shwetalsoni_gsod'21 proposal_wechaty.md

GSOD'21 Proposal for Wechaty

Technical Writer

Shwetal Soni

Contact Email

sonishwetal704@gmail.com

Professional Information

- **Writing Samples**
  - Tutorial for writing Smart Contracts on Tezos
  - Getting over Imposter Syndrome
- **My Resume**
  shwetalsoni_resume
- **My Portfolio**
  shwetalsoni.github.io
- **Overview of my recent work**
  Recently I had written a beginner-friendly tutorial for writing smart contracts on SmartPy which takes the user from introducing to blockchain to finally deploying a smart contract. To make it easier to understand I have attached screenshots guiding them throughout the whole process ensuring they learn by doing. I got it reviewed by some people and they really found it useful and easy to follow.

Project Title

Create easy to learn tutorials for beginner users of Wechaty

Abstract

Tutorials are lessons that take the reader by the hand through a series of steps to complete running Wechaty code. They are what our project needs in order to show a beginner that they can achieve something with it. They are wholly learning-oriented, and specifically, they are oriented towards learning 'how' rather than learning 'what'. Tutorials need to be useful for the beginner, easy to follow, meaningful, extremely robust, and kept up-to-date. The Tutorial project should:

- Allow the user to learn by doing
- Get the user started
- Make sure that the tutorial works all the time
- Ensure the user sees results immediately
- Make the tutorial repeatable
- Focus on concrete steps, not abstract concepts
- Provide the minimum necessary explanation
- Focus only on the steps the user needs to take
- Work with the contributor team to update the documentation on the Wechaty site.

Current Situation

Users come to Wechaty website seeking how to use Wechaty to build a chatbot application for solving their problems. Unfortunately, the learning curve is not friendly to our users, they always struggle with setting up with their development environment, can not get started with the simplest example source code, and find it hard to know what they can do with our API. This problem leads to us losing lots of new users, an increased number of questions asked in the community, and lowered user satisfaction.

Analysis

Being a new user to wechaty I encountered a lot of problems while running Ding Dong Bot. For deep analysis, I asked some of my friends to go through the documentation and try running the bot. Almost everyone faced a lot of issues while running the bot. This clearly depicts that the Tutorial is not properly structured.

Following are the problems I have found out with the tutorials:

- Quick Start is a place where most of the beginners will head over to and should be mentioned at the top of tutorials unlike now it is inside a `Getting Started` dropdown.
- Don't mention something a learner doesn't need to know in order to complete the tutorial. Inside Quick Start under the first heading `In Next 10 min,` I found out that the points written there about getting QR Code for wechaty and QR Code for WhatsApp are useless and even misleading. It messed me up when clicking on `WeChat` QR Code it didn't open and later found out that it has nothing to do with the tutorial.
- Tutorials overview didn't have necessary info on what are the ways in which users can get started. In many good documentations, I have found out that they provide the main link on the overview page so that users can head over from there directly.
- Tutorials should always be working but Usage with Heroku is not working.
- Video Tutorials are like icing on the cake and should be provided in a language that is widely spoken other than Chinese. I even found a comment on youtube requesting for English video.
- There is a link Getting started without leaving your browser after `Let's run your first Ding Dong Bot` given in the read more section by `Huan Li`. It is very well written and should be mentioned separately in the sidebar under the Gitpod option.
- Currently Tutorials only teach the user to set up wechaty but a tutorial should also give the user an experience of working with the features of wechaty.

Based on the above analysis I propose the following structure:

```
Getting Started with Wechaty
|- Overview
|  -> Beginner Level
|      -> Running on Gitpod
|      -> Running on Google Cloud Shell
|      -> Running Locally
|      -> Creating your first bot from scratch
|  -> Intermediate and Advanced Level
|      -> Usage with Docker
|      -> Using with Heroku
|      -> Using Plugin with wechaty
|      -> Using Redux  with wechaty
|      -> Using Vorpal with wechaty
|- Quick Start
|  |- Running on Gitpod
|  |- Running on Google Cloud Shell
|  |_ Running Locally
|- Tutorial Series(Creating your first bot from scratch)
|  |- From Scratch
|  |  |- Overview
|  |  |_ Installation
|  |- Tutorial
|  |  |- Create a bot
```

```
 |  |  |- Dealing with messages
 |  |  |- Processing files
 |  |  |_ Writing tests
 |- Video Tutorials
 |  -> Chinese Tutorials
 |  -> Other than Chinese Tutorials
 |_ Advanced Topics
 |  |  |- Usage with Docker
 |  |  |- Using with Heroku
 |  |  |- Using Plugin with wechaty
 |  |  |- Using Redux  with wechaty
 |  |  |_ Using Vorpal with wechaty
```

- The above structure is well organized from beginner to advanced level.
- **The overview section** should give a complete overview of the topics. If a person is a beginner he can either simply run the app using any of the following options `Running on Gitpod`, `Running on Google Cloud Shell`, `Running locally` by clicking on the given links here.
  - Also the link for `Tutorial Series (Creating your first bot from scratch)` will also be given in case a user want to follow up with steps creating their own chatbot.
  - If a user has prior knowledge about wechaty, they can go to the advanced topics by clicking on the links given for respective topics `Usage with Docker`, `Using with Heroku`, etc.
- **Quick Start** should be divided into sections. If a user wants to run the app using Gitpod, there should be a separate page for that. Similarly for the remaining two options also. This will enable proper step-wise step tutorial for every option unlike now `Running with Gitpod` and `Running with Google Cloud Shell` are both on the same page messing it up and leading to skipping of some important points.
- **Tutorial Series(Creating your first bot from scratch)** is the most important part of this tutorial as this will allow the user to create their own chatbot from scratch and then playing around with the features provided by wechaty. It should be well equiped with code snippets with proper explanation and screenshots of how the result will look after running a given code/command. This will make sure that they learn by doing and will definitely gather interest of more users.
- **Video Tutorial** plays a great role in providing a better understanding and should be available in the language which is widely spoken.
- **Advanced Topics** At the end, if a user has gone through the easy concepts they can dive into intermediate and advanced concepts.

Why is my proposed tutorial better?

QuickStart will get the user started. The Tutorial Series will allow the user to learn by doing. Users can see immediate results in the form of output screenshots attached with every step. The whole structure focuses only on the steps the user needs to take with the minimum necessary explanation.

Why am I the right person for this project?

From last year I have building many personal projects and also writing Readme for that. Recently I learned to write Smart Contracts on Tezos blockchain and it had very few resources available to get the user started and hence I struggled a lot. This motivated me to write a medium article Tutorial for writing Smart Contracts on Tezos. So, I know what people expect from a tutorial. Hence, combining all my experience and analysis of this project I can assure a great tutorial.

Proposed Timeline

- **2 weeks** -> Information Gathering
- **3 weeks** -> Restructuring the tutorials. Organizing and presenting the content in a better way.
- **5 weeks** -> Creating a tutorial series.
- **2 weeks** -> Improvising and publication.

Proposed Budget

Technical Writer Stipend -> 5000 (Decided by the organization)