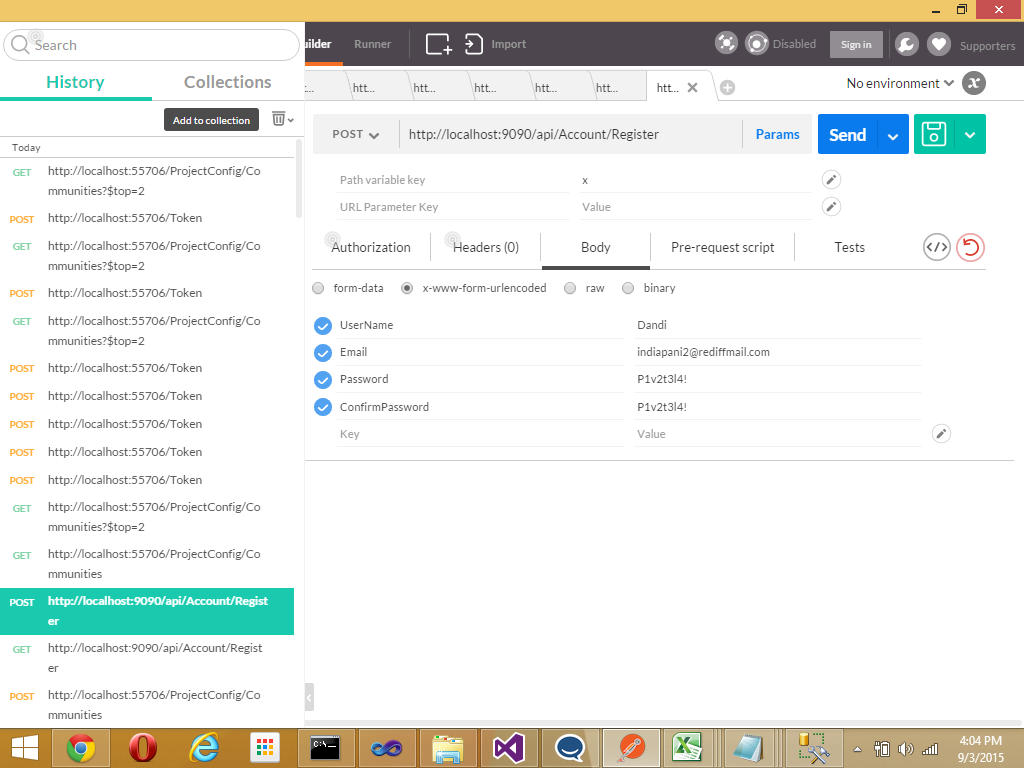
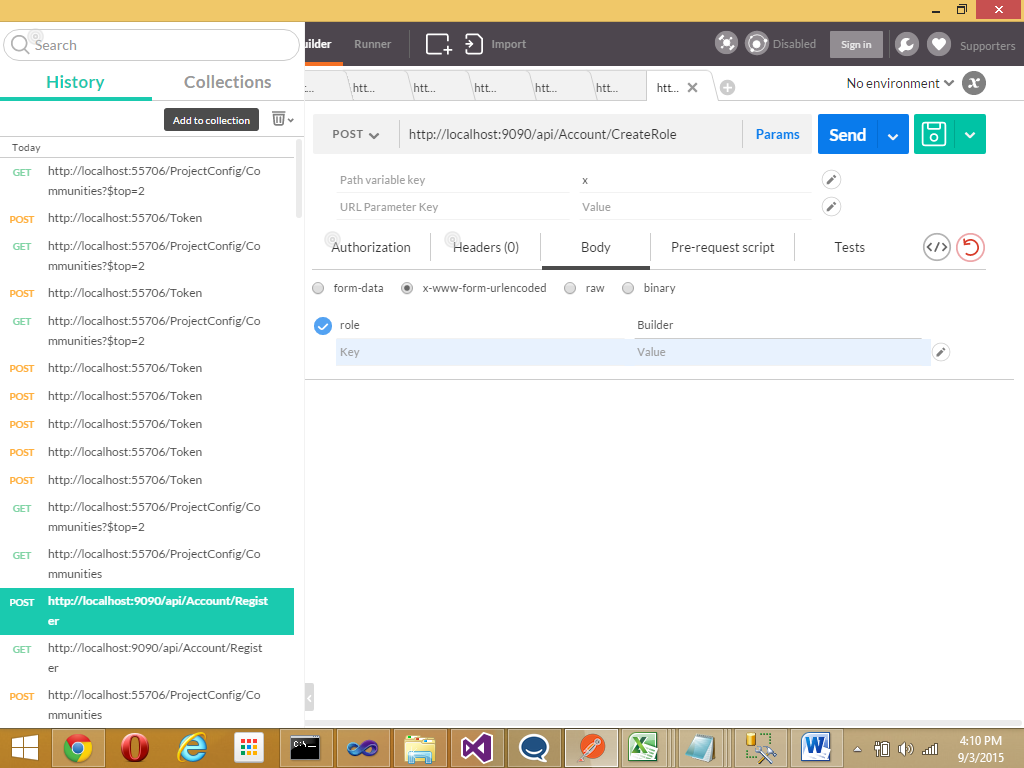
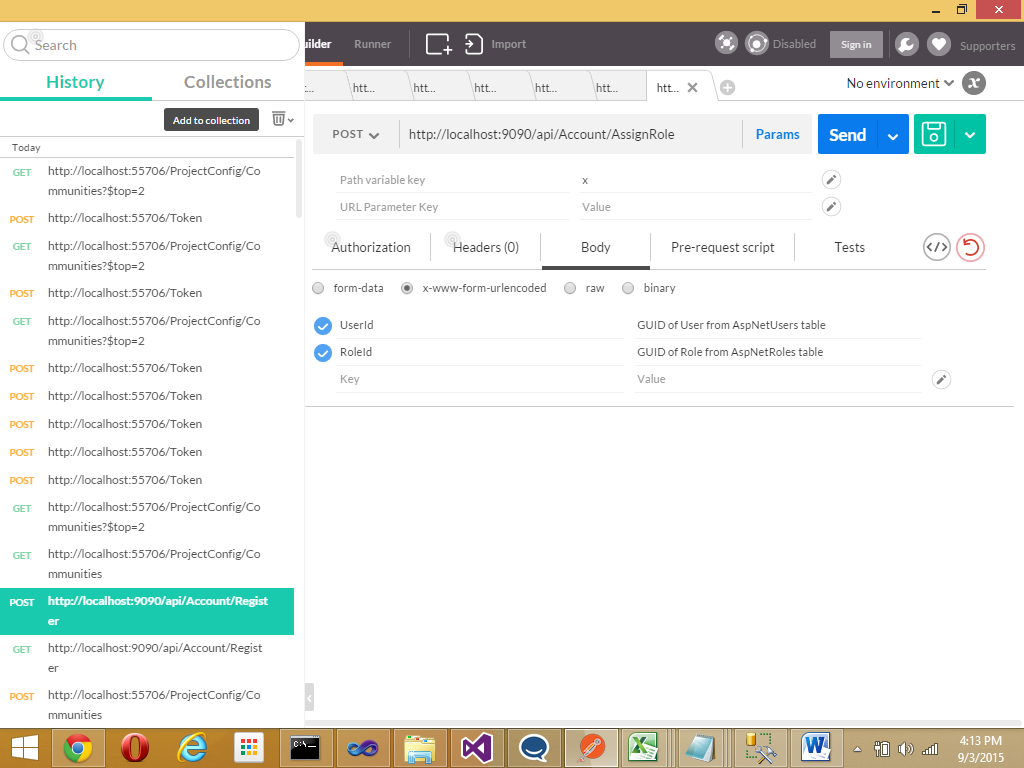
1. User Creation URL: <http://host:port/api/Account/Register> - as in the below example.



