Student Profile

Description:

Student Profile is a bilingual (AR-EN) inside-developed Portal, developed by the Department of Development at UOP Computer Center.

This Portal developed using ASP.NET, MVC with cshtml UI for the layout. This portal provide several student service such as (Academic Record, Online Registration ... etc.).

Such portal meant to be secured using UOP AD (Active Directory), each student will sign on using his Student ID and Single Sign-on Password.

Only Regular-status Students can use this portal.

Services provided:

1- Home Page:
   
   a. Section 1: Shows student information given by the Registration Department. These info are not given to be edited by the student himself, in any case these info will only be updated by the Registration Department, although these information comes from the Registration Department, they still reside in our SQL Fast-link DB, but never to update unless they modified by the Registration Department and these info should automatically reflected to our DB by the Auto-Refresh App.

   
   b. Section 2: Shows Contact information for the student, these info comes from the Registration Department as well and updated only using the Auto-Refresh App, if changed by the Registration Department an auto reflection to these info should change them automatically.
c. **Section 3:** Shows an option for the student to add/edit extra contact information, which also be restored in the same SQL DB student Table. (* Tables will specified in more details in the DB Section from this Documentation).

![Image of student contact information](image)

d. **Section 4:** Shows student Academic Record. This provide the ability to preview current and previous academic record per selected semester by the student, details marks per course as well. This section tightly related to the *Academic Evaluation* Process, while the Evaluation process activated for such a semester the student forced to complete the evaluation for each course in order to review course marks details.

![Image of academic record](image)

2. **Edugate Link:**

A link provided using selected menu named (Edugate – البوابة الإلكترونية) that calls a third party App provided by ATS Company who is responsible of developing the Registration Department Apps. The call done by using a web service (Edugate Web service) reside in the UOP main application server. This service provides a secured connection between the UOP main application server and the Registration Department main application server, the service passes the student ID and password to the third party application. The third part application will respond back to the service and opens the student profile inside the Registration Department App. It will call back the service again in case of student online registration process just to double check the password provided by the student by our profile login.

![Image of student profile](image)

This redirect action will result to open the student profile on ATS App.
3- **User Manual:**
A link provided using selected menu named (User Manual – دليل المستخدم), will open a PDF that shows a detailed user manual and description of how-to-use and info regards the Registration Department application.

4- **Skills Exams:**
This menu link will redirect the student to a page that allows him/her to register for a skills exams sessions provided by the UOP Registration Department. This process will reserve and automatic Date/Time exam schedule for the first available session. In case that any student would like to register in a specific date, he can do so by heading to the Registration Department or Computer Center (Coordinator of electronic services and technical support Section ) with a good reason to complete his registration with chosen dates available at that semester. If a student already have register, never allow him to register using the portal. In case of student absent and ask for re-register, only done by providing a formal letter from the Registration Department.

5- **Email:**
This provides the ability for the student to use his credentials to use the online outlook email account.
Login and Password Change:

1- **Login:**
Any UOP Student receive a user name (Student ID) and Random Password (Single sign-on) once he registered in the UOP university from the Computer Center Department.

2- **Password Change:**
This portal provides the ability to change password at any time. The password should meet AD criteria’s; otherwise, the process will lead to failure of password change operation.

Password criteria’s:
- Eight characters length.
- Contains at least one number.
- Contains at least one Capital Letter.
- Special characters allowed but preferred not to use (< > ! + _ - %) due to some issue with the ATS registration application.

3- **Password Reset (forgot Password):**
This simple application allow the Regular student to generate a new password at any time & for any case.

Student to complete this operation should provide simple information. Steps are very easy and clear.

He will asked to provide the following information to reset his/her password:
- Student ID.
- DOB (Date of Birth).
- National Number (for Jordanian Students), Nationality for non-Jordanian Students.
- If all provided information correct and matches students stored information a random mixed text will appears to the student and should be inserted as appears in the box (captcha), then an email with the new password will be sent to student. **Note:** for any delivery complains by the student, let him/her check spam or junk folders in the email account.
**Application Source Code:**

**Student Portal** Application Source code could be found on the TFS (Team Foundation Server).

Directory path: Tfs-srv\website\UOP Website\UOP\Website\UOP.website.StudentProfile.

