National College of Ireland

BSc Honours In Computing
Cyber Security
Academic Year i.e. 2020/2021

Gaurav Ramesh Savanur
X17114357
X17114357@student.ncirl.ie

Improving the Privacy on the Inventory Supply-chains Using Zero-Knowledge Proofs

Technical Report

Contents

Executive Summary

The technical report is a summary of the project which comprises of the introduction of the project where we are introducing the topic of supply-chain attacks and to the introduction of blockchain technology which is the underlining technology behind cryptocurrency and we provide a summary of the attacks on supply chains dependent organizations today.

The project focuses on improving the privacy of inventory supply-chains using blockchain technology and zero knowledge proofs. The target market for this application is Suppliers and Producers of goods. This project is an inventory management web application which uses Ethereum Distributed Ledger to store the data rather than a traditional database. The privacy has been enhanced in the project through the utilization of a cryptographic protocol called Zero Knowledge Proofs which hides sensitive user information while users securely login into the web application.

Introduction

1.1. Background

The reason for the interest in the project would be my interest in privacy and blockchain technology. I was interested in the security area within the blockchain field and the field of supply-chain was an interesting area as security vulnerabilities had been exploited during the NotPetya virus and Equifax hacks where users sensitive data was exploited in these hacks due to trust issues and potentials security exploits that lead to deletion of data from its company-owned databases.(Ng, A., 2020)

1.2. Aims

The project aimed to solve the privacy issues within Inventory Management Systems by using Etherium smart contracts to host the public data on the public blockchains which will provide an immutability feature for the data. Additionally, I utilised to introduce a cryptographic protocol called Zero-Knowledge Proofs (Limited, C., 2020) which enhances privacy by hiding sensitive data from the public blockchain and the firebase database.

The use of Ethereum blockchain and solidity smart contracts is the introduction to immutability to the data stored on the blockchains. Zero-knowledge proof has been used to enhance the privacy using the Zokrates module which provided zero-knowledge proof construct within Ethereum blockchains.

1.3. Structure

The document provides the requirements analysis that would be involved in the project. The technical report will go through parts of the requirements specification. The documentation provides a brief overview of the project with its key components involving requirement analysis specifications which include use case diagrams, use case description, functional, non-functional requirements, and UI mockups which would provide a detailed analysis of the technical requirements needed for the development of the project.

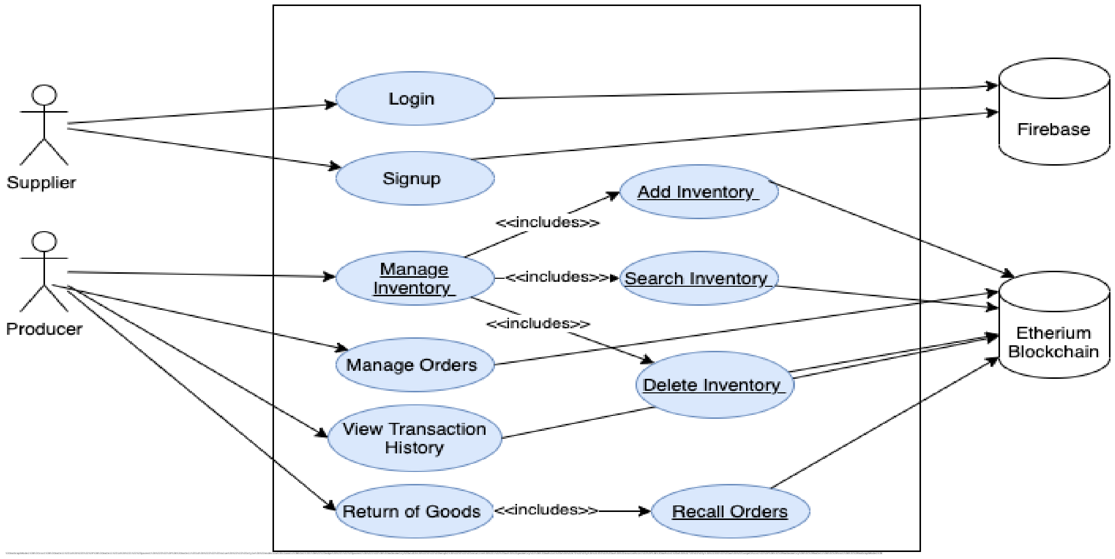
2.0 System

2.1. Requirements

The requirements for the supply chain tracker project provide a brief overview of how the project is being implemented. The requirements provides a detailed explanation of the functional and non-functional requirements that are involved in the development of the project.

2.1.1. Functional Requirements

2.1.1.1. Use Case Diagram



2.1.1.2. Requirement 1 Login

Use Case Diagram Description

Use Case ID:	UC_LOGIN_1		
Use Case Name:	log in		
Created By:	Gaurav Savanur	Last Updated By:	
Date Created	23 Nov 2020	Last Updated Date	21/11/20

Description:	This Use case allows the user to login into the system to access the relevant functions according to the user's role. The roles are Inventory production and return of goods. To login to the system, the user must enter the user ids and password. Upon successful login, the system will display the relevant user home page.
Actors: Primary Secondary	 User None

Triggers:	The use case is triggered while the user visits the website.
Pre-conditions:	The user has a valid account.
Post-Conditions:	The system displays the relevant home page
Normal Flow:	1. The user enters the user log id and password. 2. The user submits the user id and password. 3. The system validates the user id and password. 4. The system verifies the user id and password. 5. The system displays the relevant home page.
Alternate Flows:	1a The user does not have a user 1 The user triggers to register as a new user 1b the user closes the program 1 The system exits 3a Missing user id and or password 1 The system prompts for user id and password 2 The use case resumes at normal flow 4 4a Invalid user id and or password 1 The system displays “Invalid Credentials” 2 The system prompts for user id and password 3 These case resumes at normal flow 1.
Exceptions:	6a connection to the database fails The System displays the message “Service not available” 1 The System returns to normal flow 1
Includes:	None
Extends:	register new user
Priority:	High
Special Conditions:	
Assumptions:	
Notes and Issues:	

2.1.1.3. Requirement 2 SIGNUP

Use Case ID:	UC_SIGNUP		
Use Case Name:	SIGNUP User Use Case		
Created By:	Gaurav Savanur	Last Updated By:	
Date Created	23, Nov 2020	Last Updated Date	21/11/20

Description:	This Use case Signup a new user on the System. The system checks passwords for the latest NIST password requirements
Actors: Primary Secondary	 Supplier Producer
Triggers:	The use-case is triggered from the
Prthenditions:	the username is not on the system.
Post-Conditions:	The user details are registered in the database. The system displays the Login Page.
Normal Flow:	1. The user enters User Name, E-Mail, Password, Confirm Password. 2. The user submits the information. 3. The System Validates the submitted data. 4. The System Verifies the user is not already on the System. 5. The System adds the user to the database. 6. The System displays the user login page.
Alternate Flows:	3a The system checks for username 1.If no user name the system asks the user to enter a user name. 2. Use case resumes at Normal Flow 1 3b The system checks for an email. 1 . System checks for a valid email format is ked to reenter the email. 2. Usecase resumes to Normal Flow 1. 3c the system checks that the passwords match. 1 if incorrect send message Passwords do not match. 2 use case resumes at normal flow 1. 3d the system checks that the password is at least 6 characters. 1 if incorrect sends message Password is not long enough. 2 use case resumes at normal flow 1. 3e the system checks that the password contains a number. 1 if incorrect send message Passwords should contain a Number. 2 use case resumes at normal flow 1. 3f The new username is already on the system.

	1. the system sends a message Sorry could not process that request
Exceptions:	5a connection to the database fails The System displays the message “Service not available”. 1 The System waits for the user to exit.
Includes:	None
Extends:	Log in
Priority:	High
Special Conditions:	passwords match password is a minimum of 6 characters long the password contains a number
Assumptions:	
Notes and Issues:	

2.1.1.4. Requirement 3 Manage Orders Use Case

Use Case ID:	MANAGE_ORDERS_USE_CASE		
Use Case Name:	Manage Orders Use Case		
Created By:	Gaurav Savanur	Last Updated By:	
Date Created	13th Nov 2020	Last Updated Date	21/11/20

Description:	The use case checks for the validity of the use-case description where the use-case provides a detailed description of how the use-case would be implemented in the application.
Actors: Primary Secondary	 Supplier Producer
Triggers:	The use-case is triggered from the Log in use case
Pre-conditions:	The username is not on the system
Post-Conditions:	The user details are registered in the database The system displays the Login Page

Normal Flow:	1. The Producer can view a list of Inventories in the View Inventory Page. 2. The Producer can place orders for the inventory by staking etherium for the producer 3. The Ropsten Testnet validates the transaction and status of the application changes to Order Placed 4. The Supplier is notified the order is placed. The Supplier can place a delivery order to deliver the product to the Producer. 5. The System Validates the Order with the Ropsten testnet and the Order status changes stating that the order is delivered.
Alternate Flows:	
Exceptions:	5a connection to the database fails The System displays the message "Service not available" 1 The System wathe it's for the user to exit
Includes:	None
Extends:	Log in
Priority:	High
Special Conditions:	
Assumptions:	

2.1.1.5. Requirement 4 Inventory

Use Case ID:	MANAGE_INVENTORY_USE_CASE		
Use Case Name:	Manage Inventory Use Case		
Created By:	Gaurav Savanur	Last Updated By:	
Date Created	13 Nov 2020	Last Updated Date	21/11/20

Description:	The use case checks for the validity of the inventory use case where the use-case enables the supplier to add new inventory to the system.
Actors: Primary Secondary	Supplier None
Triggers:	The use case is triggered when the supplier user successfully Logins into the system.
Pre-conditions:	the username is on the system
Post-Conditions:	The user details are registered in the database

	The system displays the Inventory Page that displays the Create Product Functionality to the Supplier.
Normal Flow:	1. The Supplier enters the product name. 2. The Supplier enters the product type. 3. The Supplier enters the product category. 4. The Supplier enters the description of the products. 5. The Supplier enters the price of the items for every respective product on the system. 6. The Supplier enters the required Price for the product on the system. 7. The Supplier clicks on the submit button to add the new inventory to the system. 8. The System validates the required conditions and adds the required items onto the system.
Alternate Flow	1 b) a. Select the item from the inventory list and click on the cancel contract button. b.A dialog box appears that asks to cancel the contract. c.The system validates the contract and cancels the contract for the inventory item. 1c)a. Select the view inventory nav bar option within the web application. b.Search for the inventory id based on the list provided in the view inventory page c.Select the inventory you searched for within the web application and the web application provides a detailed description of the inventory.
Exceptions:	5a connection to the database fails. The System displays the message "Service not available" 1 The System waits for the user to exit
Includes:	None
Extends:	Log in
Priority:	High
Special Conditions:	
Assumptions:	
Notes and Issues:	

2.1.1.6. Requirement 5 Return of Goods

A description of the requirement and its priority. Describes how essential this requirement is to the overall system.

Use Case ID:	RETURN_OF_GOODS		
Use Case Name:	Return of Goods Use Case		
Created By:	Gaurav Savanur	Last Updated By:	
Date Created	30 Nov 2020	Last Updated Date	21/11/20

Description:	The use case checks for the validity of the Return of Goods Use-Case where the use-case is the extension of the inventory use-case that enables the supplier to recall products that were added as inventory onto the system.
Actors: Primary Secondary	 Supplier None
Triggers:	The use-case is triggered when the user successfully adds new inventory into the system.
Pre-conditions:	the username is not on the system
Post-Conditions:	The user details are registered in the database The system displays the Login Page
Normal Flow:	1. The Producer searches for the market the required product. 2. The Producer sells the required product that needs to be recalled from the Ethereum blockchain. 3. The system validates with the rinkby testnet to update the state of the product to recalled. 4. A notification is displayed that says the product has been recalled.
Alternate Flow	
Exceptions:	5a connection to the database fails The System displays the message “Service not available”. 1 The System waits for the user to exit.
Includes:	None
Extends:	Log in

Priority:	High
Special Conditions:	
Assumptions:	The supplier has entered the product into the inventory system.
Notes and Issues:	

2.1.1.7. Requirement 6 Transaction History

Use Case ID:	Transaction-History		
Use Case Name:	History of Order of Transactions		
Created By:	Gaurav Savanur	Last Updated By:	
Date Created	30 Nov 2020	Last Updated Date	7/12/2020

Description:	The use case checks for the validity of the Transaction History Use-Case where the use-case has the use-case enables the supplier to view the transaction history over the orders that were placed on the system.
Actors: Primary Secondary	Producer Supplier
Triggers:	The use-case is triggered when the user successfully added new product orders into the system.
Pre-conditions:	-
Post-Conditions:	The user details are registered in the database.
Normal Flow:	1. The Producer searches for the order placed on the orders page. 2. The system displays the transaction history and the time and occurrence of the events.
Alternate Flow	1.a)The Supplier recalls the product from the inventory page. b)The Supplier enters the recall comments with the description box. 2 a) The system displays the recalled products in the transaction history.
Exceptions:	5a connection to the database fails The System displays the message "Service not available". 1 The System waits for the user to exit.
Includes:	None

Extends:	Orders
Priority:	High
Special Conditions:	
Assumptions:	The producer has entered the order into the order system
Notes and Issues:	

2.1.2. Data Requirements

The Data Requirements are based on the sources that were sourced during the initial requirements gathering phase of the project. However, as it is a proof of concept for an inventory management system where zero-knowledge proofs are used to enhance privacy on the etherium blockchain and minimalistic data would be used however to provide robustness to the project I have sourced 1 data source that would be added at the later stage of the application to test and provide robustness to real-world data. The data sources are unstructured and have OPDL and MIT Licenses ensuring ethical requirements aren't the absolute concern during the development of the project. The data source sourced for the project is enlisted in the ethics document attached below in the appendix section of the document.

2.1.3. User Requirements

The user requirements would involve 2 components supplier and the producer. This Supplier would add new inventory to the system which gets recorded on the etherium blockchain. The producer can view the shipping requirements which could be approved to retrieve an order. The order would be placed on the Order's Page. The Order's Page would have the functionality to return goods which would thereby record on the blockchain with comments that the inventory has been recalled with reasons for withdrawal recorded on the blockchain.

2.1.4. Environmental Requirements

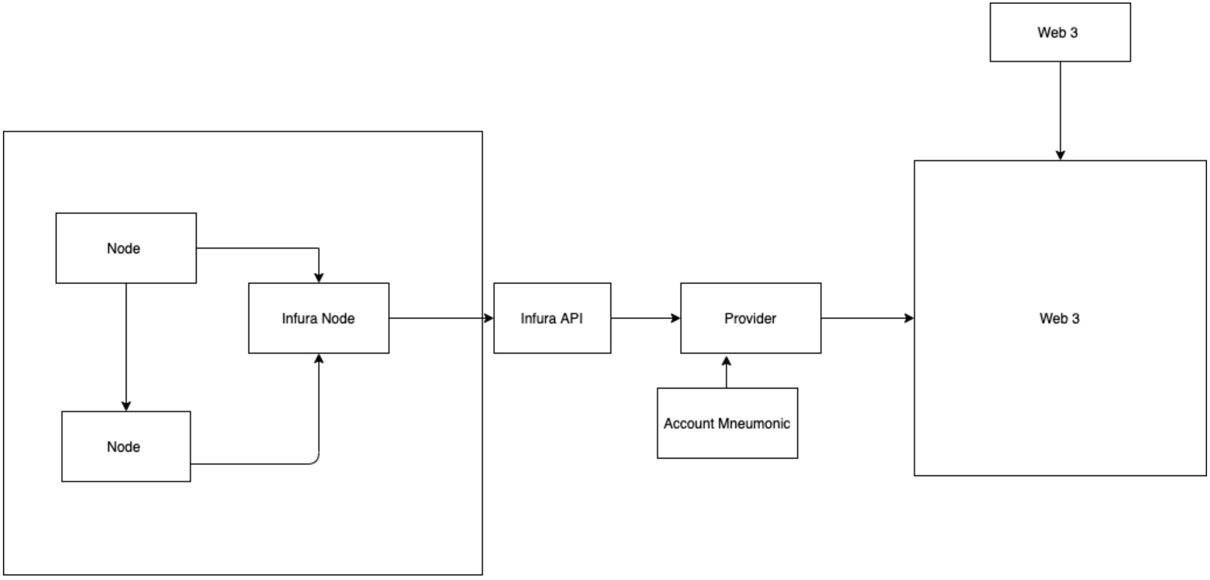
The Environmental Concerns aren't relevant to the project as the following project doesn't harm any environmental or energy laws. The application is a web application built rather than a physical application which implies cloud resources would be used to retrieve data. To host a Node Computational power would have been spent on the application thereby I decided to host the node using Infura a Consensys platform that hosts the etherium nodes. This ensures that these nodes wouldn't lead to a loss of energy resources while deploying smart-contracts to the rinkeby test net. (Ethereum API | IPFS API Gateway | ETH Nodes as a Service | Infura, 2020)

2.1.5. Usability Requirements

The usability requirements would be considered while building the project would include easy to follow design which would be implemented for the inventory management system where the user can follow the essential requirements that are involved in the building of the inventory application from adding inventory, supplier update, and return of goods page that would show the return of goods page on the web applications. Furthur A/B Testing would be done during the end of the testing phase ensuring UI is simple and easy to follow for a supplier and the producer.

2.2. Design & Architecture

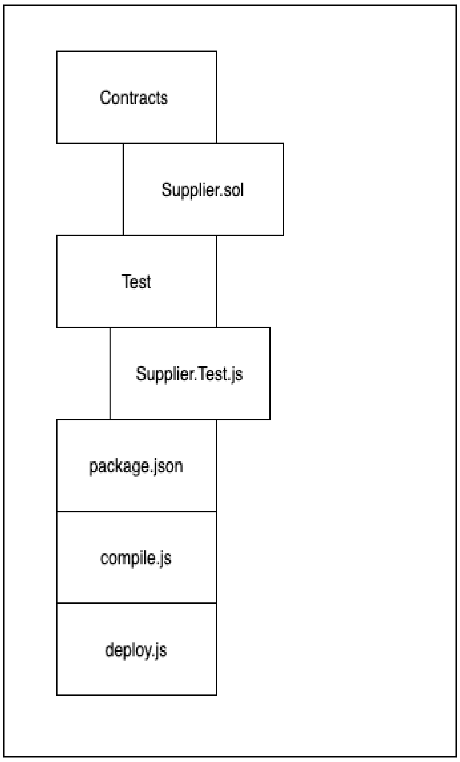
The design for the project involves the boilerplate code which is being used to deploy contracts to the rinkeby testnet. This framework is built using Javascript Framework called Node.js.The high-level design is described succinctly below on how the smart contracts built in Solidity and how it's being deployed to the rinkeby test net.
The following diagram shows the deployment process that is followed while building the environment which is being used to deploy smart-contracts to the rinkeby test net.



Infura Consensys Smart-Contract Deployment Framework Methodology

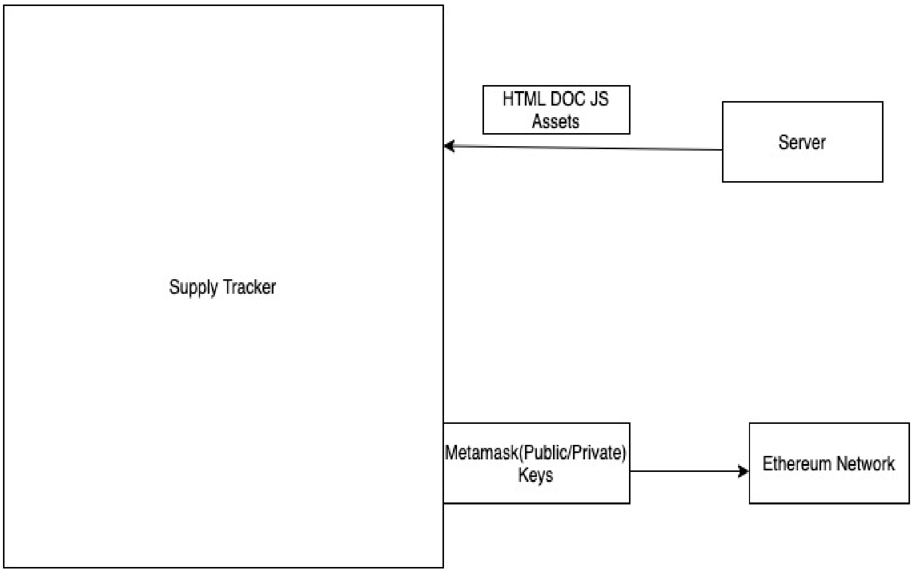
Secondly, The Inbox Project Directory is structured in a way where Separation of Concerns principles are applied when building testing and deploying the contracts. The following diagram below shows us the design of the project structure within the smart contract boiler-plate code that was used while building using the Node.js Framework. (Separation of Concerns in Software Design, 2020).

