



## Deployment

So far we have implemented and tested our dapp against **ganache**. At this point, if you have implemented all the functionality you like, you can deploy your dapp to one of the testnets (Ropsten, Rinkeby or Kovan). Follow the steps below to deploy and setup your dapp.

1. Start your Ethereum client (geth, parity etc) and make sure it is fully synced.
2. Remove the build/ directory which has ganache related content and run **truffle migrate** again.

If you want to host your application on a web server so the entire world can access it, follow the steps below. This assumes that your users will use metamask to interact with your dapp.

1. Instead of running your own IPFS node, you can just use a free hosted service such as Infura. In your app.js, replace IPFS config from localhost to Infura.

```
const ipfs = ipfsAPI({host: 'ipfs.infura.io', port: '5001', protocol: 'http'})
```

2. You might have people who don't have metamask visit your website. Instead of not showing them anything, let's at least display the products. To do that, let's again use infura's hosted Ethereum node instead of running our own. To do this, signup for free on Infura (<https://infura.io/>). Once you sign up, you should receive an API key. Use that API key and update your web3 provider in app.js from localhost to Infura's server like below

```
window.web3 = new Web3(new Web3.providers.HttpProvider("https://ropsten.infura.io/API_KEY"));
```

3. You have to then package your javascript and html files so they can be deployed to your web server. To do that, simply run the command **webpack** in your project directory and it will output all the js and html files to the build directory.
4. Copy the js and html files in to your web directory on your web server and your dapp should be accessible by anyone!

Again, do not hesitate to contact us if you run in to issues with deployment.

[Previous \(/courses/3/lessons/8-7\)](/courses/3/lessons/8-7)[Next \(/courses/3/lessons/9-2\)](/courses/3/lessons/9-2)