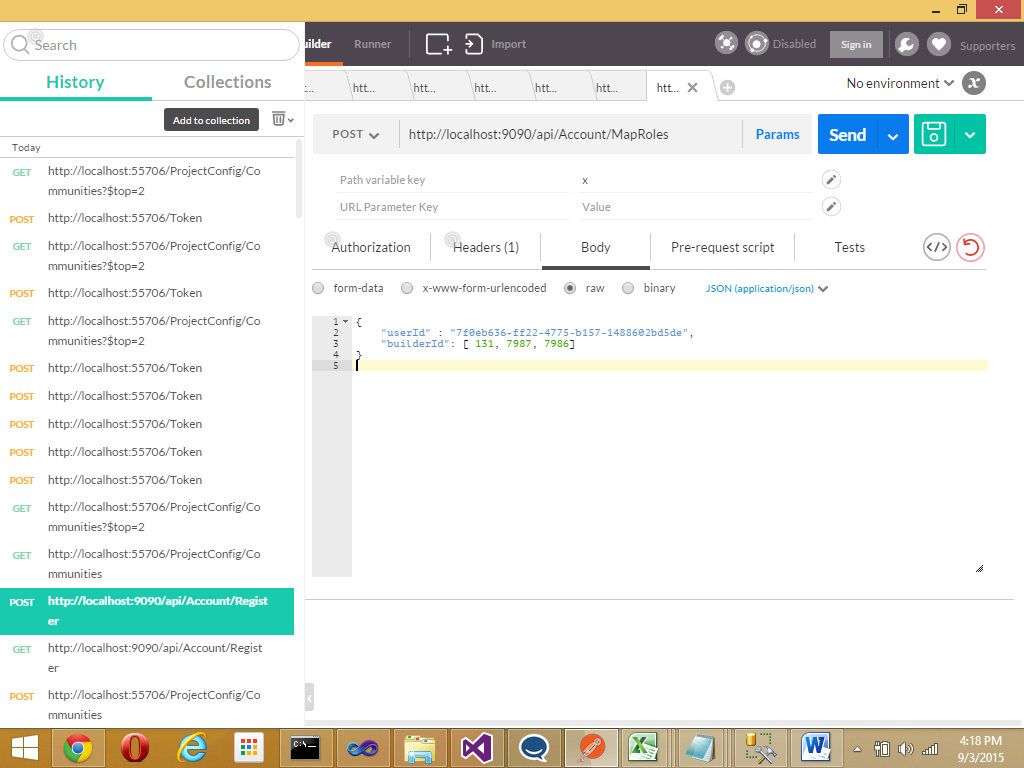
1. Creating a Role URL: <http://host:port/api/Account/CreateRole> - as in the below screen shot.



