

Web3 Speed Run

for Developers



Evolution of the Web	1
What is Web3?	3
Web3 and your place in the future	3
Decentralized Internet and Money as a Protocol	4
Programming Web3 Apps	8
What is MetaMask?	10
MetaMask for developers	11
Programming Your First Dapp	12
Web3 Development Stack	12
Ethereum Boilerplate	13
Components	14
Hooks	16
Cross-Chain Support	17
Reading Blockchain Data with Moralis	17
Run the Boilerplate	18
Important Terms All Web3 Developers Should Know	19
Roadmap ahead	21
Quiz	24
Conclusion	25

Evolution of the Web

The internet is changing forever. So is the way we build internet applications.

You are reading this ebook because you're interested in new technologies, and want to level up as a developer.

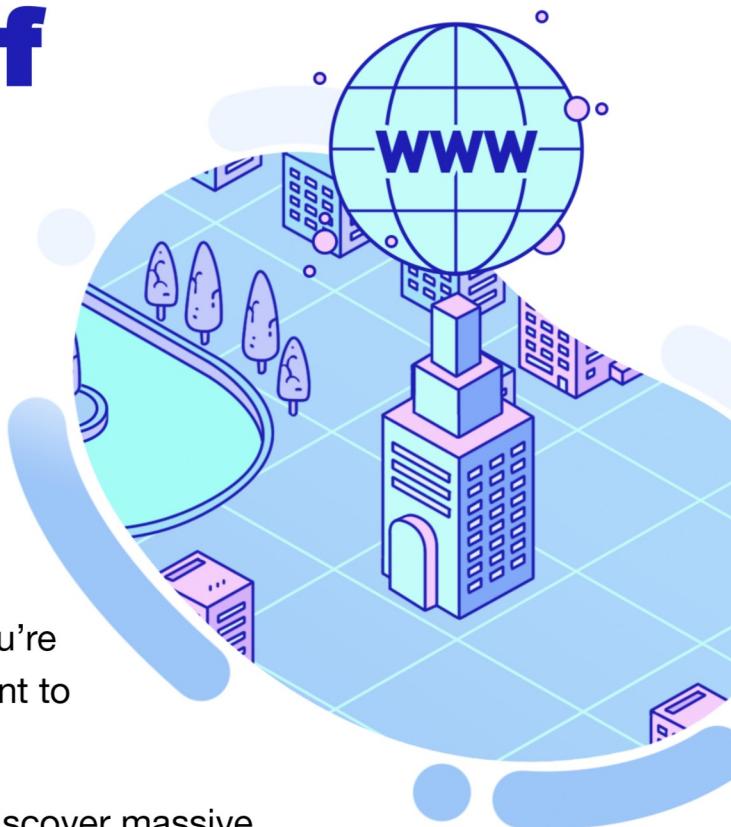
Every now and then, we (developers) discover massive opportunities. Think about the iPhone app explosion in 2009. The opportunity was there to get in early, and build your career or business on that wave of new tech. Instagram, Whatsapp and Snapchat are just a few success stories.

Or when Web2 was still new in the early 2000s, and we saw social networks like Facebook, search engines like Google, and online stores like Amazon grow to a global scale out of nowhere.

The developers that built these apps and technologies had one thing in common - they were early adopters.

Today, opportunities in Web2 development are near non-existent. Web2 is controlled by a few Big-Tech monopolies, and you have no real chance to compete with them.

But the tides are turning...



Web3 is reshaping the internet using open, peer-to-peer blockchain protocols.

The great thing is that you can reuse most of your Web2 skills when building a Web3 project or career.

This eBook will teach you:

1. What Web3 is and how it's different from Web2
2. How to build and run your first Web3 App
3. A clear roadmap of what to learn next

Pre-requisites:

1. Web Technologies like React
2. Being able to install packages with NPM
3. Interest in Web3 Technologies

At the end of the book there will be a Quiz to test your knowledge!

