A quote for you

# Introduction

What first pops into my mind to build a random quote generator application, is that it should be classy and simple. This would mean a clear heading, a large quote block and some social and other actions at the bottom. Optionally we want to click on the author to get more quotes from that specific author. Also, we want to base the design and usability on mobile-first.

## URLs

|  |  |
| --- | --- |
| URL | <https://quoteforyou.netlify.com/> |
| GitHub | <https://github.com/jnieberg/a-quote-for-you> |

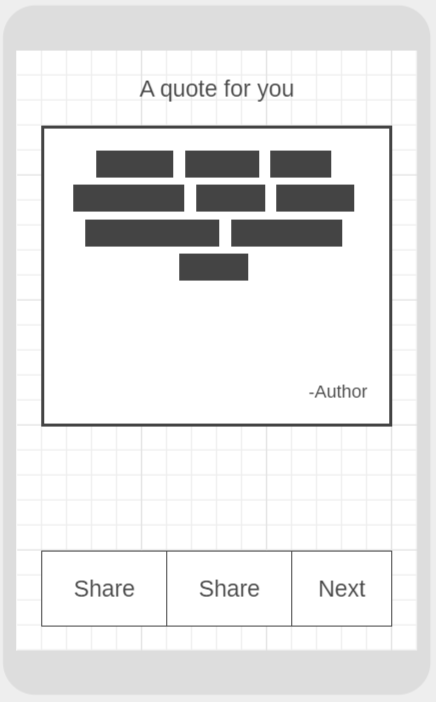
## NPM Commands

After cloning the app from the GitHub repository, use one of the following commands:

# Setup

## Wireframe

I came up with the following wireframe. It contains a heading, a quote and author block, 2 social buttons and a button to refresh the quote.



### Heading

A simple heading “A quote for you”

### Quote block

The quote block contains the quote and an author. Clicking on the author will show a random quote from the specific author.

### Share block

I decided to add two share buttons: Facebook and Twitter. There is also a *next* button to show a new quote.

## Theme

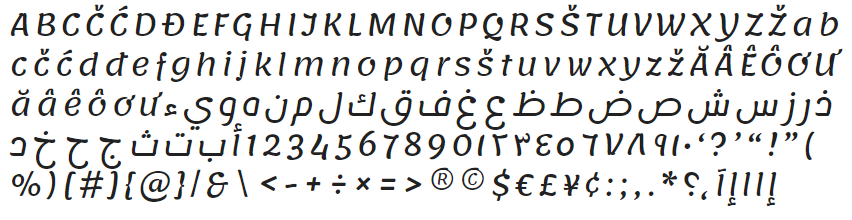
### Colours

The theme of the application should be classy and warm. Hence, using warm colours and a beautiful font. I found a [colour scheme generator](https://coolors.co/edcb96-f7c4a5-9e7682-605770-4d4861), that gave me the following pattern:



### Font

Additionally, I decided to use a Google font for the quote block. In this case it became [*Lemonada*](https://fonts.google.com/specimen/Lemonada). For the other text I decided to go for *Segoe UI*.



### Images

I also wanted to make the quote panel look like something you hang on your wall at home, so I found a good *SVG* border for it.



## Framework

### Initialization

I decided to use Angular 8 for this application. I have setup the framework with the use of Angular CLI and used a router, just in case I need one.

### Language

To assure the code quality, TypeScript would be a good set to work with.

### Styling

I decided to choose Stylus as a pre-processor for CSS. This because the linting and automatic sorting extensions available in Visual Studio Code are very useful for this pre-processor.

### Testing

Because testing is always necessary, I decided to use the standard Jasmine unit testing tool.

### Components and services

I decided to work with the following components:

* **Quote component.** Main area for the quotes.
* **Share component.** Area for the social share buttons.
* **Quote service.** Used for calling quote API.
* **Share service.** Calls the right configuration settings.

### Configuration file

All configurable text, API endpoint data, link URLs and mock data is centralized into one configuration file. This makes is easier to apply changes to text without searching through the correct files.