

Setting up Your Development Environment

Getting started is often the most frustrating part of application development. Fortunately, many of the modern JavaScript frameworks have made great strides in providing tools to get you up and running as quickly as possible. Below, we walk through the basics of getting your system set up to develop.

Integrated Development Environment (IDE)

There are a bunch of different IDEs on the market, and in general you should be able to use any of these that you like and are comfortable with. If you don't have a personal preference, we recommend trying [Visual Studio Code](#) - it is quite excellent and comes with a variety of tools to make your development experience seamless. And it's free!

Git

Chances are, you probably want to keep your application in source control. Git is one of the big players in this space (it's also the backbone for BitBucket).

You may already have git installed. To check, run:

```
```sh git --version ```
```

in a command prompt. If you don't have git installed, you can download it from the [Git Website](#).

If you are behind a proxy, you'll also need to set up the git proxies as well:

```
```sh git config --global http.proxy http://url.to.your.proxy:port git config --global https.proxy http://url.to.your.proxy:port ```
```

The proxy addresses may vary depending on your location. Consult your local IT department if you're not sure.

Node

The majority of JavaScript frameworks (and Brightlayer UI resources) are available via the Node Package Manager (NPM). This package manager allows you to install a variety of third party tools, packages, and other dependencies that you need for your applications. To use NPM, you will need to install [NodeJS](#).

Yarn

Our projects and example code are set up to be run using yarn, but if you prefer, you're welcome to use NPM in your projects instead. Instructions for installing yarn on various operating systems can be found on the [Yarn Website](#).

To check if you have yarn installed, run:

```
```sh yarn -v ```
```

from a command prompt. If you are behind a proxy, you will also need to configure your proxy settings. It may be sufficient for you to set `HTTP\_PROXY` and `HTTPS\_PROXY` in your environment variables. However, you may need to also explicitly set proxy values for yarn:

```
```sh yarn config set proxy http://url.to.your.proxy:port yarn config set https-proxy http://url.to.your.proxy:port ```
```

The proxy addresses may vary depending on your location. Consult your local IT department if you're not sure. Once you have configured the proxy, you should be able to install packages by using:

```
```sh yarn add ```
```

*If you are installing packages globally using yarn, you will need to ensure that your PATH variable has an entry pointing to your global yarn install directory. This location*

*will vary based on your operating system.*

---

# Hybrid App Development

---

If you are setting up for building a mobile app, there are some additional steps you will want to take to prepare. The following requirements are the basics. Depending on your selected [framework](#), there will be other specific setup steps detailed in our Getting Started Guides.

## Prerequisites

### Hardware

Mobile development is resource-intensive, so you will need a development machine that is up to the task. Your machine should at least match the following minimum requirements:

- a Mac (if you plan on building or testing iOS locally)
- 16GB RAM or more (running multiple emulators can easily push you over 20GB)
- Solid State Hard Drive (SSD)
- 5/6 Gen Core i7 with Virtualization Technology or better

### Software

There are a few software tools that you will need before you can get started:

- [Android Studio](#) (required for building for Android)
- [xCode](#) (Mac-only, required for building for iOS)