

# Jaka postać w League of Legends najbardziej do Ciebie pasuje?

Dominik Skóra 42086

Kacper Preyzner 42084

## 1. Wprowadzenie

Witaj w naszym innowacyjnym systemie eksperckim, który został stworzony z myślą o entuzjastach League of Legends! Nasz system poświęcony jest wyborowi idealnej postaci w tej popularnej grze, przy użyciu prostego, a zarazem skutecznego quizu. League of Legends oferuje ogromną różnorodność postaci, z różnymi umiejętnościami i cechami. Dzięki naszemu systemowi, proces wyboru staje się jeszcze bardziej intuicyjny. W trakcie krótkiego quizu, odpowiadasz na proste pytania, wykluczając lub dodając określone umiejętności i cechy postaci. Na końcu drzewa decyzyjnego otrzymasz rekomendację, która idealnie dostosowuje się do Twojego stylu gry. Dlaczego wybraliśmy ten temat? Po pierwsze, League of Legends cieszy się ogromną popularnością, a my chcemy ułatwić graczom proces wyboru postaci. Po drugie, nasza pasja do tej gry skłoniła nas do stworzenia narzędzia, które sprawi, że doświadczenie z LoL stanie się jeszcze bardziej satysfakcjonujące.

## 2. Algorytm

- a) Wybieramy linie jaka nas interesuje
- b) Wyświetlane jest pierwsze pytanie wraz z dwiema odpowiedziami, na naszym ekranie
- c) W zależności od odpowiedzi idziemy na odpowiednią gałąź drzewa decyzyjnego
- d) Po udzieleniu wszystkich odpowiedzi program przesuwa nas do Controllera który pobiera informacje z ResultList.php i odpowiada za ukazanie nam jaka postać jest najbardziej zbliżona do cech które wybraliśmy

## 3. Kod Programu

## ResultList.php

```
public function getLucian()
{
    $name = "Lucian";
    $linkToImage = "https://ddragon.leagueoflegends.com/cdn/img/champion/splash/Lucian_0.jpg";
    $description = "The Purifier";
    return new Result($name, $linkToImage, $description);
}

public function getVayne()
{
    $name = "Vayne";
    $linkToImage = "https://ddragon.leagueoflegends.com/cdn/img/champion/splash/Vayne_0.jpg";
    $description = "The Night Hunter";
    return new Result($name, $linkToImage, $description);
}

public function getVarus()
{
    $name = "Varus";
    $linkToImage = "https://ddragon.leagueoflegends.com/cdn/img/champion/splash/Varus_0.jpg";
    $description = "The Arrow of Retribution";
    return new Result($name, $linkToImage, $description);
}

public function getSamira()
{
    $name = "Samira";
    $linkToImage = "https://ddragon.leagueoflegends.com/cdn/img/champion/splash/Samira_0.jpg";
    $description = "The Desert Rose";
    return new Result($name, $linkToImage, $description);
}

public function getSivir()
{
    $name = "Sivir";
    $linkToImage = "https://ddragon.leagueoflegends.com/cdn/img/champion/splash/Sivir_0.jpg";
    $description = "The Battle Mistress";
    return new Result($name, $linkToImage, $description);
}

public function getJinx()
{
    $name = "Jinx";
    $linkToImage = "https://ddragon.leagueoflegends.com/cdn/img/champion/splash/Jinx_0.jpg";
    $description = "The Loose Cannon";
    return new Result($name, $linkToImage, $description);
}

public function getKalista()
{
    $name = "Kalista";
    $linkToImage = "https://ddragon.leagueoflegends.com/cdn/img/champion/splash/Kalista_0.jpg";
    $description = "The Spear Of Vengeance";
    return new Result($name, $linkToImage, $description);
}

public function getKaiSa()
{
    $name = "Kai'Sa";
    $linkToImage = "https://ddragon.leagueoflegends.com/cdn/img/champion/splash/KaiSa_0.jpg";
```