What is Web3?

Web3 and your place in the future



"The only way you can predict the future is to build it."

Alan Kay

Web3 isn't just coming. It's already here, and it's growing very fast. The Web3 market is already worth trillions, and is expected to 10x by 2030.

What does this mean for developers like you?

Massive opportunities, much like those of the iPhone app era and early days of Web2.

Web3 needs your skills now.

If you missed the mobile and Web2 revolutions - this is your chance.

Get in early, actively shape the next evolution of the internet, and build the future.

This ebook will equip you with everything you need to dive into the world of Web3 - right now.

We'll show you the huge professional and creative opportunities that Web3 offers, and how you can get involved fast using your Web2 skills.

Decentralized Internet and Money as a Protocol

Since the 1990s, the Web has constantly evolved. As of January 2021, around 60% of the global population are plugged in.

We're now at the next big turning point. But how did we get here? Here's an overview:

	Web1	Web2	Web3
Basic Concept	Static Content Sharing information	Dynamic Content Connecting people Centralized by Big Tech	Decentralized Internet Money as Protocol Removing Intermediaries
Type	Informative Web	Social Web	Decentralized Web
Year	1990 - 2000	2000 - 2020	2020 onwards

Web1: Read

Web1 was pretty simplistic. It was read only, with only a select few actively creating content. The vast majority of users simply consumed static pages and content.

Most websites were owned by companies and institutions. Most websites were also just (quite boring) recreations of traditional print media.



“The original thing I wanted to do, was to make it a collaborative medium, a place where we [could] all meet and read and write.”

Tim Berners-Lee's
Vision in 1989

Web2: Read / Write

Thankfully, Web2 (where we are now) was and is much more exciting. The introduction of frontend technologies like Javascript, and backend technologies like PHP, allowed developers to program dynamic websites.

User generated content and social networking created a new online age of creation and collaboration.

Today, we can read, write, view, and upload content on various platforms. We share, connect, and communicate with each other across the globe.

One problem is that most of Web2 is now controlled by a few Big-Tech corporations, and all user activity is monitored.

Another problem is that there is no way for developers to build financial applications. There is no way to program money. We can only integrate some payment processors.

Imagine if you could actually build and program money...

What if money worked like email, as an Internet protocol anyone can study and build upon, no matter where in the world they are, no matter who they are..?

Web3: Read / Write / EXECUTE

Although it's still early days, Web3 is out there right now.

Powered by blockchain, Web3 is already more open, fair, and empowering for individual users than Web2 can ever be.

It solves the problems we face in Web2.

► How?

By removing the need for trust, and by removing the need for permission.

To interact with Web3 applications, all you need is a wallet. This wallet contains your private key, which is used to sign transactions. Those transactions are then checked, confirmed and processed by the blockchain.

Blockchains are really just databases, but the crucial difference is they are split across thousands of different nodes, rather than centralized on a few servers.

Each node holds a complete copy of all data stored on the blockchain. Once data is verified, accepted, and written to the blockchain, it cannot be altered.

This means blockchains enable secure, peer-to-peer transactions without human intermediaries.

It also means anyone with a wallet can program and interact with them. No trust, no permission required.

The Time to Act is NOW

You can see now why Web3 will change everything. It already is.

Huge earning potential, especially for developers and innovators, is a major driving force. Blockchain developer salaries already average around \$100,000 per year, and this will only increase as Web3 adoption rises.

Web3 also opens the doors to massive innovation.

Let's be honest: there are no real surprises in Web2 development anymore.

Pioneers like Google and Facebook staked their claim early, and are now unshakable.

Web3, on the other hand, still has space for you to create a groundbreaking project, or trailblaze a world first approach to finance.

The gold rush has started, and as a Web2 developer, you already have an advantage.

It's already happening, in fact, and it's open to anyone. There are 13-year-olds in India building cross-chain financial dApps. Small dev teams in Australia creating decentralized data brokerages, solo game devs building play to earn MMO's, and much more.