Smart-Contract Deployment Framework File Structure



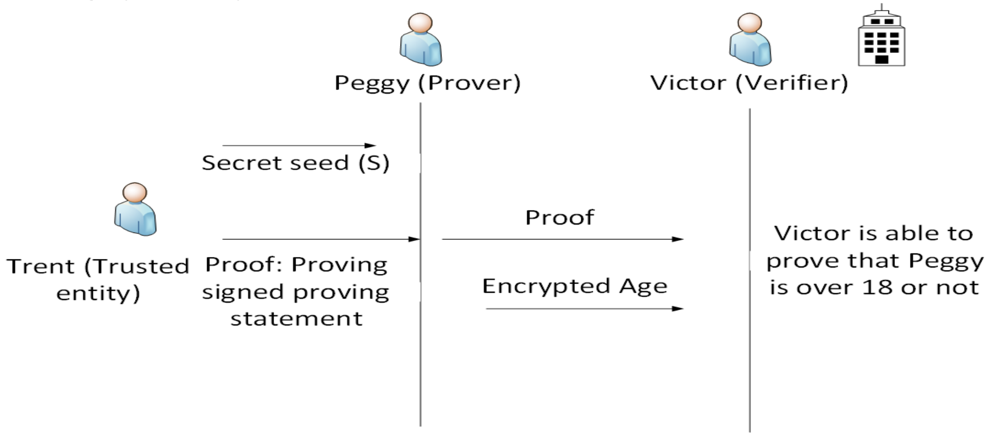
Finally, The Angular Web Application would interact with the rinkeby supply-chain contract using the Web3 framework. The Web3 Framework is being used to interact with HTML Webpages and provides an interface to convert EVM byte code to JavaScript compile time code which helps the developer to send and retrieve information from the smart contract. The following Separation of Concerns design principle construct provided above has been followed in the web application while developing the web application. (web3.js - Ethereum JavaScript API — web3.js 1.0.0 documentation, 2020).The following diagram below provides a brief explanation of how the Angular application would be developed and how the data interaction is being followed while building the inventory management system on the ethereum blockchain.

Inventory Web Application Interaction With the Ethereum Blockchain



2.3. Implementation

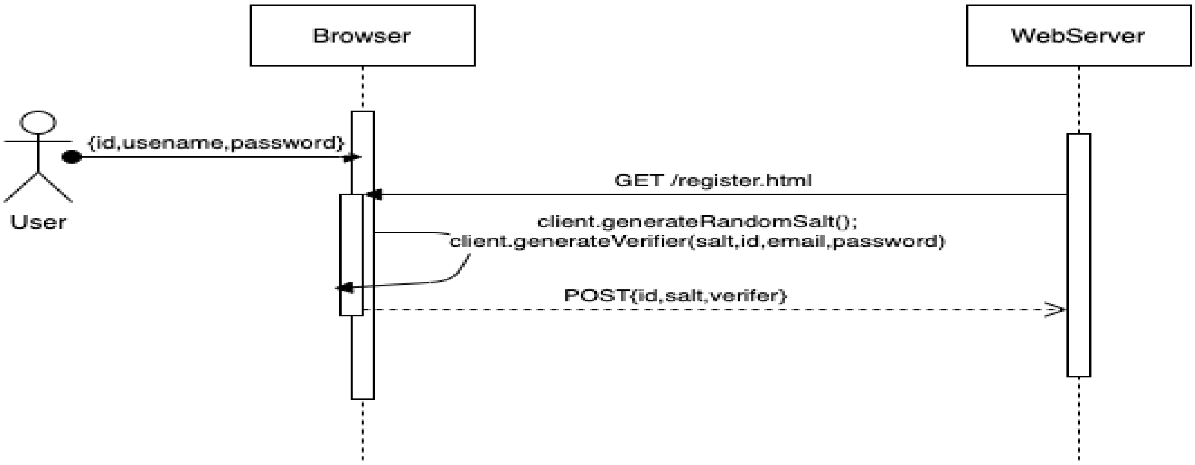
The main algorithm being used is implementing the smart contracts that model the structures of the database on the Ethereum blockchain using solidity which is used to develop smart contracts. Another potential cryptographic standard used in the project is the usage of zero-knowledge proofs to improve anonymity and enhancing the privacy of the distributed database by using a prover and verifier construct. The diagram below briefly describes the zero-knowledge proof implementation


Peggy (Prover)
Victor (Verifier)
Secret seed (S)
Trent (Trusted entity)
Proof: Proving signed proving statement
Proof
Encrypted Age
Victor is able to prove that Peggy is over 18 or not

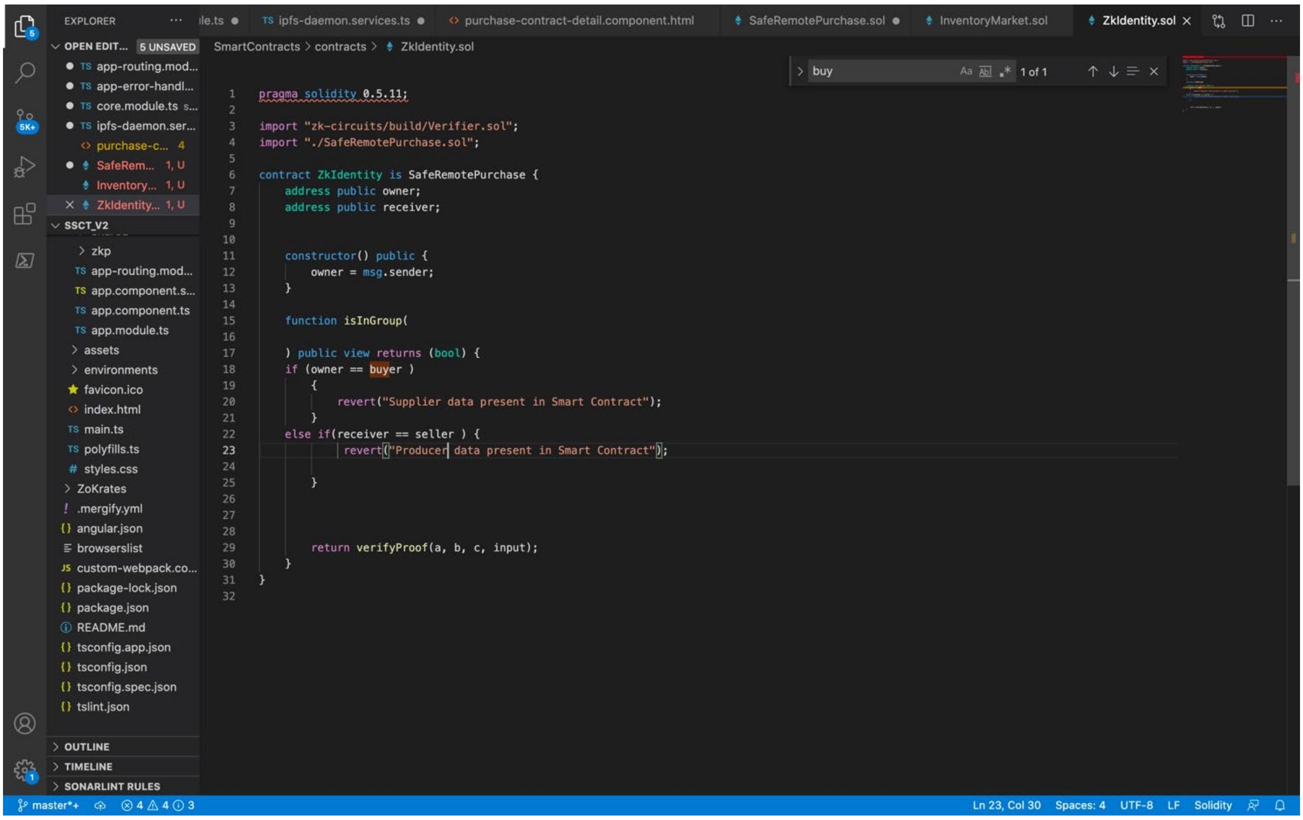
The following Zero-knowledge proof constructs explain the mathematical representation of a ZKP which is represented below. The following representation shows us how the transaction and identity verification would be used within the project...The following representation shows the implementation of the Turing machine in the P, V, S construct where they represent Turing machines. The view representation below represents the interaction between the prover and verifier construct. To Exemplify, The following construct is describing the interactive proof which is the result S . The interactive proof of implementing zero-knowledge is a probabilistic verifier that exists for the given construct at given input between P and V.

$$\forall x \in L, z \in \{0,1\}^*, \mathrm{View}_{\hat{V}}\left[P(x) \leftrightarrow \hat{V}(x,z)\right] = S(x,z)$$

The project aims to implement two zero-knowledge proofs. The first zero-knowledge proof is the implementation of zero-knowledge proofs between the supplier and the producer hiding the supplier login information while the producer logs into the inventory web application. This helps prevent password leakage and the zero-knowledge proof implemented in the project wouldn’t require any key exchange. (Limited 2020). The following sequence diagram below shows us how zero-knowledge proof password verifier is implemented in the inventory management web application.



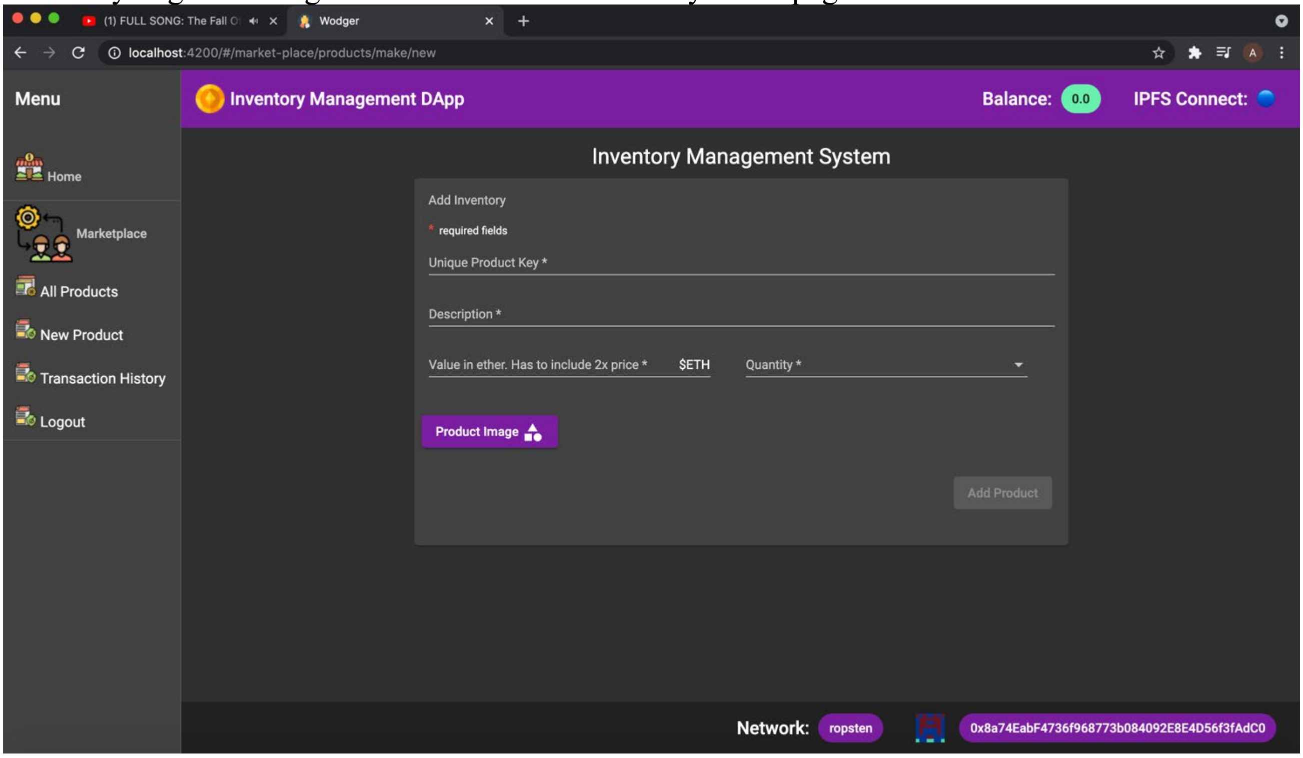
The second zero knowledge proof is the implementation of zero-knowledge proof to hide the transaction data from the public ledger ensuring the public ledger is compliant with PCI Compliance which is the standard for encrypting financial information. (PCI Compliance Guide Frequently Asked Questions | PCI DSS FAQs, 2020). The transactional details includes hiding the supplier and producer etherium contract addresses through the utilization of zero-knowledge proofs.



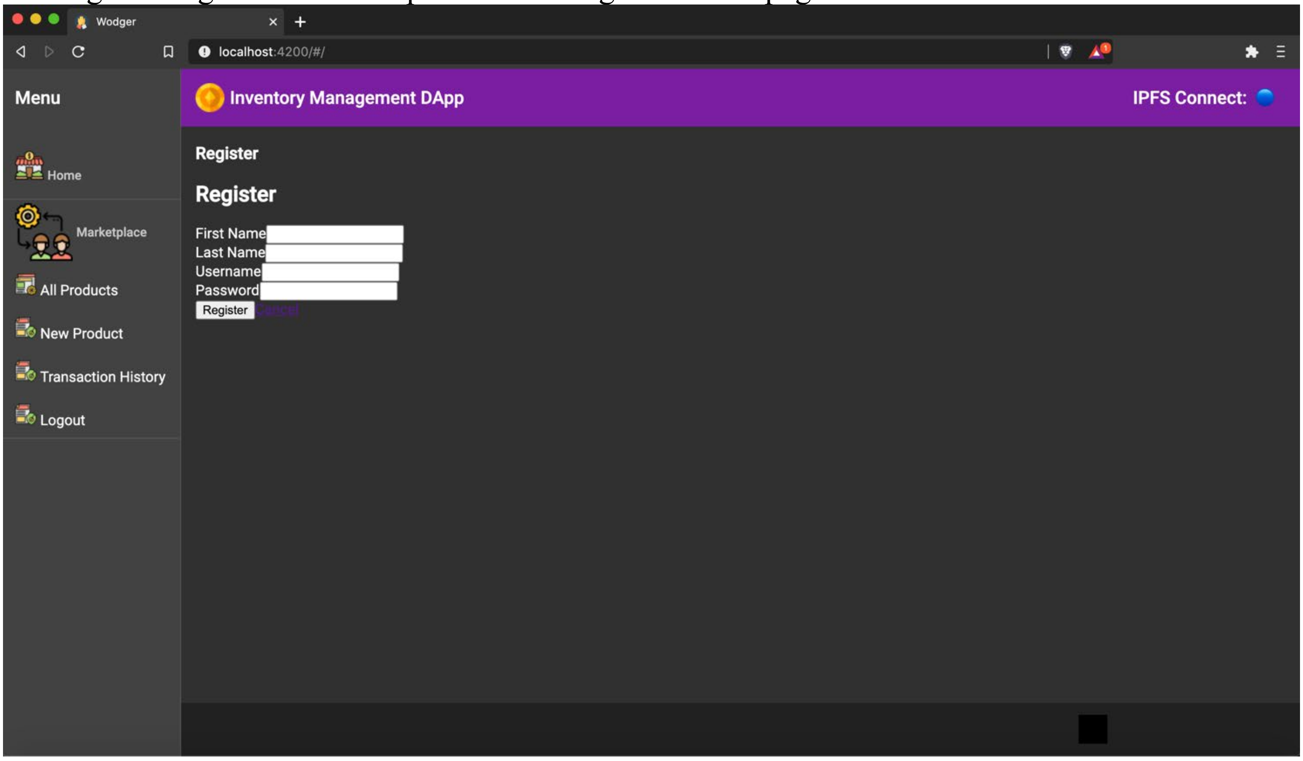
```
pragma solidity 0.5.11;

import "zk-circuits/build/Verifier.sol";
import "./SafeRemotePurchase.sol";

contract ZkIdentity is SafeRemotePurchase {
    address public owner;
    address public receiver;


    constructor() public {
        owner = msg.sender;
    }

    function isInGroup(

    ) public view returns (bool) {
    if (owner == buyer )
        {
            revert("Supplier data present in Smart Contract");
        }
    else if(receiver == seller ) {
            revert("Producer data present in Smart Contract");

        }



        return verifyProof(a, b, c, input);
    }
}
```

2.4. Graphical User Interface (GUI)

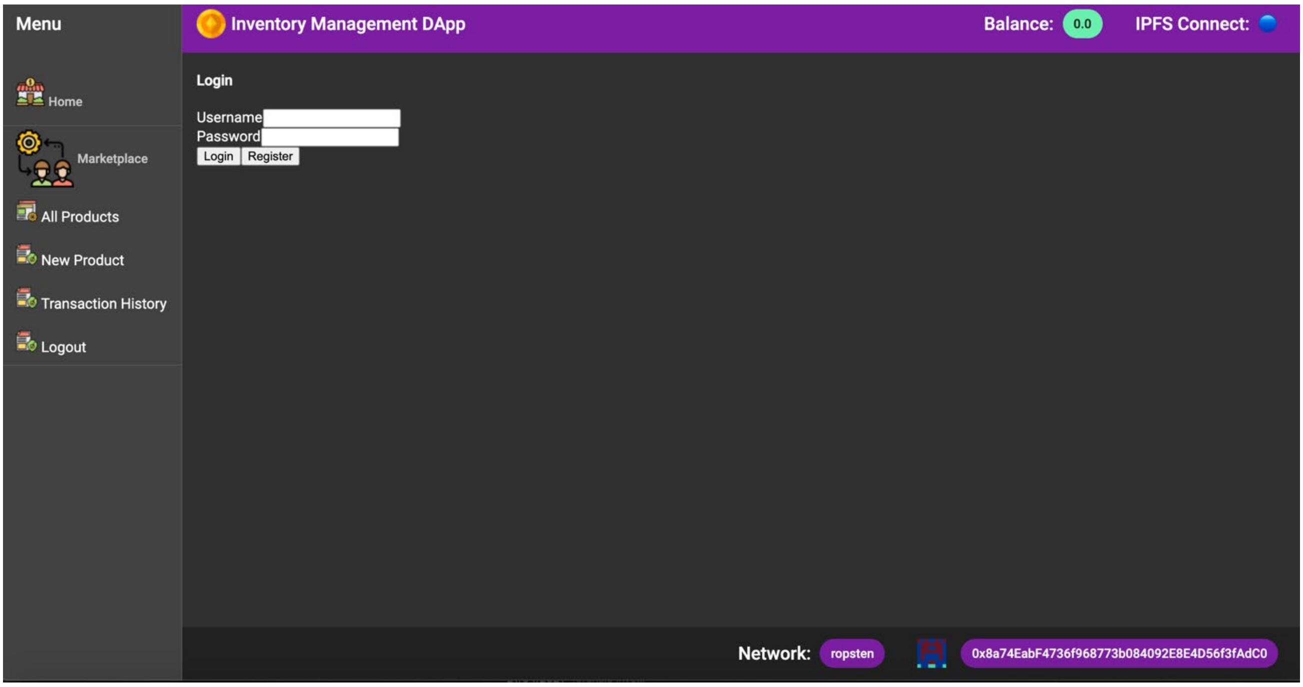
Inventory Page: The Page shows how to add inventory to the page



Register Page: Allows new producers to register on the page



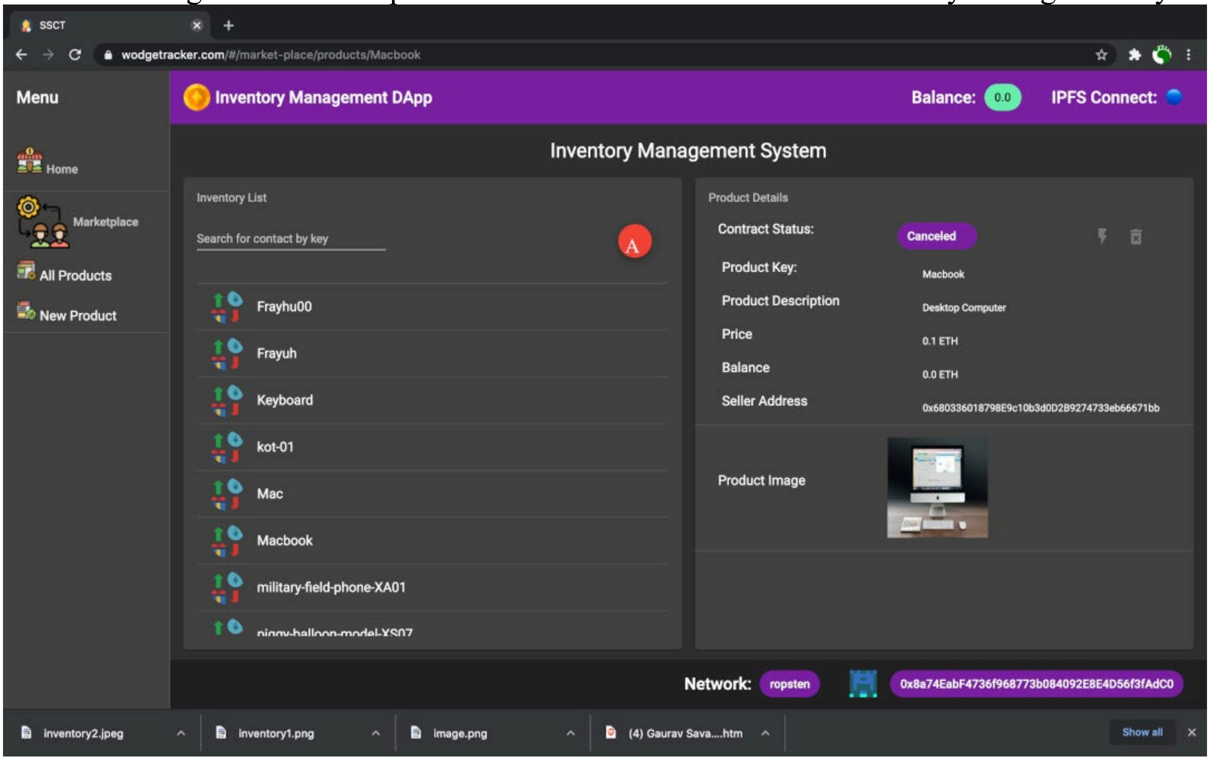
Login Page: This Allows the. Supplier and producer to login to the respective pages.