1. Assigning role to a user URL: <http://host:port/api/Account/AssignRole> - example as in following screen shot. The UserId need to be GUID from AspNetUsers table and RoleId need to be GUID from AspNetRoles table.

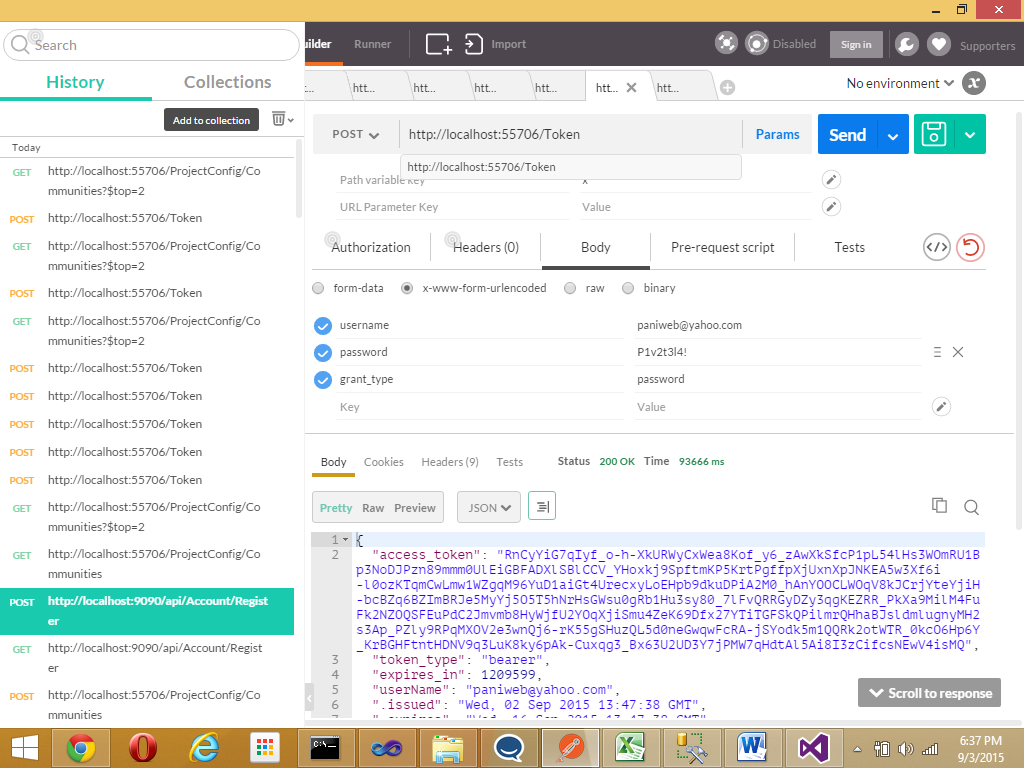


1. Mapping role to builder to restrict access. URL: <http://host:port/api/Account/MapRoles> as in following screen shot.

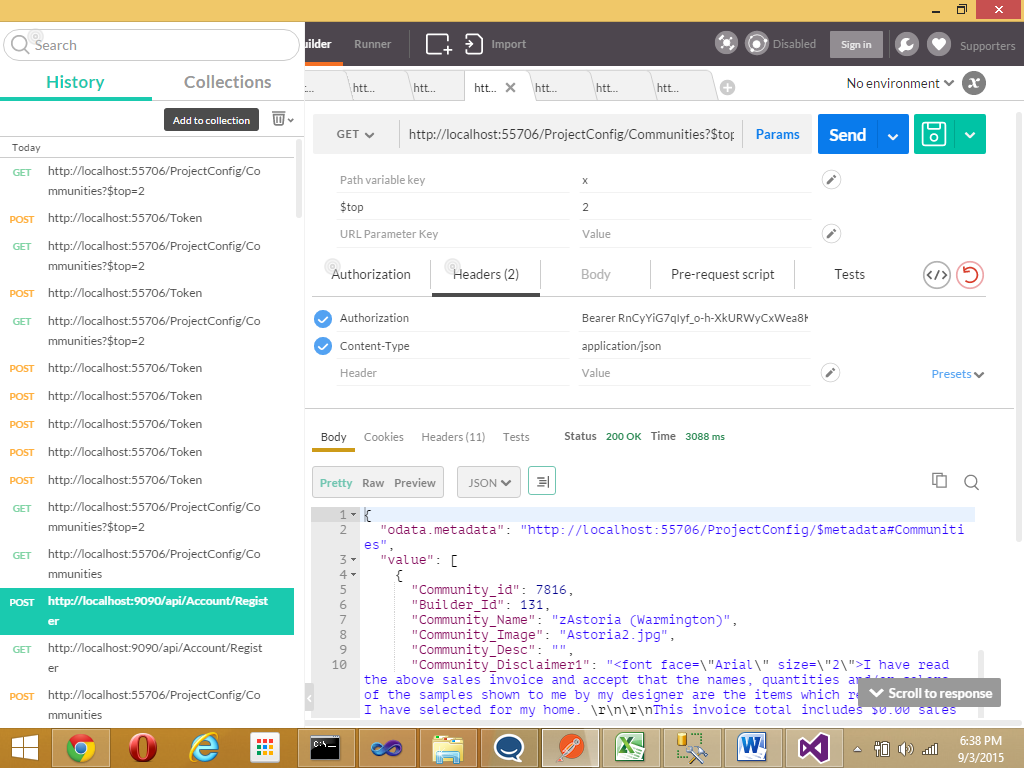


1. URL for Communities: <http://host:port/ProjectConfig/Communities>
2. URL for Plans: <http://host:port/ProjectConfig/Plans>
3. URL for Phases: <http://host:port/ProjectConfig/Phases>
4. URL for Lots: <http://host:port/ProjectConfig/Lots>

Note: The URLs for Communities, Plans, Phases and Lots are all are same for POST, GET and PUT operations. Whatever the URL being used there need to be Authorization Header added to the request with Bearer and Token obtained using <http://host:port/Token> URL. The token can be obtained by sending in the credentials along with grant\_type as password – see the screen shot below:



Authorization can be sent using a request as indicated in the following screen shot.



Note: The current APIs all are authorized to Builder role only. We need to have a valid user created with Builder role to be able to use any of the ProjectConfig APIs.