

Dynamic Web Design

Alpha prototype Report (Group 1)

Group Member: Ziyu Du; Luke Cao; Luoying Bao

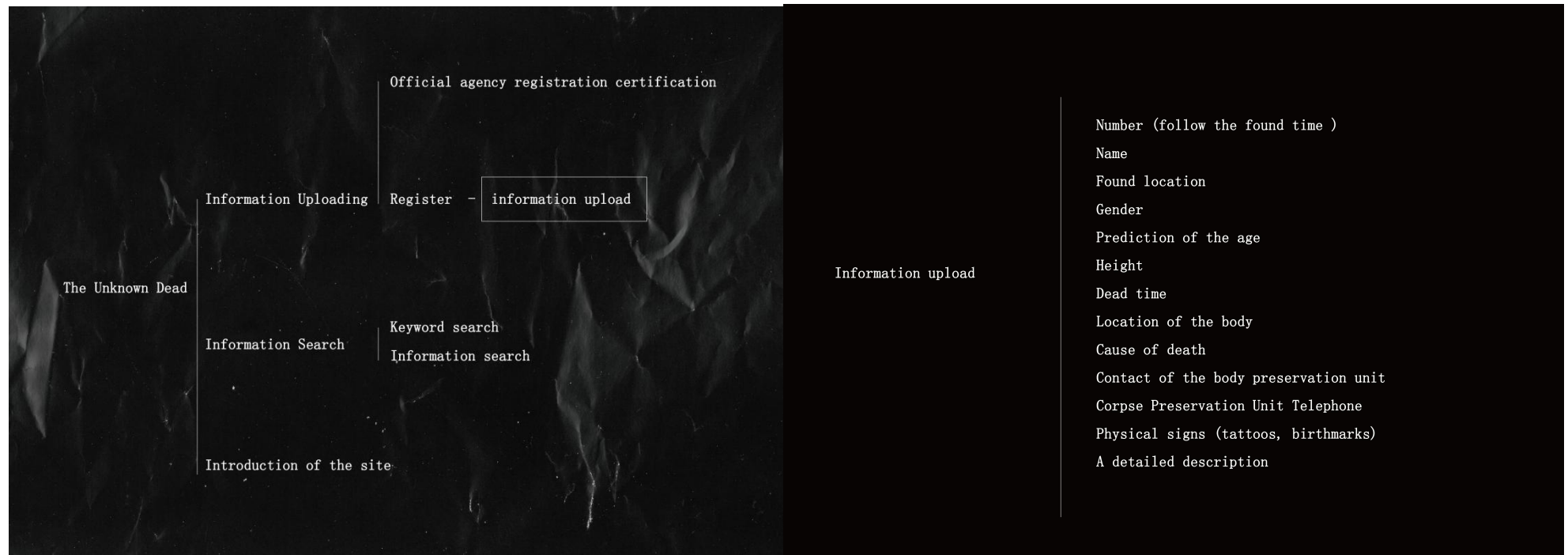
[https://media.ed.ac.uk/media/Video%20of%20Nameless%20Dead%20\(group1\)/1_snnkkzjz](https://media.ed.ac.uk/media/Video%20of%20Nameless%20Dead%20(group1)/1_snnkkzjz)

