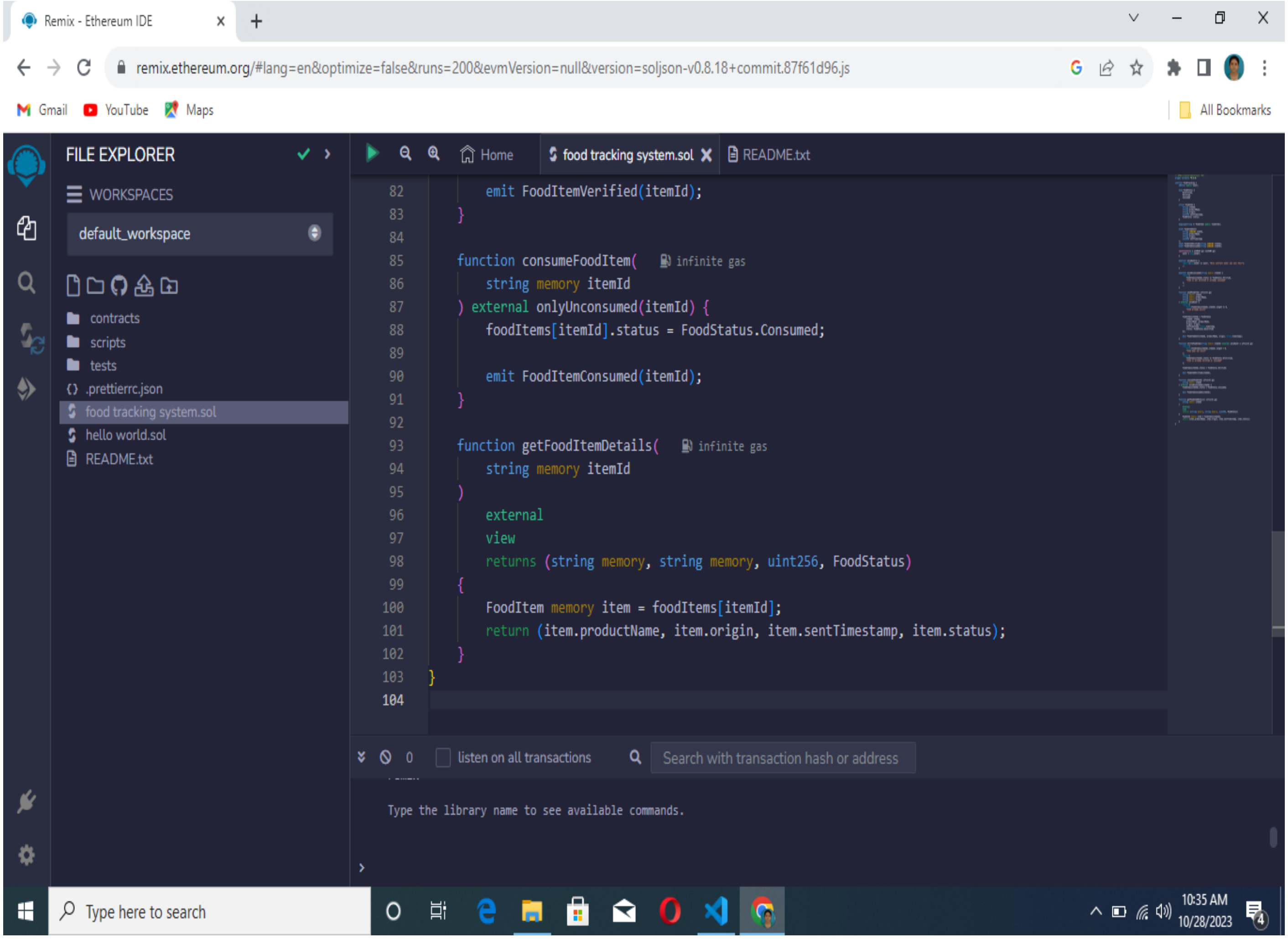
6_Problem_Statement_6_Food_Tracking (1).zip\Food-Tracking - ZIP archive, unpacked size 1,352,210 bytes

Name	Size	Packed	Type	Modified	CRC32
..			File folder		
food-tracking	1,349,643	337,409	File folder	9/7/2023 1:55 ...	
FoodTracking.sol	2,567	709	SOL File	8/17/2023 8:54 ...	D3E88588

