3. Выбор метода маршрутизации и определение основных маршрутов.  
metanit.com/sharp/aspnet5/11.2.php

Стандартным методом определения маршрутов для ASP является использование метода MapRoute. Определяется ряд паттернов запросов, например:

endpoints.MapControllerRoute(  
 **name**: "default", **pattern**: "{**controller**=Home}/{**action**=Index}/{**id**?}");

Данный вызов определяет контроллер и действие по умолчанию (таким образом, что вызов website.net эквивалентен вызову website.net/Home/Index), а также возможность передачи меты в виде id. Не существует смысла особым образом усложнять маршрутизацию в разрабатываемом проекте, выберем банальный метод и спроектируем конкретную структуру.

Общий паттерн: website.net/{controller=Home}/{actionName?}/{meta?}

Для контроллера Home:  
*website.net/Home/Summary* **или** *website.net/Home***или** website.net // Домашняя страница сайта  
*website.net/Home/Policy* // Страница политик и соглашений

Для контроллера Project:  
*website.net/Project/Summary* ***или*** *website.net/Project //* Последние опубликованные проекты и т.п.  
*website.net/Project/Create //* Создание нового проекта  
*website.net/Project/Show/ProjectId\_Int32 // Основная страница проекта*  
*website.net/Project/Edit/ProjectId\_Int32 // Редактирование проекта*website.net/*Project*/Delete*/ProjectId\_Int32 // Удаление проекта*

Для контроллера ProjectsGroup:  
*website.net/*ProjectsGroup*/Summary* ***или*** *website.net/*ProjectsGroup *//* Последние опубликованные группы  
*website.net/*ProjectsGroup*/Create //* Создание нового проекта  
*website.net/*ProjectsGroup*/Show/*ProjectsGroup*Id\_Int32 // Основная страница проекта*  
*website.net/*ProjectsGroup*/Edit/*ProjectsGroup*Id\_Int32 // Редактирование проекта*website.net/ProjectsGroup/Delete*/*ProjectsGroup*Id\_Int32 // Удаление проекта*

Для контроллера Document:  
website.net/Document*/Summary* или website.net/Document // Последние добавленные документы  
website.net/Document/Create // Создание нового документа  
website.net/Document/Show/DocumentId\_Int32 *// Основная страница документа*  
website.net/Document/Edit/DocumentId\_Int32 *// Редактирование документа*  
website.net/Document/Delete/DocumentId\_Int32 *// Удаление документа*

*Для контроллера User:*website.net/*User/Summary* **или** website.net/*User* // Последние регистрации на сайте  
website.net/*User/*Create // Регистрация нового пользвателя  
website.net/*User*/Show/*UserId*\_Int32 *// Основная страница аккаунта*  
website.net/*User*/Edit/*UserId*\_Int32 *// Редактирование основных данных аккаунта*  
website.net/*User*/Delete/*Users*Id\_Int32 *// Удаление аккаунта*

Все «действия» (action) должны определять запрос HttpGet, но дополнительно:  
Create: HttpPost  
Edit: HttpPatch  
Delete: HttpDelete

*Как было сказано ранее, в проект изначально закладывается инфраструктура для преобразования его в своеобразный блог. Изначально это заключается в публикации любого материала на сайте от имени пользователя. Естественно в период существования сайта без возможности регистрации на нём любого человека любая публикация будет производится от пользователя с id:0 и name:Admin. Данные сведения, поначалу, будут умирать на этапе контроллера, однако извлечение из БД будет производится с целью обеспечения более легкой дальнейшей модификации.*