**inSight 2017**

**Department : ECE department**

**Problem Title : Resume (online profile) builder and achievement tracker.**

**Problem Code : PS06**

**Problem Description :**

The students attend technical events like seminars, paper presentations, hackathons and many more. The faculties go to different colleges as guest lecturers, also to seminars and other events. In order to track these achievements of the faculty and students an application needs to be designed which would benefit the students and faculties by helping them build their resume(an online profile) and help the admin to keep track of the achievements. The application should provide students the interface to create their resume and store their inputs or responses in the form of a database so that it becomes a record for the department. This would enable the departments to track achievements of their students and faculties efficiently. This should also have provisions to edit their profile whenever they which to update their profile or resume. The application should be able to generate a report of these achievements whenever required.

**Proposed solution:**

We propose to build a web application which will allow the users to create and maintain their resume online. The users can update their achievements which will be displayed in the achievement section. This application can be used by the departments to generate a report when required in the pdf format

How the system works:

* The authorities will first register themselves as the administrators for the application this registration will happen via command line interface for security reasons also any number of admins can register using a secrete key.
* The users need to register themselves for the application using the signup page by providing their personal information such as name, mobile number etc.
* The details such as password will be encrypted using a hash function for security reasons and the same hash value will used to verify the user details during authentication.
* The users will be presented with a form in which they need to present their information which needs to appear in the resume.
* The users will also be presented with several templates for their resume one of which needs to be chosen by the user.
* The personal details will be displayed in their profile page which can be viewed by the students or admin. The students will have the option to edit /modify their personal details at any point of time.
* Since these details will be stored the database the authorities will be able to search for a particular user. The resume can also be saved in a pdf format for their later use.
* The application will rate the users resume depending on how well the resume is made(The rating will be based on the skills and achievements)
* The admin will also be able to generate a report of all students who have completed a particular achievements (ex: winners of abc event) in a table format which can be saved as pdf for later use.
* The achievements and skills can be updated in the system as when they are acquired by the users which will be displayed in a separate section.
* The application will allow the other users of the system to comment on ones achievements or resume(Ex: Congratulatory message or request for collaboration etc)

The web application will be built in python using the Flask framework along with various other libraries such as SQLAlchemy (an ORM), Flask-Weasyprint (to convert html to pdf), Flask-Bcrypt (A cryptographic hash function for encrypting/decrypting passwords), Tinymce (A WYSIWYG editor plugins for creating notices) etc. We will use SQLite3 database for storing user data and also TWILIO ( A cloud telephony service) for sending messages to the user.

**Technology Stack:**

**OS**: Linux

**Server:** Apache

**Database**: SQLITE3

**Scripting Language**: Python

**Web-Framework**: Python FLASK

**Frontend-framework**: Twitter Bootstrap

**Object Relational Mapper (ORM)**: SQLAlchemy

**Use Case:**

****

|  |  |
| --- | --- |
| **UC 1** | **Login/Signup** |
| **Description** | A Visitor can log in with certain credentials to become an authorised user or Signup to become a authorised user |
| **Actors** | Visitor, User, Server |
| **Normal Sequence** | 1. A Visitor visits a website, and goes to the “login” page  2. The Visitor enters in a username and password, and submits the form to the Server  3. The Server checks their username and password against the list of existing user accounts, and veriﬁes that the account exists  4. If account does not exist then the user is redirected to signup page to complete signup process and again redirected to login page afterwards  5. The Server updates the Visitor’s session to indicate that they are an authorised User  6. The Visitor becomes a User, and is redirected to a Profile page |

|  |  |
| --- | --- |
| **UC 2** | **Edit Profile/Create Resume** |
| **Description** | A Visitor can edit his personal details etc. |
| **Actors** | User, Server |
| **Normal Sequence** | 1. The User requests the server to provide the Edit profile form  2. The Visitor is ﬁlling out an entry ﬁeld in a form (e.g. an e-mail address in a registration form)  3. Once the ﬁeld is ﬁlled out, the Client submits the value to the Server for validation  4. If the Server ﬁnds the input is invalid (e.g. the address is malformed), it returns the result to the Client  5. The Client graphically highlights the ﬁeld, and displays a message next to it informing the Visitor the input is invalid, why it is invalid, (“this e-mail address already exists in our database;”)  6. The Visitor corrects the error (e.g. a valid email address)  7. The input is submitted again to the Server, and it is now correct; the ﬁeld highlight and error message is removed  8.The updated information is saved in the database and the user is redirected to the profile page where the updated information is displayed along with the ratings |

|  |  |
| --- | --- |
| **UC 3** | **View Profile/Resume** |
| **Description** | A User can view his personal information along with Resume etc. |
| **Actors** | User, Server |
| **Normal Sequence** | 1. A Visitor requests for the profile page  2. The Server constructs the page by querying the details of current user  3. The Visitor receives the HTML-only page  4. The received HTML page is displayed by the browser  5. if the user wish to save the resume he can do so by clicking generate pdf button  6. The HTML content of the page is then converted to the pdf format using the WeasyPrint and is then returned to the user |

|  |  |
| --- | --- |
| **UC 4** | **Search record/Generate report** |
| **Description** | The Admin can Generate report or search for a particular entry |
| **Actors** | Admin, Server |
| **Normal Sequence** | 1. The admin enters the search criteria in the search box  2. The Server constructs the page by querying list of all matching entries present in the database  3. The Server then paginates the list of entries and returns the html only page.  4. The Admin receives the HTML-only page and is displayed by the browser  5. If the Admin wishes to generate report then he can do so by clicking generate report button.  6. The HTML content of the page is then converted to the pdf format using the WeasyPrint and is then returned to the Admin |

|  |  |
| --- | --- |
| **UC 2** | **Update achievements** |
| **Description** | A Visitor can edit his personal details etc. |
| **Actors** | User, Server |
| **Normal Sequence** | 1. The User requests the server to provide the Edit achievements form  2. The Visitor is ﬁlling out an entry ﬁeld in a form (e.g. an e-mail address in a registration form)  3. Once the ﬁeld is ﬁlled out, the Client submits the value to the Server for validation  4. If the Server ﬁnds the input is invalid (e.g. the address is malformed), it returns the result to the Client  5. The Client graphically highlights the ﬁeld, and displays a message next to it informing the Visitor the input is invalid, why it is invalid, (“this e-mail address already exists in our database;”)  6. The Visitor corrects the error (e.g. a valid email address)  7. The input is submitted again to the Server, and it is now correct; the ﬁeld highlight and error message is removed  8.The updated information is saved in the database and the user is redirected to the achievements page where the updated information is displayed |

**Team Members details:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Role** | **Name** | **USN** | **Email id** | **Phone no.** |
| Team Leader | VIKRAM S RAO | 1SI14CS131 | vikramsrao96@gmail.com | 9591254802 |
| Team Member 1 | SHARANBASAPPA | 1SI14CS420 | poojarisharan28@gmail.com | 8050291788 |