

Ministerul Educației Naționale și Cercetării Științifice

Universitatea OVIDIUS Constanța

Facultatea de Matematică și Informatică Specializarea Informatică

[NUME].

Lucrare de licența

Coordonator științific

**Lect. univ. Dr. [NUME]**

Absolvent

**[NUME]**

Constanta 2016

# Capitolul 1

## Motivație

.

## Introducere

.

## Tema proiectului

.

# Capitolul 2

## Tehnologii utilizate

### .NET

Aplicația web “Helpy” urmează modelul WebForms și este bazat pe platforma .NET de la Microsoft.

**.NET** (dotNET) este o întreagă platformă ce oferă soluții și servicii pentru Desktop, Web, Server, Sisteme Embeded (folosind .NET micro-framework sau .NET Compact-Framework) și Mobile. Platforma oferă de asemenea numeroase librării, limbaje de programare, modele de aplicații, baze de date. Aceasta este impărțită în două componente CLR (Common Language Runtime) și FCL (Framework Class Library).

**Common Language Runtime** (CLR) este practic o mașină virtuală ce execută aplicatii .NET printr-un proces numit “just-in-time compilation” care, practic transmite codul compilat în instrucțiuni mașina direct procesorului pentru al executa. CLR oferă de asemenea și managementul memoriei (împreuna cu Garbage Collector), managementul firelor de executie și al proceselor, securitate și tratarea excepțiilor. Acesta la un nivel jos implementează sistemul VES (Virtual Execution System).

**Framework Class System** (FCS) este o colecție de librării standard ce include funcționalități pentru soluții Desktop, Web/Server și Mobile. Câteva exemple principale:

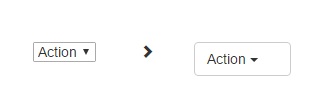
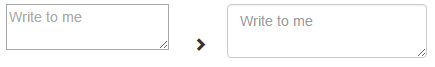
* **Interfață cu utilizatorul** folosind WinForms, WPF, Win32 sau ASP (Web)  
  WinForms este cel mai popular pe sistemele de tip desktop, acesta oferă o utilizare ușoară a tuturor componentelor și controalelor disponibile in Win32. Acestă librărie a fost implementată în NET Framework încă din versiunea 3.0 (Windows XP). Spre deosebire de aplicațiile de tip batch, care acestea așteaptă ca utilizatorul să facă o acțiune, WinForms funcționează pe baza evenimentelor, acestea declanșânduse doar la cerere, oferind funcționalitate similară cu cel al callback-urilor. Toate controalele oferă o funcționalitate minima de UI, de exemplu : poziție, culoare, font, text, mărime, etc.., dar și evenimente comune : click, MouseEnter, MouseLeave, Drag’n’Drop, etc..
* **DataAccess** folosind namespace-ul System.IO.  
  Conține funcții ce permit citirea/scrierea fișierelor sau operarea pe directoare, Cateva exemple :
  + Clasa statică **Directory** conține o colecție de metoda statice ce permit crearea directoarelor, mutarea lor sau parcurgerea lor și a subdirectoarelor
  + Clasa **DriveInfo** ce oferă informații despre partitiile sistemului (Spațiu liber, Format - NTFS sau FAT32, Tip - CD-ROM, Fix, ..)
  + Similar cu clasa **Directory**, .NET oferă și clasa statică **File** ce o conține metode pentru managementul strict al fișierelor.
* **Baze de date** folosind **ADO.NET (ActiveX Data Objects)**  
  Prezent încă din 3.0 (la fel ca WinForms), ADO.NET este o tehnologie ce oferă comunicarea și manipularea bazelor de date relaționale și non-relaționale. La nivel de concept acesta este împarțit în două entități : consumatori și furnizori  
   Consumatorii sunt aplicațiile ce au neveoie de date  
   Furnizorii sunt componentele software ce asigura datele consumatorilor
* **Aplicatii web** folosind **ASP.NET (Active Server Page)** utilizând modelul WebForms sau MVC.  
  ASP.NET este (similar cu PHP) un limbaj de scripting făcut pentru crearea dinamica de pagini web. ASP.NET include :
  + **Entity Framework** pentru implementarea ușoară a unui ORM (Object Relation Mapping) ce crează clase oglindă pentru tabele din baza de date și oferă suport pentru percurgere a datelor cu LINQ
  + **Razor** ce ușurează sintaxa folosind simbolul ‘=’ în loc de ‘<%= %>’
  + Controale de tip asp: ce pot fi rulate pe server
  + ASP.NET Handler ce implementează System.Web.IHttpHandler ce ajută la capturarea mesajelor non-HTTP.
  + ASP.NET AJAX pentru trimiterea de mesaje asincrone
  + ASP.NET SignalR ce face comunicarea dintre client și server prin sockeți foarte ușoară.
* **Criptografie și algoritmi** folosind namespace-ul System.Security.Cryprografy
* Expresii regulate și operații pe șiruri de caractere