This is all possible because blockchain technology removes the need for middle men through smart contracts - automated programmable protocols that exist on the blockchain.

There are no gatekeepers that can monopolize or lock out individuals.

That means anyone, anywhere with an internet connection and a small pool of starting funds, can enter the Web3 world of decentralized finance, gaming, and trading.

For developers, it also means there are no multinational conglomerates preventing you from building and deploying your ideas.

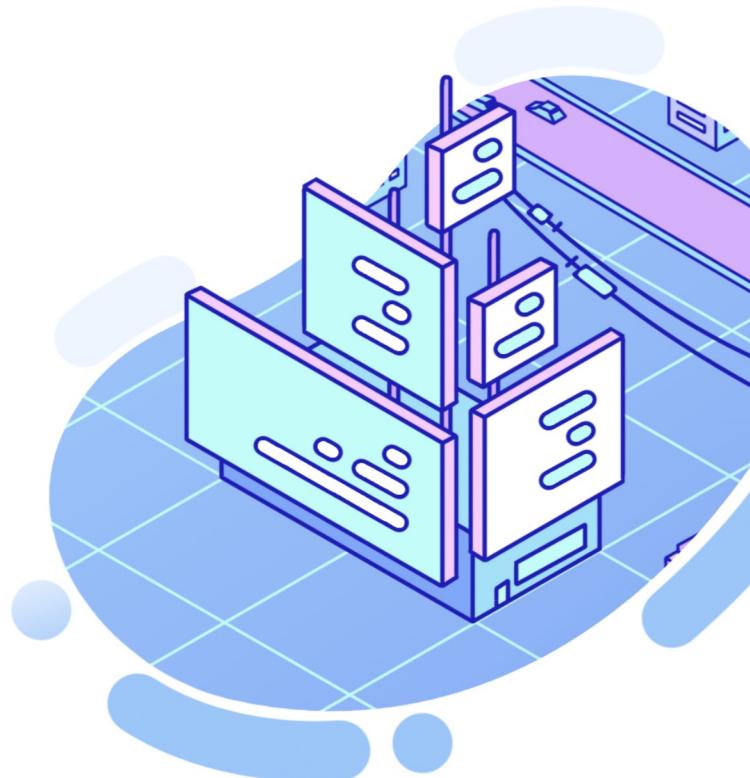
Programming Web3 Apps

Thanks to a well-developed infrastructure, Web2 development is easy these days.

Tools like Amazon Web Services, Microsoft Azure Cloud and Firebase take the headache out of hosting, scaling, and backend management.

The good news? Web3 also has powerful SDKs and tools that make development a breeze - especially if you know Web2.

If you're familiar with frontend or backend libraries like React or NodeJS, you're halfway there.



This ebook will show you how to start creating your own dApps, from setting up a wallet and DEX (decentralized exchange), to sending transactions. We'll cover all the basics, and all the key tools and SDKs you'll need.

► **So what's the first step?**

Understanding the main difference between how users interact with Web2 vs Web3

In Web2, we primarily use email and passwords; these are tied in some way to our identity. In Web3, we can login using a wallet like MetaMask (the most popular browser extension wallet).

These wallets don't require any form of personal ID. Instead, they use public and private key cryptography.

Let's explore what MetaMask is and how we can interact with it as developers.

What is MetaMask?