Network: ropsten

0x8a74EabF4736f968773b084092E8E4D56f3fAdC0

Orders Page: This allows producers to add new orders to the inventory management system



Frayhu00
Frayuh
Keyboard
kot-01
Mac
Macbook
military-field-phone-XA01
piggy-balloon-model-XS07

Contract Status: Canceled

Product Key: Macbook

Product Description: Desktop Computer

Price: 0.1 ETH

Balance: 0.0 ETH

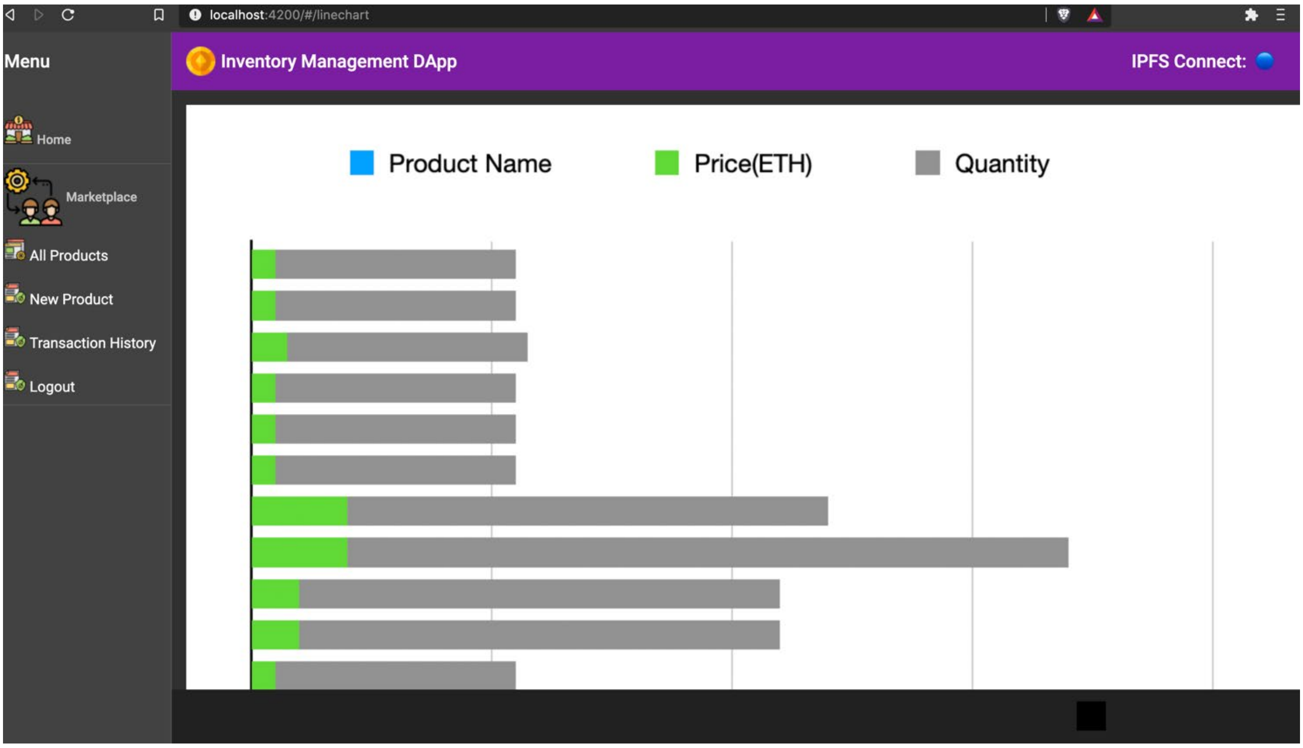
Seller Address: 0x680336018798E9c10b3d0D2B9274733eb66671bb

Product Image

Network: ropsten

0x8a74EabF4736f968773b084092E8E4D56f3fAdC0

Transactional History; Shows a Line Chart of inventory and orders that have been added to the system which helps in tracking inventory data.



2.5. Testing

The Testing tools that would be used in the Project would be Karma and Protractor for unit testing and integration tests in the application.The following tests are reported in the appendix which would ensure the application would adhere to OWASP Secure Design Principles for developing secure web applications. (Burp Suite - Application Security Testing Software, 2020).

The goal for testing the web application and smart contracts is to test the functions to ensure they perform as expected which would involve using the Mocha Testing Framework using assertion tests. Secondly, Karma would be used to perform Unit tests on Login and Register Pages, unit tests on the Inventory, Recall of Products and Orders(Shipping) pages functions are working as expected. (2017 Top 10 | OWASP, 2020). The following checks would be provided while unit testing the web application and the smart contract.

To further provide a qualitative analysis Privacy Assessment Model has been used to assess the overall privacy and security of the system which would provide detailed feedback over the security analysis performed based on OWASP's top 10 risks. the Formula that would be used while performing the qualitative analysis is (Damage + Reproducibility + Exploitability + Affected Users + Discoverability) / 5.(Qualitative Risk Analysis with the DREAD Model - Infosec Resources, 2020).

In Conclusion, The evidence of risk assessment reports and testing code-coverage snapshots is provided in the appendix to provide evidence of privacy-based testing and unit testing performed on the web application.

2.6 Evaluation

The system was evaluated against the DPIA and Blockchain AABN Privacy Assessment Model and Smart Contract Best Practices Considerations. The DPIA and blockchain AABN privacy assessment model gives us a brief understanding of the evaluation functions used in the project. This involves the usage of security-based evaluations that would be used to evaluate the projects. Furthermore, ethically obtained data has been used to provide robustness to the project which enhances the functionality of the application. While evaluating privacy within the application 6 evaluation criteria were followed. These evaluation criteria include identifying privacy impact methods, choosing the evaluation criteria, evaluation of existing privacy evaluations, analyzing relevant literature for privacy guidance during the development of Projects, and finally, a privacy impact assessment was done to evaluate the privacy of ethically obtained data within the etherium distributed ledger.The privacy impact assessment evaluated for the project is attached in the appendix of the document.

3.0 Conclusions

The advantages of the project for inventory management systems would be the immutability feature of databases that allows adding the data to an immutable ledger that ensures data isn't stolen or deleted from the database as changes made to the block on the blockchain would need appropriate permissions and significant computational power would be needed to find the hash of the block on the blockchain networks.

Secondly, privacy-enhanced protocols like zero-knowledge proofs can be used to enhance privacy to the blockchain data which enhances the privacy of identity data on the inventory management systems by hiding the relevant supplier or producer data while entering the data onto the distributed ledger .Disadvantages of the usage of blockchain-enabled inventory management systems would be time consuming and immutable nature of the database as transactions need to be verified before it could be added to the blockchain. It's a time-consuming process to perform actions like adding inventory to the ropsten etherium blockchain.

Furthermore, the usage of etherium based dapps would involve the usage of metamask extensions that is a necessity while authenticating onto the etherium blockchain. Therefore, the metamask extension needs to be added to the web application as a necessity to connect with the ropsten etherium testnet. The project benefits include the development of a production-ready inventory management system that can authenticate with the etherium blockchain which can be used within the web application by suppliers and producers to add new inventory and orders into the web application. Finally, zero-knowledge proofs have been used to hide sensitive identity and payment information from the user which helps the user to comply with GDPR and PCI Compliance which are integral while handling sensitive user and payment data within an organization.

To Conclude, the inventory management system used within the web application is a decentralized web application built on top of the etherium blockchain. The strengths of the web application include the usage of privacy-enhancing cryptographic protocols and etherium distributed ledger which enhances the privacy of inventory management systems and thereby provides the suppliers and producers a secure inventory web application.

4.0 Further Development or Research

The Inventory Supply-chain project emphasizes the usage of Zero-Knowledge Proofs which uses privacy-enhancing cryptographic protocols like the prover verifier construct to hide sensitive data in the web application. Additional Research on the cryptographic protocols would lead to better hashing protocols for the data. Quantum cryptography is an approach that could be taken for the implementation of Zero-Knowledge proofs.

Companies like Google have built their first Quantum Computer for solving difficult cryptographic challenges and its evolution could be detrimental to blockchain technology as the block hash network could be broken through the usage of Quantum Computers. (Savage, 2021). Therefore, The usage of Quantum Resistant Protocols is a necessity to build quantum-resistant cryptographic protocols, Additional Time and research on the project would help in furthering the knowledge within cryptography through the development of a proof of concept of quantum-resistant zero-knowledge proofs which could be developed using the Schor's algorithm that would further enhance the privacy of data on the etherium blockchain and provide resistance to quantum computing.

5.0 References

1. Owasp.org. 2020. *2017 Top 10 | OWASP*. [online] Available at: <https://owasp.org/www-project-top-ten/2017/Top_10> [Accessed 2 December 2020].
2. Infosec Resources. 2020. *Qualitative Risk Analysis With The DREAD Model - Infosec Resources*. [online] Available at: <https://resources.infosecinstitute.com/topic/qualitative-risk-analysis-dread-model/> [Accessed 3 December 2020].
3. Portswigger.net. 2020. *Burp Suite - Application Security Testing Software*. [online] Available at: <https://portswigger.net/burp> [Accessed 3 December 2020].
4. Alexey Naumov. 2020. *Separation Of Concerns In Software Design*. [online] Available at: <https://nalexn.github.io/separation-of-concerns/> [Accessed 3 December 2020].
5. Infura. 2020. *Ethereum API | IPFS API Gateway | ETH Nodes As A Service | Infura*. [online] Available at: <https://infura.io/> [Accessed 3 December 2020].
6. Inboundlogistics.com. 2020. *Going The Distance: Securing Supply Chains From Cyber Attack - Inbound Logistics*. [online] Available at: <https://www.inboundlogistics.com/cms/article/going-the-distance-securing-supply-chains-from-cyber-attack/> [Accessed 3 December 2020].
7. Ng, A., 2020. *How The Equifax Hack Happened, And What Still Needs To Be Done*. [online] CNET. Available at: <https://www.cnet.com/news/equifaxs-hack-one-year-later-a-look-back-at-how-it-happened-and-whats-changed/> [Accessed 3 December 2020].
8. Greenberg, A., 2020. *The Untold Story Of Notpetya, The Most Devastating Cyberattack In History*. [online] Wired. Available at: <https://www.wired.com/story/notpetya-cyberattack-ukraine-russia-code-crashed-the-world/> [Accessed 7 December 2020].
9. Web3js.readthedocs.io. 2020. *Web3.Js - Ethereum Javascript API — Web3.Js 1.0.0 Documentation*. [online] Available at: <http://web3js.readthedocs.io/> [Accessed 7 December 2020].
10. PCI Compliance Guide. 2020. *PCI Compliance Guide Frequently Asked Questions | PCI DSS Faqs*. [online] Available at: <https://www.pcicomplianceguide.org/faq/> [Accessed 8 December 2020].
11. 2020. [online] Available at: <https://resources.whitesourcesoftware.com/blog-whitesource/dast-dynamic-application-security-testing> [Accessed 8 December 2020].
12. Limited, C., 2020. *Zero-Knowledge Protocols Without Magic*. [online] Cossacklabs.com. Available at: <https://www.cossacklabs.com/blog/zero-knowledge-protocols-without-magic.html> [Accessed 21 December 2020].
13. Savage, N., 2021. *Google&rsquo;s Quantum Computer Achieves Chemistry Milestone*. [online] Scientific American. Available at:

<https://www.scientificamerican.com/article/googles-quantum-computer-achieves-chemistry-milestone/> [Accessed 1 February 2021].

14. Gdpr.eu. 2021. [online] Available at: <https://gdpr.eu/wp-content/uploads/2019/03/dpia-template-v1.pdf> [Accessed 8 February 2021].
15. T. Lu, R. Yan, M. Lei , and Z. Lin, "AABN: Anonymity assessment model based on Bayesian network with application to the blockchain," in China Communications, vol. 16, no. 6, pp. 55-68, June 2019, doi: 10.23919/JCC.2019.06.005.

6.0 Appendices

This section should contain information that is supplementary to the main body of the report.

6.1. Project Plan

Project Plan included in Proposal attached in the appendix.

6.2. Reflective Journal

Reflective Journal(October 2020)

8.0 What?

9.0 In my software project module I started out thinking about the idea on what I wanted to do during the summer of 2020 as I was really interested in cybersecurity so I decided to specialise in cybersecurity when I was asked to choose a Specialisation and I was told to look at few projects so I pondered upon different ideas and the idea of securing the supply chains using cybersecurity and blockchain technology was really that was in my mind since a few months so I decided this would be one of the ideas I would like to pitch in my project pitch video.

10.0

11.0 So What?

12.0 I was interested in using the blockchain so I started taking courses in smart contract development using solidity online and started to learn more about supplychain attacks and how it is mitigated today.When classes started in September 2020 We were told to submit a project pitch and were outlined guidelines to pitch the project .I pitched in the idea of SSCT(Secure Supply Chain Tracker) which utilises the power of blockchain technology to secure the data with advanced security features like zkp that hides the private and sensitive data from the users .I felt that pitching a idea related to blockchain was really interesting as I had never played a lot around smart contracts and solidity which are to me felt really challenging composition to the project as I believe I would be able to work on a challenging component in my final year of College.

13.0 Now What?

14.0 I am waiting for a approval for the project pitch which would be the next step for me in regards to writing the project proposal and ethics document.We took classes on time management which I would like to utilise in my software project module using project management tools like trello and creating a project plan would really help me keep in track with the deadlines and objectives I would like to meet in the first few weeks of project development.

Reflection Journal November 2020

16.0 What Happened

17.0 My Workload for the Month of November involved getting feedback from my project supervisor over the project pitch over the potential challenges that are involved in building the project.Following the feedback the issues in regards to the project pitch were addressed in the proposal and were looked upon while submitting the Project Proposal. The Project Proposal submitted contained detailed information in regards to the reasons over choosing the project,project plan,potential technological frameworks and testing environments that would be used to build the blockchain web application and the smart contracts.

18.0

19.0 Then What?

20.0 The current progress looks into the feedback of the project proposal received over providing lucid objectives over building the zero knowledge proofs that would be implemented in the project and providing a extension to the gantt chart provided to the months of May ensuring the Project span lasts from September to May which will provide the detailed structure over how the project is broken down and implemented across the last 2 semesters.In terms of development progress I started working on the boilerplate code which would act as a factor or a base to test and deploy the contracts. This was build with the Node.js Framework ,Mocha and Infura Node which were used to test and deploy the test contract to the rinkeby test network.

21.0 What Next?

22.0 My Plan for the next few weeks is to continue to build the working prototype which is a requirement for the midpoint presentation.My plan is to build the inventory smart contract,creation of a interaction with angular and smart contract using Web3.js.This would be needed to be completed in the following weeks.Secondly the need to build a High level design for the midpoint presentation would be a priority for the project.

December 2020
In December I was busy refining my final assessments which involved refining my work based on the feedback that was received for the project proposal. I had started working on the smart contract deployment code during the 3rd week of November which is integral for the supply-chain web application. Furthermore, I spent time on the development of the project which involved focusing on the frontend of the web application.

In The second week of December, I started my critical components of the project which involved the Requirement Specification Document and the Frontend of the application adhering to the criteria defined in the Midpoint Documentation. Finally, on December 22nd I Submitted my Midpoint presentation along with the Requirement Specification Document. To be Honest, The time-limit did go beyond the recommended timeslot however I wanted to cover all the criteria as expected in the rubrics.

Furthermore, in the coming few weeks in January, I would like to focus on the completion of the rendering of data that was deployed to the Ethereum Test net and the completion of the second zero-Knowledge Proof that would provide robustness to the data that is represented in the web application.

January 2021

The following journal reflects on the project progress over January. I started Spending Time on the project after I finished my examination period. I focused on developing and rendering the smart contract data from the solidity smart contract that has been deployed on the ropsten testnet into the angular web Application. I utilized web3.js and Ngrx state management javascript libraries to perform these actions. These actions ensured the data that had been received from the smart contract will perform similar actions to the likes of a CRUD based Inventory Web Application.

The progress for January is that I was able to complete the CRUD-based functionality which allows users to add new inventory,view orders,update orders and retract inventory contracts within the web application. This allowed suppliers and producers to get a robust view of tracking inventory data on the Web Application and tracks user data from the supplier to the producer in an effective way by showing all stages involved in Inventory Supply-chain Management.

My Future Goals for the project are to integrate the Login and Signup functionality and Start Writing Tests for the Frontend part of the Web Application which will provide robustness and adhere to secure coding practices within software development. Furthermore, during February and March, I would spend time on writing Penetration testing reports, Building the Second Zero-Knowledge Proof for supplier data, and finally auditing the smart contract against known solidity smart contract attacks.

February 2021

The Following Document reflects upon the progress of the Software Project in February 2020. I was able to start and submit my first draft of the DPIA(Data Privacy Impact Assessment) for the Inventory Web Application. Furthermore, I finished my second zero-knowledge proof which involves hiding the supplier and producer addresses within the web application by providing a proof of identity that says the address exists rather than providing the sensitive transactional addresses on the public inventory web-application.Finally, I was able to submit a draft for the project showcase poster which was finalized and approved by careers office.

In March, I plan to integrate the login and signup functionality within the blockchain inventory web application and how I would get a transaction history graph implemented within the inventory application. The implementation of these 2 functionalities would culminate in the completion of the project that I had planned on the building during the requirement planning phase in the project.

To Summarise,My progression of developing a web application is on an appropriate track. I am on a good progression track to finalise and submit the project by May 9th.

March 2021

My progress in the month of March includes development of the web application. I was able to successfully integrate the login and signup pages within the web application. My Plan for month of April is to start the testing of the front end of the web application and continue to work on other deadlines for other modules. For this project I believe I have completed my ideation of the project from the start to finish building a web application for the inventory supply chains. My plan would be to continue to work on the project and build tests for the frontend part of the web application.

April 2021

My Reflection Journal primarily reflects on the progress during the month of April and Summary of overall progress of the project. I was able to complete the project poster and the testing reports that are necessary for the testing requirements and project showcase. My plan for the first few weeks in May is to finalize my codebase and technical document based on feedback received from the supervisor and submit it on May 16th. Furthermore, I plan to prepare my final presentation slides for the project after the submission of the project based on the guidelines provided during the software project meetings.

22.1. Ethics Document

National College of Ireland

DECLARATION OF ETHICS CONSIDERATION

School of Computing

Name: Gaurav Ramesh Savanur

StudentID:17114357

Program: BSHC

Year :4

Module: **Software Project**

Project Title: **Improving the Privacy of Inventory Supply chains using Zero-Knowledge Proofs**

Please circle (or highlight) as appropriate

This project involves human participants	**No**

Introduction

Secondary data refers to data that is collected by someone other than the current researcher. Common sources of secondary data for social science include censuses, information collected by government departments, organizational records , and data originally collected for other research purposes. Primary data, by contrast, is collected by the investigator conducting the research.

A project that does not involve human participants requires the ONLY completion of the Declaration of Ethics Consideration Form and submission of the form on module's Moodle page

A project that involves human participants requires ethical clearance and an Ethics Application Form must be submitted through the module's Moodle page. Please refer to and ensure compliance with the ethical principles stated in NCI Ethics Form available on the Moodle page.

The following decision table will assist you in deciding if you have to complete the Declaration of Ethics Consideration Form or/and the Ethics Application Form.

Public Data	Y	Y	Y	Y	N	N	N	N
Private Data	Y	Y	N	N	Y	Y	N	N
Human Participants	Y	N	Y	N	Y	N	Y	N
Declaration of Ethics Consideration Form	x	X	x	X	X	X	x	

Ethics Application Form	X		X		X		X	

Please circle (or highlight) as appropriate

The project makes use of secondary dataset(s) created by the researcher	Yes
The project makes use of public secondary dataset(s)	Yes
The project makes use of non-public secondary dataset(s) Approval letter from non-public secondary dataset(s) owner received	No

Sources of Data:

It is students' responsibility to ensure that they have the correct permissions/authorizations to use any data in a study. Projects that make use of data that does not have authorization to be used, will not be graded for that portion of the study that makes use of such data.

Public Data

A project that makes use of public secondary dataset(s) ***does not need ethics permission****, but* ***needs a letter/email from the copyright holder*** *regarding potential use.*

Some websites and data sources allow their data sets to be used under certain conditions. In these cases, a letter/email from the copyright holder is NOT necessary, but the researcher should cite the source of this permission and indicate under what conditions the data are allowed to be used. See Appendix I for examples of permissions granted by Fingal Open Data, and Eurostat website.

Where websites or data sources indicate that they do not grant permission for data to be used, you will still need a letter/email from the copyright holder. For example, see Appendix II for an example from the Journal of Statistics Education.

Private Data

A project that makes use of non-public (private) secondary dataset(s) must receive data usage permission from School of Computing.

An approval letter/email from the owner *(e.g. institution, company, etc.)* ***of the non-public secondary dataset must be attached to the Declaration of Ethics Consideration****. The letter/email must confirm that the dataset is anonymised and permission for data processing, analysis and public dissemination is granted.*

Evidence for use of secondary dataset(s)

Include dataset(s) owner letter/email or cite the source for usage permission

Data.opennepal.net. 2020. *Data Related To Logistics: Maintenance Of Stock (2011-14) | Opendata*. [online] Available at: <http://data.opennepal.net/content/data-related-logistics-maintenance-stock-2011-14> [Accessed 4 October 2020].

I would be using the Logistics data for the sample dataset in the project for tracking the data over and improving the security flaws using smart contracts.

CHECKLIST

Non-public/private secondary dataset(s) -Owner letter/email is attached to this form ***OR*** Citation and link to the web site where permission is granted – provided in this form	No

ETHICS CLEARANCE GUIDELINES WHEN HUMAN PARTICIPANTS ARE INVOLVED

The Ethics Application Form must be submitted on Moodle for approval prior to conducting the work.

Considerations in data collection

- Participants will not be identified, directly or through identifiers linked to the subjects in any reports produced by the study
- Responses will not place the participants at risk of professional liability or be damaging to the participants' financial standing, employability, or reputation
- No confidential data will be used for personal advantage or that of a third party

Informed consent

- Consent to participate in the study has been given freely by the participants
- participants can understand the project goals.
- Participants have been given understandable information sheets
- Likely benefits of the project itself have been explained to potential participants
- Risks and benefits of the project have been explained to potential participants
- Participants have been assured they will not suffer physical stress or discomfort or psychological or mental stress
- The participant has been assured s/he may withdraw at any time from the study without loss of benefit or penalty
- Special care has been taken where participants are unable to consent for themselves (e.g children under the age of 18, elders with age 85+, people with intellectual or learning disability, individuals or groups receiving help through the voluntary sector, those in a subordinate position to the researcher, groups who do not understand the consent and research process)
- Participants have been informed of potential conflict of interest issues
- The onus is on the researcher to inform participants if deception methods have to be used in a line of research

I have read, understood, and will adhere to the ethical principles described above in the conduct of the project work.

Signature:

Gaurav Ramesh Savanur

Date: 03/10/20

Appendix I

1) Nepal Open Data: http://data.openNepal.net

Appendix II

Nepal Open Data : Data.opennepal.net. 2020.
Users are free to access and use the data under the Open Data Commons Open Database License (ODbL) licence free of charge

License Statement

Open Data Commons Open Database License (ODbL) Summary
This is a human-readable summary of the ODbL 1.0 license. Please see the disclaimer below.

You are free:

- *To share*: To copy, distribute and use the database.
- *To create*: To produce works from the database.
- *To adapt*: To modify, transform and build upon the database.

As long as you:

- *Attribute*: You must attribute any public use of the database, or works produced from the database, in the manner specified in the ODbL. For any use or redistribution of the database, or works produced from it, you must make clear to others the license of the database and keep intact any notices on the original database.
- *Share-Alike*: If you publicly use any adapted version of this database, or works produced from an adapted database, you must also offer that adapted database under the ODbL.
- *Keep open*: If you redistribute the database, or an adapted version of it, then you may use technological measures that restrict the work (such as DRM) as long as you also redistribute a version without such measures.

Disclaimer

This is not a license. It is simply a handy reference for understanding the ODbL 1.0 — it is a human-readable expression of some of its key terms. This document has no legal value, and its contents do not appear in the actual license. Read the full ODbL 1.0 license text for the exact terms that apply. (Data.opennepal.net. 2020)

22.2. Privacy Impact Assessment Report(Modelled Based on the Template Provided by GDPR.eu and the Blockchain AABN Anonymity Assessment Model)

Step 1: Identify the need for a Privacy Impact Assessment

Explain broadly what the project aims to achieve and what type of processing it involves. You may find it helpful to refer or link to other documents, such as a project proposal.Utilize the AABN Privacy Impact Model Sender and Receiver sets to analyze the data anonymity standards.

The project aims to improve the privacy of inventory management systems using blockchain technology and zero-knowledge proofs to hide sensitive data from users and improve the security of inventory management systems through the utilization of the following technology. The project proposal Is linked in the appendix section of the technical report which provides an overview of the project. The inventory data used in the application is obtained from the Nepal open data open source repository which has been utilised to test the data on the Ethereum blockchain. The sender sets utilised in the application involve the input data obtained while the supplier adds inventory into the web application. The receiver sets include the order data that is received by the supplier from the producer and the delivery order updates that are provided when the user successfully delivers the product to the producer.

Step 2: Describe the data processing

Describe the nature of the processing: how will you collect, use, store and delete data? What is the source of the data? Will you be sharing data with anyone? You might find it useful to refer to a flow diagram or other way of describing data flows. What types of processing identified as likely high risk are involved?

The data stored on the application would be added by suppliers and producers where inventory data would be entered into the Ethereum distributed ledger through the inventory form and furthermore, producers can review the inventory list and place orders for the inventory. The data flow involved in the project has been described in the use case diagram, Use-Case Description and the design and architecture section of the project which describes the Project and provides a brief overview of the processes involved while building the decentralized inventory web application.

Describe the scope of the processing: what is the nature of the data, and does it include special category or criminal offense data? How much data will you be collecting and using? How often? How long will you keep it? How many individuals are affected? What geographical area does it cover?

The nature of data used for testing the proof of concept of building the web application is sources from ethical sources. The data sources for the proof of concept utilizes the Nepal Open Data Source- http://data.openNepal.net .The source utilized is free to access database which is licensed under the OdbL(Open Data Commons Opens Database Licence) license which is free to access database.

The data sources are ethically obtained and no human participants were involved during the testing and development phase of the project. Further details in regards to ethical concerns are addressed in the ethics document and has been addressed in the initial requirement stages of the project.

Describe the purposes of the processing: what do you want to achieve? What is the intended effect on individuals? What are the benefits of the processing – for you, and more broadly?

The purpose of processing the data is to improve the privacy and immutability of data through the utilization of the Ethereum blockchain and zero-knowledge proofs. The intended effect on individuals include providing a safer way to store the inventory supply chain data and provide privacy enabled registration system which enhanced the privacy of the identity verification system by improving the cryptographic standards that are being utilized in the web application. The benefits of the processing of data include providing a safer way for producers and suppliers to add inventory and place orders on a web application. Secondly, the risk of failed execution of the delivery of the order when an order is placed is significantly reduced as both the supplier and the producer needs to stake Ethereum while the supplier delivers the order to the producer and Finally, the introduction of zero-knowledge proofs solved the key exchange dilemma that was involved during the introduction of Symmetric-key encryption while successfully logging a user onto the web application.

Step 3: Assess Relevant Stakeholders

Consider how to consult with relevant stakeholders: describe when and how you will seek individuals' views – or justify why it's not appropriate to do so. Who else do you need to involve within your organization? Do you need to ask your processors to assist? Do you plan to consult information security experts or any other experts?

Stakeholders have been assessed in the project through the utilization of analysis based approaches where inventory management system was analysed to figure out the challenged involved while building the web application. The challenges were assessed with technologies that would help in easing the cybersecurity attacks that are faced by organizations.

The plan to utilize zero-knowledge proofs and blockchain technology came about due to the fact that cybersecurity attacks had impacted the integrity of data stored on the Ethereum blockchain. This was further utilized to build a decentralized web application on the Ethereum blockchain.

Step 4: Assess necessity and proportionality

Describe compliance and proportionality measures, in particular: what is your lawful basis for processing? Does the processing achieve your purpose? Is there another way to achieve the same outcome? How will you ensure data quality and data minimization? What information will you give individuals? How will you help to support their rights?

The lawful basis of processing the data involve utilization of OpDL based license based data sources. The data stored on the ropsten etherium blockchain would serve as a standard while processing data sources within a decentralized web application. The data minimization and quality been emphasized in the web application through sourcing, cleaning, and testing smaller datasets of data within the web application. The web application provides the supplier and producers a CRUD-based application to add inventory and place orders on the web application. Furthermore, The web application data has been assessed with GDPR and PCI ensuring the financial information and identity data is processed with the utmost discretion through the implementation of zero knowledge proofs to encrypt the supplier data and transactional data which has been implemented within the project to provide privacy to sensitive data.

Step 5: Identify and assess risks

Describe the source of risk and nature of The potential impact on individuals. Include associated compliance and corporate risks as necessary.	**Likelihood of harm**	**Severity of harm**	**Overall risk**
1. inventory data 2. orders data 3. user data 4. payment data	Low Low Low Low	Low Low Medium Medium	Low Low Low Low

Step 6: Identify measures to reduce risk

Identify additional measures you could take to reduce or eliminate risks identified as medium or high risk in step 5

Risk	Options to reduce or eliminate risk	Effect on risk	Residual risk	Measure approved
1.	utilization of blockchain technology to track inventory and orders data	eliminated	low	yes
2.	utilization of zero-knowledge proof to hide sensitive supplier data, producer data and payment data.	low	low	yes

Step 7: Assessment Result

The Formula which is used in the Report is

DREAD = (damage + reproducibility + Exploitability + affected users) / 5

- Damage Potential 2

  The Damage Potential in the Web Application is ranked at 2 due to the fact the data is stored on an immutable database ie the ropsten etherium blockchain which tracks the data on the database and the project ensures privacy by hiding sensitive information through the usage of Zero-Knowledge Proofs.

- Reproducibility -0

  Reproducing the data from the Inventory Management Web Application would be hard as OWASP Secure Coding Principles have been applied while developing the application and Common Smart Contract Security Flaws have been looked into while developing the smart contracts.

- Exploitability –0

  Threat actors wouldn’t be able to affect the system since OWASP Secure Design Principles like Principle or Fail Surely and Separation of Concerns Principle has been applied on both the frontend and the smart-contract boilerplate code ensuring the resources are separated into separate classes thereby reducing the risk of data breaches and provides interoperability to independent classes within the project.

- Affected Users

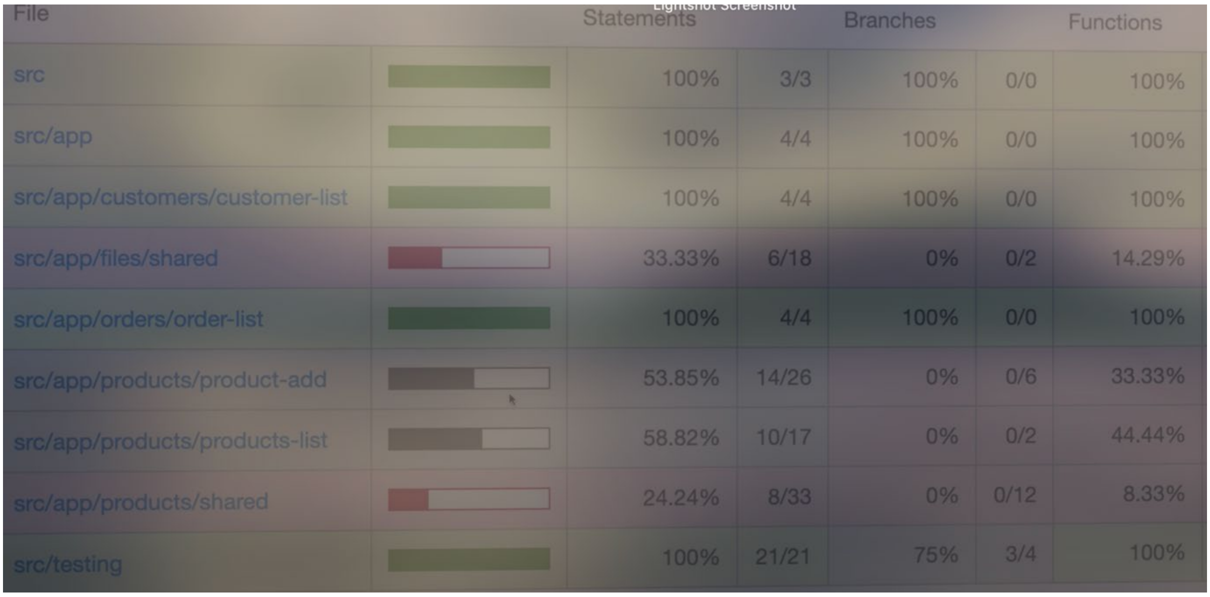
  5 – The users affected are the Supplier and the Producer. The principle of Secure Defaults has been applied to the security system while developing the application where the passwords are checked against known pawned passwords out in the world before the user is registered on to the system. However, Social Engineering Attacks is a potential way users can get access to the system account however passwords and database credentials are kept quite securely in different locations which obey the separation of duties principle and make security simple.

- Discoverability

  2 The risk to exploit the system is low as secure design principles have adhered to while developing above how location use of advanced brute-force techniques and tools like burp or Paros or usage of rootkits is a potential way where a bot could gain administrative credentials and gain control over user accounts on the system.

To Summarise the assessment of the system would lead us to (0+5+0+5)/5=2 which shows the system is at low risk. This concludes the qualitative analysis that was performed on the inventory web application.

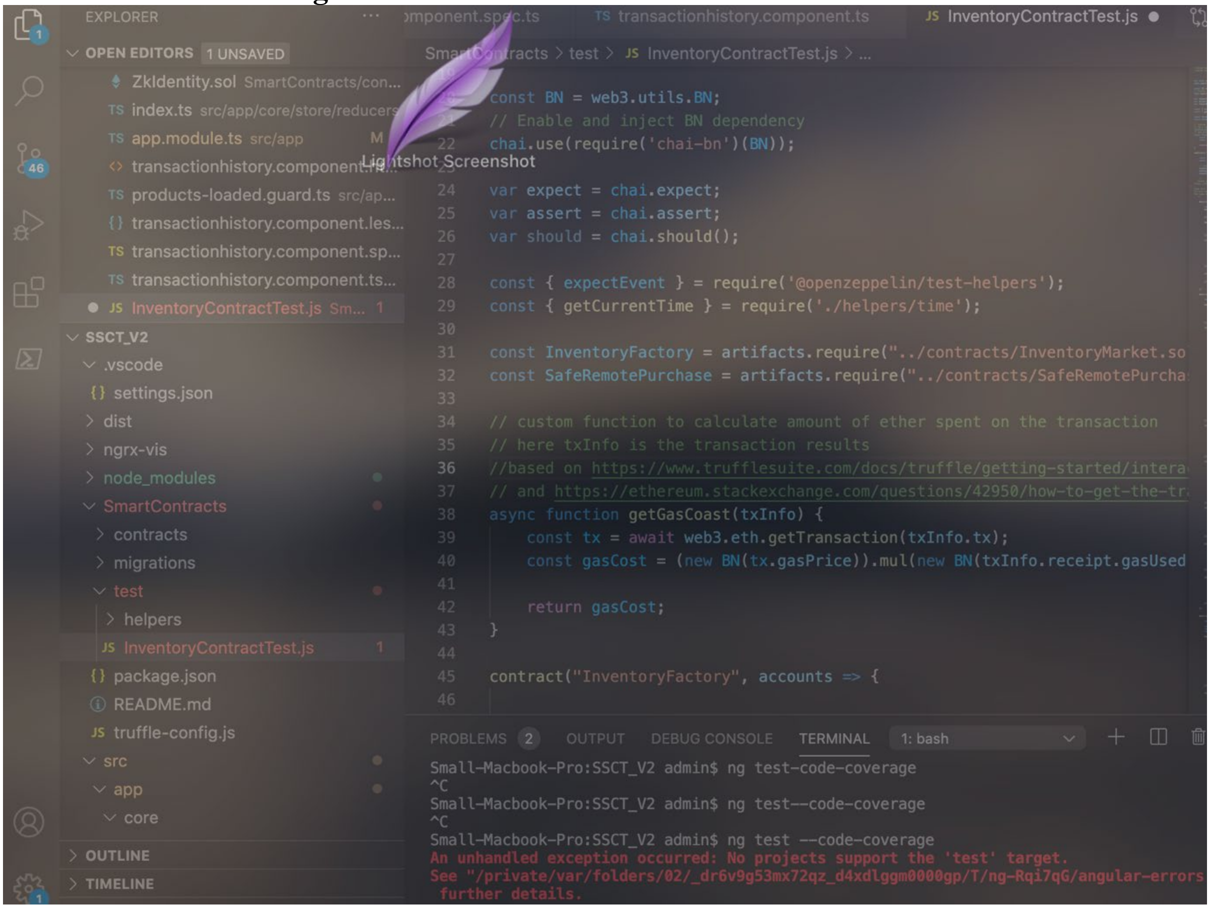
testing frontend code coverage:



Lightshot Screenshot

File		Statements		Branches		Functions
src	🟩	100%	3/3	100%	0/0	100%
src/app	🟩	100%	4/4	100%	0/0	100%
src/app/customers/customer-list	🟩	100%	4/4	100%	0/0	100%
src/app/files/shared	🟥	33.33%	6/18	0%	0/2	14.29%
src/app/orders/order-list	🟩	100%	4/4	100%	0/0	100%
src/app/products/product-add	▨	53.85%	14/26	0%	0/6	33.33%
src/app/products/products-list	▨	58.82%	10/17	0%	0/2	44.44%
src/app/products/shared	🟥	24.24%	8/33	0%	0/12	8.33%
src/testing	🟩	100%	21/21	75%	3/4	100%

Smart Contract Testing



EXPLORER
OPEN EDITORS 1 UNSAVED
ZkIdentity.sol SmartContracts/con...
index.ts src/app/core/store/reducers
app.module.ts src/app M
transactionhistory.component.ht...
products-loaded.guard.ts src/ap...
transactionhistory.component.les...
transactionhistory.component.sp...
transactionhistory.component.ts...
InventoryContractTest.js Sm... 1
SSCT_V2
.vscode
settings.json
dist
ngrx-vis
node_modules
SmartContracts
contracts
migrations
test
helpers
InventoryContractTest.js 1
package.json
README.md
truffle-config.js
src
app
core
OUTLINE
TIMELINE
omponent.spec.ts
transactionhistory.component.ts
InventoryContractTest.js
SmartContracts > test > InventoryContractTest.js > ...
Lightshot Screenshot
PROBLEMS 2
OUTPUT
DEBUG CONSOLE
TERMINAL
1: bash

```
const BN = web3.utils.BN;
// Enable and inject BN dependency
chai.use(require('chai-bn')(BN));

var expect = chai.expect;
var assert = chai.assert;
var should = chai.should();

const { expectEvent } = require('@openzeppelin/test-helpers');
const { getCurrentTime } = require('./helpers/time');

const InventoryFactory = artifacts.require("../contracts/InventoryMarket.so
const SafeRemotePurchase = artifacts.require("../contracts/SafeRemotePurcha

// custom function to calculate amount of ether spent on the transaction
// here txInfo is the transaction results
//based on https://www.trufflesuite.com/docs/truffle/getting-started/intera
// and https://ethereum.stackexchange.com/questions/42950/how-to-get-the-tr
async function getGasCoast(txInfo) {
    const tx = await web3.eth.getTransaction(txInfo.tx);
    const gasCost = (new BN(tx.gasPrice)).mul(new BN(txInfo.receipt.gasUsed

    return gasCost;
}

contract("InventoryFactory", accounts => {

```

```
Small-Macbook-Pro:SSCT_V2 admin$ ng test-code-coverage
^C
Small-Macbook-Pro:SSCT_V2 admin$ ng test--code-coverage
^C
Small-Macbook-Pro:SSCT_V2 admin$ ng test --code-coverage
An unhandled exception occurred: No projects support the 'test' target.
See "/private/var/folders/02/_dr6v9g53mx72qz_d4xdlggm0000gp/T/ng-Rqi7qG/angular-errors
 further details.
```

22.3. Project Proposal

National College of Ireland

Project Proposal

Improving the Privacy on the Inventory Supply-chains Using Zero-Knowledge Proofs

6/11/2020

BSHC4
Cyber Security
Academic Year 2020/2021
Gaurav Ramesh Savanur
17114357
X17114357@student.ncirl.ie

Contents

23.0 Objectives

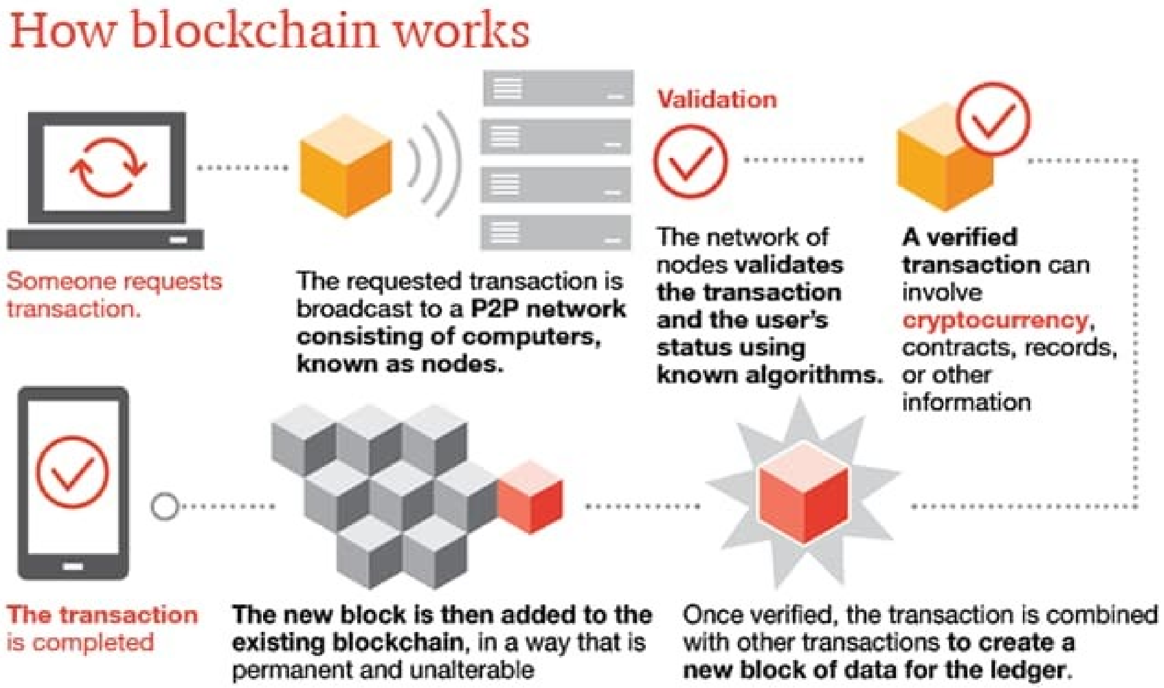
The Objectives of the Project is to securing the supply chain especially the 5 critical components Inventory, Production, Location, Transport and Return of Goods which are critical components within an organization ensuring the security of these components would help streamline and providing privacy to the data from the buyer to the supplier critically ensuring the data is kept at the most discretion in the project.

The project would be looking into 3 critical components within the logistics supply chain Inventory, Location, and Production. They were integral to the parts of the logistics which need to be secured so creating smart contracts for these three components would help them to be immutable since these contracts can be put upon the Etherium blockchain and securing the blockchain using cryptographic protocols would ensure the security of the project. The basis of the project was to provide an immutability feature by introducing an immutable database to track transactions and one of the challenges that were faced were looked into identity verification and trust-based cryptographic protocols could be utilized to provide identity verification based privacy-enhancement to sensitive data.

24.0 Background

The Motivation in developing the project in the Supply chain especially in Inventory Management Systems came across due to faults in security and privacy of transactional and sensitive identity data on the inventory application which lead me to look into blockchain technology ie. etherium to provide privacy by providing Zero-Knowledge proof implementation to Transaction data. The interest in the following area was based on events with recent hacks on the supply chain sector with the Mazar Virus in 2018 and the Equifax hack in 2019 who were involved in the deletion of the entire supply-chain which resulted in the logistics industry to move to a paper-based approach for 2 months before the Mazars infrastructure was brought back online whereas in the Equifax hack trust was lost since any user(third party) could be verified and can gain access to information using third party download links. (Barrett, 2020).

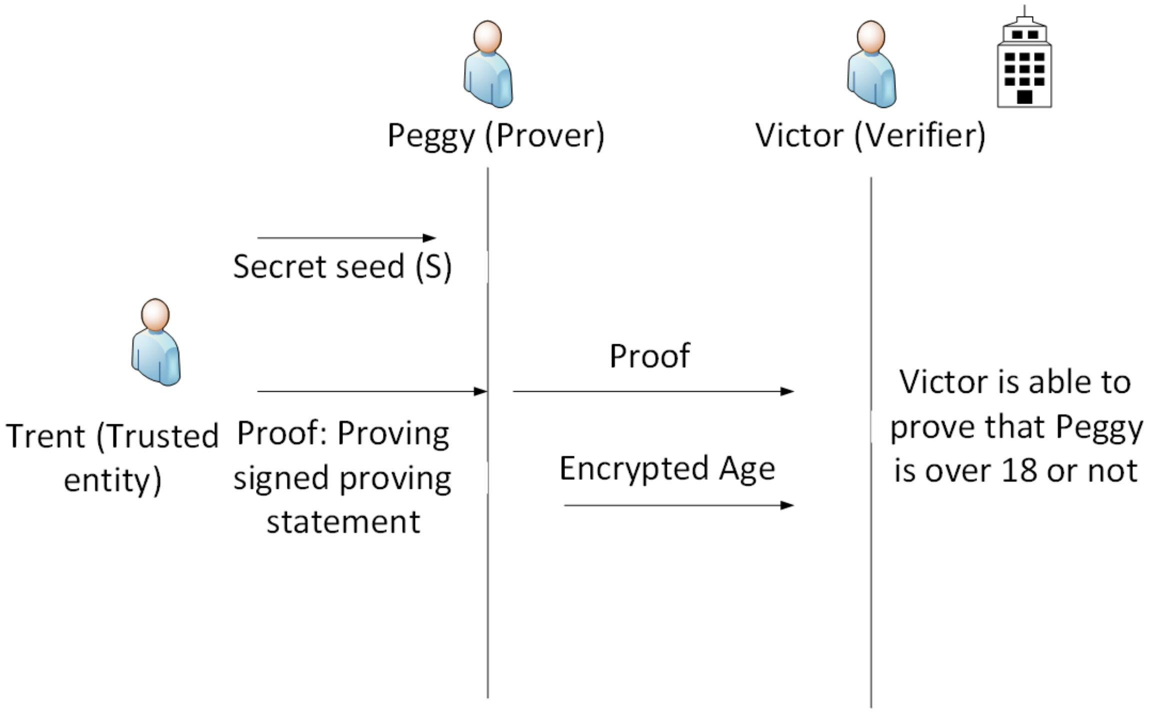
The current proof of concepts that I came across since the introduction of blockchain technology involves adding data to the etherium blockchain using smart-contracts. To give a brief overview of the technology Blockchain is an immutable distributed database with a network of nodes that are cryptographically hashed with blocks. If a user requests a transaction the user asks the blockchain. The blockchain verifies the user status and transaction using distributed algorithms. Once verified the transaction creates a new block of data in the ledger. The new block of data added exists on the blockchain which is immutable and unalterable once entered into the etherium network.


How blockchain works
Someone requests transaction.
The requested transaction is broadcast to a P2P network consisting of computers, known as nodes.
Validation
The network of nodes validates the transaction and the user's status using known algorithms.
A verified transaction can involve cryptocurrency, contracts, records, or other information
Once verified, the transaction is combined with other transactions to create a new block of data for the ledger.
The new block is then added to the existing blockchain, in a way that is permanent and unalterable
The transaction is completed

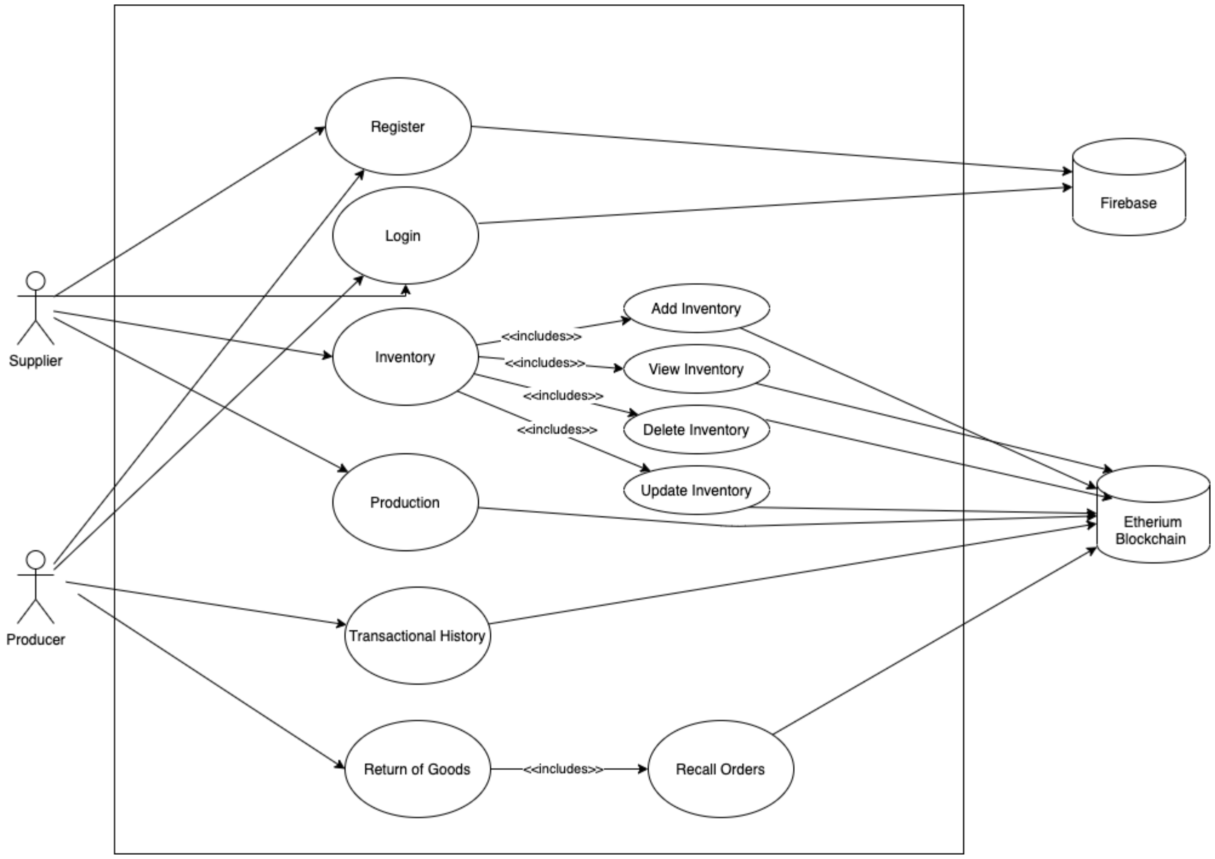
The current projects in the supply chain area focus on testing data on the blockchain however they have not focused on enhancing the privacy of sensitive data which hasn't been dealt with the existing applications. Therefore the usage of Zero-Knowledge proofs provides an additional layer of privacy which would be the focus of the project to provide anonymity to sensitive data.

The importance of protecting sensitive data on the blockchain is due to the implementation of GDPR which resulted in the importance of protecting users sensitive data which lead me to research on potential Implementation of cryptography standards to secure sensitive data from the public database where I came across a cryptographic protocol called Zero-Knowledge Proofs which hides sensitive data from the public blockchain ensuring public sensitive data Is hidden from the user providing proof the existence of the data using key id's and by using a prover and verifier construct.

The prover provides proof that information is available on the blockchain using a special keyword or id and hiding the sensitive data in a zkp which provides trust for the supplier in regards to the presence of information and thereby adhering to GDPR compliance and provides security to key security risks described in OWASP 2017 TOP 10 cyber-security risks. (2017 Top 10 | OWASP, 2020).The diagram below describes how zero-knowledge proofs works based on the prover verifier construct.


Peggy (Prover)
Victor (Verifier)
Secret seed (S)
Proof
Trent (Trusted entity)
Proof: Proving signed proving statement
Encrypted Age
Victor is able to prove that Peggy is over 18 or not

25.0 Technical Approach



Implementation of the project

The project aims to build a web application using Angular which would be used for the frontend part of the web application. Interaction with the contracts would be done using Solidity and Node.js(Javascript) which would be integrally used in the project for developing the smart contracts and integrating with the angular web application. (Solidity — Solidity 0.7.1 documentation, 2020).

Node.js and Mocha would be used to build a framework to deploy and test smart contracts developed in the project. The web application and ropsten and ganache would be the test network used to test the smart contracts for inventory, production, and location. Implementation of securing the sensitive data would be done using Zokarates which provides an easy integration of zero-knowledge interactive proof which enables the implementation of prover verifier to construct on the etherium blockchain. (Mocha - the fun, simple, flexible JavaScript test framework, 2020).

To Furthur Clarify how a Zero-Knowledge Proof would work in the web application. The first zero-knowledge proof is to work on the specifics of the supplier's login information hidden from the public blockchain by implementing the zero-knowledge proof on the login credentials by hashing the information that is sent between the client and the firebase server using the zero-knowledge proof cryptographic protocols. The reason for the utilization of the zero-knowledge proof over traditional public-key cryptography is since zero-knowledge proof doesn’t require any key exchange and it doesn’t leak information as it derives temporary keys from secret key transportation which can be enabled after authentication. (Limited, 2020).

The second zero-knowledge proof would be to hide the transaction details of the supplier and the receiver from the public blockchain while a user places an order. The current approach looks into the implements using SHA-256 with transaction details where the need to secure data to the highest Cryptographic standards is necessary for the project’s need to be implemented where all credit card numbers are marked with asterisk symbols while transmitting it over secure channels however the encryption methodologies used uses a key to encrypt data which would be enhanced with zero-knowledge proofs where the identity of the user isn't shared and users are safe from identity theft from Social Engineering attacks. (Credit Card Encryption, 2020).

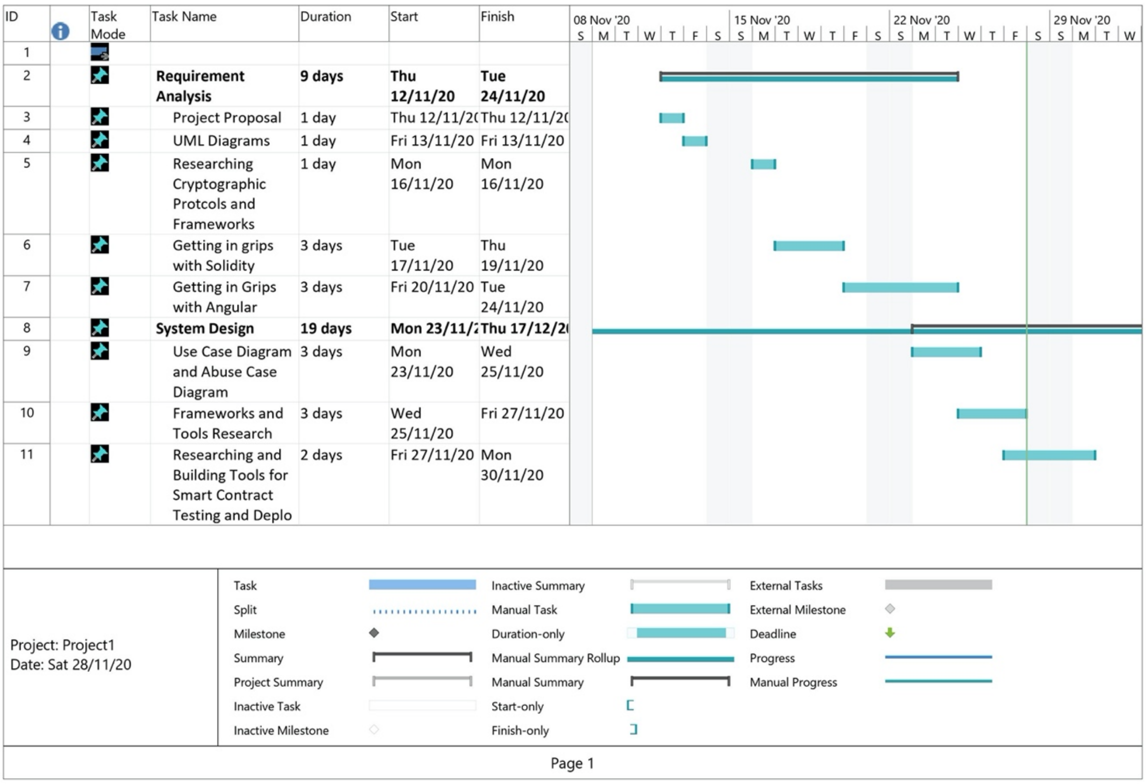
26.0 Special Resources Required

The project would involve usage of Consensys Blockchain Courses and Udemy to get in grips with Angular which would be the frontend stack for the project and use Node.js to communicate with web services within the web application and external databases. Alongside this, I would be using Solidity for the implementation of Supply chain smart contracts for the supply chain use-cases like Inventory, Production, Location, and the Return of Goods.

For enhancing the privacy on the blockchain, Zero-Knowledge Proofs (ZKP) would be used to communicate with the external data source providers where ethical standards would be considered. I aim to use my user input data in the project however I have sourced few data sources which are OPDL(Open Source Data Licences) and MIT Licences oriented data sources, Therefore relevant data issues are not of absolute concern in the project.

