FOOD TRACKING SYSTEM

A PROJECT REPORT

Submitted by

TEAM ID: NM2023TMID11502

VISHNUP	**622620104032**
JASWANTHKUMARP	**622620104006**
SATHYAB	**622620104024**
SWETHAS	**622620104029**

In partial fulfillment for the award of the degree

BACHELOR OF ENGINEERING

IN

COMPUTER SCIENCE AND ENGINEERING

SHREENIVASA ENGINEERING COLLEGE

TABLE OF CONTENT

1. INTRODUCTION

PROJECTOVERVIEW:

The food industry is undergoing a significant transformation, with increasingdemandfortransparencyandtraceabilitythroughout the supply chain.Foodsafety,authenticity, andsustainability areparamountconcerns for consumers,regulators,andproducersalike.Blockchaintechnology,withits decentralizedandimmutableledger,offersapromisingsolutionto addresstheseconcerns.Thispaperpresentsanovelapproachto implementingafoodtrackingsystemusingsmartcontractsonthe Ethereumblockchain.

The proposed system leverages Ethereum's smart contract capabilitiestocreateatransparentandsecureplatformfortrackingfood products from farm to fork. Each food item is assigned a unique digital identity,anditsjourneythroughthesupplychainisrecordedonthe blockchain.Thisdigitalledgerensuresdataintegrityandenablesreal-time accesstocriticalinformationsuchasorigin,processing,andtransportation details

PURPOSE:

The throughout the supply chain. By leveraging blockchain technology,specificallyEthereum'ssmartcontractcapabilities,your project aims to Provide a transparent and immutable ledger system for trackingfoodproductsfromtheirorigintotheendconsumer.Enable detailedtrackingoffooditems,ensuringthatconsumershaveaccessto criticalinformationabouttheproductstheyconsume.

Empower consumers to make more informed choices about theproductstheypurchase,supportingsustainabilityeffortsinthefoodindustry. In summary, the purpose of our project is to revolutionize thefood industryby leveragingblockchain technology tocreatea secure,transparent,andtraceablefoodtrackingsystem.Thissystemwillnotonl y addresscurrentindustryconcernsbutalsopavethewayformore sustainableandaccountablepracticesinthefuture.

2. LITERATURESURVEY

Existingproblem:

Anexistingproblemprojectcanaddressisthelackoftransparency andtraceabilityinthefoodsupplychain.Currently,consumersoftenhave limitedaccesstodetailedinformationaboutthejourneyoftheirfood productsfromfarmtotable.Thislackoftransparencycanleadto concernsaboutfoodsafety,authenticity,andsustainability.Instancesof foodborneillnessesorrecallsfurtherhighlighttheneedforamorerobust trackingsystem.

ByimplementingafoodtrackingsystemwithEthereum’ssmartcontracts, youcanprovideasolutiontothisproblem.Thistechnology allowsforthecreationofadecentralizedandimmutableledger,ensuring that criticalinformationabouteachfooditem'sorigin,processing,andtransportationissecurelyrecordedandaccessibleinreal-time.This addressestheexistingproblembysignificantlyenhancingtransparency andtraceabilitythroughoutthefoodsupplychain.

References

Iflookingforareferencetosupportyourprojectonimplementinga foodtrackingsystemwithEthereumsmartcontracts,youmightconsider citingarelevant academicpaperorareputable sourcerelatedto blockchaintechnology,foodsupplychains,orsmartcontracts.

Nakamoto, S.(2008). Bitcoin: A Peer-to-Peer Electronic Cash System.Retrievedfromhttps://bitcoin.org/bitcoin.pdf

Pleasenotethatthisexampleisagenericreferencetotheoriginal Bitcoinwhitepaper,asIdon'thaveaccesstospecific,project-specific sources.Foryourproject,youshouldfindandcitesourcesthataredirectly relevanttoyourimplementationofafoodtrackingsystemwithEthereum smartcontracts.

Merkle,R.C.(1987).Adigitalsignaturebasedonaconventional encryption function. In Advances in Cryptology—CRYPTO ’87 (pp.369-378).Springer

ProblemStatementDefinition:

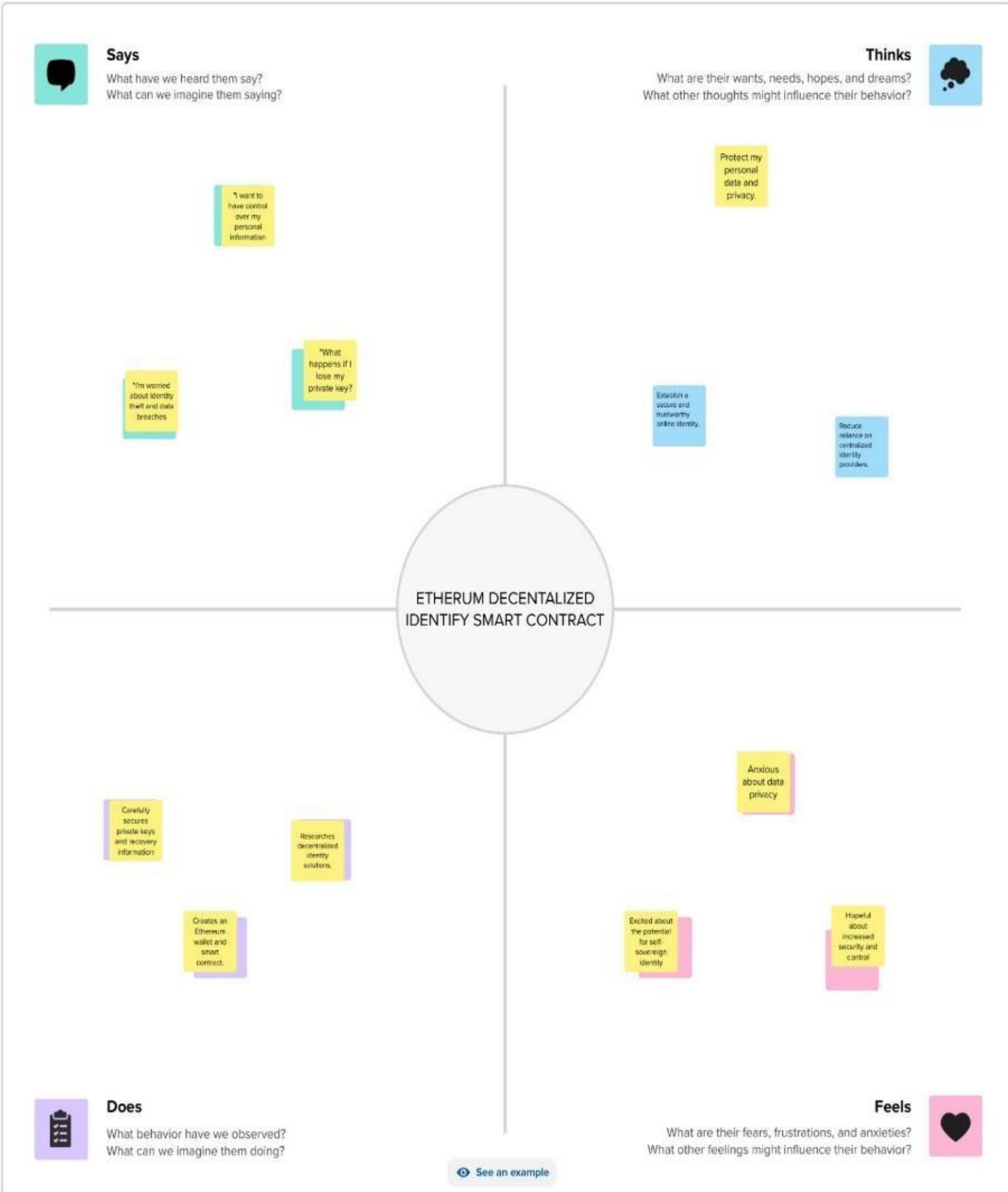
"In the current food industry landscape, concerns regarding transparency,traceability,food safety,authenticity, andsustainability persistentlychallengestakeholdersacrossthesupplychain.Consumers, regulators,andproducersalikefacedifficultiesinaccessingaccurateand real-timeinformationabouttheorigin,processing,andtransportation of foodproducts.Thislackoftransparencynotonlyhindersconsumertrust butalsoposesriskstofoodsafetyandtheintegrityofthesupplychain

Toaddressthesecriticalconcerns,thisprojectaimstoimplementa foodtrackingsystemusingsmartcontractsontheEthereumblockchain. Theobjectiveistoestablishasecure,decentralized,andtransparent platformthatassignsuniquedigitalidentitiestofooditems,enablingthe recordingoftheirjourneyfromfarmtofork.Byleveragingblockchain technology,theproposedsystemintendstorevolutionizethefoodindustry byproviding stakeholderswith real-time accessto trustworthyand immutabledata,ultimatelyenhancingtrust,safety,andsustainabilityinthe foodsupplychain."