#### Platforme alternative pentru .NET

* .NET Micro Framework - este o colecţie mult mai mică de librării, în jur de 70 de clase și 400 de metode, concepută pentru dispozitive cu memorie mică și performanțe slabe.
* .NET Core - varianta open source pentru .NET
* Mono - variantă de .NET ce funcționează pe mai multe platforme (Linux, Mac,Windows)

### Bootstrap

Bootstrap este un framework open source pentru design-ul front-end-ului ce contine clase, template-uri, font-uri, icoane și alte componente folosind clase **CSS** și **JavaScript** dezvoltat de Twiter original numit Twiter Blueprint. Clasele din Bootstrap sunt optimizate și facut și pentru design responsive, însemnând că aplicațiile web ce folosesc Bootstrap iși vor face layout-ul astfel încât site-ul va fi funcțional și atât pe rezoluții desktop dar și mobile. Bootstrap este compatibil cu majoritatea browsere-lor moderne (Google Chrome, Firefox, Internet Explorer, Microsoft Edge, Opera, Safari, etc..). Din versiunea 4.0 (alpha) a fost adaugat și suportul pentru Sass și Flexbox. Cateva componente oferite de Bootstrap folosind clasa form-control ar fi :

* Dropdown  
  
* Button  
  
* Textbox  
  
* Textarea  
  

Toate aceste componente se pot modifica și combina după bunul plac :



### jQuery

jQuery este o librărie gratuită Open-Source JavaScript folosită pentru crearea paginilor cu conținut dinamic fiind ce-a mai populară librarie de JavaScript utilizată în peste 14 milioane de site-uri.

Librăria este folosită în principal pentru navigare printre elementele DOM ale pagini folosind selectorul $(‘.selector’), dar aceasta are funcționalități multiple, câteva dintre acestea fiind :

* **Crearea de animații și animarea elementelor** folosind funcția:
  + .animate(properties[,duration] [, easing] [,completing]) - aceasta execută o animație utilizând proprietăți CSS.  
    $(“div.anim”).animate({“left”:”+=100”}, “fast”);
* **Crearea de efecte** folosind funcțiile
  + .fadeIn();
  + .fadeOut();
  + .slideIn();
  + .slideOut();
  + .toggle();
* **Capturarea evenimentelor**:
  + Evenimente de browser :
    - .resize();
    - .scroll();
  + Evenimente de pagina:
    - .load();
    - .ready();
    - .unload();
  + Evenimente pe formulare:
    - .submit();
    - .focums();
    - .select();
  + Evenimente de tastatura:
    - .keyDown();
    - .keyPress();
    - .KeyUp();
  + Evenimente de mous:
    - .click();
    - .mouseMove();
    - .mouseEnter();
    - .mouseLeave();
* **Trimiterea de mesaje AJAX ( Asynchronous JavaScript and XML)**$.ajax({  
   type: 'POST',  
   url: '/register.aspx',  
   data: {  
   fname : 'Vlad',  
   lname : 'Faluta',  
   },  
  }).done(function(msg) {  
   //success  
  }).fail(msg){  
   //fail  
  });
* **Manipularea DOM** folosind metodele :
  + .append();
  + .html();
  + .prepend();
  + .text();
* **Parsare JSON (JavaScript Object Notation)** folosind:
  + .serialize();
  + .deserialize();

