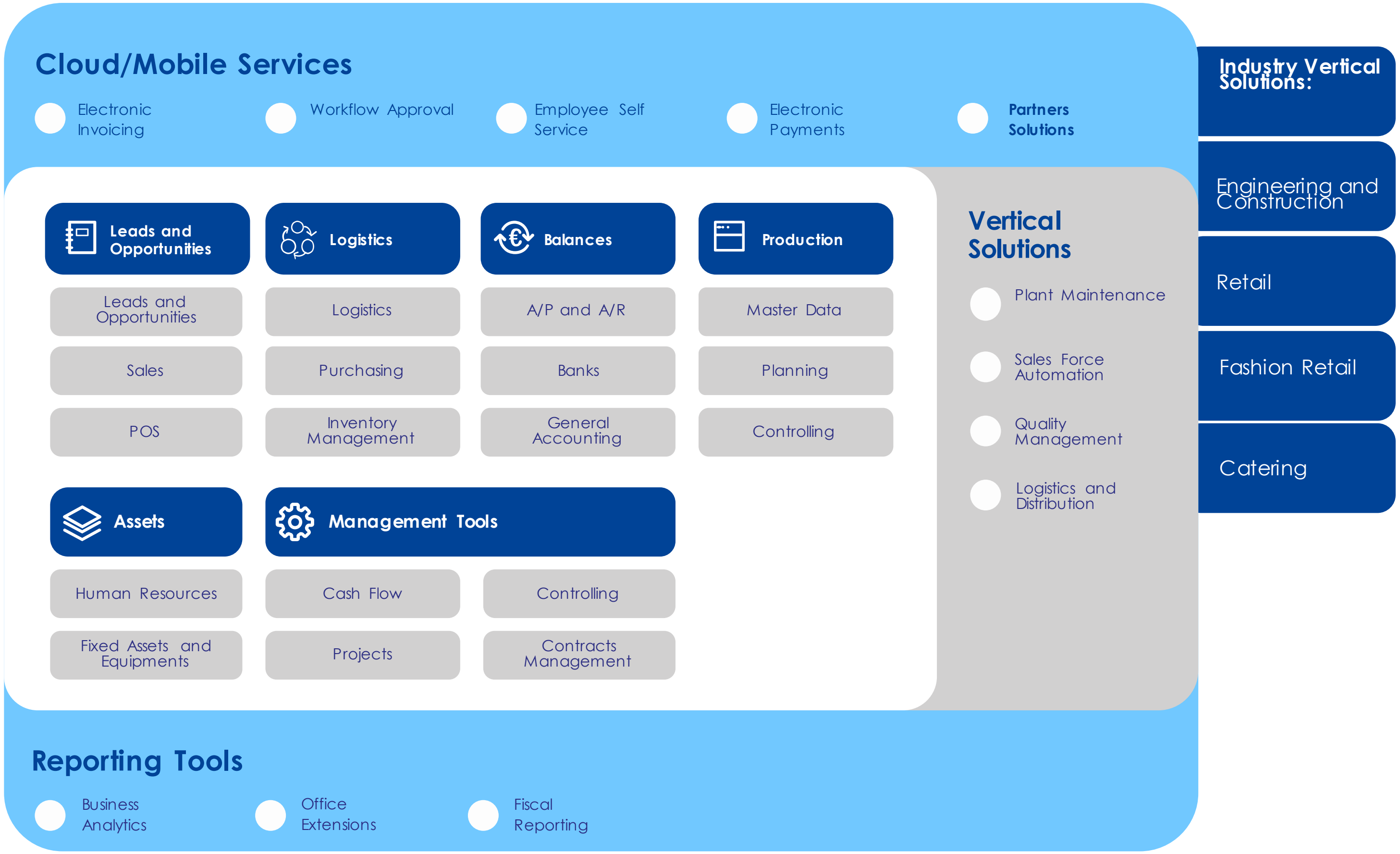
Universidade do Porto
FEUP Faculdade de Engenharia

Sistemas de Informação
Information Systems

Team Projects/Integration with ERP Primavera

Primavera Solutions



Cloud/Mobile Services

- Electronic Invoicing
- Workflow Approval
- Employee Self Service
- Electronic Payments
- **Partners Solutions**

Leads and Opportunities
- Leads and Opportunities
- Sales
- POS

Logistics
- Logistics
- Purchasing
- Inventory Management

Balances
- A/P and A/R
- Banks
- General Accounting

Production
- Master Data
- Planning
- Controlling

Assets
- Human Resources
- Fixed Assets and Equipments

Management Tools
- Cash Flow
- Projects
- Controlling
- Contracts Management

Vertical Solutions

- Plant Maintenance
- Sales Force Automation
- Quality Management
- Logistics and Distribution

Industry Vertical Solutions:
- Engineering and Construction
- Retail
- Fashion Retail
- Catering

Reporting Tools

- Business Analytics
- Office Extensions
- Fiscal Reporting

Team Projects

Evaluation

- **60% of the global evaluation of the Information System lecture**
  - Team project involving the application integration with PRIMAVERA ERP
- **Team Project detail**
  - *12,5% - Functional specification*
  - *15% - Interoperability (Test w/ ERP)*
  - *15% - Prototipe Presentation*
  - *(50%) Final Presentation*
    - 40% - Apresentação final (15 Minutos + 5 de perguntas e respostas)
    - 10% - Relatório que acompanha a apresentação final (máximo de 20 páginas - tamanho mínimo da letra: Times-New-Roman 12)
  - *7.5% Weekly Report (https://docs.google.com/forms/d/1CobCze3KWAtxptXG5neVzWO7OjW3m1I56cefCYAB44A/viewform)*
    - Weekly activity report (we will consider only the last weekly entry until Friday 23h59)

Team Projects Plan & Milestones

	09-10-2015	16-10-2015	23-10-2015	30-10-2015	13-11-2015
Week	**4**	**5**	**6**	**7**	**8**
Summary	Workshop VM/IIntegration. DEMO. Support on the adjustment of the students Mockup and definition of scope of the team project and of the specifications	Project Tracking and Support	Project Tracking and Support	Project Tracking and Support	Project Tracking and Support
Student Deliverables	**Functional Specifications** Information Architecture and Final Project Plan - Gantt with tasks (list of features, alignment to the structure and to the functional module and resources)		**Interoperability** : WS layer Demo: Model, Controller + API C# Primavera + ERP validation Primavera + Documentation		
Evaluation	**15%**	Continuous	**15%**	Continuous	Continuous

Team Projects Plan & Milestones

Universidade do Porto
FEUP Faculdade de Engenharia

	20-11-2015	27-11-2015	04-12-2015	11-12-2015	18-12-2015
Week	9	10	11	12	13
Summary	Project Tracking and Support	Project Tracking and Support	Project Tracking and Support	Project Tracking and Support	Evaluation
Student Deliverables	**Prototype Presentation** (with views/forms)				**Final Presentation and Report**
Evaluation	15%	Continuous	Continuous	Continuous	50%

Weekly Report	5%
Total	100%

Team Projects Themes

PID	**P1**
Project	**Online/Web Store**
Overview	Develop an Online store that allows to the customer, besides the usual shopping cart and purchase order registration, access the product catalog, inquire inventory by wharehouse/store location, and other functionalities. The user should also inquire their order status.
ERP - Extensibility Objectives	Create an External Auth layer to create customer, stock inquire, create orders, get order status (table "cabecdocstatus" on ERP).
ERP - Basic Usage Objectives	Proper Master Data definition (Wharehouses/Stores, Items, Item Categories, Customers, ...) Item reservations (stock)
Information/Funcionalities Reference Sources/Examples	-Amazon.com, Fnac.pt, Edições Piaget
Remarks	The Primavera's ERP does not have document status for the OLStore purpose. "cabecdocstatus" linked tables must be used to inquire the order status. The suggested status should be interpreted as so.

Team Projects Themes

PID	**P2**
Project	**Sales Order Picking**
Overview	Create a Web App that allows the user to make a picking process of a sales order, generating a delivery note and manage the inventory
ERP - Extensibility Objectives	Create an External layer to authenticate the user, create a picking wave, suggest routes, read inventory, register exceptions on inventory, ... Create ERP user tables to register the picking lists, ...
ERP - Basic Usage Objectives	- Proper Master Data definition (Wharehouses/Stores, Items, Item Categories, Customers, ... - Get Sales Orders, Create and adjust inventory/stock - Create warehouses/locations, ...
Information/Funcionalities Reference Sources/Examples	- Microsoft Dynamics AX (https://www.youtube.com/watch?v=hZOnccfP3nM) - SAP ERP, ...
Remarks	Primavera's ERP does not have WarehouseManagement capabilities as many other ones. The picking process can be complex. Picking Waves, Inventory Replenishment algorithms, Routes should be addressed.

Team Projects Themes

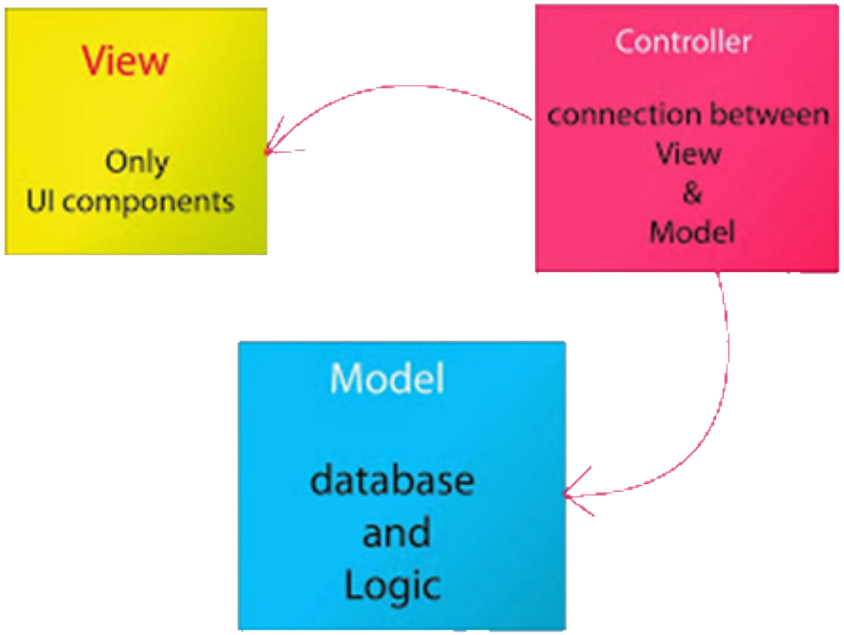
PID	P3
Project	**360º Company Dashboard**
Overview	Develop a Web app to provide to a complete information of the company status, regarding sales, purchases, inventory, accounts receivable, accounts payable ..., using a dashboard to provide high level/graphical information with drill down functionalities
ERP - Extensibility Objectives	Create an External Authorization layer , get company information related to sales, accounting, inventory, ...
ERP - Basic Usage Objectives	- Proper use of master data, create customers, ... - Register Sales Orders, Delivery Notes, Invoices - Register basic accounting transactions - Register payments and receivables
Information/Funcionalities Reference Sources/Examples	- SAP Business Objects, QlikView, ...
Remarks	This project is more complex than it looks, inspite of not having inbound ERP transactions and postings. The proper usage of the ERP and a complete data insertion should be addressed.

Team Projects Themes

PID	P4
Project	**Sales Force Automation**
Overview	Develop a Web App to manage the agenda of the sales representative, regarding target customers, register customer visits, register sales orders, customer profiles and status, ...
ERP - Extensibility Objectives	Create an External Authorization layer , create sales rep, create customer groups/targets, create visiting routes Register Sales Orders, Create Leads/Prospects Register summaries of customer visits and needs Manage product catalogues, ...
ERP - Basic Usage Objectives	- Proper Master Data definition (Wharehouses/Stores, Items, Item Categories, Customers, ... - Get Sales Orders, Create and adjust inventory/stock - Reserve stock
Information/Funcionalities Reference Sources/Examples	- SAP CRM (Sales), Microsoft Dynamics CRM, Oracle Siebel, Vtiger, SugarCRM.
Remarks	This is also a very valuable project and market demanded. Mobile SFA is also an alternative (iOS, Android) for the Web App.

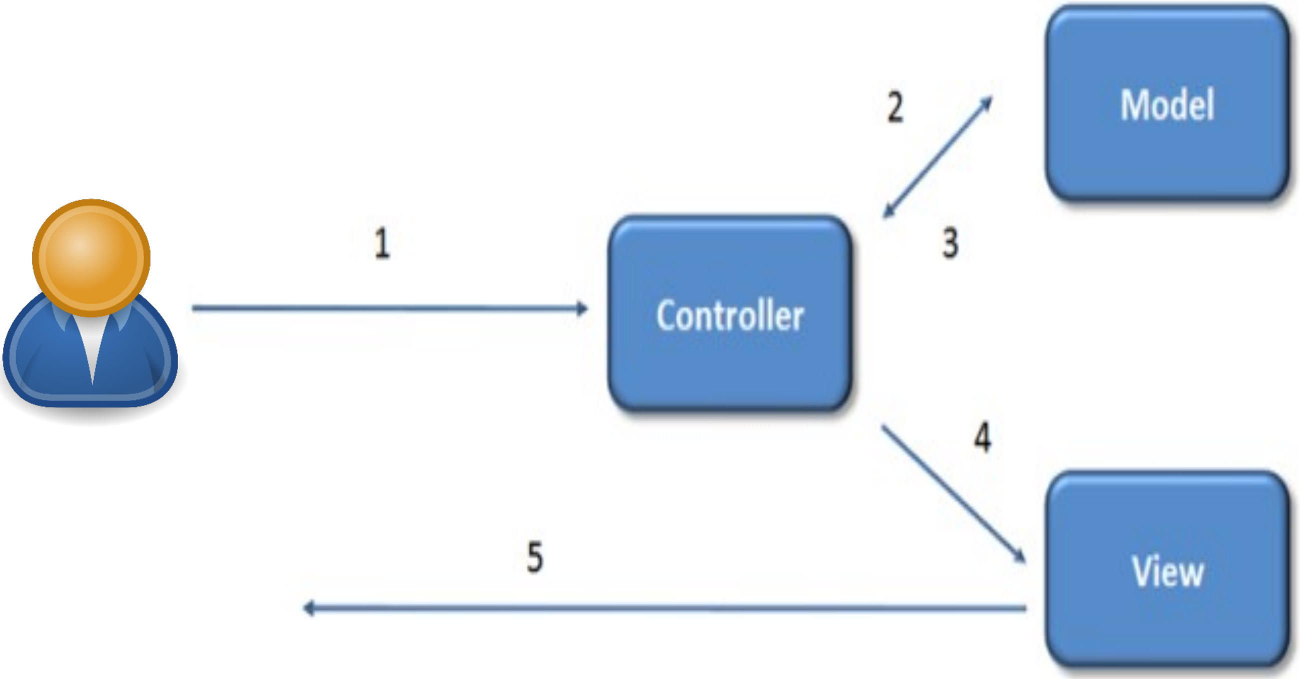
MVC (Model View Controller)

