Using Mac

How to set up a local development environment for Celo using Mac.

tip While many commands will be the same, you may need to follow the instructions for your specific OS when installing software. The Celo docs have some resources forWindows development . Before building on Celo, you need to set up a development environment to make sure you have the proper tools to build an application. This setup includes a combination of general development tools, Celo specific tools, and mobile development tools.

Web2 Prerequisites

Xcode

Xcode is Apple's integrated development environment for macOS, used to develop software for macOS, iOS, iPadOS, watchOS, and tvOS.

While Xcode falls under the Celo Prerequisites, Xcode takes a long time to download, which is why it is first on this list. It's best to start this download first, or even start downloading it before going to bed (depending on your internet connection).

Node, NPM and NVM

Node

Node is a JavaScript runtime that allows you to execute JS code outside a web browser. It is an open-source server environment and runs on various platforms.

NPM

npm stands for Node Package Manager ands the world's largest software registry. npm is what you'll use for installing published JS packages. npm comes bundled with Node, so you won't need to install it separately.

NVM

nvm stands for Node Version Manager. Different projects often require different versions of Node, and nvm makes it easy to switch between those versions. This is recommended if you plan to develop a lot of JS-based applications andnpm also recommends nvm . For Windows users, there arenodist andnvm-windows .

Install Node and npm via nvm

curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.39.1/install.sh |

bash Verify the installation by running:

command

-v nvm which should output nvm.

You can now install Node withnvm install.

info You can switch node versions using

nvm use < version

Homebrew

Homebrew is a way of managing packages on macOS.

Install Homebrew

/bin/bash -c

" (curl

-fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh) " Get the latest packages

brew update

Yarn

yarn is a package manager similar tonpm . For the most part, they do the same job and projects will let you know in theREADME orpackage.json which package manager they recommend using. Usually, you can use either or, but you should not use both in the same project. Running theCelo Development Blockchain requiresyarn , so you can install it now.

Install yarn

Option 1: Since you already havenpm installed, you can use it to installyarn

npm

install

-g

yarn This will install the Yarn global binary.

info The-g is short for--global . This flag means you are installing this package on your system and it will be accessible from the cli no matter what project you are in. Option 2: Installyarn with Homebrew

Update brew if you haven't recently

brew update

Install yarn

brew install

yarn

Expo

Expo is a framework and a platform for universal React applications. It lets you build mobile applications for both iOS and Android while using the same JavaScript/TypeScript codebase. This means that you don't need to be a "mobile developer" to build mobile apps!

Install Expo

Since you already setupnpm , you can now install theexpo-cli package.

npm

install

-g expo-cli

Docker

Docker is an application focused on build and deployment tools. It allows developers to build and share containerized apps. Rather than making the developer manually install a bunch of packages, a dev can just download a Docker image and start it and all the application installation and setup will be taken care of for them.

Install Docker

You might need to installDocker Desktop for your OS .

Once you've done that, you can accessdocker via your cli.

docker

--version

Celo Prerequisites

Truffle

Since the Celo Blockchain hasshared ancestry with Ethereum and maintains full EVM compatibility, you can use Ethereum

tools to develop Celo applications.Truffle is a development framework for Ethereum and assists with writing your contracts, testing, and deploying.

Similar to how you might use a framework like React instead of vanilla JavaScript, you can use Truffle to abstract away a lot of the imperative details from Solidity (the language used to write smart contracts).

Install Truffle

npm

install

-g truffle Configuring Truffle

You will need to configure Truffle to work with Celo. Connecting to Celo and managing transactions is easiest withContractKit . You can import contractkit directly into yourtruffle.config.js file in your Truffle project, add a private key and network details. You can seethis example config file for reference.

The Truffle deployer may have trouble estimating the deployment transaction gas limit, for which you will receive an error like:

Error: *** Deployment Failed ***

"Migrations" -- invalid argument 0 : json: cannot unmarshal invalid hex string into Go value of type hexutil.Bytes. You can resolve this by specifying thegas field in the network details intruffle.config.js .

For example:

alfajores :

{ provider : kit . connection . web3 . currentProvider , network_id :

44787 , gas :

4000000 , }

Local Development Blockchain

SeeLocal Development Chain w/ Protocol Contracts for how to get started with developing on a local test environment.

