# JavaScript for Front-End – Regular Exam Profile

You have been tasked to create a simple Front-End application, which looks like **profile user interface**. The GDPR was adopted as an attempt to give more control to users on their **personal data** so you should create an interface that allows them to **modify** their data.

## Task

The task is split into **2 sub-tasks**:

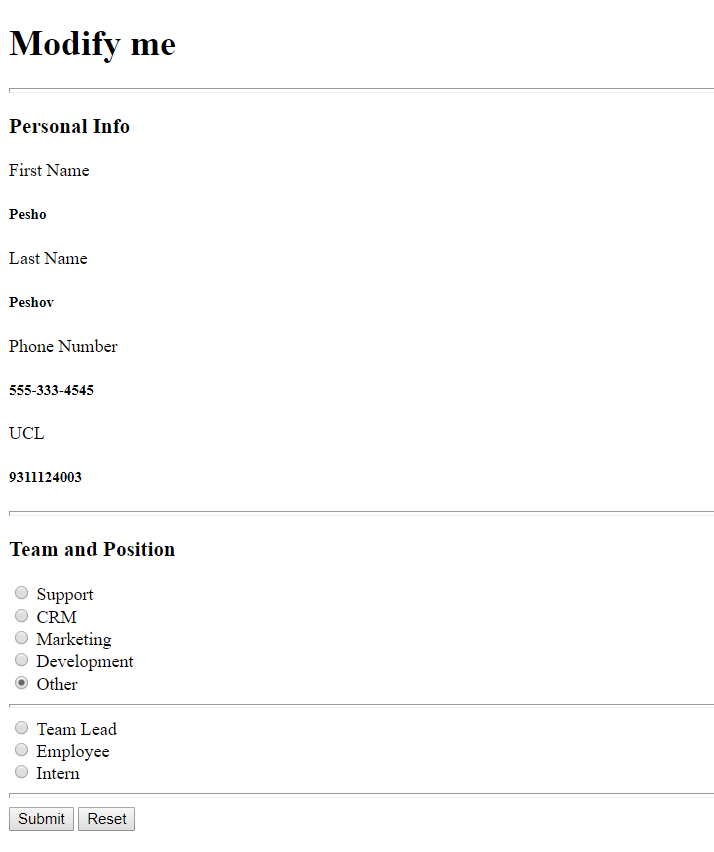
* A simple but fancy **web page design**.
* A simple but dynamic **web page functionality**.

The clients are understanding people, and they understand that you are quite novice as a developer, so they are willing to accept **ONLY** the **web page design** (without **functionality**) or **ONLY** the **web page functionality** (without **design**), if you are **unable** to **implement** one or the other.

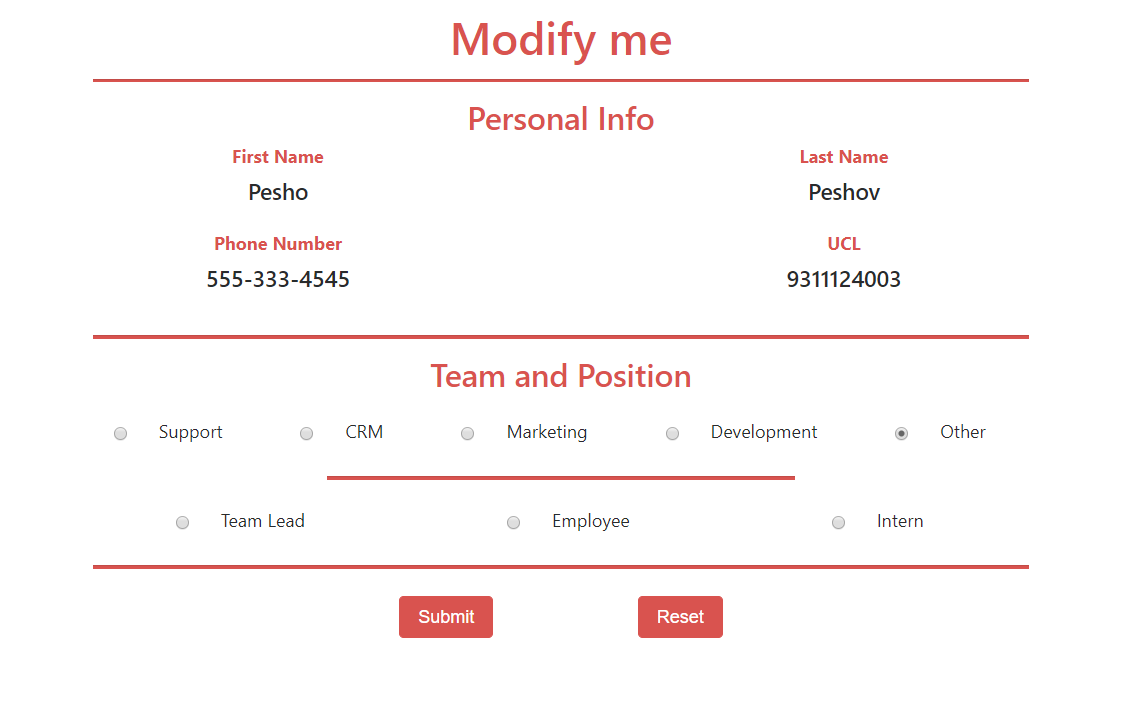
## Overview

### Web Page Design

The web page should look like this **WITHOUT design**:



The web page should look like this **WITH design**:



### Web Page Functionality

The application contains **2 main functionalities** – to **modify** the **personal data** and to **choose** a **team** and a **position in it**.

