Didgah Email Process Specification

Prepared by:

Bamdad Pirooz
Fariborz Abad

Summary:

This document outlines the design approach and functionality of the Didgah Email Process, aiming to improve the current email process of the Didgah. It addresses the issues with the current system and proposes a solution for efficient email sending, downloading, and uploading. The specification also covers system maintenance and configuration options to provide flexibility and comprehensive logging.

Issues With Current System:

The email service provider, SendGrid, blocks the sender under certain conditions:

- When many receivers block or report the sender's email address as spam or phishing.
- When a large number of emails are sent from the sender's email address within a short time.
- When the sender's email address sends many emails with similar content, links, or keywords used in phishing emails.

Controlling these variables becomes challenging when dealing with a significant volume of emails. Hence, a configurable system needs to be designed to send a high number of emails periodically, allowing self-configuration to avoid the aforementioned problems with minimal user intervention.

Solution:

The email process consists of two components: a web application and a threaded process. Here's how it works:

- The web application utilizes a calendar to determine the number of emails (N) to be sent on a particular date.
- If N is 0, no emails are sent on that day. Otherwise, N emails are scheduled throughout the day.
- Each hour of the day is marked as either True or False. If True, the web app starts an Email Process thread, passing N/T (number of emails divided by the hours the process invoked) to the thread.
- For example, if N=24 on a given day and the hours 8 and 16 are set to True, two threads are initiated—one at 8:00 AM and one at 4:00 PM—each sending 12 emails.

- The Email Process thread is responsible for sending emails. It runs periodically according to the designed schedule, either within the web application or as a separate batch process on a server.
- The thread can send a maximum of 30 emails per hour, with the number of emails passed from the web application. The emails are sent randomly within each minute of the hour.
- After sending an email, the process updates the database based on the returned result.
- If we utilize Google Cloud, we pay for their system based on our usage. We can develop scripts to create virtual machines (VMs). The script is running inside our web application, installs the necessary email processing software on the server we created on google cloud, sends emails, and then shutdown, and deletes the server after use. This script is triggered by the web application for each thread, allowing us to send emails from a new server with a new IP address each time. The system will be used for one hour during the time we send emails, as specified.
- We intend to send a limited number of emails, ensuring that the content is well-designed and properly formatted to avoid being marked as spam or phishing by the recipients. Assuming that we have determined the optimal number of emails to send as 600 per day, we will distribute this number evenly throughout the 24-hour period. Thus, each server will only send 600/24, which is equivalent to 25 emails per server per day.
- In case a few recipients block our address or report it as phishing, we can promptly remove those users from our database. Furthermore, if we purchase email addresses to send our emails, it would be even more advantageous as we can employ different email addresses. In the event that one email address is blocked, we can switch to another one. This approach ensures that we can send out the maximum number of emails possible, reaching millions of people within Iran.

Download Emails:

The Didgah Email system provides functionality for downloading emails based on specific criteria:

- Users can query the system to retrieve emails sent within a certain period, emails with incorrect addresses, or emails blocked by receivers.
- Users have the option to create filters to refine their search.
- A list of emails matching the filter criteria is displayed.
- By clicking a button, users can initiate the download process.
- The system saves the email records as a CSV file on the backend system.
- Users can download the CSV file to a specified location on their file system.

Upload Emails:

The Didgah Email system allows users to upload emails using a CSV file adhering to the agreed format:

- Users select a CSV file containing the email records they wish to upload.
- If there are parsing failures for certain records during the upload, the system logs these failures and continues to upload the remaining records.
- The successfully parsed email records are saved in the database.

System Maintenance and Configuration:

The Didgah Email system provides configuration options to meet user requirements:

- Users can configure email contents using service provider templates or design their own templates.
- Users have the ability to configure the number of emails sent each day of the year.
- Users can configure the hours in the 24-hour clock when the email process should be invoked.
- User can add the speed factor a number between (–1,1).
    - The speed factor of 0.1 means to add 10% to the total email sent daily.
- Testing Period: We start to test the system with internal emails used by member of Didgah for inspecting the email boxed closely.