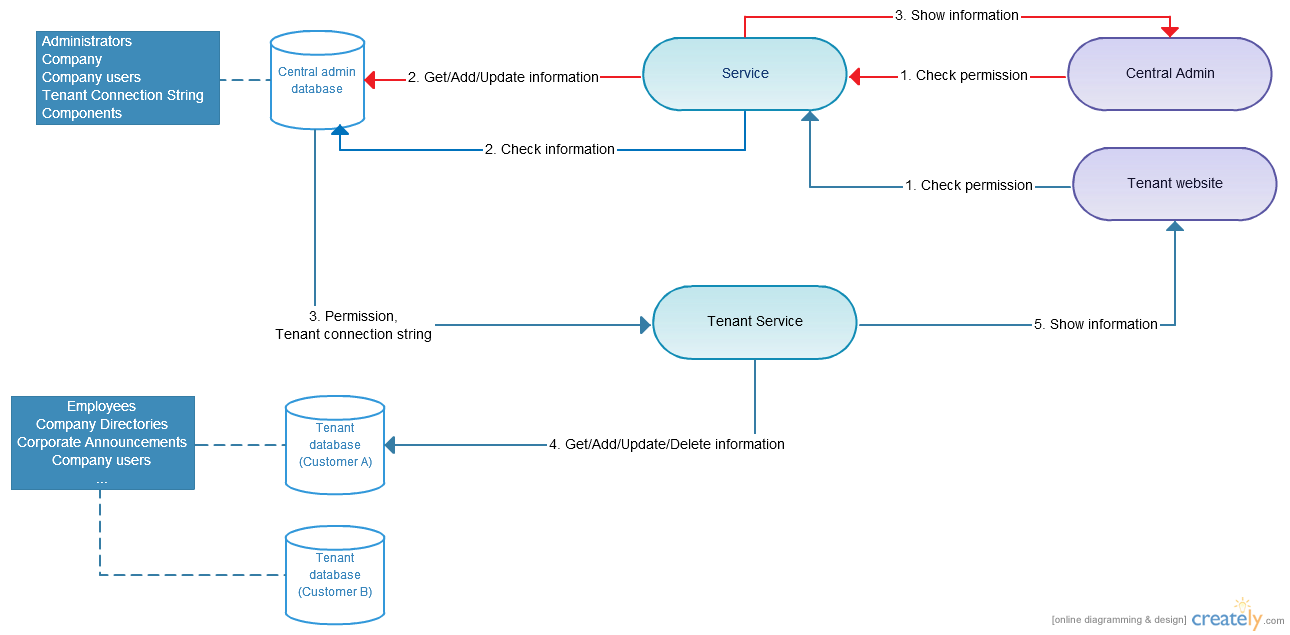
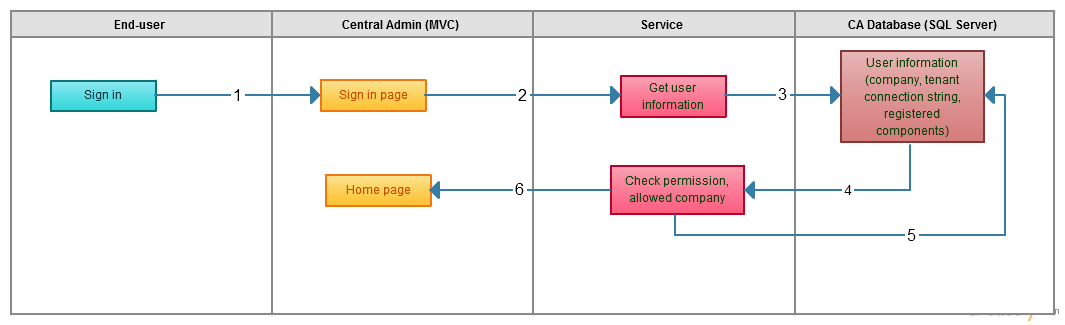
# Architecture diagram