- Model (Data):
  - Application Data
  - Logic and Functions
  - Business rules
- View (UI):
  - User interface
  - Presentation layer
- Controller (Code):
  - User action
  - Control Flow
  - Communications between users (through the view layer) and the model



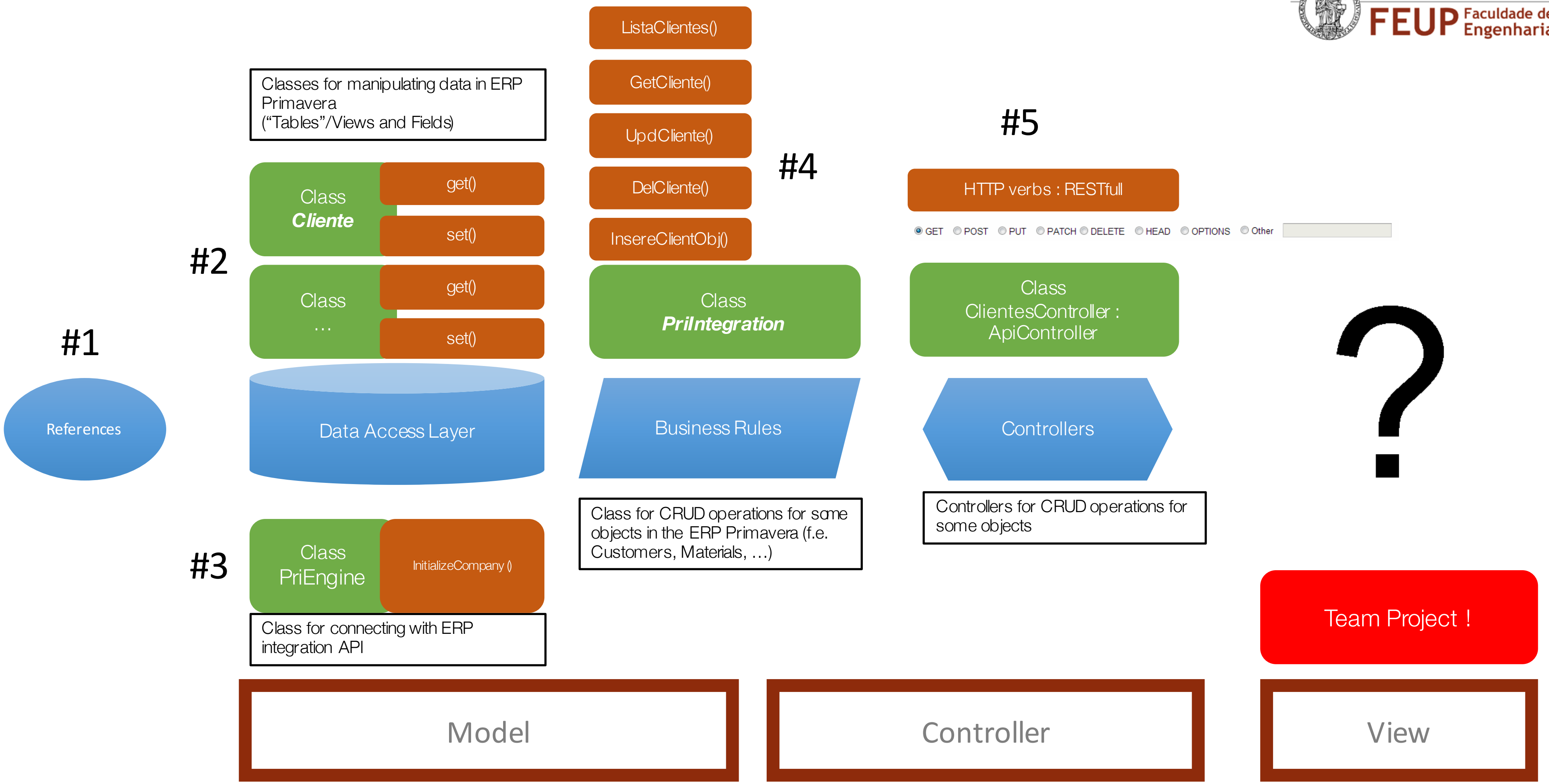
Interaction with MVC

1. The request comes from the client and hits the Controller.
2. The Controller calls the Model in order to perform some "business" operations.
3. The Model returns the results of the operations back to the Controller.
4. The Controller decides which View needs to be rendered and sends it the data that must be rendered.
5. Finally the View renders the output and sends the response back to the client.



Explanation “FirstRest” / “HelloPrimavera” Project C#

Universidade do Porto
FEUP Faculdade de Engenharia



#1

References

#2

Classes for manipulating data in ERP Primavera
(“Tables”/Views and Fields)

Class *Cliente*

get()

set()

Class …

get()

set()

Data Access Layer

#3

Class PriEngine

InitializeCompany ()

Class for connecting with ERP integration API

#4

ListaClientes()

GetCliente()

UpdCliente()

DelCliente()

InsereClientObj()

Class *PriIntegration*

Business Rules

Class for CRUD operations for some objects in the ERP Primavera (f.e. Customers, Materials, …)

#5

HTTP verbs : RESTfull

● GET ○ POST ○ PUT ○ PATCH ○ DELETE ○ HEAD ○ OPTIONS ○ Other

Class ClientesController : ApiController

Controllers

Controllers for CRUD operations for some objects

?

Team Project !

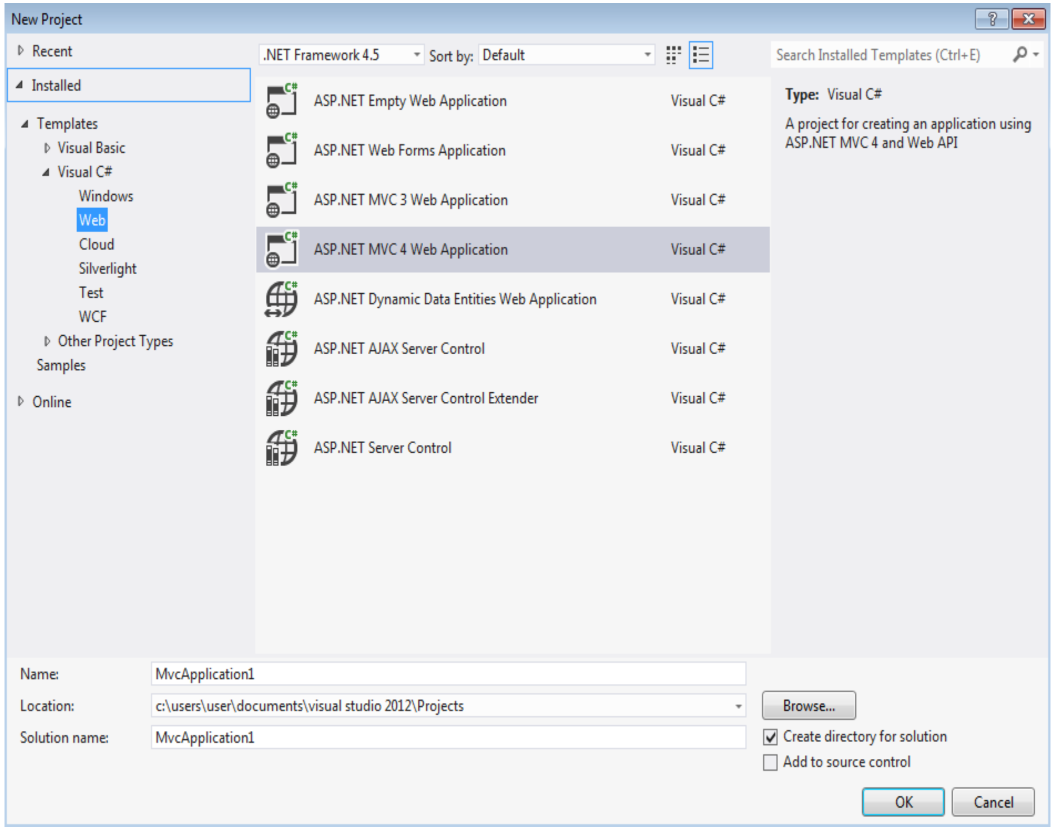
Model

Controller

View

Visual Studio 2013 Express

New Project



New Project

- Recent
- Installed
  - Templates
    - Visual Basic
    - Visual C#
      - Windows
      - Web
      - Cloud
      - Silverlight
      - Test
      - WCF
    - Other Project Types
  - Samples
- Online

.NET Framework 4.5 Sort by: Default

Search Installed Templates (Ctrl+E)

ASP.NET Empty Web Application	Visual C#
ASP.NET Web Forms Application	Visual C#
ASP.NET MVC 3 Web Application	Visual C#
ASP.NET MVC 4 Web Application	Visual C#
ASP.NET Dynamic Data Entities Web Application	Visual C#
ASP.NET AJAX Server Control	Visual C#
ASP.NET AJAX Server Control Extender	Visual C#
ASP.NET Server Control	Visual C#

Type: Visual C#

A project for creating an application using ASP.NET MVC 4 and Web API

Name: MvcApplication1

Location: c:\users\user\documents\visual studio 2012\Projects

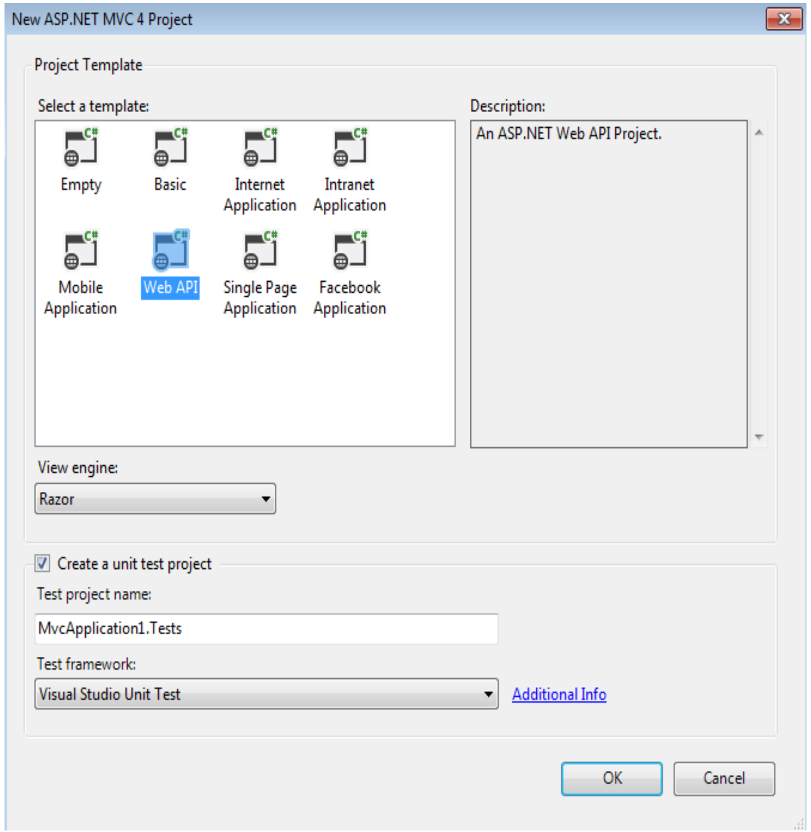
Solution name: MvcApplication1

☑ Create directory for solution

☐ Add to source control

Visual Studio 2013 Express

Project Template



New ASP.NET MVC 4 Project

Project Template

Select a template:

Description: An ASP.NET Web API Project.

View engine: Razor

☑ Create a unit test project

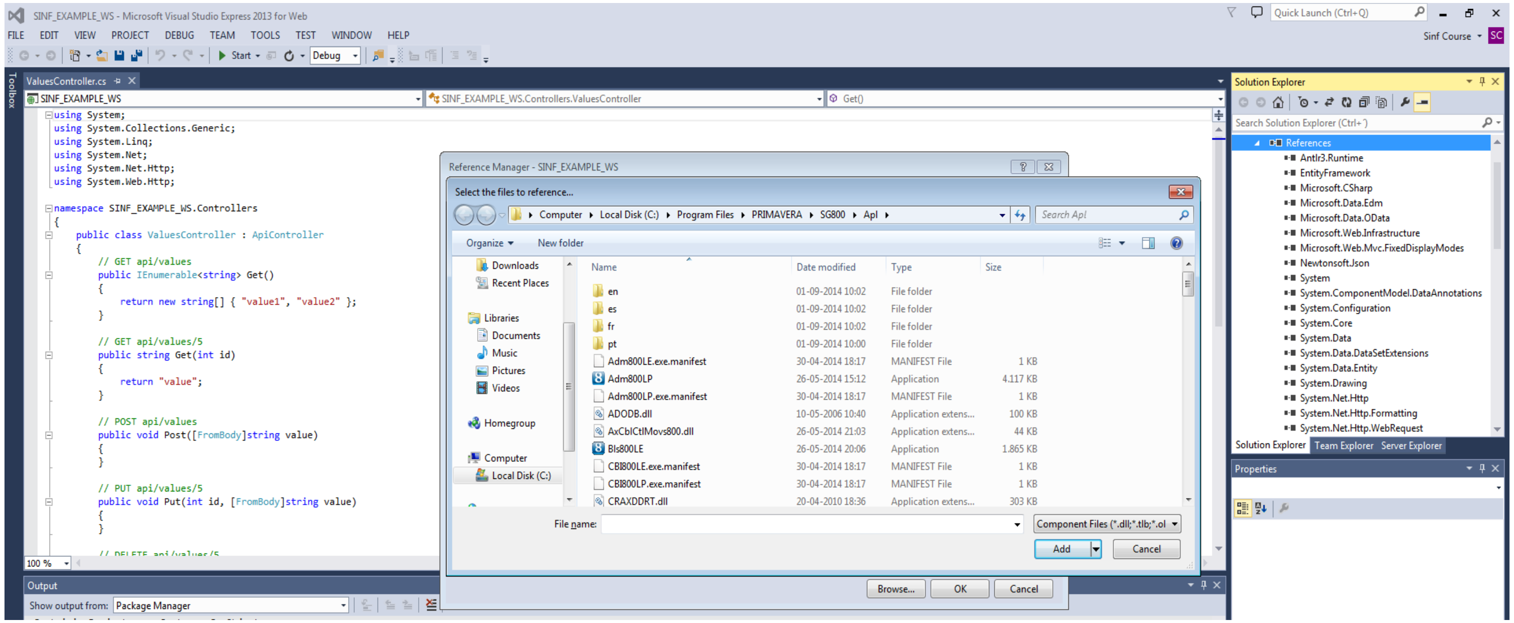
Test project name: MvcApplication1.Tests

Test framework: Visual Studio Unit Test

Visual Studio 2013 Express

Universidade do Porto
FEUP Faculdade de Engenharia

#1 Add Primavera references



SINF_EXAMPLE_WS - Microsoft Visual Studio Express 2013 for Web

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Net;
using System.Net.Http;
using System.Web.Http;

namespace SINF_EXAMPLE_WS.Controllers
{
    public class ValuesController : ApiController
    {
        // GET api/values
        public IEnumerable<string> Get()
        {
            return new string[] { "value1", "value2" };
        }

        // GET api/values/5
        public string Get(int id)
        {
            return "value";
        }

        // POST api/values
        public void Post([FromBody]string value)
        {
        }

        // PUT api/values/5
        public void Put(int id, [FromBody]string value)
        {
        }

        // DELETE api/values/5
```

Show output from: Package Manager

Reference Manager - SINF_EXAMPLE_WS

Select the files to reference...

Computer ▸ Local Disk (C:) ▸ Program Files ▸ PRIMAVERA ▸ SG800 ▸ Apl ▸

Name	Date modified	Type	Size
en	01-09-2014 10:02	File folder	
es	01-09-2014 10:02	File folder	
fr	01-09-2014 10:02	File folder	
pt	01-09-2014 10:00	File folder	
Adm800LE.exe.manifest	30-04-2014 18:17	MANIFEST File	1 KB
Adm800LP	26-05-2014 15:12	Application	4.117 KB
Adm800LP.exe.manifest	30-04-2014 18:17	MANIFEST File	1 KB
ADODB.dll	10-05-2006 10:40	Application extens...	100 KB
AxCblCtlMovs800.dll	26-05-2014 21:03	Application extens...	44 KB
Bls800LE	26-05-2014 20:06	Application	1.865 KB
CBl800LE.exe.manifest	30-04-2014 18:17	MANIFEST File	1 KB
CBl800LP.exe.manifest	30-04-2014 18:17	MANIFEST File	1 KB
CRAXDDRT.dll	20-04-2010 18:36	Application extens...	303 KB

File name:

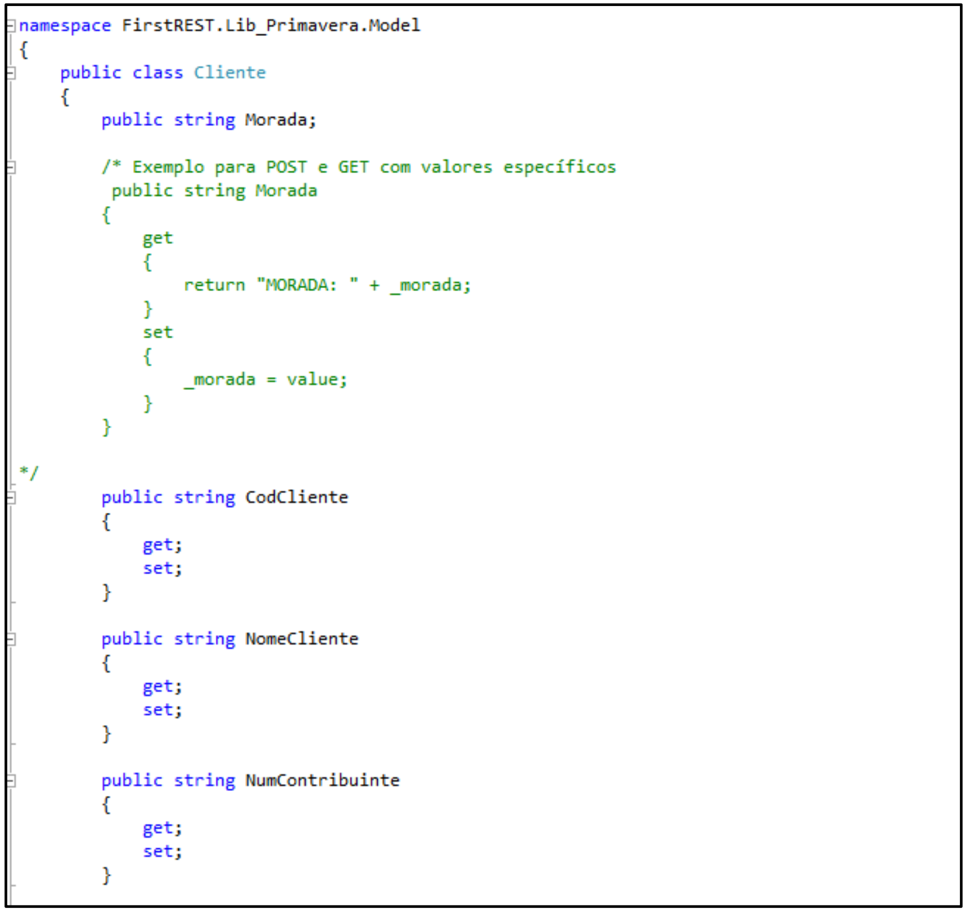
Component Files (*.dll;*.tlb;*.ol

- References
  - Antlr3.Runtime
  - EntityFramework
  - Microsoft.CSharp
  - Microsoft.Data.Edm
  - Microsoft.Data.OData
  - Microsoft.Web.Infrastructure
  - Microsoft.Web.Mvc.FixedDisplayModes
  - Newtonsoft.Json
  - System
  - System.ComponentModel.DataAnnotations
  - System.Configuration
  - System.Core
  - System.Data
  - System.Data.DataSetExtensions
  - System.Data.Entity
  - System.Drawing
  - System.Net.Http
  - System.Net.Http.Formatting
  - System.Net.Http.WebRequest

Visual Studio 2013 Express

#2 Model Definition

Cliente

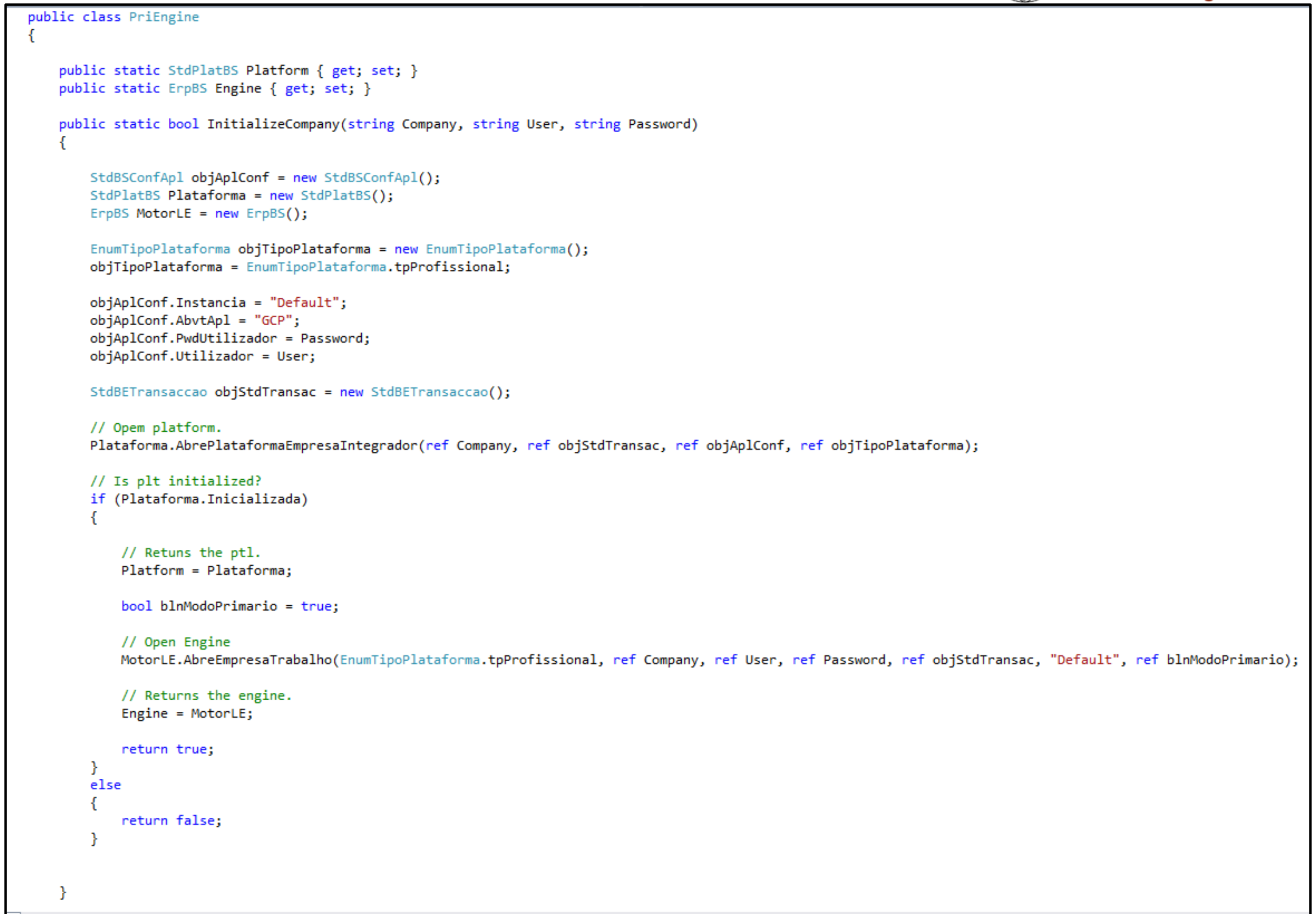