Introduction to MetaMask

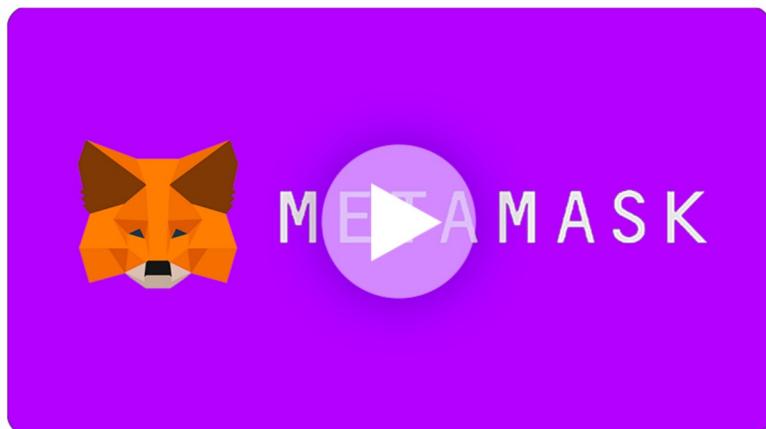
MetaMask is a popular software cryptocurrency wallet that is used to interact with the Ethereum Blockchain.

A software cryptocurrency wallet allows users to hold and control their digital assets through a browser extension or mobile app. It is also used to authenticate and interact with Web3 applications.

In August this year, MetaMask surpassed 10 million users. The company behind the wallet, ConsenSys, reported that MetaMask has grown more than 1,800% in a single year.

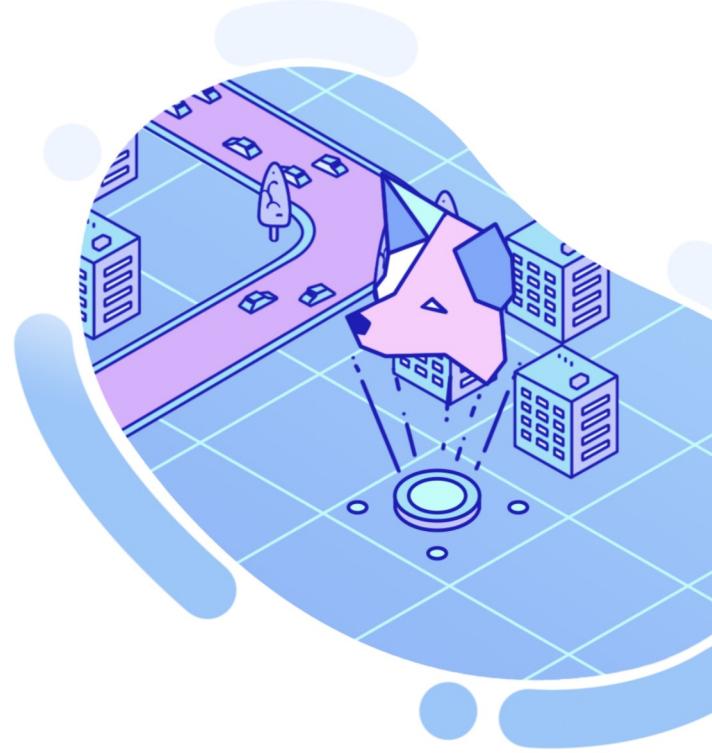
Get started with MetaMask by watching this video to get an initial understanding of what MetaMask is.

What is MetaMask?



MetaMask is a crypto wallet & gateway to blockchain apps, and is available as both a browser extension and as a mobile app.

🔗 <https://www.youtube.com/watch?v=YVgfHZMFFFQ>



Wallets like MetaMask are your identity in the Web3 universe.

In Web2 you authenticate with email and passwords, while in Web3 you authenticate with your private key, which wallets like MetaMask hold on your behalf. These wallets also connect to Web3 dApps.

MetaMask for developers

Now that you have a basic understanding of what MetaMask is, let's learn how to integrate it with your websites.

Watch this video to start with:

How to Use MetaMask as a developer:

<https://www.youtube.com/watch?v=9kdVAeZ7knk>

As you can see the video uses the Ethereum Boilerplate. Let's discuss what the Ethereum Boilerplate is in detail.

Programming Your First Dapp

In this section you will learn how to build and run your first Web3 application. Web3 applications are so called decentralized applications and are often referred to as “dApps”.



Web3 Development Stack

First and foremost it's important that you get an overview of the different technologies that are required to build a Web3 dApp.

