NEW MEDIA &
COMMUNICATION
TECHNOLOGY

ASP.NET MVC

NEW MEDIA &
COMMUNICATION
TECHNOLOGY

Routing

Routing

➔ URL = eerste contact met webapplicatie

➔ Verzorg URL’s binnen uw webapplicatie

➔ Belangrijk voor SEO, Google zal hier rekening mee houden bij ranking

Routing

- Enkele richtlijnen voor URL'S (www.useit.com)
  - Kies een duidelijke domeinnaam
  - Korte URL's
  - Makkelijk te tikken URL's
  - URL met site structuur reflecteren
  - Hackable URL's
    - http://www.wijnen.be/cataloog/rood/2012
    - http://www.wijnen.be/cataloog/rood/
  - URL's die niet wijzigen
  - Meer info
    - https://2002-2012.mattwilcox.net/archive/entry/id/990/

Routing

Mappen van inkomende browser request naar MVC controller action

Routing

- 2 manieren
  - Klassieke routing manier
  - Attribuut routing
    - Wij kiezen deze manier

Routing

➔ Global.asax

```
0 rererences
public class MvcApplication : System.Web.HttpApplication
{
    0 references
    protected void Application_Start()
    {
        AreaRegistration.RegisterAllAreas();
        GlobalConfiguration.Configure(WebApiCo   g.Register);
        FilterConfig.RegisterGlobalFilters(Gl   alFilters.Filters);
        RouteConfig.RegisterRoutes(RouteTable.Routes);
        BundleConfig.RegisterBundles(BundleTable.Bundles);
    }
}
```

Routing tables met alle routes

Routing

➔RouteConfig.cs (App_Start folder)

```
1 reference
public class RouteConfig
{
    1 reference
    public static void RegisterRoutes(RouteCollection routes)
    {
        routes.IgnoreRoute("{resource}.axd/{*pathInfo}");

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

Routing

➔RoutData klasse

➔ Opvragen route parameters

```
var controller = RouteData.Values["controller"];
var action = RouteData.Values["action"];
var id = RouteData.Values["id"];
```

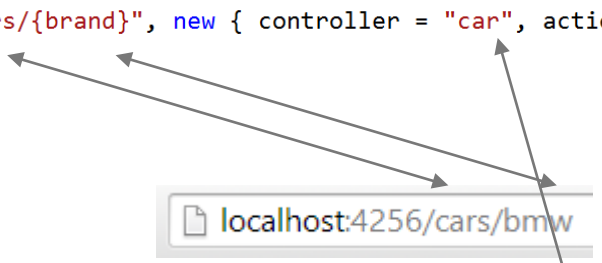
Routing

➔Nieuw route toevoegen

- ➔ RouteConfig.cs
- ➔ Oppassen waar je deze plaats
- ➔ Route die eerst werkt zal genomen worden

Routing

```
routes.MapRoute("car", "cars/{brand}", new { controller = "car", action = "search", brand = UrlParameter.Optional });
```

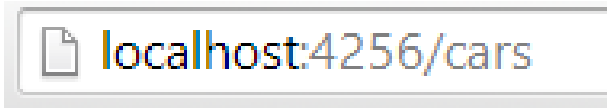


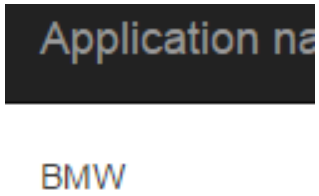

```
0 references
public class CarController : Controller
{
    0 references
    public ActionResult Search(string brand)
    {
        return View();
    }
}
```

Routing

➔ Default waarde opgeven

```
0 references
public ActionResult Search(string brand = "BMW")
{
    return View();
}
```





Routing

➔ Klassiek manier

- ➔ Lastiger om te beheren
- ➔ Goed voor bestaande projecten
- ➔ Centrale locatie voor beheer routes

➔ Wij gaan voor Attribute Routing kiezen

Routing

➔Attribute Routing

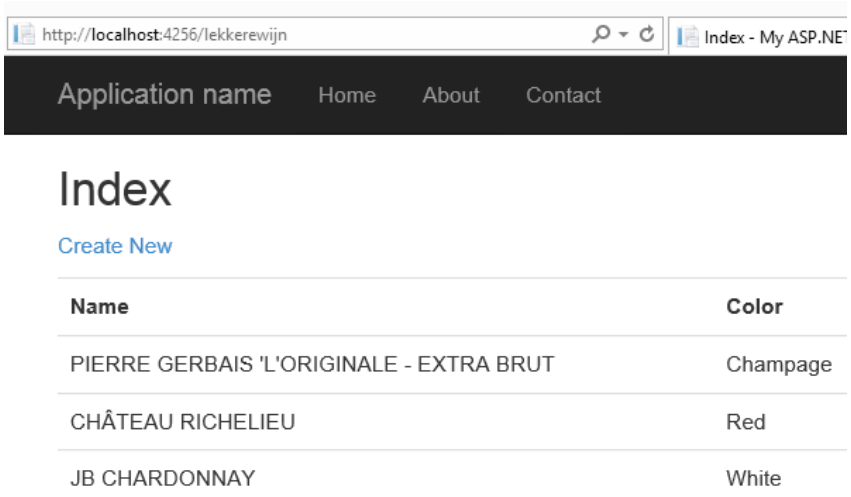
- Wijzig RouteConfig.cs
- Verwijder Routes en vervang door MapMvcAttributesRoutes();

```
1 reference
public static void RegisterRoutes(RouteCollection routes)
{
    routes.MapMvcAttributeRoutes();
}
```

Routing

- [Route()] attribute
  - Plaatsen boven ActionMethod
  - Meerdere mogelijk

```
[Route("")]
[Route("lekkerewijn")]
[Route("wines")]
0 references
public ActionResult Index()
{
    return View(Wines);
}
```

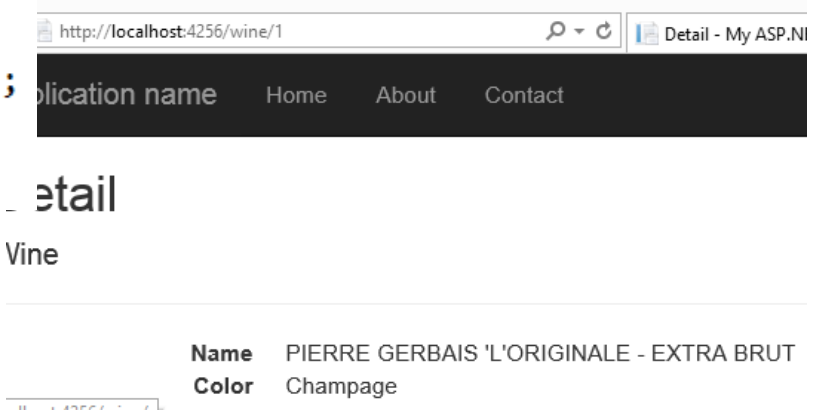


Routing

➔ Route Parameters

➔ Binnen de route template parameter opgeven

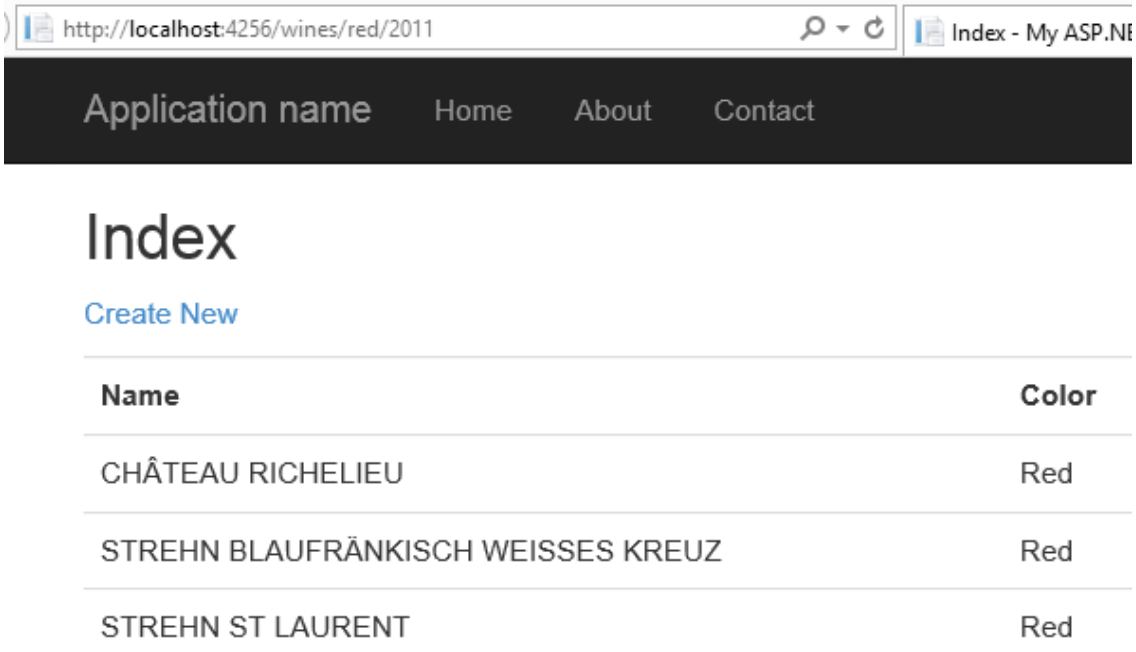
```
[Route("wine/{id}")]
0 references
public ActionResult Detail(int? id)
{
    Wine wine = Wines.Find(w => w.Id == id.Value);
    return View(wine);
}
```



Routing

➔ Meerdere parameters mogelijk

```
[Route("wines/{color}/{year}")]
0 references
public ActionResult ByColorAndYear(string color, int year)
{
```



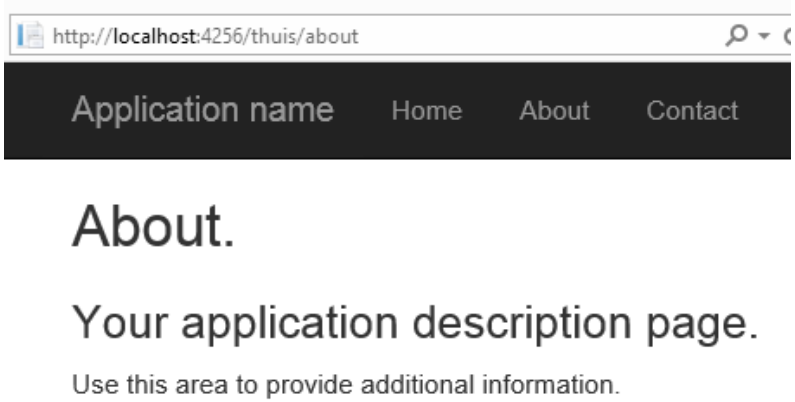
Routing

➔ Route op Controller niveau

➔ Attribute op actionmethod hebben voorrang

Action binnen controller

```
[Route("thuis/{action}")]
0 references
public class HomeController : Controller
{
```



Routing

➔ Route Constrains

➔ Onderstaande voorbeeld

➔ wine/1

➔ wine/rood ➔ Crash ➔ Id = Integer

```
[Route("wine/{id}")]
0 references
public ActionResult Detail(int? id)
{
    Wine wine = Wines.Find(w => w.Id == id.Value);
    return View(wine);
}
```

Routing

➔ Je kan type opgeven in parameter

➔ wine/{id:int}

```
[Route("wine/{id:int}")]
0 references
public ActionResult Detail(int? id)
{
    Wine wine = Wines.Find(w => w.Id == id.Value);
    return View(wine);
}
```

Routing

➔ Andere constrains

Constraint	Description	Example
alpha	Matches uppercase or lowercase Latin alphabet characters (a-z, A-Z)	{x:alpha}
bool	Matches a Boolean value.	{x:bool}
datetime	Matches a **DateTime** value.	{x:datetime}
decimal	Matches a decimal value.	{x:decimal}
double	Matches a 64-bit floating-point value.	{x:double}
float	Matches a 32-bit floating-point value.	{x:float}
guid	Matches a GUID value.	{x:guid}
int	Matches a 32-bit integer value.	{x:int}

Routing

➔ Andere constrains

length	Matches a string with the specified length or within a specified range of lengths.	{x:length(6)} {x:length(1,20)}
long	Matches a 64-bit integer value.	{x:long}
max	Matches an integer with a maximum value.	{x:max(10)}
maxlength	Matches a string with a maximum length.	{x:maxlength(10)}
min	Matches an integer with a minimum value.	{x:min(10)}
minlength	Matches a string with a minimum length.	{x:minlength(10)}
range	Matches an integer within a range of values.	{x:range(10,50)}
regex	Matches a regular expression.	{x:regex(^\d{3}-\d{3}-\d{4}$)}

Routing

➔ Route Defaults

- ➔ Als we geen action opgeven
- ➔ Default waarde nemen uit attribuut

```
[Route("thuis/{action=Index}")]
0 references
public class HomeController : Controller
{

    0 references
    public ActionResult Index()
    {
```

Routing

- Optional parameters
  - Als niet alle parameters verplicht op te geven zijn
  - Parameter eindigen met ?
  - Nullable type gebruiken

```
[Route("wines/{color}/{year?}")]
0 references
public ActionResult ByColorAndYear(string color, int? year)
{
```

Routing

- Web API Routing
  - Zelfde als in ASP.NET MVC

```
0 references
public class WineMobileController : ApiController
{
    [Route("api/wines")]
    0 references
    public IEnumerable<Wine> Get()
    {
        return Data.GetWines();
    }
}
```

  - http://www.asp.net/web-api/overview/web-api-routing-and-actions/routing-in-aspnet-web-api

NEW MEDIA &
COMMUNICATION
TECHNOLOGY

Error Handling

Error Handling

➔Probleem in controller action

➔Exception gooien

```
0 references
public ActionResult About()
{
    throw new Exception("Probleemje");
    return View();
}
```

Server Error in '/' Application.

Probleemje

Description: An unhandled exception occurred during the execution of the current web request. Please review the stack trace for more information about the error and where it c

Exception Details: System.Exception: Probleemje

Source Error:

```
Line 16:        public ActionResult About()
Line 17:        {
Line 18:            throw new Exception("Probleemje");
Line 19:
Line 20:            return View();
```

Source File: c:\Users\dieter\OneDrive\NMCT\2014 - 2015\Server Side Application\Week 6\Theorie\ErrorHandling\ErrorHandling\Controllers\HomeController.cs **Line:** 18

Error Handling

➔ Mooie foutmelding weergeven

➔ Web.Config

On = altijd eigen foutmeldingscherm gebruiken

RemoteOnly = eigen foutmelding enkel bij bezoekers van website, op server echte fout tonen

Off= nooit eigen foutmelding gebruiken

```
<system.web>
  <customErrors mode="On"/>
  <authentication mode="None" />
  <compilation debug="true" targetFramework="4.5" />
  <httpRuntime targetFramework="4.5" />
</system.web>
```

Error Handling

➔ HandleErrorAttribute

➔ Toevoegen via FilterConfig.cs

```
1 reference
public class FilterConfig
{
    1 reference
    public static void RegisterGlobalFilters(GlobalFilterCollection filters)
    {
        filters.Add(new HandleErrorAttribute() { View = "Error"});
    }
}
```

Error Handling

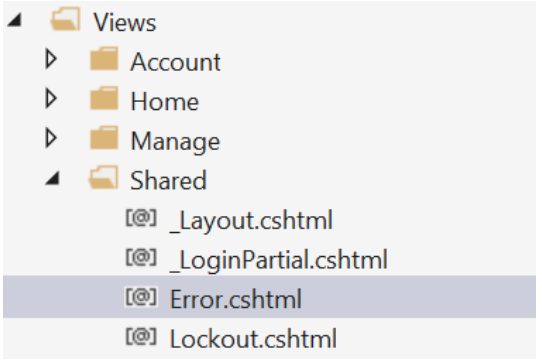
➔ Global.asax

➔ Filters registreren bij opstarten

```
0 references
public class MvcApplication : System.Web.HttpApplication
{
    0 references
    protected void Application_Start()
    {
        AreaRegistration.RegisterAllAreas();
        FilterConfig.RegisterGlobalFilters(GlobalFilters.Filters);
        RouteConfig.RegisterRoutes(RouteTable.Routes);
        BundleConfig.RegisterBundles(BundleTable.Bundles);
    }
}
```

Error Handling

➔ Locatie View error weergave




```
@model System.Web.Mvc.HandleErrorInfo

@{
    ViewBag.Title = "Error";
}

<h1 class="text-danger">Error.</h1>
<h2 class="text-danger">An error occurred while processing your request.</h2>
```

NEW MEDIA &
COMMUNICATION
TECHNOLOGY

ActionResults

ActionResult

- ActionResult
  - Abstracte Klasse
  - Default return value in controller
  - Veel specifieke klassen erven van ActionResult

ActionResult

➔ActionResult

Action Result	Helper Method	Description
ViewResult	View	Renders a view as a Web page.
PartialViewResult	PartialView	Renders a partial view, which defines a section of a view that can be rendered inside another view.
RedirectResult	Redirect	Redirects to another action method by using its URL.
RedirectToRouteResult	RedirectToAction RedirectToRoute	Redirects to another action method.
ContentResult	Content	Returns a user-defined content type.
JsonResult	Json	Returns a serialized JSON object.
JavaScriptResult	JavaScript	Returns a script that can be executed on the client.
FileResult	File	Returns binary output to write to the response.
EmptyResult	(None)	Represents a return value that is used if the action method must return a **null** result (void).

ActionResult

- ViewResult
  - Specifieke klasse die view zal terugkeren
  - Erft van ActionResult
  - Probeer vanaf nu ViewResult te gebruiken bij View teruggave

```
0 references
public ViewResult Index()
{
    return View();
}
```

ActionResult

- RedirectResult
  - Specifieke klasse die redirect zal uitvoeren
  - Erft van ActionResult

```
0 references
public RedirectResult About()
{
    return Redirect("http://www.howest.be");
}
```

ActionResult

- RedirectToRouteResult
  - Redirect naar andere ActionMethode
  - Erft van ActionResult

```
0 references
public RedirectToRouteResult Contact()
{
    return RedirectToAction("Index");
}
```

ActionResult

- JsonResult
  - Zal Json data terugkeren
  - Erft van ActionResult
  - Nu beter gebruik maken Web API

```
0 references
public JsonResult GetJson()
{
    var movies = new List<object>();

    movies.Add(new { Title = "Ghostbusters", Genre = "Comedy", Year = 1984 });
    movies.Add(new { Title = "Gone with Wind", Genre = "Drama", Year = 1939 });
    movies.Add(new { Title = "Star Wars", Genre = "Science Fiction", Year = 1977 });

    return Json(movies, JsonRequestBehavior.AllowGet);
}
```

ActionResult

- ➔ FileResult
  - ➔ Terugkeren van een file ➔ download
  - ➔ Relative path omzetnaar naar absoluut
    - ➔ Server.MapPath()
  - ➔ Erft van ActionResult

```
0 references
public FileResult GetFile()
{
    byte[] fileBytes = System.IO.File.ReadAllBytes(Server.MapPath(@"~/Week 1 - Theorie.pptx"));
    string fileName = "Week 1 - Theorie.pptx";
    return File(fileBytes, System.Net.Mime.MediaTypeNames.Application.Octet, fileName);
}
```

ActionResult

➔RedirectPermantent

➔RedirectToActionPermanent

➔RedirectToRoutePermanent

HTTP 301 (Moved Permanently) status code whereas Redirect will send an HTTP 302 status code.

NEW MEDIA &
COMMUNICATION
TECHNOLOGY

Caching in ASP.NET MVC

Caching

Verbeteren snelheid

Gebruikers willen niet wachten

Caching

- Verschillende mogelijkheden
  - OutputCache in ASP.NET MVC
  - Distributed Caches (Cloud omgevingen volgende semester)
    - Redis
    - Velocity
    - Memcache

Caching

- OutputCache attribuut
  - Plaatsen boven method
  - Duur van de cache instellen in seconden

```
[OutputCache(Duration = 25)]
0 references
public ViewResult GetData()
{
    return View(Data);
}
```

Caching

- Locatie Cache
  - Any (client/server/proxy)
  - Client (in de browser
  - Downstream (proxy)
  - Server
  - ServerAndClient (alles behalve proxy)
  - None (geen caching)
- Locaties zelf te bepalen via location property
  - Default ➔ Any location

Caching

- ➔ Wat met persoonlijke info ?
  - ➔ Nooit cachen op server
    - ➔ Andere gebruikers kunnen verkeerde info zien
  - ➔ Location property invullen
    - ➔ OutputCacheLocation.Client
    - ➔ NoStore => niet cachen in proxy servers

```
[OutputCache(Duration = 3600, VaryByParam = "none", Location = OutputCacheLocation.Client, NoStore = true)]
0 references
public string GetName()
{
    return "Hi " + User.Identity.Name;
}
```

Caching

- Wat met parameters ?
  - Attribuut VaryByParam
    - None == nooit cachen
    - Parameter naam == versie met parameter cachen

```
[OutputCache(Duration = 10, VaryByParam = "id")]
0 references
public ViewResult Details(int id)
{
    return View(Data.GetWine(id));
}
```

Caching

- Cache Rules in web.config
  - Makkelijk central te beheren
  - Één profile meerdere actions en controllers
  - Applicatie niet hercompileren
    - Enable/disable cache

Caching

➔ web.config in root

➔ <system.web>

```
<caching>
  <outputCacheSettings>
    <outputCacheProfiles>
      <add name="Cache30Sec" duration="30" varyByParam="none"/>
      <add name="Cache1uur" duration="3600" varyByParam="none"/>
    </outputCacheProfiles>
  </outputCacheSettings>
</caching>
```

Caching

➔ CacheProfile attribuut gebruiken op action

```
[OutputCache(CacheProfile = "Cache30Sec")]
0 references
public ActionResult Contact()
{
    ViewBag.Message = "Your contact page.";
    return View();
}
```

NEW MEDIA &
COMMUNICATION
TECHNOLOGY

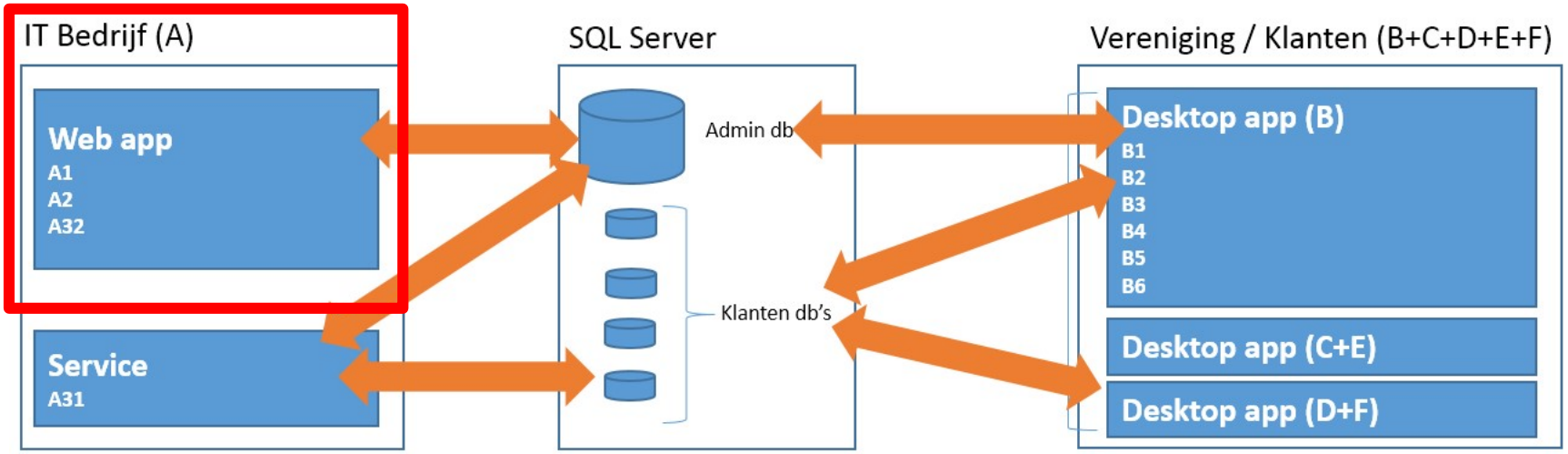
Project

Project

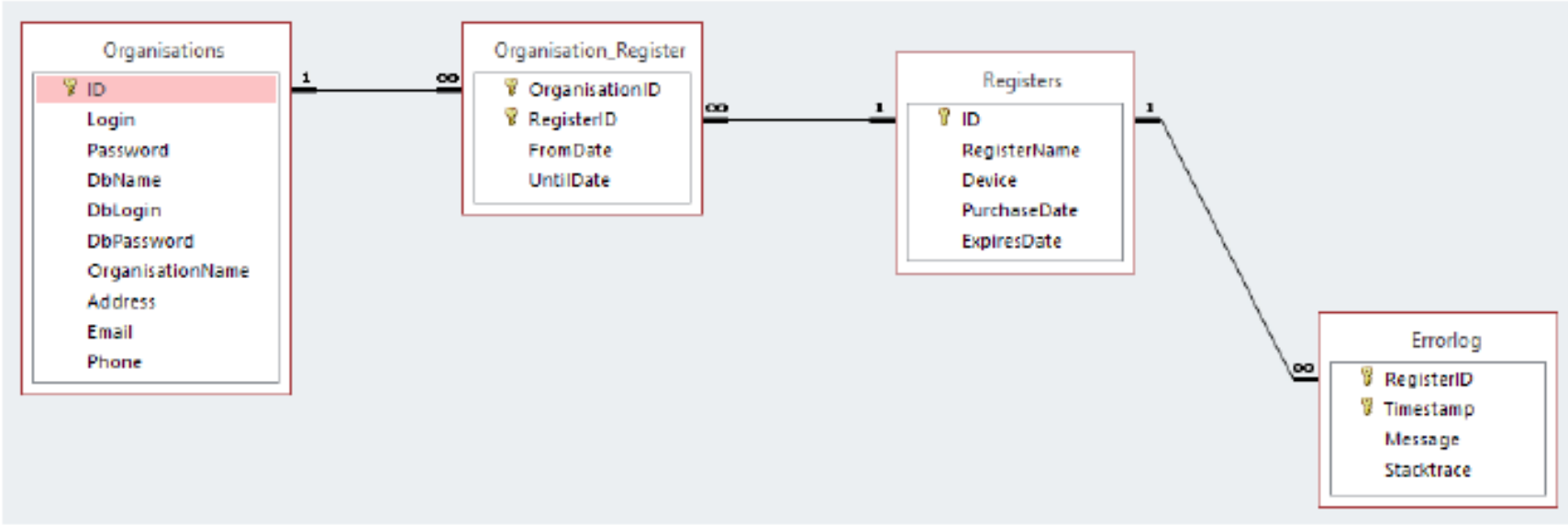
➔ 25% van de punten voor deze module

➔ Evaluatie in de examenperiode via presentatie

Project


IT Bedrijf (A)
Web app
A1
A2
A32
Service
A31
SQL Server
Admin db
Klanten db's
Vereniging / Klanten (B+C+D+E+F)
Desktop app (B)
B1
B2
B3
B4
B5
B6
Desktop app (C+E)
Desktop app (D+F)

Project


Organisations
ID
Login
Password
DbName
DbLogin
DbPassword
OrganisationName
Address
Email
Phone
1
∞
Organisation_Register
OrganisationID
RegisterID
FromDate
UntilDate
∞
1
Registers
ID
RegisterName
Device
PurchaseDate
ExpiresDate
1
∞
Errorlog
RegisterID
Timestamp
Message
Stacktrace

Project

IT bedrijf (A)

Beheer verenigingen (A1)

- Kan een nieuwe vereniging registreren met een login en een wachtwoord (A11)
- Kan een database genereren voor een nieuwe vereniging (A12)
- Kan basisinformatie over een vereniging invoeren (A13)
- Kan basisinformatie over een vereniging bewerken (A14)
- Kan basisinformatie over een vereniging weergeven (A15)
- Kan een lijst van verenigingen die het systeem gebruiken weergeven (A16)

Project

Beheer kassa’s (A2)

- Kan een nieuwe kassa toevoegen aan zijn assortiment (A21)
- Kan een kassa toekennen aan een vereniging (A22)
- Kan een kassa wijziging van vereniging (A23)
- Kan een overzicht van alle kassa’s opvragen per vereniging (A24)
- Kan een overzicht van alle beschikbare kassa’s opvragen (A25)

Project

Logging (A3)

- Kan een logboek van foutmeldingen en waarschuwingen van de kassa's opvragen (A32)