How the bot operates:
- On each run, the bot fetches a batch of articles, extracts key words from the text, and finds other potentially related articles by using the mediawiki search engine to find other articles that contain the same keywords.
- An options.json file is used to determine the bots behavior for articles that have an existing related article sub page, and those that do not have one.
- The bot logs the api calls it makes and uses another log to track all visited pages. The log for visited pages is also used by the bot to pick up from where it left off on its previous run.

Installing Dependencies for the Bot:
- Install the Composer PHP dependency manager
  - [Linux installation instructions](#)
- Libraries used:
  - mediawiki-api-base
  - rake-php-plus
- What the composer.json file should contain:

```
{
  "require": {
    "addwiki/mediawiki-api-base": "~3.0",
    "donatello-za/rake-php-plus": "^1.0"
  }
}
```

- Then you can run the command:

<p align="center" style="color:red">composer update</p>

which will automatically install all dependencies listed in the composer.json file

Bot Directory:
- The bot resides on the server at /home/messana/citizendium_suggestion_bot. The bot is executed by running the bot.php file.
- **citizendium_suggestion_bot**
  - bot.php
  - composer.json
  - composer.lock
  - composer-setup.php
  - logger.php (class file for logging)
  - **Logs**
    - api_log_file
    - page_log_file
  - options.json
  - requests.php (wrapper around media-wiki-api to make api calls easier)
  - **vendor**

Chron commands for running the bot:
- 0 7,12,17 * * * cd /home/messana/citizendium_suggestion_bot && php bot.php
  The bot runs 3 times a day at 7am, 12pm and 5pm.

The Options.json file:
- There are only two fields in this file
  - "append_flag" : true or false
  - "create_new_flag" : true or false
- When append_flag is set to true, related articles will be appended to an existing related articles subpage, otherwise it ignores existing pages.
- When create_new_flag is set to true, articles that do not have an existing related articles subpage will have one created, and then it will be populated with the set of related articles found by the bot.

Logging
- The bot creates two new logs each day, a log for tracking api calls and a log for keeping track of the pages the bot has visited.
- The api log file will be named "api_log_y-m-d", where the current date in y-m-d format is appended to the end of the name. It keeps track of all api calls the bot makes.
- The log file for tracking pages visited is named "page_log_y-m-d", where the current date in y-m-d format is appended to the end of the name. It tracks visited pages, and also maintains the bot's state. At the beginning of each run, the bot reads the last line of the latest page log file to figure out where it left off on its last run.
- All logs are kept for a maximum of 7 days. Log files older than that are deleted by a cron job that runs every sunday.

Chron commands for cleaning up logs:
- 0 6 * * SUN cd /home/messana/citizendium_suggestion_bot/logs && find -mtime +7 -type f -delete
- This command cleans up log files older than a week. It runs at 6am every Sunday.