The video below gives a great overview of the tech stack.

 [Blockchain Development in 2021](#)



Tech Stack Explained
by Ivan on Tech

 <https://www.youtube.com/watch?v=0zoJzQ4U740>

Most dApps need very similar base functionality.

This functionality includes:

- ▶ Logging in a user with MetaMask
- ▶ Fetching User Transactions
- ▶ Fetching User Token and NFT Balances
- ▶ Listening for real-time events that happen on chain

Luckily, we don't have to code this base functionality from scratch.

Instead we can use a boilerplate starter project with all of this already pre-programmed for us.

Ethereum Boilerplate

There are a variety of boilerplates and starter projects to help you build Web3 dApps fast.

The most popular is the Ethereum React Boilerplate which you can find here: <https://github.com/ethereum-boilerplate/ethereum-boilerplate>

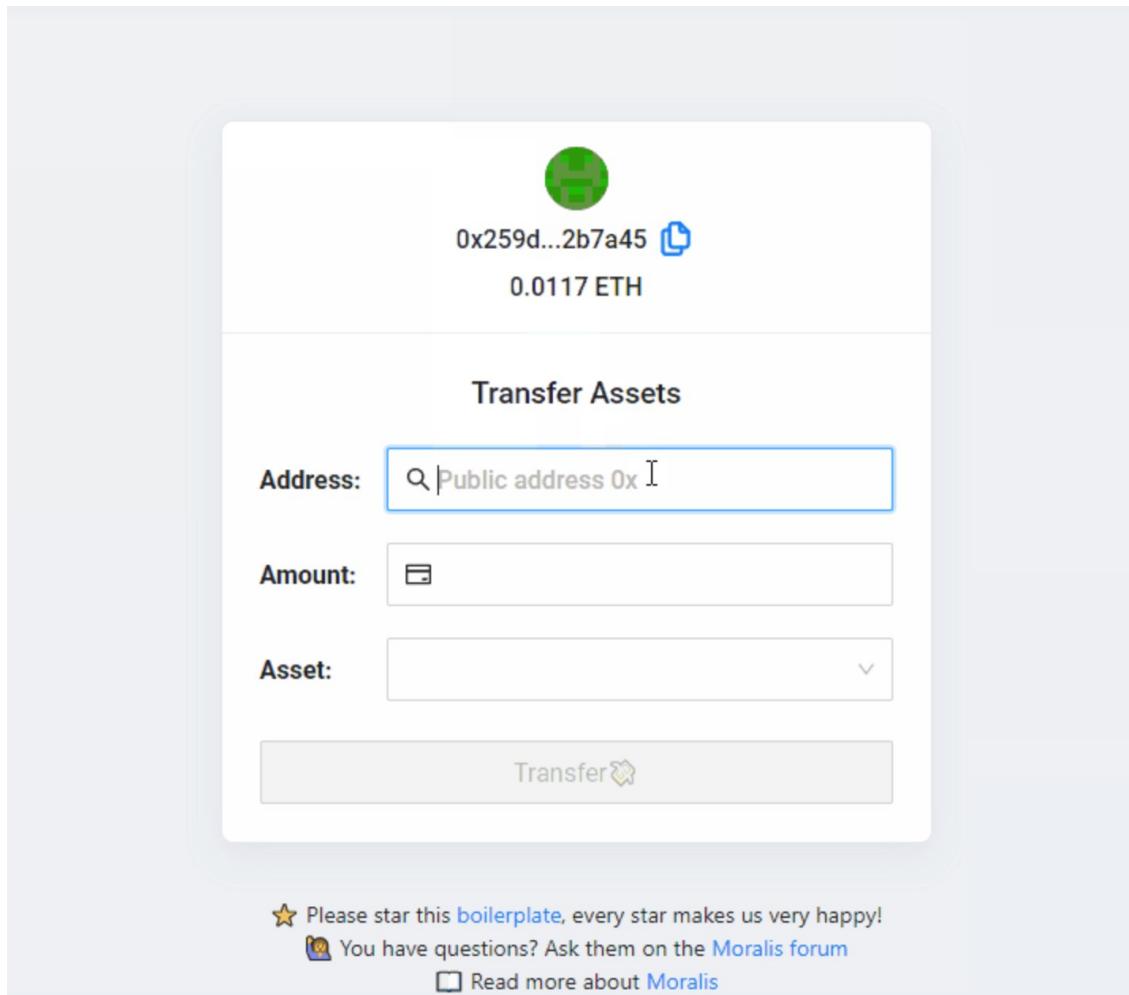
This boilerplate will save you months of work, as it comes with the most important Web3 hooks and components built-in. If you know React you'll feel right at home.

Watch this video to get an overview of the boilerplate:

<https://www.youtube.com/watch?v=svE0v0argZg>

Components

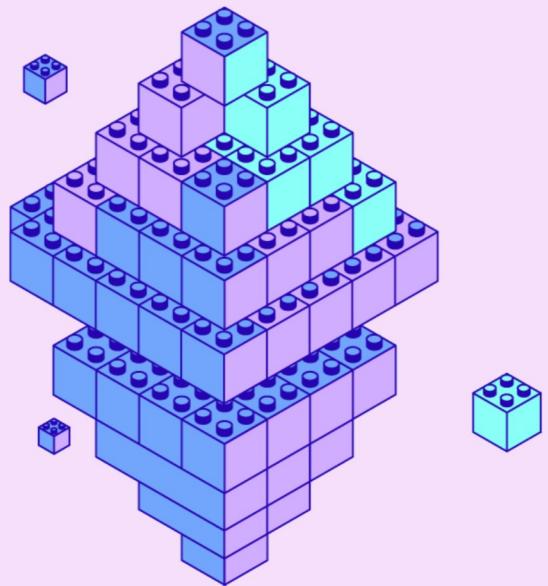
Components are finished, ready to use UI elements. For example the <Wallet /> component gives you an entire ready to use wallet that you can implement in your dApp.



Below is a list of currently supported components - check the boilerplate repo to see the [latest components](#). Don't worry if you don't understand what these components do. We will explain this more in-depth later.

🔗 Ethereum Components

- ▶ [`<Account />`](#)
- ▶ [`<AddressInput />`](#)
- ▶ [`<Chains />`](#)
- ▶ [`<CoinPrice />`](#)
- ▶ [`<ERC20Balance />`](#)
- ▶ [`<ERC20Transfers />`](#)
- ▶ [`<DEXs />`](#)
- ▶ [`<Wallet />`](#)
- ▶ [`<Blockie />`](#)
- ▶ [`<NativeBalance />`](#)
- ▶ [`<Contract />`](#)



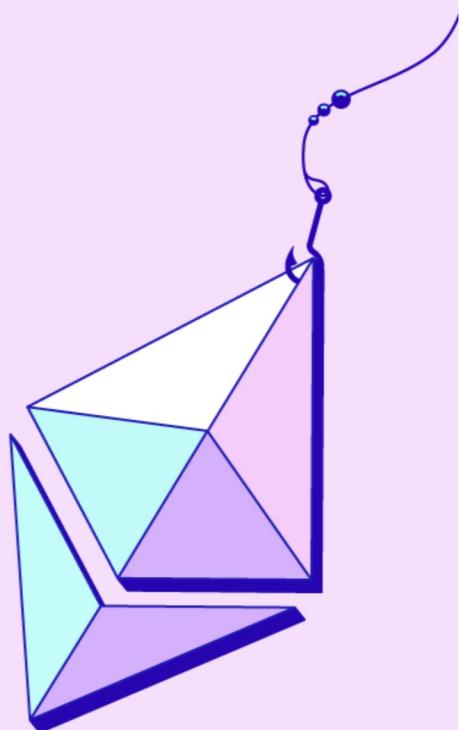
Hooks

Hooks are important features that all dApps need that allow you to fetch data. Hooks do not include user interface, which is great if you want to implement your own UI from scratch.

