MONASH University

FIT5032 - Internet Applications Development

Week 08A - Sending Email (Using SendGrid)

Last updated: 29th July 2018
Author: Jian Liew

Objectives

Estimated Time To Complete - 30 minutes
Upon the completion of this tutorial, you will gain a basic understanding of

- Sending email using SendGrid API

DoubtFire Submission

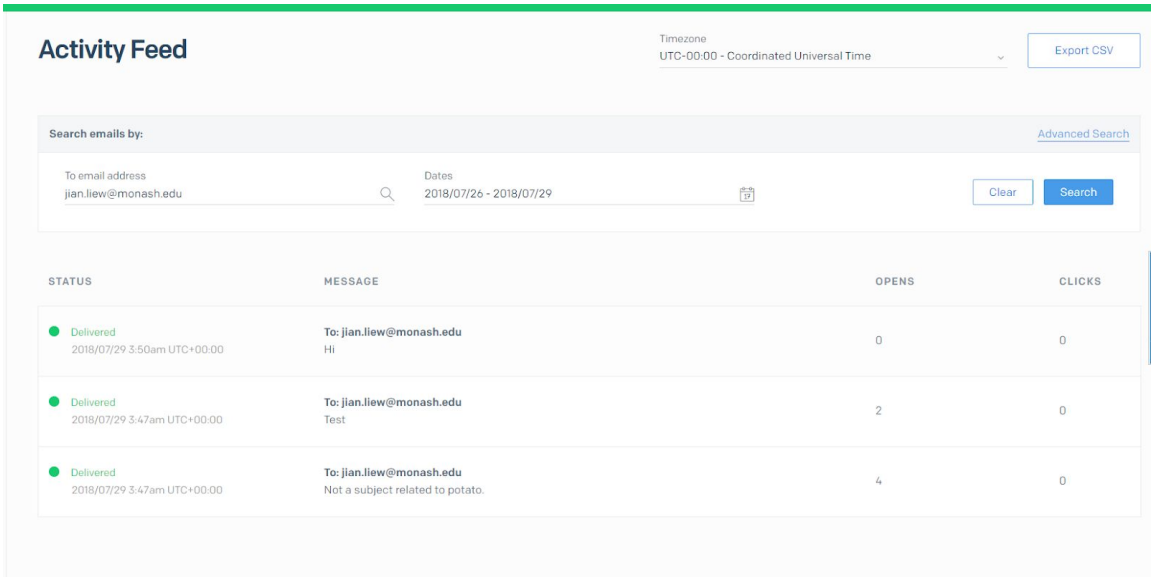
- A screenshot of your SendGrid console.
- A document detailing the advantages and disadvantages of using a 3rd party emailing tool. (In this case SendGrid)

Please do not send me or the Teaching Team emails when testing this. Please send the emails to yourself or your friend.

Tutorial Structure

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. (Sample of required DoubtFire screenshot is below)



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".

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 email, you do not really know if it is successful as this process is handed off to the mail servers. (That is why you normally see, most messages that is returned will state that they email "should" arrive). You can also go read up regarding async and sync.
2) In order to generate more beautiful emails, you can actually use HTML inside of the emails to make it look nicer.

Conclusion

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