# **Design documentation for Matrimonial Website**

**Problem Statement:**

Matrimonial website’s purpose is to create a matchmaking business by applying the information in the field. The site will provide the information of the male or female based on the filtering criteria like age, region, caste, etc. The site will provide the different type of accessibility like administrator, paid user, free user and first time guest.

**Front-End Design:**

The front-end part will be containing following:-

**Registration Page: -** To register a new user. Will need to fill username, first name, last name, password and user type.

**Login Page: -** To login a user. Will need to fill username, password and user type.

**Welcome Page: -** The welcome page will be having option to edit the profile and Search for profile.

**Edit Profile Page: -** The profile will be having three details, which are educational details, Address and Summary.

**Search Profile Page: -** Search page will be having option to filter based on like education, caste, age, region, religion, etc.

**Database Design:**

The database will be having four tables named user, education, address and Summary.

**User: -** This table will be containing the username, password, first name, last name and foreign key to education, address and summary.

**Address: -** This table will be containing the address of the user.

**Education: -** This table will be containing the educational details of the user.

**Summary: -** This table will be containing the summarize details like age, caste, religion, hobbies, height, weight, introduction, etc.

**Back-End Design:**

The back-end part is designed using Spring MVC framework and hibernate framework. There will be following classes: -

**User: -** This is a POJO class for User details.

**Education: -** This is a POJO class for the educational details.

**Address: -** This is a POJO class for the Address of user.

**Summary: -** This is a POJO class for the Summary about the user.

**Matrimonial: -** This is the main class to the project.

**Search: -** This is the action class to search the user.