Documentation Guide

[Introduction](#_j3anwz85avhn) 2

[Tools](#_d1c0wfd35eqd) 2

[Environment Setup](#_3fo5tlvw2r80) 3

[Build and Run the document locally](#_mq0gnwjtrzi3) 4

[Project structure](#_gjpgie43ovrp) 6

[Creating files for documentation](#_3j2qqm3) 7

[Styling formats](#_afihd7cjiji8) 8

[How the right side menu is displayed automatically on the parent page.](#_ttoli2xzk3zo) 15

[How to Modify the header bar](#_snl7158pjc72) 17

[How to Modify the Footer](#_gzfcy7ufsqug) 18

[How to Change the font color.](#_cqa03sqde7i2) 19

[How to add Search bar](#_l26jov7zby80) 20

[How to add a video](#_zgtywm1270jr) 21

[Site Configuration](#_vv1r2jx6c22k) 22

[Sidebar content creation](#_9xn4h2wovuxp) 24

[How to publish the document](#_qu3l2kd2iyz) 27

[How to add new Images in Documentation](#_9l3jmud3k4g7) 28

[How to commit files to GitHub.](#_7cwkaql0d6ri) 30

[How to Create Pull Request (PR)](#_slb9u05lbpqz) 34

[How to resolve Github Conflict](#_xjnlo81ciwxo) 36

[Documentation checklist](#_xqfkt5zganxi) 36

[**Quick Links**](#_hc7lcy8oai4t) **37**

## 

## 

## 

## Introduction

This guide provides the steps and procedure to create documents for:

* Knowledgebase
* User Guide
* User Support

## Tools

[Docusarus](https://docusaurus.io/docs/en/installation) - To create a web based portal to display the visual content and text of User Documentation.

[Visual Studio Code](https://code.visualstudio.com/download) - IDE

[NodeJs](https://nodejs.org/en/download/) - Framework

[Git](https://git-scm.com/downloads) - Source Code Management

## 

## Environment Setup

1. Download ***NodeJS*** according to the Operating system used.
2. Install ***npm*** ( Node Package Manager ).
3. To verify whether NodeJs and npm are installed. Run the below commands,

**For NodeJS -** “Node -v”

**For Npm -** “Npm -v”

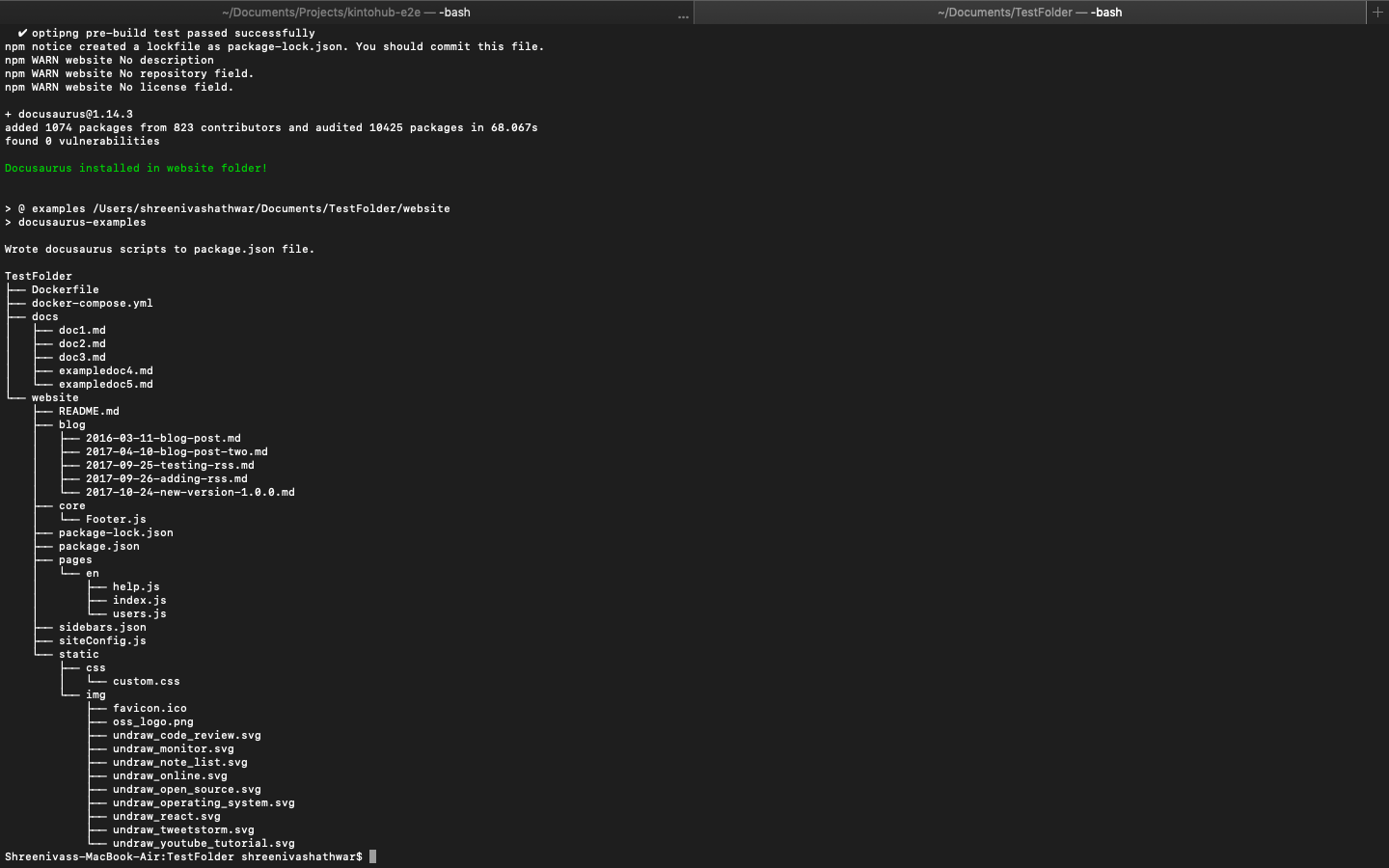
1. Download the ***Visual Studio Code.***
2. Download ***Git.***

To check git version - “**git --version**”

1. Open the “Terminal” and type cd Documents.
2. Create a new folder for the Documentation project using “**mkdir Testfolder**” ( New folder is created ).

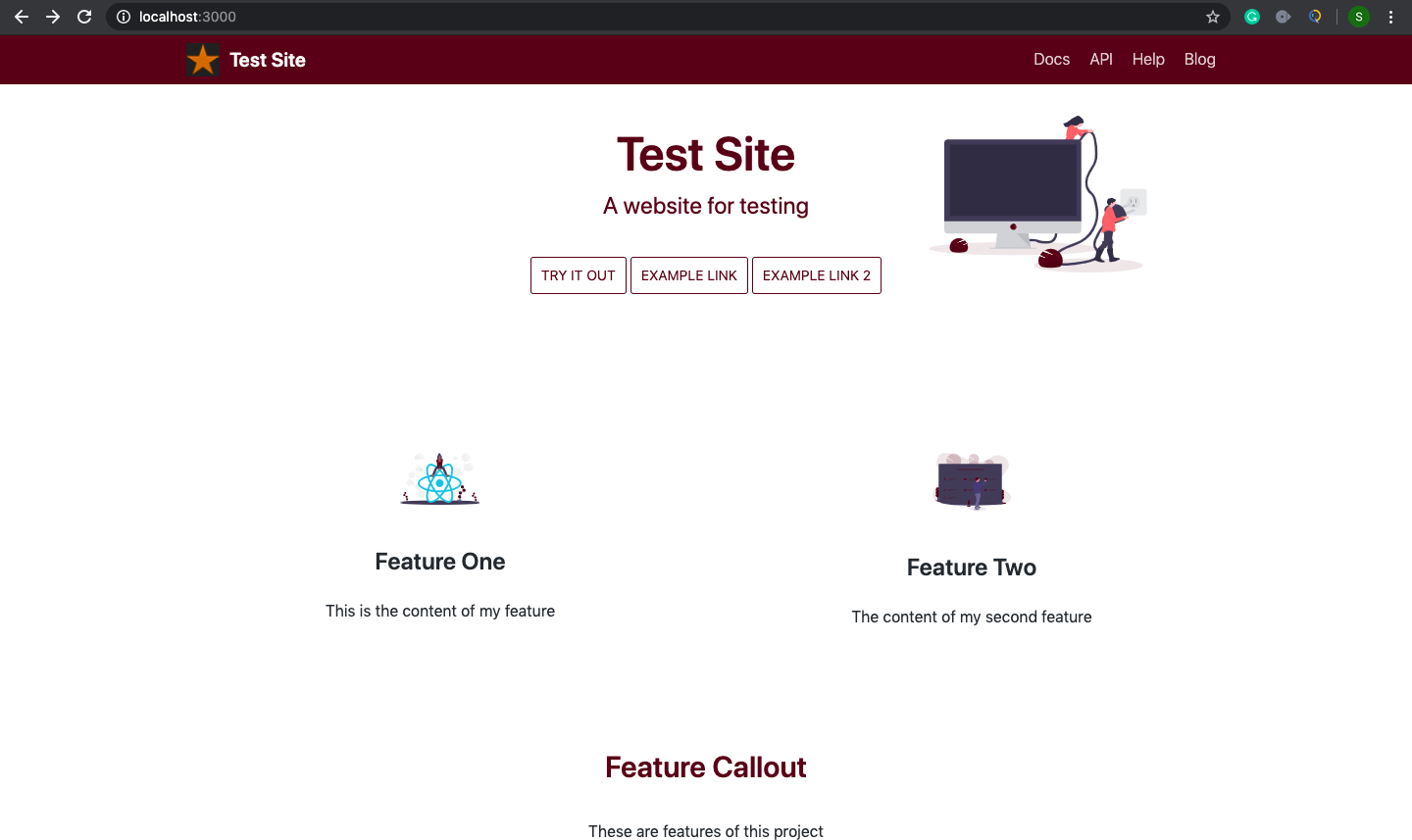
Note> If you have an existing then checkout to this folder and proceed to Step 13.

