**Technical Interview Tasks**

**for Python Backend Developer**

# **Introduction**

The purpose of this task is to get to know a little about how you approach the problem of taking an existing codebase and implementing new features.

After unzipping the repository, take a moment to read the README.md to get the application up and running, and familiarise yourself with what the app is doing. Then please proceed with implementing the tasks outlined in this document.

Please use git to commit your solution to each task before proceeding to the next task.

When you have completed the tasks, please create a zip archive of your updated repository, and send it back to us.

# **Tasks to complete**

## **User registration should require a strong password**

There are currently no restrictions for passwords entered during registration.

Your task is to update the registration functionality so that an error is returned if the user's password does not meet the following requirements:

* 6 to 20-characters long
* At least one numeric character

## **User can add keywords when creating a blog post**

Your task is to allow users to add keywords when creating a blog post, and to store these keywords in the database.

You can assume the user will input keywords as a string, e.g. “keyword 1, keyword 2”.

## **Add an admin user type with full view and delete permissions**

Your task is to add an “admin” user type. When a user registers, they should be able to declare that they are an admin user.

Admin users should have the following permissions:

* An admin user should be able to view all users’ posts
* An admin user should be able to delete any user’s posts

## **Make post creation asynchronous**

Your task is to add a delay between the submission of a new post and it appearing in the system.

* After receiving a post, implement a **background** task which sleeps for 30 seconds, and then creates the post in the database

Hint: you may make use of flask\_executor, asyncio, celery, etc if you wish.