**Password Reset** application source code could be found on the TFS Server as well.

Directory path: Tfs-srv\website\UOP Website\UOP\Website.Secured\UOP.Website.Secured.Registration.

**Database Schema:**

**Student Portal:** mainly depend on the Fast-Link MS-SQL DB, which contains relatively but not limited to the following tables:

- **SIS_********** > information like student information and academic record could retrieved.
- **Marks_******** > information like course marks details could retrieved.
- **Questionnaire******** > mainly used for Academic Evaluation.

A fully detailed Schema Diagram could be found in the Student Profile Project Solution under the **Uop.ObjectModel.Students** Project holds the name **Student.edmx**

**Notes & Cases:**

**Student Portal:**

1- Login Failed:

   a. Wrong Student ID or (and) Password.  
   b. Student not registered in Active Directory.

2- Login successfully but without entering the profile:

   a. Student Status is not regular or withdrawal. (Status code <> 1 or <> 2).

3- Login in successfully but can’t login in to the Edugate:

   a. Student status code are not allowing him to enter his Edugate profile, student should refer to registration department to fix his/her academic situation.

4- Password change fail:

   a. Student try to submit a new password that does not meets the Single sign-on password criteria.

5- Reset Password failed:

   a. Student does not provide correct information.  
   b. Student not registered in the UOP AD.
6- Any other ATS Edugate issues not related to the CC Development Department student should refer to Registration Department.

7- General Edugate link failure:
   a. Check the web service on the UOP Application Sever, if NO issues call ATS to check if they have any issues.

8- Student course submit registration failed:
   a. Student Password contains one of these Char’s (< >! + _ - %), solution is to change password with new one not to contain any of these characters. We could take no action regards this matter.

**Academic Evaluation:**

Every semester Mr. Yosef Bakr, or who may be in charge will call or contact you by email to activate the Evaluation System and a give you the closing dates as well.

IN CASE of **Activation**: set the (IsLive) Field value to True in the **Questionnaire** Table.

IN CASE of **Deactivation**: set the (IsLive) Field value to False in the **Questionnaire** Table.

*Important: Make sure from the current semester information, and make a full Refresh using the refresh system in order to migrate all the Data from Registration Department.*

**Deployment and package building:**

Following steps explains how to build and deploy the student portal application in case of any changes to it:

1- After completing the desired updates to the application, make sure that its 100% free of error, i.e. Build succeeded.
2- Check in the latest updates using the TFS.
3- Get latest version just to double check that both client version & server version are same.
4- Go to Team Explorer Tab in VS 2010.
5- Look for the build with the name of your project, in our case the build should have the following name regards the student profile (Student Profile CI, Student Profile Deploy).

6- Queue a new StudentProfile CI. (Not mandatory, just to make sure that all goes fine). Right click on the StudentProfile CI and select queue new build.
A confirmation pop window will appear select Queue and you will see that the TFS starts the build operation.

7- A new window will open in vs2010 and show the current build process.
8- After this build operation ends successfully, use the same steps from (6-7), but this time to the StudentProfile Deploy package, as below.

Now after all builds ends remote to the UOP main Application Server.
9- In the application server, you will find a shortcut to the TFS Live Packages; it is a mapped network driver, probably with the Z letter indicator. 

10- Enter the driver, and then you will found several folders that contains sub folders with the latest build to the desire application. In our case will look for StudentProfile Deploy folder.