1. Run the “Docusarus” command “**npx docusaurus-init**” ( This will initialize the required files for Documentation ).
2. After running the “Docusarus” command, your “Terminal” should be displayed as shown below.



## Build and Run the document locally

1. To run the “Document” locally we must change the directory first using the “**cd website**”.
2. To run the site locally using the “**npm start**” command. ( The site should display as shown below ).



1. Replace the text inside the “**package.json**” file using the mentioned text below: ( [Click Here](https://drive.google.com/open?id=1vlnyT8PU4H1TyscfV_1xCUUnfa-izMgn) to see the sample of Package.json file).

{

"scripts": {

"examples": "docusaurus-examples",

"start": "npm-run-all -p watch-css dev:watch",

"dev:watch": "nodemon --exec npm run dev",

"dev": "docusaurus-start",

"build": "export $(cat .env | xargs) && npm run build-css && docusaurus-build",

"publish-gh-pages": "export $(cat .env | xargs) && docusaurus-publish",

"write-translations": "docusaurus-write-translations",

"version": "docusaurus-version",

"rename-version": "docusaurus-rename-version",

"build-css": "node-sass-chokidar style/ -o static/css/",

"watch-css": "npm run build-css && node-sass-chokidar style/ -o static/css/ --watch --recursive"

},

"devDependencies": {

"docusaurus": "^1.14.0",

"node-sass-chokidar": "^1.2.2",

"nodemon": "^1.17.2",

"npm-run-all": "^4.1.2"

},

"nodemonConfig": {

"ignore": [

"i18n/\*"

]

},

"dependencies": {

"underscore.string": ">=3.3.5"

}

}

1. Run the “**npm install**” command to install the contents in the “**package.json**” file.
2. To confirm that the changes have been updated by using the “**npm run dev**” command to host the site locally.
3. To create your own custom site remove the existing “**index.js**” file from “**website/pages**” and add your own “**index.html**”.
4. The “index.html” file should point “**docs/<Filename1>.md**”.
5. When you run the site locally the contents present in the “**docs/<Filename1>.md**” will be displayed.

## 

## Project structure

` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `

<Projectname>

|

|\_\_.circleci

|\_\_assets

|\_\_docs

| |\_\_assests

| |\_\_ ….

| |\_\_ ….

|\_\_website

| |\_\_ ….

| |\_\_package.json

| |\_\_sidebars.json

| |\_\_ siteConfig.js

| |\_\_ ….

| |\_\_ ….

|\_\_.gitignore

|\_\_package-lock.json

|\_\_README.md

|\_\_SETUP.md

` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `

**docs:** This is the folder where all the content files are stored.

***Note***: File created should be in the “**.md**” format.

**assets:** This is the folder where all the images needed for the documentation are stored.

***Note***: Images should be in the “**.png**” format.

To change images to .png format use this online [tool](https://image.online-convert.com/convert-to-png). ( If your using windows then only use this tool, Not recommended for Mac ).

**package.json:** This is the file that contains the development dependencies.

**sidebars.json:** This is the file used to create sidebar content.

**siteConfig.js:** This is the file that contains the configurations of the documentation site.

## Creating files for documentation

The new file “<File Name>.md” is created for adding a new separate section in an existing document. When a new file is created which is equal to one web page.

1. Open the documentation project “<Project Name>” in the Visual Studio Code.
2. On the right panel of the Visual Studio Code, you will get the project structure displayed.
3. Right-click on the “**docs**” folder and select the option “**New file**” to create a new file.
4. Give a name for the new file and make sure you mention “**.md**” at the end of the file name.

## 

## 

## Styling formats

1. How to mention the **title**,

` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `

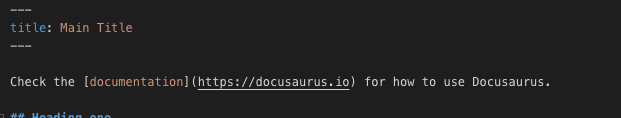
Example format:

**---**

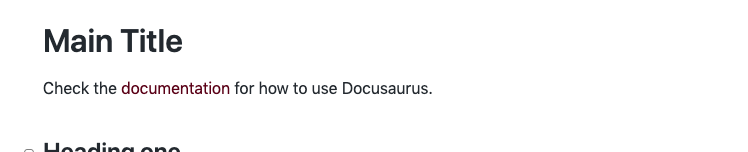
**title: Main Title**

**---**

` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `



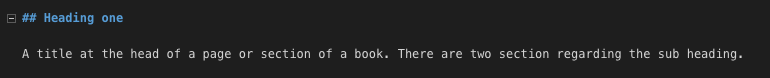
And this how it looks in the application.



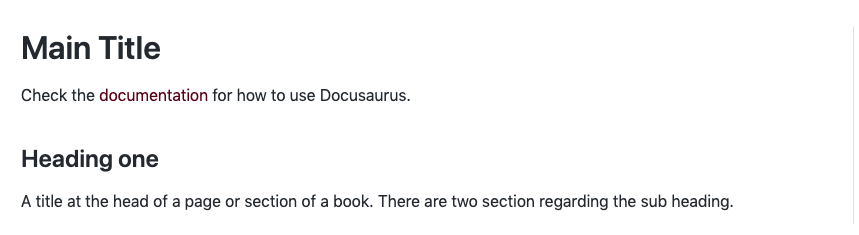
1. How to mention a **heading**,

` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` Example format: **## Heading one**

` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `



And this how it looks in the application.

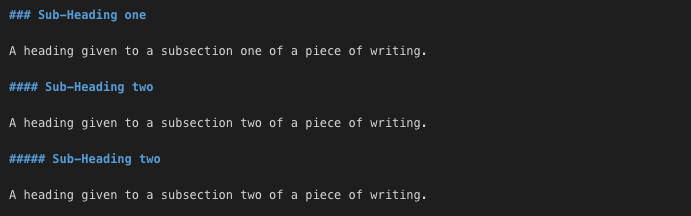


1. How to mention a **sub-heading**,

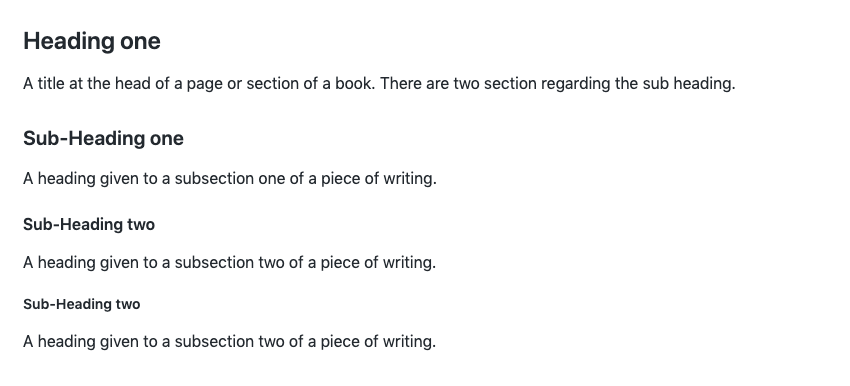
` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` Example format: **### Sub-Heading one**

` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `

**Note**: By increasing the number of **#** in the document, the heading size will also vary.



And this how it looks in the application.