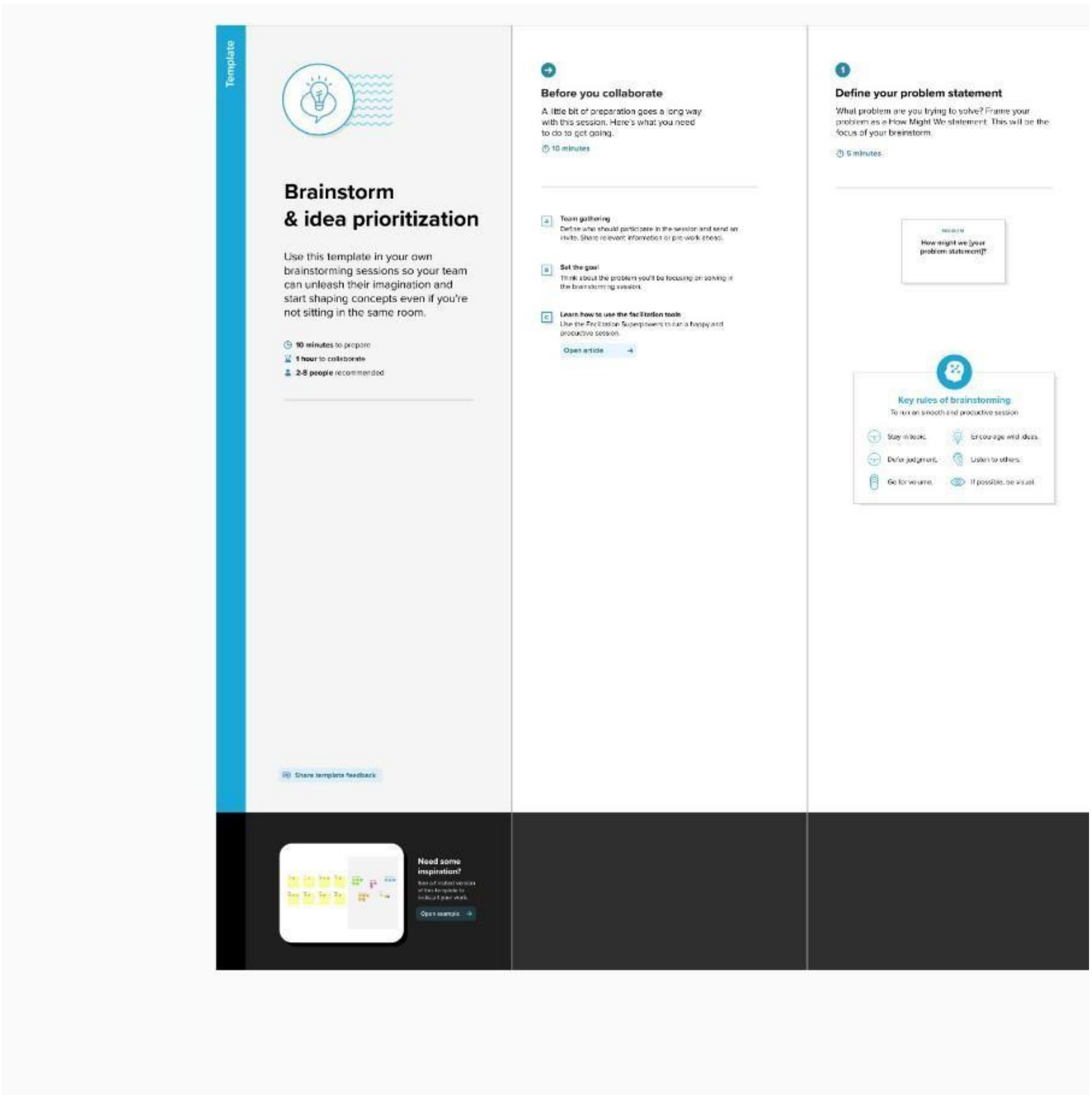
Thisproblemstatementsuccinctlyoutlinesthecurrentchallengesin thefoodindustryandclearlyarticulatesthegoalsandobjectivesofyour project. It provides a solid foundationfor the development and implementationofyourfoodtrackingsystem.

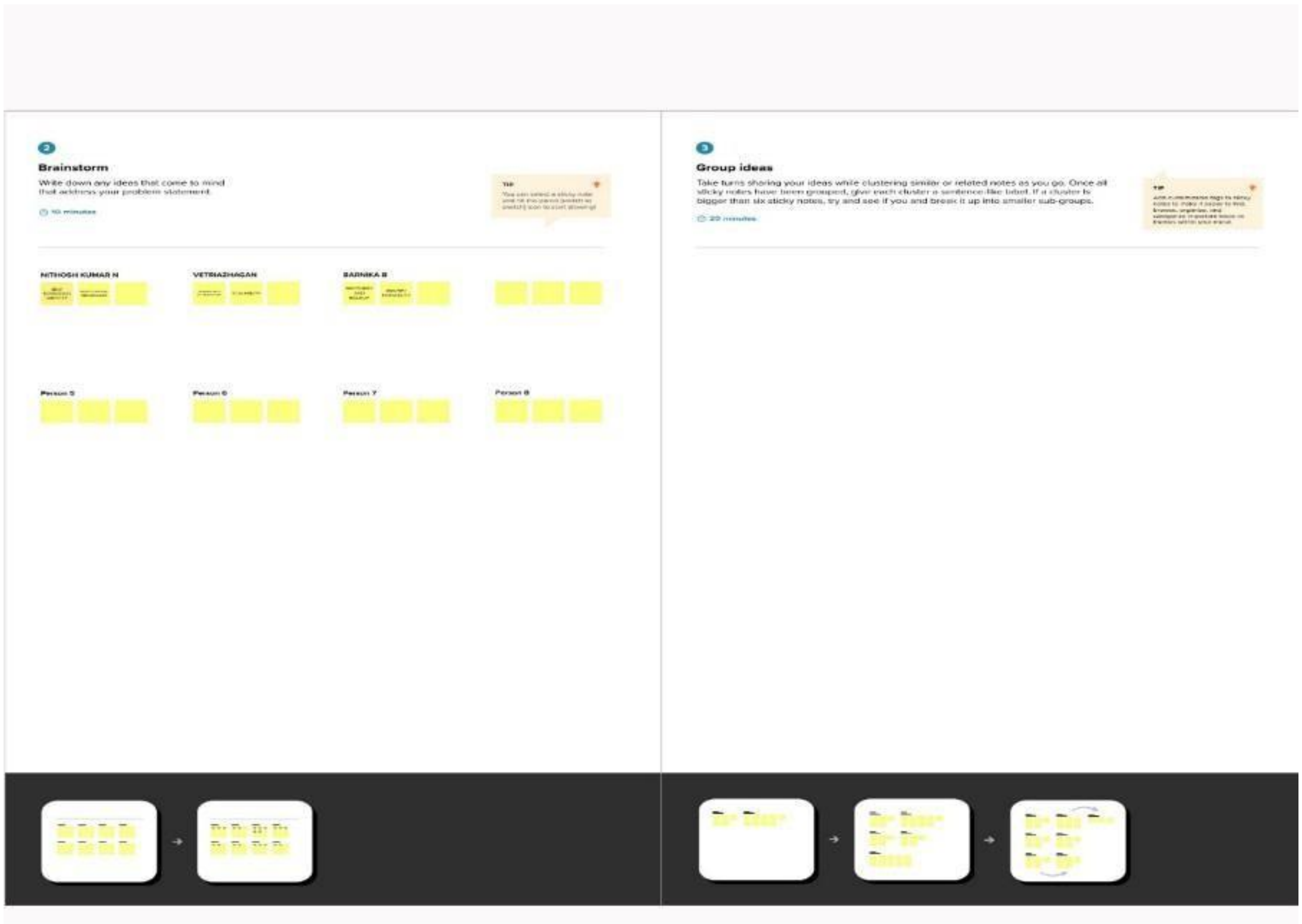
3. IDEATION&PROPOSEDSOLUTION:

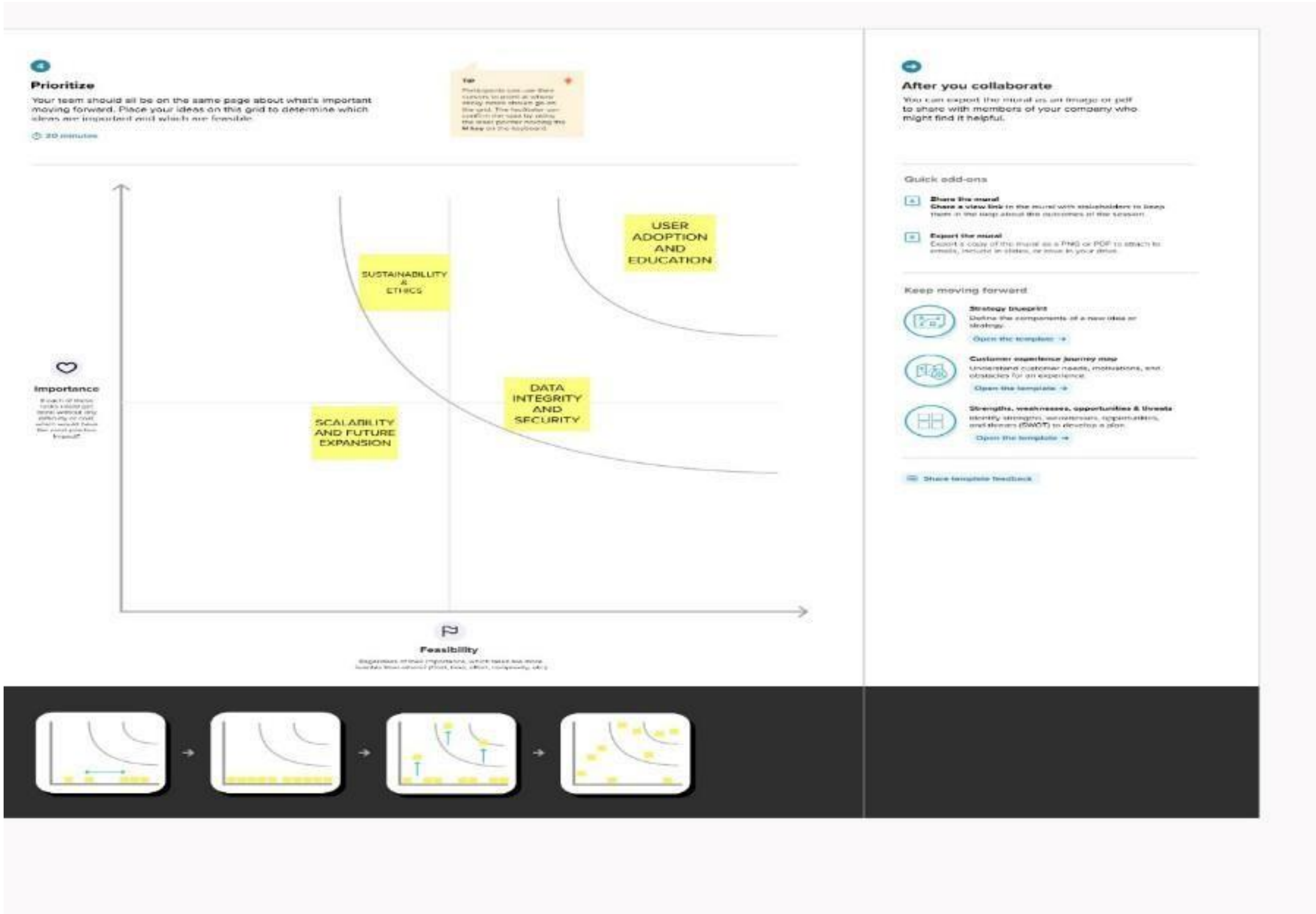
EmpathyMapCanvas:



Ideation&Brainstorming


Template
Brainstorm
& idea prioritization
Use this template in your own brainstorming sessions so your team can unleash their imagination and start shaping concepts even if you're not sitting in the same room.
10 minutes to prepare
1 hour to collaborate
2-8 people recommended
Share template feedback
Need some inspiration?
Open example
Before you collaborate
A little bit of preparation goes a long way with this session. Here's what you need to do to get going.
10 minutes
Team gathering
Define who should participate in the session and send an invite. Share relevant information or pre-work ahead.
Set the goal
Think about the problem you'll be focusing on solving in the brainstorming session.
Learn how to use the facilitation tools
Use the Facilitation Superpowers to run a happy and productive session.
Open article
Define your problem statement
What problem are you trying to solve? Frame your problem as a How Might We statement. This will be the focus of your brainstorm.
5 minutes
How might we [your problem statement]?
Key rules of brainstorming
To run an smooth and productive session
Stay in topic.
Encourage wild ideas.
Defer judgment.
Listen to others.
Go for volume.
If possible, be visual.


2
Brainstorm
Write down any ideas that come to mind that address your problem statement.
10 minutes
NITHOSH KUMAR N
VETRIAZHAGAN
BARNIKA B
Person 5
Person 6
Person 7
Person 8
3
Group ideas
Take turns sharing your ideas while clustering similar or related notes as you go. Once all sticky notes have been grouped, give each cluster a sentence-like label. If a cluster is bigger than six sticky notes, try and see if you and break it up into smaller sub-groups.
20 minutes


Prioritize
Your team should all be on the same page about what's important moving forward. Place your ideas on this grid to determine which
USER ADOPTION AND EDUCATION
SUSTAINABILLITY & ETHICS
Importance
DATA INTEGRITY AND SECURITY
SCALABILITY AND FUTURE EXPANSION
Feasibility
After you collaborate
You can export the mural as an image or pdf to share with members of your company who might find it helpful.
Quick add-ons
Keep moving forward

4. REQUIREMENTANALYSIS:

Functionalrequirement:

1.UserRegistrationandAuthentication:

- Thesystemshouldallowuserstocreateaccountswithaunique usernameandpassword.
- Usersshouldbeabletologinsecurely.

2. InputFoodInformation:

- Usersshouldbeabletoinputdetailsaboutthefoodtheyconsume, includingname,quantity,servingsize,andpreparationmethod.

3. NutritionalInformationRetrieval:

- Thesystemshouldretrieveanddisplaynutritionalinformationforco mmonfoodsfromareliabledatabaseorAPI.

4. TrackConsumptionHistory:

- Usersshouldbeabletoviewtheirhistoricalfoodconsumptionrec ords.

5. SetDietaryGoals:

- Users should be able to set personalized dietary goals, such as calorieintake,macronutrientdistribution,orspecificdietarypreferences(e. g.,vegetarian,gluten-free)

6. GenerateReports:

- Thesystemshouldgeneratereportssummarizingdaily,weekly,or monthly food intake, including calorie count, macronutrient distribution, andotherrelevantmetrics.

4.2.Non-FunctionalRequirements:

1. Performance:

   - Thesystemshouldrespondtouserinteractions(inputtingfood, viewingreports)within2secondsundernormalloadconditions.

2. Security:

   - User data,includingpersonalinformationandfoodconsumption history, should be stored securely and protected against unauthorized accessorbreaches.

3. Usability:

   - Theuserinterfaceshouldbeintuitive,easytonavigate,andaccessibl etousersofvaryingtechnologicalproficiency.

4. Scalability:

   - Thesystemshouldbedesignedtohandleapotentiallylargeuserbase and an increasing volume of food data without significant performancedegradation.

5. ReliabilityandAvailability:

• Thesystemshouldbeavailable24/7withamaximumdowntimeof 99.9%peryear. Itshouldalsohavebackup andrecoveryprocedures in place.

