Hello everyone , My name is Nazar. And today I want to present to you what I have done to the first demo.

First of all I want to start from my first task. I have been added a global exception handling to our project by providing the custom middleware that will be handled all of the exception that occurs.

So I will be open the folder middleware. And here is my ErrorHandlingMiddleware class which contains a constructor and a few methods. The main one is HandleExceptionAsync that take as a parameters a HttpContext(this class Encapsulates all HTTP- specific information about an individual HTTP request.) and the second parameter is some specific exception. Then we have GetExceptionCode method which return a status code of the exception depending on the type of exception. For example in the case when exception type is UserNotFound , will be returned the HttpStatusCode.NoContent. In case IncorrectPasswordException wil be returned StatusCode.BadRequest. And as default will be returned InternalServerError code. So when we receive this Status Code. Then our method will be created the response based on status code and exception message. And return this response. You were supposed to notice that standard classes are not used here and yes. Its our custom class. I created this class for specific purpose. For example UserNotFoundException class is used when...we try to extract user from database and this user does not exist exist the we throw this exception , and it will be successfuly handled by middleware and return a respone. It is all good But why we need it . So move on to the UI part of our aplication and try to log in. So I enter the email , then enter some password. And cllick on log in button. But as we see we not logged in. But how user may know that something going wrong ? There were no reports that something went wrong. To prevent this situation we receive a response from back end.And in UI based on this response we will notify the user whats happend.  
The next part of my job was to create safety storage to our sercets. So I decided to use Microsoft **Azure Key Vault it** is a cloud-hosted management service that allows users to encryp t**keys** and small secrets by using **keys** that are protected by hardware security modules .So I successfuly created this vault on azure web site. And then locate our secrets (What I mean under the secrets , for example its conectionString to our database that locate on azure , and JWT key )in this storage. So after this i created a configuration file azurekeyvault.json and here a set a setting in order to have access to my azure key vault. And that in startup class in constructor I set this azure access configuration to our application.After that I can turn to our secrets througth this configuration properties only by pointing the name of secret. You may noticed that this configuration properties is located in IoCContainer. Its some static class. I do this class in order to have access to configuration properties across all solution.

And the last point is that also wrote the unit test to ErrorHandlingMiddleware class to check for his correct working. I used Nunit framework.Here I have two tests in which I verify the correct working this middleware.s