1. How to attach an **image**,

` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `

Example format: **![<*Name of the Image*>](/docs/assets/auth-token.png)**

` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `



1. How to mention a **note**,

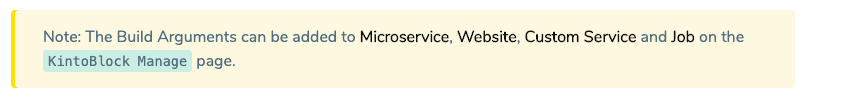
` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `

Example format: **> Note <Required Text>**

` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `



And this how it looks in the application.



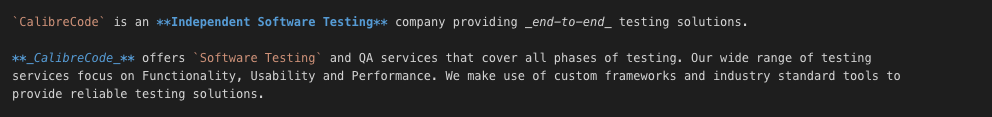
1. How to **highlight** words,

` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `

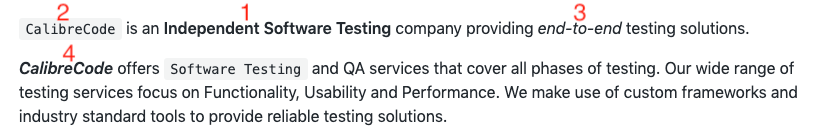
Example format:

1. To make **Bold** use **\*\*<Text>\*\*** at the starting and end of the word. Like this **\*\*Text needs to be bold\*\***.
2. To **highlight** use **`<Text>`** at the starting and end of the word. Like this **`Text needs to be highlighted`**.
3. To make **Italic** use **\_<Text>\_** at the starting and end of the word. Like this **\_Text is Italic\_.** ( This only used to make text Italic without Bold )
4. To make **Bold Italic** use **\*\*\_<Text>\_\*\*** at the starting and end of the word. Like this **\*\*\_Text is Bold Italic\_\*\***.

` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `



And this how it looks in the application.



1. How to mention the link to internal files,

` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `

Example format: **[ Text ](file name with extension “.md”)**

` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `



And this how it looks in the application. Clicking on the word navigates you to the file mentioned.



1. How to represent external links,

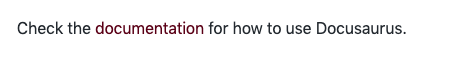
` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `

Example format: **[Text](Link)**

` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `



And this how it looks in the application.



1. How to mention any code,

` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `

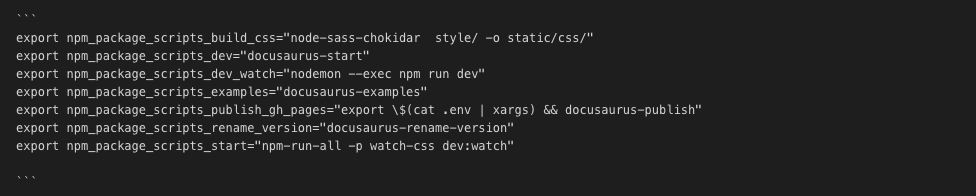
Example format:

**```**

**Code**

**```**

` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `



And this how it looks in the application. This format gives you a **scroll** option.



1. How to mention steps with numbering,

` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `

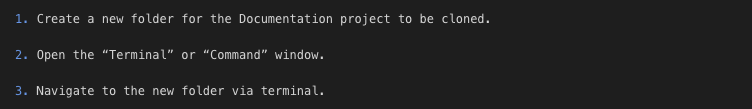
Example format:

**1. Step1**

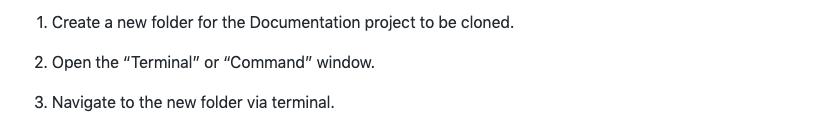
**2. Step2**

**3. Step3**

` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `



And this how it looks in the application.



1. How to create a table

` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `

Example format:

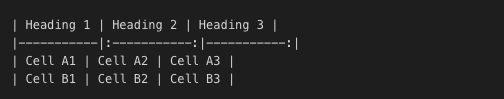
**| Heading 1 | Heading 2 | Heading 3 |**

**|-----------|:-----------:|-----------:|**

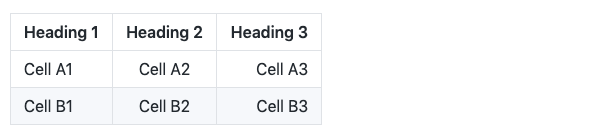
**| Cell A1 | Cell A2 | Cell A3 |**

**| Cell B1 | Cell B2 | Cell B3 |**

` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `



And this how it looks in the application.



## How the right side menu is displayed automatically on the parent page.

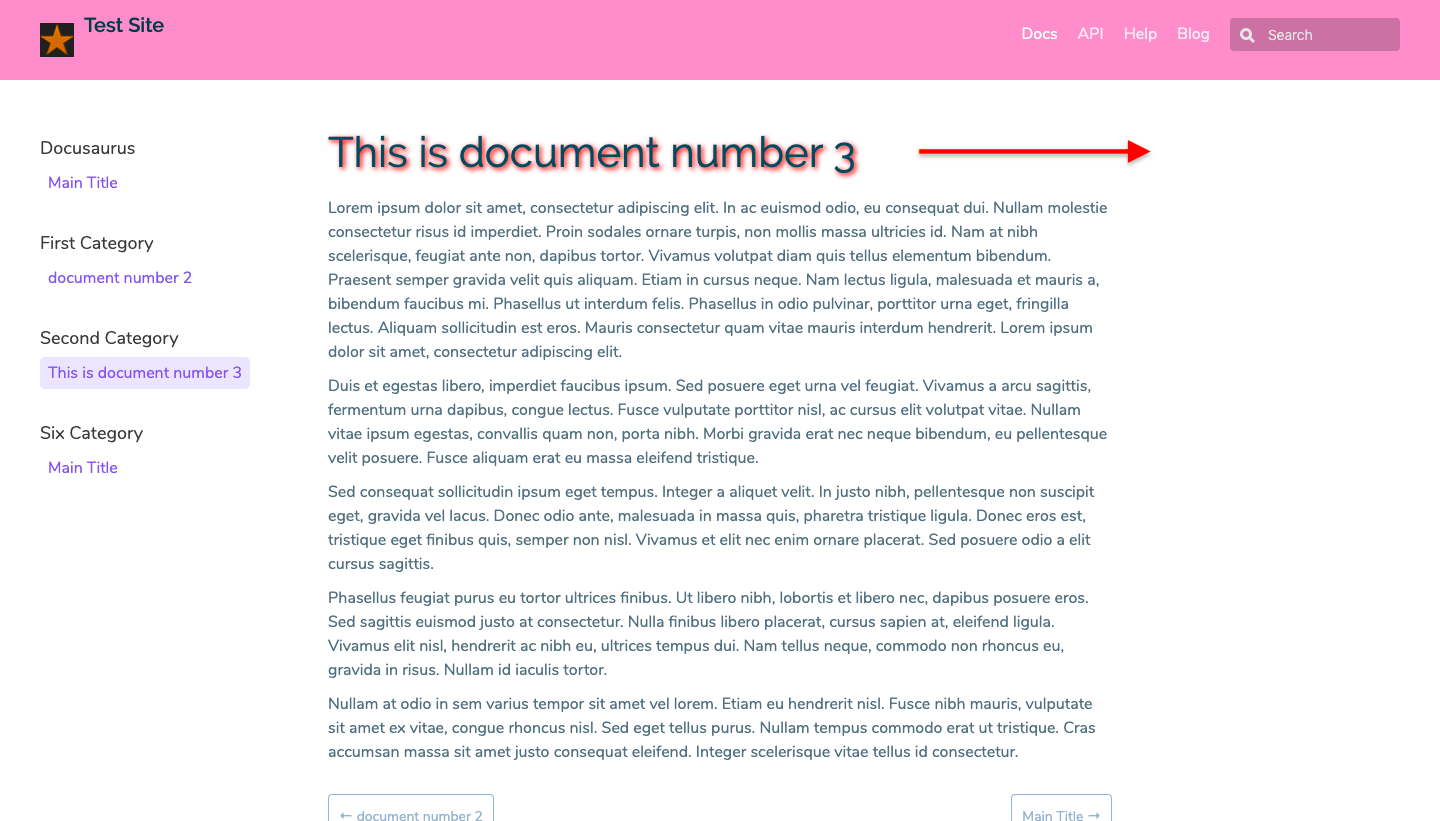
When a heading or subheading is added to the page, it automatically gets reflected in the right side menu to display the heading and sub-heading on the page so that the user can access the page in an easier format.

` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `

And this how it looks in the application looks if there are heading and sub-heading added in a page.



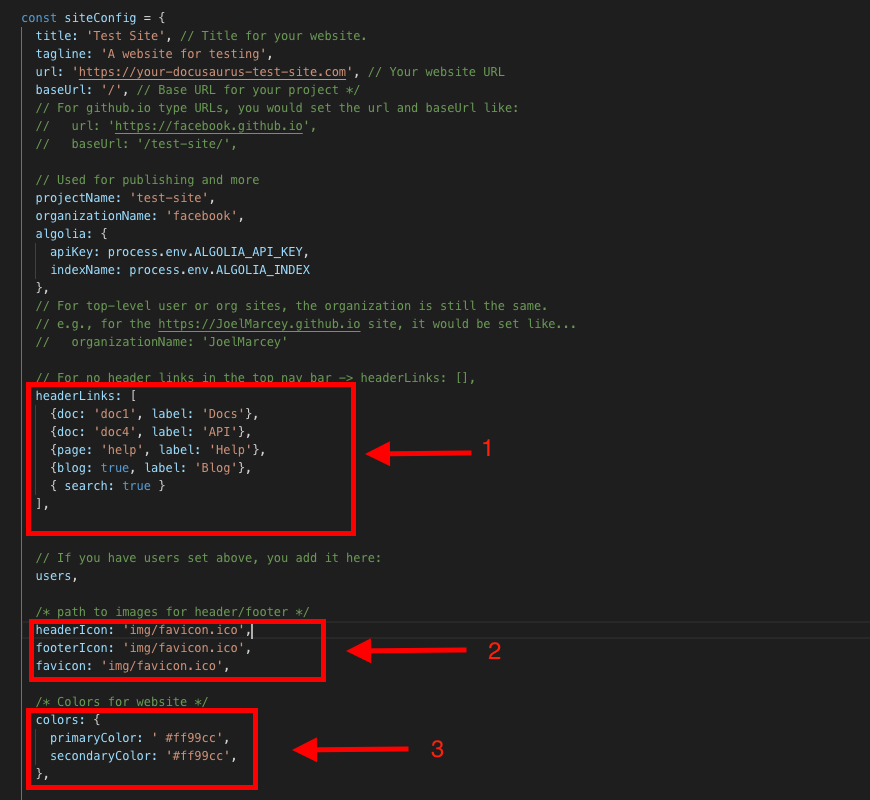
And this how it looks in the application looks if there are not heading and sub-heading present in the documentation.



## 

## How to Modify the header bar

To modify the header bar in the documentation, go to the “**website/siteConfig.js**” file.



1. HeaderLink indicates all the Links that can be present in the Header of the Documentation site.
2. HeaderIcon, FooterIcon indicates the icon present on the website. You can specify the required location will change the Icon of the website.
3. Colors indicate the Primary and Secondary color given to the header bar.

## How to Modify the Footer

To modify the footer bar in the documentation, go to the “**website/core/Footer.js**” file.

Add the required links in the footer for your documentation site. [Click Here](https://docs.google.com/document/d/1q7sKXxZyWjfZk75apHQai1xQqRoeTmN8Hgu698BOlKo/edit?folder=0ALOyMVNtkvb1Uk9PVA) for Sample Footer Code.

` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `

Example format:

**const React = require('react')**

**class Footer extends React.Component {**

**docUrl(doc, language) {**

**const baseUrl = this.props.config.baseUrl**

**return baseUrl + 'docs/' + (language ? language + '/' : '') + doc**

**}**

**pageUrl(doc, language) {**

**const baseUrl = this.props.config.baseUrl**

**return baseUrl + (language ? language + '/' : '') + doc**

**}**

**imgUrl(img) {**

**const baseUrl = this.props.config.baseUrl**

**return `${baseUrl}img/${img}`**

**}**

**render() {**

**return (**

**……………(Your footer code)**

**)**

**}**

**}**

**module.exports = Footer**

## 

## How to Change the font color.

To modify the font color in the documentation, go to the “**website/static/css/custom.css**” file.

Modify the required font style as required to your website.

` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `

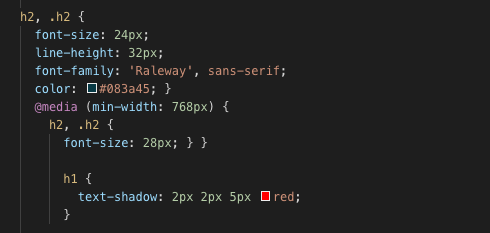
Example format:

**h1 {**

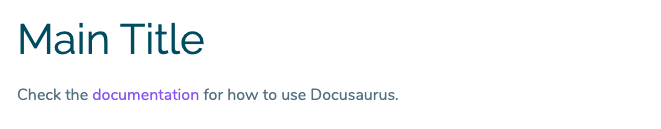
**text-shadow: 2px 2px 5px red;**

**}**

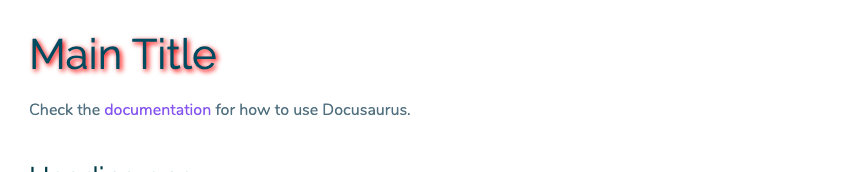
` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `



And this how it looks in the application before the change.



And this how it looks in the application after the change.



## How to add Search bar

To add search bar in your documentation add the below code in “**website/siteConfig.js**”

` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `

**algolia: {**

**apiKey: process.env.ALGOLIA\_API\_KEY,**

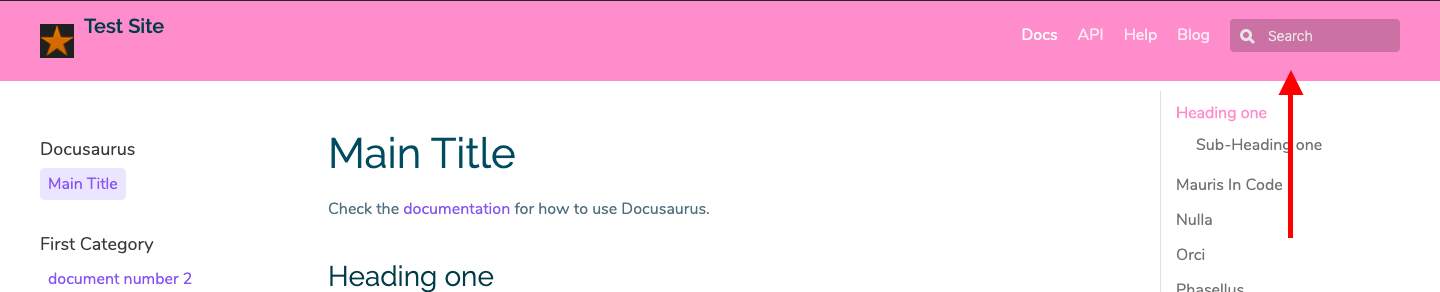
**indexName: process.env.ALGOLIA\_INDEX**

**},**

` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `



And this how it looks in the application.



## 

## How to add a video

Video can be added to documentation in two ways.

` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `

1. Adding a video without control (Autoplay), Here is the code to add the video.

<video autoplay loop muted width="100%">

<source src="/docs/assets/angular.mp4"

type="video/mp4">

Sorry, your browser doesn't support embedded videos.

</video>

1. Adding a video without control (Pause and Play), Here is the code to add the video.

<video autoplay loop muted controls width="100%">

<source src="/docs/assets/angular.mp4"

type="video/mp4">

Sorry, your browser doesn't support embedded videos.

</video>

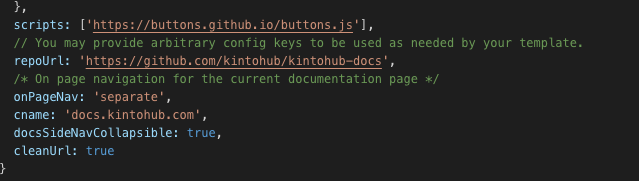
Note > If the video format is not in mp4 then the video might not support sometimes.

## 

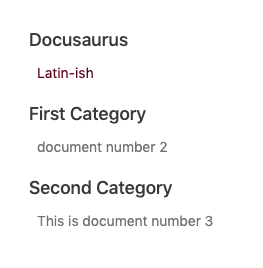
## Site Configuration

To make sidebar **expandable** and **collapsible**. Make sure to add the following line in the “site-configuration” file.

