



tawk.to Frontend Developer Test

We feel that the best place to really show us your skills is somewhere you feel comfortable. This test should not take you some crazy amount of time to complete.

This take home test is used to determine your understanding of Vue js and components.

What is the test?

You will be building a simple knowledge base site in Vue js. You will need to create 2 views/pages; the home and category view.

Below is the design for the home view.

Knowledge base doesn't have to be boring

Everything you need to Manage your Messaging



Getting Started

5 articles

Last update 2 days ago



Chat Widget Customization

9 articles

Last update 2 days ago



Using the Dashboard

12 articles

Last update 2 days ago



Advanced Features

5 articles

Last update 2 days ago



Integrations

14 articles

Last update 2 days ago



E-commerce Integrations

10 articles

Last update 2 days ago

In the home view, there needs to be a search input field and a list of categories.

You will need to query for the list of categories and display them as shown above. Only enabled categories should appear. The categories should be ordered based on the order number.

When a category is clicked, get the list of articles for that category view and show the list as below. Only show published articles.

Knowledge base doesn't have to be boring

Everything you need to Manage your Messaging



All categories > Getting started



5

Getting started

Updated 2 weeks ago



De quibus cupio scire quid sentias.
Quid, de quo nulla dissensio est illum
mallem lebares sentias nulla dissensio.



Creating a New Account

Updated Jun, 15 2019



Adding a Widget to Your Website

Updated Jun, 15 2019



How to Add Additional Websites to the Dashboard

Updated Jun, 15 2019



How to Invite and Manage Agents

Updated Jun, 15 2019



The Difference Between Sites, Pages & Profiles

Updated Jun, 15 2019



Other categories



Chat Widget Customization

9 articles

Last update 2 days ago



Using the Dashboard

12 articles

Last update 2 days ago



Integrations

14 articles

Last update 2 days ago

You can find the design here ->

<https://www.figma.com/file/AeDrzKULFKrTRzrH5WkMgU/Frontend-Test-1>. We use figma for our design needs. Fontawesome is used for icons and it is already included in the source. If there are icons that do not exist, it is fine.

Getting Started

Clone the repository from <https://bitbucket.org/tawkto/frontend-test/src/master/>

You will have to run `npm install` inside the directory to install required modules for vue and webpack.

We have set up a webpack dev server for you. The entry point is 'src/app.js' and 'public/index.html' is the html file it'll serve.

The dev server will run on localhost:9000 and just run npm start to run it!

You can use axios to request for the data and the list of endpoints are :

1. Request list of categories -> <http://localhost:9000/api/categories>
2. Request articles for a category - > <http://localhost:9000/api/category/<category-id>>
3. Request info of an author -> <http://localhost:9000/api/author/<author-id>>

The data returned by the endpoints are just mockups so they may be not correct. You do not have to worry about the data. Just use whatever that is returned.

For this test you can use Javascript or Typescript. If you want to do unit testing, you can go ahead and do it. Feel free to use any modules that you think will be necessary.

Browser Requirement

The solution should be tested in all browsers. For IE the minimum requirement is **IE 10** and for Safari the minimum version is **Safari 8**.

Submitting

1. Make sure everything is tested and is running smoothly.
2. Push up to your repo
3. Reply back to the initial email the link to your repository.

Data Structure

1. Category

id - The unique id of a category [string]

title - Title/Name of the category [string]

description - Description/Explanation of the category [string]

createdOn - The date & time the category was created [string]

updatedOn - The date & time the category was updated [string]
enabled - The visibility status of the category [boolean].
order - The position of the category in the list [integer].
icon - Icon that represents the category[string]
totalArticle - Total articles written for that category

2. Article

id - The unique id of an article [string]
title - Title of the article [string]
createdOn - The date & time the article was created [string]
updatedOn - The date & time the article was updated [string]
authorId - The id of the author [string]
content - The content of the article [string]
icon - Icon that represents the article [string]
status - Status of the article, there can be possible values; draft, published, archived [string]

3. Author

id - The unique id of an author [string]
name - The name of the author [string]