Selected 1 folder, 1,349,643 bytes

Total 1 folder, 1 file, 1,352,210 bytes

17 items 1 item selected 336 KB



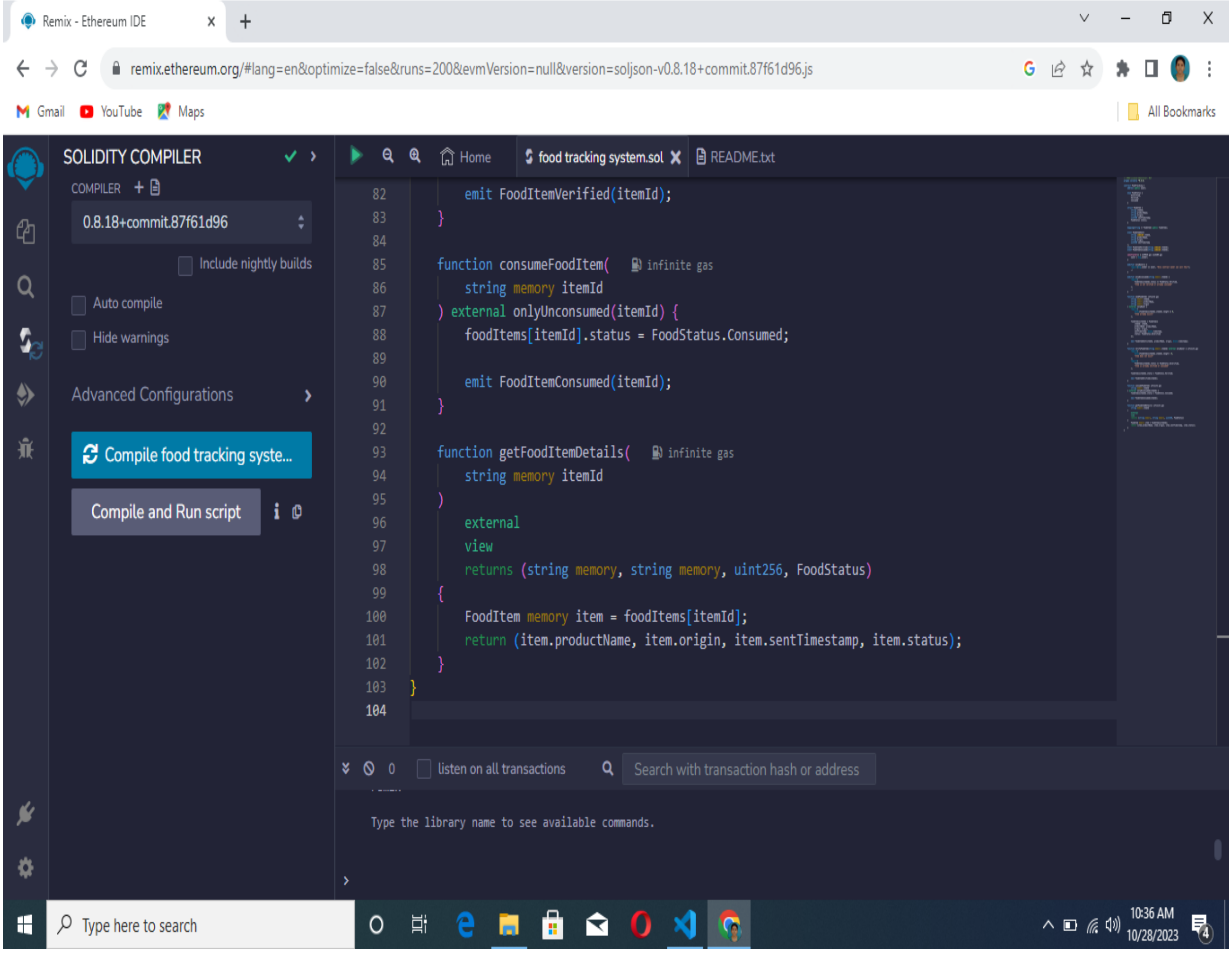
```
        emit FoodItemVerified(itemId);
    }

    function consumeFoodItem(
        string memory itemId
    ) external onlyUnconsumed(itemId) {
        foodItems[itemId].status = FoodStatus.Consumed;

        emit FoodItemConsumed(itemId);
    }

    function getFoodItemDetails(
        string memory itemId
    )
        external
        view
        returns (string memory, string memory, uint256, FoodStatus)
    {
        FoodItem memory item = foodItems[itemId];
        return (item.productName, item.origin, item.sentTimestamp, item.status);
    }
}
```

```
Type the library name to see available commands.
```



remix.ethereum.org/#lang=en&optimize=false&runs=200&evmVersion=null&version=soljson-v0.8.18+commit.87f61d96.js