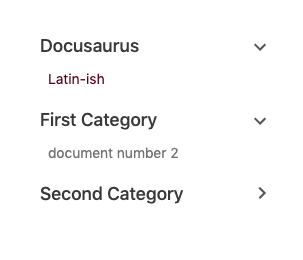
**“ docsSideNavCollapsible: true”**



And this is how it looks in the application before adding **“ docsSideNavCollapsible: true”**

****

And this is how it looks in the application after adding **“ docsSideNavCollapsible: true”**

****

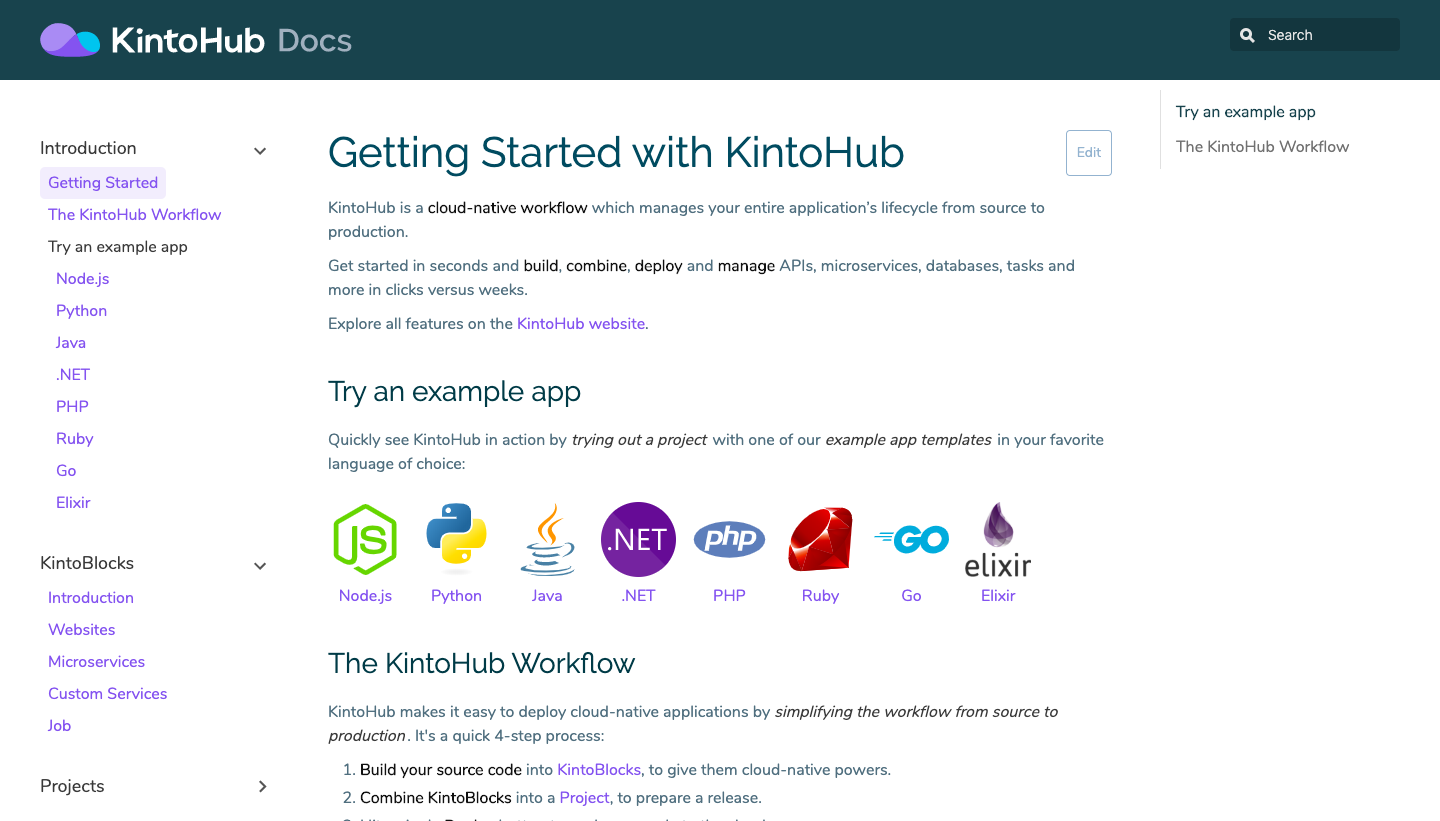
## 

## Sidebar content creation

The below images describe how a sidebar configuration file looks like.

## 

And this is how it looks like in the application.



The below is the skeleton format for the sidebar,

` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `

{

"docs": {

"Section 1": [

“Subsection 1”,

“Subsection 2”,

“Subsection 3”

],

"Section 2": [

“Subsection 1”,

“Subsection 2”,

“Subsection 3”

],

"Section 3": [

“Subsection 1”,

“Subsection 2”,

“Subsection 3”

],

}

}

` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `

Here is the example of sidebar.json file.

{

"docs": {

"Docusaurus": ["doc1"],

"First Category": ["doc2"],

"Second Category": ["doc3"],

"Six Category": ["doc6"]

},

"docs-other": {

"First Category": ["doc4", "doc5"]

}

}

` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `

**docs:** Docs is mentioned here as it consists of all the content files to be displayed in the application.

**Section 1:** This is where you mention the name for the main section.

**Subsection 1:** This is where you mention the subsection. You should mention the name of the file here. Make sure you mention the file name excluding the format “**.md**” as it’s not needed her to mention.

After entering the contents in the sidebar you must build the document using the **“npm run build”** command to see the changes. The contents present in the sidebar.json file will be added to the en.json file.

**Note**: The name of the sidebar should not contain any space or special characters. If the file name contains any one of these then file will not be recognised in the document and will not be updated in the sidebar. If we add slash ( / ) then it will be considered as path.

Bad way of giving file name.

1. SignUp/SignIn.md
2. Signup SignIn.md

How to find issues within the documentation

We can find the issues with the documentation by building the documentation using Build Command.

Run this command **“npm run build”** from the “**website**” folder in the terminal to find the broken parts in the document.

## How to publish the document

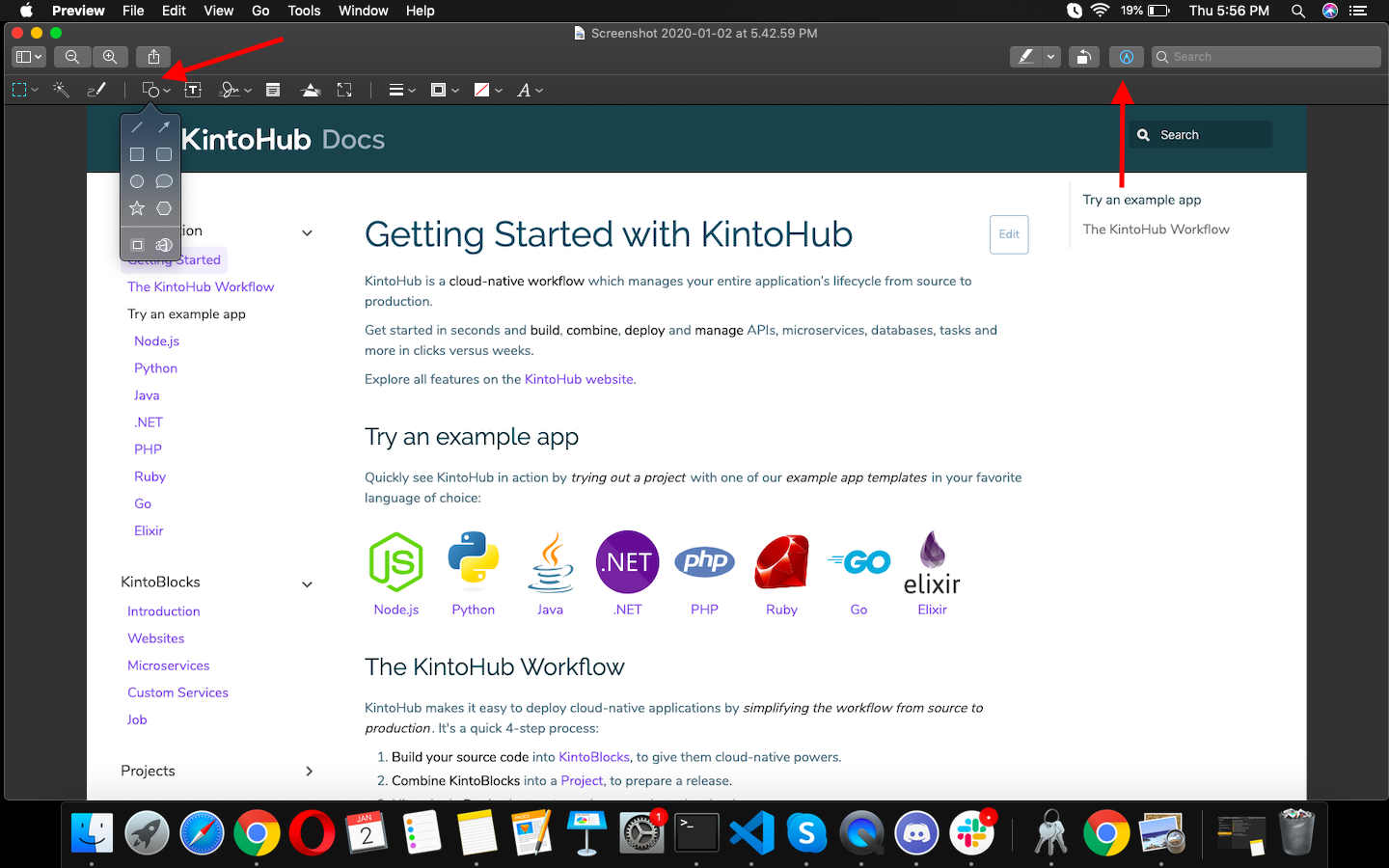
Docusaurus generates a static HTML website that is ready to be served by your favorite web server or online hosting solution. First, we need to build, this will generate a build directory inside the website directory containing the .html files from all of your docs and other pages included in pages.