```
namespace FirstREST.Lib_Primavera.Model
{
    public class Cliente
    {
        public string Morada;

        /* Exemplo para POST e GET com valores específicos
         public string Morada
        {
            get
            {
                return "MORADA: " + _morada;
            }
            set
            {
                _morada = value;
            }
        }

*/
        public string CodCliente
        {
            get;
            set;
        }

        public string NomeCliente
        {
            get;
            set;
        }

        public string NumContribuinte
        {
            get;
            set;
        }
```

Visual Studio 2013 Express

#3 Connection to ERP Primavera Engine

Class PriEngine

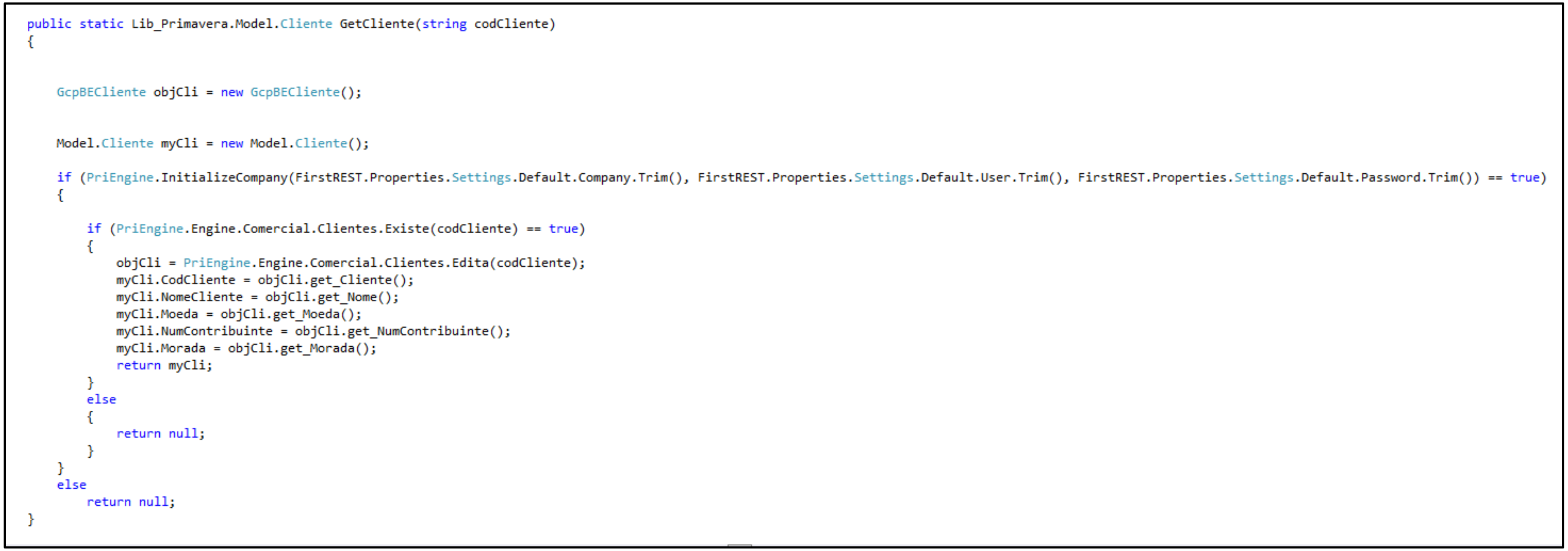


```
public class PriEngine
{

    public static StdPlatBS Platform { get; set; }
    public static ErpBS Engine { get; set; }

    public static bool InitializeCompany(string Company, string User, string Password)
    {

        StdBSConfApl objAplConf = new StdBSConfApl();
        StdPlatBS Plataforma = new StdPlatBS();
        ErpBS MotorLE = new ErpBS();

        EnumTipoPlataforma objTipoPlataforma = new EnumTipoPlataforma();
        objTipoPlataforma = EnumTipoPlataforma.tpProfissional;

        objAplConf.Instancia = "Default";
        objAplConf.AbvtApl = "GCP";
        objAplConf.PwdUtilizador = Password;
        objAplConf.Utilizador = User;

        StdBETransaccao objStdTransac = new StdBETransaccao();

        // Opem platform.
        Plataforma.AbrePlataformaEmpresaIntegrador(ref Company, ref objStdTransac, ref objAplConf, ref objTipoPlataforma);

        // Is plt initialized?
        if (Plataforma.Inicializada)
        {

            // Retuns the ptl.
            Platform = Plataforma;

            bool blnModoPrimario = true;

            // Open Engine
            MotorLE.AbreEmpresaTrabalho(EnumTipoPlataforma.tpProfissional, ref Company, ref User, ref Password, ref objStdTransac, "Default", ref blnModoPrimario);

            // Returns the engine.
            Engine = MotorLE;

            return true;
        }
        else
        {
            return false;
        }


    }
```

Visual Studio 2013 Express

#4 Business Rules

GetCliente (string codCliente)

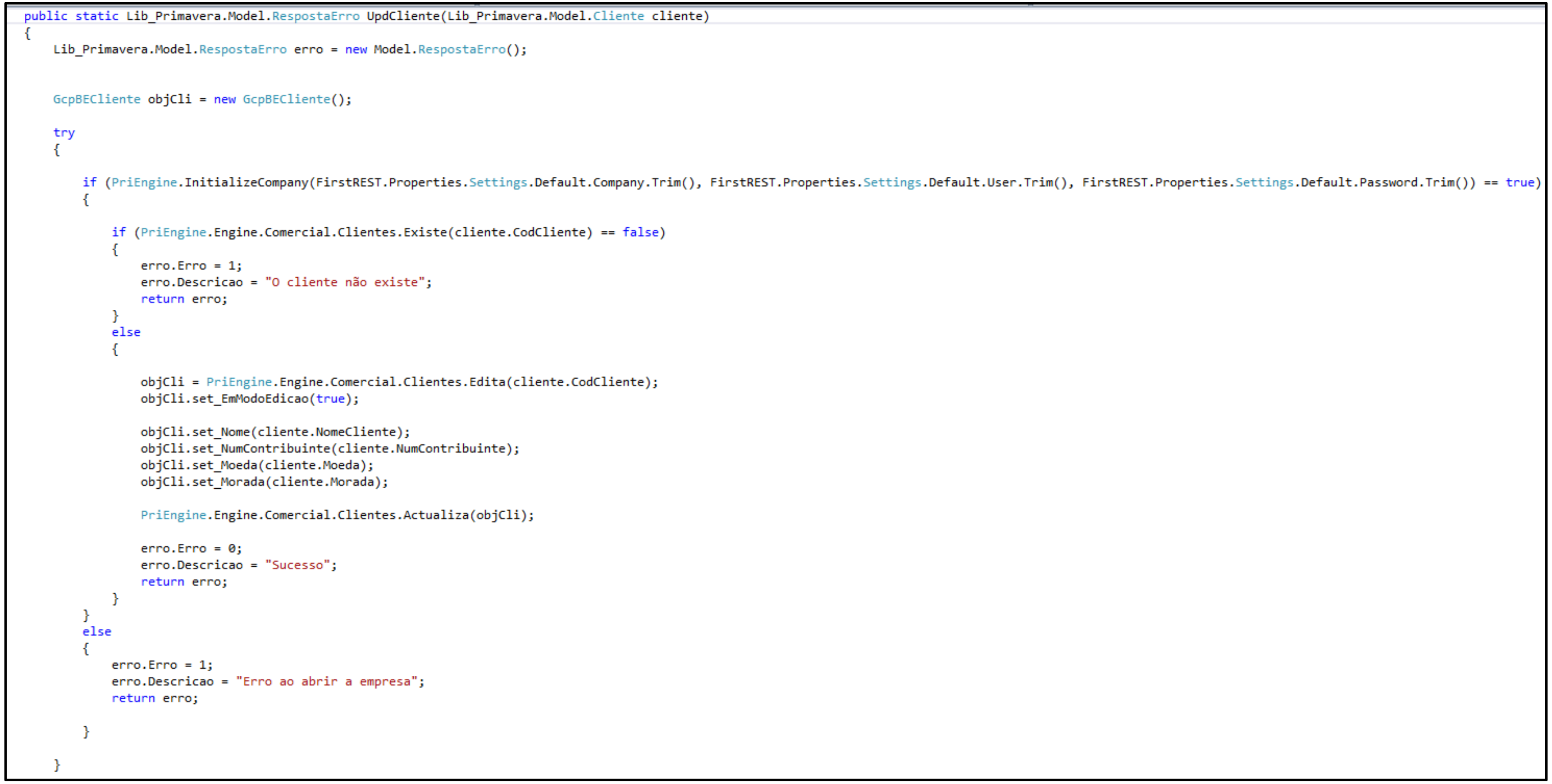


```
public static Lib_Primavera.Model.Cliente GetCliente(string codCliente)
{


    GcpBECliente objCli = new GcpBECliente();


    Model.Cliente myCli = new Model.Cliente();

    if (PriEngine.InitializeCompany(FirstREST.Properties.Settings.Default.Company.Trim(), FirstREST.Properties.Settings.Default.User.Trim(), FirstREST.Properties.Settings.Default.Password.Trim()) == true)
    {

        if (PriEngine.Engine.Comercial.Clientes.Existe(codCliente) == true)
        {
            objCli = PriEngine.Engine.Comercial.Clientes.Edita(codCliente);
            myCli.CodCliente = objCli.get_Cliente();
            myCli.NomeCliente = objCli.get_Nome();
            myCli.Moeda = objCli.get_Moeda();
            myCli.NumContribuinte = objCli.get_NumContribuinte();
            myCli.Morada = objCli.get_Morada();
            return myCli;
        }
        else
        {
            return null;
        }
    }
    else
        return null;
}
```

Visual Studio 2013 Express

#4 Business Rules

UpdCliente(LibPrimavera.Model.Cliente)



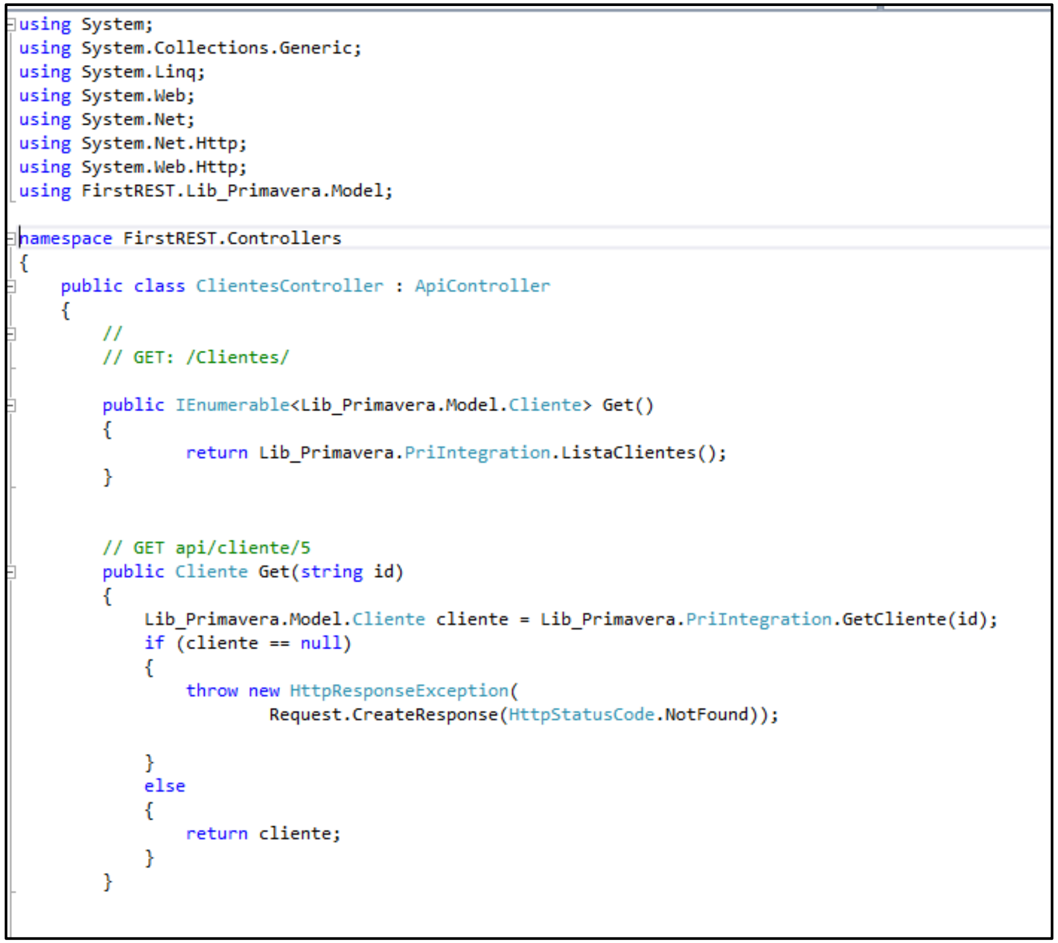
```
public static Lib_Primavera.Model.RespostaErro UpdCliente(Lib_Primavera.Model.Cliente cliente)
{
    Lib_Primavera.Model.RespostaErro erro = new Model.RespostaErro();


    GcpBECliente objCli = new GcpBECliente();

    try
    {

        if (PriEngine.InitializeCompany(FirstREST.Properties.Settings.Default.Company.Trim(), FirstREST.Properties.Settings.Default.User.Trim(), FirstREST.Properties.Settings.Default.Password.Trim()) == true)
        {

            if (PriEngine.Engine.Comercial.Clientes.Existe(cliente.CodCliente) == false)
            {
                erro.Erro = 1;
                erro.Descricao = "O cliente não existe";
                return erro;
            }
            else
            {

                objCli = PriEngine.Engine.Comercial.Clientes.Edita(cliente.CodCliente);
                objCli.set_EmModoEdicao(true);

                objCli.set_Nome(cliente.NomeCliente);
                objCli.set_NumContribuinte(cliente.NumContribuinte);
                objCli.set_Moeda(cliente.Moeda);
                objCli.set_Morada(cliente.Morada);

