MONASH University

Studio 7 and 8

Name and Student Id: Vinaya Datta Kavuluri - 31131611

Self-Evaluation {To be highlighted by Student only}:

Need Help	Work in Progress	Pass	Credit	Distinction	High Distinction

TASK 7.1 (PASS AND CREDIT LEVEL):

Below is the PPT that our group designed during Week – 7. The topic we were given is, MD5, SHA1 (Its weakness and what's the alternative).

Microsoft PowerPoint Presentation

PPT on MD5, SHA1

Looking back at it. We rushed through the presentation. There are a lot of things that need improvements. First of all, the topic itself should need more introduction. We should have dedicated more time/ slides to explain what MD5 and SHA1 are in the first place. Although we've outlined the differences between them, we could have gone into more detail. Also, our presentation feels a bit bland with no colour, images or animations. These kinds of presentations feel uninteresting and boring to the listeners. Also, we did not mention anything about the alternatives part which is part of the topic. The information about SHA-2 and SHA -3 being better alternatives should be included in the presentation along with their respective advantages and disadvantages.

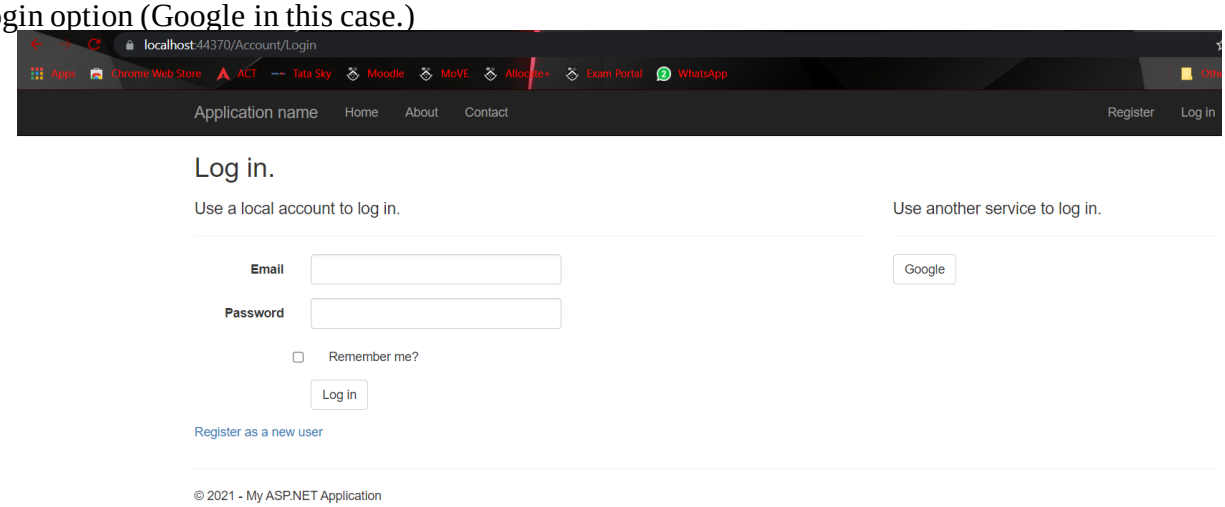
The second version of SHA which is called SHA-2 has many potential variants and one of the most used ones is SHA-256. It is recommended by the National Institute of Standards and Technology (NIST) to be used instead of MD5 or SHA-1. It returns a hash value of 256-bits, or 64 hexadecimal digits. It is also more secure than either MD5 or SHA-1. SHA-3 is still yet to be completely developed as its introduced only in 2015. The algorithm is way different when compared to SHA-2.

References: https://www.freecodecamp.org/news/md5-vs-sha-1-vs-sha-2-which-is-the-most-secure-encryption-hash-and-how-to-check-them/