After the building is successfully generated, run the command **“npm run publish-gh-pages”.** The command needs to be run from the “**website**” folder of the documentation project via terminal.

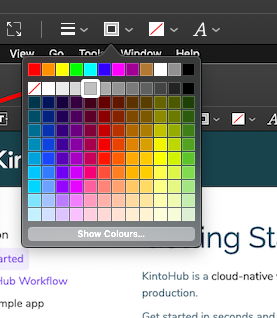
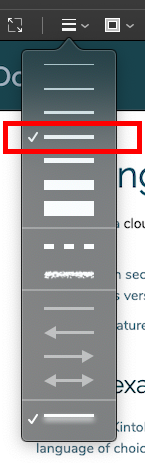
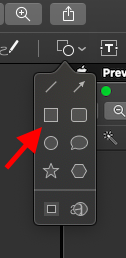
## 

## How to add new Images in Documentation

1. Capture the required page image using “**Command+Shift+3**” in Mac.
2. Open the image.
3. Click on the “**Pen icon**” to edit the image.
4. Click on the Highlighting “**Shapes**” drop-down. ( Shown in the below image )



1. Select “Square” Shape for the border of the image, line length as “4” and color as light grey as highlighted in the below image. ( Adjust the Square shape as it fits the border of the image)



1. Select the “Rounded Square” with the “Red” color to highlight the contents in the image.
2. Rename the image as required and move it to the assets folder inside the Documentation project.

## 

## How to commit files to GitHub.

There are two methods of committing the code to GitHub

1. From Terminal (This method can be used to push the changes made to the content of the file).
2. From Github client (This method is used to push the screenshot or small changes made to the document).

After committing the code make sure to verify the changes by running the document locally.

` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `

To push code from terminal follow the steps cited below:

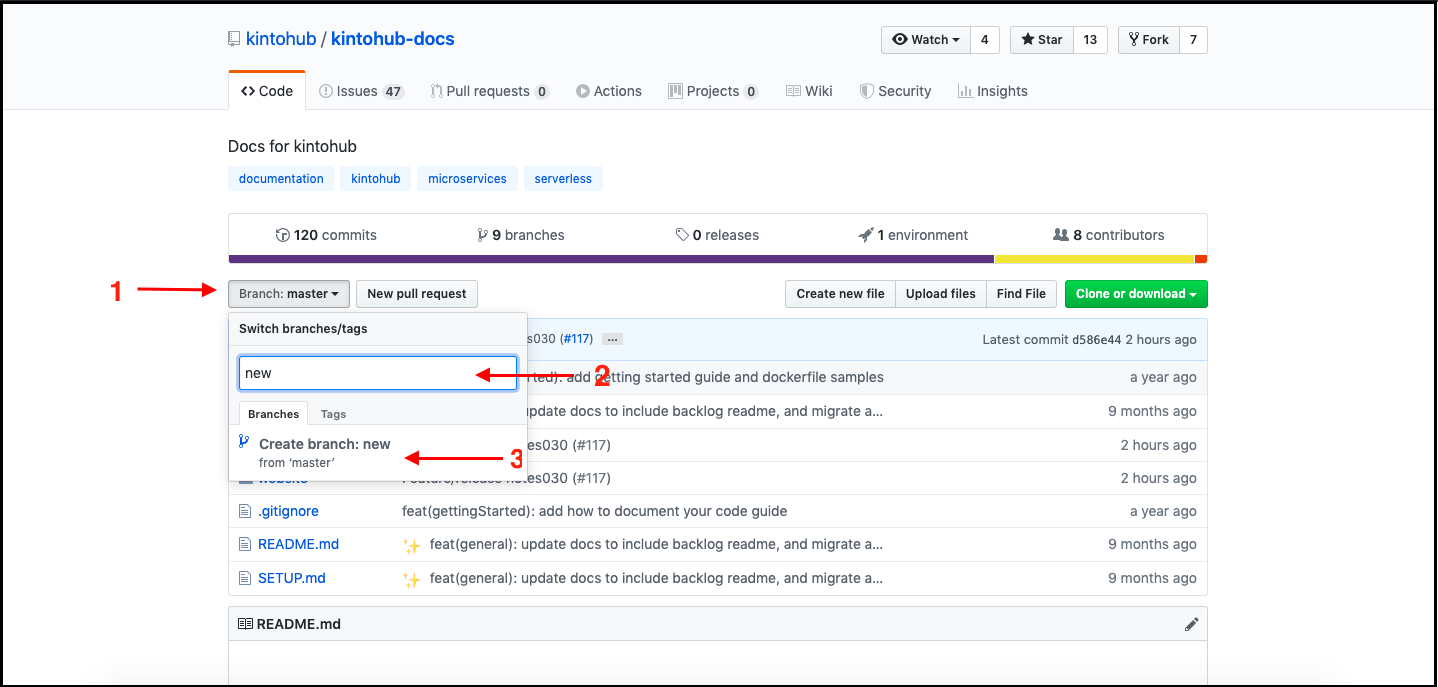
1. Open the “Terminal” and go to the project folder.
2. Run “**git config --list**” ( To check the user account).
3. Run “**npm run dev**” ( To run the document locally).
4. Run “**git status**”
5. Run “git pull” (If any changes are already present in Github and not in your local machine)
6. Run “**git config --global user.name "<username>"**”
7. Run “**git config --global user.email <user email>”**
8. Run “**git add .**”
9. Run “**git commit -m “Changes made in test”**”
10. Run “**git push**”

Now you have successfully pushed the code to Github account.

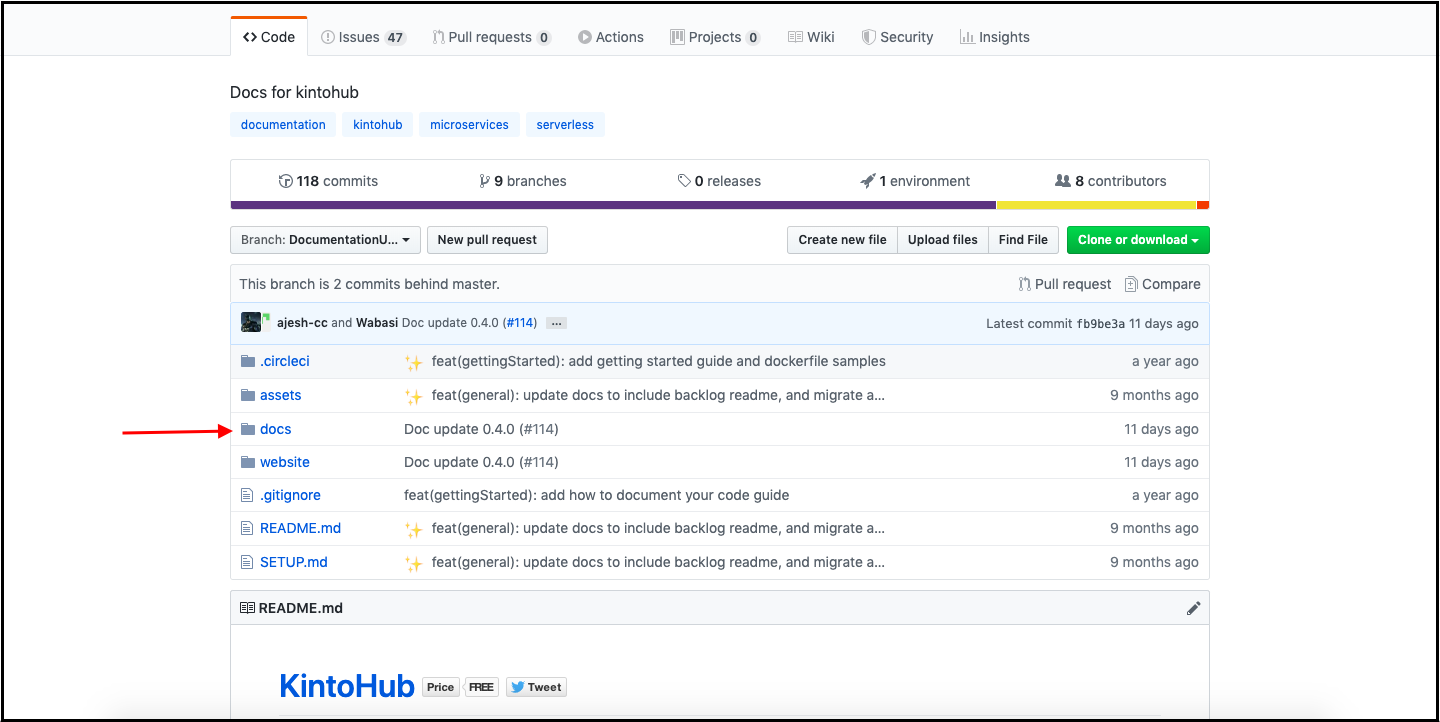
` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` ` `