Below is a list of hooks that will ensure that you easily read and write blockchain data. Don't let this list intimidate you - we will explain what the different hooks do later on.

🔗 Ethereum Hooks

- ▶ [useAPIContract\(\)](#)
- ▶ [useWeb3Contract](#)
- ▶ [useERC20Balance](#)
- ▶ [useERC20Transfers\(\)](#)
- ▶ [useNativeBalance\(\)](#)
- ▶ [useNativeTransactions\(\)](#)
- ▶ [useNFTBalance\(\)](#)
- ▶ [useNFTTransfers\(\)](#)
- ▶ [useNFTTransfers\(\)](#)
- ▶ [useIPFS\(\)](#)
- ▶ [useChain\(\)](#)
- ▶ [useTokenPrice\(\)](#)
- ▶ [useInchDex\(\)](#)



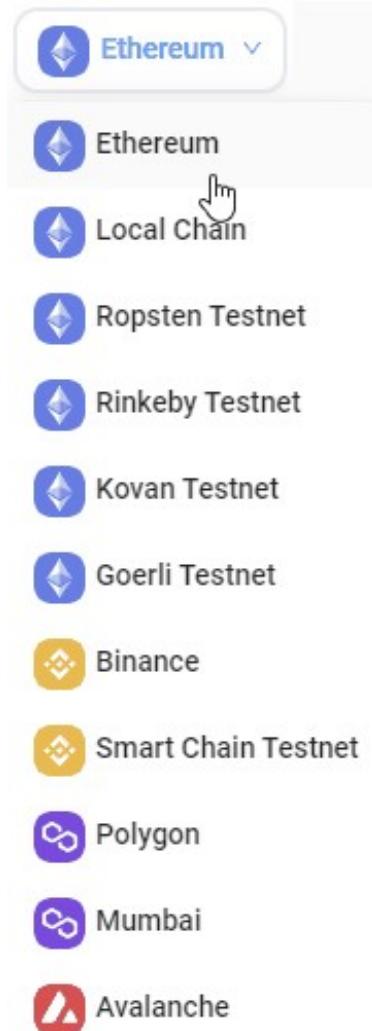
Cross-Chain Support

Although the boilerplate is called “Ethereum Boilerplate” it can actually be used to build and deploy dApps on practically any blockchain. Most blockchains today use the same programming language and have the same interface as Ethereum.

Such blockchains are called “EVM-compatible” as they run their own implementations of the Ethereum Virtual Machine.

You already saw in the “MetaMask for Developers” video linked above how this boilerplate supports different chains like Polygon, Avalanche, and Binance Chain.

More blockchains are spun up all the time. There’s even a chance more have launched while you’ve been reading this ebook!



Reading Blockchain Data with Moralis

The Ethereum Boilerplate above is based on Moralis, a piece of technology that Web3 dApps use to fetch data from the blockchain fast and easy.

Using Moralis in to connect your dApp to the blockchain speeds up dApp development by 80% or more. All the data you need is already extracted and indexed from the blockchain for you.

In the “[MetaMask for Developers](#)” video we show you how to sign up on [moralis.io](#), start a server and fetch the server url and app id in order to connect to the blockchain. Soon, you’ll be able to self-host your own Moralis server if you prefer, instead of using the one hosted by [moralis.io](#).

You can learn more about Moralis by reading the [docs](#). We highly recommend you learn all about Moralis as it will save you a lot of time and effort.

Run the Boilerplate

The most important thing is that you get the Boilerplate running on your computer. Connect it to Moralis by filling out the variables in the .env file and login into the boilerplate with your wallet.

Once you have the boilerplate running you have a solid base on which you can build whatever Web3 project you want.

If you have any questions about the boilerplate or need any tech help, please don’t hesitate to [ask in our forum](#) dedicated specifically for questions about this ebook.

