

Blockchain final project

InvoiceFund -Decentralized working capital financing

Team:

Name: Daniyal Adilbekov

Group: SE-2435

Name: Bibifatima Bisesheva

Group: SE-2437

Name: Ataniyaz Mutigolla

Group: SE-2437

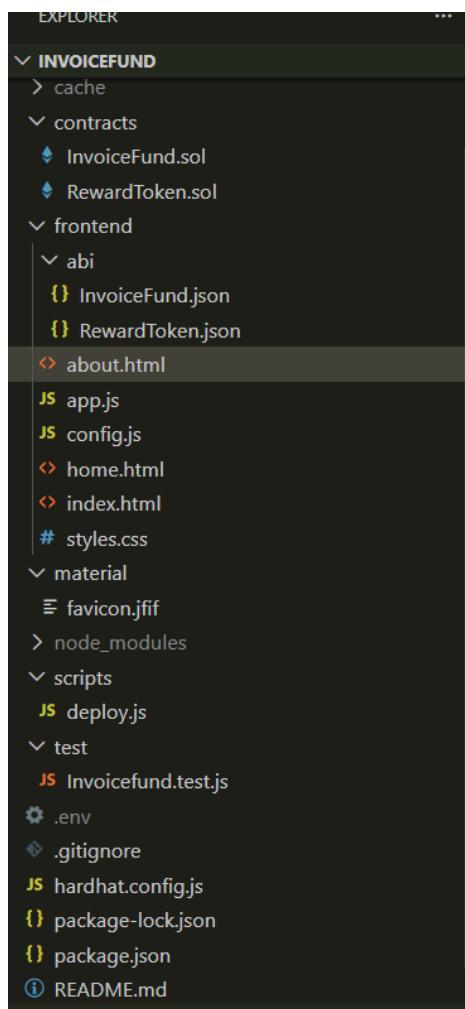
Project Overview

So, basically, InvoiceFund is a decentralized crowdfunding platform designed to provide early liquidity to businesses awaiting invoice payments. Instead of relying on banks or intermediaries, funding is raised through smart contracts. Participants receive ERC-20 reward tokens, which demonstrate proportional participation and accountability on the blockchain. The system operates entirely on the Ethereum Sepolia testnet using MetaMask.

This project was developed as a final project and operates exclusively on an Ethereum test network using free test tokens.

This project was developed as a blockchain final project

Project architecture



Home.html where is main information about project

Home.html where is main information about project

127.0.0.1:5173/frontend/home.html

Home DApp About

What is InvoiceFund?

So, basically, InvoiceFund is a decentralized crowdfunding platform designed to provide early liquidity to businesses awaiting invoice payments. Instead of relying on banks or intermediaries, funding is raised through smart contracts. Participants receive ERC-20 reward tokens, which demonstrate proportional participation and accountability on the blockchain. The system operates entirely on the Ethereum Sepolia testnet using MetaMask.

Why blockchain?

- Tracking contributions on-chain
- Reward token system
- Using Sepolia testnet

What is the meaning of reward tokens?

They are automatically minted by the smart contract during contributions and serve as an on-chain proof of involvement, ensuring transparency, traceability, and accountability without intermediaries. While the tokens have no real monetary value, they demonstrate core tokenization concepts.