SOLIDITY COMPILER

COMPILER

0.8.18+commit.87f61d96

□ Include nightly builds

□ Auto compile

□ Hide warnings

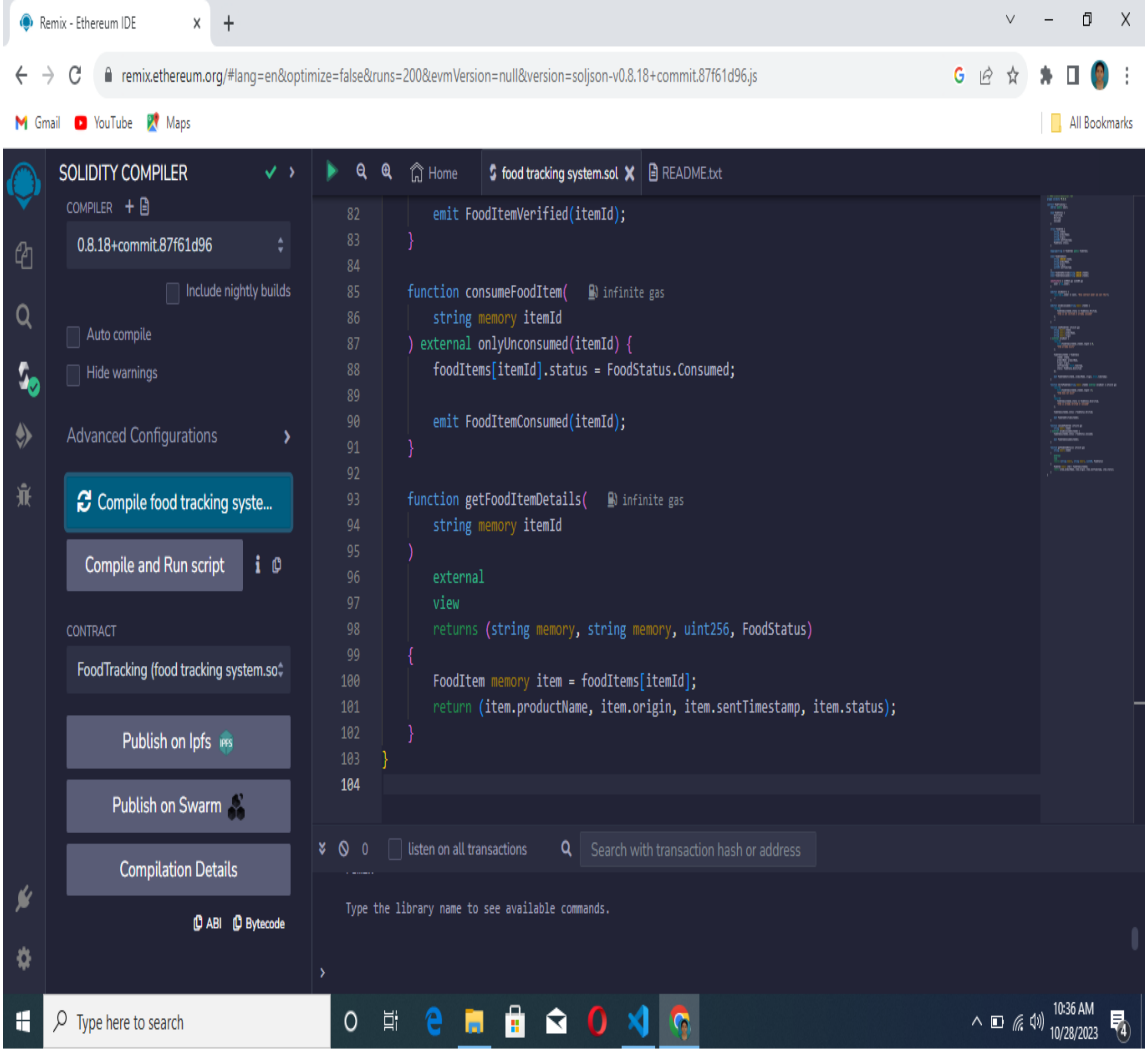
```
        emit FoodItemVerified(itemId);
    }

    function consumeFoodItem(       infinite gas
        string memory itemId
    ) external onlyUnconsumed(itemId) {
        foodItems[itemId].status = FoodStatus.Consumed;

        emit FoodItemConsumed(itemId);
    }

    function getFoodItemDetails(       infinite gas
        string memory itemId
    )
        external
        view
        returns (string memory, string memory, uint256, FoodStatus)
    {
        FoodItem memory item = foodItems[itemId];
        return (item.productName, item.origin, item.sentTimestamp, item.status);
    }
}
```

```
Type the library name to see available commands.
```