Red line: when administrator sign in

Blue line: when normal user sign in

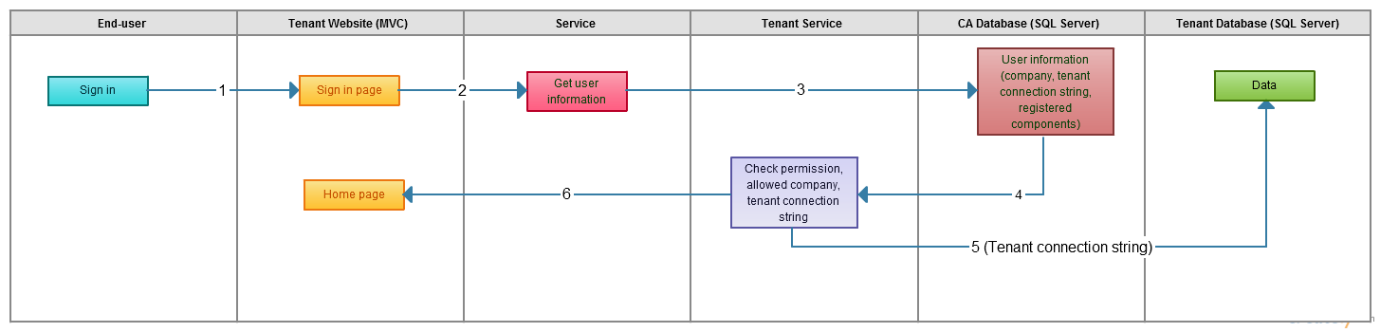


## Administrator sign in



1. Administrator sign in:
   1. Administrator sign in to Central Admin website (ex: <https://ca.kumo-eip.com>).
   2. Central Admin website send input information to the Service.
   3. Service use input information (username and password), connect to Central Admin database (CA database).
   4. CA database return user information (Username, password, permission) base on input information.
   5. Service check the account is valid or not. If it’s valid (username and password is correct and has admin permission), it’ll connect to the CA database to get data (company, components).
   6. Central Admin website will show data returning from Service. Administrator can add/update/delete data to CA database. When administrator add a new company, a separate database will be created for the new company by PowerShell script. To running PowerShell script when a new company is added, CA website will be hosted in Azure Virtual machine (how to connect from internet?)

## Normal user sign in



1. Normal user sign in: (KUMO want to use one url ex: abc.com, when user sign in, the information of their company will be show, so Company users will be saved in CA database and tenant database or only CA database? Each user will be had different permission to view his company’s components)
   1. Normal user sign in to Tenant website (ex: https:// kumo-eip.com)
   2. Tenant website send input information to the Service.
   3. Service use input information (username and password), connect to Central Admin database (CA database) to check the account is valid or not.
   4. If it’s valid (username and password is correct), CA database will return data (company, components, tenant connection string) to the Tenant Service.
   5. Tenant Service will user the connection string from CA database to connect to corresponding tenant database and get data.
   6. Tenant website show data returning from Tenant Service.

# Database Design

## Central Admin core table

### Client Table

|  |  |  |  |
| --- | --- | --- | --- |
| Client Table | | | |
| Property | **Type** | **Required** | **Description** |
| Id | Number | Yes | ID, Auto generate |
| Client Name | String | Yes | Name of client |
| DBName | String | Yes | Name of database store all business data |
| DBUser | String | Yes | Username to connect database |
| DBPassword | String | Yes | Encrypt password using to for client website connect to db |
| DBServer | String | Yes | IP Address or host name of DB Server. Assume that is Azure SQL server. |
| Status | Choice | Yes | Single value using these option : Active, Inactive, Deleted, Pending, Deploying |
| SiteUrl | String | Yes |  |
| Client Code | String | Yes | The company name or Unique code to identity client in the tenant site |
|  |  |  |  |

Depend on business requirement, we may add some extra field to hold Client Information such as Address, Phone Number, Contact …..

### CA User table

|  |  |  |  |
| --- | --- | --- | --- |
| User | | | |
| Property | **Type** | **Required** | **Description** |
| Id | Number | Yes | ID, Auto generate |
| Username | String | Yes | User name to login to CA |
| Email | String | Yes | Email or user to receive email/reset password |
| Password | String | Yes | User password storage as hash using SHA-256 Algorithm |
| Role | String | Yes | There use will have 3 roles, CA Admin; Tenant Admin |
|  |  |  |  |