#### Personal Data Functionality

The first functionality is the **Personal Data Functionality**. The application should provide functionality to modify the personal data in the page.

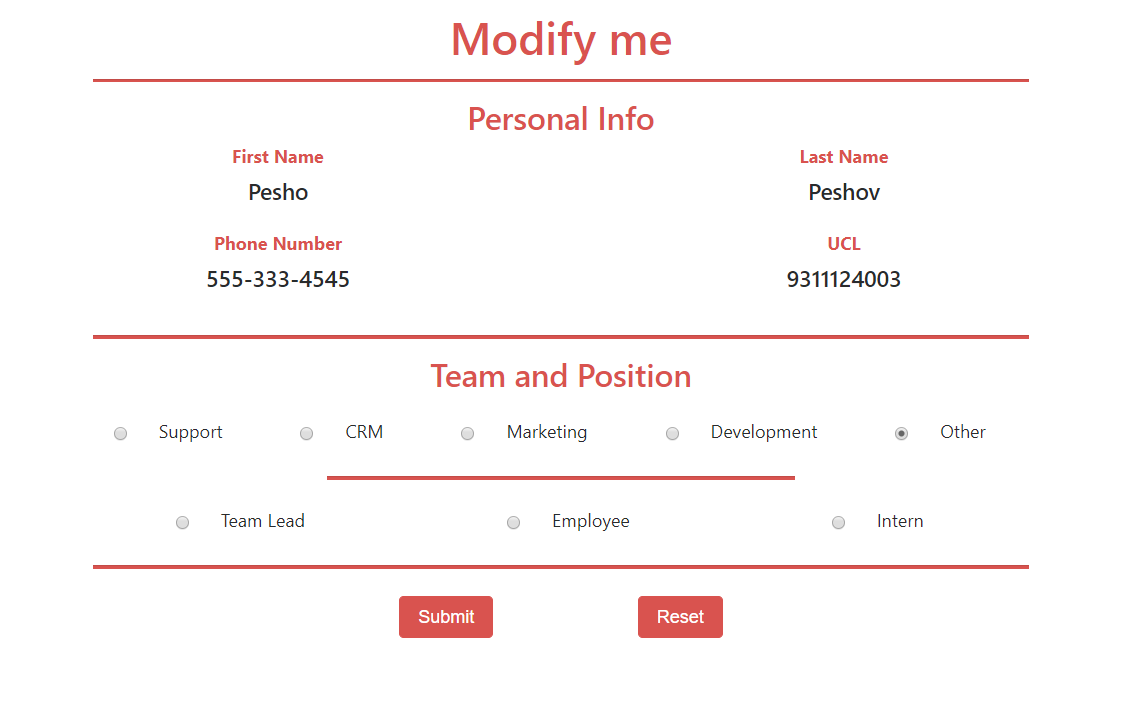
You should implement the page so that **initially** the personal data fields represent the following values:  
**[Pesho], [Peshov], [555-333-4545], [9311124003].**

When you **click** one values, it should change to an **input field**, which you should be able to **modify**. The **input field** should **contain** the **previous value**. In other words, if the **element** had the **text** “Pesho”, the **input field** should have a **value** “Pesho”.

When you click the [Enter] key (from the **keyboard**), while **typing** on the **input field**, it should take the **new value**, **remove** the **input field**, and bring back an **element** with the **new value**. Check the **GIF**s for a live explanation.

The input fields **types** are **[First Name] – text, [Last Name] – text, [Phone Number] – text** and **[UCL] – number**.

By **default**, the **personal data part** from the **page** should look like this:



When you **click** the [Submit] or [Reset] buttons you should **delete** the **changed data** and **replace** it again with the **initial values** from the beginning.

#### Team and Position Functionality

The second functionality is the **Team and Position Functionality**. The application should provide functionality to choose between 5 **different teams** and **for each team – 5 different positions**, each depending on the **team choice**.

You will have **5 teams**, which should be **represented** by **radio buttons**, the **default checked** being – [Other].   
When you **click** one of them, you should view **different positions**, which are again represented by **radio buttons.** The **teams** and their **positions** are:

* Support - [Tech Support], [Medical Support], [Assistant Support]
* CRM - [Community Manager], [Customer Care Manager], [Lead Administrative Officer]
* Marketing - [PR Manager], [Social Media Manager], [Marketing Specialist]
* Development - [Junior Developer], [Regular Developer], [Senior Developer]
* Other - [Team Lead], [Regular Employee], [Intern]

**Each team** choice represents **different** **positions**, you can see example below**:**



You will be given a **GIF** representing the functionality of the **web application**.

## Constraints

The clients are not that much into Front-End technologies, however:

* They heard **Bootstrap** is a good web page styling framework, and they have requested that you **use it**.
* They heard JavaScript makes pages quite dynamic, and **jQuery** is a good library, so you must **use it**.

That being said, you have several constraints to follow, during the implementation of the web application:

* You must use **ONLY** **Bootstrap** for the **web page design**.
* You must use **ONLY** native **JavaScript** & **jQuery** for the **web page functionality**.

## Hints

The <hr> (horizontal line) does not quite have a styling with **Bootstrap** which could make it look like the screenshots, so here is a **hint**. Implement **inline style** and set its height to 2px.

You can check how to **handle** **clicking** a **specific** **key** from the **keyboard**, with **jQuery**.