<table>
<thead>
<tr>
<th>Name</th>
<th>Date Modified</th>
<th>Type</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>ArabixFoo</td>
<td>1/15/2012 4:50 PM</td>
<td>File folder</td>
<td></td>
</tr>
<tr>
<td>Embassy Deploy</td>
<td>9/27/2012 11:46 AM</td>
<td>File folder</td>
<td></td>
</tr>
<tr>
<td>EmbassyTracker Deploy</td>
<td>10/10/2012 9:12 AM</td>
<td>File folder</td>
<td></td>
</tr>
<tr>
<td>Feedback Deploy</td>
<td>2/6/2012 4:02 PM</td>
<td>File folder</td>
<td></td>
</tr>
<tr>
<td>GraduateStudentPortal Deploy</td>
<td>5/3/2012 11:30 AM</td>
<td>File folder</td>
<td></td>
</tr>
<tr>
<td>Installers</td>
<td>6/13/2011 11:44 AM</td>
<td>File folder</td>
<td></td>
</tr>
<tr>
<td>MediaConference Deploy</td>
<td>12/13/2011 10:32 AM</td>
<td>File folder</td>
<td></td>
</tr>
<tr>
<td>PersonalPagestamp</td>
<td>11/30/2011 10:25 AM</td>
<td>File folder</td>
<td></td>
</tr>
<tr>
<td>ResolutionArchivingSystem Deploy</td>
<td>4/3/2012 3:54 PM</td>
<td>File folder</td>
<td></td>
</tr>
<tr>
<td>StudentMarks Deploy</td>
<td>8/15/2014 2:23 PM</td>
<td>File folder</td>
<td></td>
</tr>
<tr>
<td>StudentProfile Deploy</td>
<td>2/5/2014 11:42 AM</td>
<td>File folder</td>
<td></td>
</tr>
<tr>
<td>Transfer</td>
<td>5/23/2012 2:01 PM</td>
<td>File folder</td>
<td></td>
</tr>
<tr>
<td>Upp. Website, Secured, Evaluation Deploy</td>
<td>8/12/2012 10:31 AM</td>
<td>File folder</td>
<td></td>
</tr>
<tr>
<td>Upp. Website, Secured, Registration Deploy</td>
<td>1/26/2014 12:27 PM</td>
<td>File folder</td>
<td></td>
</tr>
<tr>
<td>WebServices Deploy</td>
<td>5/30/2012 3:53 PM</td>
<td>File folder</td>
<td></td>
</tr>
<tr>
<td>Website Security Deploy</td>
<td>2/26/2014 4:13 PM</td>
<td>File folder</td>
<td></td>
</tr>
</tbody>
</table>

11- Enter the folder and you will see latest folders that contains the top saved builds for the application. You will notice that folder name contains a date of latest build as a part of its name. As shown below.
12- Head to the folder of your choice. For example, we will choose the selected folder in the above image. You will see after opening the folder the following contents, as shown below.
13- Head to the _PublishedWebsites folder, open it, and then select the StudentProfile Package folder.

<table>
<thead>
<tr>
<th>Name</th>
<th>Date Modified</th>
<th>Type</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>StudentProfile</td>
<td>2/5/2014 11:42 AM</td>
<td>File folder</td>
<td></td>
</tr>
<tr>
<td>StudentProfile_Package</td>
<td>2/5/2014 11:42 AM</td>
<td>File folder</td>
<td></td>
</tr>
<tr>
<td>Uop.Website.Secured.StudentAnnouncement</td>
<td>2/5/2014 11:42 AM</td>
<td>File folder</td>
<td></td>
</tr>
<tr>
<td>Uop.Website.Secured.StudentAnnouncement...</td>
<td>2/5/2014 11:42 AM</td>
<td>File folder</td>
<td></td>
</tr>
</tbody>
</table>

14- Open this folder, you should see something like this,

<table>
<thead>
<tr>
<th>Name</th>
<th>Date Modified</th>
<th>Type</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>StudentProfile.deploy.cmd</td>
<td>2/5/2014 11:41 AM</td>
<td>Windows Command...</td>
<td>13 KB</td>
</tr>
<tr>
<td>StudentProfile.deploy-readme.txt</td>
<td>2/5/2014 11:41 AM</td>
<td>Text Document</td>
<td>4 KB</td>
</tr>
<tr>
<td>StudentProfile.SetParameters.xml</td>
<td>2/5/2014 11:41 AM</td>
<td>XML Document</td>
<td>1 KB</td>
</tr>
<tr>
<td>StudentProfile.SourceManifest.xml</td>
<td>2/5/2014 11:41 AM</td>
<td>XML Document</td>
<td>1 KB</td>
</tr>
<tr>
<td>StudentProfile.zip</td>
<td>2/5/2014 11:41 AM</td>
<td>WinRAR ZIP archive</td>
<td>7.178 KB</td>
</tr>
</tbody>
</table>