jQuery este extins si mai mult prin intermediul plug-in-urilor ce acopera o foarte multe domenii, de la Helpere AJAX, Servicii Web și Liste dinamice până la manipularea cookie-urilor, ferestre modale și drag’n’drop, existând de asemenea și framework pentru testare automată de cod pentru detectarea erorilor și măsurarare performanțelor.Pentru mobile există versiunea **jQuery Mobile** optimizată pentru mobile ce implementează evenimente specifice mobile, de exeplu : touch, pitch, drag, zoom, long press, etc…

### MySql

MySql este un sistem de gestiune al bazelor de date relationale gratuit si Open-Source până a fost cumpărat de Oracle, in momentul în care a fost cumpărat un developer de la MySql (Michael "Monty" Widenius) a făcut fork la MySql și a creat MariaDb care este absolut gratuit în comparație cu MySql ce trebuie plătit dacă se folosește în aplicații comerciale. MySql își structurează datele astfel :

* MySql rulează pe un server,
* Server-ul poate avea mai multe baze de date,
* O baza de date poate avea mai multe tabele,
* Un table are coloane
* Fiecare coloana este carecterizată printr-un tip de data (int, text, date, etc..)

MySql foloseste două engine-uri de memorare:

* **MyIsam** - folosit înprincipal pentru View-uri (‘vederi’)
* **InnoDB** - folosit pentru Tabele.

Aventajele InnoDB față de MyIsam sunt :

* Posibilitatea creării tranzacțiilor
* Tranzactii ACID
* Indecsi
* Chei străaine
* Coloane virtuale (doar în MariaDB)

