



Översikt

- Analysera arkitektur för informationen.
- Konfigurera routes.
- Skapa struktur för navigering.

.eerec

Lektion 1: Analysera arkitektur för informationen

- Introduktion.
- Arkitektur för information.
- Optimering för sökmotorer.
- Optimering för sökmotorer –web bot.
- Optimering för sökmotorer –tips.

.eerec

Introduktion

- MVC-applikation kan bli väldigt stor, komplex och kan innehålla olika typer av information som relaterar till varandra.
- Utmaning är att säkerställa så att användare lätt kan lokalisera informationen.
- Ett sökverktyg kan vara ett bra hjälpmedel, men du skall även försöka designa din webbapplikation på sådant sätt att användare kan navigera med en kort sekvens av länkar.
- Du måste även se till att sökmotor på Internet kan genomsöka din webbsite.

.eerc

Arkitektur för information

- När du analyserar use cases och user stories för att planera model för din webbapplikation, måste du försöka ta fram vad för typ av objekt som din webbapplikation skall hantera.
- Model beskriver logisk struktur för objekt i din webbapplikation.
- Men du måste också tänka på hur dessa objekt, presenteras för dina användare.

.eerc

Arkitektur för informationen (forts.)

- Arkitekturen för informationen skall presenteras i formen av navigeringskontroller.
- Site Menu:
 - Huvudmeny.
- Tree view:
 - Visar meny i trädstruktur.
- Breadcrumb Trails:
 - Navigationskontroll som visas var användare är i strukturen.
- Presentera informationen genom URL, lättare att länka till.

.eerc

Optimering för sökmotorer

- Flesta besökare hittar webbapplikation genom sökmotorer.
- Besökare tenderar att använda de som ligger överst.
- Utvecklare och administratörer måste försöka att de till att deras webbapplikation kommer så långt upp som möjligt.
- Search Enging Optimization (SEO) används för detta.



Optimering för sökmotorer –web bot

- Sökmotor genomsöker innehåller i din webbapplikation med ett program som kallas för web bot.
- Web bots undviker ofta länkar i form av GUID eller långa söksträngar i URL.
- Om nyckelord finns i URL prioriteras dessa.



Optimering för sökmotorer –tips

- Använd meningsfull <title>-element.
- Använd <meta name="nyckelord">.
- Använd <meta name="beskrivning">.
- Använd olika <title> och <meta> taggar för de olika sidorna.
- Använd ett domännamn som inkluderar nyckelord.
- Se till att navigeringskontroller ger möjlighet för web bots att gå igenom site.
- Se till att URL inte inkluderar GUID eller långa söksträngar.



Lektion 2: Konfigurera routes

- ASP.NET Routing Engine.
- Default route.
- MVCHandler.
- Default route -exempel.
- RouteTable-objekt.
- Egenskaper för routes.

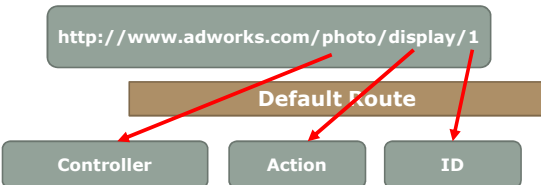
.eerec

ASP.NET Routing Engine

- Routing handlar om hur URL formuleras och hur dessa korresponderar till controllers och actions.
- Routing opererar inte på protokoll, server, domän eller portnummer för URL, bara för kataloger och filnamn i den relativa URL:en.
- Två syften för routes:
 - Anlys begäran och skicka vidare till rätt controller och action. Dessa kallas för inkommande URL:s.
 - Formulera URL för länk eller andra element. Om du använder helpers såsom `Html.ActionLink()` eller `Url.Action()` i MVC Views, kommer dessa att konstruera länkar enligt routingtabell.

.eerec

Default route



- Skräddarsydd routes:
 - För att göra URL lättare för användare att förstå.
 - För att förbättra sökrankningen.

.eerec

MVCHandler

- Objekt för MVCHandler skapar Controller Factory.
- Controller factory är ett objekt som instansierar controller som skall svara på förfrågan.
- Controller factory undersöker routing table för att försöka räkna ut rätt controller klass.
- Controller factory skapar controller object, MVC Handler kallar på metoden Execute i denna controller.
- Controller Action Invoker undersöker URL i begäran, tittar i routing table för att räkna ut action i Controller object som skall köras.

.eeec

MVCHandler (forts.)

- Controller ActionInvoker kör action.
- Skapar oftast ny instans av modelklass, model object skickas till View för att visas som resultat.

.eeec

Default route -exempel

```
Routes.MapRoute(  
    name: "Default",  
    url: "{controller}/{action}/{id}",  
    default: new {  
        controller = "Home",  
        action = "Index",  
        id = UrlParameter.Optional  
    }  
);
```

.eeec

RouteTable-objekt

- Varje MVC-applikation har ett routingtable object, i detta objekt sparas routes,
- Du kan lägga till vägar, genom att kalla på metoden `MapRoute()`.
- I mall för MVC-applikation finns en dedicerad fil `RouteConfig.cs`.
- I `Global.asax.cs`, finns metoden `Application_Start()`, denna metod kallar på `RouteConfig.RegisterRoutes()`, vilket betyder att routes läggs till när MVC-applikation startas.

.esrec

Egenskaper för routes

- *Name*, string, används för att tilldela ett namn. Name är inte med i matchningen.
- *URL*, string, används vid matchning:
 - "photo/{title}" matchar begäran som startas med photo och som innehåller ytterligare parametrar.
- *Constraints*, ytterligare parameter, t ex "photo/{id}" för att matcha photo med specifikt id. Kommer även att matcha photo/create, använd regular expression för att specificera bara siffror.

.esrec

Egenskaper för routes (forts.)

- Routes undersöks i den ordning som de finns i tabellen.
- Om träff hittas, kommer denna att användas, även om det finns annan route.
- Default matchar alla request, även de utan relativ URL.

.esrec

Övning Lägga till routes



Lektion 3: Skapa struktur för navigering

- Viktigt med bra navigering .
- MVC Site Map Provider.
- MVC Site Map Provider -konfiguration.
- Lägga till Menu Controls.

.eerec

Viktigt med bra navigering

- Försäkra dig om att användare lätt kan besluta om vilken länk som han/hon skall klicka på.
- Tillhandahåll navigationsmöjlighet:
 - Top Menus.
 - Tree views.
 - Breadcrumbs Trails.
 - Footer Menus.
- Använd MVC Site Map Provider för att snabbt bygga arkitektur för informationen tillsammans med navigering.

.eerec

MVC Site Map Provider

- Är en MVC-komponent som lagrar en logisk hierarki för din webbapplikation.
- Lagrar konfiguration i en XML-fil, den hierarki som skapas är oberoende av model, controllers och actions i din applikation.
- Inkluderar även HTML helpers, som kan användas för att rendera menyer, tree views och breadcrumbs trails.

.esrec

MVC Site Map Provider -konfiguration

- När funktion installeras, har element lagts till i web.config-filen.
- Elementet:
 - <siteMap> har lagts till i sektionen <system.Web>

.esrec

Lägga till Menu Controls

- Efter det att du har lagt till site map provider och skapat site map file, kan du lägga till menyer, tree views och andra navigationskomponenter till din view.
- Rendera meny:
@Html.MvcSiteMap().Menu(false, false, true)
- Rendera Breadcrumb Trail:
@Html.MvcSiteMap().SiteMapPath()

.esrec

Övning Bygga navigering för site



Repetitionsfrågor