15- Now, to perform a build action, do the following:
   a. Hold the shift key + click right click using the mouse in any white space.
   b. You will see the following:
c. Now, choose “Open command window here” from the list.

d. A command window will open.
e. Now, type what you see below in the command for testing the application before performing the build action to life server:

```
“StudentProfiel.Deploy.cmd /t”
```

![Command Line Image]

f. Now, deploy the package using the same command replace the “t” with “y” and hit enter.

```
“StudentProfiel.Deploy.cmd /y”.
```

g. Now, go and check that your application is running the latest updates you made.

*Note: the above steps applied for any application that have a deployment package on the TFS server under the same directory.*
Alumni Portal

Introduction:

Alumni portal is an online ASP.NET web site that divided into two parts, first one is the Website, and the second is the Alumni portal.

The Web site contains latest news, events and announcements for alumni. In addition, it has the yearbook and alumni magazine. Alumni photo album and other info.

Alumni Portal provides to the alumni to login to his/her alumni profile and add/edit personal information so the university can contact with him/her at any time.

Alumni can update his CV for recruiting matters.

UOP export alumni info in order to issue statistics and reports to show current alumni different statuses and work information statistics.

Services provided:

1- Website:
The main alumni website as mentioned in the introduction above provides several information to alumni like news, new opening jobs, university news, yearbook, alumni magazine, university location, contact information...Etc.
2- **Alumni Portal:**
   
   a. **Registration:**
   
   UOP Alumni should register to use the alumni portal services and update his information, like contact info, work info ... etc.

   The system ask the student to submit a few of his/her academic information in order to complete the registration process as shown above.

   Then he will be able to login into the portal and update his/her data.

   b. **Login:**
   
   After student complete the registration operation successfully, he will be able to login into the portal and update his/her data.

   **Note:** Regards the student password, this application uses the password stored in SIS_STUDENTS table, field name password, and this application still not connected to the single sign on system due to old student’s records who does not have single sign on credentials.

   c. **Portal Services:**
   
   Here the alumni can browse the main menu to select a desirable option or category to update his/her data.
Main menu include the following:

a. **Home Page**: shows student general information like name, faculty, department and student id.

b. **General Information Page**: shows the following information and the ability to add student national ID in case not inserted before. The below cannot be edited by alumni since they stored and retrieved from UOP DB.
c. **Contact Information Page**: shows a form where the alumni can edit his/her contact information as below.

![Contact Information Form](image1.png)

---

d. **Work Status Page**: shows the current work status for the alumni and any related information like job description, job type, job contact information... etc.

![Work Status Form](image2.png)

---

e. **Academic Qualification page**: allows the alumni to add/edit his/her academic qualification record as shown below.

![Academic Qualification Form](image3.png)
f. **Work experience page:** allows the alumni to add/edit his work experience record as shown below.

   ![Work experience page](image)

   - **F. Work experience page:** allows the alumni to add/edit his work experience record as shown below.

   ![Work experience page](image)

   - **G. Search a job page:** allows the alumni to submit a request for a job from a selected type and description like shown below.

   ![Search a job page](image)

   - **G. Search a job page:** allows the alumni to submit a request for a job from a selected type and description like shown below.

   ![Search a job page](image)

   - **H. Change Password:** In this page, alumni can change his password as shown below.

   ![Change Password](image)

   - **H. Change Password:** In this page, alumni can change his password as shown below.

   ![Change Password](image)

   - **I. Forgot Password Page:** allows pre-registered alumni to retrieve his/her password from the DB in case of loose or forgotten, as shown below. New password will sent to the alumni registered email. Alumni should enter his student Id in order to complete the operation in addition with a security code to confirm human interaction with the application.