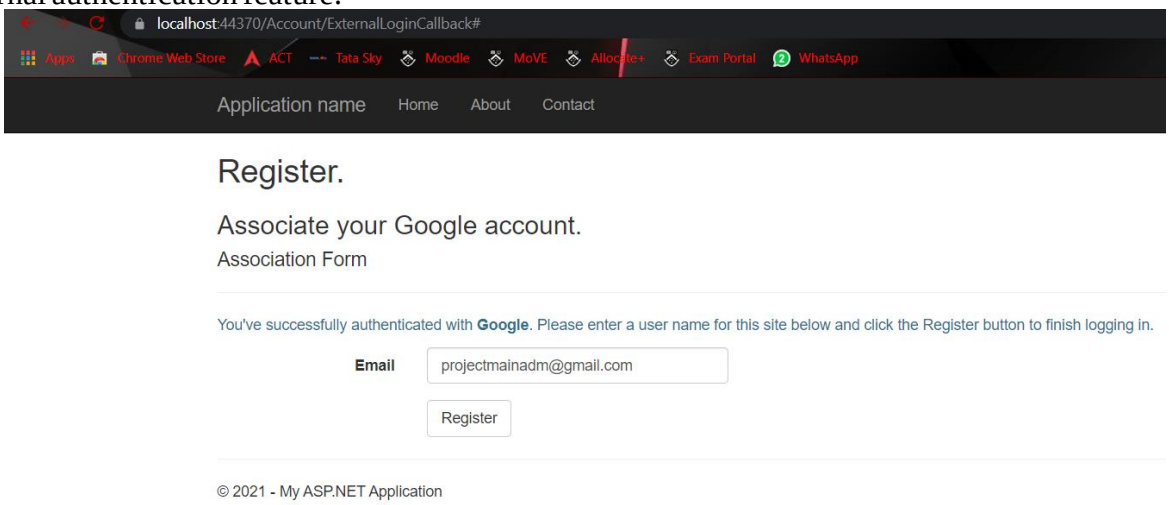
TASK 7.2 (DISTINCTION AND HIGH DISTINCTION LEVEL):

Screenshots:

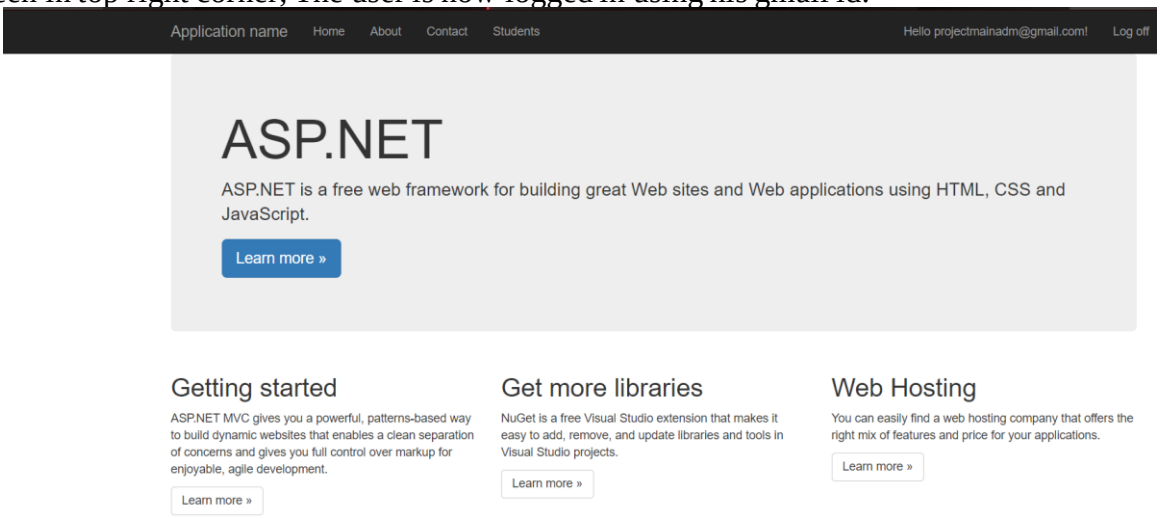
The below screenshot presents the login screen of our web application implementing an external login option (Google in this case.)



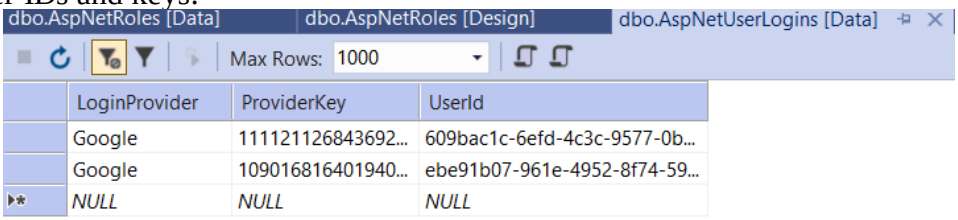
This screenshot shows that a new user is successfully authenticated using their google id via external authentication feature.



As seen in top right corner, The user is now logged in using his gmail id.



This below screenshot is taken from AspNetUserLogins to show the details of two different accounts logged in using the external google authentication. As evident here, these accounts have different User IDs and keys.



