FIT5032 Suzhou Studio 6

Chief Examiner: ABM Russel

TOPICS

- Gain a basic understanding of 3rd party emailing tools
- Sending email using SendGrid API

INSTRUCTIONS

- Form a group of 2-3 students

EFOLIO TASK 6.1 (PASS AND CREDIT LEVEL)

DESCRIPTION:

- Obtain API key from SendGrid
- Integrate SendGrid into ASP.NET application for sending emails

WHAT TO SUBMIT:

- A screenshot of your SendGrid console with at least one email delivered

EVALUATION CRITERIA:

- Correct usage of SendGrid API key
- Correct screenshot of the SendGrid console

EFOLIO TASK 6.2 (DISTINCTION AND HIGH DISTINCTION LEVEL)

DESCRIPTION:

- Research and understand the key features of 3rd party emailing tools
- Briefly discuss the advantages and disadvantages of using a 3rd party emailing tool

WHAT TO SUBMIT:

- A document detailing the advantages and disadvantages of using a 3rd party emailing tool (In this case SendGrid)

EVALUATION CRITERIA:

- Correctly identify the advantages of a 3rd party emailing tool
- Correctly identify the disadvantages of a 3rd party emailing tool

STUDIO ACTIVITIES

The following example is based on SendGrid. However, you may use any alternative Email sender APIs.

This tutorial will be structured slightly different in comparison to the weeks before this. In this tutorial, you will be given a completed project. You are expected to understand based on looking at the codes or reverse engineering.

Upon the completion of this tutorial. You should be able to look at your SendGrid console and see its features.

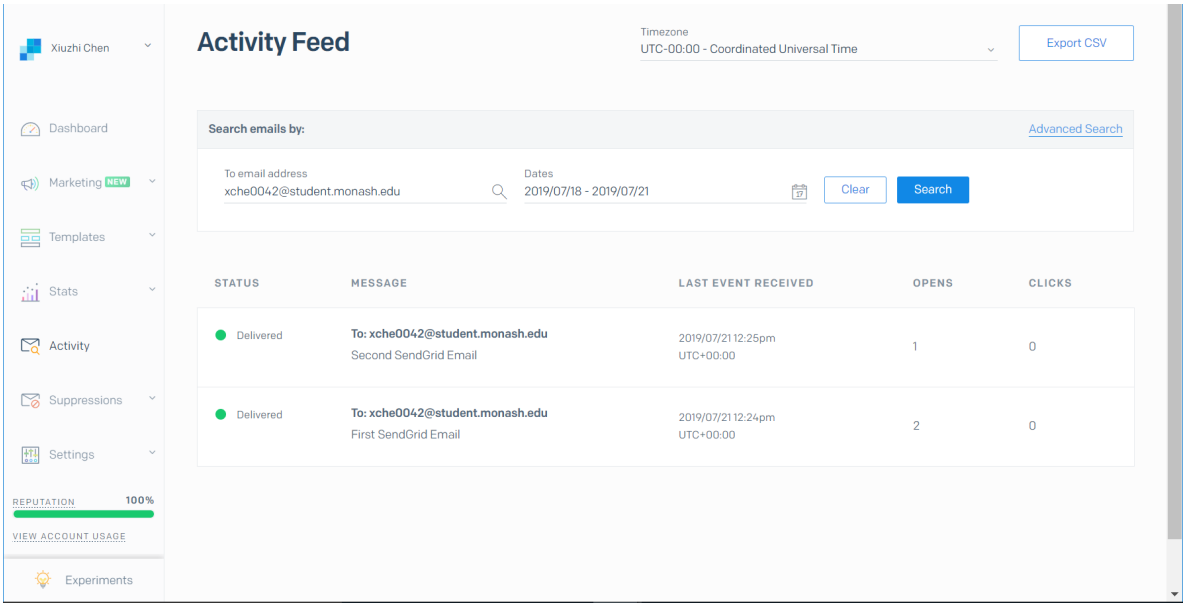

Figure 1: SendGrid Console

You just need to change the API key to yours so that it is working.

Step 1

Obtaining an API Key - You will first need to obtain an API Key from SendGrid (You will need to sign up).

https://app.sendgrid.com/guide/integrate/langs/csharp

Step 2

You will need to change the string value at EmailSender.cs to use your API key.

Step 3

Run the application and select the link "Send Email".

MONASH University

Detailed Explanation (Optional reading if you want to)

1) Here, what I did was quite simple, I first introduced a new namespace called Utils, in this namespace I created a new class called EmailSender. (You can call this namespace "Service" if you want). Essentially, I am "trying" to create reusable and testable code. It is quite common to introduce "Services" and "Utils" or "Data Access Layers" in bigger applications.

2) After that, I created a ViewModel called SendEmailViewModel. Here, I defined, what the input values for it are. In this case, it would be to the **"To"**, **"Subject"** and **"Contents"**. I also added some Attribute so that the email can be validated when MVC creates the scaffolding.

3) Then, I create a View based on the Model. Here, I then change the HTMLHelpers slightly as the TextArea that did not look so nice thanks to automatic markup generated incorrectly. I also introduced a ViewBag return message so that there is a message returned.

Important Things to Understand

1) When sending an email, you do not know if it is successful as this process is handed off to the mail servers (That is why you normally see, most messages that are returned will state that the email "should" arrive). You can also go read up regarding async and sync.

2) To generate more beautiful emails, you can use HTML inside of the emails to make it look nicer.

CONCLUSION

Upon the completion of this tutorial, you would understand how to use a 3rd party API to send your emails. Using SendGrid is a good example as you can do fancy things like checking if the email is opened, etc. In a way, you can create marketing strategies as well (You can read more about it on their website).