   ![Forgot Password Page](image)

   - **I. Forgot Password Page:** allows pre-registered alumni to retrieve his/her password from the DB in case of loose or forgotten, as shown below. New password will sent to the alumni registered email. Alumni should enter his student Id in order to complete the operation in addition with a security code to confirm human interaction with the application.
**Application Source Code:**

Application Source code could found on the TFS (Team Foundation Server).

Directory path: Tfs-srv\website\UOP Website\UOP\GraduateStudentPortal.

**Database Schema:** mainly depend on the Fast-Link MS-SQL DB, which contains relatively but not limited to the following tables:

- Alumni*****.
- Grad*****.
- SIS_******.

**Cases & Notes:**

Almost no real cases occurs regards this system.

**Deployment and package building:**

No deployment package regards this system, only what you have to is update whatever you want, and then copy the desired updated files to their location in the UOP Application server under the following directory, **but remember not to copy the web.config files (leave as on the server).**

Remote to the server to the following path: “D:\ClassicWebsite\Sub Webs\UOPAlumni.”
Alumni Desktop Application:

Introduction:

This system is a desktop application, VB.NET based, used by the dean of student affairs, department of Alumni.

This system developed to maintain alumni information, such as (not limited to); alumni work information and status, statistics, reports, sending emails ... etc.

Services provided:

1- Add/edit alumni work Status:

This form allows the user to add/edit the alumni current work status and information related to his/her record.

When search for specific alumni using his/her student ID, information will filled in the alumni personal and academic profile as stored in our main DB. This information are not editable by the user, its only gives the user simple information about this alumni.

The last section in this form will allow the user to add/edit work status information for specific selected alumni.
2- **Settings and pre-definitions:**
This section have three main sub sections:
- **Add Main Work Sections:**
  This form allows the user to add/edit main work section and save them as system pre-defined settings.

- **Add Work Field:**
  This form allows the user to add/edit sub work section and save them as system pre-defined settings.
- **Add Quit Job Reason:**
  This form allows user to add/edit job-quitting reasons as description and save them as system pre-defined settings.

3- **Sending emails:**
This form allows user to send email(s) to group of alumni with new or announcements by user choice. This mail sender allows the user to attach files with extensions like .doc, .pdf or images as well. With the ability to choose alumni from different graduation years
4- **Statistical Reports:**
This system provides several reporting features developed based on customer (user) demand along with system development phase. These reports used to give many indicators for the user regards numbers of alumni, work job statuses... etc.
These reports build to generate dynamic reports based on user criteria by his choice.
Types of reports are:
- Alumni work status report based on personal information.

- Alumni work status report based on academic information.
- Alumni work status report based on country and work section type.

- Alumni work status report based on achieves and user notes.
- All UOP alumni Report.

- All alumni without conditions.
5- **Statistical Reports:**
Reports with charts.

- Statistical report based on majors.

- Detailed Statistical report based on majors.
- All Alumni Statistical report.

- Statistical report based on residence location & nationality.
- Statistical report based on gender and work status.

**Login and Password Change:**

Users allowed to login to this system using a stored username and password. Table that contains such data is **GradStd_Users**, in the main DB. To add/edit users and passwords use this table.

**Application Source Code:**

Application Source code could found on the TFS (Team Foundation Server).

Directory path: Tfs-srv\website\UOP Website\WinForms\GradStudents.

**Note:** Also, you can find a version from the system save on the external HD belongs to the Department of programing.

**Database Schema:** mainly depend on the Fast-Link MS-SQL DB, which contains relatively but not limited to the following tables:

- Alumni********.
- Grad*****.
- SIS_*******.

**Cases & Notes:**

Almost no real cases occurs regards this system.
**Deployment and package building:**

Once you wish to edit/add features to this system, you can build it using VS 2010 and then publish a new version from it on the WSUS server using the following directory path `\\wsus\app\`.

Automatically the new version will updated and replace user’s version in the next time he use the app.