ДОДАТОК А

Приклади коду

```php
class RecommendationFacade extends Facade
{
    2 usages
    protected $caseBasedService;
    2 usages
    protected $constraintBasedService;

    1 usage
    public function __construct(
        CaseBasedRecommendationService $caseBasedService,
        ConstraintBasedRecommendationService $constraintBasedService
    ) {
        $this->caseBasedService = $caseBasedService;
        $this->constraintBasedService = $constraintBasedService;
    }

    no usages
    public function getRecommendations(array $attributes, array $constraints, array $weights): array
    {
        $caseBasedRecommendations = $this->caseBasedService
            ->getRecommendations($attributes, $weights);
        $constraintBasedRecommendations = $this->constraintBasedService
            ->getRecommendations($constraints, $weights);

        return [
            'case_based' => $caseBasedRecommendations,
            'constraint_based' => $constraintBasedRecommendations,
        ];
    }
}
```

Приклад 1 – Код фасаду RecommendationFacade

```php
class RecommendationController extends Controller
{
    2 usages
    protected $recommendationFacade;

    1 usage
    public function __construct(RecommendationFacade $recommendationFacade)
    {
        $this->recommendationFacade = $recommendationFacade;
    }

    1 usage
    public function getRecommendations(Request $request)
    {
        $attributes = $request->input( key: 'attributes');
        $constraints = $request->input( key: 'constraints');
        $weights = $request->input( key: 'weights');

        $recommendations = $this->recommendationFacade
            ->getRecommendations($attributes, $constraints, $weights);

        return response()->json($recommendations);
    }
}
```

Приклад 2 – Код контроллеру

```php
class ConstraintBaseFootballer extends Model
{
    protected $fillable = [...];

    no usages
    public function euclideanDistance($vector1, $vector2)
    {
        $distance = 0;
        for ($i = 0; $i < count($vector1); $i++) {
            $distance += pow( num: $vector1[$i] - $vector2[$i],  exponent: 2);
        }
        return sqrt($distance);
    }

    no usages
    public function cosineSimilarity($vector1, $vector2)
    {
        $dotProduct = 0;
        $magnitude1 = 0;
        $magnitude2 = 0;
        for ($i = 0; $i < count($vector1); $i++) {
            $dotProduct += $vector1[$i] * $vector2[$i];
            $magnitude1 += pow($vector1[$i],  exponent: 2);
            $magnitude2 += pow($vector2[$i],  exponent: 2);
        }
        $magnitude1 = sqrt($magnitude1);
        $magnitude2 = sqrt($magnitude2);
        return $dotProduct / ($magnitude1 * $magnitude2);
    }
```

Приклад 3 – Модель з бази знань Constraint-Based

ДОДАТОК Б

Відомість кваліфікаційної роботи

Позначення	Найменування	Дод. відомості
	Текстові документи	
1.	Пояснювальна записка	90 с.
	Інші документи	
2.	Презентаційні матеріали	20 плакатів

.	.	Прізвище та ініціали.	Підп.	Дата			
Розробив		Жукотинський Т.Г.			Дослідження та застосування знання-орієнтованого підходу до розробки рекомендаційних систем	Шифр групи	Код напр./спец.
Перевірив		Рябова Н.В.				СШІм-22-3	122
Н.контр.		Малєєва І.А.				ХНУРЕ кафедра ШІ	
Затв.		Філатов В.О.					