remix.ethereum.org/#lang=en&optimize=false&runs=200&evmVersion=null&version=soljson-v0.8.18+commit.87f61d96.js

SOLIDITY COMPILER

COMPILER

0.8.18+commit.87f61d96

□ Include nightly builds

□ Auto compile

□ Hide warnings

CONTRACT

FoodTracking (food tracking system.so

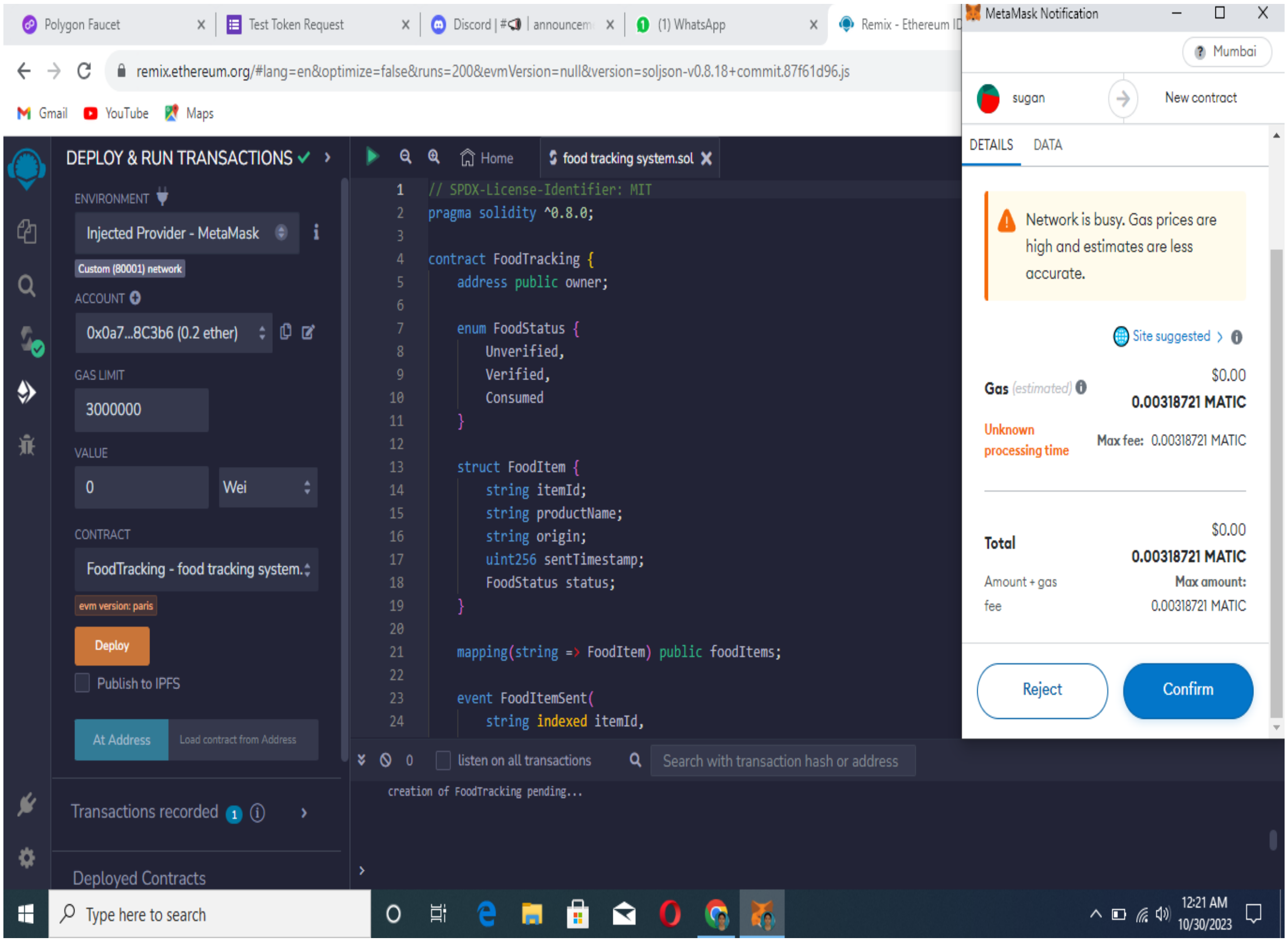
```
        emit FoodItemVerified(itemId);
    }

    function consumeFoodItem(
        string memory itemId
    ) external onlyUnconsumed(itemId) {
        foodItems[itemId].status = FoodStatus.Consumed;

        emit FoodItemConsumed(itemId);
    }

    function getFoodItemDetails(
        string memory itemId
    )
        external
        view
        returns (string memory, string memory, uint256, FoodStatus)
    {
        FoodItem memory item = foodItems[itemId];
        return (item.productName, item.origin, item.sentTimestamp, item.status);
    }
}
```

```
Type the library name to see available commands.
```



ENVIRONMENT: Injected Provider - MetaMask

Custom (80001) network

ACCOUNT: 0x0a7...8C3b6 (0.2 ether)

GAS LIMIT: 3000000

VALUE: 0 Wei

CONTRACT: FoodTracking - food tracking system.

evm version: paris

□ Publish to IPFS

```
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;

contract FoodTracking {
    address public owner;

    enum FoodStatus {
        Unverified,
        Verified,
        Consumed
    }

    struct FoodItem {
        string itemId;
        string productName;
        string origin;
        uint256 sentTimestamp;
        FoodStatus status;
    }

    mapping(string => FoodItem) public foodItems;

    event FoodItemSent(
        string indexed itemId,
```

```
creation of FoodTracking pending...
```

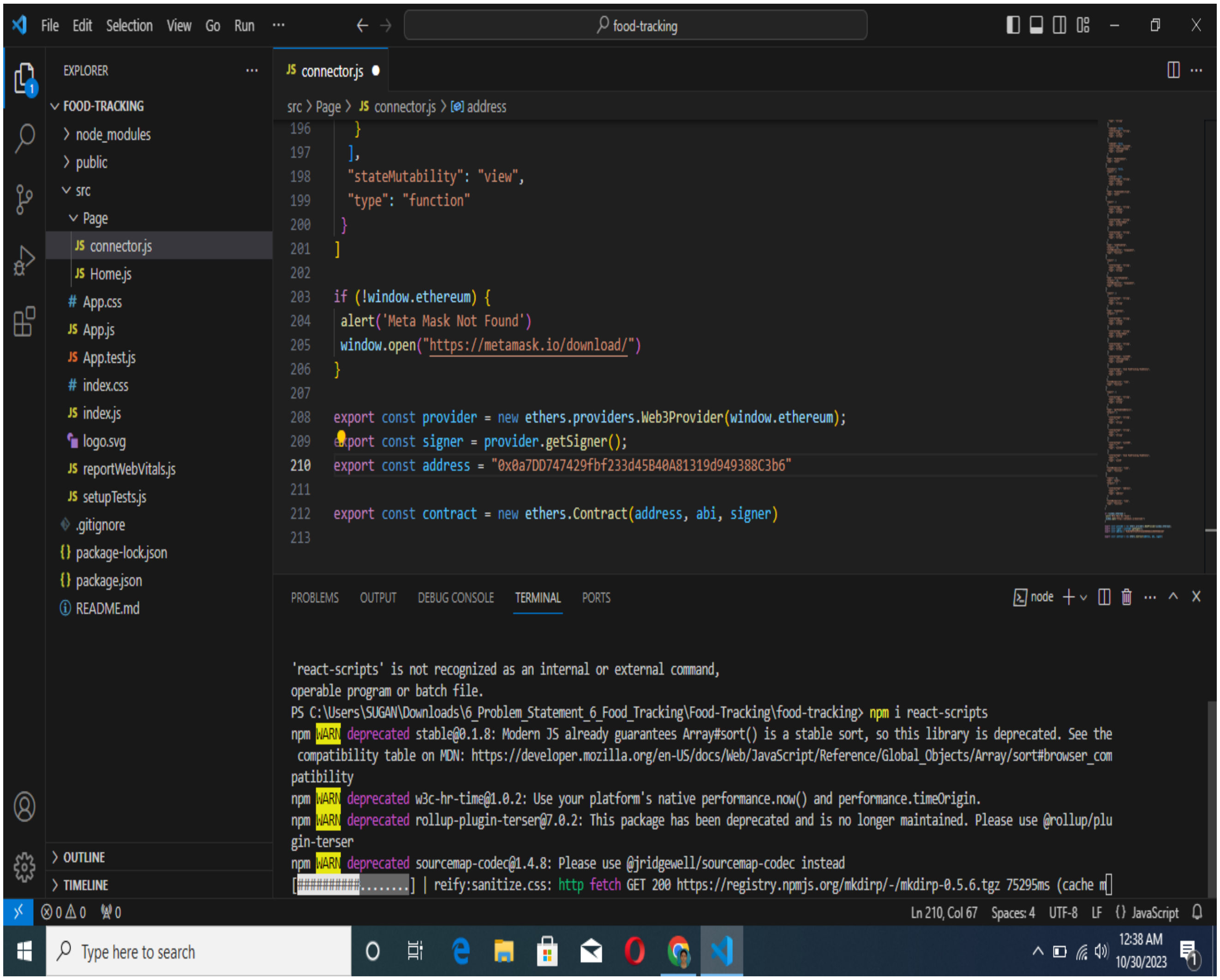
Network is busy. Gas prices are high and estimates are less accurate.

