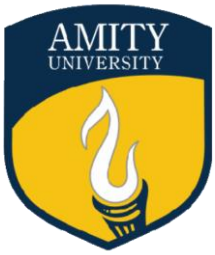
In House Report

On

"**Design Of Blockchain Based Cryptocurrency Application**"

Submitted to:

**Amity University Uttar Pradesh**




In partial fulfilment of the requirement for the award of the degree of

**Bachelor of Technology**

In

**Computer Science and Engineering**

by

**Pranav Kumar**
**(A2305221263)**

Under the guidance of

**Dr. Garima Aggarwal**

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
AMITY SCHOOL OF ENGINEERING AND TECHNOLOGY
AMITY UNIVERSITY UTTAR PRADESH

---

**DECLARATION BY STUDENT**

This is to confirm that I, PRANAV KUMAR student in the B. TECH (CSE) application of the 2021-2025 Batch at the Amity School of Engineering and Technology, Amity University Uttar Pradesh, Noida, have worked for the NTCC summer internship under the direction and supervision of DR. GARIMA AGGARWAL. This report meets the requirements for the Graduate Degree in B.Tech. in Partial Fulfillment (CSE). To the best of my knowledge, this report's contents are the result of original research, and no text has been taken verbatim from any other study. I am aware that Amity School of Engineering and Technology has the right to revoke the report in the event of noncompliance.

# AMITY UNIVERSITY
## UTTAR PRADESH

On the basis of declaration submitted by Pranav Kumar, student of B. Tech (CSE), we hereby curtly that the project titled "Create Dashboard Using Power BI" which is submitted to Department of Computer Science & Engineering. Amity School of Engineering and Technology, Amity University Uttar Pradesh. Noida, in partial fulfilment of the requirement for the award of the degree of BACHELOR OF TECHNOLOGY in Computer Science and Engineering, is an original contribution with existing knowledge and faithful record of work carried out by them under my guidance and supervision.

To the best of my knowledge, this work has not been submitted in part or full for any Degree or Diploma to this University or elsewhere.

**Dr. Garima Aggrawal**
**Associate Professor**
**Department of Computer Science and Engineering**
**Amity School of Engineering and Technology**
**Amity University Uttar Pradesh**

**ACKNOWLEDGEMENT**

The happiness that comes from knowing a work was completed successfully would be inadequate without mentioning the individuals whose tireless collaboration made it possible and whose ongoing direction and support successfully crowned all efforts. I want to express my gratitude to my faculty mentor, DR. GARIMA AGGARWAL, who has been the main impetus behind my successful completion of the project. She has always been there to answer any questions I may have had and to point me in the correct path for the project. I couldn't have finished the assignment without her assistance and motivation. Additionally, I want to thank my fellow classmates for all their support, suggestions, and encouragement during the process.

# INDEX

# LIST OF FIGURES

## Chapter 1 - Abstract

As you know that web 3.0 is our future and all our website and application should be decentralized. This decentralization makes our data more secure and safe because now there is no one owner of it, and the technology which this decentralized application use is the blockchain. Today there are so many applications deployed on blockchain like your cryptocurrency, crypto token, NFT and many marketplaces like open sea. Basically blockchain is the chain of blocks in which every block have the data and its own hash and if someone changed the data then hash will be changed. We can easily convert some data into hash but to change hash into data is next to impossible. In this report I created crypto token on ICP live blockchain. The main difference between crypto token and crypto currency is that crypto token use existing blockchains and whereas crypto currency use its own blockchain but both can be used in exchange and in equity. My crypto token name is Kalki which is based on Krishna kalyug avatar. I use Motoko as backend to make this token I deployed it on ICP live blockchain I used various tools modules to make this token like dfx and react.

*Fig 1. Image of NFT*

We added some functionality to this crypto token. This token can be reedemable by new user who is authorized with internet identity we added internet identity as security for login to your account. This token is also tranferrable to different user accounts and also you can check balance of your account. In making of this token I learned many skills like react for front end devlopment and also motoko for backend, dfx tool for creating, deploy.there are two type of deployment on blockchain first is local deployment in which DApp deployed locally other user on network can't use that but this deployment is free of cost no cost needed to deploy. And the second deployment is on network, in this deployment every user on internet can easily interact with your DApp but this deployment cost you very high, some blockchain like etherium cost you very very much that normal devloper can't manage. The icp blockchain cost you less than other blockchain and that's the main reason for choosing ICP blockchain. On ICP you need cycles to deploy your project and you can get 20T cycle free for first time after that you need to buy ICP token and burn them to get cycles. Every devloper on ICP have their own default identity to which wallet is linked and you can redeem your cycles into that wallet and use them for deployment. To start devlopment on blockchain you need to install linux system and DFX command line tool and link MOTOKO with your VScode. Now you ready to deploy your first DApp on ICP. This project is really fun it tooks me 20 days to complete this project, I learned many skills and things from this project. I used resources like IC docs and my course to get this project done.

## Chapter 2 - Introduction

What is Crypto Token? A crypto token is a digital currency or assets which created on blockchain technololgy. It represent utility or a value within a platform. Cryptoken initially provided by initial coin offerings.unlike cryptocurrency which built on their own blockchain, crypto token built on existing blockchain.crypto token can be used in various purposes like they can be an asset of company and can be trade on exchange. They can also use in showing the ownership of any service or assets. Token are divided mainly into two types- 1. utility token  2. security token.

1. Utility token - utility token basically used for providing right to any service or product. You can use that utility token in that service or product like icp token which can be used to create cycle and help in deploying the project on the blockchain. This type of token not for trading or exchange and for investment.
2. Security token – this type of token are digital asset which show the ownership of any company asset or real- world assets this token are for investment and trade on  crypto exchange.

Token price is not constant. The price will be changed according to demand and supply. If supply is low and demand is high price will go up and if the supply is high and the demand is low price will go down. This price action makes token an asset.

How to start blockchain devlopment? Now you know so much purpose of crypto token the next question come in mind is how to create them, for creation of crypto token or any devlopment on blockchain you need to learn various skills and do courses for blockchain development.before you choose any course first decide that on which blockchain you want to work like in etherium or bitcoin or ic after deciding choose the course respectively. I choose ic blockchain for my devlopment there are many reason to choose ic as blockchain main reason is that live deployment on ic network is cheaper then other blockchains. After that you have to learn blockchain programming language like python, motoko etc motoko is a new programming language for devlopment on ic block chain so I learn motoko as my blockchain programming language. After this you have to set up integerate all your environment workspace to make everything working.

According to me or stats blockchain devlopment is the one of the highest paying job in the world and as we know web 3.0 is the future and the peopls get more conciouse about data safety and security. In no time soon world will tranfer to the web 3.0 generation of decentralized applications that cant be owned by single owner or single entity every user have access to that application equally which make them more secure and safe and next to impossible for hack and steal or edit the data. Web 3.0 is the web next generation that not come completely yet so we have to make ourselves ready for this generation and technology as the computer science student.

KALKI – 10$^{th}$ incranation of lord vishnu, our crypto token name is kalki.you are thinking of why these name but something should be left secret. Now lets talk about some features and functionality of our own crypto token which take about 20 days to complete. Our crypto token platform or wallet name is KALYUG and there is an login authentication to go to your wallet that authentication is from internet idenity and this ajuthentication is special because unlike others google facebook signin these big platform take your data. But with internet identity your data is protected on your finger tip your wallet or account can only be accessed by your internet identity like fingerprint or mobile phone these makes you wallet account secure from fake login. After login to your account you see the kalyug platform which give you some functionality over your own crypto token. The first section is of free reedemable 10000 kalki token. You can claim these free 10000 token under some conditions like you cant claim again and again only one time you can claim to your identity or your account. The second section is of check balance in which you can enter your internet identity and check your current account balance and you also check balance of any internet idenity on that blockchain that using our kalki crypto token. The last section on our platform is of transaction or transfer token in these section you can give your token to anyone of your friends and mates you just need their internet identity and enter the amount of token you want to give and click on tranfer and all done that amount of token will be deducted from your account and added on that principal id to which you send your crypto token. This kalyug platform is decentralized deployed on

ic blockchain these means no one own that platform everyone on that ic is the owner of kalyug. Now you guys are thinking of that how i created all this section in platform. That was preety simple as we go further I will tell you every step in methodology section of these report. But for now in simple words I just use react as frontend and connect all that event listed on frontend end to backend and in my backend motoko function are working every section have their own backend function these section call their backend function from the frontend simply by using import export method in react. I know this is little confusing but don't worry as we go further you will get to know everything about how and what I created. Internet computer blockchain work on canister all our backend code store in one canister and deploying of that canister on live chain have some cost but I have my free cycles from dfinity faucet you can also claim your free cycle from dfinity faucet and start creating your own crypto token

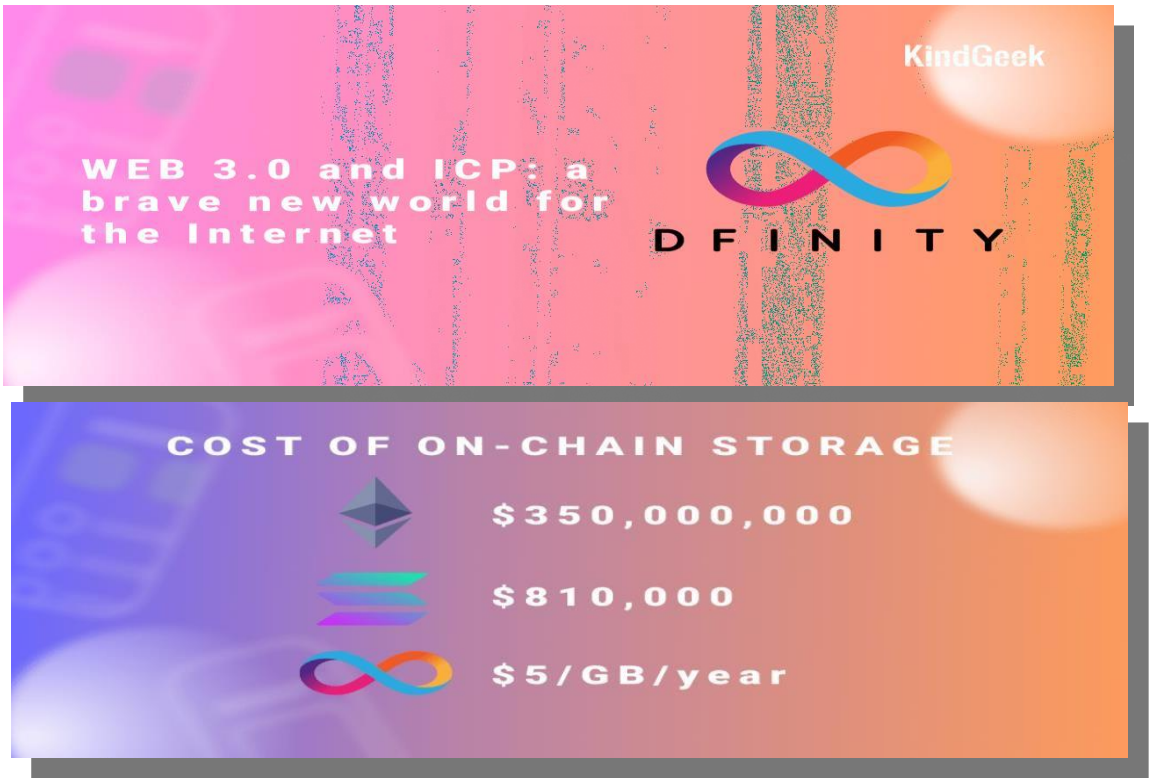



*fig 2. Information about dfinity*

## Chapter 3 - Literature Review

To be honest I don't know so much about history and literature about cryptocurrency and blockchain. I used many resources and references to tell you about literature of cryptocurrency.

Bitcoin is not the first digital currency the first digital currency was created in 1980 by a man named David Chum and his project was called digi caash following this we had other project such as eagled hash cash and be money but none of these project succeeded and one of the main reason because they were centralized they can be hacked or shut down at any moment in addition these digital cash projects were not able to solve the problem of double spending this is where a person takes the same piece of digital currency and they can spend it on two three or even more places so this would never work practical but then in 2008 following the financial recession a person or a group by the name of santoshi nakamoto released the bitcoin white paper or a guideline so the first decentralized cryptocurrency that solved the double spend problem still dont know who santoshi nakamoto is until this day at one point people believed it was the man dorian nakamoto but the proof the evidence just did not hold up and it does not really matter who santoshi nakamoto is bitcoin is already decentralized and its already running. [2]So wether we fin out who it was or who that person what that person what their reputation is it wont make a difference but just for curiousitty sake many people do believe that santoshi nakamoto was the man hal finny who Is early user in bitcoin space but mr Howle finny is no longer alive with us today. The actual bitcoin blockchain did not start until january 3rd 2009 with the genesis block where the first bitcoin block that was added to the blockchain and you might be wondering who was using bitcoin in 2009 and it might be as no surprise to you it was not popular at all so the people using it were the cypher punks and the libera tyrians these were the people looking for privacy these were the people looking for a form of money outside the government control and in the beginning there was no exchanges there was no buying no selling the way people got bitcoin by using their regular desktop regular cpus to mine the bitcoin it was very simple it wasn't until may of 2010 that the first bitcoin transaction occurred for a tangible asset

at the time in may of 2010 a man bought two pizzas for the price of 10000 bitcoin which at the time was not very much this man name was laszlo hanok and today people laugh at this man the reason they laugh at him because 10000 bitcoin at the time now is worth about 80 million dollars so people laugh at this man but the truth is without this important transaction bitcoin might not have ever reached $1 you could make the case that this was the transaction that really changed bitcoin that really made it a legitimate currency and following this people started to take notice in 2010 a man by name of jed McCaleb he released the mount Gox bitcoin exchange where people can buy and sell bitcoin and at one point in 2014 mount Cox accounted for 70% of bitcoin transactions but in 2014 mount Gox was hacked and we really don't know if it was hacked or if it was stolen or what went but 740 thousand bitcoin were lost and never seen again and during this time where people are realizing that this bitcoin this digital currency could be used people are willing to buy and sell it so this man on the screen here Boss Ulbricht he had the idea of creating the silk road which was an underground in the market where people can buy preety much anything illegal that you can think of so this was in 2011 that this man released the silk road but in 2013 it was shut down they found out that Ross albrecht was the person and running this they shut down the website and they forced him to give up and they seized his bitcoin so with all of this going on cryptocurrency well main really bitcoin started to gain some popularity in the tech world or in the computer science world and following this Talaq butyrin was working at bitcoin magazine and he liked bitcoin he was a thin but he saw some problems with it so in 2014 he decide to create and announce his own project called etherium which would take bitcoin which was just simply digital money and he would elevate it and he would start to build smart contracts and etherium was touring complete you can program on it and this is where peoples mind started to open up in 2014 they start to realize there is so much more to this whole space of crypto currency and blockchain and from this in 2014 and moving on we startto see big devlopment a big hype around all coins all of these projects these companies these block chains they had these new tokens with very specific use cases and with all these new coins coming out we also had new exchanges coming out it was not just mount

gox anymore or these little small exchanges we had coin base which really emerged from the space gained a lot of respect and by nantz as well so now people were buying and selling crypto currencynow moving back to etherium project with the etherium project it was all of these ideas of smart contracts and ways to raise money so etherium had somethimg know as a decentralized autonomous organization and they wanted to raise money through it with smart contract craft with smart contract but I believe it was in 2016 in july of 2016 the dow hack occurred the money that was tied up in this smart contracts was stolen and this created an big problem again this was in 2016 and people said we want our money back but we know with blockchain the whole concept is that its final once it occurs that's it its over so what etherium decided to do was something known as hard fork and they split from the chain so that's why today we have etherium classic and etherium and from here\ we started to see other projects follow this concept hard forks of seperating which is why today we had bitcoin bitcoin cash bitcoin SD bitcoin diamond bitcoin gold and list goes on so this was in 2016 but then as cryptocurrency start gaining some mainstream popularity 2017 was the year of total hype this is when cryptocurrency exploded maybe the greatest bull market of any asset in the history of the world this is when bitcoin reached almost $20000 the hype was real there was memes everywhere people were talking about it to their uber drivers they were talking about it at the dinner table and the hype got so crazy thatr companies wanted to participate in something as ICO or initial coin offering this was around the time of 2017 and 2018 as well and this was an easier way for a company to raise money rather than going through the traditional wall street route and it was so easy all you had to do was throw up a white paper give an idea and you'd raise money and majority of these projects were a scam and many people lost money so in 2018 as governments as theworlds start realizing that theres a lot of risk in this field this is when the financial institutions this is when the government stepped in this this is when we had people like jamie dimon the CEO of chase coming out or JPMorgan coming out and saying that bitcoin is a fraud and this was the idea this was the mindset across financial institutions this is when we saw the SCC starting to get involved with cryptocurrency trying to regulate it this is when

we saw how governments and countries around the world such as India such as China being very hostile to bitcoin and cryptocurrency so we went from this hype in 2017 to 2018 this major crash and people had a big problem with cryptocurrency but as time went on and we went into 2019 this is where we saw some of the financial institutions and governments actually coming on to the good side of bitcoin having a positive outlook is where we saw companies suc as BAC devloped which is create by the same parent company as in new york stock exchange this is when we saw fidelity one of the oldest most trusted and respect legacy institutions entered the crypto space so this is 2019 we're seeing a shift we are seeing governments coming out and making rules and regulations on crypto currency some fair and some very unfair to keep it simple. in 2020 what is the new thing we talked about just exchanges in the beginning all coins we talked about icos we talked about regulations whats going on now is defi decentralized finance this is the movement of taking traditional applications in finance and making them decentralized this includes projects for example such as lending and borrowing and insurance so this is where we are at in 2023 the history of cryptocurrency is preety short