27.0 Project Plan

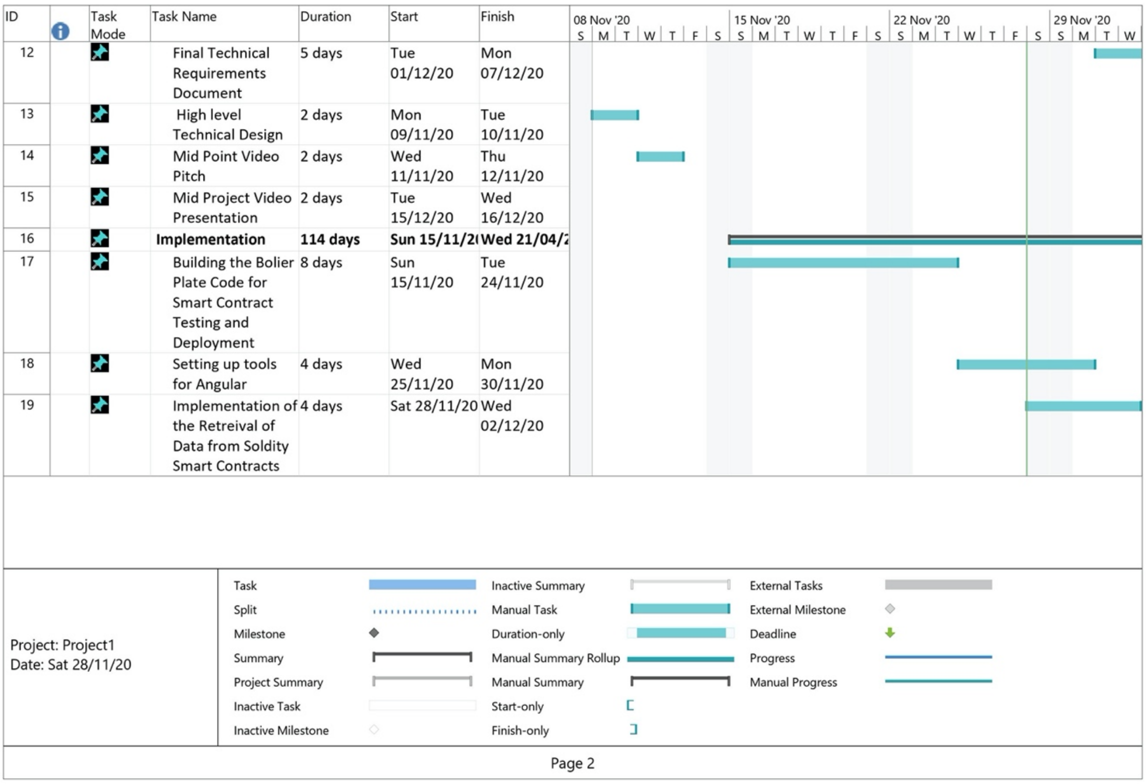
Gantt chart using Microsoft Project with details on implementation steps and timelines
Gantt chart attached within the folder



ID	Task Mode	Task Name	Duration	Start	Finish
1					
2		**Requirement Analysis**	**9 days**	**Thu 12/11/20**	**Tue 24/11/20**
3		Project Proposal	1 day	Thu 12/11/2(	Thu 12/11/2(
4		UML Diagrams	1 day	Fri 13/11/20	Fri 13/11/20
5		Researching Cryptographic Protcols and Frameworks	1 day	Mon 16/11/20	Mon 16/11/20
6		Getting in grips with Solidity	3 days	Tue 17/11/20	Thu 19/11/20
7		Getting in Grips with Angular	3 days	Fri 20/11/20	Tue 24/11/20
8		**System Design**	**19 days**	**Mon 23/11/2**	**Thu 17/12/2(**
9		Use Case Diagram and Abuse Case Diagram	3 days	Mon 23/11/20	Wed 25/11/20
10		Frameworks and Tools Research	3 days	Wed 25/11/20	Fri 27/11/20
11		Researching and Building Tools for Smart Contract Testing and Deplo	2 days	Fri 27/11/20	Mon 30/11/20

Project: Project1
Date: Sat 28/11/20

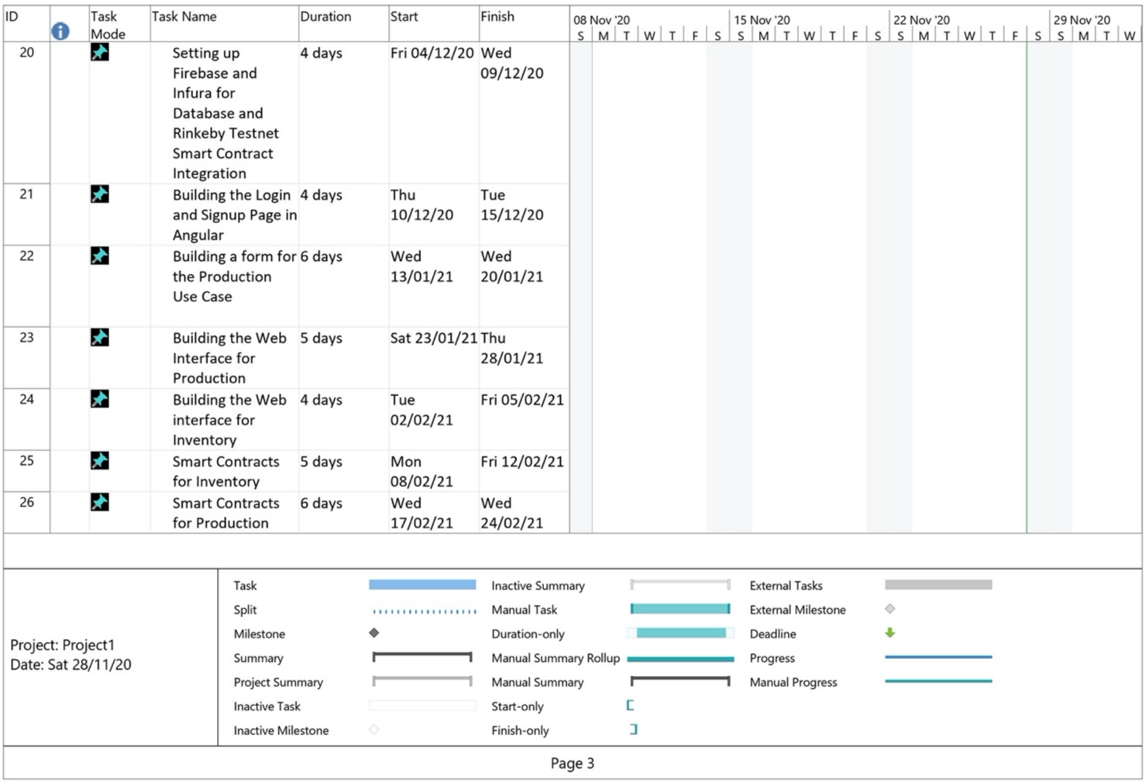
Page 1



ID	Task Mode	Task Name	Duration	Start	Finish
12		Final Technical Requirements Document	5 days	Tue 01/12/20	Mon 07/12/20
13		High level Technical Design	2 days	Mon 09/11/20	Tue 10/11/20
14		Mid Point Video Pitch	2 days	Wed 11/11/20	Thu 12/11/20
15		Mid Project Video Presentation	2 days	Tue 15/12/20	Wed 16/12/20
16		**Implementation**	**114 days**	**Sun 15/11/2(**	**Wed 21/04/2**
17		Building the Bolier Plate Code for Smart Contract Testing and Deployment	8 days	Sun 15/11/20	Tue 24/11/20
18		Setting up tools for Angular	4 days	Wed 25/11/20	Mon 30/11/20
19		Implementation of the Retreival of Data from Soldity Smart Contracts	4 days	Sat 28/11/20	Wed 02/12/20

Project: Project1
Date: Sat 28/11/20

Page 2

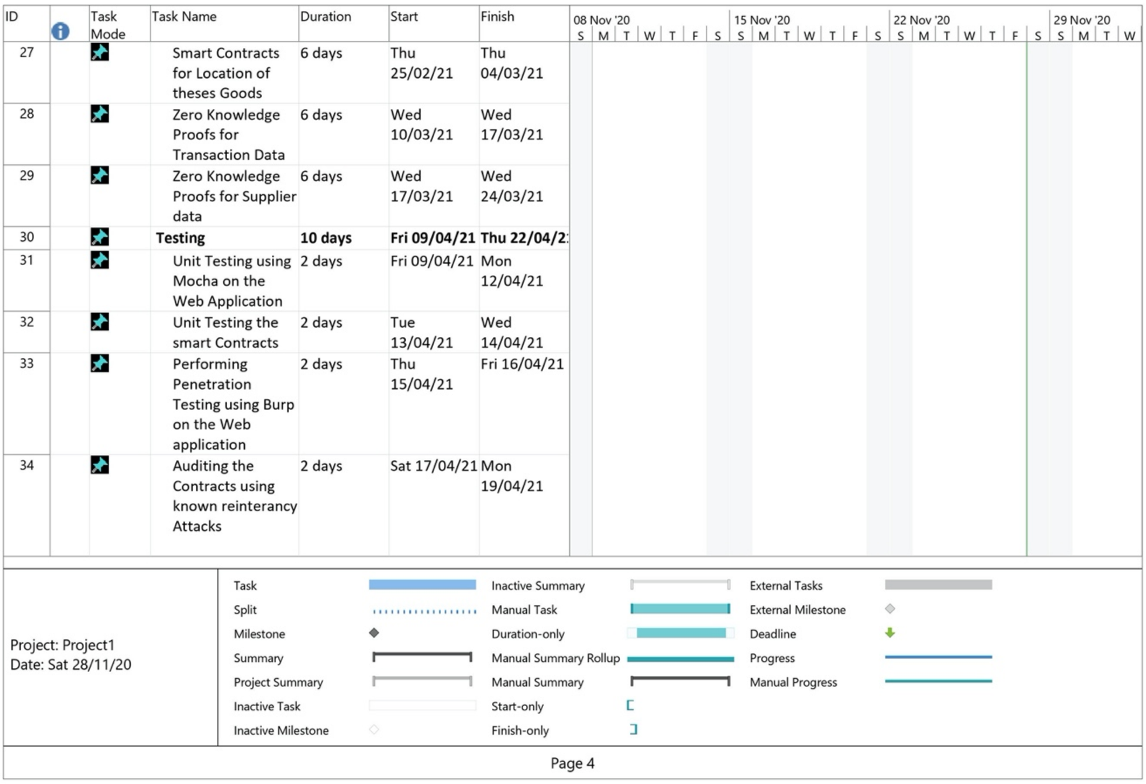


ID	ℹ	Task Mode	Task Name	Duration	Start	Finish
20		📌	Setting up Firebase and Infura for Database and Rinkeby Testnet Smart Contract Integration	4 days	Fri 04/12/20	Wed 09/12/20
21		📌	Building the Login and Signup Page in Angular	4 days	Thu 10/12/20	Tue 15/12/20
22		📌	Building a form for the Production Use Case	6 days	Wed 13/01/21	Wed 20/01/21
23		📌	Building the Web Interface for Production	5 days	Sat 23/01/21	Thu 28/01/21
24		📌	Building the Web interface for Inventory	4 days	Tue 02/02/21	Fri 05/02/21
25		📌	Smart Contracts for Inventory	5 days	Mon 08/02/21	Fri 12/02/21
26		📌	Smart Contracts for Production	6 days	Wed 17/02/21	Wed 24/02/21

Project: Project1
Date: Sat 28/11/20

Task	▬	Inactive Summary	▭	External Tasks	▬
Split	┈	Manual Task	▬	External Milestone	◇
Milestone	◆	Duration-only	▬	Deadline	⬇
Summary	┌─┐	Manual Summary Rollup	▬	Progress	─
Project Summary	┌─┐	Manual Summary	┌─┐	Manual Progress	─
Inactive Task	▭	Start-only	[		
Inactive Milestone	◇	Finish-only	]		

Page 3

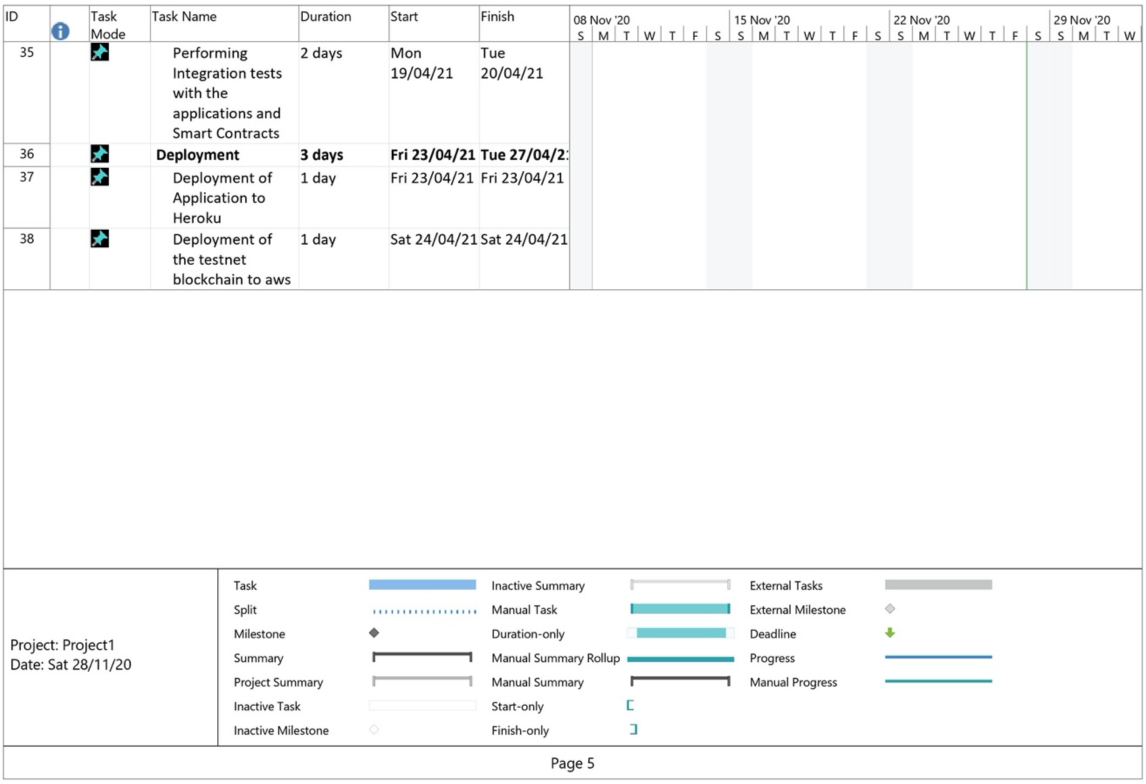


ID	ℹ	Task Mode	Task Name	Duration	Start	Finish
27		📌	Smart Contracts for Location of theses Goods	6 days	Thu 25/02/21	Thu 04/03/21
28		📌	Zero Knowledge Proofs for Transaction Data	6 days	Wed 10/03/21	Wed 17/03/21
29		📌	Zero Knowledge Proofs for Supplier data	6 days	Wed 17/03/21	Wed 24/03/21
30		📌	**Testing**	**10 days**	**Fri 09/04/21**	**Thu 22/04/2**
31		📌	Unit Testing using Mocha on the Web Application	2 days	Fri 09/04/21	Mon 12/04/21
32		📌	Unit Testing the smart Contracts	2 days	Tue 13/04/21	Wed 14/04/21
33		📌	Performing Penetration Testing using Burp on the Web application	2 days	Thu 15/04/21	Fri 16/04/21
34		📌	Auditing the Contracts using known reinterancy Attacks	2 days	Sat 17/04/21	Mon 19/04/21

Project: Project1
Date: Sat 28/11/20

Task	▬	Inactive Summary	▭	External Tasks	▬
Split	┈	Manual Task	▬	External Milestone	◇
Milestone	◆	Duration-only	▬	Deadline	⬇
Summary	┌─┐	Manual Summary Rollup	▬	Progress	─
Project Summary	┌─┐	Manual Summary	┌─┐	Manual Progress	─
Inactive Task	▭	Start-only	[		
Inactive Milestone	◇	Finish-only	]		

Page 4

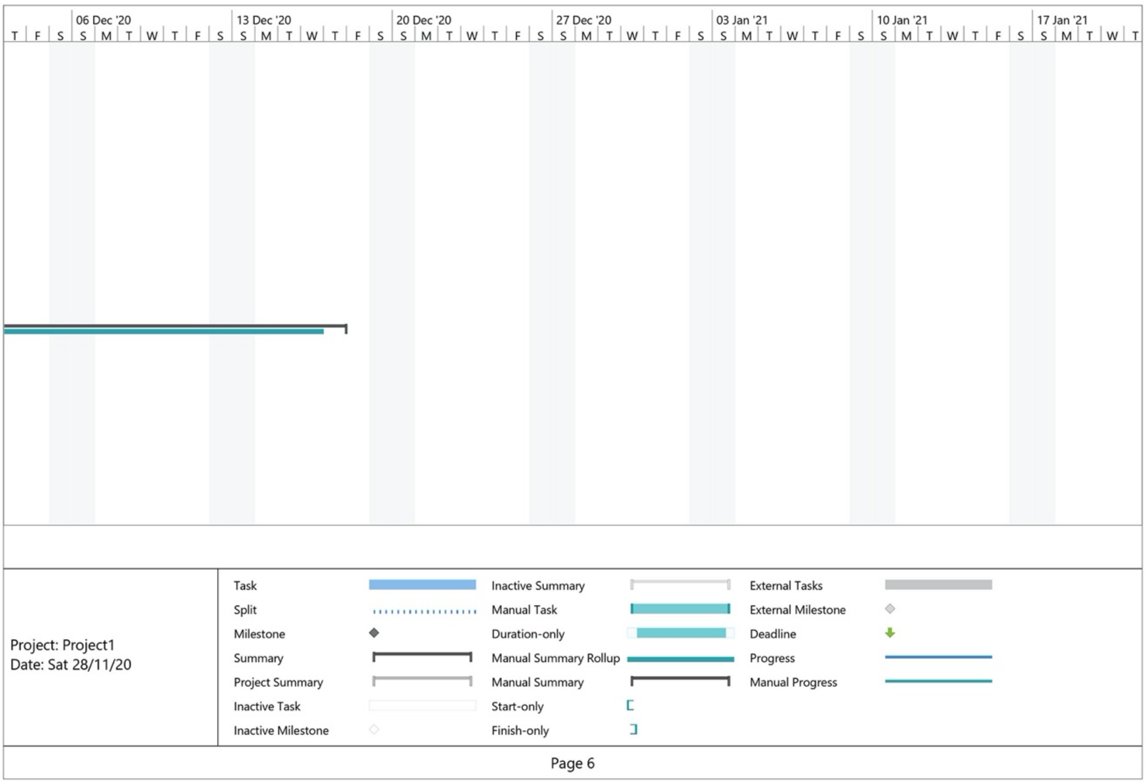


ID		Task Mode	Task Name	Duration	Start	Finish
35			Performing Integration tests with the applications and Smart Contracts	2 days	Mon 19/04/21	Tue 20/04/21
36			**Deployment**	**3 days**	**Fri 23/04/21**	**Tue 27/04/2**
37			Deployment of Application to Heroku	1 day	Fri 23/04/21	Fri 23/04/21
38			Deployment of the testnet blockchain to aws	1 day	Sat 24/04/21	Sat 24/04/21

Project: Project1
Date: Sat 28/11/20

Task	▬	Inactive Summary	⌐¬	External Tasks	▬
Split	┈┈┈	Manual Task	▬	External Milestone	◇
Milestone	◆	Duration-only	▬	Deadline	⬇
Summary	⌐¬	Manual Summary Rollup	▬	Progress	—
Project Summary	⌐¬	Manual Summary	⌐¬	Manual Progress	—
Inactive Task	▭	Start-only	[		
Inactive Milestone	◇	Finish-only	]		

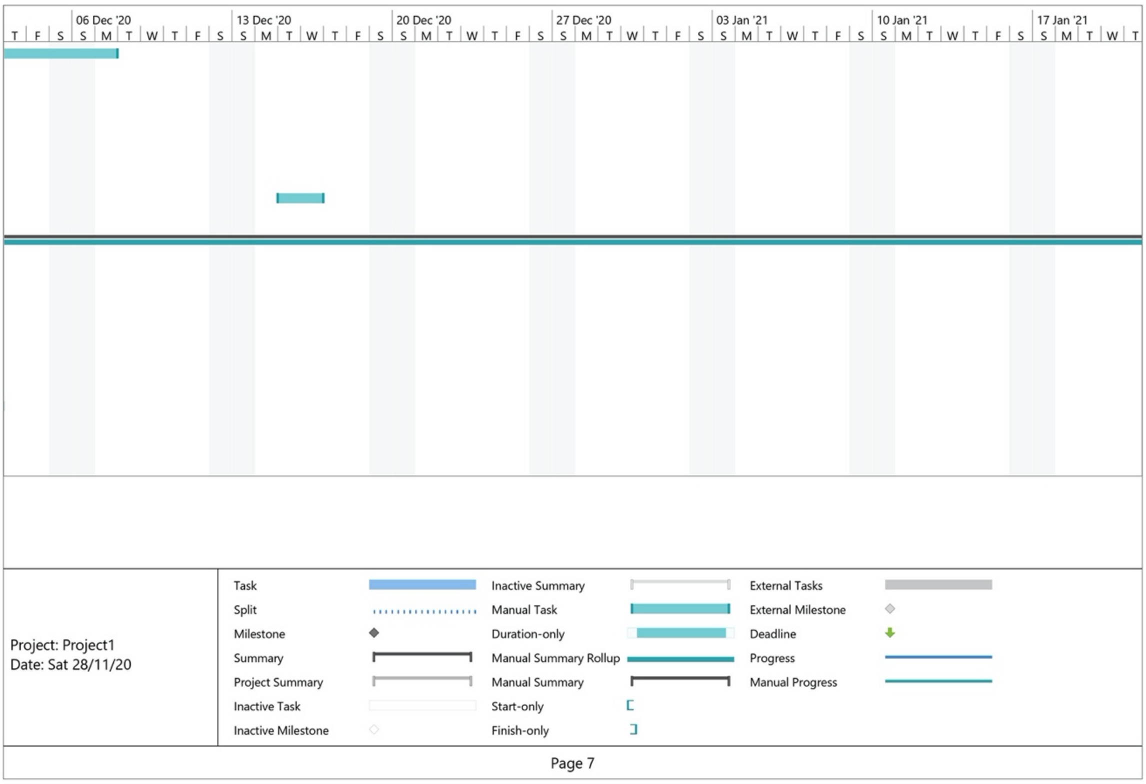
Page 5



Project: Project1
Date: Sat 28/11/20

Task	▬	Inactive Summary	⌐¬	External Tasks	▬
Split	┈┈┈	Manual Task	▬	External Milestone	◇
Milestone	◆	Duration-only	▬	Deadline	⬇
Summary	⌐¬	Manual Summary Rollup	▬	Progress	—
Project Summary	⌐¬	Manual Summary	⌐¬	Manual Progress	—
Inactive Task	▭	Start-only	[		
Inactive Milestone	◇	Finish-only	]		

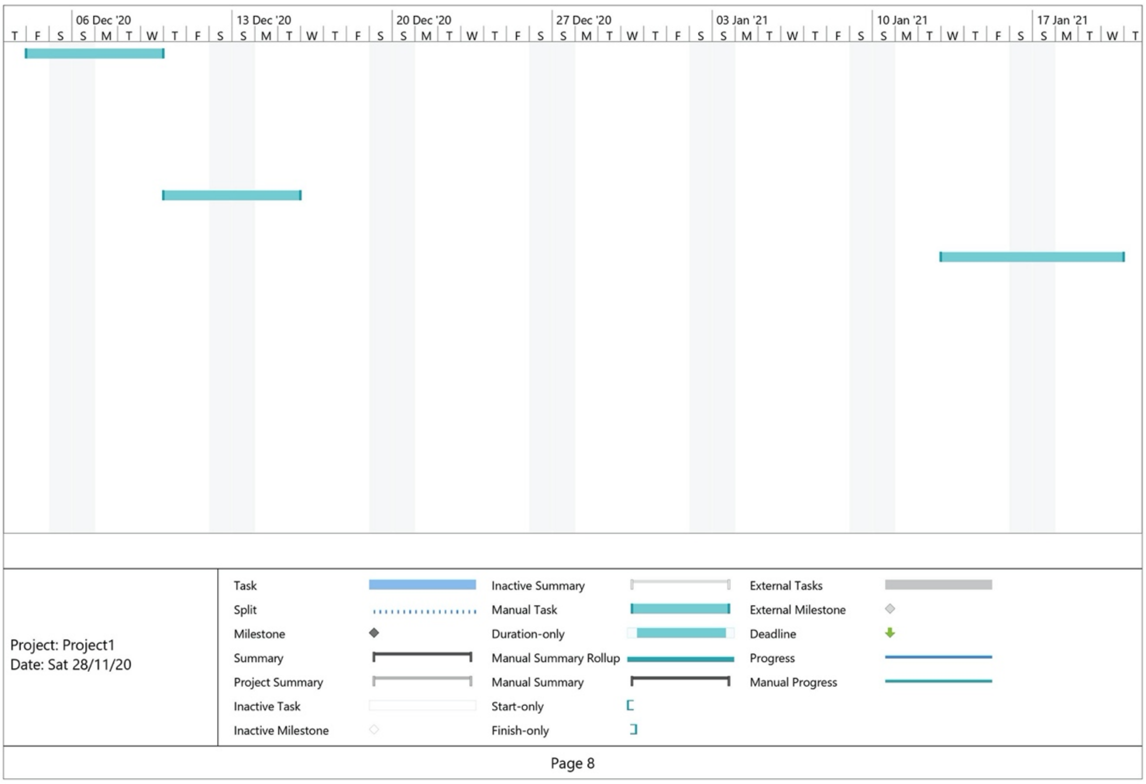
Page 6



Project: Project1
Date: Sat 28/11/20

Task	▬	Inactive Summary	┌─┐	External Tasks	▬
Split	┈┈┈	Manual Task	▬	External Milestone	◇
Milestone	◆	Duration-only	▬	Deadline	⬇
Summary	┌─┐	Manual Summary Rollup	▬	Progress	─
Project Summary	┌─┐	Manual Summary	┌─┐	Manual Progress	─
Inactive Task	▭	Start-only	[		
Inactive Milestone	◇	Finish-only	]		

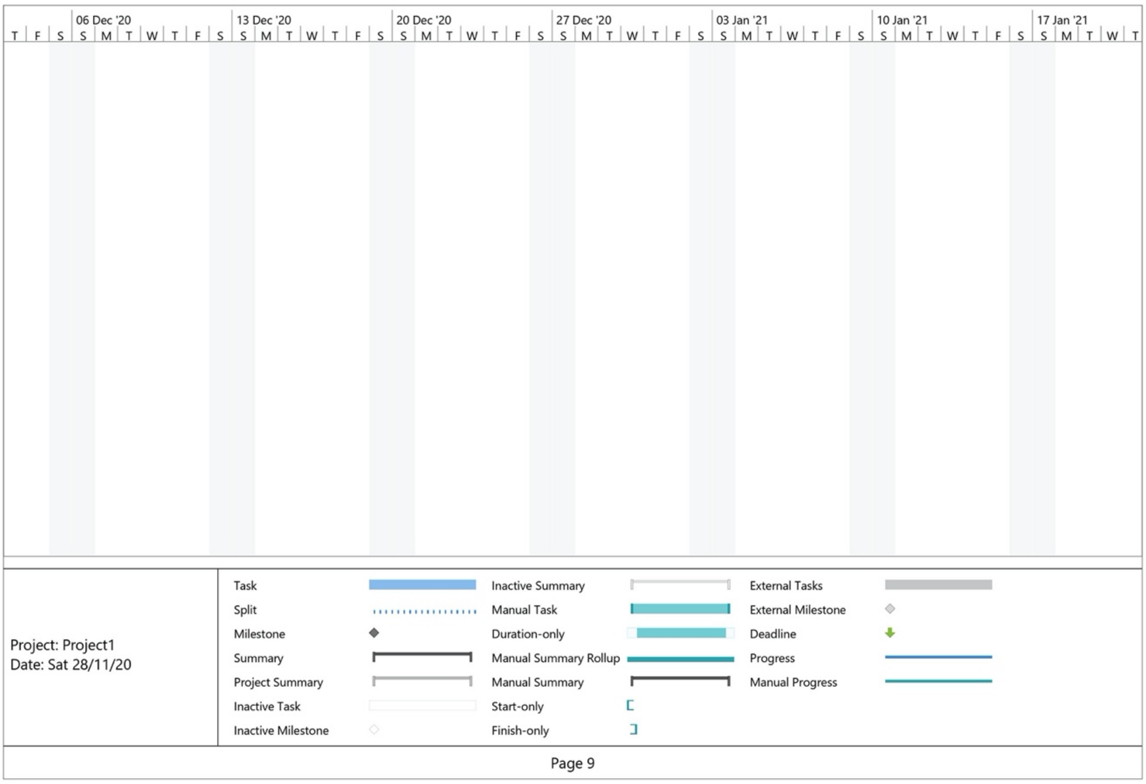
Page 7



Project: Project1
Date: Sat 28/11/20

Task	▬	Inactive Summary	┌─┐	External Tasks	▬
Split	┈┈┈	Manual Task	▬	External Milestone	◇
Milestone	◆	Duration-only	▬	Deadline	⬇
Summary	┌─┐	Manual Summary Rollup	▬	Progress	─
Project Summary	┌─┐	Manual Summary	┌─┐	Manual Progress	─
Inactive Task	▭	Start-only	[		
Inactive Milestone	◇	Finish-only	]		

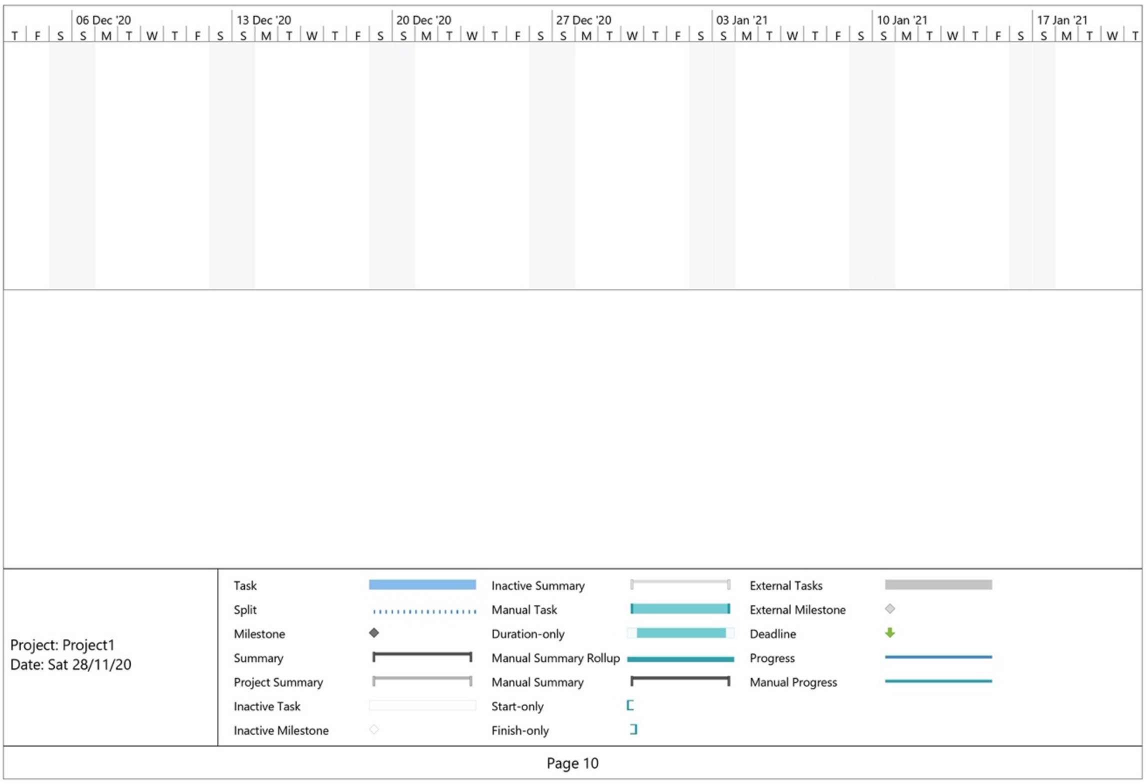
Page 8



Project: Project1
Date: Sat 28/11/20

Task	▬	Inactive Summary	┌─┐	External Tasks	▬
Split	┈┈┈	Manual Task	▬	External Milestone	◆
Milestone	◆	Duration-only	▬	Deadline	⬇
Summary	┌─┐	Manual Summary Rollup	▬	Progress	─
Project Summary	┌─┐	Manual Summary	┌─┐	Manual Progress	─
Inactive Task	▭	Start-only	[		
Inactive Milestone	◇	Finish-only	]		

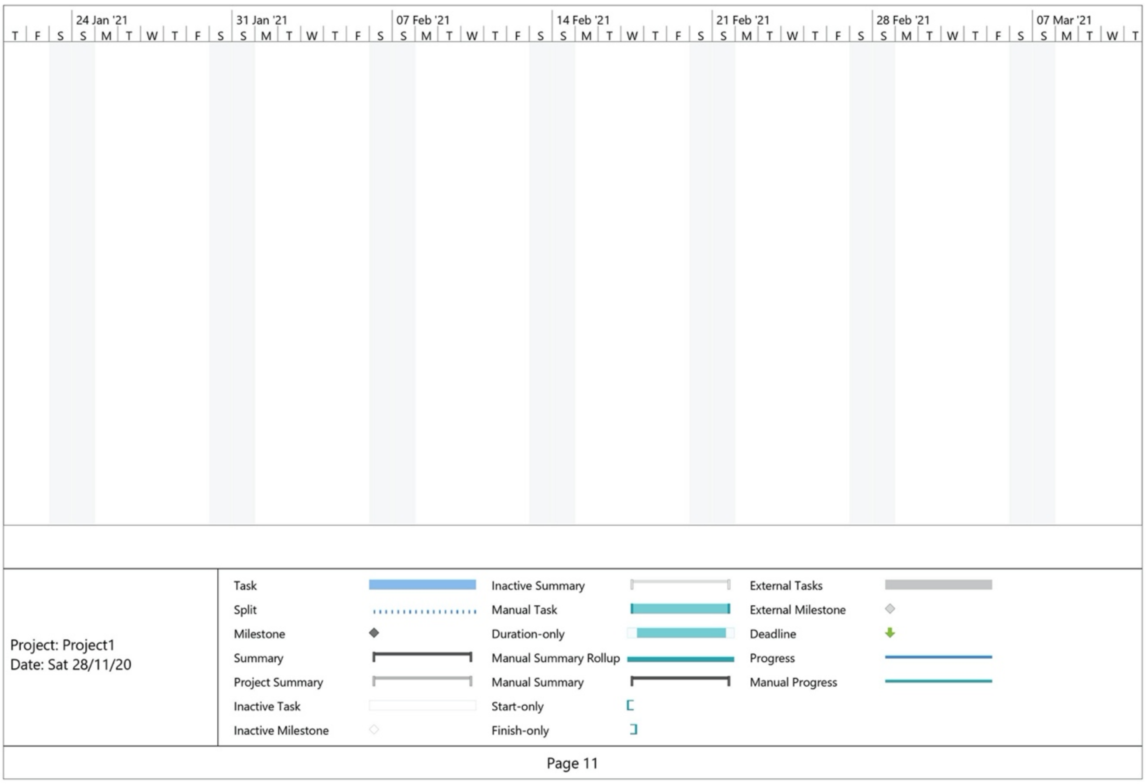
Page 9



Project: Project1
Date: Sat 28/11/20

Task	▬	Inactive Summary	┌─┐	External Tasks	▬
Split	┈┈┈	Manual Task	▬	External Milestone	◆
Milestone	◆	Duration-only	▬	Deadline	⬇
Summary	┌─┐	Manual Summary Rollup	▬	Progress	─
Project Summary	┌─┐	Manual Summary	┌─┐	Manual Progress	─
Inactive Task	▭	Start-only	[		
Inactive Milestone	◇	Finish-only	]		

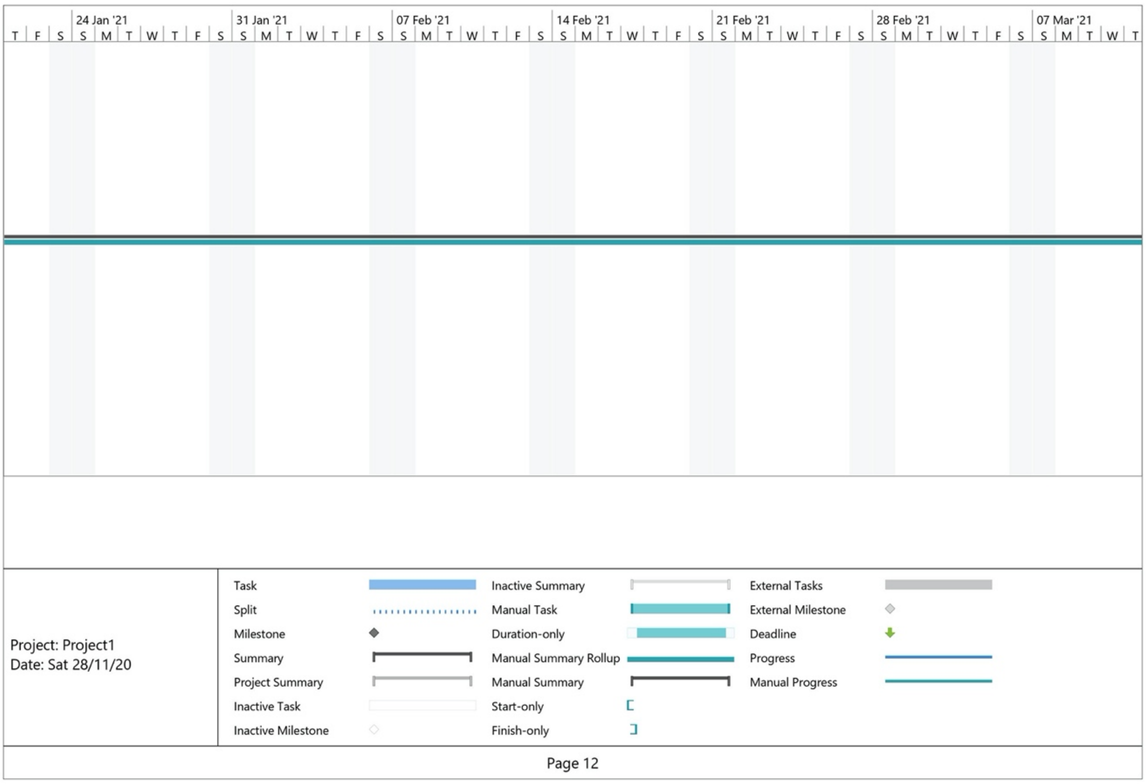
Page 10



Project: Project1
Date: Sat 28/11/20

Task	▬	Inactive Summary	⊓	External Tasks	▬
Split	┅	Manual Task	▬	External Milestone	◇
Milestone	◆	Duration-only	▬	Deadline	↓
Summary	⊓	Manual Summary Rollup	▬	Progress	—
Project Summary	⊓	Manual Summary	⊓	Manual Progress	—
Inactive Task	▭	Start-only	[		
Inactive Milestone	◇	Finish-only	]		

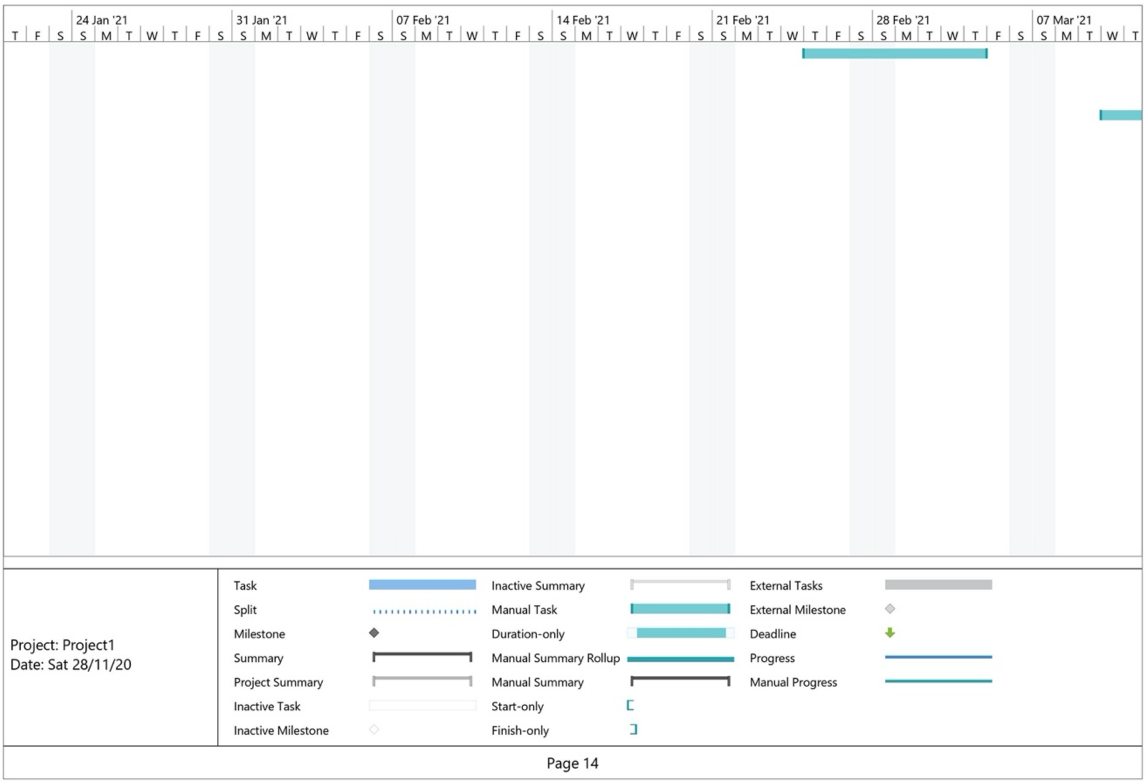
Page 11



Project: Project1
Date: Sat 28/11/20

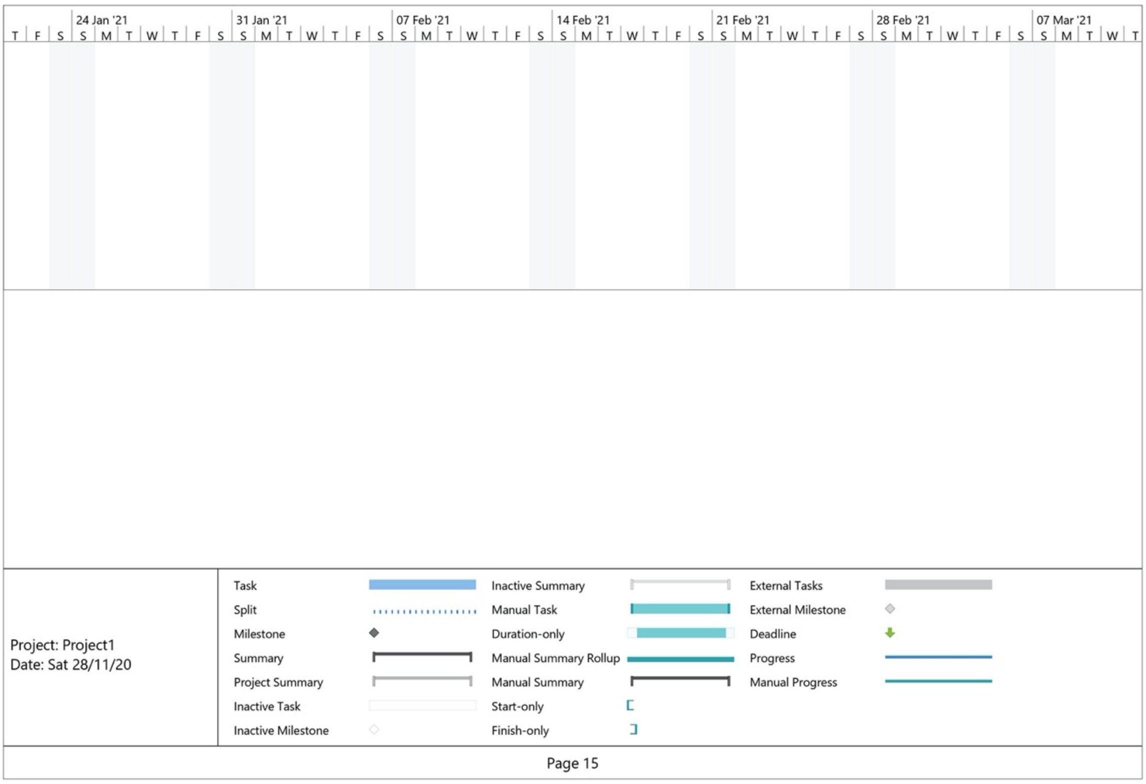
Task	▬	Inactive Summary	⊓	External Tasks	▬
Split	┅	Manual Task	▬	External Milestone	◇
Milestone	◆	Duration-only	▬	Deadline	↓
Summary	⊓	Manual Summary Rollup	▬	Progress	—
Project Summary	⊓	Manual Summary	⊓	Manual Progress	—
Inactive Task	▭	Start-only	[		
Inactive Milestone	◇	Finish-only	]		

Page 12



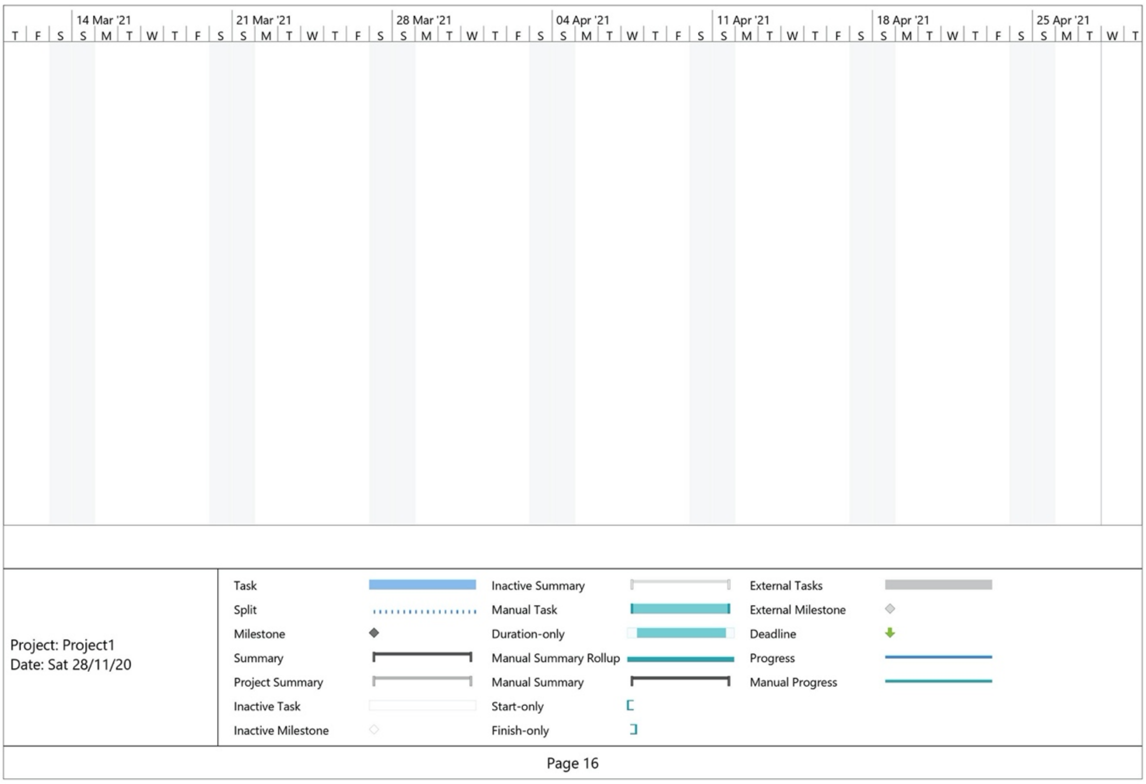
Project: Project1
Date: Sat 28/11/20

Page 14



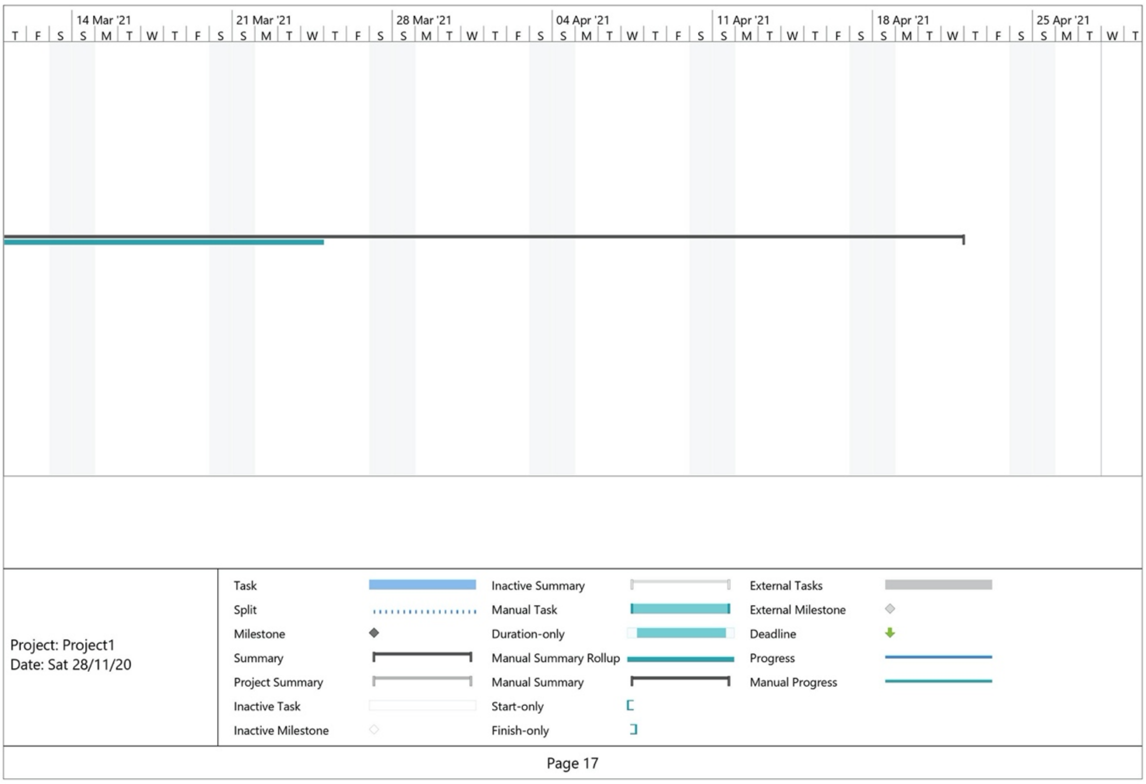
Project: Project1
Date: Sat 28/11/20

Page 15



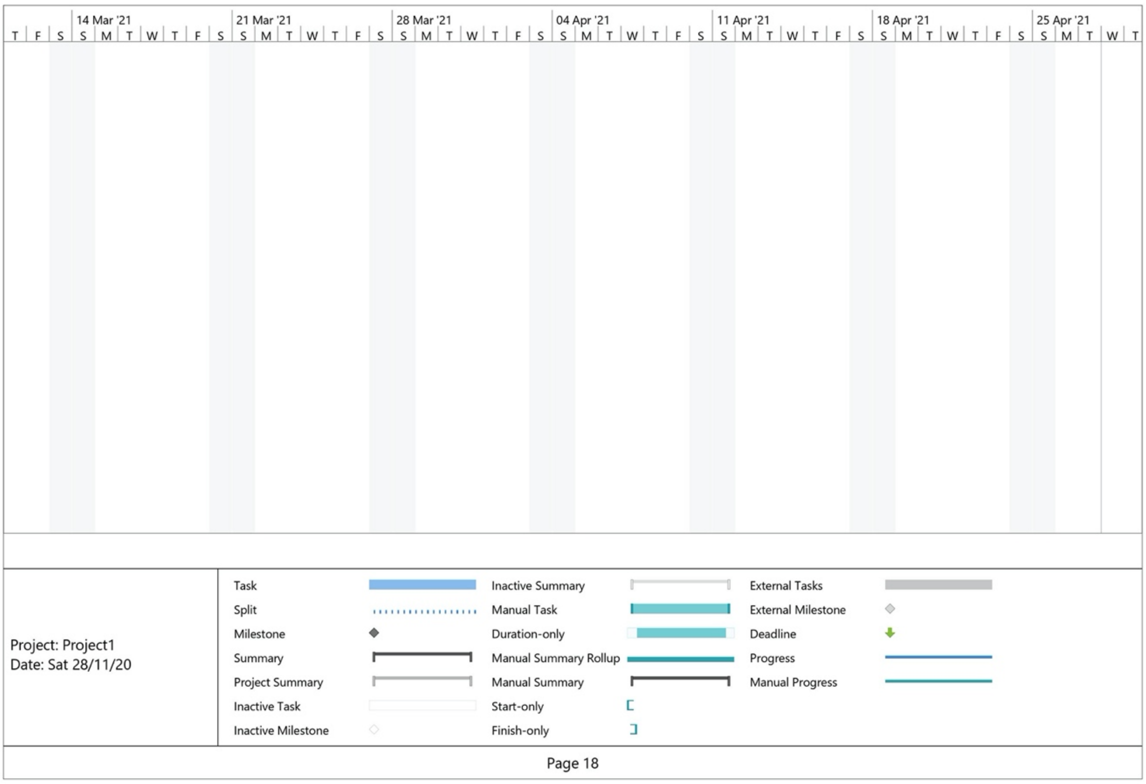
Project: Project1
Date: Sat 28/11/20

Page 16



Project: Project1
Date: Sat 28/11/20

Page 17



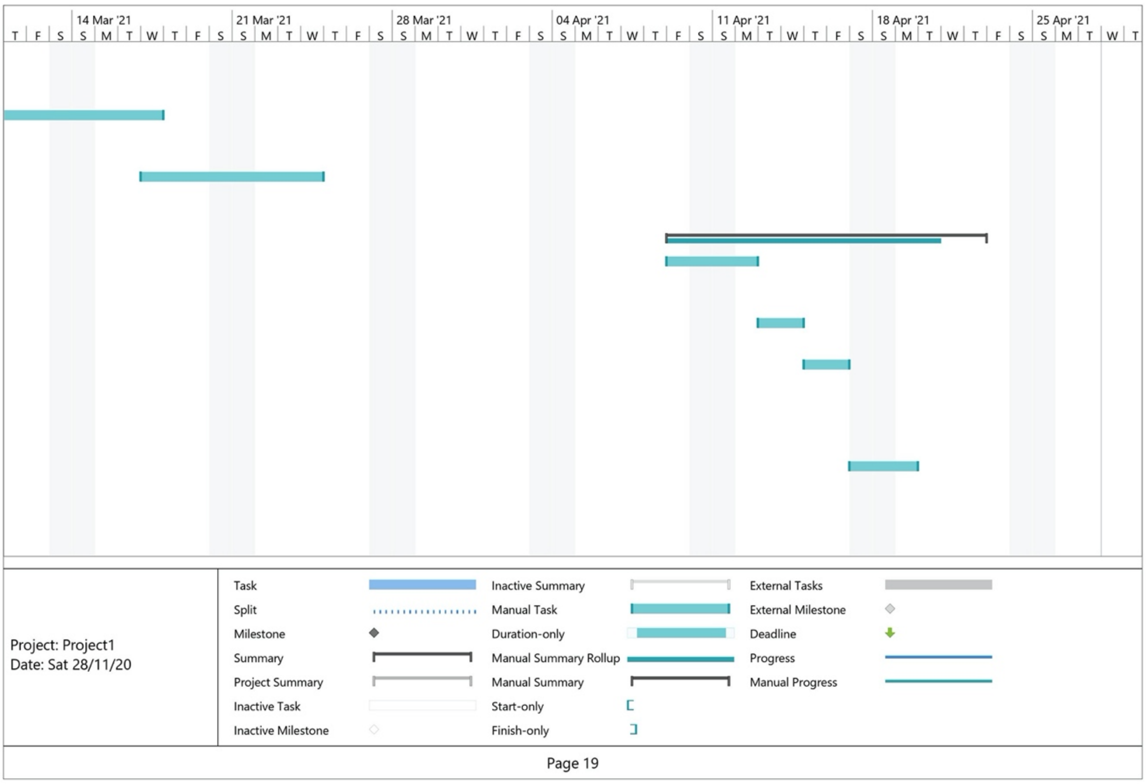
Project: Project1
Date: Sat 28/11/20

Task
Split
Milestone
Summary
Project Summary
Inactive Task
Inactive Milestone

Inactive Summary
Manual Task
Duration-only
Manual Summary Rollup
Manual Summary
Start-only
Finish-only

External Tasks
External Milestone
Deadline
Progress
Manual Progress

Page 18



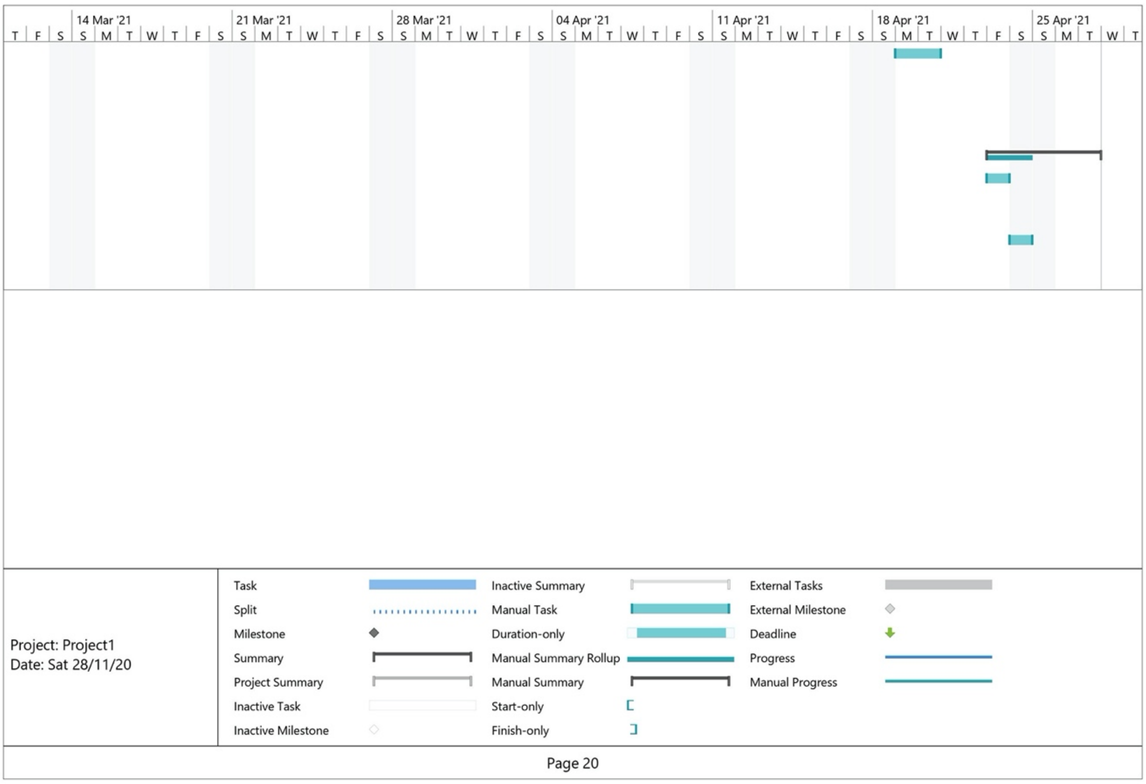
Project: Project1
Date: Sat 28/11/20

Task
Split
Milestone
Summary
Project Summary
Inactive Task
Inactive Milestone

Inactive Summary
Manual Task
Duration-only
Manual Summary Rollup
Manual Summary
Start-only
Finish-only

External Tasks
External Milestone
Deadline
Progress
Manual Progress

Page 19



Project: Project1
Date: Sat 28/11/20

Page 20

28.0 Technical Details

Implementation language and principal libraries

The Project would be a web application that would be built using Angular. Angular will be used as the front end of the web application as the UI Layer of the web application. Solidity will be used for the implementation of the project's smart contracts. Ganache would be used initially to test the smart contract on the local ropsten test net before deploying it to the public live testnet.

Angular frontend would be using the routing modules that would be used to route between web pages for example adding, deleting, and updating products on the inventory system.Zokrates(https://zokrates.github.io/) will be used for the implementation of cryptographic protocols to encrypt sensitive data as described earlier in the proposal using the verifier and supplier contract implementation.Web3.js integration will be an integral part of the project interacting with the smart contracts and the Public blockchain where it would act as an intermediary converting the smart contract code into an EVM byte code which thereby interacts with the public blockchain and also integrates well with the truffle test environment.

Mocha and Chai would be used to test the web application and bcrypt protocols will be added to the encryption of login credentials on the web application.

29.0 Evaluation

Unit tests will be written in each implementation of the smart contract using mocha and Jest frameworks which can be used to test the angular UI as well as test the implementation of the smart contracts using hooks like before,beforeEach, after all, after each and ensuring these tests validate these conditions would help in reducing bugs within the code which complies with development principles in Smart Contract Development(Test Early and Often).

The evaluation criteria for testing the application would be Testing the Web Application for SQL Injections and Cross side scripting, Smart Contract testing would be done based on Consensys Secure Development Recommendation Criteria which looks into Compliance with security flaws in smart contracts which is integral in ensuring common security attacks like reentrancy attacks or integer overflow attacks is considered while developing the web application Furthermore, works like avoiding state changes after external calls and not handling errors in external calls will ensure the reliability of the smart contracts against potential attacks before deploying the contracts to the testnet. (Consensys.github.io. 2020).

The Evaluation criteria will be assessed with the Qualitative DREAD Security Assessment Model and a Score would be provided in the Final Documentation which would provide a qualitative analysis of security concerns and compliance with OWASP Secure Coding Principles and Consensys Smart Contract Development Guidelines. (Owasp.org. 2020).

Finally, Performing Penetration tests and using Secure Software Development Principles and Standards on the web application will ensure privacy is of the utmost concern in the development of the project.

30.0 Bibliography

1. Consensys.github.io. 2020. *Secure Development Recommendations - Ethereum Smart Contract Best Practices*. [online] Available at: <https://consensys.github.io/smart-contract-best-practices/recommendations/> [Accessed 6 November 2020].
2. Owasp.org. 2020. *2017 Top 10 | OWASP*. [online] Available at: <https://owasp.org/www-project-top-ten/2017/Top_10> [Accessed 2 December 2020].
3. Barrett, B., 2020. *How 4 Chinese Hackers Allegedly Took Down Equifax*. [online] Wired. Available at: <https://www.wired.com/story/equifax-hack-china/> [Accessed 2 December 2020].
4. Docs.soliditylang.org. 2020. *Solidity — Solidity 0.7.1 Documentation*. [online] Available at: <https://docs.soliditylang.org/en/v0.7.1/> [Accessed 2 December 2020].
5. Owasp.org. 2020. [online] Available at: <https://owasp.org/www-pdf-archive/OWASP_SCP_Quick_Reference_Guide_v2.pdf> [Accessed 2 December 2020].
6. Angular.io. 2020. *Angular*. [online] Available at: <https://angular.io/guide/forms> [Accessed 2 Decembers 2020].
7. Investopedia. 2020. *Credit Card Encryption*. [online] Available at: <https://www.investopedia.com/terms/c/credit-card-encryption.asp> [Accessed 2 December 2020].
8. Mochajs.org. 2020. *Mocha - The Fun, Simple, Flexible Javascript Test Framework*. [online] Available at: <https://mochajs.org/> [Accessed 3 December 2020].
9. Limited, C., 2020. *Zero-Knowledge Protocols Without Magic*. [online] Cossacklabs.com. Available at: <https://www.cossacklabs.com/blog/zero-knowledge-protocols-without-magic.html> [Accessed 21 December 2020].