<https://ziyudu.edinburgh.domains/fatfree/DeadPeopleSystem/index>

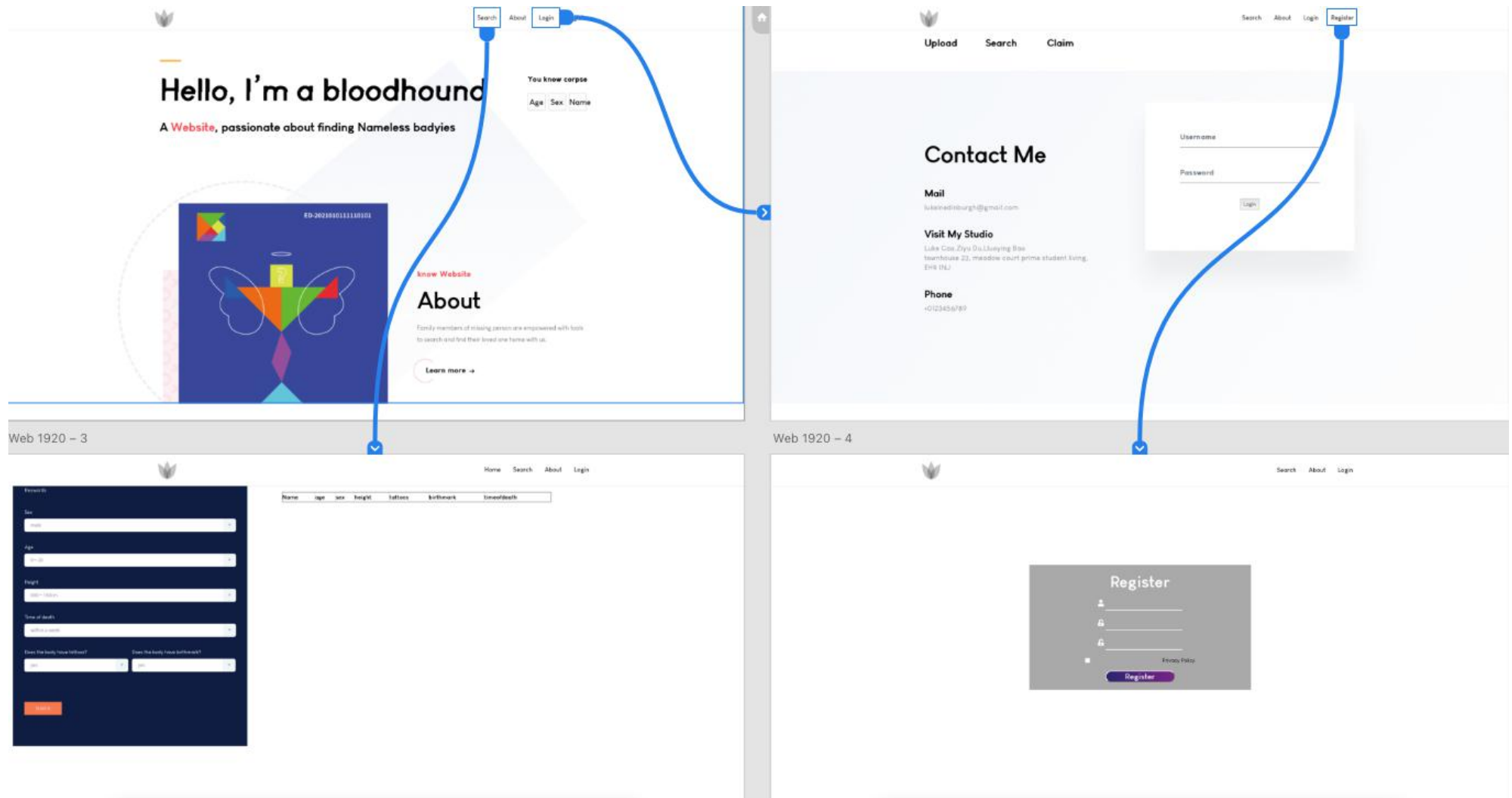
Background and Motivation

Expensive funeral costs are a serious problem and appear to have worsened over the years. According to these expensive funeral costs and other special circumstances, during the popularity of Covid-19, medical examiners' offices and local news has indicated that the number of unidentified bodies is climbing. So in our project, we are aiming to establish a website to connect police departments in different areas and help people search and claim unidentified bodies by posting body information on.

Introduction



1.0 User flow[1]



2.0 User flow[1]

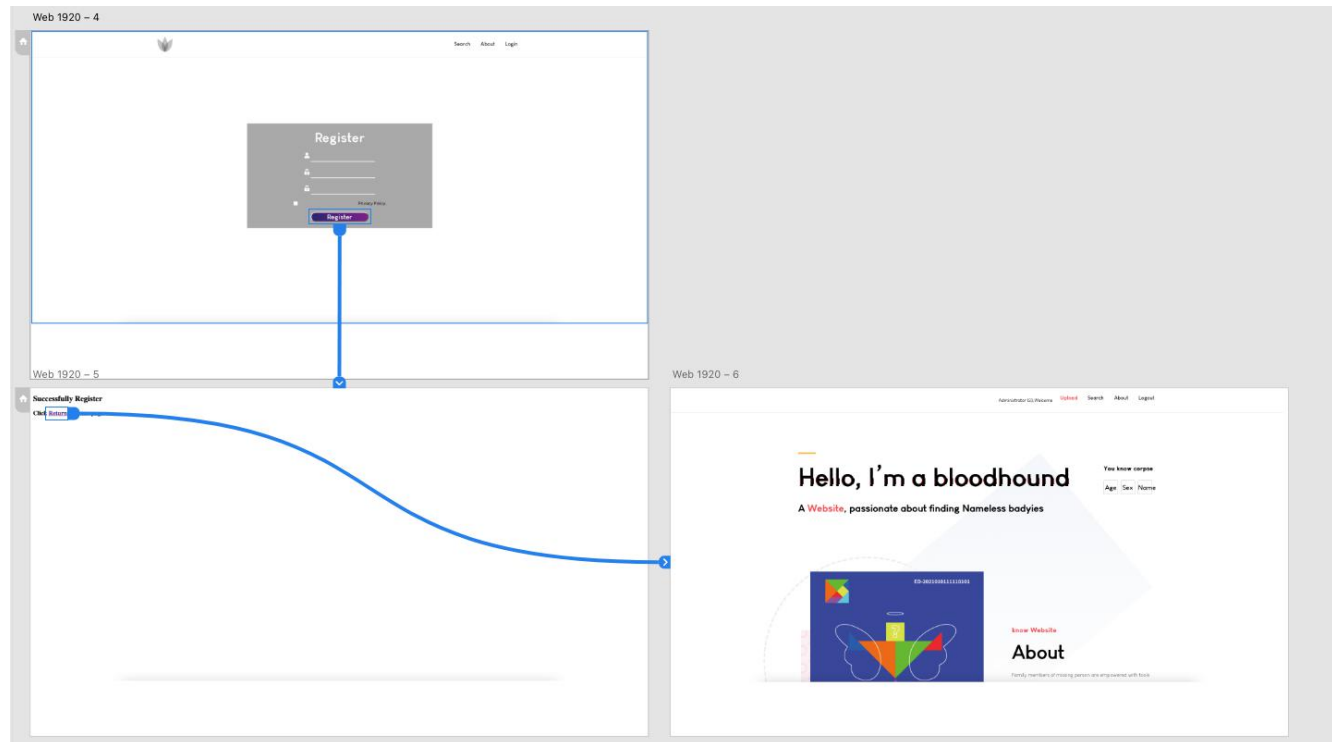


Figure: flow chart [2]

1.1 Front End

The Alpha prototype of our project contains three core functions: search, administrator register and login. These functions are mainly composed of html, which converts part of our ui design into simple lines of code. At the same time, we used some things on bootstrap to complete our front-end design, such as the animation design of the search bar and homepage.

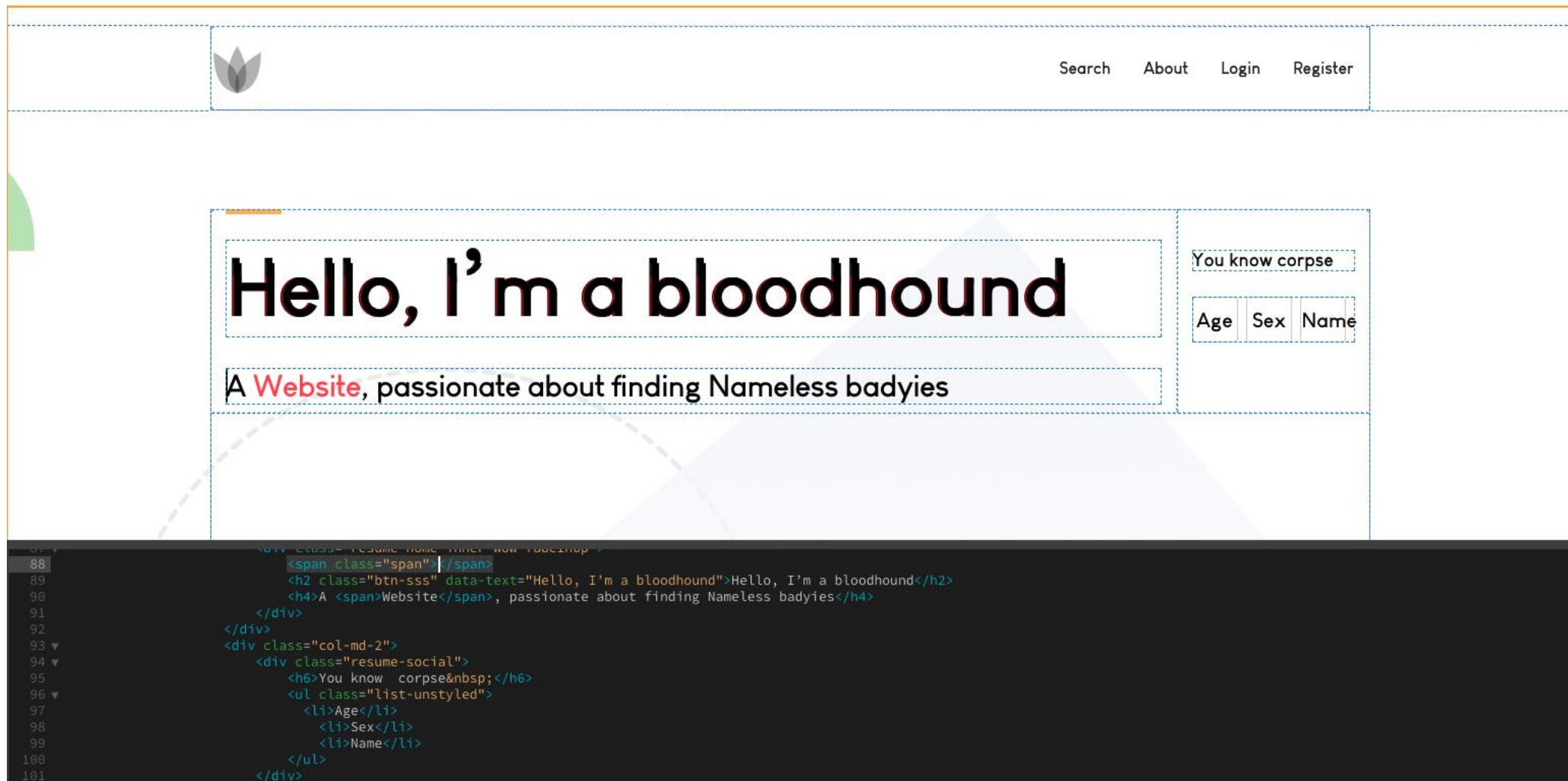


figure: homepage [2]