	LoginProvider	ProviderKey	UserId
	Google	111121126843692...	609bac1c-6efd-4c3c-9577-0b...
	Google	109016816401940...	ebe91b07-961e-4952-8f74-59...
▶*	NULL	NULL	NULL

This is the screenshot from Startup.Auth.cs file which is implementing external authentication using google ID.

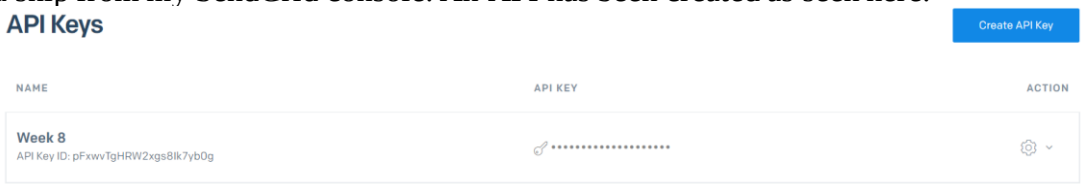
```
Startup.Auth.cs | HomeController.cs | AccountController.cs | Create.cshtml | _Layout.cshtml | StudentsController.cs
FIT5032_MyIdentity | FIT5032_MyIdentity.Startup | ConfigureAuth(IA
50                  //    clientId: "",
51                  //    clientSecret: "");
52
53                  //app.UseTwitterAuthentication(
54                  //   consumerKey: "",
55                  //   consumerSecret: "");
56
57                  //app.UseFacebookAuthentication(
58                  //   appId: "",
59                  //   appSecret: "");
60
61                  app.UseGoogleAuthentication(new GoogleOAuth2AuthenticationOptions()
62                  {
63                      ClientId = "866499648998-evb11tb2m4iuhqcb3d7okfstpq1kl0oq.apps.googleusercontent.com",
64                      ClientSecret = "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"
65                  });
66              }
67          }
68      }
```

Link to code repository:
https://github.com/KVD1302/weekly_activities_5032/tree/main/Week%207

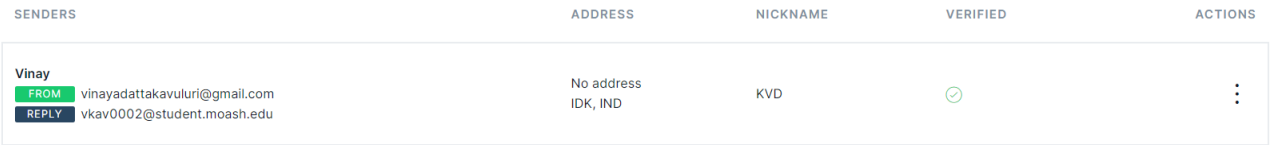
TASK 8.1 (PASS AND CREDIT LEVEL):

Screenshots:

This is a snip from my SendGrid console. An API has been created as seen here.



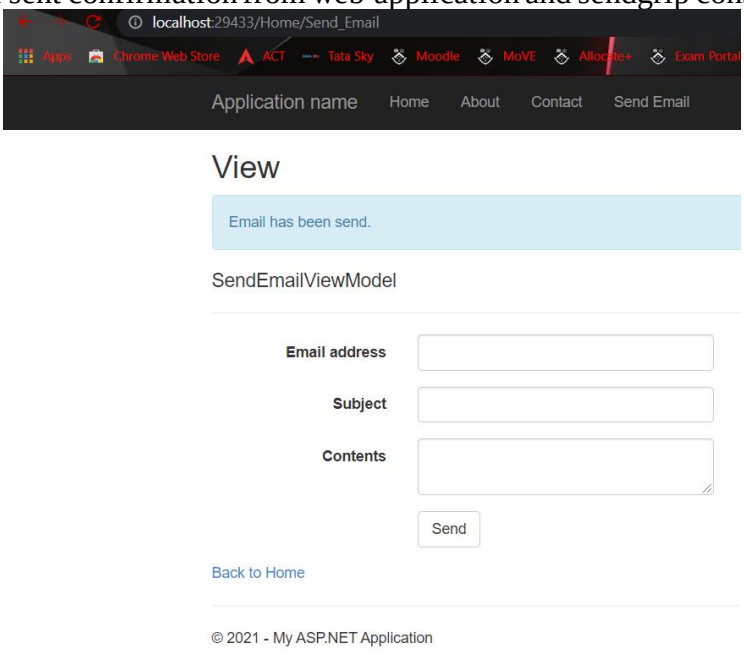
Sender's information are also created and verified.