6. Compatibility:

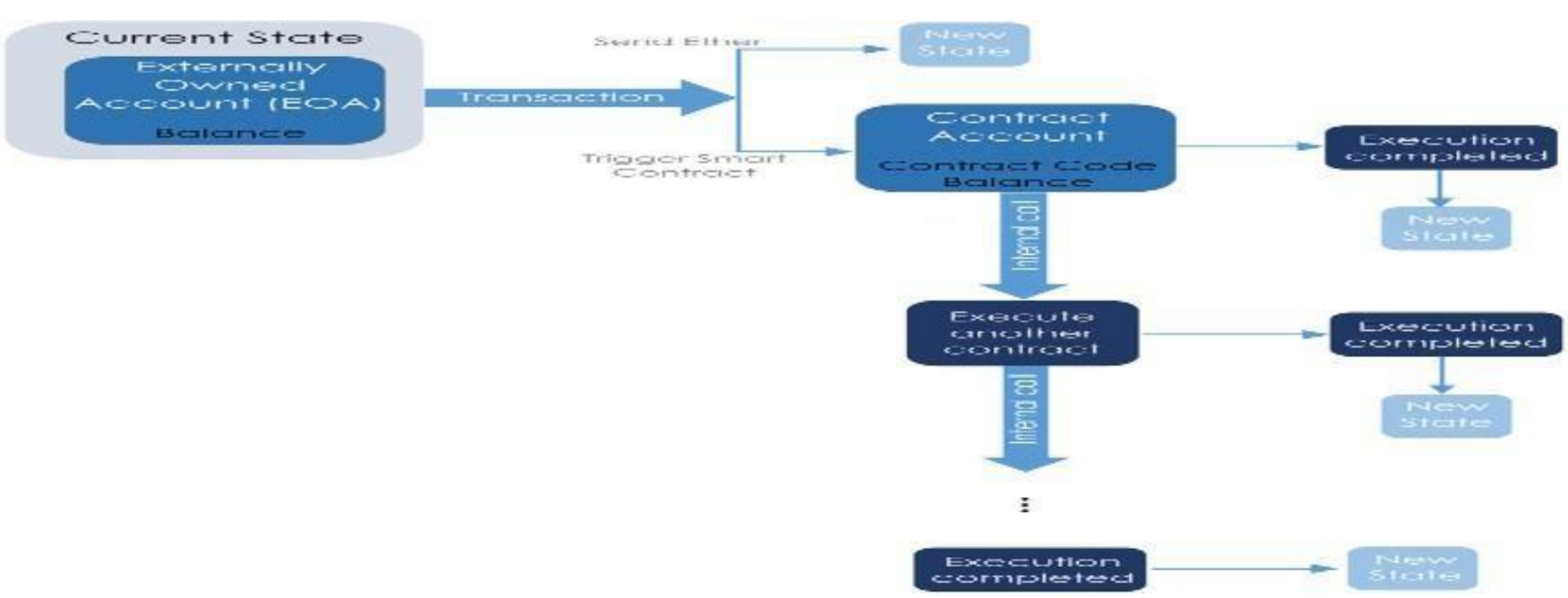
• The system should be compatible with various devices and platforms(e.g., web browsers, mobile apps) toensure aseamless user experience.

7. DataPrivacyandCompliance:

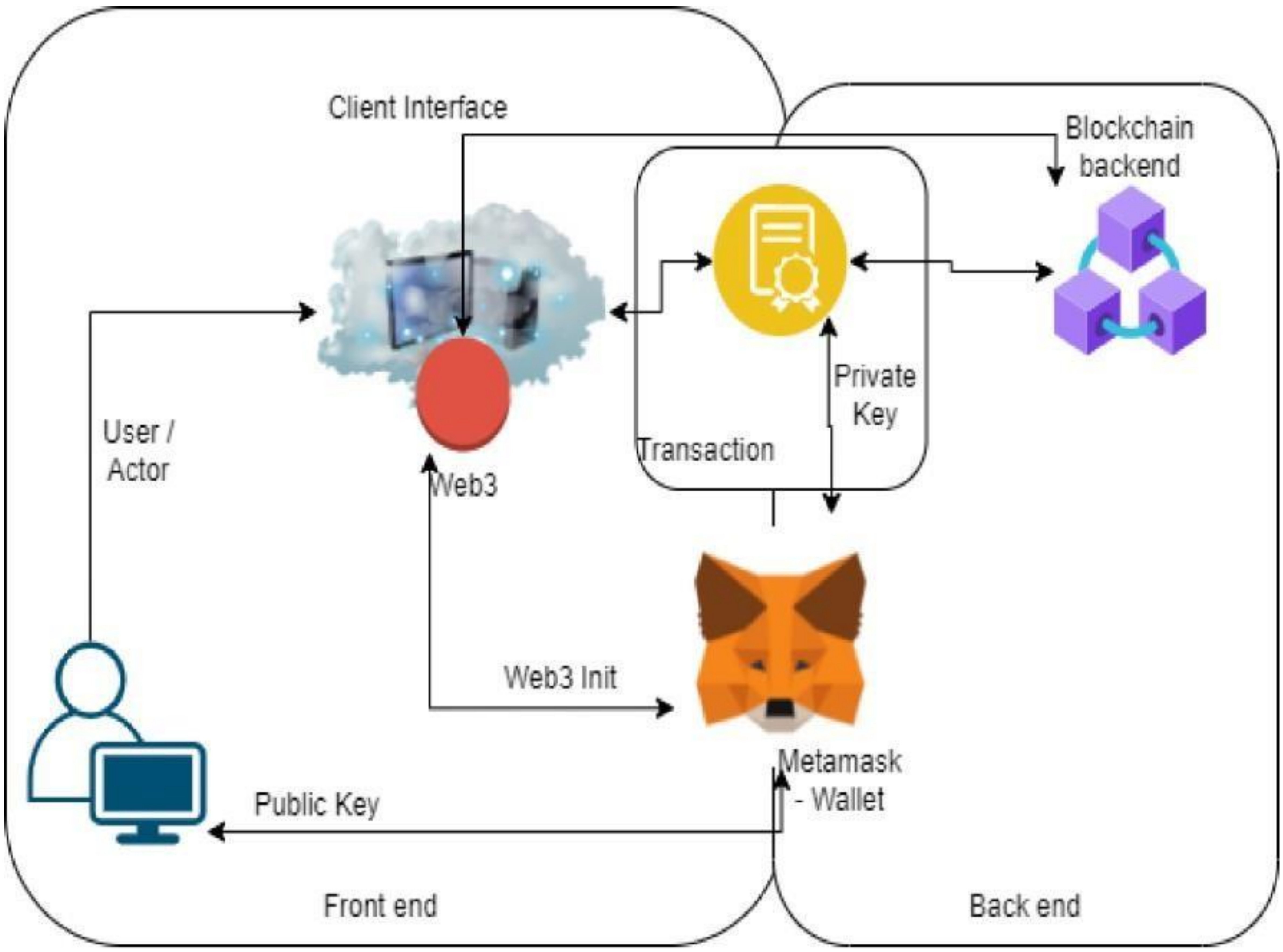
• Thesystemshouldcomplywithdataprotectionregulations(e.g.,G DPR)

5. PROJECTDESIGN:

DataFlowDiagrams&UserStories:

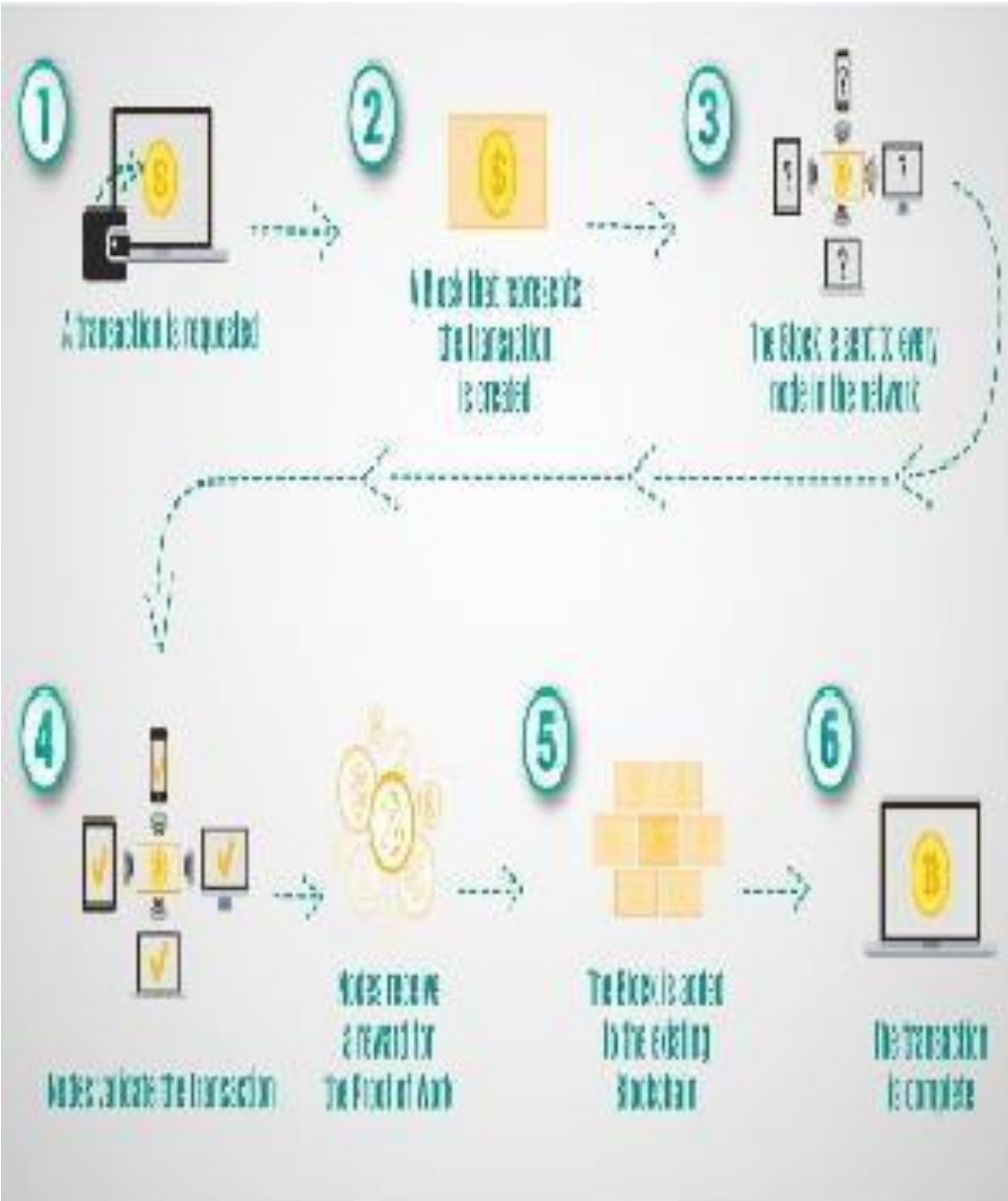


SolutionArchitecture:



6. PROJECTPLANNING:

TechnicalArchitecture:



7. CODING&SOLUTIONING:

SPDX-License-Identifier:Thisspecifiesthelicenseunderwhichthecodeisdistributed.Inthiscase,it'susingtheMITlicense.

Pragmasolidity^0.8.0;:Thisindicatesthatthecontractisdesignedto workwithSolidityversion0.8.0orhigher.Itensurescompatibilitywiththe specifiedversion.

CONTRACTSTRUCTURE:

ContractFoodTracking{...}Thisdefinesthecontractnamedfood Tracking.Allthefunctions,variables,andstructuresrelatedtofood trackingaredefinedwithinthiscontract.

STATEVARIABLE:

Addresspublicowner;:ThisvariablestorestheEthereumaddressoftheowner(deployer)ofthesmartcontract.Itisdeclaredaspublic, meaningitcanbeaccessedexternally.

MAPPING:

Mapping(string=>FoodItem)publicfoodItems;:Thiscreatesa mappingnamedfoodItemsthatassociatesauniquestring(theitemID)withaFood Itemstruct.Thismappingallowsforefficientretrievaloffooditemdetails.

SENDFOODITEMFUNCTION:

FunctionsendFoodItem(..)externalonlyOwner{...}:Thisfunctionallowstheownertosendinformationaboutafooditemtotheblockchain. ItcreatesanewFoodItemstructandstoresitinthefoodItemsmapping.

VERIFYFOODITEMFUNCTION:

FunctionverifyFoodItem(..)externalonlyOwner{...}:Thisfunction allowstheownertoverifyafooditem,changingitsstatusfromUnverified to Verified

CONSUMEFOODITEMFUNCTION:

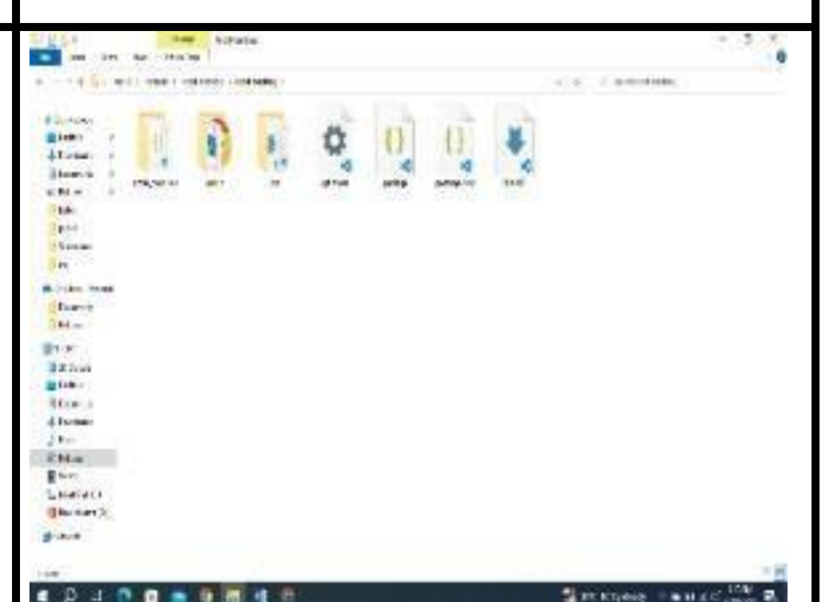
FunctionconsumeFoodItem(...)externalonlyUnconsumed(itemId){... }:ThisfunctionallowstheownertomarkafooditemasConsumed, provideditihasalreadybeenverified.

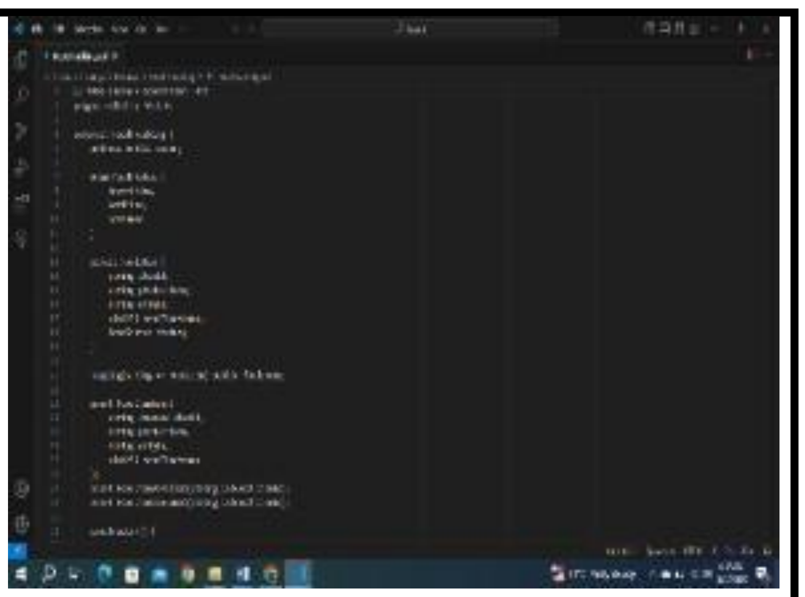
Feature1:

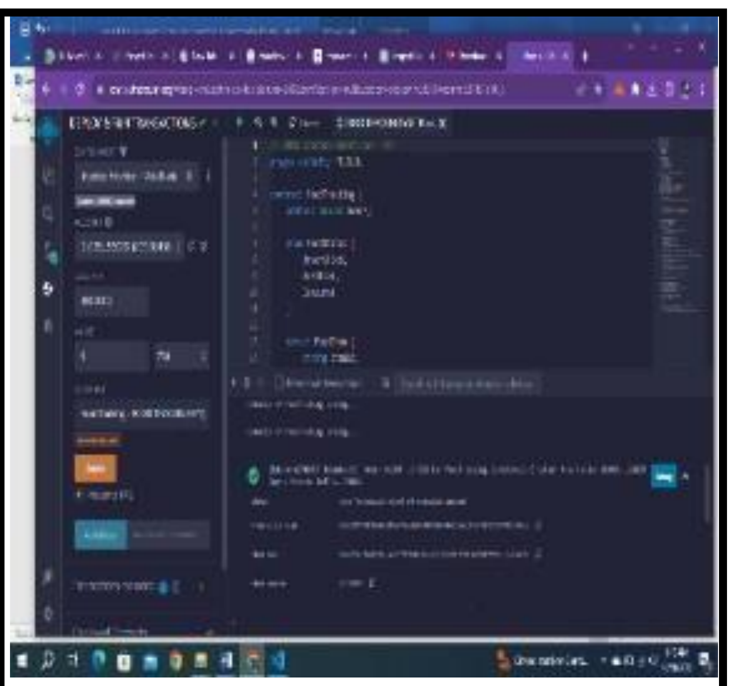
ThisSoliditysmartcontract establishesaframeworkfortrackingfood items,ensuringthatonlytheowner canperformcertainoperations,and that food items progress through defined states (Unverified, Verified, Consumed). Remember to conduct thorough testing and consider potential security vulnerabilities before deploying it on the Ethereum blockchain.

8. PERFORMANCETESTING:

PerformaceMetrics:

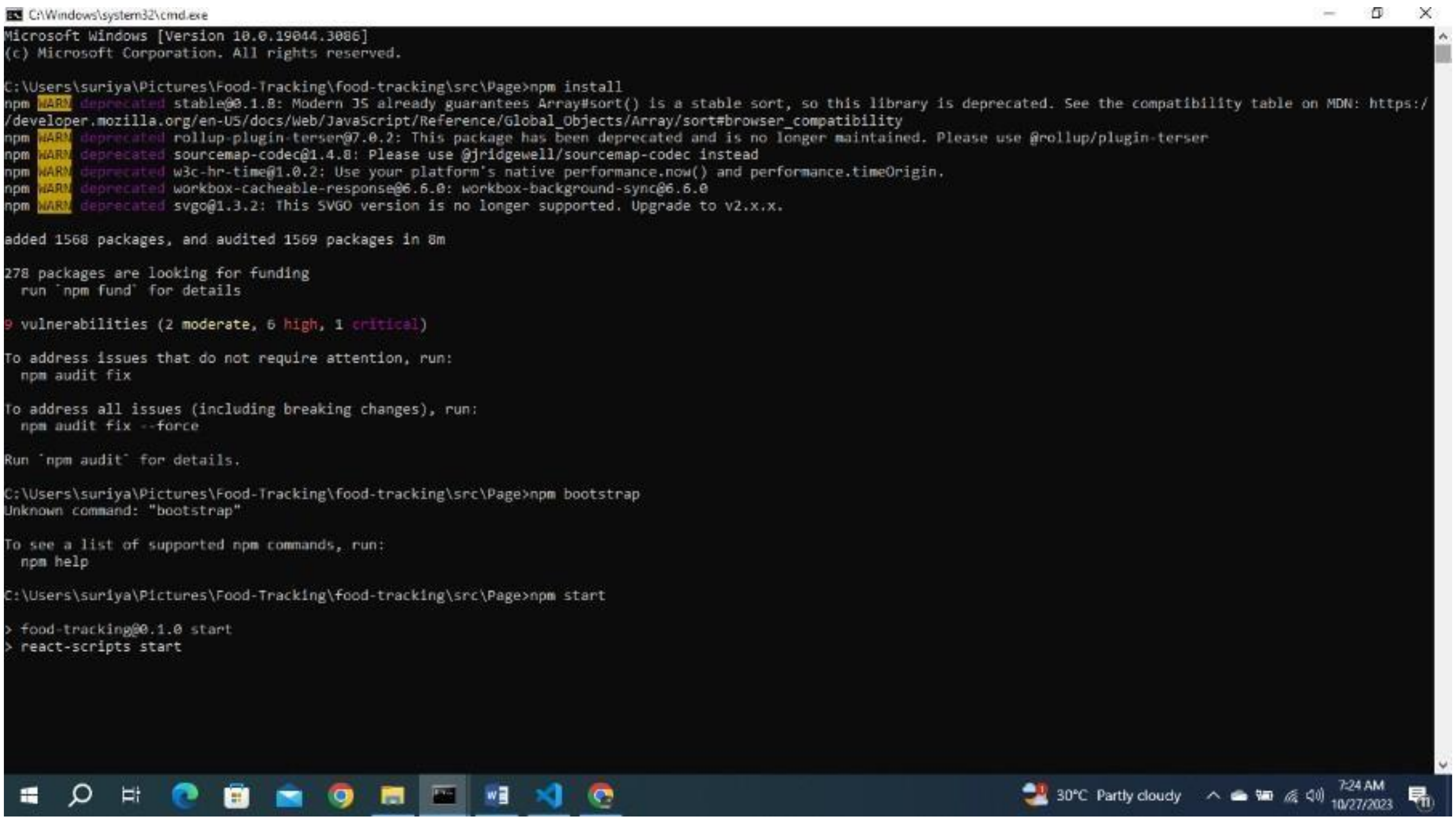
S.No.	Parameter	Values	Screenshot
1.	Information gathering	SetupallthePrerequisite:	