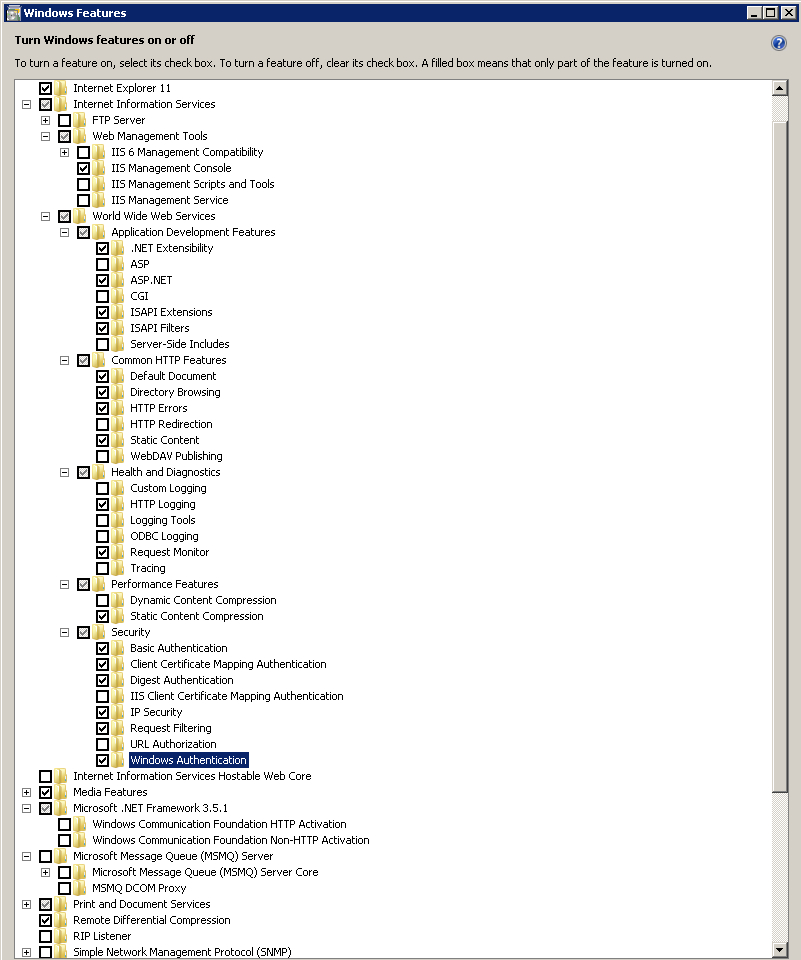
# Capitolul 3

## Instalarea aplicației

### Instalarea back-end-ului

Pentru a instala aplicația de back-end este nevoie de un PC sau server cu cel puțin următoarea configurație :

* **OS** : Windows 7 Professional / Windows Server 2008
* **Procesor** : Dual Core 1.4 GHz 64bit (Recomandat 2.0 GHz)
* **RAM** : 1GB (Recomandat 2 GB)
* **HDD** : 2GB (Recomandat 5GB)

Deoarece este o back-end-ul este o aplicație ASP.NET este necesar instalarea IIS (Internet Information Services) minim versiunea 7.0.

Pentru instalarea IIS-ului se intră în Control Panel > Programs and Features > Turn Windows features on or off, și se selectează ca în figura următoare. (*Figura 1*)

Mai departe se descarcă MySQL Server (care se găsește în directorul MySQL\_Server) iar la configurarea lui se mai, pe lângă utilizatorul root, se mai adaugă un utilizator, cu drepturi globale, cu numele wproject și o parolă aleasă de dumneavoastră.

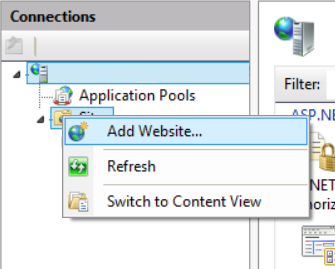
Mai departe folosind se intră în MySQL Command Line Client (se poate găsi în Start, sau în C:\Program Files\MySQL\MySQL Server 5.x\bin\mysql.exe). Odată pornită linia de comandă se va introduce parola de root, apoi se va crea o baza de date folosind comanda

CREATE DATABASE wproject

USE wproject

După crearea bazei de date se va folosi dump-ul inițial pentru crearea tabelelor folosind comanda

SOURCE c:\Users\User\Desktop\Exemplu\_cale\WProject\Install\dump.sql

După execuția dump-ului putem trece la configurarea server-ului, se intra in Start si se caută **Internet Information Services Manager** sau se apasă WIN + R iar în fereastra apăruta scriem **inetmgr**.În fereastra apărută deschidem arborele din stânga până vedem un elementul **Sites**, dăm click dreapta pe el, iar din meniul contextual apărut selectăm „Add Website...” (*figura 2*). În fereastra apărută la Site Name completăm **WProject.Dispatcher**, la Physical path selectăm c:\ iar în c creăm un nou director numit **wproject\_dispatcher**,după crearea lui îl selectă și apăsăm Ok**.**

La port vom pune 8002 și apăsăm Ok. Apoi vom copia conținutul directorului **Server Files** în directorul creat adineaori (*c:\wproject\_dispatcher*).

Intrăm în directorul c:\wproject\_dispatcher și vom deschide cu un editor text fișierul **Web.config**.