## MainController.php

```
MainController.php X
app > Http > Controllers > MainController.php
1  <?php
2
3  namespace App\Http\Controllers;
4
5  use Illuminate\Http\Request;
6  use App\Classes\Question;
7
8  class MainController extends Controller
9  {
10
11     public function index()
12     {
13         $question = $this->getDamageTypeQuestion();
14         $questionString = $question->getQuestionString();
15         //dd($question);
16         $answers = $question->getAnswers();
17
18
19         return view('main', compact('questionString', 'answers'));
20     }
21     public function show(Request $request)
22     {
23         dd($request->input('selectedAnswer'));
24     }
25 }
```

## Przykładowe tworzone drzewo Drzewo Adc

```
Adc.php X
app > Traits > Adc.php
1  <?php
2
3  namespace App\Traits;
4
5  use App\Classes\Answer;
6  use App\Classes\MultipathQuestion;
7  use App\Classes\Question;
8  use App\Traits\ResultList;
9
10 trait Adc
11 {
12     use ResultList;
13     private $Adc = [];
14     public function getAdc()
15     {
16
17         $this->createResultList();
18         $this->createQuestionList();
19
20         return $this->Adc;
21     }
22
23     private function createResultList()
24     {
25         $this->createTristana();
26         $this->createZiggs();
27         $this->createTwitch();
28         $this->createVayne();
29         $this->createVarus();
30         $this->createLucian();
31         $this->createMissFortune();
32         $this->createEzreal();
33         $this->createYasuo();
34         $this->createSivir();
35         $this->createCaitlyn();
36         $this->createDraven();
37         $this->createJinx();
38         $this->createKalista();
39         $this->createKaisa();
40         $this->createSamira();
41         $this->createAshe();
42         $this->createJihn();
43         $this->createNilah();
44         $this->createAphelios();
45         $this->createKogmaw();
46         $this->createKayah();
47         $this->createZeri();
48         $this->createSona();
49     }
50 }
```

```

Traits > Adc.php
    $this->createADC_AD_Question();
    //-----AP ADC-----
    $this->createADC_APDMG_Long_Question();
    $this->createADC_APDMG_Short_Question();
    $this->createADC_APDMG_Question();
    $this->createADC_AP_Question();
    $this->createEntryADCQuestion();
}

private function createVayne()
{
    $this->Adc[] = $this->getVayne();
}
private function createTristana()
{
    $this->Adc[] = $this->getTristana();
}
private function createZiggs()
{
    $this->Adc[] = $this->getZiggs();
}
private function createTwitch()
{
    $this->Adc[] = $this->getTwitch();
}
private function createVarus()
{
    $this->Adc[] = $this->getVarus();
}
private function createLucian()
{
    $this->Adc[] = $this->getLucian();
}
private function createMissFortune()
{
    $this->Adc[] = $this->getMissFortune();
}
private function createEzreal()
{
    $this->Adc[] = $this->getEzreal();
}

private function createYasuo()
{
    $this->Adc[] = $this->getYasuo();
}
private function createSivir()
{
    $this->Adc[] = $this->getSivir();
}
private function createCaitlyn()
{
    $this->Adc[] = $this->getCaitlyn();
}

private $createADC_AP_Question;
public function createADC_AP_Question()
{
    $questionString = "What type of damage is adc supposed to do?";
    $answers = [];
    $answer = new Answer('Prefer to be typical AP DMG', $this->createADC_APDMG_Question);
    $answers[] = $answer;
    $answer = new Answer('Want to be Hybrid AP/AD DMG', $this->createADC_Hybrid_Question);
    $answers[] = $answer;
    $questionToAdd = new MultipathQuestion($answers, $questionString);
    $this->Adc[] = $questionToAdd;
    $this->createADC_AP_Question = $questionToAdd;
}

//-----AP/APDMG-----

private $createADC_APDMG_Question;
public function createADC_APDMG_Question()
{
    $questionString = "What range ap adc?";
    $answers = [];
    $answer = new Answer('Short range', $this->createADC_APDMG_Short_Question);
    $answers[] = $answer;
    $answer = new Answer('Long range', $this->createADC_APDMG_Long_Question);
    $answers[] = $answer;
    $questionToAdd = new MultipathQuestion($answers, $questionString);
    $this->Adc[] = $questionToAdd;
    $this->createADC_APDMG_Question = $questionToAdd;
}

//-----AP/APDMG/Short-----

```

```

646 private $createADC_Hybrid_GoodvsTanks_Question;
647 public function createADC_Hybrid_GoodvsTanks_Question()
648 {
649     $questionString = "Choose which adc hybrid good at tanks suits you better";
650     $answers = [];
651     $answer = new Answer('Beats tanks in percentage, amazing attack speed and a very large range of skills', $this->GetKog);
652     $answers[] = $answer;
653     $answer = new Answer('Oneshots tanks under ap, hybrid hits powerful damage, has a stun that extends to others', $this->GetKog);
654     $answers[] = $answer;
655     $questionToAdd = new MultipathQuestion($answers, $questionString);
656     $this->Adc[] = $questionToAdd;
657     $this->createADC_Hybrid_GoodvsTanks_Question = $questionToAdd;
658 }
659
660
661 public function getQuestionByQuestionString($questionString)
662 {
663     $questions = $this->getAdc();
664     //dd($questions);
665     foreach ($questions as $question) {
666         //dd($question->questionString);
667         if ($question->questionString == $questionString) {
668             return $question;
669         }
670     }
671 }
672
673 }

```

## Jak obsługiwane są pytania

```

pytania.blade.php X
resources > views > livewire > pytania.blade.php
1 <div class="fixed w-screen min-h-screen text-white font-mono flex flex-col items-center justify-center" style="background-color: #1a202c; color: white;">
2 <div class="text-4xl mb-11">{{ $questionString }}</div>
3 <div class="rounded-xl p-5 h-64 w-1/3 flex-col justify-center" style="background-color: rgba(0, 0, 0, 0.8);">
4
5     @if ($finished)
6         <p class="flex justify-center mb-10 capitalize">{{ $description }}</p>
7         
8         <a href="/main" class="hover:text-orange-400 capitalize mt-5 self-center w-full duration-300">try again</a>
9     @endif
10
11     <div class="grid grid-cols-2 h-full w-full duration-300">
12         @if (! $finished)
13             <button wire:click="handleClick('{{ $answerA }}')"
14                 class="text-3xl text-center text-white flex justify-center items-center hover:text-orange-400 duration-300">{{ $answerA }}</button>
15             <button wire:click="handleClick('{{ $answerB }}')"
16                 class="text-3xl text-center text-white flex justify-center items-center hover:text-orange-400 duration-300">{{ $answerB }}</button>
17         @endif
18     </div>
19 </div>
20 </div>
21

```

## Wybór drzewa postaci

```

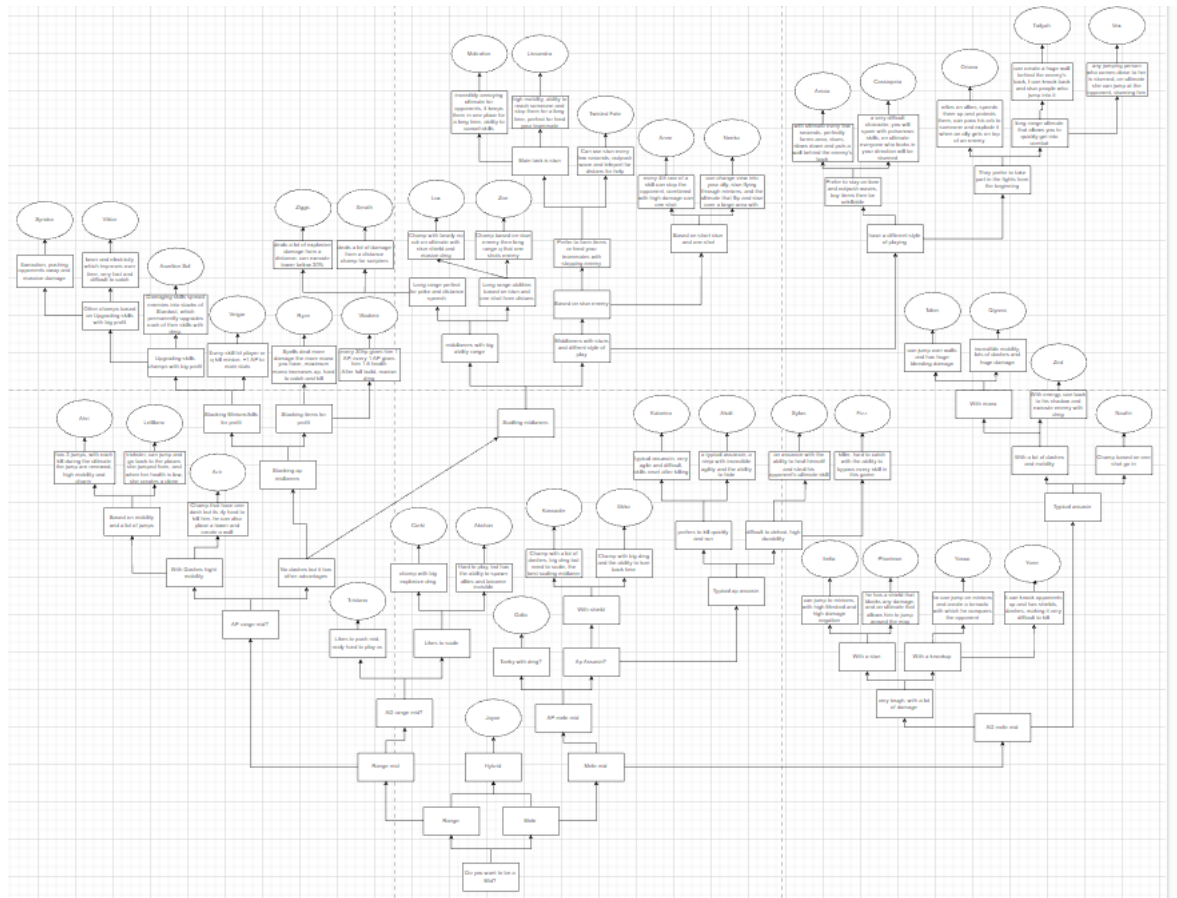
</head>
<body>
    <div class="background-behind"></div>
    <div class="container">
        <div class="button-container">
            <a href="pytania" class="button" style="top: 140px; left: 190px;">
                
                <div class="overlay">
                    <div class="overlay-text">Top</div>
                </div>
            </a>
            <a href="pytaniamid" class="button" style="top: 400px; left: 450px;">
                
                <div class="overlay">
                    <div class="overlay-text">Mid</div>
                </div>
            </a>
            <a href="pytaniaadc" class="button" style="top: 650px; left: 800px;">
                
                <div class="overlay">
                    <div class="overlay-text">ADC</div>
                </div>
            </a>
            <a href="pytaniajungle" class="button" style="top: 350px; left: 250px;">
                
                <div class="overlay">
                    <div class="overlay-text">Jungle</div>
                </div>
            </a>
            <a href="pytaniasupport" class="button" style="top: 650px; left: 700px;">
                
                <div class="overlay">
                    <div class="overlay-text">Support</div>
                </div>
            </a>
            <a href="/" class="back-button">Back to Main Page</a>
        </div>
    </div>

```

## Drzewa ekspertowe

### Rola Top

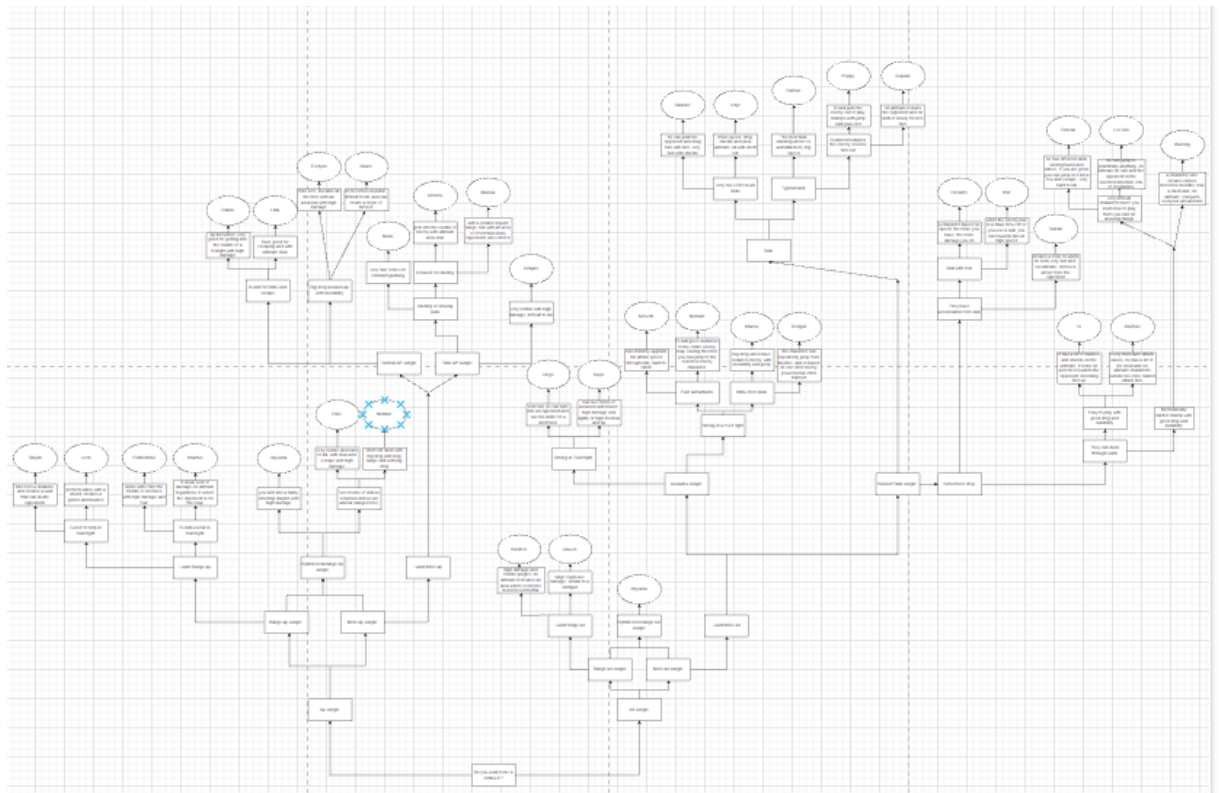
## Rola Mid



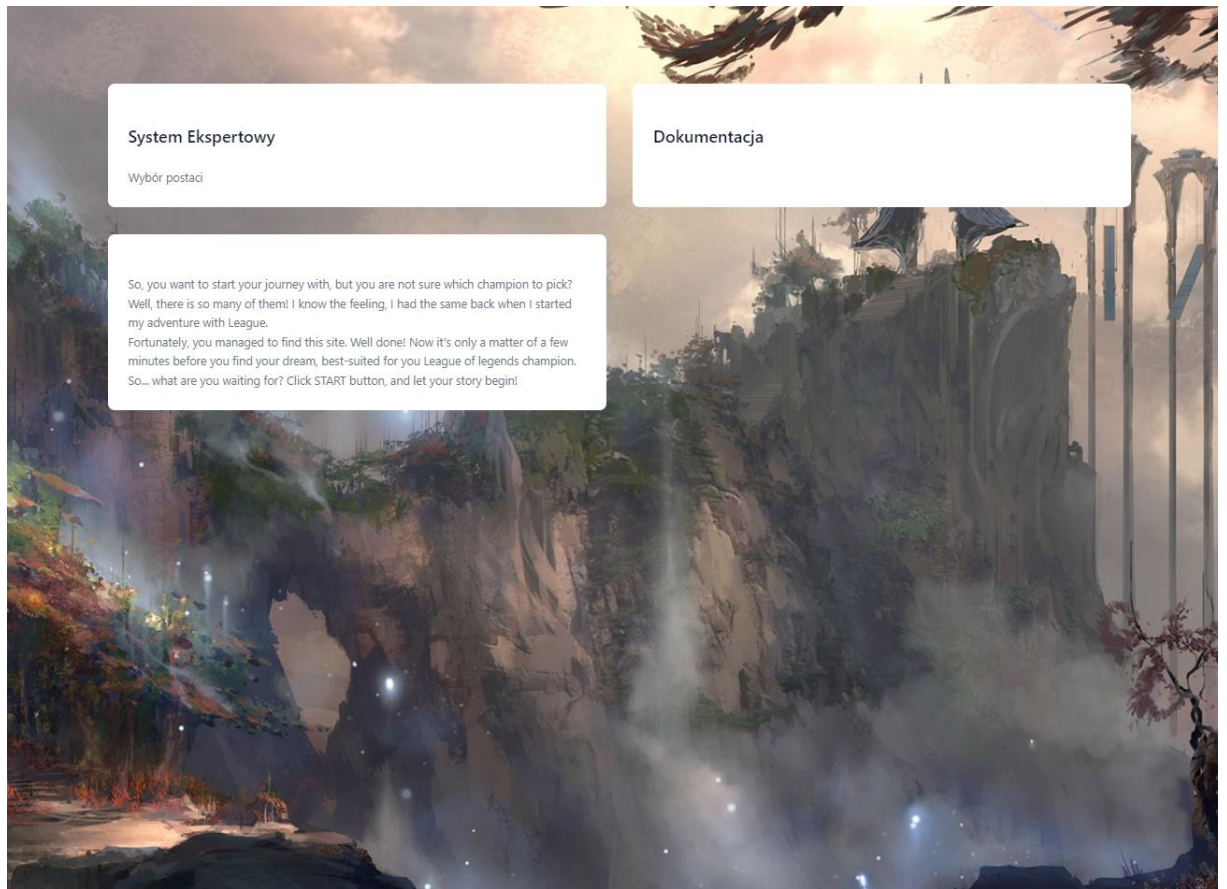
[illegible]



## Rola Jungle



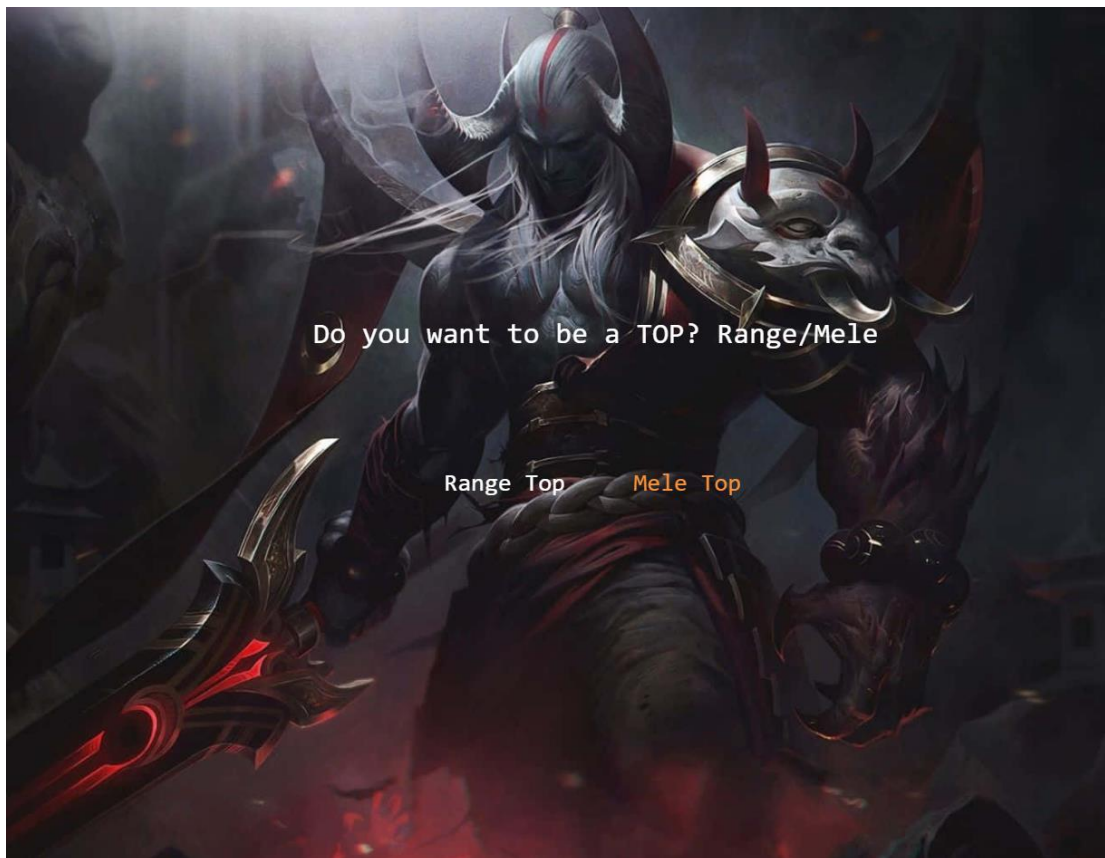
## 4. Interfejs użytkownika



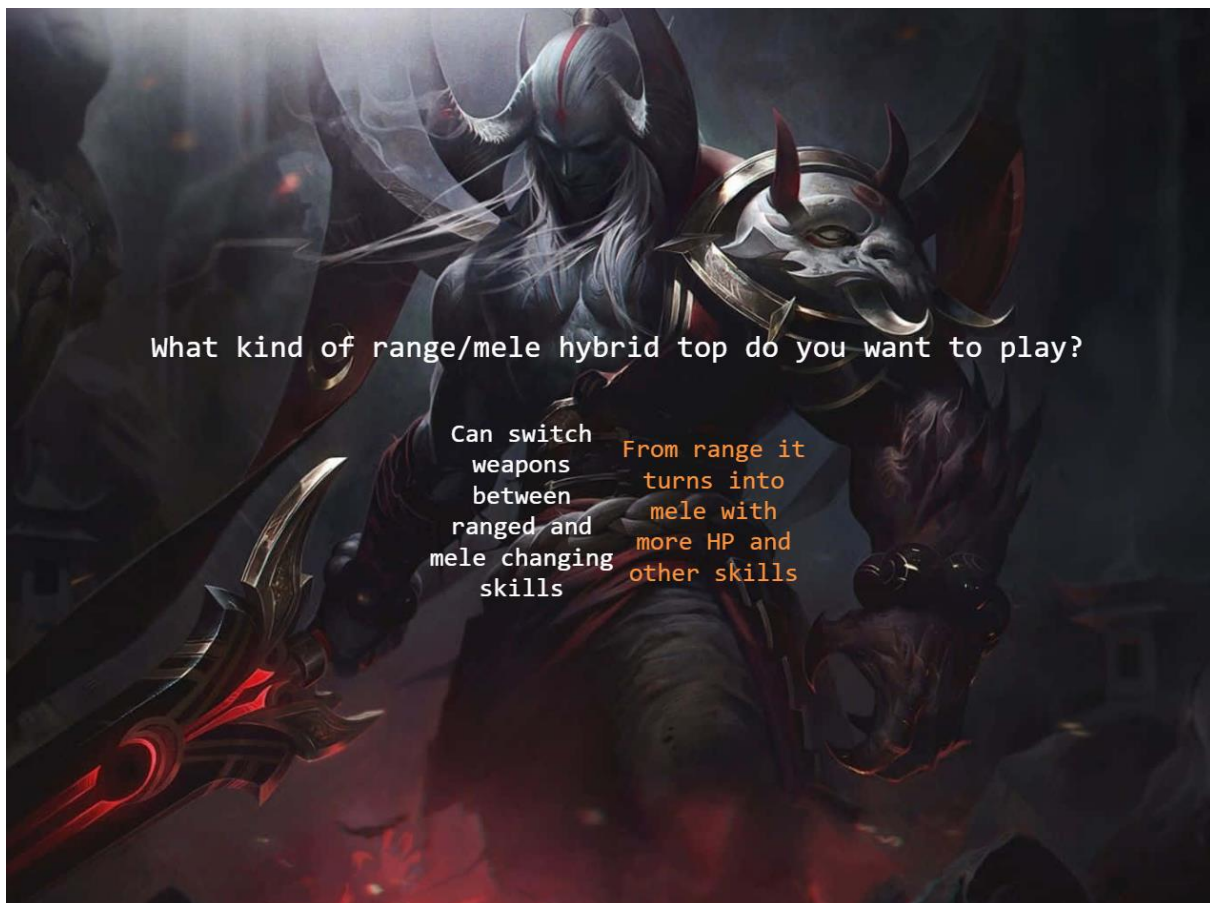
## Wybór drzewa



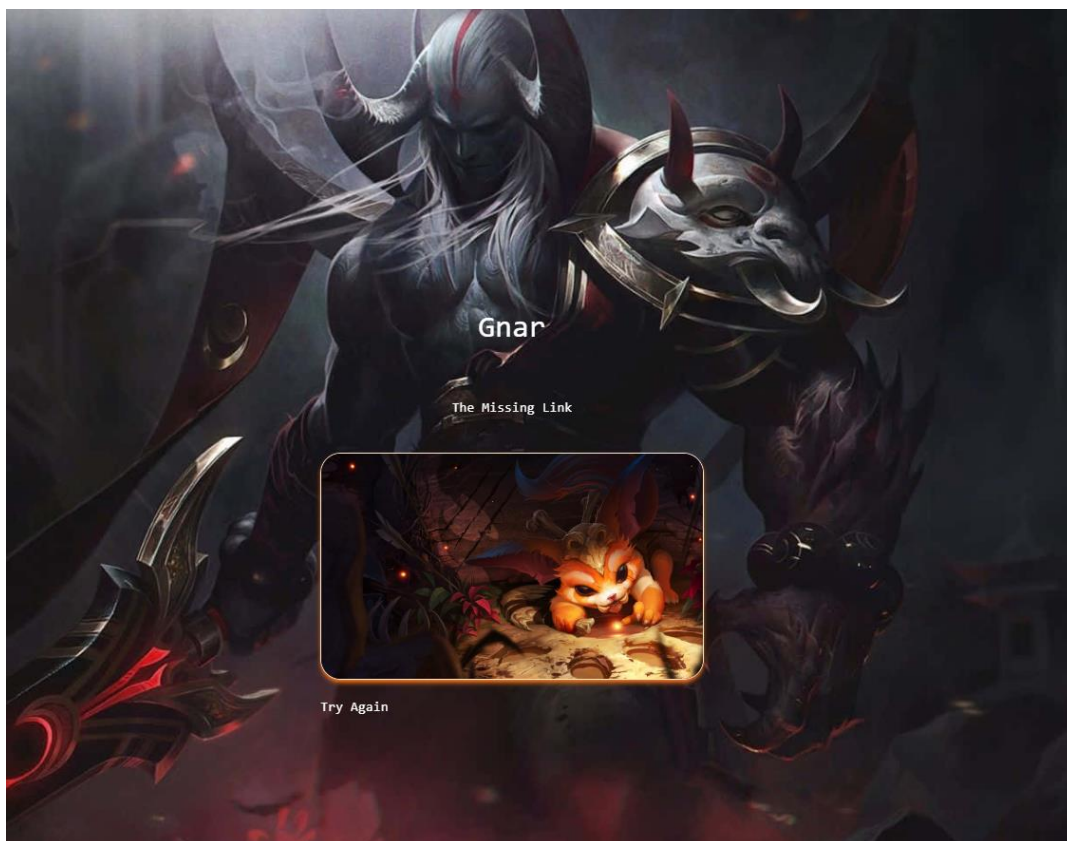
## Przykładowe drzewo Top







## Wyświetlenie wyniku



## 5. Wnioski

Praca nad programem nie sprawiła nam trudności ponieważ mamy już doświadczenie, wiedzę ekspercką w tym temacie oraz dużo chęci, jedynym problemem była ilość kodu do napisania z racji na ilość postaci w league of legends, zaimplementowaliśmy wszystkie 159 postaci, może nie było to ciężkie albo bardzo monotonne ponieważ mieliśmy do zrobienia 5 różnych drzew same tworzenie poprawnego schematu zajęło min 2h a implementacja go wynosiła średnio 1000 linijek kodu, więc była to długa i monotonna praca. Zarówno z wdrożeniem jak i z budową logiki programu nie było problemów.