Version1: Student Management

* 5 layer architechtur

1. web site
2. web api
3. service
4. model/entity/class/Domain
5. entity framework

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version 1: Student Management | | | | |
| Task 1: Configure Website | Description | Perform By | Hours | Status |
| Create new empty website |  |  |  | Done |
| 1.Bootstrap install  2.Jquery install  3.AngularJs install  4.AngularJs Route install  5.AngularJs Animate install  6.AngularJs Sanitize install  7.AngularJS ui.bootstrap install  8.Jquery toaster install |  |  |  | Done |
| Create new app folder  AngularJs configuration  1.module  2. routing  3. index page |  |  |  | Done |
| 1.Layout design  2.Home page design  3.Custom css folder  4.Custom script folder  5.Image folder |  |  |  | Done |
| AngularJs directive configuration for  Toggle |  |  |  | Done |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version 1: Student Management | | | | |
| Task 2 : Student Module | Description | Perform By | Hours | Status |
| Student User Interface Design  1.Student add  2.Sutudent list view & pagination & Search & Delete  3.Student Details view & Delete  4.Student Update |  |  |  | Done |
| Student controller Design  1.Student add  2.Sutudent list view & pagination & Search & Delete  3.Student Details view & Delete  4.Student Update  5.add routing  6.add in index |  |  |  | Done |
| AngularJs service configuration  1.data constant service  2.student service  3. add in index |  |  |  | Done |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version1: Basic Student Management | | | | |
| Task 3: validation | Description | Perform By | Hours | Status |
| Client Side validation |  |  |  | To Do |
| Server side validation |  |  |  | To Do |
| Database Exception handling |  |  |  | To Do |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version1: Basic Student Management | | | | |
| Task 5: service configuration | Description | Perform By | Hours | Status |
| Create new class library: file>new>project> |  |  |  | Done |
| 1.add reference of entity /class  2.add reference of EF |  |  |  | Done |
| 1.create model class  2.create Service |  |  |  | In Progress |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version 1: Student Management | | | | |
| Task 4: Api configuration | Description | Perform By | Hours | Status |
| Create new empty web api: file>new>project>web>empty>web api |  |  |  | Done |
| Configure web api  1.install cors2  2.Enable cors2 in WebApiConfig file  3.add serilization json in WebApiConfig file  4.install entity framework 6  5. add connetion string in WebConfig file |  |  |  | Done |
| 1.Add student controller  2.Add origin |  |  |  | Done |
| 1.Add reference of service  2.add student view model  3.add response view model |  |  |  | Done |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version1: Basic Student Management | | | | |
| Task 7: Entity Framework | Description | Perform By | Hours | Status |
| Create new class library: file>new>project> |  |  |  | Done |
| 1.install entity framework 6  2.add connetion string in WebConfig file  3.add reference of entity/class |  |  |  | Done |
| 1.create db context  2.mange many to may relationship in dbcontext  3.manage some initial configuration in dbcontext  4.manage lazy load/eager load in dbcontext |  |  |  | In Progress |
| create Repository |  |  |  | In Progress |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Version1: Basic Student Management | | | | |
| Task 6: entity /class design | Description | Perform By | Hours | Status |
| Create new class library: file>new>project> |  |  |  | Done |
| 1.create entity/entity  2. relationship design  3. property set  4. lazy load/eager load manage |  |  |  | In Progress |