**Resolve**

**Version 1.0**

**University of Washington, Seattle**

**School of Dentistry**



**Revision and Signoff Sheet**

**Change Record**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Author** | **Version** | **Change Reference** |
| 14th March 2020 | Palash Jhamnani | 1.0 | First Draft |
| 1st May 2020 | Palash Jhamnani | 1.0.1 |  |

**Reviewers**

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Reviewer** | **Version** | **Comments** |
|  | Anya L. Levysmith | 1.0 |  |

Contents

[Objective 4](#_Toc39229128)

[Chosen Technology 4](#_Toc39229129)

[Test 4](#_Toc39229130)

[Database 6](#_Toc39229131)

# Objective

Resolve is a case management software developed to manage cases, which are service requests created by University of Washington (UW) community members in the UW School of Dentistry. These requests/cases then go through an approval workflow. Reporting dashboards would be created using the database entities for business reporting purposes.

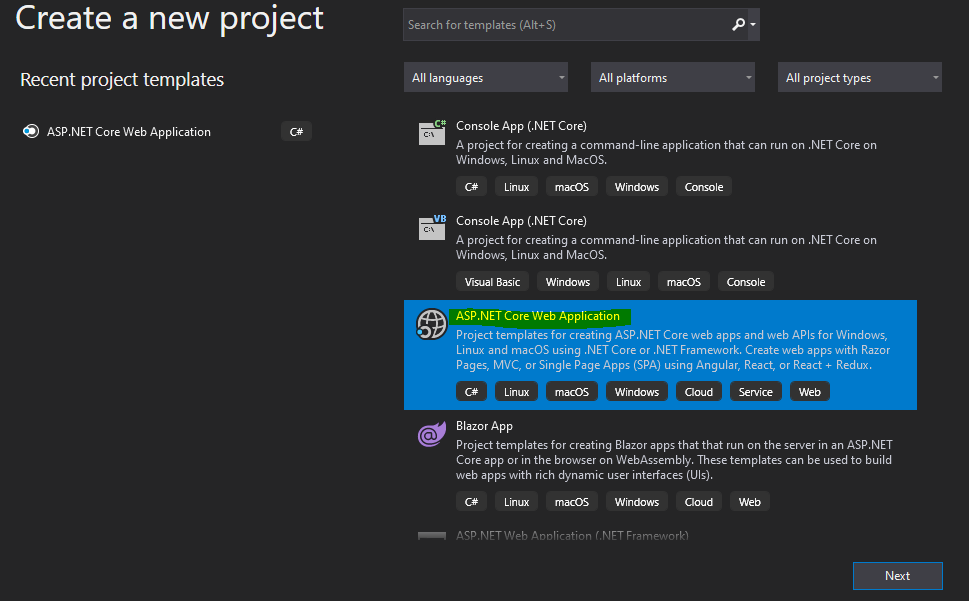
**GitHub Link:** <https://github.com/palashjhamnani/Resolve.git>

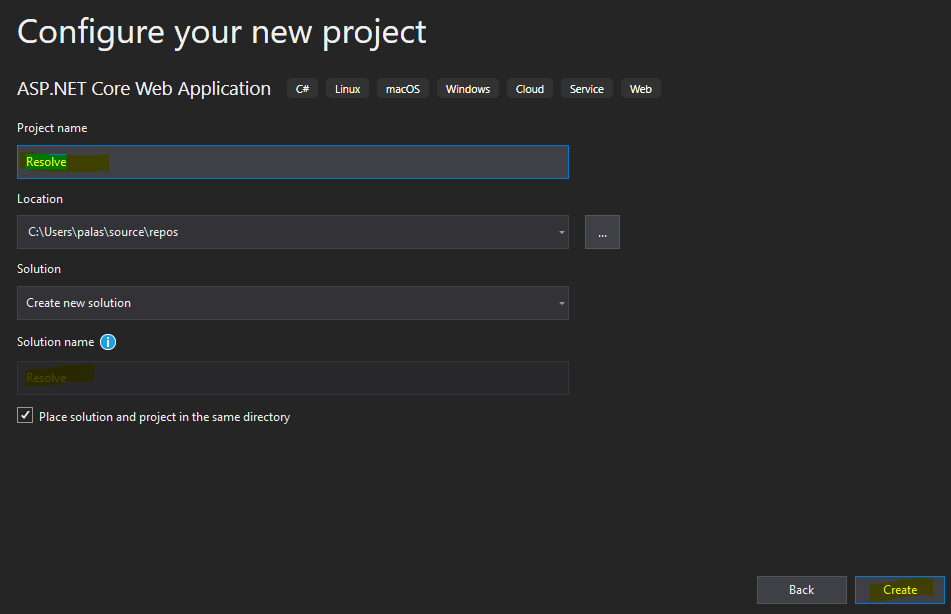
## Chosen Technology

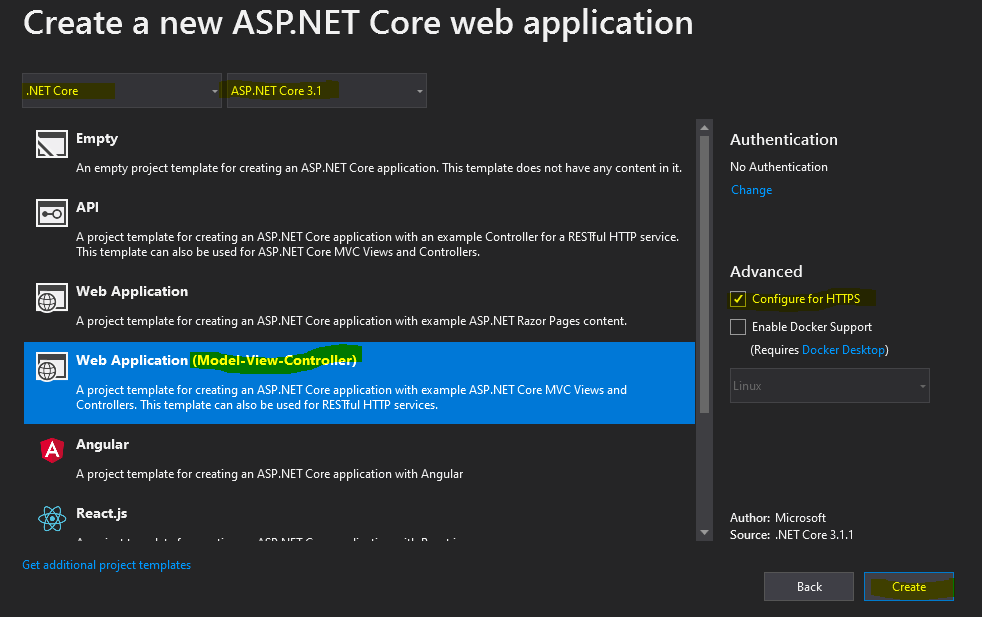
Microsoft .Net Core Framework for building the web application

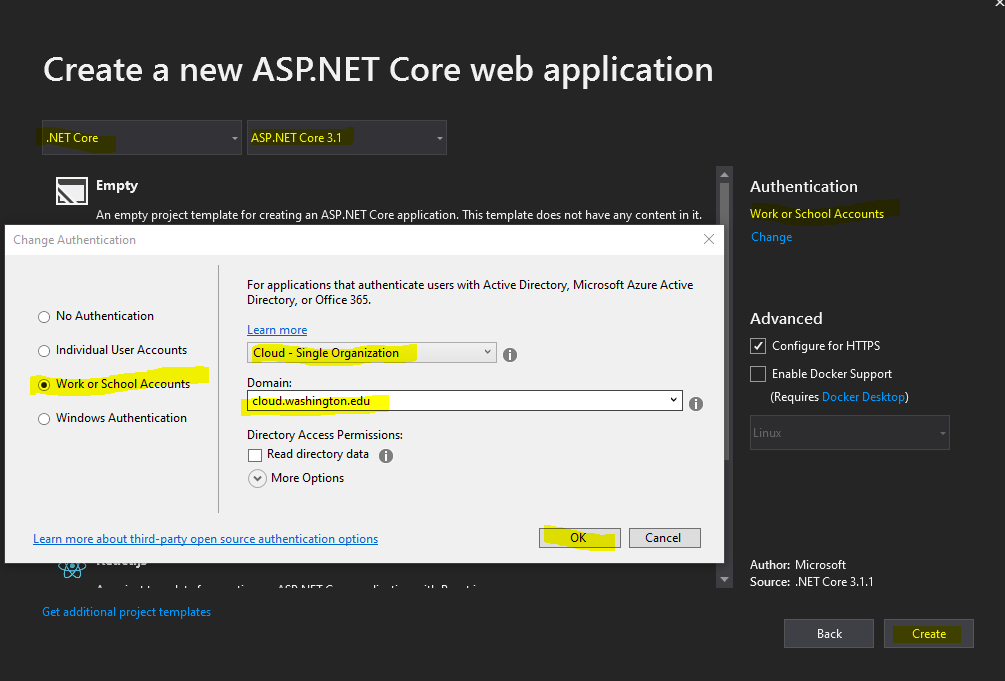
The chosen options have been highlighted in YELLOW in the screenshots below, while developing the application.

## Test









Link to authentication document for integrating UW Net ID

<https://itconnect.uw.edu/wares/msinf/authn/ldap/ldap-authentication-primer/>

<https://itconnect.uw.edu/wares/msinf/authn/ldap/>

<https://wiki.cac.washington.edu/display/infra/Shibboleth+for+UW+Web+Applications>

# 

**Entity Description Table**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No** | **Entity Name** | **Attributes** | **Why is Entity Included?** | **Relation with other Entities** |
| **Folder: Models** | | | | |
| 1 | LocalUser | * FirstName * LastName * NetID * EmailID | Local User table is managed automatically. An entry in this table is created when a user logs in to Resolve for the first time. The data is picked from AzureAD | None |
| 2 | LocalGroup | * LocalGroupID * GroupName * GroupDescription * FK: LocalUser | This table is managed manually by the admin. Groups are created as part of the application setup. ID comes from AzureAD (ObjectID) | One LocalUser is assigned to a LocalGroup as a default lead/approver |
| 3 | UserGroup | * FK: LocalUser * FK: LocalGroup | This table is managed automatically. Entries are created as per User claims from AzureAD | Many-to-many mapping between LocalUser and LocalGroup |
| 4 | CaseType | * CaseTypeID * CaseTypeTitle * LongDescription * FK: LocalGroup | This table is manually managed by the admin. Newer Case Types can be added as required by the application users. | One LocalGroup is assigned to the CaseType as a default group to which the CaseType belongs |
| 5 | Case | * CaseID * Auto: CaseCID * OnBehalfOf * CaseStatus * Auto: CaseCreationTimestamp * FK: CaseType * FK: LocalUser | Case table is managed by the Users. An entry is created for every new case created by a user. | Many-to-One relationship with CaseType and LocalUser. A new Case can be of one CaseType and can be created by one LocalUser |
| 6 | OnBehalf | * PK, FK: Case * FK: LocalUser | This table is managed by the Users. When a case is created by a user on behalf of another user, an entry is created in this table. One case can have only one entry; hence Case is PK. | One-to-One relationship with Case table, and Many-to-One relation with LocalUser table |
| 7 | CaseAttachment | * CaseAttachmentID * FK: Case * FK: LocalUser * FilePath * FileName * Auto: AttachmentTimestamp | Managed by LocalUsers, an entry created for every new attachment, created for a Case by a LocalUser. | Many-to-One relation with Case table and LocalUser table. Many entries can be created for the same Case by many Users |
| 8 | CaseComment | * CaseCommentID * Comment * Auto: CommentTimestamp * FK: Case * FK: LocalUser | Managed by LocalUsers, an entry created for every new comment, created for a Case by a LocalUser. | Many-to-One relation with Case table and LocalUser table. Many entries can be created for the same Case by many Users |
| 9 | CaseAudit | * CaseAuditID * Auto: AuditTimestamp * AuditLog * FK: Case * FK: LocalUser | Managed automatically. An entry is created for every action taken by any user. A function call has to be made at code level by developer for data generation in this table | Many-to-One relation with Case table and LocalUser table. Many entries can be created for the same Case by many Users |
| 10 | GroupAssignment | * FK: Case * FK: LocalGroup | By default, a Case is assigned to one group (as defined in CaseType table), however, if further group assignments are required, then an entry is created in this table. | Many-to-One relation with Case table and LocalGroup table. One Case can be assigned to multiple LocalGroups |
| 11 | Approver | * FK: Case * FK: LocalUser * Approved * Order | One entry is created in this table automatically for every new case created (Case -> CaseType -> DefaultGroup -> DefaultApprover). However, more approvers can be added by the admin or the pre-assigned approver. | Many-to-One relation with Case table and LocalUser table. One Case can be assigned to multiple LocalUsers |
| **Sub Folder: CaseTypeModels** | | | | |
| 12 | SampleCaseType1 | * PK, FK: Case * Attribute1 * Attribute2 * … | To add specific details for a Case Type, new entities can be created by the admin/developer as required by the users. For every new Case Type added, a new model must be created under the “~/Models/CaseTypeModels/” folder, if specific details are required for that CaseType | One-to-One relation with Case table. Only one entry can be created in this table per Case. |
| 13 | SampleCaseType2 | * PK, FK: Case * Attribute1 * Attribute2 * … |  |  |
|  | … | * PK, FK: Case * … |  |  |

PK: Primary Key

FK: Foreign Key

Auto: Automatically Generated by Database on addition of a row

Database used: Microsoft SQL Server

Below is the connection string specified in the *appsettings.json* file.