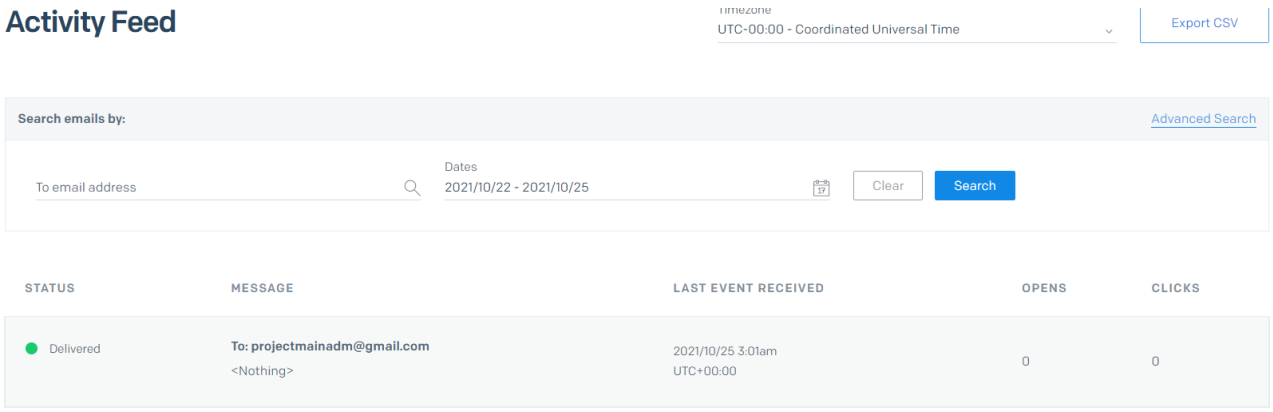


Then, the email is sent through the web application after modifying the EmailSender.cs file with created API key.

```
FIT5032_Week08A | EmailSender.cs | SendEmailViewModel.cs | NuGet - Solution
FIT5032_Week08A | FIT5032_Week08A.Utils.EmailSender | API_KEY
1   using SendGrid;
2   using SendGrid.Helpers.Mail;
3   using System;
4   using System.Collections.Generic;
5   using System.Linq;
6   using System.Threading.Tasks;
7   using System.Web;
8
9   namespace FIT5032_Week08A.Utils
10  {
        2 references
11      public class EmailSender
12      {
13          // Please use your API KEY here.
14          private const String API_KEY = "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";
15
            1 reference
16          public void Send(String toEmail, String subject, String contents)
17          {
18              var client = new SendGridClient(API_KEY);
19              var from = new EmailAddress("noreply@localhost.com", "FIT5032 Example Email User");
20              var to = new EmailAddress(toEmail, "");
```

Screenshots of Email sent confirmation from web application and sendgrip console.





TASK 8.2 (DISTINCTION AND HIGH DISTINCTION LEVEL):

Advantages of using a third-party emailing tool (SendGrid specific):

- We can track whether the recipient opened the mail or not. This is really good for marketing purposes.
- We can select to send the mails using numerous templates in SendGrid.
- We can also filter out any spam content using these services.
- We can also integrate our custom API to the mailing service.
- We can track recipient's unsubscribe patterns.
- They are highly scalable. So that it accommodates increase in the number of users and emails sent through it.
- Supports all kinds of businesses – small, medium and enterprise which makes it affordable and universally usable.
- Supported on multiple platforms – iOS, Android, Web.
- We can also bulk send emails at a rapid rate of over 4,000 emails per second.

- 24 x 7 Support is also provided.
- Automated queue handling and throttle threat detection.
- Partnerships with mailbox providers (including Gmail).
- Troubleshooting to identify delivery issues.

These factors make third-party emailing services like SendGrid highly reliable. Now, coming to the Disadvantages of using a third-party emailing tool (SendGrid specific):

- Although SendGrid offers custom IP addresses, it is only available on their most expensive plan. Therefore, its deliverability can't be trusted due to the usage of shared IPs. This is an issue faced by a lot of other third-party emailing services as well.
- If anyone using their service sends spam emails, all the users will experience deliverability issues.
- It is also a bit difficult to do integrations with SendGrid.
- Also, SendGrid requires the users to know how to do code. So, it is not that user friendly.

References:

1. https://comparecamp.com/sendgrid-review-pricing-pros-cons-features/
2. https://systeme.io/blog/sendgrid-vs-mailchimp#4