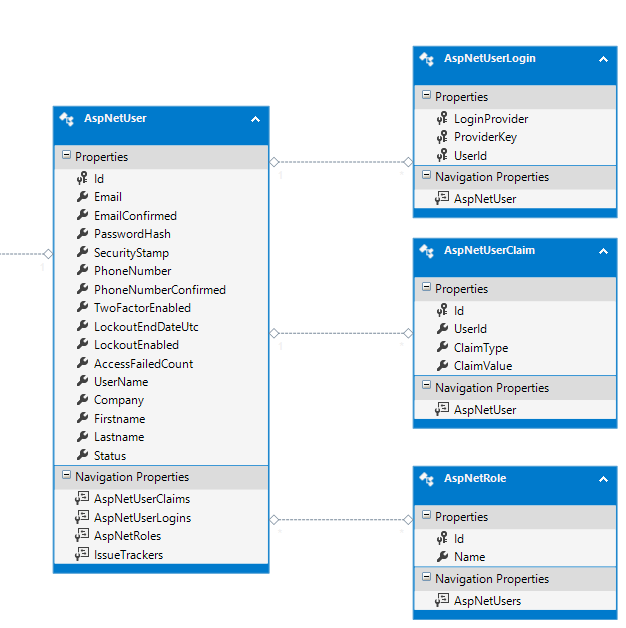
### Client services

|  |  |  |  |
| --- | --- | --- | --- |
| ClientService | | | |
| Property | **Type** | **Required** | **Description** |
| Id | Number | Yes | ID, Auto generate |
| Client | Dropdown | Yes | Select from client table |
| Modules | Multiple Checkbox | Yes | Select modules that current client using:  Core Module  HR  CRM  ……. |
|  |  |  |  |

## Tenant Site database

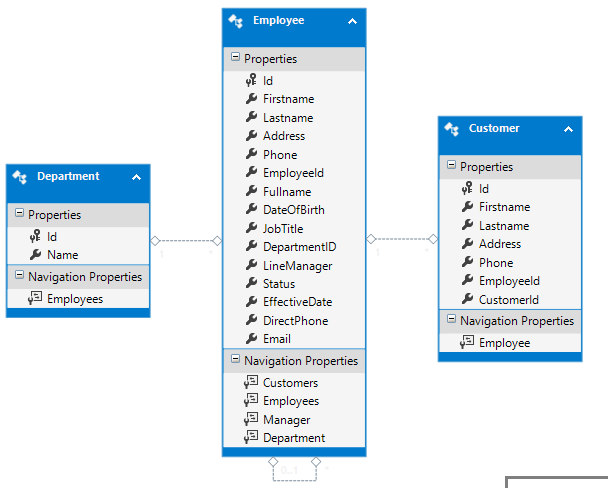
### User & Authentication core

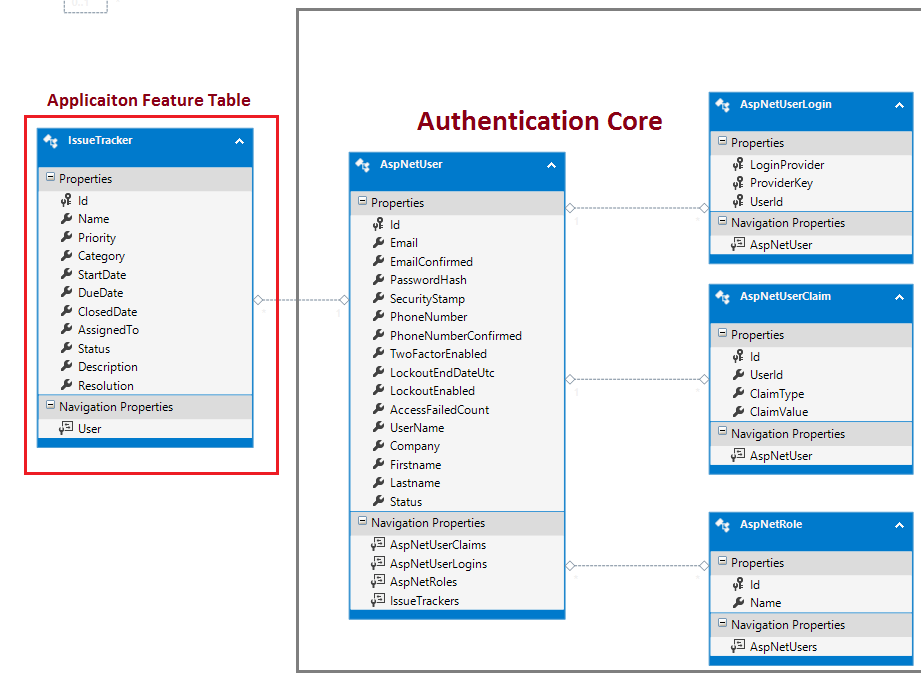
User & Authentication module will delivery by [Identity Framework 2.0](http://www.asp.net/identity/overview/getting-started/introduction-to-aspnet-identity) as the basic model below. This framework was developed and being supported by Microsoft and it provide all the basic function for authentication and authorization which meet to the standard web security and flexibility.