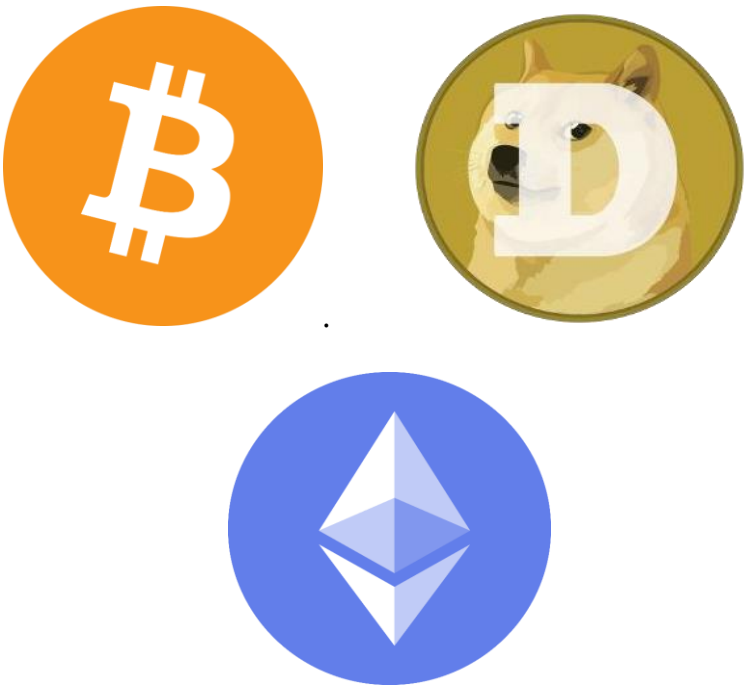


*Fig 3. Some cryptocurrency*

## Chapter 4 - Methodology

In the methodology section we will talk about methods or steps we used to build our crypto token if you want to build your own crypto token from scratch then this section is very bemeficial for you. For creation of crypto token you first need to install or integerate some tools or languages into your system just follow the following step to install all main resources into your system:

**Step 1:** First you have to install IC SDK on your system. IC SDK is tool used for creating, deploying and managing all your canister on internet computer blockchain.Canister are like blocks of code. Remember you can not nstall IC SDK on windows you need a linux system to install IC SDK and if you are window user you can install wsl window subsystem for linux to install IC SDK. To install IC SDKon your system you have to write following command:

```
sh -ci "$(curl -fsSL https://internetcomputer.org/install.sh)"
```

*Fig 4.1. installing dfx*

This command will install latest dfx into your system.

**Step 2:** Now you installed dfx successfuly then the next step is to install vs code and integerate motoko. Motoko is programming language that used in internet computer blockchain devlopment. Motoko is fresh and new it is easy to learn and easy for devlopers to use. You can integerate motoko easily by istalling extension of motoko from vs code.

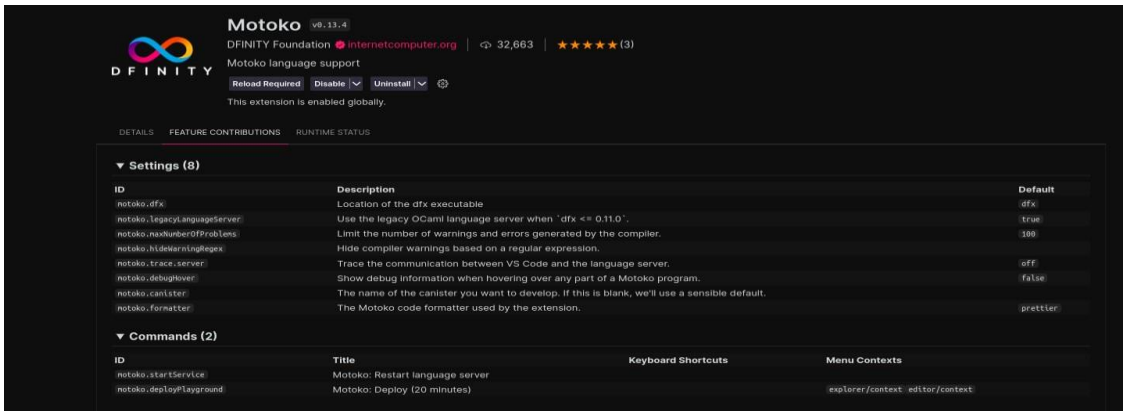



*Fig 4.2. Installing Motoko*

After installing all of the above you are ready to go for your first small Dapp to deploy on internet computer block chain just follow the next step to deploy your first Dapp on internet computer.

**Step 3:** In these step you will create your first DApp. First go the the directory in which you want create your project then write command “dfx new Token”.


*Fig 4.3. creation of all important files*

This will make token folder and generate all important files, modules used in project.

Now open vs code and change directory to Token folder. You will see all of this files folders there.

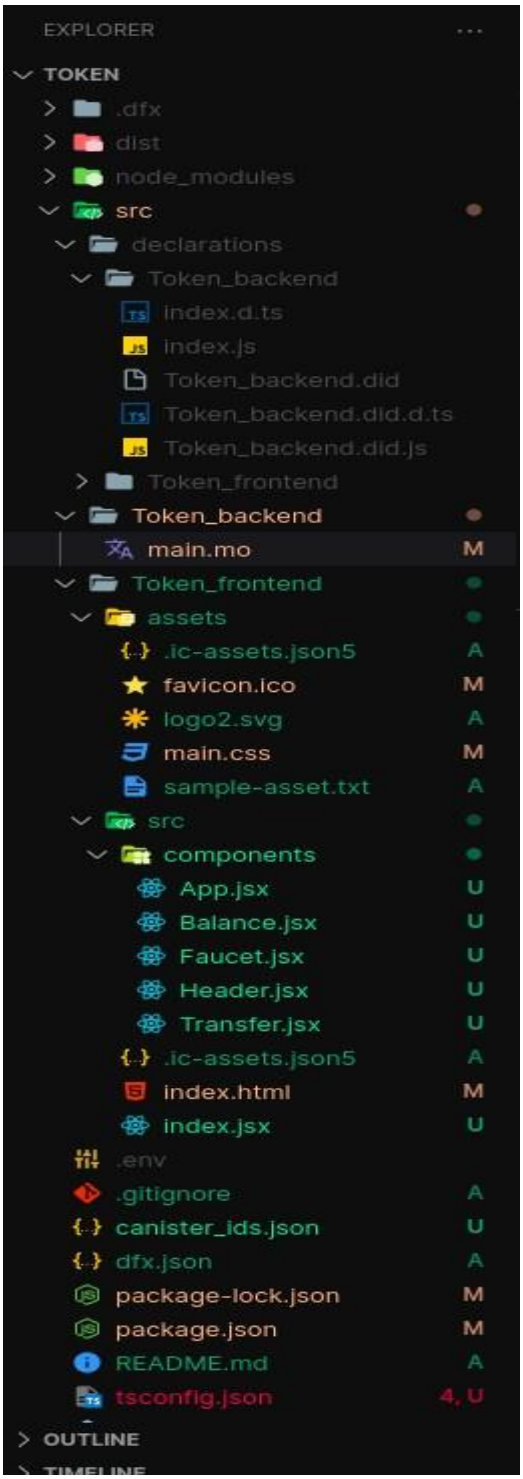
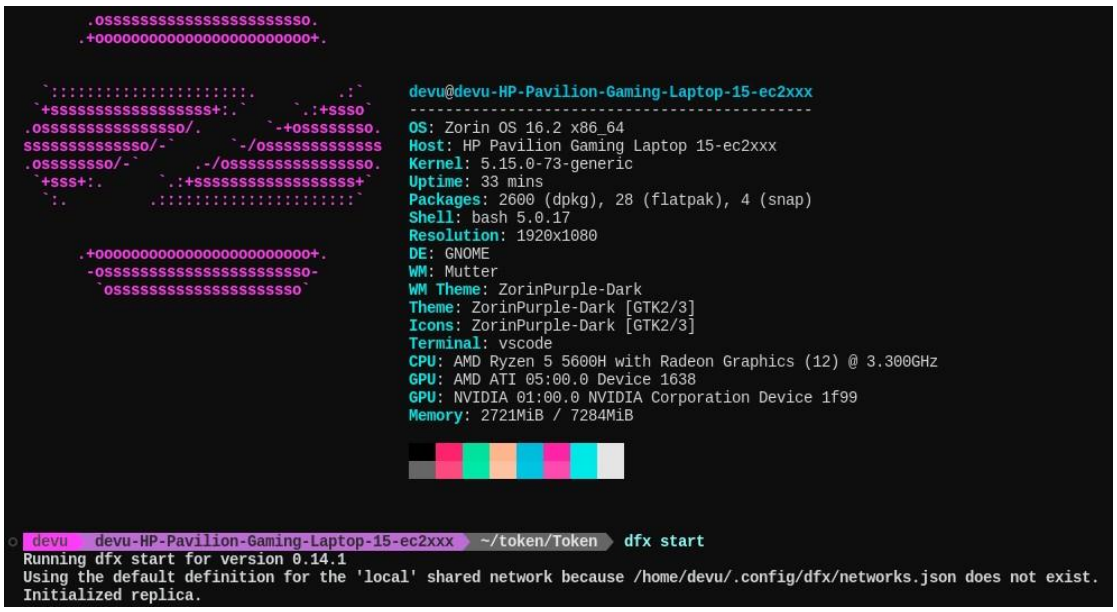



*Fig 4.4. all files*

Now write following command in terminal “dfx start”




this command will make your project ready for deployment.

Now deploy your Dapp using command “dfx deploy”. This command will deploy all your canister on IC

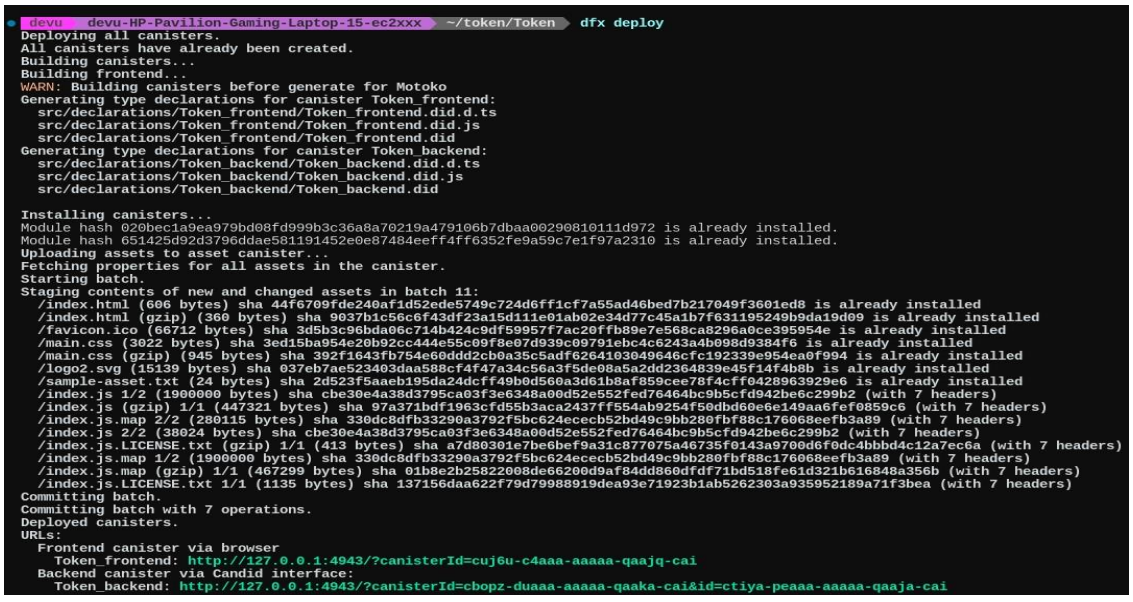



*Fig 4.5 dfx server start*

After successful deployment you can see your frontend by click on the link of token frontend remember this frontend is of all default programmes which are automatically generated at the time of creation of new project we will change all this backend and frontend code in order to make our token working perfectly. This greeting frontend is just bydefault program created by dfinity.

### 4.1 KALKI Backend Code

In backend of KALKI there is an motoko code which is responsible for all our token work now we see and get to know how each line of that motoko code working:

```
import Principal "mo:base/Principal";
import Nat "mo:base/Nat";
import Text "mo:base/Text";
import HashMap "mo:base/HashMap";
import Debug "mo:base/Debug";
import Iter "mo:base/Iter";

actor Token {

  let owner : Principal = Principal.fromText("3efgq-enuf6-kf2qg-sinxx-mdev7-5k47m-mmn7l-hcnex-axkjs-c4t4l-wqe");
  let totalsupply : Nat = 1000000000;
  let symbol : Text = "KALKI";


  private stable var balanceentries: [(Principal, Nat)] = [];

  private var balances = HashMap.HashMap<Principal, Nat>(1,Principal.equal,Principal.hash);

   if(balances.size() < 1 ){
       balances.put(owner,totalsupply);
       };
```

*Fig 4.1.1. all imports*

#### 4.1.1 Imports ( ):

This file name is main.mo mo is the extension used for motoko you can find this file on location src/token_backend/main.mo/ this is the main file or backend file. As you see in this file first we import some modules which are needed in this project. First import is of principal principal is type of data like text. The difference is this data type used to store identity which later used by hash function. This principal function can convert any text to principal and any principapl to text by using fromText() and toText() functions. These both functions are come with principal module. After that we import other data

type like nat for natural number and text for text then we import HashMap, HashMap is like a dictionary which need any unique data type like principal the HashMap used for connecting user identity(principal) to all other data type. Don't worry we will talk about HashMap deeply later soon but for now this is enough. After that we use debug module for printing our code output in terminal. Iter module we will cover soon but for now leave.

### 4.1.2Actor Token { }

Now our main code for token start with actor token{} actor consist of all our main functions and this whole actor deployed as a single canister on blockchain IC. In this actor token first we create owner which data type is principal we gave principal as data type because principal id is unique and we give this owner the value of our unique id you can also get your principal id from terminal by just typing "dfx identity get-principal".


*Fig 4.1.2. principal id*

After getting your principal id from terminal store it in code and convert that text to principal by using fromText() method. After that we create total supply constant which contain total no of token and have a data type of nat. And then constant symbol which contain our crypto token name "KALKI" and have a data-type of text. Then we create balance entries which is private means only motoko function can access them and its stable because we dont want it to remove all our entries at the time of re-deployment and it have data-type of array of objects.

### 4.1.3 HashMap ( )

We created balances with the data type of HashMap this hashmap contains two input first is principal data type and second is nat and these HashMap help in storing the data of the owner and his number of tokens by just making the link between owner and his amount or balance.

### 4.1.4 Conditional Statement ( )

This conditional statement verify that if there is no one using your token then all the token will be linked to yourself.

### 4.1.5 BalanceOf ( )

```
public query func balanceof(who: Principal) : async Nat {

  let balance : Nat = switch (balances.get(who)) {
    case null 0;
    case (?result) result;
  };
```

*Fig 4.1.5. balance of function*

This is one of the function of our crypto token actor balanceof() these function tell us the balance of user. This function take principal as arguement and get the value of token he or she have and in this function there is security check also that the user exist or not if not then null will be return.

### 4.1.6 GetSymbol ( )

```
public query func getsymbol() : async Text{
  return symbol;
};
```

*Fig 4.1.6. Get symbol function*

This function give the symbol back to frontend whenever the front end make calls to this function.

### 4.1.7 Payout ( )

```
public shared(msg) func payout() : async Text{
  if(balances.get(msg.caller) == null){

    let amount = 10000;
    let result = await transfer(msg.caller,amount);

    return result;

   }else{
      return "Already claimed "
   }

};
```

*Fig 4.1.7 Payout function*

This payout function give first user a free reedemable 10000 kalki token this function check wether the user exist or not if user dont exist and make a call to this function the this function tranfer 10000 kalki token to his/her KALYUG account. This function call or use another tranfer function to perform this task.

### 4.1.8 Transfer ( )

```
public shared(msg) func transfer(to: Principal,amount: Nat) : async Text{

   let frombalance = await balanceof(msg.caller);

   if(frombalance > amount){

      let newfrombalance : Nat = frombalance - amount;
      balances.put(msg.caller,newfrombalance);

      let tobalance = await balanceof(to);
      let newtobalance = tobalance + amount;
      balances.put(to,newtobalance);

      return "success";
    }else{
      return "insufficient funds";
    }
};
```

*Fig 4.1.8 transfer function*

This tranfer function is used to tranfer the amount of token to another user this takes principal and amount as arguement and do the security check of insufficient funds and return the respected output if output is success this function deduct the amount from the caller who call this function using msg,caller and add the amount to the principal which given as the input arguement and then update our balances hash.

### 4.1.9 PreUpgrade ( )

```
system func preupgrade(){
    balanceentries := Iter.toArray(balances.entries());
};
```

*Fig 4.1.9 preupgrade function*

PreUpgrade is the system function it is responsible for save your data before redeployment of your project iter method is like for loop which select each element. In this function iter taking each entry from balances and storing it in array of balanceentries.

### 4.1.10PostUpgrade ( )

```
system func postupgrade(){
    balances := HashMap.fromIter<Principal,Nat>(balanceentries.vals(),1,Principal.equal,Principal.hash);
     if(balances.size() < 1 ){
     balances.put(owner,totalsupply);
     }
};

};
```

*Fig 4.1.10. Postupgrade function*

PostUpgrade is like recover all your data after re deployment from balance entries to balance HashMap.

## 4.2 Token Front-End Code

In token frontend code we used react as our front-end to setup react with motoko you have to follow some steps you can view these steps on the dfinity docs follow the following link to setup react as your frontend.

**“https://internetcomputer.org/docs/current/developer-docs/frontend/custom-frontend”**

In our token frontend we have react components and one html file and one css file and some assets we will review each file one by one so lets start with react root file.

### 4.2.1 Index File

```
import ReactDOM from 'react-dom'
import React from 'react'
import App from "./components/App";
import { AuthClient } from "@dfinity/auth-client";

const init = async () => {


  const authClient = await AuthClient.create();

  if(await authClient.isAuthenticated()) {
    handleAuthenticated(authClient);
  }else{

  await authClient.login({
    identityProvider: "https://identity.ic0.app/#authorize",
    onSuccess: () => {
            handleAuthenticated(authClient);
    }
  });
  }
}

async function handleAuthenticated(authClient) {
  const identity = await authClient.getIdentity();
  const userprincipal = identity._principal.toString();
  console.log(userprincipal);
  ReactDOM.render(<App  loggedInPrincipal={userprincipal}/>, document.getElementById("root"));
}

init();
```

Fig 4.2.1. Index file

In index.jsx file first we import main react modules that are necessary for react and then import react component app and import authclient function from dfinity. Auth client gives user an unique internet identity. And make your account secure by your identity like fingerprint for login. Then in init function we set up our authentication code.

### 4.2.2 App File

```
import React from "react";
import Header from "./Header";
import Faucet from "./Faucet";
import Balance from "./Balance";
import Transfer from "./Transfer";

function App(props) {

  return (
    <div id="screen">
      <Header />
      <Faucet  userprincipal={props.loggedInPrincipal}/>
      <Balance />
      <Transfer />
    </div>
  );
}

export default App;
```

*Fig 4.2.2 App file*

This is our app.jsx file in this file all other components are imported and this is the the file which loads up from index.jsx and this loads all other remaining components of project. We use react props to pass authenticaed id between different components like faucet.

### 4.2.3 Header File

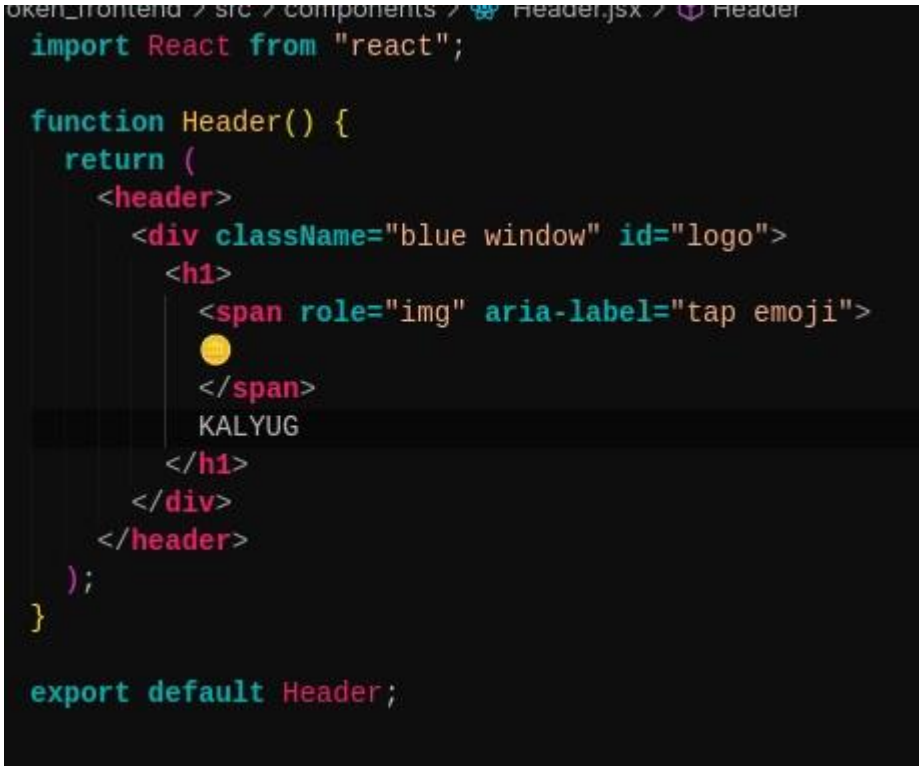
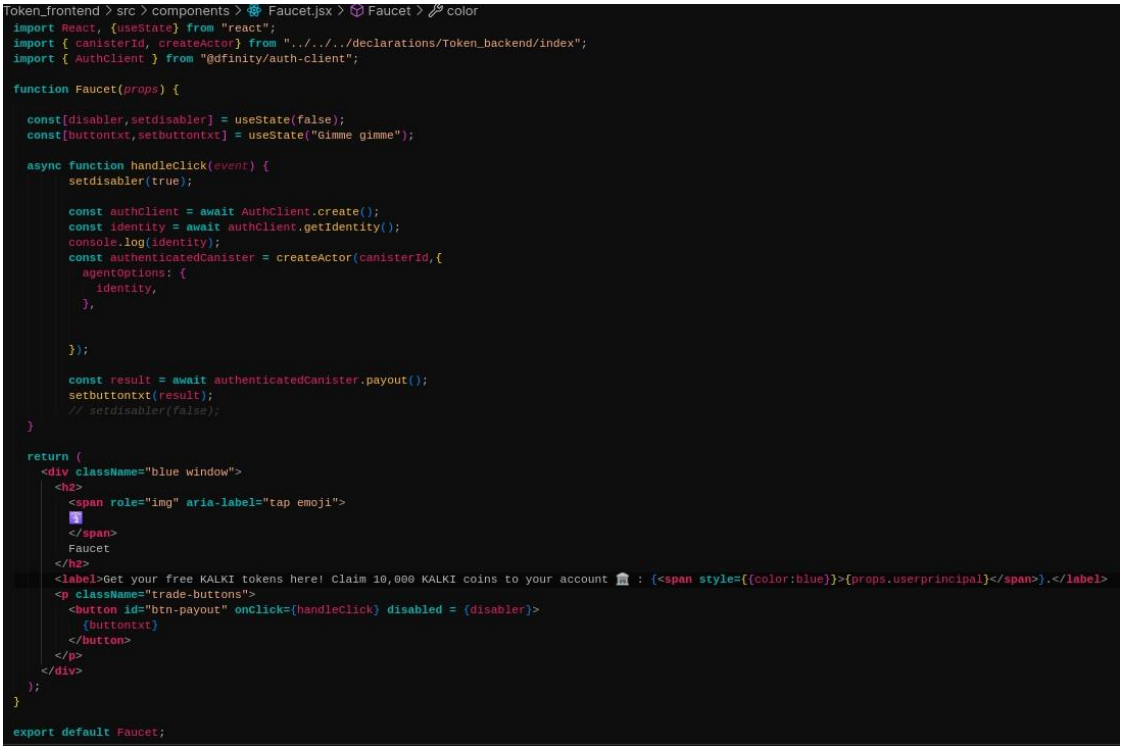



*Fig 4.2.3 Header file*

This is our header.jsx which include heading and normal html code like name of the wallet KALYUG. This is the first section of our website.

### 4.2.4 Faucet File



Token_frontend > src > components > Faucet.jsx > Faucet > color

```jsx
import React, {useState} from "react";
import { canisterId, createActor} from "../../../declarations/Token_backend/index";
import { AuthClient } from "@dfinity/auth-client";

function Faucet(props) {

  const[disabler,setdisabler] = useState(false);
  const[buttontxt,setbuttontxt] = useState("Gimme gimme");

  async function handleClick(event) {
        setdisabler(true);

        const authClient = await AuthClient.create();
        const identity = await authClient.getIdentity();
        console.log(identity);
        const authenticatedCanister = createActor(canisterId,{
          agentOptions: {
            identity,
          },


        });

        const result = await authenticatedCanister.payout();
        setbuttontxt(result);
        // setdisabler(false);
  }

  return (
    <div className="blue window">
      <h2>
        <span role="img" aria-label="tap emoji">
        🚰
        </span>
        Faucet
      </h2>
      <label>Get your free KALKI tokens here! Claim 10,000 KALKI coins to your account 🏦 : {<span style={{color:blue}}>{props.userprincipal}</span>}.</label>
      <p className="trade-buttons">
        <button id="btn-payout" onClick={handleClick} disabled = {disabler}>
          {buttontxt}
        </button>
      </p>
    </div>
  );
}

export default Faucet;
```

*Fig 4.2.4 faucet file*

This is faucet.jsx file which responsible for calling the backend payout() function this file call payout() and from that you get you free 10000 kalki token. For calling to backend function you just need to import backend to this file like we did in second line.

### 4.2.5 Balance File


*Fig 4.2.5. Balance file*

Balance.jsx file this file call the balanceof() backend function and tell you the balance of the account you gave this is the second section of our website.

### 4.2.6 Transfer File

```jsx
import React, { useState } from "react";
import { canisterId,createActor } from "../../../declarations/Token_backend/index";
import {Principal} from "@dfinity/principal";
import { AuthClient } from "@dfinity/auth-client";

function Transfer() {

  const [recipentid,setid] = useState("");
  const [amount,setamount] = useState("");
  const [isdisabled,setdisabler] = useState(false);
  const [ishidden,sethidden] = useState(true);
  const [feedback,setfeedback] = useState("");
  async function handleClick() {
    sethidden(true);
    setdisabler(true);
    const recipient = Principal.fromText(recipentid);
    const amounttotransfer = Number(amount);


    const authClient = await AuthClient.create();
    const identity = await authClient.getIdentity();
    const authenticatedCanister = createActor(canisterId,{
      agentOptions : {
        identity,
      },


    });


    const result = await authenticatedCanister.transfer(recipient,amounttotransfer);

    setfeedback(result);
    sethidden(false);
    setdisabler(false);

  }
```

*Fig 4.2.6 Transfer file*

This is tranfer.jsx file which responsible for calling tranfer method from backend. And this is the last section of our website. Note that how the backend method is called is important we have to make calls by authenticated client in case of faucet and tranfer method because they are shared method they need msg.caller identity.

*Thats all now we successfully created our crypto token.*

## Chapter 5 - Result and Analysis

we successfully create our KALKI crypto token.

**Output:**

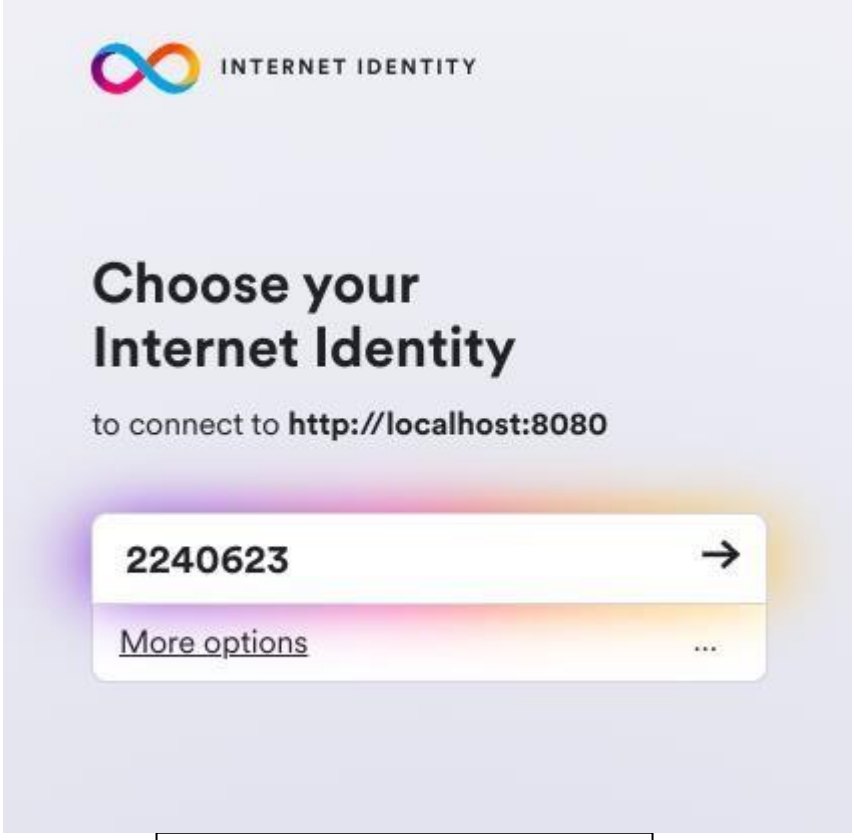



*Fig 5.1 internet identity output*

first window is of internet identity which give you an identity and authorize you like a login page by your fringer print if you are right person then it will access to your wallet.


*Fig 5.2 NFT web outputt output*

Second window is our main window of crypto token wallet this wallet have three section as we discussed first faucet the followed by balance check then followed by transfer amount. We made this wallet little classic by using CSS and BOOTSTRAP.

**I deployed this token on live blockchain you can also access this crypto token.**

**Link of crypto token: https://galnq-mqaaa-aaaal-acqwq-cai.icp0.io/**

## Chapter 6 - Conclusion

This project is really fun and knowledgeable, this blockchain and cryptocurrency are very important topic because third generation of web is coming. So we have to learn about all these subject and the best part is that these topics are not boring. They are very interesting.

## Chapter 7 - Refrences

In making of this project I used only one resource that is the course on Udemy of web-development very famous or highest rated best seller course.

[1] Course name- the complete 2023 web development

bootcampProfessor- Dr Angela yu

Courselink-https://www.udemy.com/course/the-complete-web-development-bootcamp/

[2] YouTube video about history of cryptocurrency