AELF BOILERPLATE INTRODUCTION

aelf Boilerplate is a development template based on aelf blockchain system with built-in development scaffoldings and simple demos. You can use aelf Boilerplate to quickly build your own blockchain system and develop smart contracts and DAPPs based on aelf blockchain system.

This tutorial will guide you through a demo, and you will able to build an aelf blockchain environment in just 5 minutes.

Demo Introduction

This tutorial includes 4 demos:

1. **HelloWorld Smart Contract Demo**: Guides you to develop smart contracts based on aelf blockchain system using aelf Boilerplate

2. **JS SDK Demo:** Guide you to build a DAPP that interacts with the above *HelloWorld Smart Contract* by using *JS SDK*

3. **aelf Browser Plugin Demo**: Guide you to develop DAPP based on aelf blockchain system by using aelf browser plugin

4. **DAPP Demo**: Guide you to run DAPP based on aelf blockchain system - Bingo Game

Note:

This tutorial only shows you how to run demos on Windows Systems. Procedure for other operating systems are slightly different. Check aelf official Gitbook for details: https://docs.aelf.io/main

Installation

Prior to setting up the blockchain there are a number of programs you may need to install.

Minimum Requirement

To successfully complete this tutorial you first need to install the following dependencies, please refer to your platforms installation guide:

- Git for version control, follow these instructions.
- Dotnet core sdk for building and testing, you can download it here.
- To run the JSSDK demo you'll need to install Nodejs

Mac users also need to install Homebrew.

IDE

Strictly speaking you don't need an IDE for this tutorial but it is highly recommended. If you don't already have one we recommend you try Visual Studio Code (vscode) with the C# extension:

- installation instructions here.
- working with C# here.

Related links can be found in the official Gitbook: https://docs.aelf.io/main

Blockchain Setup

After completing the above requirements you can proceed to setting up the aelfboilerplate and building the blockchain.

(1) Download and Open *AElf.Boilerplate*

Open command prompt, navigate to your favorite working directory using the cd command and download AElf.Boilerplate by entering the following command:

git clone https://github.com/aelfProject/aelf-boilerplate

When the download is complete, open *AElf.Boilerplate* folder in *Visual Studio Code*.

If asked to add some "required assets" say **yes**. There may also be some dependencies to restore: for all of them choose **Restore**.

(2) Download *protobuf* script

Windows is slightly complex. You need to open a PowerShell console as administrator. Enter your clone directory so your PowerShell is currently in the root of aelf-boilerplate clone.

Run the following:

Set-ExecutionPolicy RemoteSigned , type *Y* or *Yes*

and then

chain\scripts\install_choco.ps1 (Windows)

MacOS Open *Visual Studio Code* terminal and download the *protobuf* script with the following command:

sh Chain/scripts/install.sh (MacOS)

(3) Build your blockchain

Open *Visual Studio Code* terminal and start the node with the following commands in order:

Cd chain/src/AElf.Boilerplate.Launcher

Dotnet build

Dotnet run bin/Debug/netcoreapp2.2/AElf.Boilerplate.Launcher

Running Demos

1. **Run *HelloWorld* Smart Contract Demo**

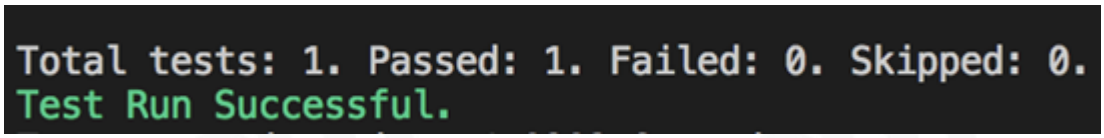
Test the contract by navigating to the *HelloWorldContract.Test* folder using the following command in the terminal:

```
Cd aelf-boilerplate/chain/test/HelloWorldContract.Test
```

Then execute the following command:

```
Dotnet test
```

Result:



2. Run *JS SDK* Demo

Open Visual Studio Code terminal to start the node

Run JS SDK Demo with the following command:

Cd.. *(Until you are back in the 'aelf-boilerplate' folder)*

Then navigate to the '*JSSDK*' folder using the following command

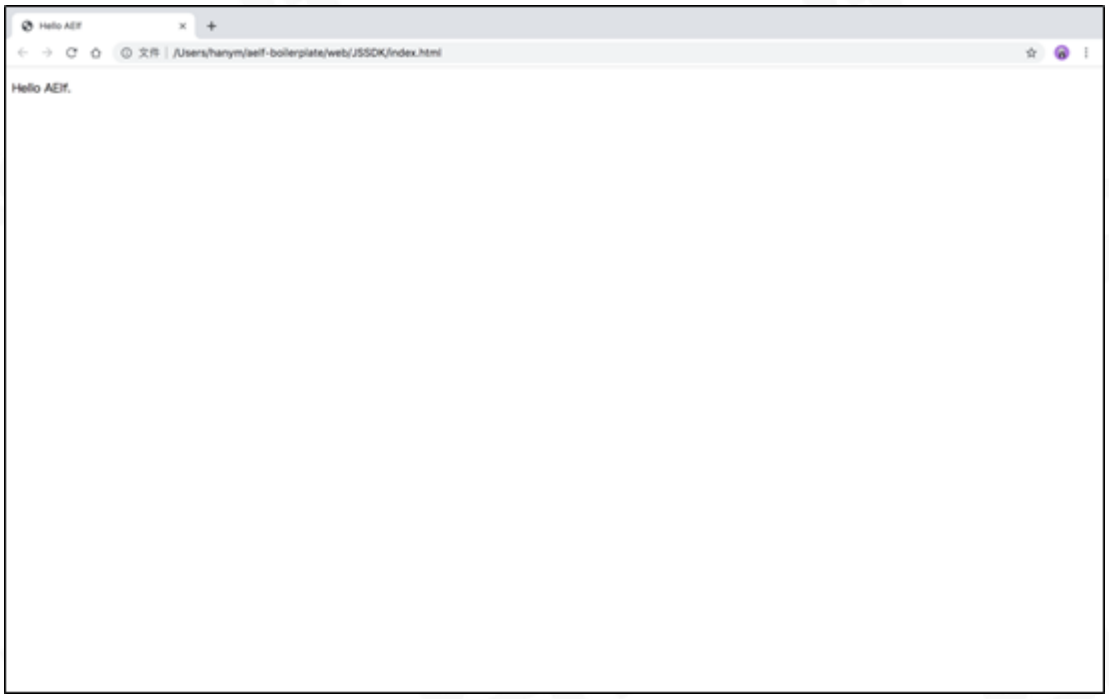
Cd web/JSSDK

Once in the folder, execute the following commands:

Npm install

Npm start

Result:



3. Run aelf browser plugin Demo

Open *Visual Studio Code* terminal to start the node

Run aelf browser plugin with the following command:

Cd.. *(Until you are back in the 'web' folder)*

Then navigate to the '*browserExtension'* folder using the following command:

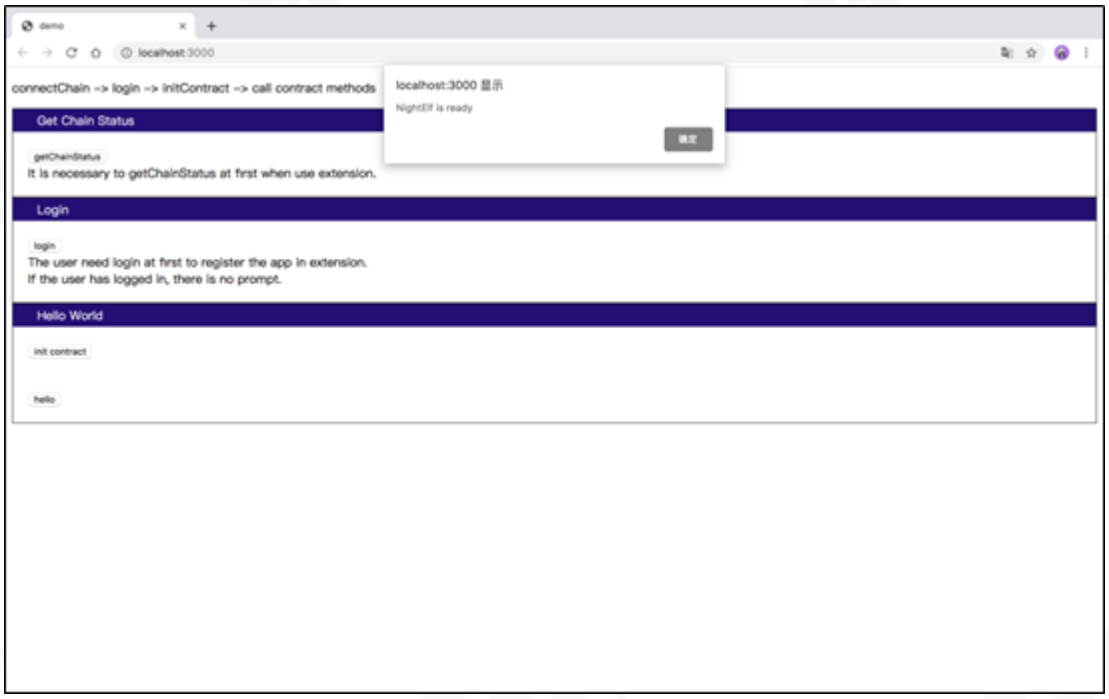
Cd browserExtension

Once in the folder, execute the following commands:

Npm install

Npm start

Result:



4. Run DAPP Demo - BingoGame

Open Visual Studio Code terminal to start the node

Run BingoGame with the following command:

Cd.. *(Until you are back in the 'web' folder)*

Then navigate to the '*browserBingo'* folder using the following command:

Cd browserBingo

Once in the folder, execute the following commands:

.\index.html

(Mac users: Open index.html)

Result: