

Webbportfolio

Tina Durmén Blunt

tina.blunt@gaia.se

tinablunt.com

gaia
effektfabriken

*Vi kombinerar smart IT med
kreativ digital kommunikation.
Lite enklare. Lite skönare.*



Visual studio gratis 😊

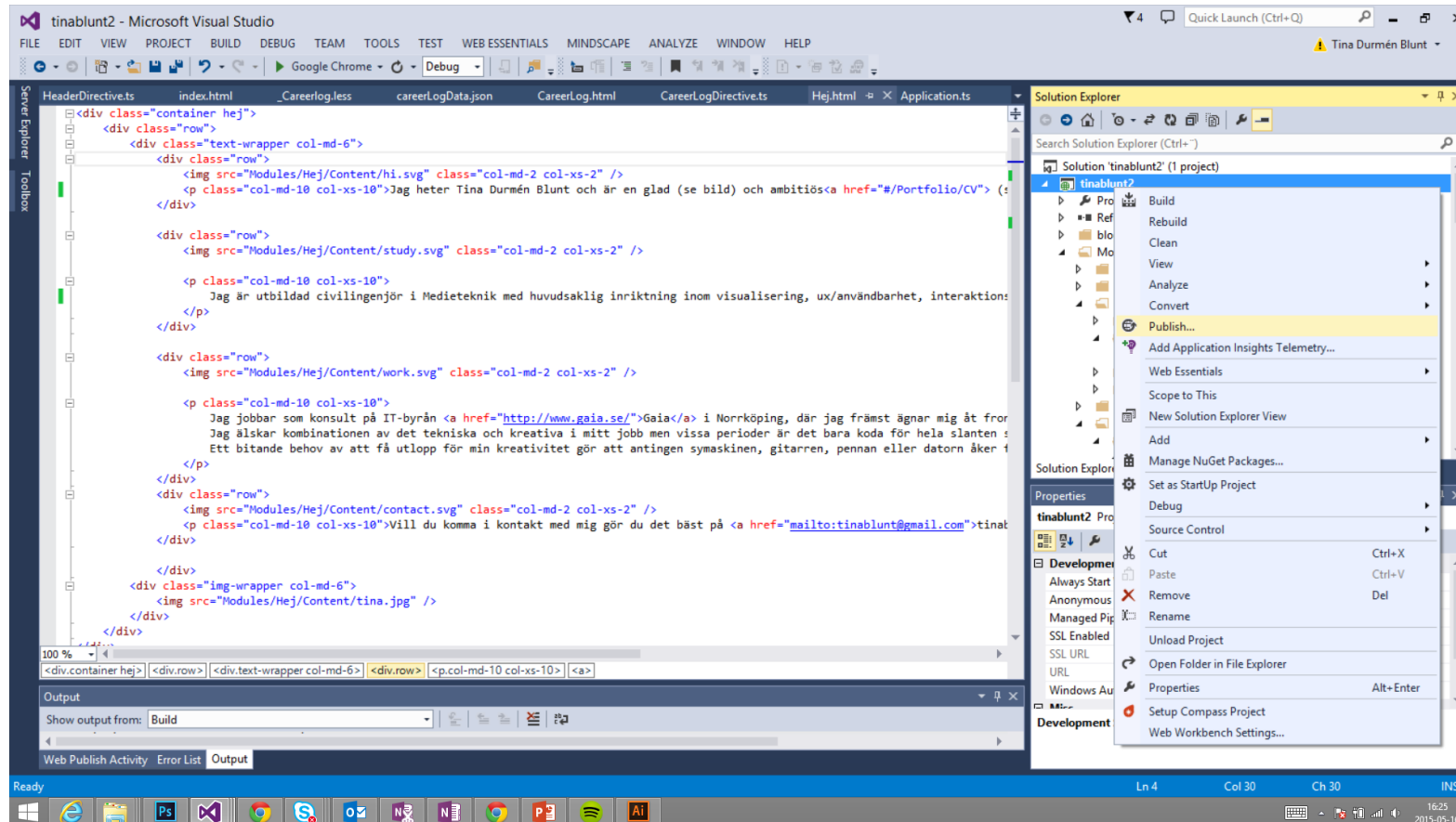
<https://www.visualstudio.com/en-us/products/free-developer-offers-vs.aspx>

Github

<http://michaelcrump.net/setting-up-github-to-work-with-visual-studio-2013-step-by-step/>

<https://github.com/>

Jag har min egen sida på webshotellet one.com. Där kan man ställa in ftp-lösenord osv som man sen använder när man publicerar från Visual Studio. Man kan även köpa domäner på one.com. Alternativ till webshotell är Azure eller Github som har ett visst antal gratis-sidor man kan sätta upp.



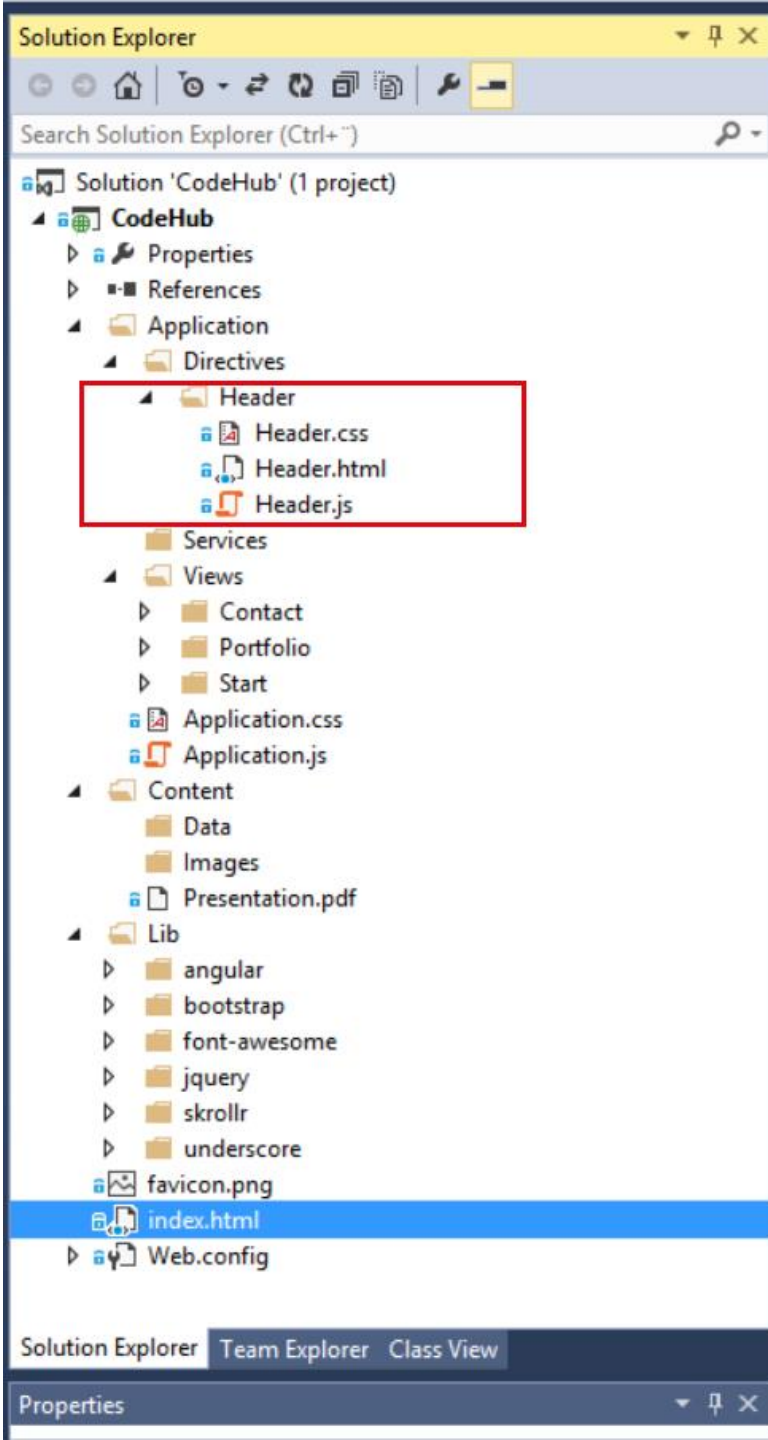
Kodskellettet

<https://github.com/tblunt/CodeHub>

Kodstruktur

Angular

Skrollr



Struktur

Vad ska man tänka på?

Det ska vara lätt att hitta i och förstå koden.

Delar ska vara så utseparerade det går för att man lätt ska kunna flytta, ta bort och återanvända dem.

Dynamisk webb med Angular

<https://angularjs.org/>

Angular

Controllers

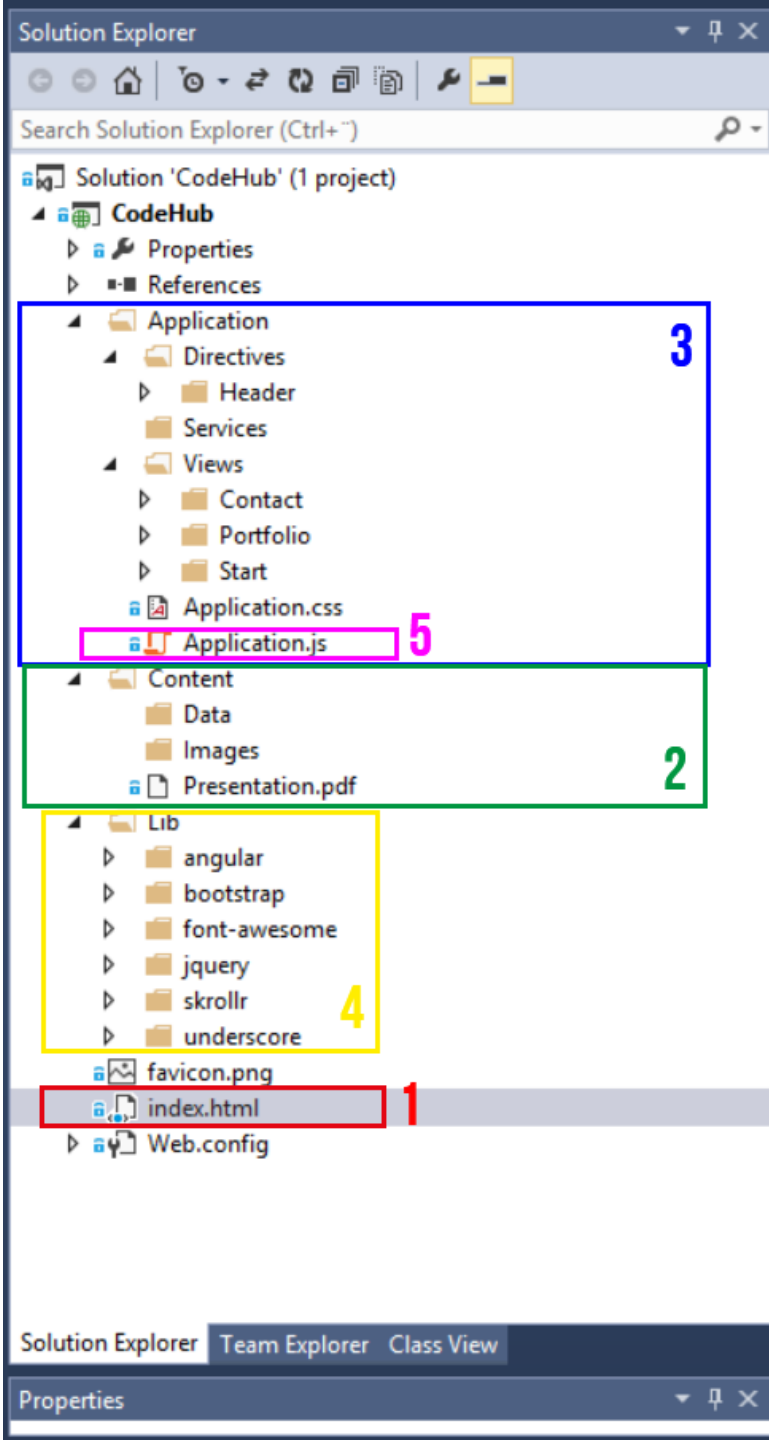
Kontrollerar allt som har med vyn att göra.