Urmând structura configuration > applicationSettings

Vom găsi următoarele chei :

Pentru cheia <setting name=”MysqlServer” ...> vom pune la value

localhost

Așa și pentru restul cheilor folosind modelul :

<setting name="MysqlServer" serializeAs="String">

    <value>localhost</value>

</setting>

<setting name="MysqlPort" serializeAs="String">

    <value>3306</value>

</setting>

<setting name="MysqlDatabase" serializeAs="String">

    <value>wproject</value>

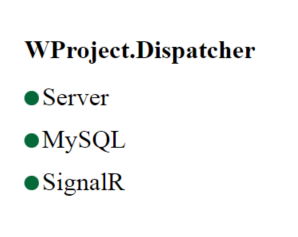
</setting>

<setting name="MysqlUser" serializeAs="String">

    <value>wproject</value>

</setting>

<setting name="MysqlPassword" serializeAs="String">

    <value>parola\_aleasă</value>

</setting>

Salvăm fișierul și putem intra într-un browser și introducem la adresă

http://localhost:8002

Iar în pagina apărută putem vedea starea conexiunilor, și eventual în caz de probleme ce putem face să le remediem.

Dacă toate bulinele sunt verzi înseamnă că serverul a fost instalat cu succes și putem trece la etapa următoare.

### Instalarea front-end-ului

Instalarea front-end-ului este mult mai simplă, din directorul Front End se deschide fișierul WProject\_Install.exe iar acesta va instala aplicația, și va face și o comandă rapidă de desktop. După instalare se intră în WProject (din icoana de pe desktop sau din meniul de Start) și va apărea o fereastra care va spune că clientul nu este configurat, vi se va cere să introduceți adresa dispecerului. Dacă aplicația rulează pe același PC cu server-ul este îndeajuns sa puneți

localhost

În caz că aplicația rulează pe un alt calculator este necesar să puneți adresa/IP-ul server-ului, ex:

172.16.1.103

După ce ați completat adresa apăsați Ok și așteptați până se verifică conexiunea, dacă verificarea s-a încheiat cu succes vă puteți conecta cu

Nume **admin**

Parola **wproject**

## Intrarea în aplicație

În caz că atunci când se intră în aplicație și nu există proiecte disponibile, dacă utilizatorul conectat este administrator, acesta va fi redirecționat către o secțiune de inițiere de aplicație ce conține 3 pași, pe toată perioada configurării aplicația va fi inutilizabilă de orice fel de utilizator.

### Crearea utilizatorilor

În primul rând va fi configurat utilizatorul **admin**. El va fi obligat să își introducă un E-Mail, să își introducă o parola (diferită de wproject), opțional să își schimbe numele.

Apoi va introduce utilizatorii – nume și email – aceștia primind pe mail un link de setare a parolei. Opțional le va putea seta drepturile de acces.

Mai departe vor fi create grupurile (acesta fiind un pas opțional) cu drepturile de acces pe fiecare grup și utilizatorii ce fac parte din grupurile respective.

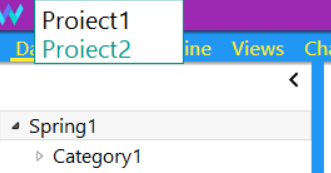
### Crearea proiectelor

Cel de al doilea pas este configurarea proiectelor. Fiecare utilizator ca să poată folosi aplicația trebuie să fie inclus în cel puțin un proiect. Astfel încât administratorul va crea proiectele, le va asigna celorlalți administratori și va include utilizatorii în ele. Iterațiile și categoriile se vor face mai târziu de fiecare administrator din modulul de administrare.

### Configurarea aplicației

Ultimele setări [TODO DE COMPLETAT]

## Utilizare aplicației

Puteți observa în partea de sus a aplicației proiectul curent în care lucrați. Dacă v-ați conectat cu un utilizator ce are acces la mai multe proiecte, sau un administrator ce deține mai multe proiecte puteți chiar modifica proiectul curent folosindu-vă de control-ul de tip ComboBox. 

### Dashboard

Dashboard-ul reprezintă metoda principală și ce-a mai ușoară de lucru cu task-uri și backlog-uri.