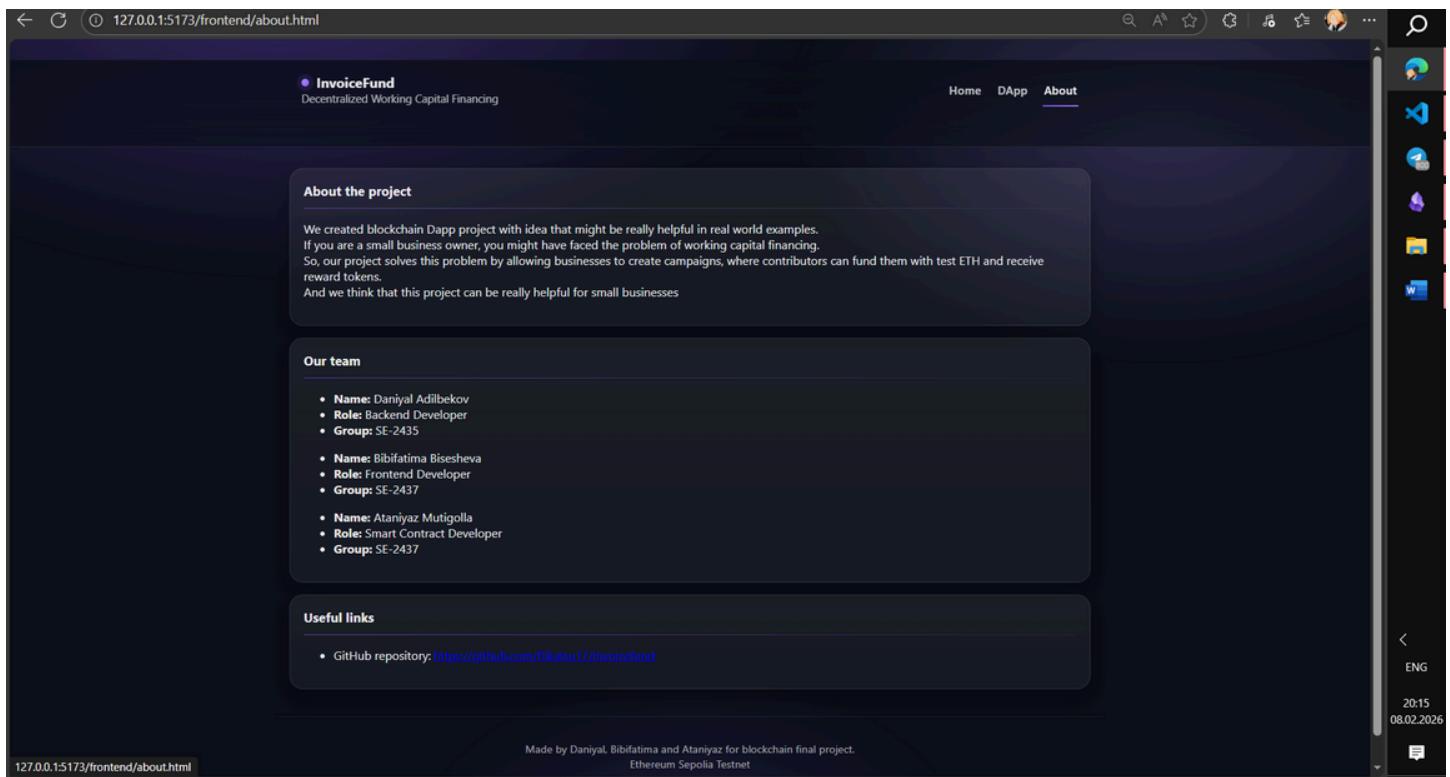
Try the DApp

Go to the DApp page to connect MetaMask, create campaigns, and contribute

Open DApp

Made by Daniyal, Bibafatima and Ataniyaz for blockchain final project.
Ethereum Sepolia Testnet

About.html where you can see information about creators and links



The screenshot shows the 'About' section of the InvoiceFund DApp. At the top, there's a header with the project name 'InvoiceFund' and its description 'Decentralized Working Capital Financing'. Below the header, there are three main sections: 'About the project', 'Our team', and 'Useful links'. The 'About the project' section contains a paragraph about solving the problem of working capital financing for small business owners. The 'Our team' section lists three members with their names, roles, and group numbers. The 'Useful links' section includes a link to the GitHub repository. The bottom of the page shows copyright information and the URL '127.0.0.1:5173/frontend/about.html'. On the right side of the browser window, there are various icons for file operations like search, copy, paste, and refresh.

About the project

We created blockchain Dapp project with idea that might be really helpful in real world examples.
If you are a small business owner, you might have faced the problem of working capital financing.
So, our project solves this problem by allowing businesses to create campaigns, where contributors can fund them with test ETH and receive reward tokens.
And we think that this project can be really helpful for small businesses

Our team

- Name: Daniyal Adilbekov
• Role: Backend Developer
• Group: SE-2435
- Name: Bibifatima Bisesheva
• Role: Frontend Developer
• Group: SE-2437
- Name: Ataniyaz Mutigolla
• Role: Smart Contract Developer
• Group: SE-2437