Directives

Egna html-element. Bra för att kunna återanvända och separera kod.

Services

Har hand om att hämta och bearbeta data.



1. index.html

Länkar till alla script och css. Innehåller topnavigeringen och div med ng-view som angular stoppar in innehåll i beroende på [url:en](#).

2. Content

Bilder, json-data och annat.

3. Application

Angular-modulen som bygger upp hela sidan. I vissa fall har man fler moduler med direktiv, controllers och services.

3. Application.js

Där modulen initialiseras samt där ng-route binder alla vyer till [url:en](#).

4. Lib

Alla externa skript.

5. Application.js

Modulen initieras och vyerna knyts till [url:en](#). Så fort man ska lägga till en ny vy så måste man lägga till den här.

```

<h1>List på några kurser</h1>
<div ng-repeat="item in courseData">
  <h3>{{item.title}}</h3>
  <p>{{item.description}}</p>
</div>

```

Ng-repeat (courseData.html)

Loopa över data-array som sitter på scopet (courseData.js)

```

<div class="nav">
  <div class="nav-item" ng-class="{ 'selected': selectedView == '/' }">
    <a href="#/">
      Start
      <span class="fa fa-angle-double-down"></span>
    </a>
  </div>
  <div class="nav-item" ng-class="{ 'selected': selectedView == '/Portfolio' }">
    <a href="#/Portfolio">
      Portfolio
      <span class="fa fa-angle-double-down"></span>
    </a>
  </div>
  <div class="nav-item" ng-class="{ 'selected': selectedView == '/Kontakt' }">
    <a href="#/Kontakt">
      Kontakt
      <span class="fa fa-angle-double-down"></span>
    </a>
  </div>
</div>

```

ng-class (header.html)

Sätter en klass beroende på specificerat villkor.

```

(function () {
  'use strict';

  //hämtar Application-modulen som redan är skapad i Application.js
  var application = angular.module('Application');

  //Binder direktivet till vår modul
  application.directive('header', ['$location', '$route', function ($location, $route) {

    function link($scope, $element, $attrs) {

      //vyn man befinner sig på ('/', '/Portfolio' eller '/Kontakt')
      $scope.selectedView = $location.path();

      //triggas då användaren klickar i headern
      $scope.$on('$routeChangeStart', function() {
        $scope.selectedView = $location.path();
      });

    }

    return {
      templateUrl: "Application/Directives/Header/Header.html",
      restrict: 'E',
      link: link
    };
  }]);
})();

```

templateUrl: binder html till scriptet

Restrict: vilken typ av direktiv det ska vara. I det här fallet gör vi ett helt eget element <header> men man kan även lägga till direktivet som ett attribut: <div header>

Link: Funktionene som körs när direktivet ritas ut.

Direktiv

Application.directive('header'

'header' blir namnet på direktivet <header>

Kamelbokstäver översätts i html till bindestreck t.ex:

'courseList' blir <course-list>

Parallax scrolling med skrollr.js

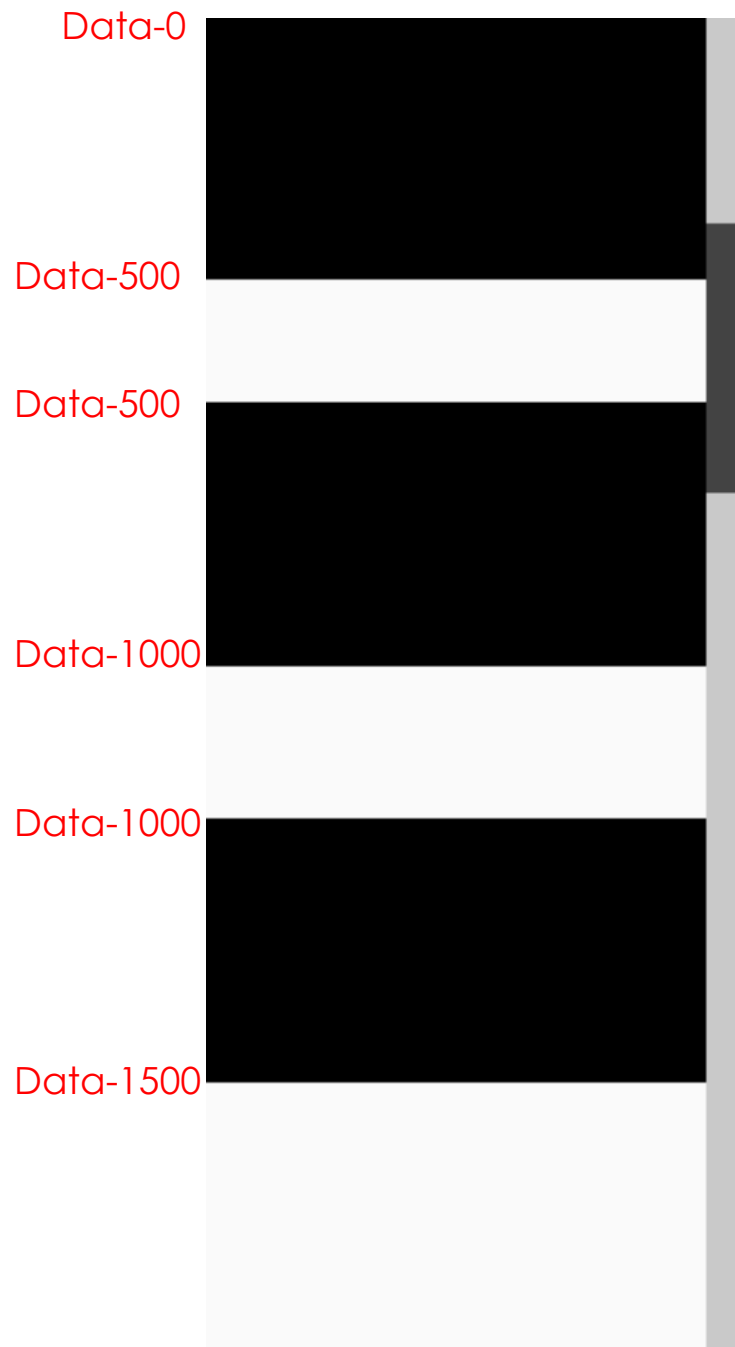
<http://prinzhorn.github.io/skrollr/>

Skrollr

Binder css-animationer till olika tidpunkter med hjälp av data-attribut på html-elementen. Ex:

```
<div data-200="opacity:0;" data-400="opacity:1;" ></div>
```

Detta betyder att när scrollen kommit ner 200px så börjar animationen. Så mellan 200 och 400 så kommer opaciteten gå från 0 till 1.



I PortfolioController.js lägger skriptet på skrollrs data-attribut på alla divs som har klassen 'view' (svarta rutorna).

`getKeyframes()`

Hämtar alla views. Varje vy få en keyframe-tid. Alltså när det är den vynos tur att börja animera. Keyframen räknas ut med alla vyernas höjder som ligger innan. Från bilden till vänster kommer keyframe se ut såhär:

`[0,500,1000,1500]`

`animateViews()`

Sätter på data-attributen på alla view-div med hjälp av keyframes-arrayen.

Animerar även menyknapparna till vänster med samma keyframevärden.

I Portfolio.html kan du sätta på egna animationer eller göra animationen längre på view-divsen. Sätter du inte attributen kommer vyn flyga vertikalt bara (se animateViews() där default-värdena sätts).

Before-enter

After-enter

Enter-duration

Before-leave

After-leave

Leave-duration

```
.....  
<div class="view" id="nummer2"  
  before-enter="opacity:0;"  
  after-enter="opacity:1;"  
  before-leave="opacity:1;"  
  after-leave="opacity:0;"  
  enter-duration="500"  
  leave-duration="500">  
  <div class="container">  
    <course-list></course-list>  
  </div>  
</div>
```