

Narzędzia sztucznej inteligencji

|  |  |  |  |
| --- | --- | --- | --- |
| **Kierunek studiów:** | *Informatyka* | **Rok studiów:** | *III* |
| **Numer grupy:** | *C4* | | |
| **Rok akademicki:** | *2014/2015* | **Semestr:** | *VI* |

|  |  |
| --- | --- |
| Temat: |  |
| System ekspertowy – Wybór komputera przenośnego firmy Asus. (sprawozdanie nr 2) | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Lp.** | **Nr indeksu** | **Imię i nazwisko** | **Data oddania I** | **Data oddania II** | **OCENA** |
| 1. | 86316 | Radosław Szymański | 05.05.2015 |  |  |
| 2. | 84123 | Artur Stelmach |  |

|  |  |  |
| --- | --- | --- |
| **Termin zajęć:** | | **Prowadzący:** |
| dzień: | *Wtorek* | Dr inż. Szczepan Paszkiel |
| godzina: | *7:30* |

1. **Schemat blokowy systemu eksperckiego.**





1. **Baza danych - ERD.**
2. **GUI systemu eksperckiego.**
3. **Kwestia modelowania ekspertyz.**

Dla każdego laptopa względem każdej odpowiedzi udzielonej przez użytkownika

sprawdzamy jakiego typu jest pytanie. Jeśli pytanie jest typu logiki boolowskiej sprawdzamy czy odpowiedź użytkownika nie zgadza się z encją **LaptopBoolValues** dla danego laptopa usuwamy go z kolekcji rozpatrywanych laptopów. Jeśli pytanie jest typu logiki rozmytej pobieramy odpowiednią encję **LaptopFuzzyValues** mnożymy jego wartość przez wagę odpowiedzi użytkownika i dodajemy wynik do całkowitej punktacji laptopa. Następnie zwracamy posortowaną kolekcję.

foreach (var answer in model.Answers)

{

foreach (var laptop in context.Laptops)

{

var question = context.Questions

.FirstOrDefault(x => x.Id == answer.Key);

if (question.GetType() == typeof(BoolQuestion))

{

var laptopAnswerValue = context.LaptopBoolValues

.Where(x => x.BoolQuestionId == answer.Key)

.Where(x => x.LaptopId == laptop.Id)

.Select(x => x.Value)

.FirstOrDefault();

if (Convert.ToInt32(laptopAnswerValue) != answer.Value)

{

if (laptopsWithSummary.ContainsKey(laptop))

laptopsWithSummary.Remove(laptop);

}

}

else

{

var laptopAnswerValue = context.LaptopFuzzyValues

.Where(x => x.FuzzyQuestionId == answer.Key)

.Where(x => x.LaptopId == laptop.Id)

.Select(x => x.Value)

.FirstOrDefault();

if (laptopsWithSummary.ContainsKey(laptop))

laptopsWithSummary[laptop] += answer.Value \* laptopAnswerValue;

}

}

}

return View(laptopsWithSummary.OrderByDescending(x => x.Value)

.Select(x => x.Key));