Gas (estimated) $0.00 0.00318721 MATIC

Unknown processing time Max fee: 0.00318721 MATIC

Total $0.00 0.00318721 MATIC

Amount + gas fee Max amount: 0.00318721 MATIC



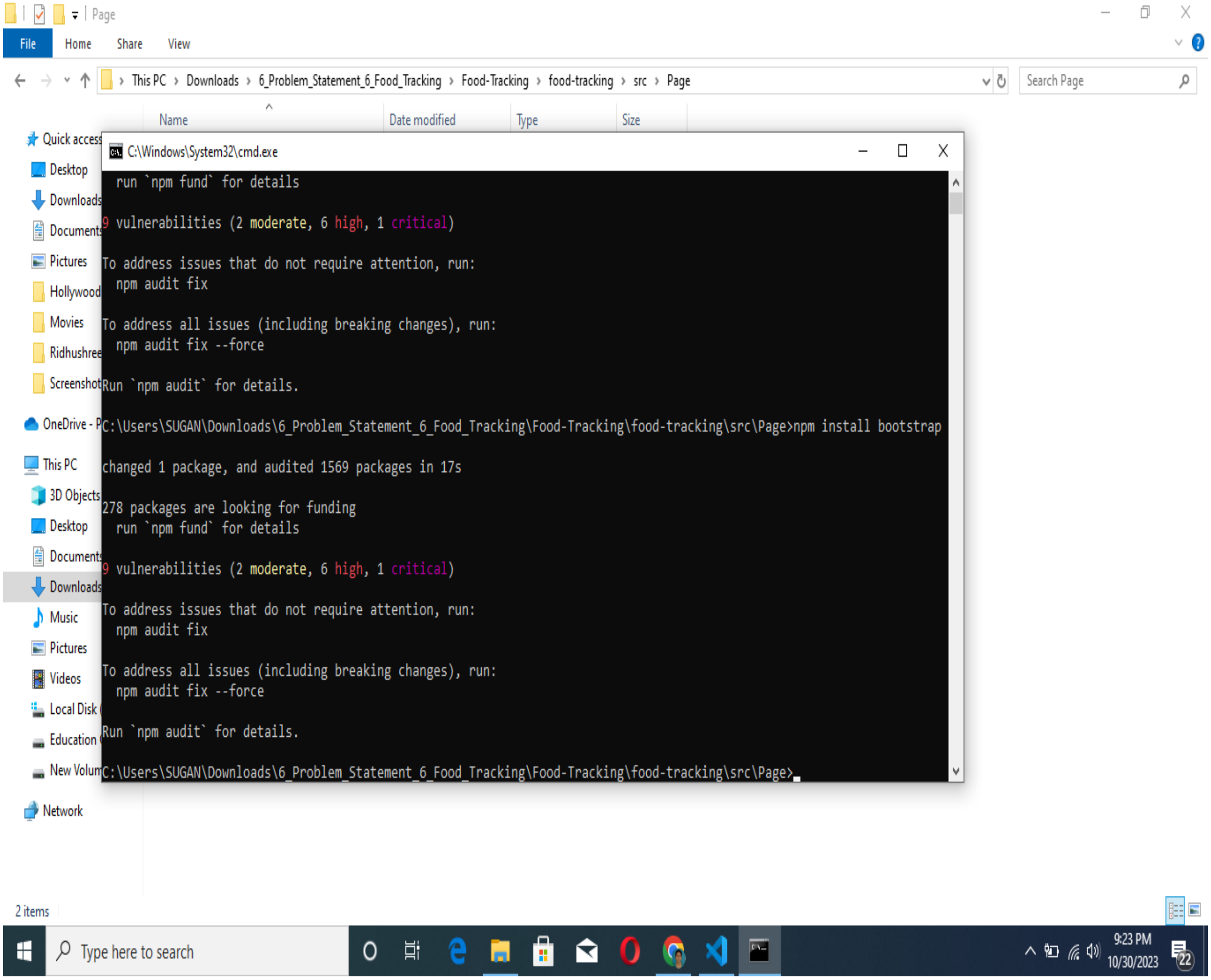
```
      }
    ],
    "stateMutability": "view",
    "type": "function"
  }
]

if (!window.ethereum) {
  alert('Meta Mask Not Found')
  window.open("https://metamask.io/download/")
}

export const provider = new ethers.providers.Web3Provider(window.ethereum);
export const signer = provider.getSigner();
export const address = "0x0a7DD747429fbf233d45B40A81319d949388C3b6"

export const contract = new ethers.Contract(address, abi, signer)
```

```
'react-scripts' is not recognized as an internal or external command,
operable program or batch file.
PS C:\Users\SUGAN\Downloads\6_Problem_Statement_6_Food_Tracking\Food-Tracking\food-tracking> npm i react-scripts
npm WARN deprecated stable@0.1.8: Modern JS already guarantees Array#sort() is a stable sort, so this library is deprecated. See the
 compatibility table on MDN: https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Global_Objects/Array/sort#browser_com
patibility
npm WARN deprecated w3c-hr-time@1.0.2: Use your platform's native performance.now() and performance.timeOrigin.
npm WARN deprecated rollup-plugin-terser@7.0.2: This package has been deprecated and is no longer maintained. Please use @rollup/plu
gin-terser
npm WARN deprecated sourcemap-codec@1.4.8: Please use @jridgewell/sourcemap-codec instead
[##########........] | reify:sanitize.css: http fetch GET 200 https://registry.npmjs.org/mkdirp/-/mkdirp-0.5.6.tgz 75295ms (cache m
```



```
  run `npm fund` for details

9 vulnerabilities (2 moderate, 6 high, 1 critical)

To address issues that do not require attention, run:
  npm audit fix

To address all issues (including breaking changes), run:
  npm audit fix --force

Run `npm audit` for details.

C:\Users\SUGAN\Downloads\6_Problem_Statement_6_Food_Tracking\Food-Tracking\food-tracking\src\Page>npm install bootstrap

changed 1 package, and audited 1569 packages in 17s

278 packages are looking for funding
  run `npm fund` for details

9 vulnerabilities (2 moderate, 6 high, 1 critical)

To address issues that do not require attention, run:
  npm audit fix

To address all issues (including breaking changes), run:
  npm audit fix --force

Run `npm audit` for details.

C:\Users\SUGAN\Downloads\6_Problem_Statement_6_Food_Tracking\Food-Tracking\food-tracking\src\Page>
```