                PriEngine.Engine.Comercial.Clientes.Actualiza(objCli);

                erro.Erro = 0;
                erro.Descricao = "Sucesso";
                return erro;
            }
        }
        else
        {
            erro.Erro = 1;
            erro.Descricao = "Erro ao abrir a empresa";
            return erro;

        }

    }
```

Visual Studio 2013 Express

#5 Controllers

ClientesController

- HTTP/RESTfull
  - GET
  - PUT
  - POST
  - DELETE



```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Web;
using System.Net;
using System.Net.Http;
using System.Web.Http;
using FirstREST.Lib_Primavera.Model;

namespace FirstREST.Controllers
{
    public class ClientesController : ApiController
    {
        //
        // GET: /Clientes/

        public IEnumerable<Lib_Primavera.Model.Cliente> Get()
        {
                return Lib_Primavera.PriIntegration.ListaClientes();
        }


        // GET api/cliente/5
        public Cliente Get(string id)
        {
            Lib_Primavera.Model.Cliente cliente = Lib_Primavera.PriIntegration.GetCliente(id);
            if (cliente == null)
            {
                throw new HttpResponseException(
                        Request.CreateResponse(HttpStatusCode.NotFound));

            }
            else
            {
                return cliente;
            }
        }
```

Testing the WS with Google Advanced Rest Client

URLs:

- GET: http://localhost:49822/api/clientes
- POST: http://localhost:49822/api/clientes

  Values:

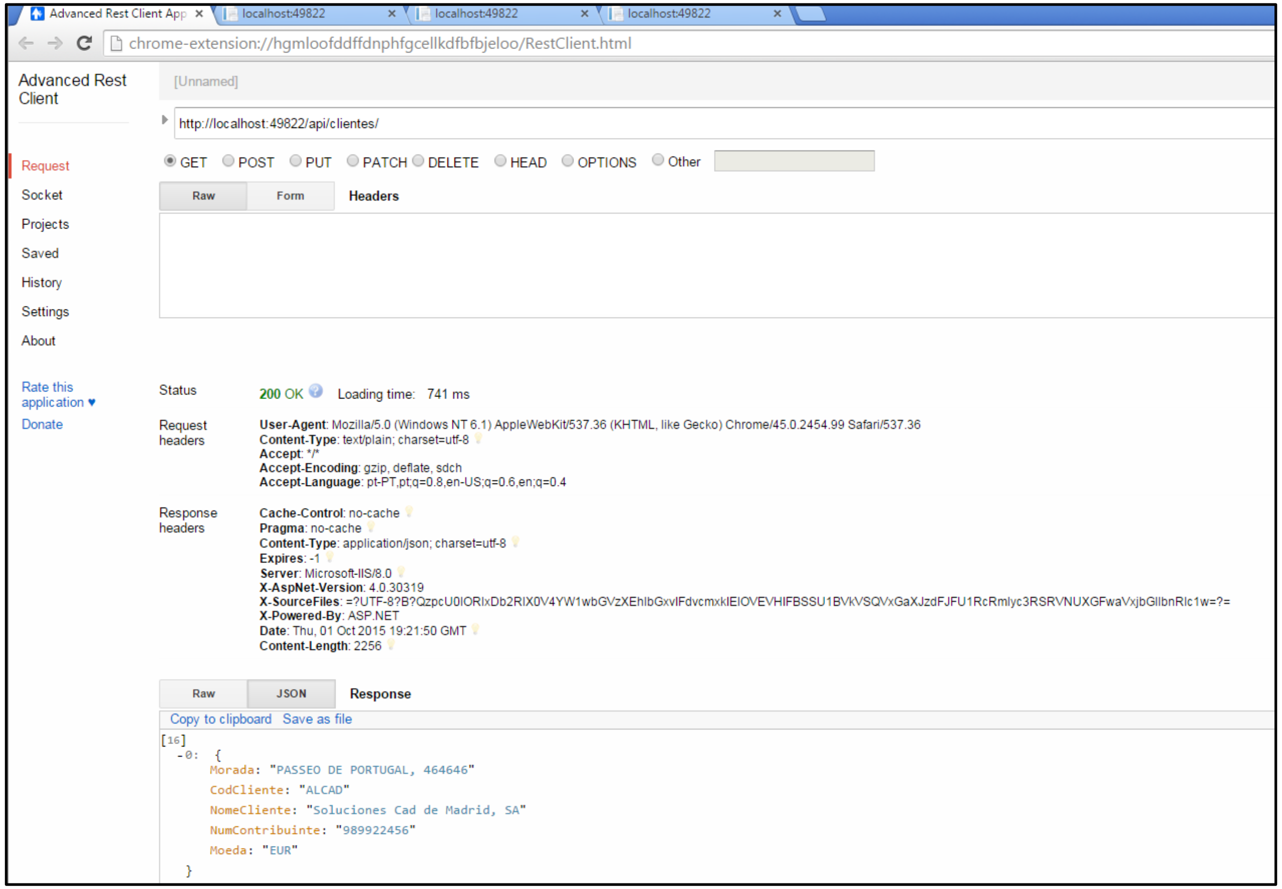
  ```
  {
  "Morada": "PASSEO DE PORTUGAL, 464646",
  "CodCliente": "<customer code>",
  "NomeCliente": "NFS001, SA",
  "NumContribuinte": "989922458",
  "Moeda": "EUR",
  }
  ```

- PUT: http://localhost:49822/api/clientes/<customer code>

  ```
  {
  "Morada": "Rua das Flores",
  "CodCliente": "<customer code>",
  "NomeCliente": "NFS001, SA",
  "NumContribuinte": "989922458",
  "Moeda": "EUR",
  }
  ```

- DELETE: http://localhost:49822/api/clientes/<customer code>

Remark: application/json (input type)



chrome-extension://hgmloofddffdnphfgcellkdfbfbjeloo/RestClient.html

http://localhost:49822/api/clientes/

Status: 200 OK Loading time: 741 ms

Request headers

```
User-Agent: Mozilla/5.0 (Windows NT 6.1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/45.0.2454.99 Safari/537.36
Content-Type: text/plain; charset=utf-8
Accept: */*
Accept-Encoding: gzip, deflate, sdch
Accept-Language: pt-PT,pt;q=0.8,en-US;q=0.6,en;q=0.4
```

Response headers

```
Cache-Control: no-cache
Pragma: no-cache
Content-Type: application/json; charset=utf-8
Expires: -1
Server: Microsoft-IIS/8.0
X-AspNet-Version: 4.0.30319
X-SourceFiles: =?UTF-8?B?QzpcU0lORIxDb2RIX0V4YW1wbGVzXEhlbGxvIFdvcmxkIEIOVEVHIFBSSU1BVkVSQVxGaXJzdFJFU1RcRmlyc3RSRVNUXGFwaVxjbGllbnRlc1w=?=
X-Powered-By: ASP.NET
Date: Thu, 01 Oct 2015 19:21:50 GMT
Content-Length: 2256
```

```
[16]
 -0:  {
       Morada: "PASSEO DE PORTUGAL, 464646"
       CodCliente: "ALCAD"
       NomeCliente: "Soluciones Cad de Madrid, SA"
       NumContribuinte: "989922456"
       Moeda: "EUR"
   }
```