



NANODEGREE PROGRAM SYLLABUS

# Blockchain Developer



# Overview

Start by learning the fundamentals of the blockchain technology and create your own private blockchain. Learn basics of bitcoin platform and how to secure a digital asset using blockchain identity. Then gain deeper understanding of Ethereum platform, and use Solidity to develop, test and deploy your own decentralized app.

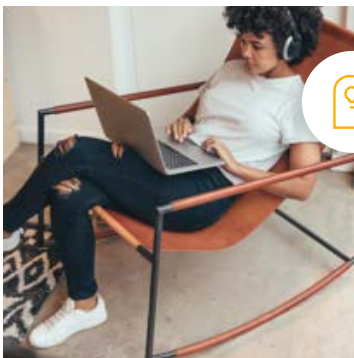
Continue to build on your blockchain skills with advanced topics such as privacy, security, payments, and oracles on blockchain. Architect and build complex systems on blockchain for different use cases such as supply chain tracking, insurance payments and decentralized marketplace



**Estimated Time:**  
4 Months at  
10hrs/week



**Prerequisites:**  
Object-oriented  
programming



**Flexible Learning:**  
Self-paced, so  
you can learn on  
the schedule that  
works best for you



**Need Help?**  
[udacity.com/advisor](https://udacity.com/advisor)  
Discuss this program  
with an enrollment  
advisor.

# Course 1: Blockchain Fundamentals

Learn the basics of how the blockchain data model works by creating your own private blockchain using Node.js and Leveldb.

## Course Project

### Create Your Own Private Blockchain

A blockchain is a shared database that features added immutability as a safe and accurate alternative to existing data storage methods. You'll learn the basics of how the blockchain data model works by creating your own private blockchain using Node.js and Leveldb.

In this project, you'll learn the fundamentals of architecting a collection of data into a blockchain data model. You'll configure how each block stores data, validate blocks, add new blocks to the chain, and create methods to validate the chain integrity.

You'll then create a back-end API web service, and migrate your private blockchain to the web service. In the process, you'll learn how to post new blocks to the blockchain via a RESTful web client. You'll then encrypt and decrypt the unique digital collectibles on a private blockchain.

## LEARNING OUTCOMES

### LESSON ONE

#### Blockchain Basics

- Explain core components that make a blockchain secure and powerful
- Define blockchain protocols and their key differences

### LESSON TWO

#### Managing Blockchain Transactions

- Create and manage identity on the Bitcoin Blockchain and establish proof-of-ownership with blockchain transactions, without the need to provide sensitive information

### LESSON THREE

#### Bitcoin core Testnet

- Explain the benefits of utilizing the Bitcoin Core testnet
- Describe the difference between the public testnet and regression testing

### LESSON FOUR

#### Blockchain Data

- Learn the relationship between different stages of transaction lifecycle using Bitcoin Core

### LESSON FIVE

#### Private Blockchains

- Explain the value of a private blockchain, and prepare for the course project

### LESSON SIX

#### Digital Assets on Blockchain

- Encode and decode digital assets on a private blockchain, and publicly prove ownership of the assets using digital identity



# Course 2: Ethereum Smart Contracts, Tokens and Dapps

Advance your blockchain skillset to the second generation of blockchain services with smart contracts utilizing the Ethereum network.

## Course Project

### Build CryptoStar Dapp on Ethereum

With Project 2, your focus moves from Bitcoin to Ethereum blockchain. You'll begin by building a decentralized app (Dapp) that allows you to create, sell, and transfer ownership of unique star token (CryptoStar) on the Ethereum blockchain using smart contracts and the non-fungible (ERC721) token standard. This service is designed to demonstrate how to claim and transfer ownership of unique digital asset ( e.g. document, deed, agreement, media, etc.) on Ethereum blockchain.

You'll build the back-end infrastructure for the CryptoStar with a pre-developed front end. This will enable you to render the service on any modern web clients.

## LEARNING OUTCOMES

### LESSON ONE

#### Ethereum Fundamentals and Development Tools

- Describe the fundamentals of Ethereum and how it is different from Bitcoin
- Build, compile, deploy, and test smart contracts using remix, ganache, truffle, and infura

### LESSON TWO

#### Smart Contracts with Solidity

- Learn Solidity, a Turing complete smart contract language
- Learn about different token standards (ERC-721, ERC-20)
- Create a fungible (ERC-20) token on Ethereum using Solidity

### LESSON THREE

#### Ethereum APP

- Develop, test and deploy a fully-functioning Dapp that allows users to create, buy and sell unique stars

# Course 3: Blockchain Architecture

Learn blockchain architecture and advanced concepts such as privacy, security and decentralized file management.

## Course Project

### Ethereum Dapp for Tracking Items through Supply Chain

By the completion of Project 2, you'll have learned the importance of Proof of Existence, which is used to verify whether a digital asset is authentic and can be trusted. In this project, you'll scale up to architect a solution that verifies authenticity for a product when multiple actors are involved.

You'll build a supply chain system on Ethereum blockchain that allows users to verify the authenticity of an item as it passes through different hands. You will architect a Dapp (Decentralized Application) authenticity management system backed by the Ethereum platform. To do so, you'll scope out the needs of the various actors in the supply chain and create smart contracts that help track product origination and verify product authenticity. You'll then tie this all together with a simple front-end that allows users to manage the product life-cycle as the product moves through the supply chain.

## LEARNING OUTCOMES

### LESSON ONE

#### Planning Blockchain Solutions

- Learn the correct technology stack to layer services and provide software solutions
- Design supporting visuals with Unified Modeling Language (UML)

### LESSON TWO

#### Privacy

- Implement several techniques to enhance privacy of blockchain such as merkle trees, zero-knowledge proofs

### LESSON THREE

#### Security & Maintenance

- Identify architecture security and maintenance risks

### LESSON FOUR

#### Distributed file system

- Create your own website and Dapp on the new decentralized storage protocol

# Course 4: Dapp with autonomous smart contracts and oracles

Advance your blockchain skill set by developing a decentralized application (Dapp) that will perform actions based on external triggers, and handle payments.

## Course Project Flight Delay Insurance Dapp

In the real-world, many smart contracts perform actions based on external triggers. These triggers can be caused by our actions, or by data received from outside sources using what are known as "oracles." To make real-world decentralized applications, smart contracts need to respond autonomously to these triggers, thereby making these applications more interactive.

In this project, you'll build a decentralized application for a use case in which you have airlines that offer flight delay insurance, and passengers who carry this insurance, and who get paid in the event their flight is delayed. You'll build multiple smart contracts which are autonomously triggered by external sources, and which handle payments based on flight delay scenarios.

### LEARNING OUTCOMES

#### LESSON ONE

##### Multiparty control and payments with smart contracts

- Build Dapp with secure, multi-sig smart contracts that autonomously receive, transfer, and pay funds.

#### LESSON TWO

##### ORACLES

- Utilize third-party data sources to inform autonomous smart contracts

#### LESSON THREE

##### Handling smart contract payments

- Create, and test, secure and cost-efficient smart contracts that handle, distribute, and test ETH payments to a smart contract



# Course 5: Capstone Project

For your capstone project, you'll use all the new skills you've acquired to build decentralized property listing application.

## Course Project Capstone

Here, you'll use all the new skills you've acquired to build decentralized property listing application.

In this project, you will represent your ownership of the property using ZK-SNARKs and then mint tokens to represent your claim to the property. You will then make these tokens available for sale on blockchain marketplace.

## LEARNING OUTCOMES

### LESSON ONE

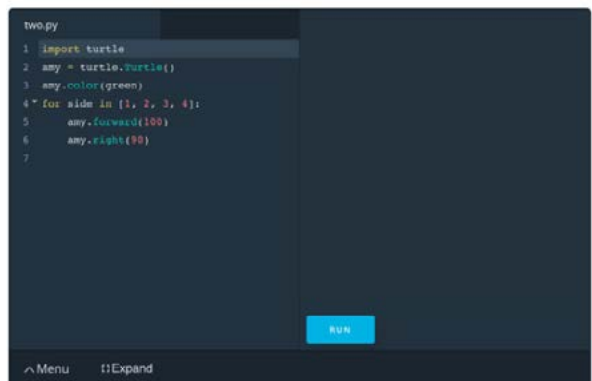
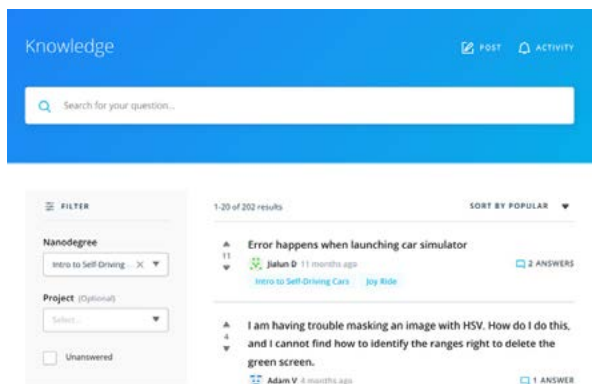
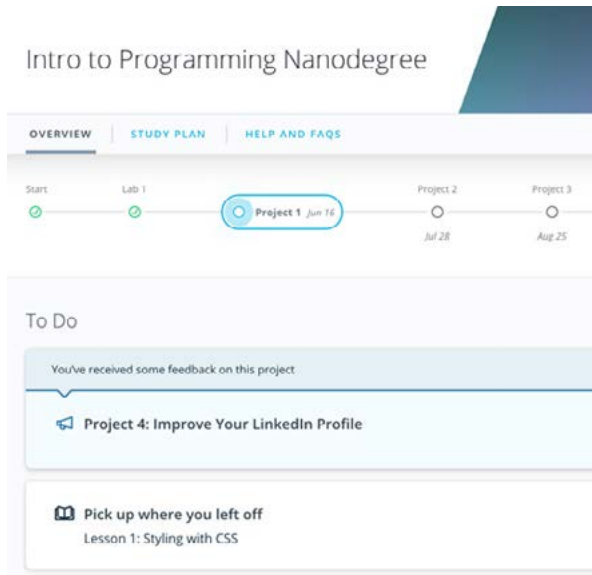
#### ZK-SNARKS

- Learn how to implement ZK-SNARKs using Zokrates framework





# Our Classroom Experience



## REAL-WORLD PROJECTS

Build your skills through industry-relevant projects. Get personalized feedback from our network of 900+ project reviewers. Our simple interface makes it easy to submit your projects as often as you need and receive unlimited feedback on your work.

## KNOWLEDGE

Find answers to your questions with Knowledge, our proprietary wiki. Search questions asked by other students, connect with technical mentors, and discover in real-time how to solve the challenges that you encounter.

## WORKSPACES

See your code in action. Check the output and quality of your code by running them on workspaces that are a part of our classroom.

## QUIZZES

Check your understanding of concepts learned in the program by answering simple and auto-graded quizzes. Easily go back to the lessons to brush up on concepts anytime you get an answer wrong.

## CUSTOM STUDY PLANS

Create a custom study plan to suit your personal needs and use this plan to keep track of your progress toward your goal.

## PROGRESS TRACKER

Stay on track to complete your Nanodegree program with useful milestone reminders.

## Learn with the Best



**Brandy Camacho**

INSTRUCTOR

Brandy is Curriculum Lead for the Blockchain Developer Nanodegree program. She was previously a Lead Technical Project Manager for Udacity. She is the founder of Network Designs, specializing in architecting software solutions.



**Jessica Lin**

INSTRUCTOR

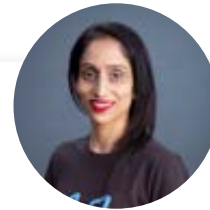
Jessica holds a Biomedical Engineering degree from Cornell University. Prior to joining Udacity, she developed and deployed enterprise healthcare technologies. She has recently taught Android Development in our Nanodegree programs.



**Joe Nyzio**

INSTRUCTORE

Joe Nyzio earned his degree in Neuroscience from Temple University. At Udacity, he's been a Content Developer on the Data Analyst, Tech Entrepreneur, and Business Analyst Nanodegree programs.



**Rachna Ralhan**

INSTRUCTOR

Rachna holds a Bachelor's of Electrical Engineering and a Master's of Software Management. She brings years of software and hardware engineering experience and has previously worked for companies such as Intel, AMD, Cypress, and Xilinx Semiconductors.

## Learn with the Best



**PK Rasam**

INSTRUCTOR

PK Rasam is Founder and Chief Blockchain Officer at LINCD, specializing in blockchain strategy and distributed ledger infrastructure services. PK's focus is on creating the next breed of Crypto Machine Intelligence-based businesses.



**Nik Kalyani**

INSTRUCTOR

Nik Kalyani is an entrepreneur and blockchain architect. He is the co-founder and CTO of WhenHub, creator of the Interface micro-contracts dApp. A Microsoft MVP, he is also the founder of TryCrypto and Walkstarter.



**Elena Nadolinski**

INSTRUCTORE

Elena Nadolinski is currently heads down building a new privacy cryptocurrency. Previously she was a Software Engineer at Airbnb, Tilt and Microsoft. Elena graduated with a degree in Computer Science from Virginia Tech in 2014.

# All Our Nanodegree Programs Include:



## EXPERIENCED PROJECT REVIEWERS

### REVIEWER SERVICES

- Personalized feedback & line by line code reviews
- 1600+ Reviewers with a 4.85/5 average rating
- 3 hour average project review turnaround time
- Unlimited submissions and feedback loops
- Practical tips and industry best practices
- Additional suggested resources to improve



## TECHNICAL MENTOR SUPPORT

### MENTORSHIP SERVICES

- Questions answered quickly by our team of technical mentors
- 1000+ Mentors with a 4.7/5 average rating
- Support for all your technical questions



## PERSONAL CAREER SERVICES

### CAREER SUPPORT

- Resume support
- Github portfolio review
- LinkedIn profile optimization

# Frequently Asked Questions

## PROGRAM OVERVIEW

### WHY SHOULD I ENROLL?

Demand for blockchain developers is growing at incredible rates, with current demand far outstripping available talent. This means wide open opportunity for anyone seeking to enter this field. Plus, blockchain is still an emerging technology, so you have the chance to start making an impact right away. The critical differentiator in an emerging field is hands-on experience, and that is exactly what you'll get in this program. You'll build projects designed in collaboration with working blockchain developers. You'll enjoy direct access to industry leaders in the field. You'll work with the Bitcoin Core and Ethereum platforms, and you'll even build your own private blockchain. If you're an experienced developer interested in pursuing a career in the blockchain space, this program offers you hands-on experience building decentralized applications, and the opportunity to master the blockchain development skills that are in high demand across industries, and at companies ranging from startups to large organizations.



### WHAT JOBS WILL THIS PROGRAM PREPARE ME FOR?

Graduates of this program will be valuable additions to any team working on blockchain-based application development. Opportunities exist in companies ranging from Fortune 500 companies to startups.

Specific roles include:

- Blockchain developer
- Blockchain engineer

### HOW DO I KNOW IF THIS PROGRAM IS RIGHT FOR ME?

If you're an experienced developer interested in working with blockchains, smart contracts, and cryptocurrencies, this program will enable you to master the skills necessary to succeed in this rapidly-growing field.

## ENROLLMENT AND ADMISSION

### DO I NEED TO APPLY? WHAT ARE THE ADMISSION CRITERIA?

No. This Nanodegree program accepts all applicants regardless of experience and specific background.

### WHAT ARE THE PREREQUISITES FOR ENROLLMENT?

To optimize your success in this program, we've created a list of prerequisites and recommendations to help you prepare for the curriculum. Prior to enrolling, you should be able to:

- Code with object-oriented programming
- Work with asynchronous JavaScript code

## FAQs Continued

- Develop the front end and back end of a web application with JavaScript
- Create and consume data using a remote API

If you believe you need more preparation, here are some additional resources you can use:

- [Intro to Computer Science](#)
- [Intro to JavaScript](#)
- [Intro to Programming Nanodegree program](#)
- [Full Stack Web Developer Nanodegree program](#)
- [JavaScript Promises \(for asynchronous programming\)](#)

### IF I DO NOT MEET THE REQUIREMENTS TO ENROLL, WHAT SHOULD I DO?

We have a number of Nanodegree programs and free courses that can help you prepare, including: [Intro to Computer Science](#), [Intro to JavaScript](#), [Programming Foundations with Python](#), [Intro to Programming Nanodegree program](#), [Full Stack Web Developer Nanodegree program](#).

## TUITION AND TERM OF PROGRAM

### HOW IS THIS NANODEGREE PROGRAM STRUCTURED?

The Blockchain Developer program is comprised of content and curriculum to support five (5) projects. We estimate that students can complete the program in four (4) months working 10 hours per week.

Each project will be reviewed by the Udacity reviewer network. Feedback will be provided and if you do not pass the project, you will be asked to resubmit the project until it passes.

### HOW LONG IS THIS NANODEGREE PROGRAM?

Access to this Nanodegree program runs for the length of time specified in the payment card above. If you do not graduate within that time period, you will continue learning with month to month payments. See the [Terms of Use](#) and [FAQs](#) for other policies regarding the terms of access to our Nanodegree programs.

## SOFTWARE AND HARDWARE

### WHAT SOFTWARE AND VERSIONS WILL I NEED IN THIS PROGRAM?

For this Nanodegree program you will need a desktop or laptop computer running recent versions of Windows, Mac OS X, or Linux. 20+ gigabytes of free disk space, 2+ gigabytes of memory (RAM), and an unmetered broadband Internet connection.

