**React Native Custom Template**

**Main Steps Involved**

1. Creating the template's repository
2. Creating the app template (make the React Native project you want to use as your template)
3. Creating the actual template structure (the app template + other config files)
4. Testing it!

## **Let's Start Initial Repo Setup**

# I'm going to setup a sample repository on Github called ****testRNTemplate**** (you can name yours whatever you want). This will contain the sample template for this article.

## **Create Initial Template App**

On your local machine, open the terminal and create an initial React Native App using

npx react-native init ProjectName

**Note: It's absolutely important to follow the capitalizations in exact name**ProjectName because the flag **--template** will convert ProjectName into the name of your app.

## **Configuring the Template App**

When you finally get your App try to run it first and make sure that it installed successfully by displaying the default React Native Welcome Screen.

## **Installing Project Dependencies and Setup**

In my case, I wanted **React Native TypeScript** and **UIKitten** install them using **yarn or npm.**

## **Testing new libraries on App.js (as a demo)**

We should check if the installation of the packages was successful by changing the code in our **app.js**

## **Creating the actual template structure**

Every custom template needs to have configuration file called **template.config.js** in the root of the project:

module.exports = {

// Placeholder name that will be replaced in package.json, index.json, android/, ios/ for a project name.

placeholderName: 'ProjectName',

// Placeholder title that will be replaced in values.xml and Info.plist with title provided by the user.

// We default this value to 'Hello App Display Name', which is default placeholder in react-native template.

titlePlaceholder: 'Hello App Display Name',

// Directory with the template which will be copied and processed by React Native CLI. Template directory should have package.json with all dependencies specified, including `react-native`.

templateDir: './template',

// Path to script, which will be executed after initialization process, but before installing all the dependencies specified in the template. This script runs as a shell script but you can change that (e.g. to Node) by using a shebang (see example custom template).

postInitScript: './script.js',

};

Create package.json and add your configurations.

{

"name": "rntemplate",

"version": "0.0.1",

"description": "React Native Template",

"repository": "https://github.com/auto-coder/RNTemplate.git",

"author": "Ahmet",

"license": "MIT"

}

**Note:** Please remember to change your repository based on your specific repository because that repository is specific to the repository I made in Step 1.

**You can create script.js**

## **template is my project name!**

## 

## **Continuation**

At the **testRNTemplate** folder, open a terminal and you can now push all of the code to the repository we created in 1.) using git.

**Give it a try by opening a terminal and running**

npx react-native init SomeApp --template https://github.com/auto-coder/RNTemplate.git

Installing from npm is probably the easiest way to go, and [publishing to npm is SUPER easy](https://docs.npmjs.com/getting-started/publishing-npm-packages). Once published to npm, you’d use your template like so…

npx react-native init MyAp --template rntemplate\_logo