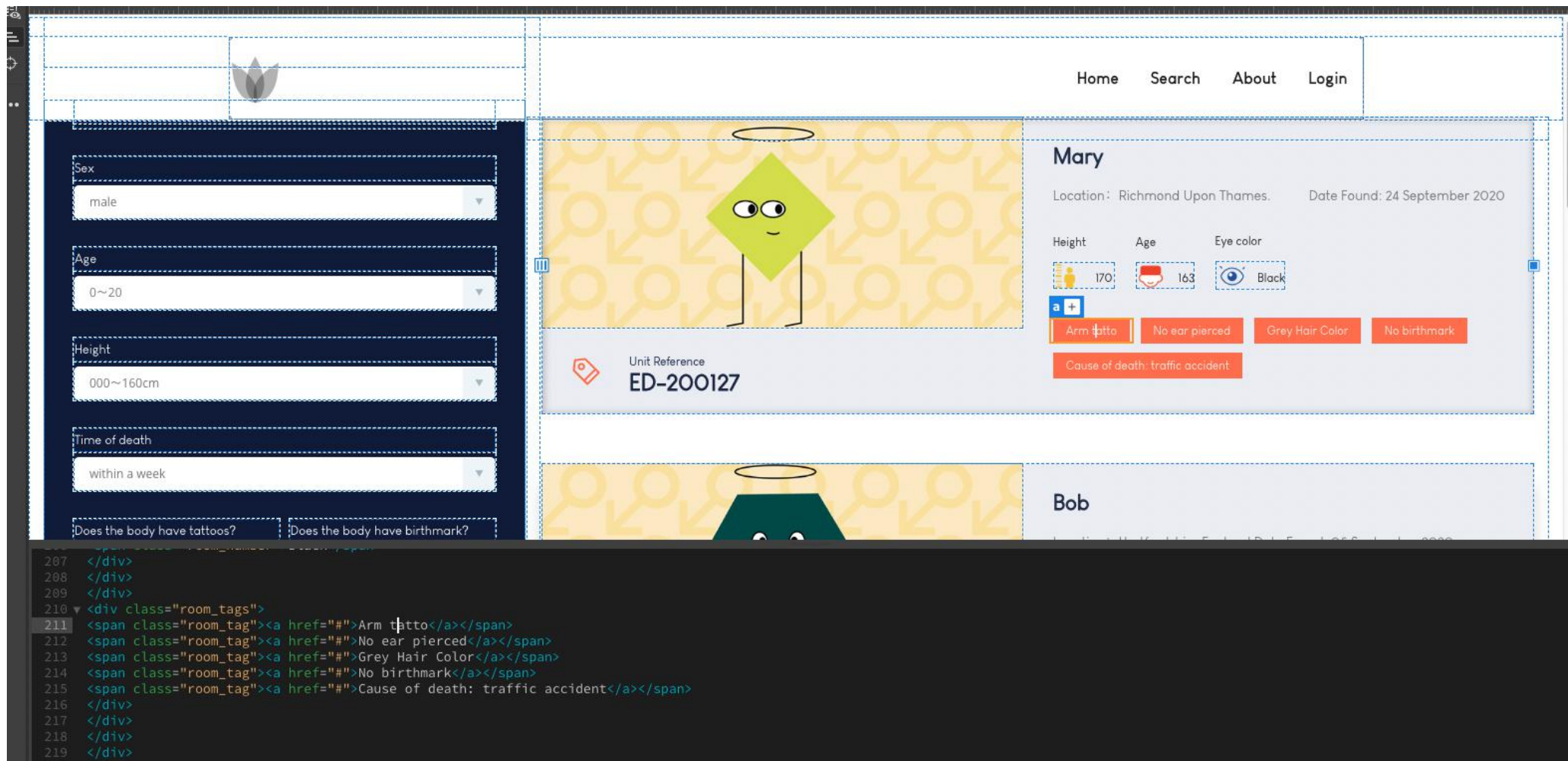


Figure: search page [2]

1.2 Back End

The Alpha prototype of our project contains three core functions: search, administrator register and login. These three functions are all completed by f3-fatfree-framework in index.php under the path of DeadPeopleSystem on server. The methodologies of these three functions are quite similar, information on form from front-end is loaded into back-end using http POST request and information will be processed such as insert, delete, query and modify. Then, the results of process from back-end will be returned to html page.

Search

The code of search function is as followed:

```
120 $f3->route('POST /search',
121 function($f3) {
122
123     $formdata = array(); // array to pass on the entered data in
124     $formdata['name'] = $f3->get('POST.name'); // whatever was called "name" on the form
125     $formdata['age'] = $f3->get('POST.age'); // whatever was called "age" on the form
126     $formdata['sex'] = $f3->get('POST.sex'); // whatever was called "sex" on the form
127     $formdata['height'] = $f3->get('POST.height'); // whatever was called "height" on the form
128     $formdata['tattoos'] = $f3->get('POST.tattoos'); // whatever was called "tattoos" on the form
129     $formdata['birthmark'] = $f3->get('POST.birthmark'); // whatever was called "birthmark" on the form
130     $formdata['timeofdeath'] = $f3->get('POST.timeofdeath'); // whatever was called "timeofdeath" on the form
131
132     $s = $formdata['sex'];
133     $t = $formdata['tattoos'];
134     $b = $formdata['birthmark'];
135     $todeath = $formdata['timeofdeath'];
136
137     case "50~70":
138         $ageLow = 50;
139         $ageHigh = 70;
140         break;
141     case "70+":
142         $ageLow = 70;
143         $ageHigh = 100;
144         break;
145 }
146 switch ($formdata['height']){//Initialize $heightLow and $heightHigh which will be used in sql query according to the condition $formdata['height']
147 {
148     case "000~160cm":
149         $heightLow = 0;
150         $heightHigh = 160;
151         break;
152     case "160~190cm":
153         // execute SQL query using $f3->get('DB')->exec() function, constrained by variables $ageLow,$ageHigh,$heightLow,$heightHigh,$s,$t,$b
154         $list = $f3->get('DB')->exec("SELECT * FROM deadinfo WHERE (age BETWEEN $ageLow and $ageHigh) AND
155 (sex=$s) AND (height BETWEEN $heightLow and $heightHigh) AND (tattoos=$t) AND (birthmarks=$b) AND (timeofdeath=$todeath)");
156
157         $f3->set('result',$list); //set $list into variable 'result'
158         echo template::instance()->render(Rc: 'search.html'); //return to search.html page
159 }
160 }
161 );
```

Figure: search function [1]

The php codes of search function can be divided into four sections. The first section from line 123 to 135 can be denoted as loading data. In this section, whenever the selections on form(front end) are called, some related data will be stored into array called formdata. The second section aims at initializing eight variables which will be used in SQL query. The SQL query can be considered as the third section of search function which is the core algorithm for overall function. In this section the SQL

query will be executed by applying exec() function and the this SQL query will be constrained by eight variables initialized in section 2. Finally, the query results will be returned to front end and refresh page.

Administrator register (The code of administrator register function is as followed)

```
80 $f3->route('POST /index-register',
81     function($f3) {
82         if($f3->get('POST.checkpolicy')){ // Check whether checkpolicy is true
83             if($f3->get('POST.password1')==$f3->get('POST.password2')){//Check whether pd1 and pd2 are same
84                 //if checkpolicy is true and pd1,pd2 are same
85                 //then insert Username and password into database
86                 $formdata = array(); // array to pass on the entered data in
87                 $formdata["username"] = $f3->get('POST.Username'); // whatever was called "Username" on the form
88                 $formdata["password"] = $f3->get('POST.password1');// whatever was called "password1" on the for
89
90                 $controller = new SimpleController;
91
92                 $controller->putIntoAdmDatabase($formdata);//insert formadate into database
93
94                 $f3->set('formData',$formdata); // set info in F3 variable for access in response template
95
96                 echo template::instance()->render( file: 'index_regsuccess.html');
97             }
98             //if password1 and password2 are different, then alert and refresh page
99             else{
100                 echo '<script type="text/javascript">alert("ERROR PASSWORD")</script>';
101                 echo template::instance()->render( file: 'index_register.html');
102             }
103         }
104         //if checkpolicy is False, then alert and refresh page
105         else {
106             echo '<script type="text/javascript">alert("Please check Service and Privacy Policy.")</script>';
107             echo template::instance()->render( file: 'index_register.html');
108         }
109     }
110 }
```

Figure: php code of administrator register [1]

This function aims at solving the problem of administrator registering. There are two constraints if an administrator registers successfully. Registrant must agree to the privacy policy and the confirmed password must be same as the one they typed in at the beginning. If there is a condition that is not met, alert interface will pop up and the page will be refreshed.

```
public function putIntoAdmDatabase($data) {  
    $this->loginMapper->username = $data["username"];           // set value for "username" field  
    $this->loginMapper->password = $data["password"];           // set value for "password" field  
    $this->loginMapper->save();                                   // save new record with these fields  
}
```

Figure: putIntoAdmDatabase function [1]

Administrator login (The code of administrator login function is as followed)

```
$f3->route('POST /index',  
function($f3) {  
    $controller = new SimpleController;  
    if ($controller->loginUser($f3->get('POST.Username'), $f3->get('POST.Password'))) { // user is recognised  
        $f3->set('SESSION.userName', $f3->get('POST.Username'));  
        // note that this is a global that will be available elsewhere  
        header( header: "location:/fatfree/DeadPeopleSystem/index-user");  
        // will always go to index-user after successful login  
    }  
    else{  
        echo "<script type='text/javascript'>alert('ERROR password or username')</script>";  
        // return to login page with the message that there was an error in the credentials  
        echo Template::instance()->render( file: 'index.html');  
    }  
}  
);
```

Figure: php code of administrator login [1]

This function waits for POST request in index page, loginUser function is used to verify the correctness of Username and Password. Whenever Username and Password is true, we will always go to index-user after successful login.

What to do next?

1. We also need to make an upload page and some simple information introduction page. The upload page includes the upload of the photo information of the deceased, as well as the map marking function, and users can save the location of the unnamed corpse online.
2. Our user interface design is based on the bootstrap template. We would have thought that this would reduce the workload but it would cause more troubles. We will redesign the user interface and the front-end design part but the back-end functions will remain constant.
3. In the back-end design, we also have many parts that need to be improved; for example, after retrieving the information of an unnamed corpse, the information display of the corpse currently only displays text and table information, and cannot view a series of information such as pictures.
4. We want to try to use three.js to use 3D interactive features on the homepage of our website. This may be difficult but we will definitely try.

Reference

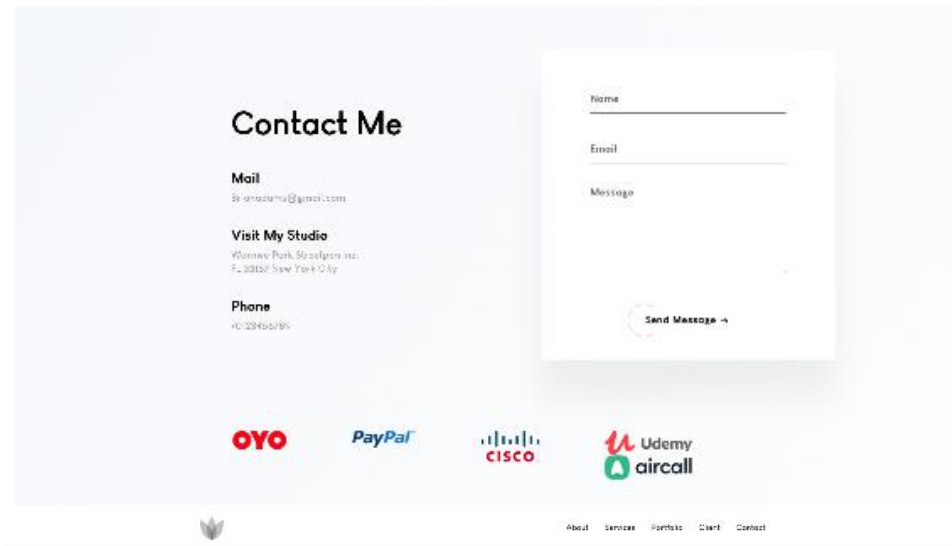
[1] FROM TUTORIAL:

```
public function getData() {
    $list = $this->deadinfo->find();
    return $list;
}
```

[2] front end Template:

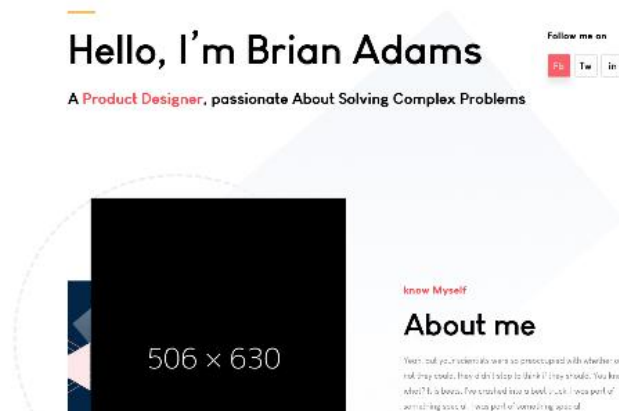
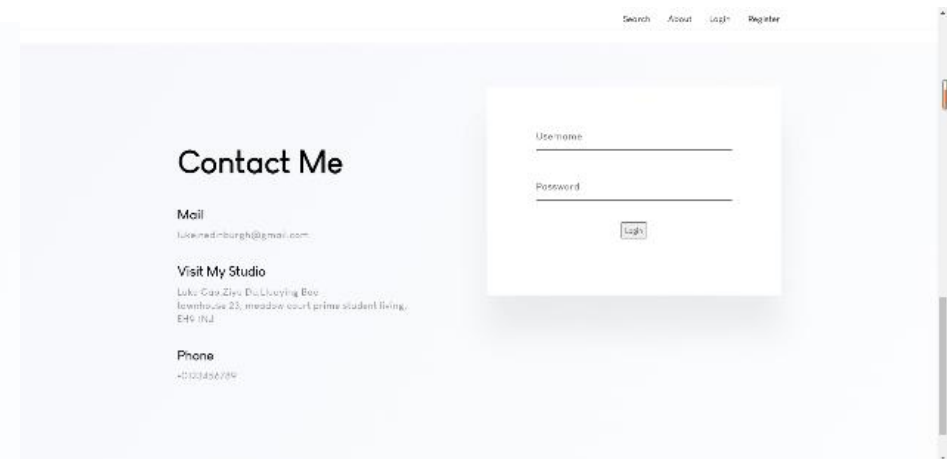
<https://ui8.net/luova-studio/products/sleek-portfolio-html-template>

Template



VS

Our Works



What I Do

With 10 years of experience, I was so preoccupied with what I could do, they didn't stop to think they should. You know what? It is better, for crashed into a beautiful one. I was part of something special. I was part of something special.

- 
- 
- 
- UX design

UI design

Branding

I Create Unique User Experience

How to use

Once you've created your information, you can upload it. You can search for information through the website, and then go to relevant agencies to find the copies.

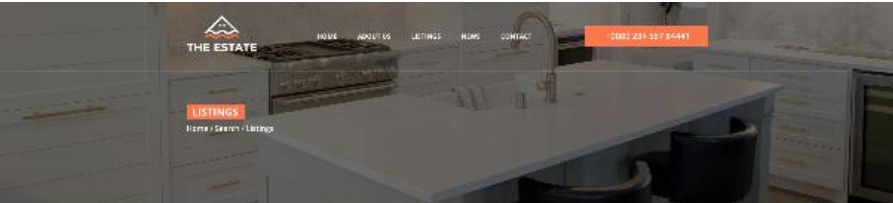
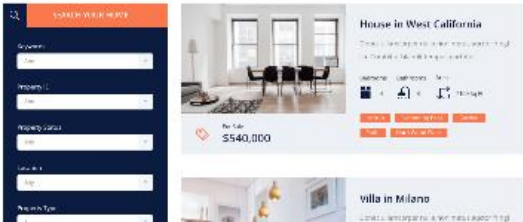
Nameless Dead

- Upload
- Search
- Claim


[3] front end Template:

<https://www.8m2.net/2019%e5%b9%b4%e5%a4%a7%e6%b0%94%e6%b5%b7%e8%93%9d%e8%89%b2%e7%81%b0%e8%89%b2%e6%88%bf%e5%9c%b0%e4%ba%a7%e7%bd%91%e7%ab%99%e6%a8%a1%e6%9d%bf/>


Template

VS



Our works



Division of labor

1. The theme of the webpage: Ziyu Du; Luke Cao; Luoying Bao
2. Main functions of the webpage & information structure: Ziyu Du; Luke Cao; Luoying Bao
3. Visual design: Luoying Bao
4. Front end design: Ziyu Du; Luke Cao;
5. Back-end design: Ziyu Du;
6. Video: Luoying Bao; Luke Cao