To push code from Github client follow the steps informed below:

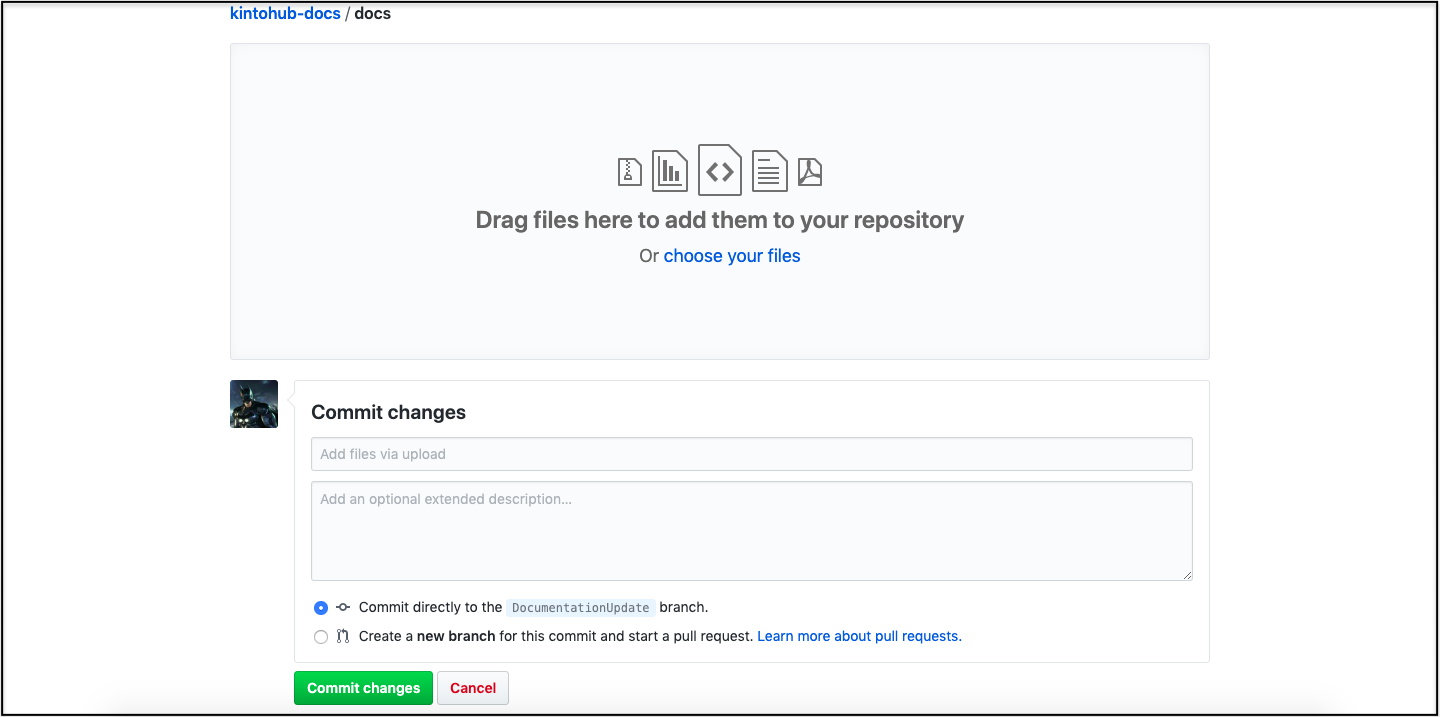
1. Login into the GitHub account. Get the username and password before you login.
2. Select the repository **“<RepoName>”.**
3. Make sure you create a **new branch** from the **master** branch.
4. Click on the **branch dropdown ( 1 )** and enter the new branch name in the **branch name field ( 2 )**.
5. Click on **create** **( 3 )** button.



1. Once a new branch is created you will be navigated to the new branch.
2. Click on the “**docs**” folder and get navigated to the folder items.



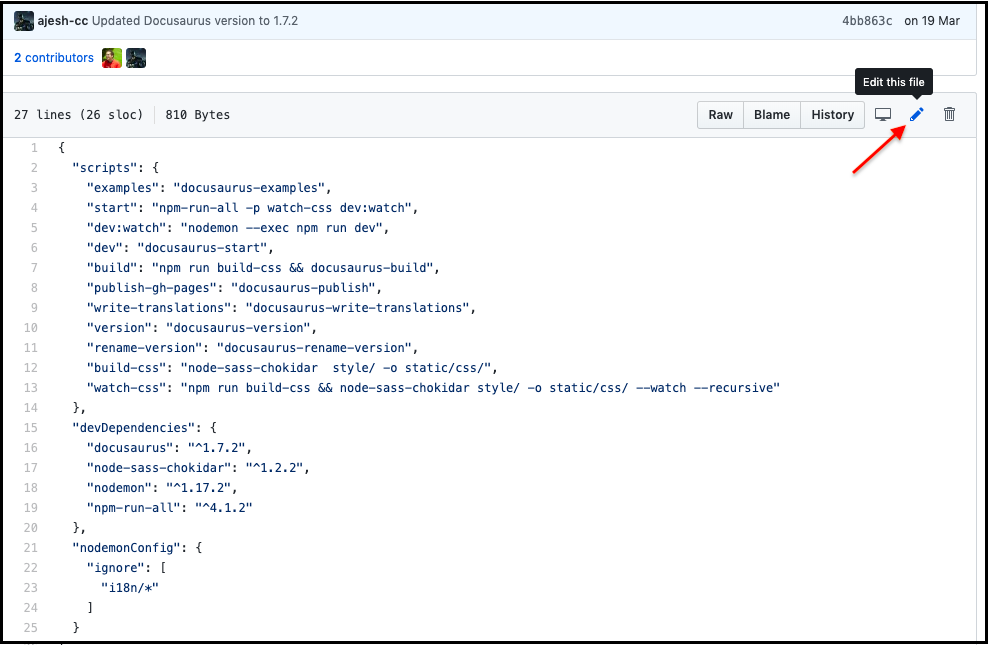
1. Click on the “**upload files**” button.
2. Upload the files needed to be committed and click on “**commit changes**” button to do a commit. ( Add the description and click on “commit changes” button ).



1. Repeat steps **7** to **9** for uploading images to the **assets** folder.

To update “package.json”, “site configuration” and “siderbar.json” file follow the steps mentioned below,

1. Click on the “**website**” folder and get navigated to the folder items.
2. Click on the “**package.json**” file and open the file.
3. Click on the “**edit**” icon.

