Deliverable #1

The Project Plan

By:

Curatolo, Christopher
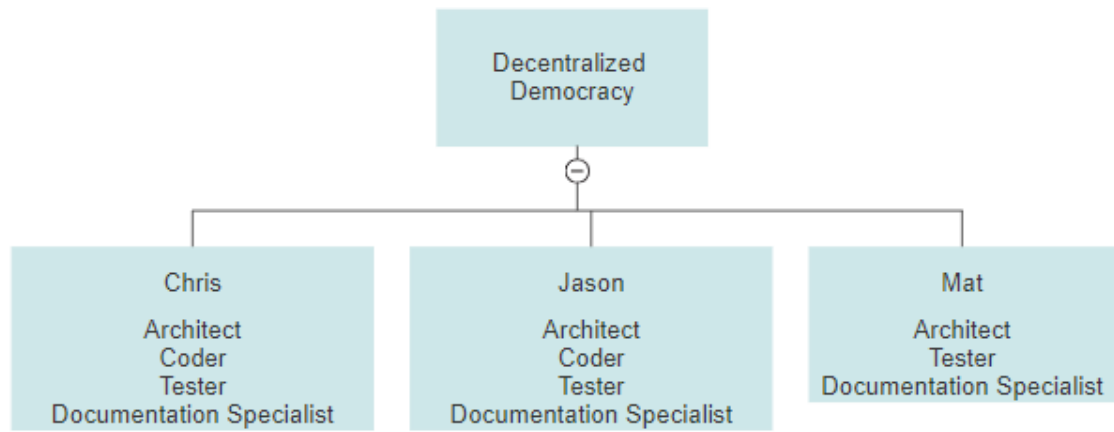
Harms, Mathew

Howse, Jason

Scope Statement

I.   In short, the project is an NFT (non-fungible token) decentralized application (Dapp). The application has three main stages: the frontend takes user input of a file and their crypto wallet address, the frontend then connects to the backend (the Ethereum blockchain) and then mints an NFT, and outputs a successful transaction and creation of the NFT on success. The client's newly created NFT is the proof of digital ownership of their uploaded file.

II.   What type of platform must the software work with?
   a. Both the desktop application and the web-based application will require a network connection. The web-based application will be usable on any device with browser software in addition to the network connection.

III.   Will the software function as a standalone application on a given computer, or will it function over a network connection?
   a. This software will include both a desktop application as well as a web-based application that will run in the event the user system is not compatible with the desktop version.

IV.   What other software, if any, must the software interact with?
   a. The desktop application uses no other software and relies on a network connection to connect to the Ethereum blockchain on the backend. The web-based application will require an internet browser for functionality.

V.   What programming language will be used for the project?
   a. C# for the backend of the desktop application
   b. Solidity for the creation of the NFT and smart contract deployment
   c. JavaScript/Typescript for frontend
   d. HTML for frontend
   e. XAML for user interface

VI.   Will the software use a graphical interface or a command line interface?
   a. Both applications, the desktop application and the web-based version, will include graphical user interfaces.

Org Chart

Decentralized Democracy

Chris

Architect
Coder
Tester
Documentation Specialist

Jason

Architect
Coder
Tester
Documentation Specialist

Mat

Architect
Tester
Documentation Specialist

Gantt Chart

	Start	Duration	End
Research Phase	9/7/2021	25	10/2/2021
Research - Implement github	9/15/2021	7	9/22/2021
Back End Phase	9/11/2021	81	12/1/2021
Back End - Environment Setup	9/11/2021	30	10/11/2021
Back End - Connect to iPFS	9/29/2021	7	10/5/2021
Back End - Create Contract	9/29/2021	7	10/5/2021
Back End - Deploy Contract to Eth	10/5/2021	8	10/13/2021
Back End - Store File	10/8/2021	3	10/12/2021
Back End - Connect to front end	10/13/2021	40	11/22/2021
Back End - Return NFT	10/25/2021	8	11/2/2021
Back End - Mint NFT	11/18/2021	8	11/26/2021
Front End Phase	10/7/2021	55	12/1/2021
Front End - Utilize Wallet	10/17/2021	8	10/25/2021
Front End - GUI Implementation	10/19/2021	11	10/30/2021
Front End - Web Functionality	10/28/2021	25	11/22/2021
Front End - App Functionality	10/28/2021	25	11/22/2021
Testing Phase	11/15/2021	19	12/4/2021
Testing - Executable Design/Creation	11/27/2021	3	11/30/2021
Testing - Backend Unit Testing	11/18/2021	13	12/1/2021
Testing - Frontend Unit Testing	11/18/2021	8	11/26/2021
Testing - Full System Testing	12/1/2021	3	12/4/2021
Testing - Documentation	11/18/2021	16	12/4/2021

Chart Title
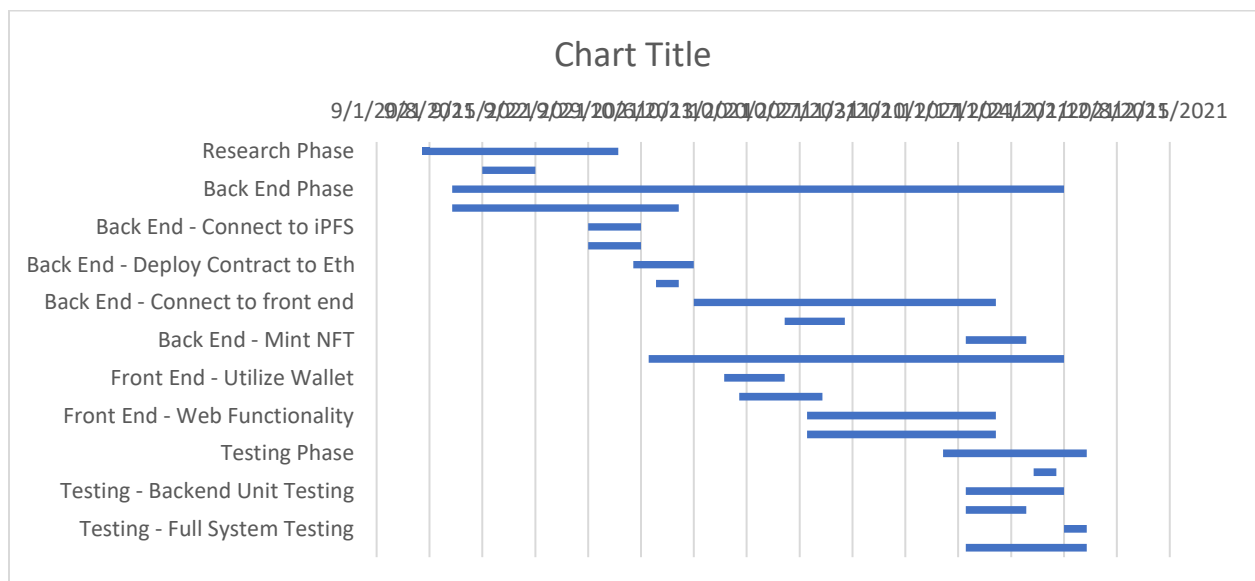
Tools and Standards

Tools used:

IDE: Microsoft Visual Studio Community 2019
TargetFramework: netcoreapp3.1
SDK: Microsoft.NET.Sdk.WindowsDesktop
Image Editor: Adobe Photo Shop Elements 13

Dependency Libraries:(NuGet Packages)
"Ipfs.Http.Client" Version="0.33.0"
"Mime" Version="3.2.3"
"Nethereum.Contracts" Version="4.0.5"
"Nethereum.Hex" Version="4.1.1"
"Nethereum.Web3" Version="4.0.5"
"Newtonsoft.Json" Version="13.0.1"
"xunit.assert" Version="2.4.1"
(Check Licenses)

Local Test Environments:
Truffle v5.4.21 (core: 5.4.21)
Solidity - ^0.8.0 (solc-js)
Node v14.17.1
Web3.js v1.5.3
Ganache v2.5.4.0

Documentation:

UML
Flow Designer

Configuration Management Plan

Version #	Changes
0.0.1	Creating smart contract and initalizing connection to eth network and IPFS
0.0.2	Creating testing environment to verify file storage pre NFT minting
0.0.3	Implementation of initial web interface
0.0.4	Connecting web interface with connections to eth network and IPFS
0.0.4.1	Deeper GUI build with button functionality to handle various connections
0.0.5	Desktop application GUI creation and connection to backend
0.0.6	Creation and implementation of creative design in GUI and file packaging
0.0.7	Implementation of the NFT minting process
1.0.0	**Production release**
1.0.1	Icons for executables implemented
1.1.0	polling to live update the combo box after mint timing
1.2.0	Add configuration files
2.0.0	Future release of file transfering/nft transfering from wallet to wallet
3.0.0	Formating of the IPFS pin

Weekly Status Reports

Group 13 Project Status Report #1

1. What was accomplished?

We have begun to learn the technology we need to use in the project and completing administrative tasks to facilitate completing the project such as setting up the dev environment using docker and setting up Jira for issue tracking.

2. How each group member contributed to the project?

Chris began Setting up the dev environment to be used to build the project. Chris also gave me and Mat material to look into to get up to speed with the technology used to build the project. I created the baseline Jira boards for use in issue tracking etc. Mat and I are currently studying Docker and IPFS which are related to this project and we are currently unfamiliar. We are using online tutorials to practice building basic level projects utilizing the basic technologies.

3. What problems were encountered?

Currently we haven't experienced any issues. IAW your notes on the scope statement providing a dummy account won't be an issue.

Group 13 Project Status Report #2

1. What was accomplished?

We have been setting up the dev environment, architecture and planning out scope and continuing to explore the technology.

2. How each group member contributed to the project?

Mat and I have been studying the Ethereum Blockchain and IPFS stuff and Chris has continued setting up the dev environment and architecture and setting up the repository.

3. What problems were encountered?

I had concerns using the actual Ethereum blockchain costs but Chris assured me we can use an online Test Ethereum network that won't have a cost associated with it.

Group 13 Project Status Report #3

1. What was accomplished?

We have been working collectively on our workflow, project management, version control, and sandbox environment.

2. How each group member contributed to the project?

Each member has created an account with github.com for our version control and project management. We all have been active in the reposiroty and project managaement boards. Also, we have a repository for our sandbox environment because we want to work on our skills and workflow.

3. What problems were encountered?

Currently, the git workflow is new to all of us. We have been working researching and using git together, fixing the problem.

Group 13 Project Status Report #4

1. What was accomplished?

We've worked to distribute efforts; Chris has worked to put together repos for our dev environment in ethereum, Jason has begun putting together the frontend framework, and Mat has started code review and is in the early stage of unit testing. That backend framework that Chris put together will give us some reference points as we begin to build out the smart contracts, ethereum network connections, etc.

2. How each group member contributed to the project?

Individual contributions were included in #1

3. What problems were encountered?

Currently we haven't experienced any issues.

Group 13 Project Status Report #5

What was accomplished?
Set up prototypes of the system in github.

How each group member contributed to the project?
Mat has been looking into unit testing and integration testing for C# and Solinity. Chris has been deploying environments to Github. Jason has been working on integrating IPFS solutions.

What problems were encountered?
IPFS connections have been somewhat difficult to integrate into a C# project directly.

Group 13 Project Status Report #6

What was accomplished?
IPFS get and pull methods developed allowing for all project elements is now easily implemented.

How each group member contributed to the project?
Jason Implemented IPFS get and pull methods on the front end. Chris loaded and configured a mac dev environment to git hub. Chris created a fronted application today that uses the Nft.Storage api.

Mat is researching methods for testing using unit tests on c# and solidity contracts.

What problems were encountered?
Configuring the API for the IPFS push was difficult because of a lack of documentation on how to correctly configure it. I ultimately figured out how the system was expecting the API info. I also had difficulties with implementing MIME and getting it to properly recognize different file types but ultimately got it to function.

Group 13 Project Status Report #7

1. What was accomplished?

Created a semi functional UI.

2. How each group member contributed to the project?

Jason began building the Desktop front end UI portion of the app. Chris built a web-based UI front end.

3. What problems were encountered?

None

Group 13 Project Status Report #8

What was accomplished

Determined precise nature of project and posted about a half complete project to github. Discussed different tacts and technologies to use. Scrapped c# project in favor of NodeJS project.

How each group member contributed to the project

Chris wrote most of the NodeJS project and posted it to github. Jason and Mat have been looking into the related technologies trying to get up to speed on it.

What problems were encountered

None in particular.

Group 13 Project Status Report #9

What was accomplished

Determined final plans for the production pushes. We determined we are going practice diversity in coding by attempting to complete the project in both WPF C# and NodeJS with React.

How each group member contributed to the project.

Jason tried to study Storybook JS and comprehend the documentation associated with the and applicable to the project. Chris worked on cleaning up the code that is already present for the nodeJS project. Mat began looking into different documentation and planning methods to use.

What problems were encountered

None in particular.

Group 13 Project Status Report #10

What was accomplished

Most of the project has been completed, The project color scheme has been selected. The contracts have been worked on (the database).

How each group member contributed to the project.

Jason has gotten all parts of the c# project online except the ethereum functionality. Chris worked designing the contract used by the project. Mat has been working on design elements of the project such as icons and color schemes.

What problems were encountered

Getting contracts deployed to local ethereum environment.

Group 13 Project Status Report #11

What was accomplished

The project has all but been complete.

How each group member contributed to the project.

Jason completed the WPF front end. Chris all but completed the back end. Mat created icons and logos associated with the project.

What problems were encountered.

n/a

Group 13 Project Status Report #12

What was accomplished

The project was completed and submitted including documentation.

How each group member contributed to the project.

Mat worked on all Deliverables #1 and some Deliverables #2 items. Jason completed and deployed the desktop apps and completed and compiled many parts of all the Deliverables. Chris completed the back end and deployed it to production. He also added documentation to his code.

What problems were encountered

Determine there was a bug in the desktop app that is imposible to fix before the due date.