Important Terms

All Web3 Developers Should Know

In the [Ethereum Boilerplate](#) the hooks and components refer to many dApp specific terminology like native, erc20, nft, dex, transfers and transactions. In this section our goal is to clarify some of these terms.

If something is not clear - [please post in our forum](#).

- 1. Native Balance** - this is the number of native tokens a user has in their wallet. For example on the Ethereum Network the native

token is ETH, on Binance Smart Chain the native token is BNB, and on Polygon the native token is MATIC.

The native token is used to pay the gas fees (transaction fees) on the network.

2. Native Transactions - all transactions that send native tokens or interact with smart contracts. Smart contracts are small programs that are running on the blockchain. You can deploy your own smart contracts.

3. ERC20 Token - The Ethereum Network allows developers to launch their own cryptocurrencies on top of the Ethereum Network. These cryptocurrencies are referred to as tokens.

To launch a token on top of Ethereum Network you have to write a smart contract that implements the ERC20 token interface. Most cryptocurrencies are implemented as ERC20 Tokens on Ethereum, Binance Smart Chain, Polygon or other EVM compatible chains.

4. ERC20 Balances - all ERC20 token balances that the user has in their wallet.

5. ERC20 Transfers - all ERC20 token transfers that the user has done.

6. DEX - means Decentralized Exchange which is a piece of code that can convert any ERC20 token to any other ERC20 token. Think of it like NASDAQ but completely decentralized. It's just a piece of code swapping tokens.

There are many more terms in Web3 that you need to learn. If you have any questions feel free to [ask in the forum](#).

Roadmap ahead

Now you know the basics of Web3 programming, and can run a basic Web3 dApp based on our boilerplate. It's time to take the next step - tinkering with the boilerplate and building on top of it.

Below are some tutorials and courses that show you how to quickly and easily build different use-cases.

[Create an NFT Minter](#)



<https://www.youtube.com/watch?v=n6rPr0sStTl>

[Build Generative NFTs](#)



<https://www.youtube.com/watch?v=KBV4FrCv4ps>

 [Create an NFT Marketplace](#)



<https://www.youtube.com/watch?v=EMOYpgl5S1w>

 [How to Add Fiat Gateway to dApps \[Full Tutorial\]](#)



<https://www.youtube.com/watch?v=5MITnoBm7YQ>

 [How to Create a DEX Like Uniswap FULL COURSE](#)



<https://www.youtube.com/watch?v=XOvtnDx1m5c>

 [Launch Binance Smart Chain BEP20 Token](#)



<https://www.youtube.com/watch?v=h4gQD7R1r94&t=1s>

Quiz

Great work - you've made it all the way here!

We've prepared a short quiz for you to ensure that you've learned the most important things from this ebook.

This ebook has covered many different topics, and we hope this quiz will help you digest and remember the most important takeaways.

Quiz

- 1. What is the difference between Web3 and Web2?**
- 2. What are dApps?**
- 3. How do users authenticate themselves when using Web3 dApps?**
- 4. Can Web2 technologies such as React, React Native, NodeJS be used to program Web3 dApps?**
- 5. How do dApps use Moralis?**

Post your answers in the thread linked below. If you don't know how to answer or need clarifications you can ask by writing in the thread.

<https://forum.moralis.io/t/ebook-quiz-answers/4361>

Conclusion

Thanks for reading this short ebook.

We hope it will inspire you to continue learning, and dig deeper into Web3!

Connect with our Community

Our goal is that you should never get stuck when learning Web3 programming. We have a dedicated Web3 programming forum where our developers can offer support, and answer your questions.



Access the forum here:

<https://forum.moralis.io/>



Join Discord To Connect to The Community:

<https://moralis.io/mage>



Documentation:

<https://docs.moralis.io/>



YouTube Tutorials:

<https://www.youtube.com/c/MoralisWeb3>



Twitter:

<https://twitter.com/MoralisWeb3>