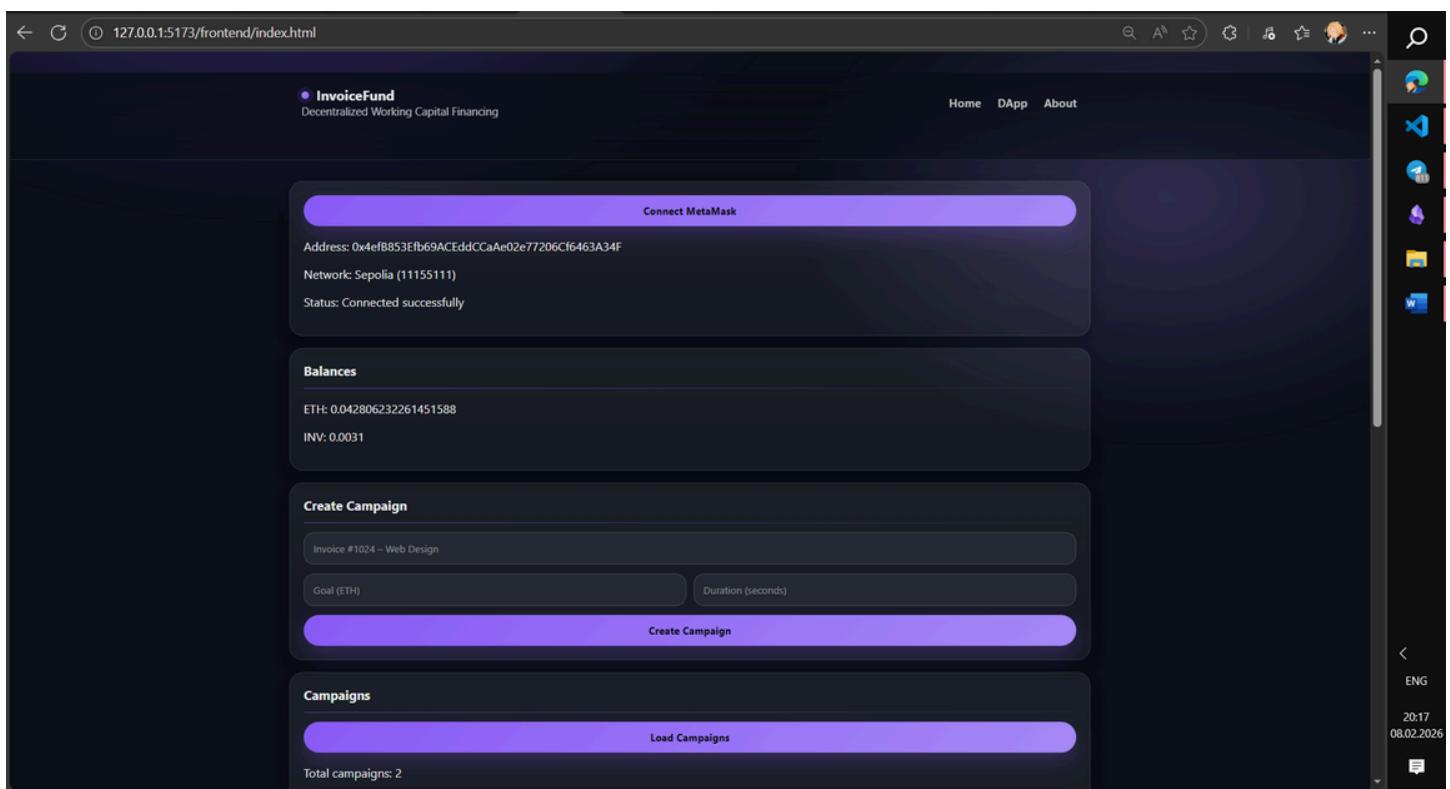
Useful links

- GitHub repository: <https://github.com/Rikatsu1/invoicefund>

Made by Daniyal, Bibifatima and Ataniyaz for blockchain final project.
Ethereum Sepolia Testnet

127.0.0.1:5173/frontend/about.html

And there is main file index.html where are functions like create campaign, list of campaigns, contribute function, and finalizing any campaign.



The screenshot shows the main interface of the InvoiceFund DApp. At the top, there's a header with the project name 'InvoiceFund' and its description 'Decentralized Working Capital Financing'. Below the header, there are four main sections: 'Connect MetaMask', 'Balances', 'Create Campaign', and 'Campaigns'. The 'Connect MetaMask' section shows the user's Ethereum address (0x4efB853Ef...), network (Sepolia), and connection status (Connected successfully). The 'Balances' section displays ETH and INV token balances. The 'Create Campaign' section allows users to create a new campaign by entering an invoice ID, setting a goal in ETH, and specifying a duration in seconds, then clicking the 'Create Campaign' button. The 'Campaigns' section shows a list of existing campaigns, with a 'Load Campaigns' button and a message indicating two total campaigns.