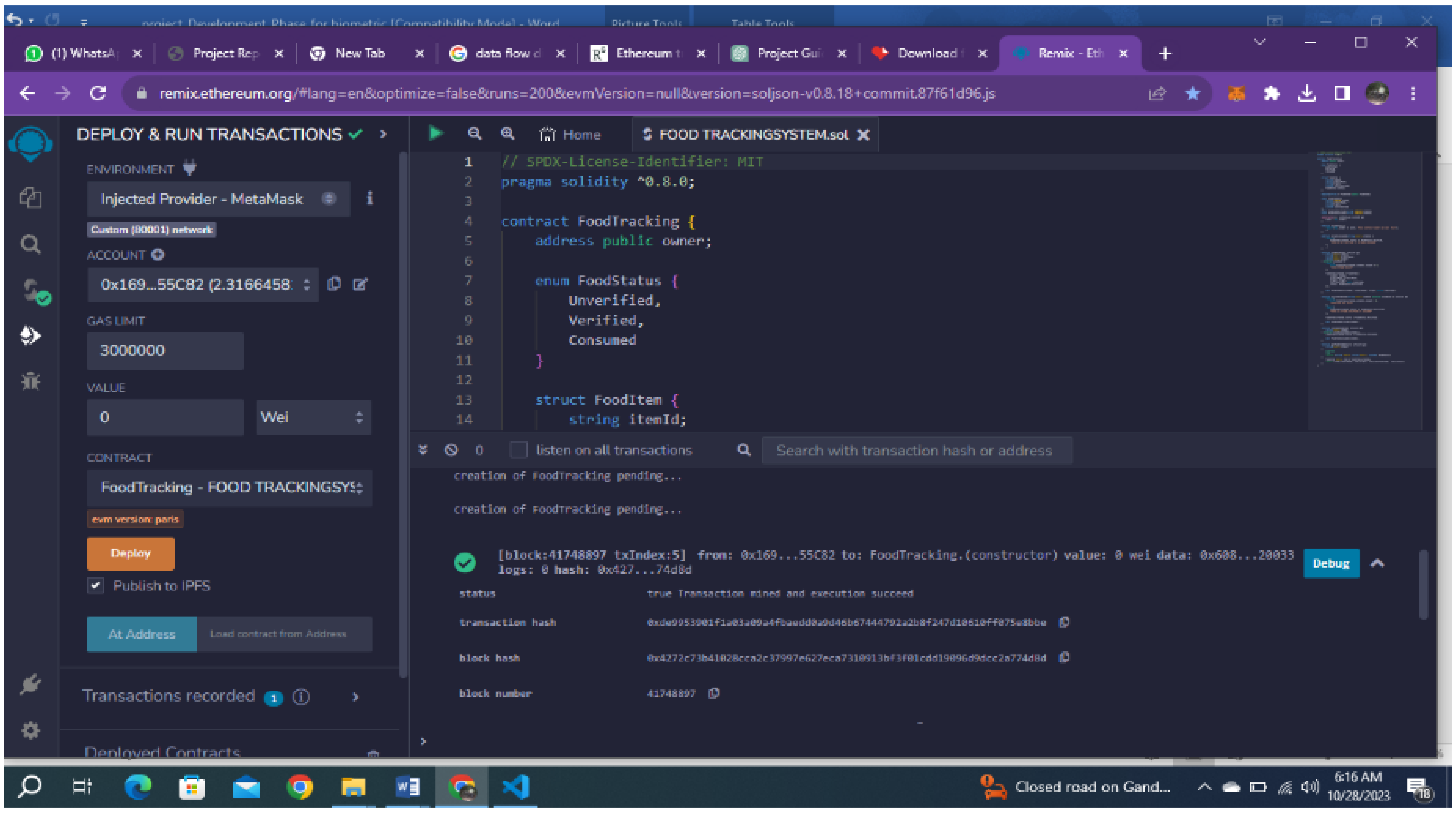
2.	Extractthezip files	Opentovscode	

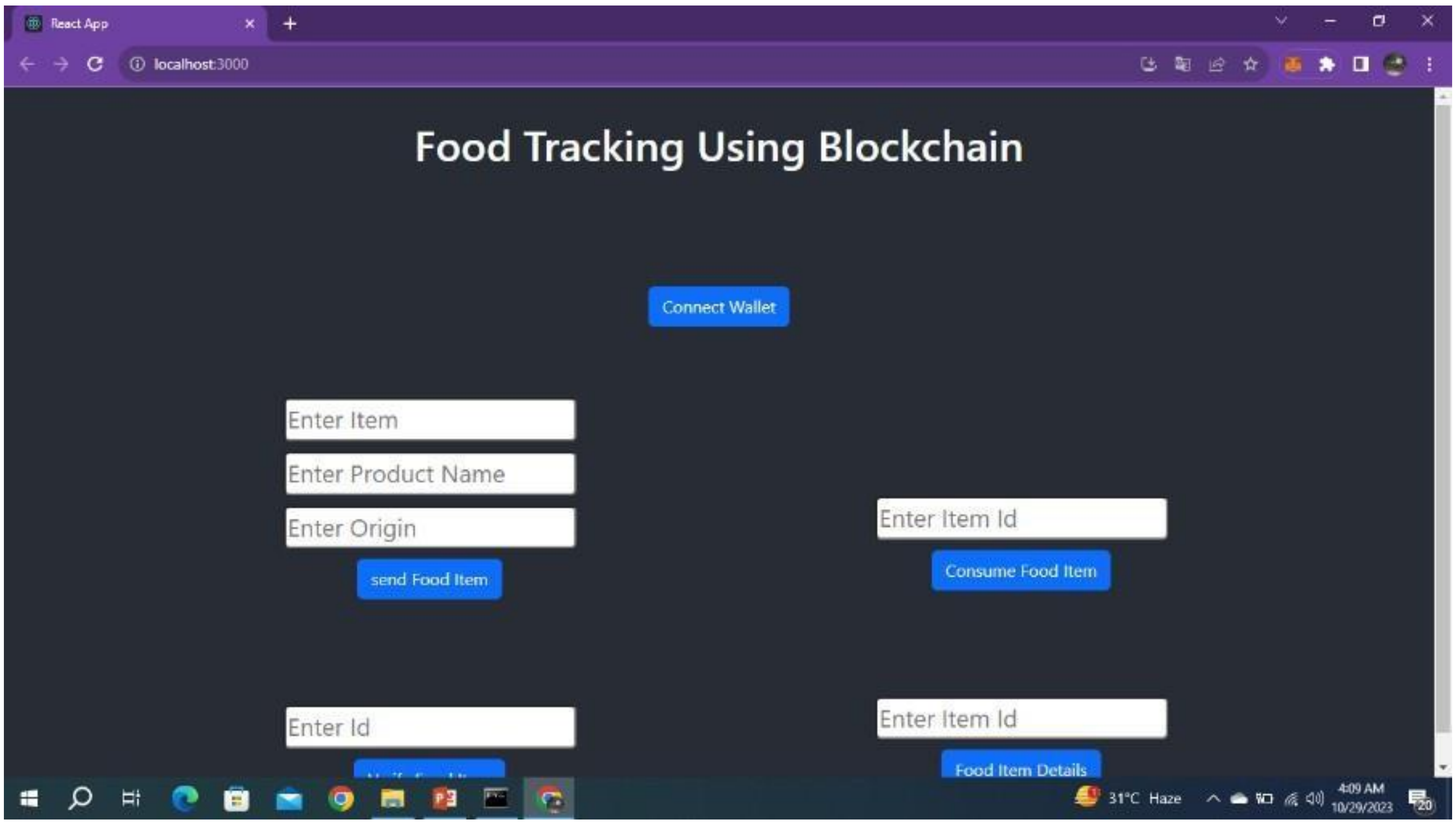
3.	RemixIde platform explorting	Deploythesmartcontract code Deployandrunthe transaction.Byselectingthe environment-injectthe MetaMask.	
4	Openfile explorer	Opentheextractedfileand click onthe folder. Opensrc,andsearchfor utiles. Opencmdenter commands1.npm install 2. npminstallbootstrap 3. npmstart	
5	{LOCALHOSTIP ADDRESS	copytheaddressandopenit tochromesoyoucanseethe front end of your project.	

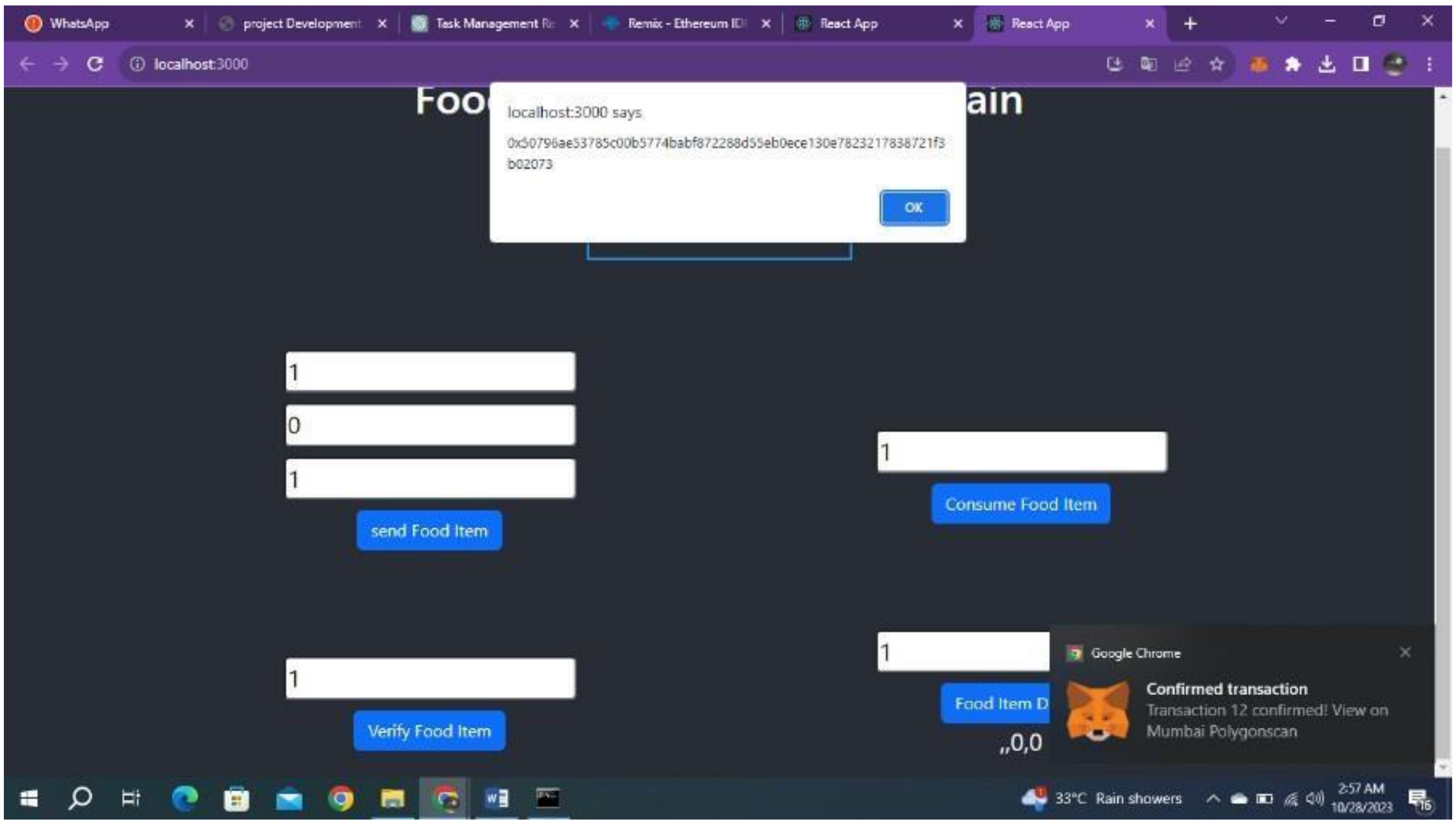
9. RESULTS:

OUTPUTSCREENSHOTS:




remix.ethereum.org/#lang=en&optimize=false&runs=200&evmVersion=null&version=soljson-v0.8.18+commit.87f61d96.js
DEPLOY & RUN TRANSACTIONS
ENVIRONMENT
Injected Provider - MetaMask
ACCOUNT
0x169...55C82 (2.3166458
GAS LIMIT
3000000
VALUE
0
Wei
CONTRACT
FoodTracking - FOOD TRACKINGSYS
Deploy
Publish to IPFS
At Address
Transactions recorded
Deployed Contracts
FOOD TRACKINGSYSTEM.sol
// SPDX-License-Identifier: MIT
pragma solidity ^0.8.0;
contract FoodTracking {
address public owner;
enum FoodStatus {
Unverified,
Verified,
Consumed
}
struct FoodItem {
string itemId;
listen on all transactions
Search with transaction hash or address
creation of FoodTracking pending...
creation of FoodTracking pending...
[block:41748897 txIndex:5] from: 0x169...55C82 to: FoodTracking.(constructor) value: 0 wei data: 0x608...20033
logs: 0 hash: 0x427...74d8d
Debug
status
true Transaction mined and execution succeed
transaction hash
block hash
block number
41748897
Closed road on Gand...
6:16 AM
10/28/2023


React App
localhost:3000
Food Tracking Using Blockchain
Connect Wallet
Enter Item
Enter Product Name
Enter Origin
send Food Item
Enter Item Id
Consume Food Item
Enter Id
Enter Item Id
Food Item Details
31°C Haze
4:09 AM
10/29/2023



10. ADVANTAGE&DISADVANTAGE:

ADVANTAGE:

RegulatoryComplianceandAccountability:

Implementing a food tracking system using smart contracts on the Ethereum blockchain ensures a high level of compliance with industryandgovernmentalregulations.Theimmutablenatureof blockchainrecordsprovidesanunalterablehistoryofeachfooditem'sjourney,which canserveasareliableaudittrail.

EnhancedConsumerTrustandConfidence:

By implementing a food tracking system using smart contracts on the Ethereum blockchain, you provide consumers with unprecedentedtransparencyintotheoriginandjourneyof theirfood products. This increased transparency leads to greater trust in thesupplychainandconfidenceinthesafetyandauthenticityofthefood theyconsume.This,inturn,canleadtohighercustomersatisfaction andloyalty,benefitingbothproducersandretailersinthelongrun.

Real-timeMonitoringandAlerts:

The system allows for real-time monitoring of food items throughout their journey. This enables immediate responses to any deviations or anomalies in the supply chain, reducing the risk of spoilageorcontamination.

FraudPrevention:

Theimmutablenatureofblockchainensuresthatoncedatais recorded,itcannotbetamperedwith.Thispreventsfraudulent activitiessuchascounterfeitingormisrepresentingtheoriginoffoodproducts.

DISADVANTAGE:

CostofImplementationandMaintenance:

Building and maintaining a blockchain-based system can be expensive, particularly in terms of development, infrastructure, and ongoing maintenance. This cost may be a significant barrier forsmallerproducersorbusinesseswithlimitedresources.

ScalabilityChallenges:

Dependingonthescaleofyourprojectandthenumberof transactionsinvolved,youmayencounterscalabilityissueswiththe Ethereumblockchain.Hightransactionvolumescouldleadtoslower processingtimesandincreasedgasfees.

IntegrationComplexity:

Integrating the blockchain system with existing systems, databases, and software used in the food supply chain may be complex and require significant customization. This could lead to disruptionsinexistingoperations.

PrivacyConcerns:

Blockchaintechnologyprovidestransparency,italsomeansthatsensitive informationmaybevisibletoallparticipantsinthenetwork.

Ensuringprivacyforcertaintypesofdata(e.g.,proprietaryrecipesortradesecrets)maybechalenging.

11.CONCLUSION:

Inarapidlyevolvingfoodindustrylandscape,theintegrationof blockchaintechnologyandsmartcontractspresentsatransformative solutiontoaddresscriticalconcernssurroundingtransparency,traceability,safety,authenticity,andsustainability.Theimplementation ofafoodtrackingsystemontheEthereumblockchain,asproposedin this project, signifies a paradigm shift towards a more secure, transparent,andaccountablesupplychain.

By leveraging the decentralized and immutable ledger capabilitiesofblockchain,coupledwiththeautomationandtrustmechanismsofferedbysmartcontracts,thissystemprovidesarobust platformforrecordingandverifyingthejourneyoffoodproductsfrom farmtofork.Theuniquedigitalidentitiesassignedtoeachitemensure that critical information, including origin, processing, and transportationdetails, is securely stored and readily accessible in real-time.

12. FUTURESCOPE:

The implementation of afood trackingsystem using Ethereum smartcontractslaysthefoundationforahostoffutureadvancements and opportunities in the food industry. As technology continues to evolve, there are several potential areas of expansion and enhancementforthisproject

TokenizationandIncentiveMechanisms:Considerintroducinga token-based system to incentivize active participation in the supply chain.Tokenscouldbeusedtorewardproducers,logisticsproviders, and consumers for their contributions to data accuracy and transparency.

Augmented Reality (AR) and Virtual Reality (VR) Integration:DevelopAR/VRapplicationsthatallowconsumerstovirtuallyexp lore thejourneyoffoodproducts,providinganimmersiveandeducational experienceaboutthesupplychain.

13. APPENDIX:

SOURCECODE:

```
//SPDX-License-Identifier:MIT
pragmasolidity^0.8.0;

 contractFoodTracking{
   addresspublicowner;

  enumFoodStatus{
    Unverified,
    Verified,
    Consumed
  }

  structFoodItem{
```

```
    stringitemId;
    stringproductName;
    stringorigin;
    uint256sentTimestamp;
    FoodStatusstatus;
}

mapping(string=>FoodItem)publicfoodItems;eventFoodItemSent

(
    stringindexeditemId,
    stringproductName,
    stringorigin,
    uint256sentTimestamp
);
eventFoodItemVerified(stringindexeditemId);eventFoodItemCon
sumed(stringindexeditemId);

constructor(){
    owner=msg.sender;
}

modifieronlyOwner(){
    require(msg.sender==owner,"Onlycontractownercancallthis");
    _;
}

modifieronlyUnconsumed(stringmemoryitemId){r
    equire(
        foodItems[itemId].status==FoodStatus.Verified,"Itemisnotveri
        fiedoralreadyconsumed"
    );
    _;
}

functionsendFoodItem(
    stringmemoryitemId,
    stringmemoryproductName,
    stringmemoryorigin
)externalonlyOwner{ require(
        bytes(foodItems[itemId].itemId).length==0,"Itemalreadyexists
        "
    );

    foodItems[itemId]=FoodItem({it
        emId:itemId,
        productName:productName,
```

```
      origin:origin,
      sentTimestamp:block.timestamp,status:Food
      Status.Unverified
    });

    emitFoodItemSent(itemId,productName,origin,block.timestamp);
  }

  functionverifyFoodItem(stringmemoryitemId)externalonlyOwner{r
    equire(
      bytes(foodItems[itemId].itemId).length>0,"Ite
      mdoesnotexist"
    );
    require(
      foodItems[itemId].status==FoodStatus.Unverified,"Itemi
      salreadyverifiedorconsumed"
    );

    foodItems[itemId].status=FoodStatus.Verified;

    emitFoodItemVerified(itemId);
  }

  functionconsumeFoodItem(string
    memoryitemId
  )externalonlyUnconsumed(itemId){foodItems[itemId].status=FoodStatus.Consumed
    ;

    emitFoodItemConsumed(itemId);
  }

  functiongetFoodItemDetails(stringmemoryit
    emId
  )
    external
    view
    returns(stringmemory,stringmemory,uint256,FoodStatus)
  {
    FoodItemmemoryitem=foodItems[itemId];return(item.productName,item.origin,item.sentTimestamp,it
    em.status);
  }
}
```

GitHub&ProjectDemoLink: