

Универзитет „св. Кирил и Методиј – Скопје“

Практичен проект

OCEAN Psychology test

Ментор:

Бобан Јоксимоски

Изработиле:

Никола Анастасовски 195012

Мартин Стојанов 191913

Содржина

Table of Contents

Содржина	2
Целта на проектот	3
Што се користи	4
Додатни фајлови и компоненти.....	5
Структура на БНП	7
Регистрирани Рути.....	10
Номерpage.....	11
Навигациско мени.....	12
Страна за прашања	13
Страна за резултати.....	15
Страна за сите резултати	19
Примени на проектот.....	21
Користена Литература	23

Целта на проектот

Во проектот се развива веб апликација во која корисниците можат да прават психолошки тестови користејќи го OCEAN моделот.

Психолошкиот тест OCEAN, познат и како Big Five personality traits или the Five-Factor Model, е широко користена рамка за разбирање на човечката личност. Ги категоризира карактеристиките на личноста во пет особини или димензии: отвореност, совесност, екстравертност, согласност и невротизам (Openness, Conscientiousness, Extraversion, Agreeableness, and Neuroticism).

Отвореност: Оваа димензија ја мери отвореноста на поединецот за нови искуства, идеи и емоции.

Совесност: Оваа димензија го одразува степенот до кој личноста е организирана, одговорна, дисциплинирана и ориентирана кон целта.

Екстраверзија: Екстраверзијата се однесува на нивото на ангажираност, дружељубивост и самоувереност што поединецот го покажува.

Согласност: Оваа димензија го мери степенот до кој личноста е сочувствителна, соработувачка и внимателна кон другите.

Невротичност: Невротичноста претставува емоционална стабилност и тенденција да се доживеат негативни емоции како анксиозност, депресија и промени во расположението.

Бидејќи најчесто не се тестираат сите особини целосно (целта на многу од овој вид на тестови на интернетот се насочени кон тестирање специфични карактеристики со некоја цел) и бидејќи е потребно поголемо психолошко знаење за да се тестираат сите особини а сите особини имаат повеќе карактеристики кои ги опишуваат, се тестираат само по една добра и една лоша карактеристика од сите 5 особини (вкупно 10).

Што се користи

Laravel (PHP) - Целиот проект е изработен користејќи го Laravel framework. Laravel е популарна рамка за веб-апликација на PHP со отворен код, позната по својата елеганција, едноставност и експресивна синтакса. Ја следи архитектонската шема на Model-View-Controller (MVC), која ја дели логиката на апликацијата на три меѓусебно поврзани компоненти: модели, погледи и контролери.

Елегантен синтаксис: Laravel нуди чист и едноставен синтаксис, што го прави лесно разбирлив и пријатен за програмерите.

MVC архитектура: Laravel следи Модел-Поглед-Контролер (Model-View-Controller) архитектурен шаблон, што помага во организацијата и раздвојувањето на логиката за податоци, приказот и контролата на апликацијата.

Моќен ORM (Објектно-релационо пресликување): Laravel има вграден ORM, наречен Eloquent, што обезбедува едноставен начин за комуникација со базата на податоци.

Безбедност: Laravel има вградени механизми за безбедност, вклучувајќи ги CSRF заштитата, хешување на лозинки, заштита на сесиите и контрола на пристапот.

Широка заедница и документација: Laravel има активна заедница и обширна документација кои обезбедуваат поддршка и ресурси за програмерите.

Tailwind CSS - Tailwind CSS е рамка за CSS која е прва во корист која обезбедува сеопфатен сет на претходно дефинирани класи за брзо градење на кориснички интерфејси. За разлика од традиционалните CSS рамки кои се фокусираат на обезбедување однапред дизајнирани компоненти, Tailwind CSS се фокусира на обезбедување на голема колекција на класи на алатки што може да се комбинираат за да се создадат сопствени дизајни.

MySQL - MySQL е систем за управување со релациона база на податоци со отворен код (RDBMS) кој широко се користи за складирање и управување со структурирани податоци. Тој е еден од најпопуларните системи за бази на податоци и често се избира поради неговата доверливост, приспособливост и леснотија на користење.

Jetstream - Jetstream обезбедува имплементација за најавување на вашата апликација, регистрација, проверка на е-пошта, автентикација со два фактори, управување со сесии, API преку Laravel Sanctum и опционални функции за управување со тим. Jetstream е дизајниран со користење на Tailwind CSS и нуди ваш избор на Livewire или Inertia скелиња.

Додатни фајлови и компоненти

```
<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">

  <title>Laravel</title>

  <!-- Fonts -->
  <link rel="preconnect" href="https://fonts.bunny.net">
  <link
href="https://fonts.bunny.net/css?family=figtree:400,600&display=swap"
rel="stylesheet"/>

  @vite(['resources/css/app.css', 'resources/js/app.js'])
  @livewireStyles
</head>
```

Во head елементот се импортираат app.js којшто е главниот Javascript фајл и app.css во којшто се импортираат Tailwind компонентите за работа со Tailwind.

```
@tailwind base;
@tailwind components;
@tailwind utilities;
```

Во tailwind.config.js фајлот се конфигурира Tailwind:

```
module.exports = {
  content: [

    './vendor/laravel/framework/src/Illuminate/Pagination/resources/views/*.blade.php',
    './vendor/laravel/jetstream/**/*.blade.php',
    './storage/framework/views/*.php',
    './resources/views/**/*.blade.php',

  ],

  theme: {
    extend: {
      fontFamily: {
        sans: ['Figtree', ...defaultTheme.fontFamily.sans],
      },
    },
  },

  plugins: [require('@tailwindcss/forms'),
require('@tailwindcss/typography')],
};
```

Каде што во content се додаваат кои фајлови да се компајлираат (во кои фајлови ќе се користи Tailwind). Во plugins се импортираат додатоци кои ќе помогнат како форми и typography.

Потоа во package.json се содржат сите зависности на проектот:

```
"devDependencies": {
  "@alpinejs/focus": "^3.10.5",
  "@tailwindcss/forms": "^0.5.2",
  "@tailwindcss/typography": "^0.5.0",
  "alpinejs": "^3.12.2",
  "autoprefixer": "^10.4.7",
  "axios": "^1.1.2",
  "laravel-vite-plugin": "^0.7.2",
  "postcss": "^8.4.14",
  "tailwindcss": "^3.1.0",
  "vite": "^4.0.0"
},
```

Тука може да се видат верзиите на зависностите како на пример tailwindcss верзија 3.1.0 и нагоре.

Во .env фајлот се наоѓа конфигурацијата на Датабазата:

```
DB_CONNECTION=mysql
DB_HOST=127.0.0.1
DB_PORT=3306
DB_DATABASE=insdb
DB_USERNAME=root
DB_PASSWORD=
```

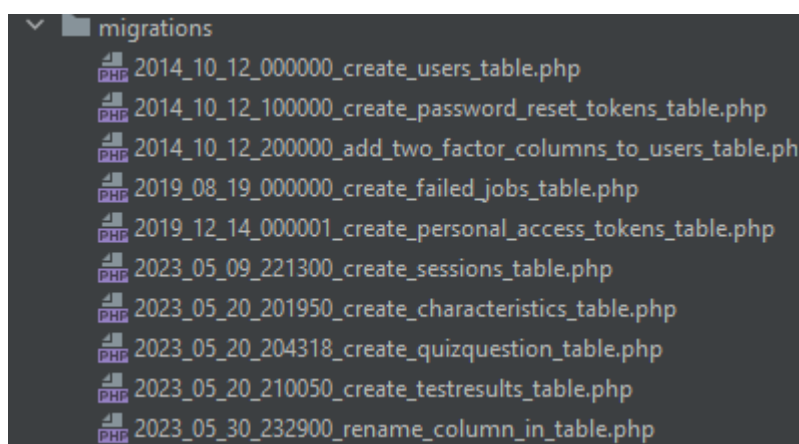
Од ова може да видиме дека базата работи на localhost на порта 3306 и името на базата е insdb а за најава се користи име root и лозинката е празна.

Дополнително во public/media ги има сите слики и видеа кои се користат за дизајнот на проектот.

Структура на БнП

Во Базата на податоци се креирани 4 табели, имено 4те главни табели кои се користат се карактеристики(characteristics), прашања(questions), тест резултати(testresults) и корисници(users).

Освен табелата корисници која се креира автоматски кога го инсталираме Jetstream сервисот за автентикација другите табели се креирани со миграции



Ова се сите миграции направени во проектот.

Структурата на базата е следна:

1. Табелата карактеристики ги состои сите карактеристики кои се тестираат во апликацијата.

	id	name	description
1	1	Adaptable	If you describe a person or animal a...
2	2	Ambitious	Someone who is ambitious has a stron...
3	3	Considerate	Someone who is considerate pays atte...
4	4	Cooperative	If you say that someone is cooperati...
5	5	Friendly	If someone is friendly, they behave ...
6	6	Aggressive	An aggressive person or animal has a...
7	7	Arrogant	Someone who is arrogant behaves in a...
8	8	Deceptive	If something is deceptive, it encour...
9	9	Egotistical	Someone who is egotistic or egotisti...
10	10	Withdrawn	Someone who is withdrawn is very qui...

За секоја карактеристика се чува автоматски генерирано id, име и опис на карактеристиката. Во моментот апликацијата работи со само овие карактеристики, но е направена да е флексибилна така што ако се додадат уште карактеристики без проблем ќе бидат пресметувани и инкорпорирани во резулатите.

2. Табелата Прашања ги содржи сите прашања што се поставуваат на корисниците кога го прават тестот, при што на секој корисник му се покажуваат сите прашања.

	id	description	characteristics
1	8	How well do you adapt to changes in...	[1,4]
2	9	How important is achieving success ...	[2,9]
3	10	How considerate are you of other pe...	[3,5]
4	11	How comfortable are you working col...	[4,10]
5	12	How approachable and warm are you i...	[5,6]
6	13	How often do you seek attention and...	[9,7]
7	14	How skilled are you at concealing y...	[8,2]
8	15	How comfortable are you being arrog...	[7,3]
9	16	How much value do you place on pers...	[10,1]
10	17	How often do you manipulate situati...	[8,6]
11	38	How do you handle challenging situa...	[1,4]
12	39	How do you prioritize your goals an...	[2,3]
13	40	How do you approach social interact...	[5,6]
14	41	How do you handle feedback and crit...	[3,7]
15	42	How do you balance your own needs a...	[9,4]
16	43	How do you adapt to unexpected changes and disruptions?	
17	44	How do you approach problem-solving...	[9,8]
18	45	How do you approach conflicts and d...	[6,2]
19	46	How do you engage with differing op...	[10,5]
20	47	How do you respond to situations wh...	[8,7]

За секое прашање се чува автоматски генерирано id, самото прашање и листа од сите карактеристики, со тоа што листата карактеристики содржи id на сите карактеристики кои ги опфаќа. Едно прашање може да опфати една или повеќе карактеристики.

3. Табелата Тест Резултати ги содржи сите резултати од сите корисници на апликацијата.

	id	user_id	chars_values	created_at	updated_at
1	43	1	{"1":95,"7":90,"3":85,"10":75,"4":65...	2023-06-17 13:21:33	2023-06-17 13:21:33
2	44	1	{"5":90,"7":90,"3":85,"6":85,"1":80,...	2023-06-18 18:35:49	2023-06-18 18:35:49
3	45	1	{"8":90,"7":85,"4":80,"3":80,"1":65,...	2023-06-19 15:02:45	2023-06-19 15:02:45

За еден резултат се чува автоматски генерирано id, корисничко id за корисникот кој го правел тестот и самите резултати.

Во колоната за резултати за секој резултат се чуваат клуч вредност парови за карактеристиките и нивниот соодветен процент во опаѓачки редослед.

4. Табелата Корисници е автоматски генерирана и ги чува сите корисници во апликацијата.

	id	name	email	email_verified_at	password
1	1	test	test@test.com	<null>	\$2y\$10\$V0S1fK2oRmMz.9BphmYNqudV99.TU...
2	3	admin	admin@admin.com	<null>	\$2y\$10\$UMK5BiP/ZUVxuV/P8jSn7uCQaDj7Y...

За секој корисник се чува автоматски генерирано id, име, адреса и хеширана лозинка.

За секоја табела има креирано соодветни модели:

1. Корисник (User) – во којшто е додадена врската со тест резултатите

```
public function testResults()
{
    return $this->hasMany(TestResult::class);
}
```

Ова е 1:M врска со табелата резултати

2. Карактеристика (Characteristic)

```
class Characteristic extends Model
{
    use HasFactory;
    protected $guarded = [];
}
```

3. Прашања (Question)

```
class Question extends Model
{
    use HasFactory;

    protected $table = 'quizquestion';

    protected $fillable = [
        'description',
        'characteristics',
    ];
}
```

Каде се наведува дека се поврзува со табелата quizquestion и се кажува дека е дозволено пополнување на колоните за опис и карактеристики.

4. Тест Резултати (TestResult)

```
class TestResult extends Model
{
    use HasFactory;

    public function user()
    {
        return $this->belongsTo(User::class);
    }

    protected $fillable = [
        'user_id',
        'chars_values',
    ];
    protected $table = 'testresults';
}
```

Тука е наведено дека оваа табела е поврзана со табелата корисник.

Регистрирани Рути

Во апликацијата се регистрирани следните get рути во web.php фајлот:

```
Route::get('/', function () {  
    return view('welcome');  
})->name('welcome');
```

Рута за главната страна, кога се пристапува до / корисникот се пренасочува кон welcome

```
Route::middleware(['auth'])->group(function () {  
    // Routes that require authentication  
    Route::get('/questions',  
'App\Http\Controllers\QuizQuestion@showQuestions')->name('questions');  
    Route::get('/test-results/{id}',  
'App\Http\Controllers\TestResultController@show')  
        ->name('testResult.show');  
    Route::get('/show-profile-results',  
'App\Http\Controllers\TestResultController@showProfileResults')-  
>name('showProfileResults');  
});
```

Ова се сите автентифицирани рути со тоа што не е дозволено пристапување до /questions(рута до страната за прашања), /test-results/{id}(рута до поединечен тест резултат), /show-profile-results (рута до сите резултати на еден корисник).

Ако не автентифициран корисник пристапи до овие рути ќе биде пренасочен до страната за најава.

Исто регистрирана е и post рута:

```
Route::post('/test-result',  
'App\Http\Controllers\TestResultController@createTestResult')-  
>name('createTestResult');
```

Која е наменета за креирање нов резултат.

Homepage

OCEAN Personality Test

[Log in](#) [Register](#)

Welcome...



The ocean personality test is based on the five-factor model, an empirical concept in psychology that measures five pre-dominant personality traits: openness, conscientiousness, extroversion, agreeableness, and neuroticism, making the acronym OCEAN. The ocean personality assessment can be used as a tool to make important talent management decisions as it helps gain in-depth insights into a test-taker's personality. The ocean personality test is based on the five-factor model theory that suggests five broad trait domains as the foundation of different personalities. The model has gone through several iterations by various researchers between the 1960s through the 1990s. The ocean personality test remains to be the most popular personality test in workplace-related situations.

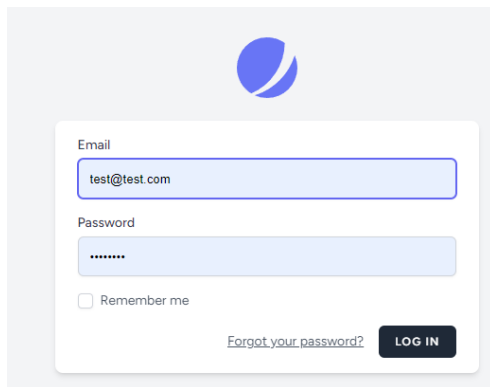
In this test, you will be tested on 2 characteristics from every of the 5 traits in total 10 characteristics.

[Take The Quiz](#)

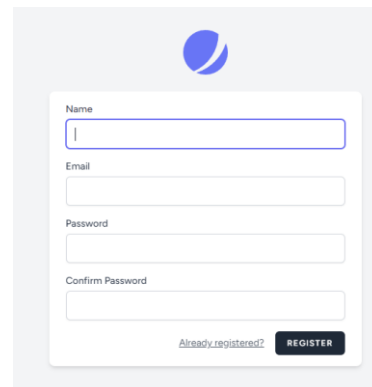
Главните функционалности на Главната страна се автентикацијата и стартување на тестот. Кога корисникот првпат се поврзува со страната има опција да се најави и да го започне тестот. Ако го кликне копчето за почнување на тестот без претходно да се најави ќе биде пренасочен на страницата за најава. Откако корисникот се најавува може да започне со тестот.

Навигациско мени

Најавувањето и Регистрирањето е контролирано од Jetstream. Алатката креира страници за најава и регистрирање коишто се поврзани со копчињата на врвот од навигациското мени.

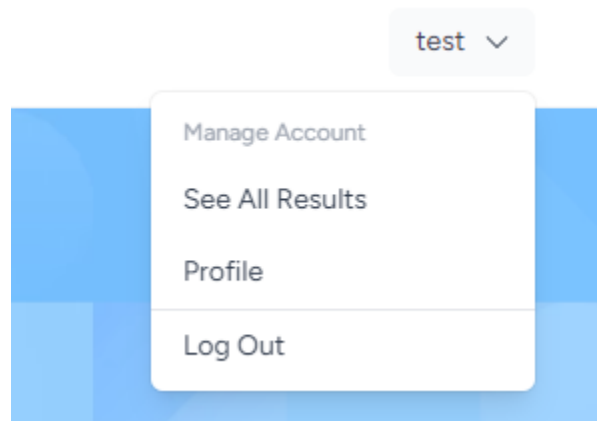


Form for logging in. It includes a blue logo at the top. Below it, there are input fields for 'Email' (containing 'test@test.com') and 'Password' (masked with dots). A 'Remember me' checkbox is present. At the bottom, there is a link 'Forgot your password?' and a 'LOG IN' button.



Form for registering. It includes a blue logo at the top. Below it, there are input fields for 'Name', 'Email', 'Password', and 'Confirm Password'. At the bottom, there is a link 'Already registered?' and a 'REGISTER' button.

После најава на местото на копчињата се појавува паѓачко мени со името на корисникот.



A user profile dropdown menu. At the top, it shows the username 'test' with a downward arrow. Below it, a white menu box contains the following options: 'Manage Account', 'See All Results', 'Profile', and 'Log Out'.

Трите опции на ова мени се да се погледнат сите резултати од тестовите кои ги има направено корисникот, потоа може да се погледне профилот каде може да се менуваат детали на сметката како лозинка и име, и последната опција е да се одјави корисникот.

Страна за прашања

Откако корисникот притиска на копчето за да го започне тестот, преку QuestionController

```
class QuizQuestion extends Controller
{
  public function load()
  {
    return view('partials.questions');
  }

  public function showQuestions()
  {
    $questions = Question::all();

    return view('partials.questions', compact('questions'));
  }
}
```

поточно методот showQuestions се пренасочува на страната /questions и во променливата questions се праќаат сите прашања од база да му се покажат на корисникот.

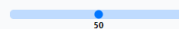
Откако корисникот ќе одговори на сите прашања најдолу кликува на копчето за да се заврши тестот и корисникот се пренасочува на резултатите од овој тест.

OCEAN Personality Test

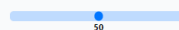
test ▾

Please answer from 0-100

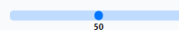
How well do you adapt to changes in your environment?



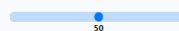
How important is achieving success and recognition in your career?



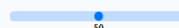
How considerate are you of other people's feelings and needs?



How comfortable are you working collaboratively in a team?



How approachable and warm are you in social interactions?



Кодот на погледот за прикажување на прашањата

```
<div class="container mx-auto mt-24 shadow-xl bg-gray-50">
  <form action="{{ route('createTestResult') }}" method="POST">
    @csrf
    <div class="flex flex-col">

      @foreach($questions as $question)
        <div class="my-12">
          <div class="text-3xl text-center">{{ $question-
>description }}</div>
          </div>
          <div class="flex flex-col items-center justify-center mb-4">
            <input type="range" min="0" max="100" step="1"
name="sliderValues[{{ $question->id }}"
              class="w-72 h-4 bg-blue-200 rounded-md appearance-
none"/>
            <span class="slider-value text-sm font-bold">50</span>
            <input type="hidden" name="questionIds[]" value="{{
$question->id }}">
          </div>
        @endforeach

        <div class="flex justify-center my-8">
          <button type="submit" class="bg-blue-500 hover:bg-blue-700
text-white font-bold py-2 px-4 rounded">
            Submit
          </button>
        </div>

      </div>
    </form>
  </div>
<script>
  // Update the slider value display dynamically
  const sliders = document.querySelectorAll('input[type="range"]');
  sliders.forEach(slider => {
    const sliderValue = slider.parentNode.querySelector('.slider-value');
    slider.addEventListener('input', () => {
      sliderValue.textContent = slider.value;
    });
  });
</script>
```

Внатре во <div>-от, има <form> елемент што го одредува атрибутот на акцијата како "{{ route('createTestResult') }}". Ова значи дека кога формата ќе биде пратена, нејзината акција ќе биде обработена од рутата со име "createTestResult". Вистинската URL адреса поврзана со оваа рута би зависела од рамката или имплементацијата на задниот дел. Директивата @csrf генерира поле за CSRF токен во формата. Ова е безбедносна мера за заштита од напади на прекршување на прекините на сесија. Со секое прашање има и id на прашањето кое ќе помогне во пресметки. Има и јаваскрипт код за ажурирање на вредноста на слајдерот.

Страна за резултати

Она за кое и е намената на системот, односно најважниот дел од системот за крајните корисници, она кое ќе биде од најголемо значење за самите нив, а воедно и причината поради која ќе го користат системот, а и целта на она поради кое ние го изградиве овој систем се самите резултати. Сите чекори кои ги наведовме до сега, целиот код, и сите функционалности можеме да кажеме дека се дел од сложувалката за да се добие крајната слика а тоа е резултатот од самиот тест. По одговарањето на сите прашања, односно едноставното поместување на слајдерот на лево или десно под секое од прашањата. Со кликање на “Submit” одговорите на прашањата се испраќаат на процесирање, односно калкулирање на тоа со колкава застапеност ја имаме секоја од карактеристиките кои ги прати системот со помош на алгоритам кој го имаме имплементирано.

Веднаш по притискање на “Submit” на корисникот му се прикажува страна кој ги содржи резултатите од тестирањето. На самиот почеток можеме да видиме дека имаме датум и саат кога е извршено тестирањето, нешто по што корисникот ќе може да ги идентификува резултатите кои ги има доколку повеќе пати го помине процесот на тестирање. Можеме да забележиме под самиот датум, дека се прикажува и еден дијаграм, или пак конкретно дијаграм од тип “radar”. Доколку видиме во самиот код на view-то кое ги рендерира резултатите, можеме да забележиме “<canvas>” елемент каде е сместен овој дијаграм. До овој елемент се пристапува преку неговото id, се инстанцира нова инстанца од дијаграм, и се доделуваат податоците кои треба да ги искористи за да го изгради овој дијаграм, односно резултатите од тестирањето. На почетокот можеме да видиме дека сме специфицирале кој модел ќе го користиме, односно use

“App\Models\Characteristic”, што ни овозможува пристап до базата на податоци за овој модел, од каде ги земаме имињата на карактеристиките според нивното id, кое го имаме во JSON листата “\$testResult”, каде имаме мапирање id на карактеристика со резултатот од тестирањето за таа карактеристика. Правиме листа од имињата на карактеристиките и освоените бодови за секоја од нив соодветно во две различни низи именувани како “characteristicNames” и “scores” кои ги предаваме како аргументи за форирање на дијаграмот. Овој дијаграм можеме да го сметаме само како сликовит и експресен приказ од резултатите од тестирањето кои корисникот може да ги забележи веднаш по завршување на тестирањето. Ова ќе биде блиц слика за она што корисникот ќе може да го види детално подолу на истата страна. Ова воедно може да биде од голема помош доколку истиот корисник направи повеќе

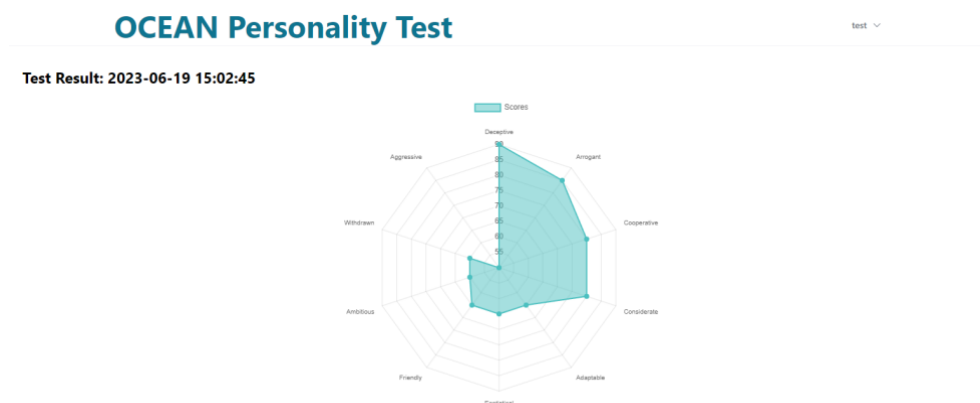
тестови, па во моментот кога ќе сака набрзина да ги погледне сите од нив да може само со отварање на резултатите од тестот да види сликовит приказ за истиот.

```
public function show($id)
{
    // Retrieve the test result from the database based on the ID
    $testResult = TestResult::where('id', $id)
        ->where('user_id', auth()->id())
        ->first();

    if (!$testResult) {
        // Handle the case where the test result is not found
        abort(404);
    }

    // Pass the test result to the view for display
    return view('partials.testresult', ['testResult' => $testResult]);
}
```

Секако само сликовиот приказ нема да биде доволен, потребно е и соодветен краток опис на секоја од карактеристиките, во една традиционална листа која од врвот кон дното ќе ги подреди карактеристиките на индивидуата според нивната јачина одредена од одговорите на прашањата во текот на тестирањето. Тука корисникот може да го види името на карактеристиките, процентуално колку од таа карактеристика е пронајдено во него според одговорите, а веднаш потоа и краток опис на карактеристиката, се со цел подобро да може да му се долови на корисникот за што конкретно се станува збор. Еден пример за сето ова што го набројав ќе може да видите во следните слики.



Ова е приказ на диаграмот, каде делот обоен во плаво е делот со карактеристиките кон кои индивидуата не најблиска според дадените одговори. Ова може да се смета еден вид пресек на карактеристиките до кои е најблиска индивидуата.

Алгоритмот за пресметување на оцена за секоја карактеристика се наоѓа во TestResult Контролерот со име на метод createTestResult. Овој метод се повикува кога се кликнува на копчето за завршување на тестот на страната за прашања.

```
public function createTestResult(Request $request)
{
    // Validate the form input
    $validatedData = $request->validate([
        'questionIds' => 'required|array',
        'questionIds.*' => 'numeric',
        'sliderValues' => 'required|array',
        'sliderValues.*' => 'numeric|min:0|max:100',
    ]);
    // Get the user ID
    $userId = auth()->user()->id;

    // Create an empty array to store the question IDs and slider values
    $characteristicScores = [];
    $characteristicOccurrences = [];
    $questionIds = $validatedData['questionIds'];
    $sliderValues = $validatedData['sliderValues'];
    foreach ($questionIds as $index => $questionId) {
        $question = Question::find($questionId);
        $chars = $question->characteristics;
        $characteristicIds = json_decode($chars, true);
        $sliderValue = intval($sliderValues[$questionId]);
        $score = $this->calculateScore($sliderValue);

        foreach ($characteristicIds as $characteristicId) {
            if (!isset($characteristicScores[$characteristicId])) {
                $characteristicScores[$characteristicId] = 0;
                $characteristicOccurrences[$characteristicId] = 0;
            }
            $characteristicScores[$characteristicId] += $score;
            $characteristicOccurrences[$characteristicId]++;
        }
    }

    foreach ($characteristicScores as $characteristicId => $score) {
        $maxPossibleScore = $characteristicOccurrences[$characteristicId] *
5;
        $percentage = ($score / $maxPossibleScore) * 100;
        $characteristicScores[$characteristicId] = intval($percentage);
    }
    arsort($characteristicScores);
    // Create and store the TestResult object in the database
    TestResult::create([
        'user_id' => $userId,
        'chars_values' => json_encode($characteristicScores),
    ]);
    $testResult = TestResult::where('user_id', $userId)->latest()->first();
    return redirect()->route('testResult.show', ['id' => $testResult->id]);
}
```

Во методот се земаат сите вредности на прашањата и id на прашањата. Потоа во циклусот се зема секое прашање од база и му се земаат карактеристиките, па за секоја карактеристика во листа се ставаат оцените пресметани со следниот алгоритам,

```
public function calculateScore($sliderValue)
{
    if ($sliderValue >= 1 && $sliderValue <= 20) {
        return 1;
    } elseif ($sliderValue > 20 && $sliderValue <= 40) {
        return 2;
    } elseif ($sliderValue > 40 && $sliderValue <= 60) {
        return 3;
    } elseif ($sliderValue > 60 && $sliderValue <= 80) {
        return 4;
    } else {
        return 5;
    }
}
```

На крајот се претвараат во проценти, се сортира листата поп оцените опаѓачки и се креира резултатот во база кој потоа се праќа на новиот поглед да му се прикаже на корисникот.

Ова е делот кој се наоѓа веднаш под диаграмот, каде во вид на листа подредена од најизразена, кон помалку изразена карактеристика кај индивидуата. Можеме да забележиме дека најпрво доаѓа името на карактеристиката болдирано, веднаш до него бројка која ја изразува

процентуалната застапеност од таа карактеристика кај индивидуата во интервал од нула до сто, а веднаш под тоа следува краток опис за истата.

Сето ова е дел од “testresult.blade.php” погледот, каде е сместен целиот код за прикажување на оваа страна.

All Characteristic scores	
Deceptive: 90	If something is deceptive, it encourages you to believe something which is not true.
Arrogant: 85	Someone who is arrogant behaves in a proud, unpleasant way towards other people because they believe that they are more important than others.
Cooperative: 90	If you say that someone is cooperative, you mean that they do what you ask them to without complaining or arguing.
Considerate: 80	Someone who is considerate pays attention to the needs, wishes, or feelings of other people.
Adaptable: 85	If you describe a person or animal as adaptable, you mean that they are able to change their ideas or behaviour in order to deal with new situations.
Egotistical: 85	Someone who is egotistic or egotistical behaves selfishly and thinks they are more important than other people.
Friendly: 85	If someone is friendly, they behave in a pleasant, kind way, and like to be with other people.
Ambitious: 60	Someone who is ambitious has a strong desire to be successful, rich, or powerful.
Withdrawn: 60	Someone who is withdrawn is very quiet, and does not want to talk to other people.
Aggressive: 50	An aggressive person or animal has a quality of anger and determination that makes them ready to attack other people.

Страна за сите резултати

Освен можноста корисникот да го види резултатот од тестот кој штотуку го направил, на корисникот му се нуди можност да ги види и резултатите од сите тестови кои досега ги има направено, односно листа од сите тестови подредени според датумот. Сето ова е дел од **“profilerevents.blade.php”** погледот, каде според кодот можеме да видиме една едноставна структура каде на врвот имаме опис на она што ни се нуди на страната односно **“All previous results”**, каде преку функцијата во **“TestResultController.php”** контролерот - **“showProfileResults()”** од база ги земаме резултатите за корисникот кој моментално е најавен според неговото id, и ги праќаме во view-то именувано како **“profilerevents.blade.php”** каде ја изминуваме листата од тестови кои ги направил дадениот корисник, поставуваме елемент кој ќе овозможи при клик на името на некој од тестовите, односно во овој случај тоа е датумот на кој тестот е направен да се повика функција во контролерот кон која ќе се предаде id од резултатот во базата и истотот ќе биде прикажан на истиот начин како и во моментот кога корисникот го завршувал тестот, односно ќе има увид на деталите од резултатот за тестот кој бил направен на наведениот датум и час. На врвот корисникот може да види информација и за тоа колку вкупно тестови има направено досега, односно за бројот на тестови зачувани во системот за него, нешто што би било од голема корист доколку корисникот долго го користи системот и има направено голем број тестови, а воедно и сака брз увид за тоа колку пати бил тестиран. Пример за овој поглед можеме да видиме во сликата подолу, каде можете да го видите сето она кое досега го опишавме.

OCEAN Personality Test

test ▾

All previous results

You have taken 3 tests.

1 2023-06-19 15:02:45

2 2023-06-18 18:35:49

3 2023-06-17 13:21:33

Кодот за креирање на погледот за сите резултати е следен

```
public function showProfileResults()  
{  
    $userId = auth()->id();  
    $testResults = TestResult::where('user_id', $userId)->latest()->get();  
    return view('partials.profilereults', compact('testResults'));  
}
```

Овој метод се наоѓа во TestResult Контролерот и го автентифицира сегашниот корисник и му ги зема сите резултати од базата за да му се прикажат во погледот.

Примени на проектот

Како што веќе кажавме намената на оваа апликација е корисниот да се спознае себе си на еден поинаков начин, одговарајќи на специфични прашања кои имаат удел во некоја од карактеристиките над кои ќе биде извршено проценувањето на карактерот на индивидуата. Целта на прашањата е корисникот да добие несекојдневни прашања, каде всушност главната цел е да се оцени како тој ќе одговори на прашањето за дадената ситуација, односно неговото размислување за дадена ситуација, по што во однос на тоа како одговорил на прашањето се доделува јачина на некоја од карактеристиките. Во системот имаме повеќе карактеристики, поврзани со повеќе прашања, односно бројот на прашања е многу поголем од бројот на карактеристики за кои водиме сметка во системот. Може да се каже на една карактеристика одговараат некаде околу три прашања. Со секој одговор добиваме, односно намалуваме некоја карактеристика, односно ја зголемуваме или намалуваме нејзината јачина во крајната проценка за карактерот на индивидуата. Секако мора да се напомене дека крајниот резултат не мора нужно да го одредува карактерот на индивидуата, туку тој се смета за проценка на нековиот карактер која може да се земе со одреден степен на точност, односно одредено ниво на доверба. Колку повеќе тестирања направиме за една индивидуа толку ќе биде поголема релевантноста на особините кои таа индивидуа ги поседува, односно ќе може да се направи споредба на претходно добиените резултати и да се увиди дали во неколку наврати ги имаме процентите за дадените карактеристики со приближно еднаква вредност. Како добра препорака на оние кои ќе го користат системот од аспект на прибирање на информации за индивидуите, односно оние на кои ќе им бидат од значење при донесување на некаква одлука за индивидуата се препорачува доколку се има можност и за средба со индивидуата еден на еден при што стручно лице од областа на човечки ресурси ќе може да се запознае со индивидуата со личен контакт, по што ќе може со помош на претходно добиените резултати од тестот кој го направила индивидуата да добие порелевантна и поиздржана слика за индивидуата која е цел на ова тестирање.

Примената на овој систем може да се види во области каде сакаме да видиме дали одредена индивидуа добро би се вклопила во тим составен од веќе познати карактери, односно да се олесни и воедно да се добие пократко време потребно индивидуата да се пронајде во средината во која претендира да биде. Типична примена во реалниот живот може да биде моментот кога имае компанија со повеќе тимови, каде секој тим е составен од карактери кои се блиски до себе, односно ова би била средина каде се поставува акцент пред сè на тоа индивидуите да имаат поголемо чувство на припадност во средината, кое воедно би имплицирало и поголема продуктивност и успешност на проектите на кои работи тимот. Па во една ваква ситуација овој тест би бил од голема помош при одлуката во кој тим можеме најмногу да ја пронајдеме индивидуата, и да можеме да бидеме уверени со големо ново на сигурност дека истата би била срдна и најдобро би се вклопила во тимот како целина. Исто така друга примена на системот е за компании со голем број на вработени, а воедно и вработени кои работат од дома, односно интернационални компании каде вработените се можеби и од различни држави, па потребен е

начин за да се направи проценка на карактерот на секоја нова индивидуа со цел да се добие подобра слика за тоа дали индивидуата добро би се вклопила во веќе постоечките тимови, каде секако и комуникацијата е доста важна со оглед на тоа дека поголемиот дел од вработените припаѓаат на различни култури.

Прашањата, како и карактеристиките кои се следат со нивното одговарање можат да бидат променети/дополнети во секое време, со цел поверодостојни резултати на крајот од тестирањето. Секако ова е почетна верзија кој понатаму континуирано ќе се надополнува со нови прашања поврзани со некоја од постоечките, или пак нова карактеристика која би се додала, а се со цел да добиеме подобар мечинг на прашањата и карактеристиките, односно прашањето да биде многу поконкретно, и многу потесно поврзано со одредна карактеристика на индивидуата. Исто така прашањата ќе бидат променувањани во прашања кои ќе можат да извлечат од оној кој се тестира што е можно повеќе од неговите вистински особини, односно од неговото “лично јас”. Сето ова подобрување ќе се врши со текот на времето, и во однос на повратниот резултат од оние кои имале потреба од системот за тоа колку била успешноста на истиот.

Системот нуди екстремно лесен за воочување и разбирање на можностите кои ги уди, со тоа што од корисникот се бара минимален напор за да отовори нов тест, да го реши со едноставно поместување на слејдер кон лево или десно, како и лесен пристап до сите претходни резултати од претходни тестирања. Системот е интуитивен и при негово градење е ставено посебен акцен на тоа да од корисникот не се бара преголема експертиза за да го совлада системот, а воедно и да не му се одземе многу од времето за време на неговото користење. Дизајнот е чист, користени се лесни бои, најчесто бела, црна и сина, бои кои не би биле нападни за корисникот. Не може да се забележат делови каде се поставени не потребни информации или не потребни функционалности, кои би го натрупувале системот и би го направиле не прегледен за корисникот.

Користена Литература

- <https://laravel.com/docs/10.x>
- <https://tailwindcss.com/>
- <https://mettl.com/glossary/o/ocean-personality-test/>
- <https://www.scienceofpeople.com/personality/>