Connect MetaMask

Address: 0x4efB853Ef...
Network: Sepolia (1155111)
Status: Connected successfully

Balances

ETH: 0.042806232261451588
INV: 0.0031

Create Campaign

Invoice #1024 – Web Design
Goal (ETH) _____ Duration (seconds) _____
Create Campaign

Campaigns

Load Campaigns
Total campaigns: 2

Campaigns

[Load Campaigns](#)

Total campaigns: 2

ID 0

Title: invoice 1025
Owner: 0x4efB853Ef869ACEddCCaAe02e77206Cf6463A34F
Goal: 0.0003 ETH
Raised: 0.00003 ETH
Deadline: 08.02.2026, 14:58:36
Finalized: true

ID 1

Title: invoice 666
Owner: 0x4efB853Ef869ACEddCCaAe02e77206Cf6463A34F
Goal: 0.0003 ETH
Raised: 0.000001 ETH
Deadline: 08.02.2026, 16:15:24
Finalized: false

Contribute

Contribute

Campaign ID

Amount (ETH)

[Contribute](#)

Finalize

Campaign ID

[Finalize Campaign](#)

How it works

The screenshot shows two overlapping interfaces. On the left is the 'InvoiceFund' web application at 127.0.0.1:5173. It displays a 'Create Campaign' form with the title 'Creating website for client', a goal of '0.00004' ETH, and a deadline of '08.02.2026, 14:58:36'. A purple progress bar indicates the campaign is being created. On the right is the 'MetaMask' extension for a browser, showing a transaction request for a Sepolia network interaction. The transaction details include a gas fee of '0.0003 s SepoliaETH' and a confirmation time of '~12 сек.'.

There is created campaign with full information

ID 2

Title: Creating website for client

Owner: 0x4efB853Efb69ACEddCCaAe02e77206Cf6463A34F

Goal: 0.00004 ETH

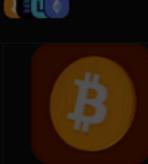
Raised: 0.0 ETH

Deadline: 08.02.2026, 23:06:52

Finalized: false

There is creating campaing in metamask

Daniyal Adilbekov ▾



Say hello to Bitcoin!
Trade, manage, earn.

Токены DeFi NFT Действия Sepolia ▾ Feb 8, 2026

Create Campaign Подтверждено

Finalize Подтверждено

Contribute Подтверждено

Create Campaign Подтверждено

Contribute Подтверждено

Create Campaign Подтверждено

Create Campaign Подтверждено

Create Campaign

Статус Просмотр в проводнике блоков
Подтверждено Скопировать ID транзакции

Из Account 1 → **Место назначения** 0xbADA6...C2c7c

Транзакция

Одноразовый код 13

Сумма -0 SepoliaETH

Лимит Газа (Единицы) 127337

Использовано Газа (Единицы) 126253

Базовая комиссия (Гвей) 1.001767412

Плата за приоритет (Гвей) 1.5

Итого платы за газ 0.000316 SepoliaETH

Макс. комиссия на газ 0.000000003 SepoliaETH

Итого 0.00031586 SepoliaETH

+ Журнал активности

Now if people want to contribute to this campaign

Goal: 0.0003 ETH
Raised: 0.000001 ETH
Deadline: 08.02.2026, 16:15:24
Finalized: false

ID 2
Title: Creating website for client
Owner: 0x4efB853EfB69ACEddCCaAe02e77206Cf6463A34F
Goal: 0.0004 ETH
Raised: 0.0 ETH
Deadline: 08.02.2026, 23:06:52
Finalized: false

Contribute

2 0.00001

Contribute

Finalize

Campaign ID

Finalize Campaign

Запрос транзакции

Прогнозируемые изменения ?
Вы отправляете - 0,00001 \$ SepoliaETH
Вы получаете + 0,001 \$ 0xaC3C1...f569e

Сеть Sepolia
Запрос от HTTP 127.0.0.1:5173
Взаимодействие с OxbADA6..C2c7c

Комиссия сети ? 0.0002 \$ SepoliaETH

Скорость Пример: ~12 сек

Отмена Подтвердить

Made by Daniyal, Bibifatima and Ataniyaz for blockchain final project.
Ethereum Sepolia Testnet

Changes total of INV

Balances

ETH: 0.042223874374651905

INV: 0.0041

And in campaign itself shows how much raised

ID 2

Title: Creating website for client

Owner: 0x4efB853Efb69ACEddCCaAe02e77206Cf6463A34F

Goal: 0.00004 ETH

Raised: 0.00001 ETH

Deadline: 08.02.2026, 23:06:52

Finalized: false

About finalizing. I can finalize the campaign only after deadline

The screenshot shows a web application interface for a campaign titled "ID 2". The campaign details are as follows:

- Title: Creating website for client
- Owner: 0x4efB853Efb69ACEddCCaAe02e77206Cf6463A34F
- Goal: 0.00004 ETH
- Raised: 0.00001 ETH
- Deadline: 08.02.2026, 23:06:52
- Finalized: false

Below the campaign details, there are two buttons: "Contribute" and "Finalize".

A MetaMask extension window is overlaid on the browser window, titled "Запрос транзакции" (Transaction Request). It displays the following information:

- Прогнозируемые изменения (Predicted changes): Нет изменений (No changes)
- Сеть (Network): Sepolia
- Запрос от (From): 127.0.0.1:5173
- Взаимодействие с (Interaction with): 0xbADA6...C2c7c
- Комиссия сети (Network fee): 0.0001 s SepoliaETH
- Скорость (Speed): Рынок ~12 сек. (Market ~12 sec.)

At the bottom of the MetaMask window are "Отмена" (Cancel) and "Подтвердить" (Confirm) buttons.

At the very bottom of the browser window, it says: "Made by Danyal, Bibifatima and Ataniyaz for blockchain final project. Ethereum Sepolia Testnet".

There it shows in metamask and campaign with id finalized changes to true

The screenshot shows a web application for 'InvoiceFund' running at 127.0.0.1:15173/frontend/index.html. The application displays three campaign entries:

- ID 0**: Title: invoice 1025, Owner: 0x4efB853EfB69ACeddCcAe02e77206Cf6463A34F, Goal: 0.0003 ETH, Raised: 0.00003 ETH, Deadline: 08.02.2026, 14:58:36, Finalized: true.
- ID 1**: Title: invoice 666, Owner: 0x4efB853EfB69ACeddCcAe02e77206Cf6463A34F, Goal: 0.0003 ETH, Raised: 0.00001 ETH, Deadline: 08.02.2026, 16:15:24, Finalized: true.
- ID 2**: Title: Creating website for client, Owner: 0x4efB853EfB69ACeddCcAe02e77206Cf6463A34F, Goal: 0.0004 ETH, Raised: 0.00001 ETH, Deadline: 08.02.2026, 23:06:52, Finalized: false.

A modal window titled 'Contribute' is open, showing a value of 2 in a dropdown field and a button labeled 'Contribute'.

In the top right corner, a MetaMask extension window is open, titled 'Say hello to Bitcoin'. It shows a balance of 0 SepoliaETH and a transaction history:

- Finalize (Подтверждено) -0 SepoliaETH (-0 SepoliaETH)
- Contribute (Подтверждено) -0.00001 Sepol... (-0.00001 SepoliaETH)
- Create Campaign (Подтверждено) -0 SepoliaETH (-0 SepoliaETH)
- Finalize (Подтверждено) -0 SepoliaETH (-0 SepoliaETH)

The MetaMask interface includes tabs for Tokens, DeFi, NFT, and Activity, with Activity selected. The date Feb 8, 2026, is also visible.

if i try to finalize before deadline, it will show as error

also about changing network, if user will change the network to something different, it will be shown in frontend. There is active connection to sepoliaETH

The screenshot shows the MetaMask wallet interface. At the top, it displays "Daniyal Adilbekov" and the network "Sepolia (11155111)". Below this, there's a purple bar with the text "Connect MetaMask". Underneath, it shows the address "0x4efb853Efb69ACEddCCaAe02e77206Cf6463A34F", the network "Sepolia (11155111)", and the status "Connected successfully". A "Balances" section shows ETH: 0.042101381688129006 and INV: 0.0041. A "Create Campaign" section is present with a "Create Campaign" button. On the right, there's a sidebar with tabs for "Токены", "DeFi", "NFT", and "Деятельность" (Activity). The "Activity" tab is selected, showing a transaction from Feb 8, 2026, labeled "Finalize" with status "Подтверждено" (Confirmed). The transaction details are "-0 SepoliaETH" and "-0 SepoliaETH". A "Say hello to Bitcoin" card is also visible.

Now if user will change to different one, it will show the message "Please switch metamask to sepolia network"

This screenshot shows the same MetaMask wallet interface as above, but with a network switch. The top bar now says "Daniyal Adilbekov" and "Sepolia (11155111)". The "Address:" field is empty, and the "Network:" field shows "-". The status message "Please switch MetaMask to Sepolia network" is displayed. The "Balances" section shows ETH: - and INV: -. The "Create Campaign" section is identical to the first screenshot. The "Activity" tab in the sidebar shows a transaction from Feb 8, 2026, labeled "Finalize" with status "Подтверждено" (Confirmed). The transaction details are "-0 SepoliaETH" and "-0 MON". A "Say hello to Bitcoin" card is also visible. The bottom right corner shows the date "09.02.2026" and time "01:57" with language "ENG".

and it also shows in console, that chainId of what user chosed is different. ChainId of sepolia is 11155111

```
▶ SES Removing unpermitted intrinsics          lockdown-install.js:1
ABIs loaded: 13 25                           app.js:84
Detected chainId: 10143                      app.js:92
ABIs loaded: 13 25                           app.js:84
Detected chainId: 10143                      app.js:92
ABIs loaded: 13 25                           app.js:84
Detected chainId: 10143                      app.js:92
>
```

There is code condition

```
await provider.send('eth_requestAccounts', []);

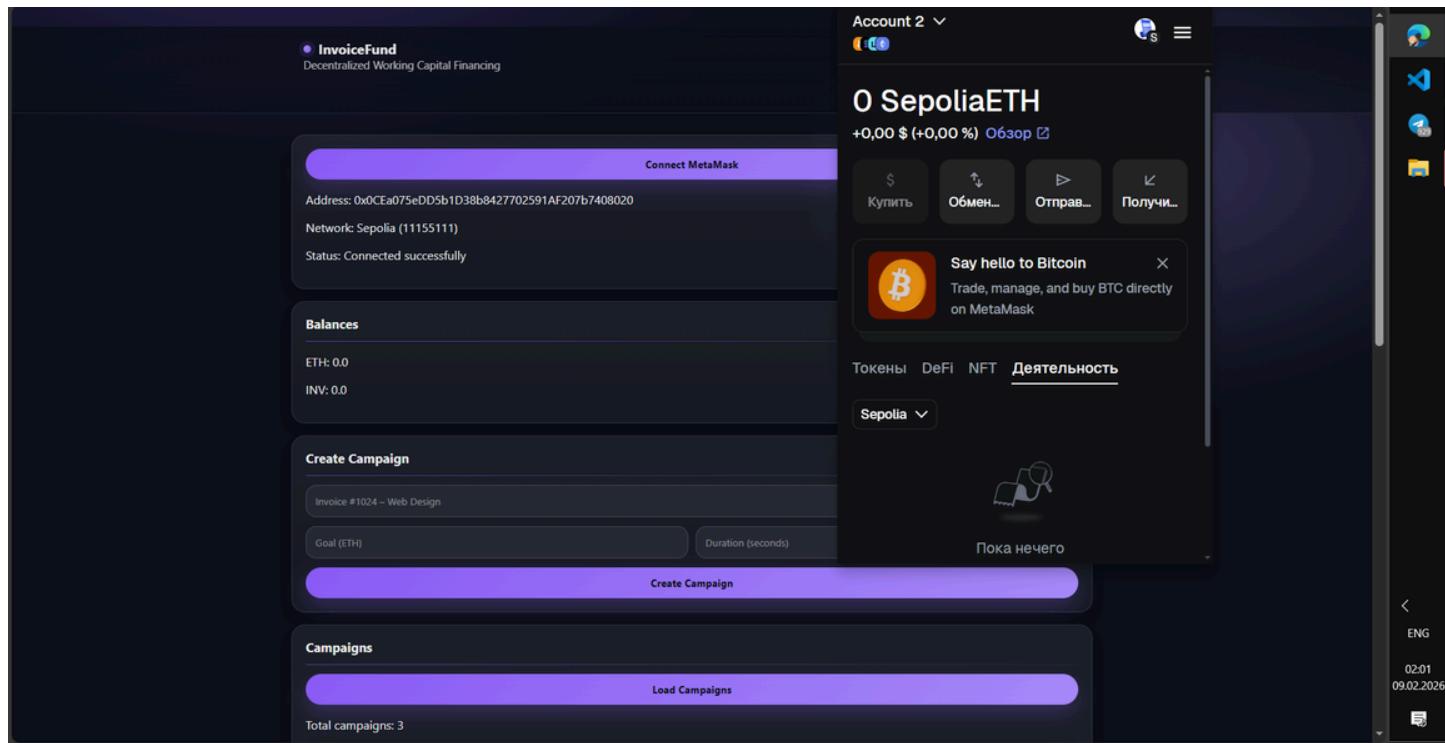
const network = await provider.getNetwork();
const chainId = Number(network.chainId);
console.log("Detected chainId:", chainId);

if (chainId !== SEPOLIA_CHAIN_ID) {
  setStatus("Please switch MetaMask to Sepolia network");
  return;
}
```

Also if accounts changes then the page will be reloaded

```
'           window.ethereum.on("accountsChanged", () => window.location.reload());
           window.ethereum.on("chainChanged", () => window.location.reload());
```

and successfully connected to the website



Purpose of the Project

The main objective of this project is to demonstrate practical knowledge of:

Using Solidity for smart contracts

ERC-20 tokens integration

Client-side blockchain interaction using JavaScript

MetaMask wallet integration

Interaction with sepolia test networks

Basic dapp architecture

The application addresses the problem of delayed invoice payments by simulating decentralized invoice crowdfunding.

Technology Stack Solidity

Solidity

Hardhat Ethereum Sepolia Test Network

ERC-20 Token Standard

JavaScript

MetaMask

HTML & CSS

Smart Contracts

The project consists of two smart contracts:

1. InvoiceFund

Responsible for:

- Creating crowdfunding campaigns
- Accepting ETH contributions
- Tracking individual and total contributions
- Finalizing campaigns after the deadline
- Minting reward tokens for contributors

2. RewardToken (ERC-20)

- Custom ERC-20 token(inv)
- Minted automatically during campaign participation
- Has no real monetary value
- Used strictly for educational demonstration

Reward formula

$$\text{Reward} = \text{contributionWei} \times \text{REWARD_RATE} / 1e18$$

Frontend Features

The client-side application allows users to:

- Connect a MetaMask wallet
- Validate the active blockchain network (Sepolia)
- View wallet address and balances (ETH & INV)
- Create crowdfunding campaigns
- Browse existing campaigns
- Contribute test ETH to campaigns
- Automatically receive ERC-20 reward tokens
- Finalizing campaigns
- All blockchain transactions are executed securely through MetaMask.

Frontend to blockchain interaction

The frontend interacts with the Ethereum Sepolia test network using app.js and the MetaMask wallet. MetaMask provides the connection between the web application and the blockchain.

After the user connects MetaMask, the application initializes an app BrowserProvider and obtains a signer representing the active wallet account. Smart contracts are accessed using their deployed addresses and ABI files.

Read-only operations (loading campaigns, checking balances) are performed via the provider, while state-changing operations (creating campaigns, contributing ETH, finalizing campaigns) are executed as blockchain transactions and require user confirmation through MetaMask.

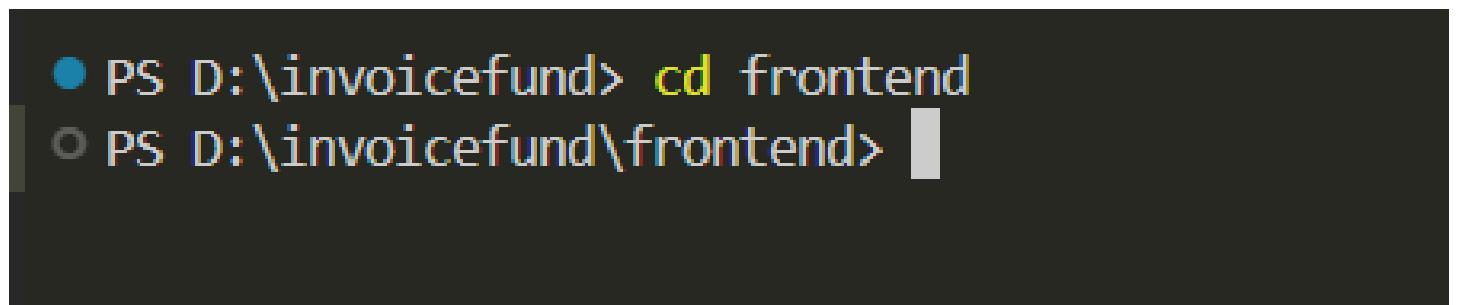
The frontend validates that the user is connected to the Sepolia test network and listens for MetaMask account or network changes to keep the application state consistent.