### Application database structure

Databse structure will be design base on the features requirements. I will be develop using the latest version of Entity Framework (current version is 6.1.2). Below are some draft design or application data





- one url

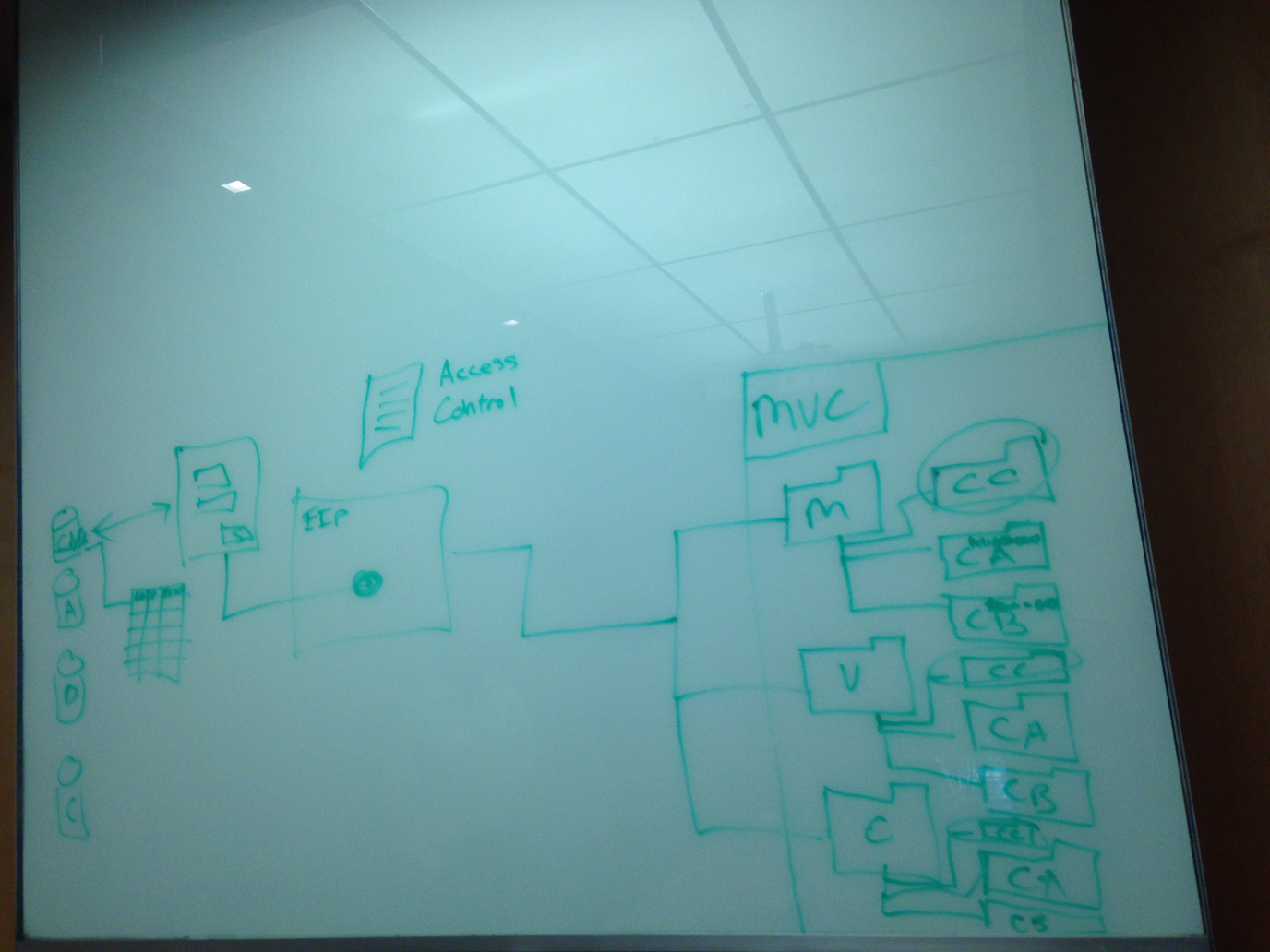
- show company base on user account

- Each company users can see the components that they registered

- Each user in company can see the component that they’re allowed to

2. Central admin is higher priority

- draw diagram like this



- document

- explain what it do, what fields in the db table

- how that will be done

- how it is going to work

- how to create database by powershell script in Azure VM

3. deadline for document: 7h30 PM tomorrow

4. how much time do you need to finish the central admin