Celo CLI

TheCelo CLI is a command-line tool for interacting with the Celo Protocol smart contracts. Some of the things you can do with it include looking at accounts, checking balances, and signing transactions.

Install Celo CLI

npm

install

-g @celo/celocli

.. .

lots of logs

- @celo/celocli@x.x.xx added x packages from x contributors in x.xs info Installation can take over a minute, so be patient depending on your internet connection.

Celo Wallet

Celo Wallet is the development wallet to onboard onto the Celo network, manage funds, and send payments. Since Celo does not have a web wallet, you will need to use the mobile wallet.

There is adetailed write-up on the celo-monorepo for the Celo Wallet. This guide only includes the summarized installation instructions for macOS. If you're on another OS, please refer to that guide instead of this document.

info Celo's focus on mobile-first means that the priority has been on mobile applications and tooling. We know there is a growing demand for web applications and have several projects underway to build the tools needed to support web-based

applications. In the meantime, you can still use your web-savvy skills to develop mobile apps without too much of a learning curve because of frameworks like React Native and Expo.

Celo Wallet Prerequisites

Watchman

Watchman exists to watch files and record when they change. You can use it with the Celo Wallet.

Install Watchman

You can installWatchman with brew

Update brew if you haven't recently

brew update

Install watchman

brew install watchman

iOS

Xcode

Xcode allows you to build and deploy the Celo Wallet. If you do not have an iOS device, Xcode can emulate one. To install this, you need an Apple Developer account (free), but you don't need to be part of the Apple Developer Program (costs money) to download Xcode.

Install Xcode

Download Xcode from theApple Developer website . It is massive (10.6 GB), so we will start this first so it can be downloaded while we do everything else.

Once this is done downloading, install it.

Cocopods, Bundler

Navigate to the iOS directory of the mobile package (/celo-monorepo/packages/mobile/ios) and run the following to installcocoapods andbundler .

install cocopods and bundler if you don't already have it

gem install cocoapods

you might need to run with sudo if this fails

gem install bundler

download the project dependencies

bundle install

run inside mobile/ios

bundle exec pod install If your machine does not recognize thegem command, you may need todownload Ruby first.

Android

Java

Java allows you to build and deploy the mobile app to Android devices.

Install Java

Install by running the following:

brew install cask brew tap homebrew/cask-versions brew install homebrew/cask-versions/adoptopenjdk8

Android Dev Tools

Install the Android SDK and platform-tools

brew install android-sdk brew install android-platform-tools Next, installAndroid Studio and add theAndroid NDK .

info These paths may differ on your machine. You can find the path to the SDK and NDK via theAndroid Studio menu . export

ANDROID_HOME

/usr/local/share/android-sdk export

ANDROID_NDK

/usr/local/share/android-ndk export

ANDROID_SDK_ROOT

/usr/local/share/android-sdk

this is an optional gradle configuration that should make builds faster

export

GRADLE_OPTS

'-Dorg.gradle.daemon=true -Dorg.gradle.parallel=true -Dorg.gradle.jvmargs="-Xmx4096m -XX:+HeapDumpOnOutOfMemoryError"' Then install the Android 29 platform

sdkmanager 'platforms;android-29

Android Emulator

For the same reason you installed the emulator iOS (you may or may not have an Android device), you can also install the Android emulator.

Install Emulator Manager

One Android emulator option is Genymotion.

brew install genymotion Under OSX High Sierra and later, you'll get a message that you need toapprove it in System Preferences > Security & Privacy > General.

Do that, and then repeat the line above.

Then make sure the ADB path is set correctly in Genymotion — set Preferences > ADB > Use custom Android SDK tools to /usr/local/share/android-sdk (same as ANDROID_HOME)

Extras

Solidity support for VSCode

Solidity support for VSCode (or your preferred IDE/text editor)

Celo smart contracts are written in Solidity. While Solidity syntax is very similar to JavaScript/TypeScript, there are some differences and JS syntax support will not like Solidity files. Installing an extension on your IDE will be a big help as you develop smart contracts. Edit this page Previous Setup Environment Next Celo Local Envorinment using Windows