How to Run the Project

Steps

1) Navigate to the frontend directory:

```
cd frontend
```

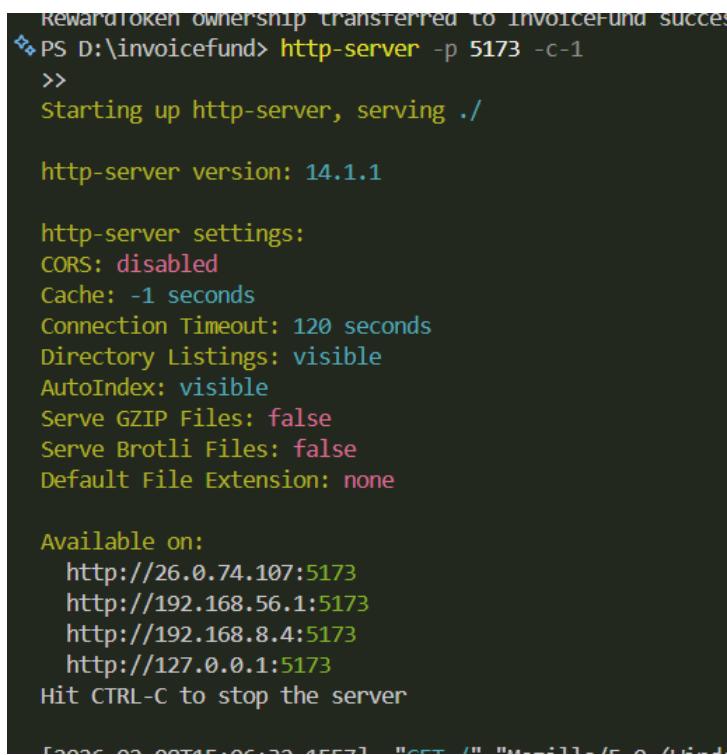


PS D:\invoicefund> cd frontend

PS D:\invoicefund\frontend>

2) Start a local static server:

```
http-server -p 5173 -c-1
```



```
RewardToken ownership transferred to InvoiceFund successfully.
* http://26.0.74.107:5173
* http://192.168.56.1:5173
* http://192.168.8.4:5173
* http://127.0.0.1:5173
Hit CTRL-C to stop the server
```

3) Open the application in a browser:

```
http://127.0.0.1:5173
```

4) Switch MetaMask to the Sepolia Test Network

Account 3

0 SepoliaETH
+0,00 \$ (+0,00 %) Обзор

Купить
Обмен...
Отправ...
Получи...

Say hello to Bitcoin
Trade, manage, and buy BTC directly on MetaMask

Токены DeFi NFT Деятельность

Sepolia

Account 3 не подключен к 127.0.0.1:5173
Подключить счет

Balances

ETH: 0.042806232261451588
INV: 0.0031

Create Campaign

Invoice #1024 – Web Design
Goal (ETH)
Duration (seconds)

Create Campaign

Campaigns

Load Campaigns

Total campaigns: 2

5) Connect MetaMask and interact with the DApp

Daniyal Adilbekov

0,0428 SepoliaETH
+0,00 \$ (+0,00 %) Обзор

Купить
Обмен...
Отправ...
Получи...

Say hello to Bitcoin
Trade, manage, and buy BTC directly on MetaMask

Токены DeFi NFT Деятельность

Sepolia

Feb 8, 2026

Finalize
Подтверждено
-0 SepoliaETH
-0 SepoliaETH

Contribute
-0.000001 Sep...

Balances

ETH: 0.042806232261451588
INV: 0.0031

Create Campaign

Invoice #1024 – Web Design
Goal (ETH)
Duration (seconds)

Create Campaign

Campaigns

Load Campaigns

Total campaigns: 2

How to get Sepolia test ethereum

Firstly u need to get your account address, then go to websites where you can get test sepoliaETH

For example: <https://cloud.google.com/application/web3/faucet/ethereum/sepolia> deploy to sepolia

Get your first drip of OG's \$OG on the Galileo testnet today! [Open faucet →](#)

Ethereum Sepolia Faucet BETA

Get free Sepolia ETH sent directly to your wallet. Brought to you by [Google Cloud for Web3](#).

Select network*

Ethereum Sepolia

*required

Wallet address or ENS name*

0x4efB853Ef...
Enter the account address or ENS name where you want to receive tokens

Get 0.05 Sepolia ETH

FAQ

How does my data get used?

What is a faucet?

How does the faucet work?

Feedback

Privacy Policy

ENG
20:13
08.02.2026

Testing

Automated unit tests are implemented using Hardhat to verify contract deployment, creating campaigns, contribution logic, reward token minting and etc. To run it write in terminal:

```
npx hardhat test
```

```
PS D:\invoicefund> npx hardhat test
[dotenv@17.2.3] injecting env (2) from .env -- tip:  audit secrets and track compliance: https://dotenvx.com/ops
```

```
InvoiceFund DApp
✓ Should deploy and transfer RewardToken ownership to InvoiceFund (2575ms)
✓ Should create a campaign with correct parameters (62ms)
✓ Should accept contributions and track raised amount (46ms)
✓ Should mint reward tokens proportional to contribution (49ms)
✓ Should not allow contribute to non-existing campaign (213ms)
✓ Should not allow finalize before deadline (54ms)
✓ Should allow finalize after deadline (52ms)
✓ Should not allow finalize twice
✓ Should not allow contribute after finalize (45ms)
```

```
9 passing (3s)
```

```
PS D:\invoicefund>
```

Conclusion

InvoiceFund demonstrates a complete decentralized crowdfunding workflow using Ethereum smart contracts, MetaMask integration, and a client-side DApp interface. The project highlights key blockchain concepts such as decentralization, transparency, tokenization, and secure user interaction on a test network.

