# Routing

* This will allow ASP.NET MVC to know where the user should go.
* It uses **routing middleware** to be able to find the appropriate controller and action to handle the request.
* You can also define the routing format in your application

## Convention-based routing

* It enables you globally define the URL formats that your app accepts and how each of those formats map to a specific action method on a given controller.
* Defined in startup.cs specifically the .UseEndpoint() method.

## Attribute routing

* It allows you to specify the routing information by decorating your controllers and actions with attributes that defines the route.

# Filters

* They are used to verify the data as well as define logic that should run before/after the action is executed.

## Types of Filters

* Authorization
  + Used to determine whether the user is authorized for that request.
  + It will run the filter first.
* Resource
  + Used for logging, caching, and other resource related operations.
  + You can set up resource filters that prevents model binding from access the form data.
  + Will run after authorization.
* Action
  + Modifies the way in which the action is executed.
* Exception
  + Used to apply global policies to unhandled exceptions that occur before anything has been written to the response body.
* Result
  + Run only when the action method has executed successfully.
  + Run code immediately before and after the execution of individual action results.

# Middleware

* It kinda is like a filter but they deal with HTTP request and response after it has left the MVC context.
* Software that is assembled into the app pipeline that handles request and response.
* It will know what controller to point to and it will know what to construct the http response to include all the necessaries files (html, css, js, etc.)
* Note- Routing middleware and CORS

# Unit Testing in MVC (specifically the controller)

## Mock

* They will replace a function that a unit test depends on and guarantee that the function will work.
* In that way, we are unit testing the controller only and not the BL layer.