În dreapta se afla Iterațiile și categoriile ce pot fi ascunse, iar în partea dreapta este tabelul cu task-uri, grupate pe backlog-uri și aliniate în funcție de starea lor (To do, In Progress, Done), cele Removed nu mai apar în Dashboard, dar pot fi urmărite în Admin. Culoarea de fundal al task-urilor sunt de două tipuri, gri sau galbene, cele galbene sunt cele atribuite utilizatorului conectat, iar cele gri altor utilizatori sau task-urile fără utilizator.

Starea task-urilor se poate seta doar trăgând de el în starea dorita (drag-n-drop) și i se poate modifica utilizatorul din controlul de tip ComboBox din partea stângă jos a fiecărui task (bineînțeles dacă utilizatorul are drept).Atunci când se dă click pe un task apare un formular de modificare avansată a task-ului respectiv. Dacă se dorește adăugarea unui noi task, trebuie deschis backlog-ul în care dorim să adăugăm task-ul și atunci când punem mouse-ul în zona de to-do a backlog-ului va apărea jos un buton „plus”, atunci când îl vom apăsa va apărea fereastra de adăugare de task.

La fel și în cazul backlog-urilor putem de asemenea seta rapid utilizatorul ce deține backlog-ul și starea acestuia din controlul de tip ComboBox din fiecare backlog, atunci când acesta este deschis, când este închis se poate observa sub titlul cui aparține backlog-ul, iar in coloanele To do, In Progress și respectiv Done câte task-uri sunt în starea respectivă, de asemenea și timpul estimat în total al task-urilor din starea respectivă. Pentru a adăuga un backlog se folosește butonul de „plus” din antetul task-urilor.

De asemenea deasupra antetului se mai afla o serie de butoane și texte, de la stânga la dreapta :

* Butonul de filtrare/căutare – filtrează task-urile după utilizator/stare sau caută după titlu și conținut.
* Timpul limită pentru iterația respectiva împreună cu un graf al task-urilor.
* Butonul de ajutor, afișează manualul aplicației cu căutare.
* Butonul de setări.
* Butonul de full-screen – folosește întreaga suprafață a ferestrei doar pentru task-uri.

### Timeline

O altă posibilitate de a vizualiza task-urile este în modul Timeline, care practic este aranjarea task-urilor pe ore și durată. După cum se poate observa mai jos se afișează toate task-urile programate în ziua curentă pe perioadă de lucru definită.

Intervalele orare pot fi mutate, pur si simplu trăgând de task la fel și perioada estimată trăgând de marginea din dreapta. În momentul în care un task este trecut pe Done acesta va apărea în Timeline cu ora și minutul exact când a fost închis, în caz că taskul este încă In Progress sau în To do intervalul va fi calculat în funcție de timpul estimat introdus.

Task-urile fără timp de început sau fără timp estimativ vor apărea totuși în Timeline dar în marginea dreapta cu roșu, iar cele din dreapta cu galben sunt task-urile ce ies din dimensiunea controlului, făcând scroll vor apărea. Task-urile Done vor fi marcate și cu un „check” în marginea dreaptă pentru o observare mai ușoară.

La fel ca la Dashboard se poate face filtrarea pe utilizatori dar în plus se pot afișa și task-urile programate pe zilele următoare. Pentru adăuga un task se folosește butonul din colțul de stânga-jos, și exact ca la Dashboard va apărea un formular de adăugare de task.

Mai jos după task-urile atribuite utilizatorului curent se pot vedea task-urile și altor utilizatori. În caz ca panoul de task-uri pe linie vor apărea un model cu toate task-urile și numărul lor împreună cu timpul estimat rămas.

Acțiunile rapide din Timeline se face dintr-un meniu contextual pe fiecare task. Din meniul contextual aveți posibilitatea să :

* Modificați starea task-ului
* Modificați utilizatorul
* Modificați timpul estimat rămas
* Copiați id-ul și titlul task-ului în clipboard