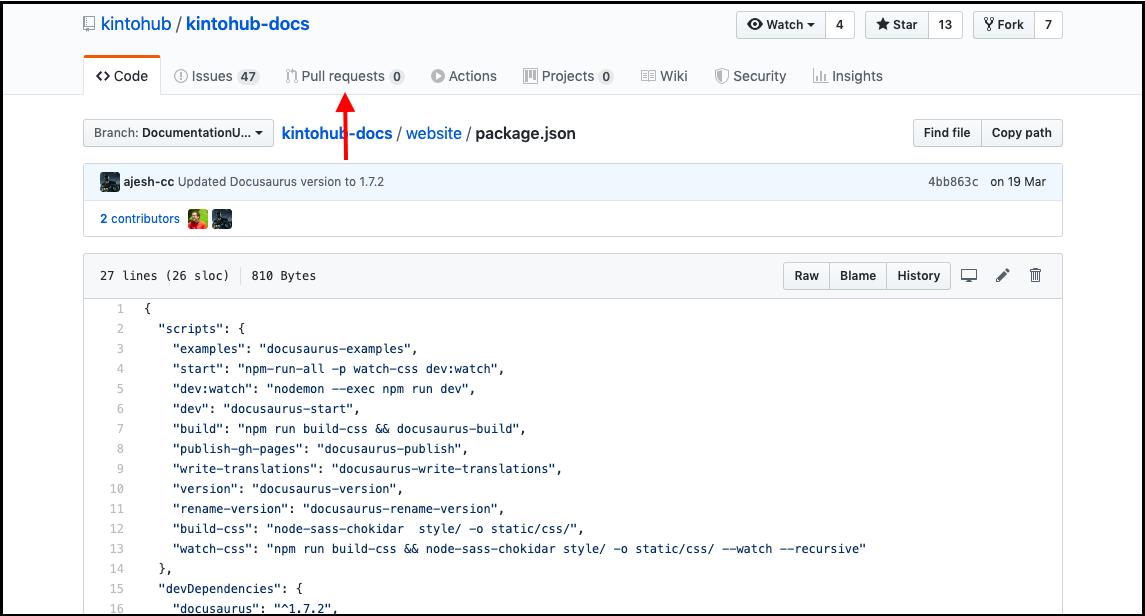


1. Do the changes and click on “**commit changes**” button to do commit.
2. Repeat the above steps to edit “**site configuration**” and “**siderbar.json**” as well.

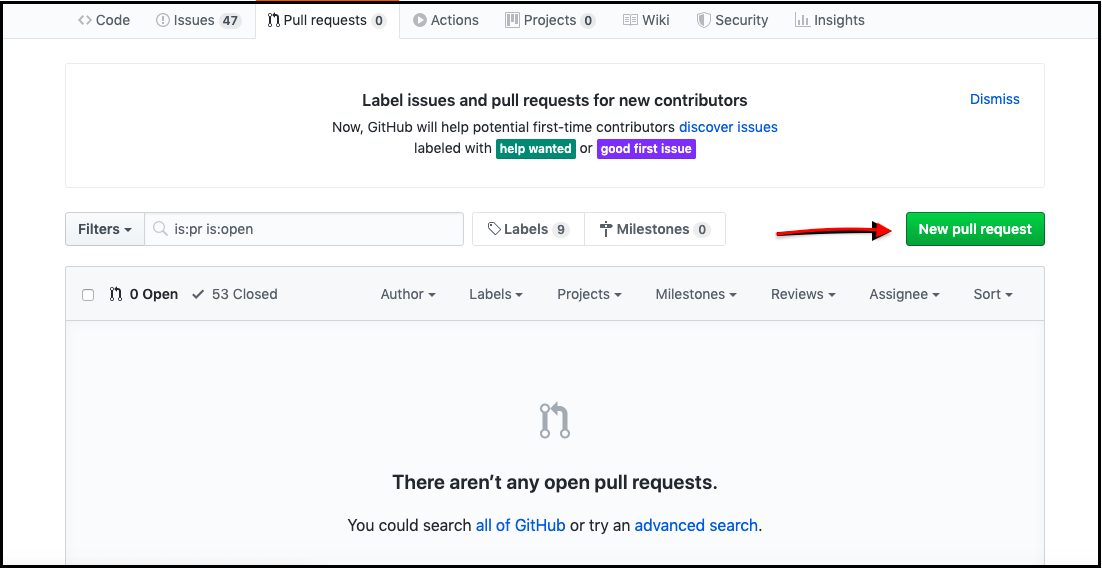
## 

## How to Create Pull Request (PR)

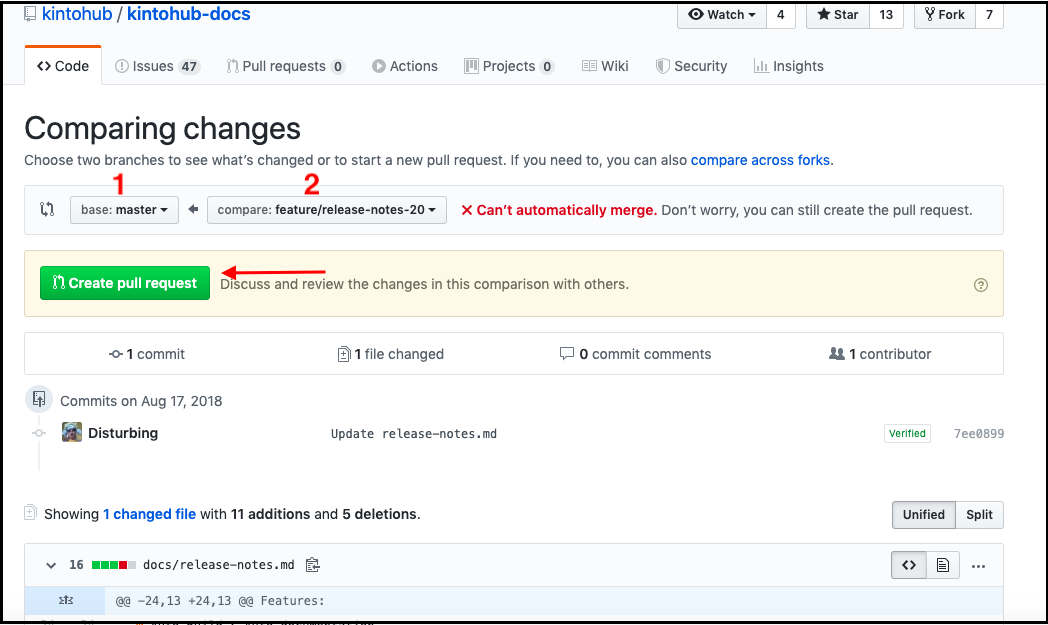
1. Once the commits are done successfully in the working branch.
2. Click on the “**pull request**” link.



1. Click on “**New Pull Request**” button.



1. Select the “**main branch**” in the **dropdown ( 1 )** and the working branch in the **drop-down ( 2 )**.



1. Make sure you select the “**assignee”** and “**reviewer”** when a PR is created.

## 

## How to resolve Github Conflict

1. First, check why the conflict occurred.
2. If the conflict has occurred then check from which file the conflict has occurred.
3. If the conflict has occurred from **(docs/website/i18n/en.json file)** then check for the changes that are done in the master branch and your branch for en.json file.
4. If the conflict has occurred from **(docs/website/sidebars.json file)** then check for the changes that are done in the master branch and your branch for sidebar.json file.
5. You can replace the .json file from your local branch and then click on "**resolve conflict**"
6. Finally, you will see the "Commit" button, click on that
7. Then inform the reviewer by adding a comment.

## Documentation checklist

* Create a new Branch from the master.
* Write the contents of Documentation in “Google Docs”.
* Check for any mistakes using the “Grammarly Tool” in Google Doc.
* Get it Reviewed by “Sir or Ma’am”.
* Make any changes if necessary.
* Check for any errors using the “**npm run build**” command.
* Open document locally using “**npm run dev**” command
* Push the changes when there are no errors.
* Send a proper message when creating PR.
* Add the “**assignee”** and “**reviewer”**
* Send the proper message informing clients.

**Note:**

1. No direct commits from the GitHub site.
2. For PR, chose the reviewer and assign yourself.
3. For PR, make sure you are logged into the proper account.

# Quick Links

[Use of Documentation](#_35nkun2)

[Software Tools & Frameworks used for Documentation](#_1ksv4uv)

[Environment Setup](#_3fo5tlvw2r80)

[Build and Run the document locally](#_ogxwmgcscnli)

[Project structure](#_gjpgie43ovrp)

[Creating files for documentation](#_3j2qqm3)

[Styling formats](#_afihd7cjiji8)

[How the right side menu is displayed automatically](#_ttoli2xzk3zo)

[How to modify the header bar](#_snl7158pjc72)

[How to modify the footer](#_gzfcy7ufsqug)

[How to change the font color](#_cqa03sqde7i2)

[How to add Searchbar](#_l26jov7zby80)

[How to add a Video](#_zgtywm1270jr)

[Site Configuration](#_vv1r2jx6c22k)

[Sidebar content creation](#_9xn4h2wovuxp)

[How to find issues within the documentation](#_11xgz7dqjvg)

[How to publish the document](#_qu3l2kd2iyz)

[How to add new Images in Documentation](#_9l3jmud3k4g7)

[How to commit files to GitHub](#_7cwkaql0d6ri)

[How to Create Pull Request (PR)](#_slb9u05lbpqz)

[How to resolve Github conflict](#_xjnlo81ciwxo)

[Documentation Checklist](#_xqfkt5zganxi)