

*Употреба штампане литературе и комуникационих уређаја није дозвољена и строго је кажњива током трајања испита.*

ПРВИ ДЕО ИСПИТА (30 минута)

1. а) [3] Кратко (са по једном реченицом) дефинисати три различита пројектна узорка за податке у веб апликацијама.

б) [3] Дат је код који имплементира модел у апликацији вести из лабораторијске вежбе 3. Објаснити који пројектни узорак за податке је примењен у том коду и на основу чега то закључујете.

```
<?php
class Vest {
    private $id;
    private $naslov;
    private $sadrzaj;
    private $autor;
    private $datum;

    public function __construct($id, $naslov, $sadrzaj, $autor, $datum){
        $this->id = $id;
        $this->naslov = $naslov;
        $this->sadrzaj = $sadrzaj;
        $this->autor = $autor;
        $this->datum = $datum;
    }

    public function __get($imeAtributa) {
        return $this->$imeAtributa;
    }

    public static function dohvatiSve(){
        $konekcija = DB::getInstanca();
        $upit = "SELECT * FROM vesti";
        $rezultat = $konekcija->query($upit);
        $niz = [];
        foreach($rezultat->fetchAll() as $red){
            $niz[] = new Vest($red['id'], $red['naslov'], $red['sadrzaj'],
                $red['autor'], $red['datum']);
        }
        return $niz;
    }
}
```

```

public static function pronadji($ident){
    $konekcija = DB::getInstanca();
    $rezultat = $konekcija->prepare("SELECT * FROM vesti WHERE id=:nesto");
    $rezultat->execute(array('nesto' => $ident));
    $vest = $rezultat->fetch();
    return new Vest($vest['id'], $vest['naslov'],
        $vest['sadrzaj'], $vest['autor'], $vest['datum']);
    }
}

```

2. [4] Уколико преузимање података са веб форме није добро испрограмирано, могуће је извести *SQL Injection* напад који резултује брисање базе. Навести могући изглед *SQL* упита који би се при овом нападу користио, ако је упит у апликацији:

```

$query = "SELECT id, name, inserted, size FROM products
WHERE size = '$size'";

```

где се вредност  $\$size$  преузима директно са улазне форме, без провере.

3. а) [3] Навести формулу за рачунање *TCF* (фактора техничке комплексности) у оквиру *use case points* метода и навести формулу како се *TCF* користи за добијање коригованих *UCP*.

б) [3] Анализирати колико *TCF* може процентуално да измени *UCP* у оба гранична случаја, при константним факторима окружења. Користити табелу са тежинским коефицијентима појединих фактора.

Фактор	Опис	Тежина
T1	Дистрибуирани систем	2.0
T2	Захтева се време одзива/перформансе	1.0
T3	Захтева се ефикасан рад крајњег корисника	1.0
T4	Интерно комплексно процесирање	1.0
T5	Вишеструка употребљивост програмског кода	1.0
T6	Захтева се лака инсталација	0.5
T7	Лакоћа употребе	0.5
T8	Преносивост на друге платформе	2.0
T9	Одржавање система	1.0
T10	Конкурентно/паралелно процесирање	1.0
T11	Захтевају се сигурносне функције	1.0
T12	Пристап од стране трећих лица	1.0
T13	Обука крајњег корисника	1.0

*Употреба штампане литературе и комуникационих уређаја  
није дозвољена и строго је кажњива током трајања испита.*

Име и презиме:	Број индекса:
----------------	---------------

ДРУГИ ДЕО ИСПИТА (150 минута)

4. Посматра се систем осигуравајуће куће. У систему постоје три типа корисника: нерегистровани корисници, регистровани корисници (осигураници) и радници у осигуравајућој кући. За регистроване кориснике и раднике у осигуравајућој кући се чувају име, презиме, држава, адреса е-поште и лозинка. За раднике у осигуравајућој кући се поред тога чувају датум склапања радног односа и датум истицања уговора.

Сви корисници овог система могу да прегледају и претражују типове осигурања. Осигуравајућа кућа продаје два типа осигурања: путно осигурање и осигурање аутомобила. За осигурања постоје категорије осигурања, у зависности од тога шта све она покривају.

Путно осигурање се купује за одређени период и његова цена зависи од државе у коју се иде, категорије осигурања и броја дана за који се узима. Износ осигурања зависи од категорије осигурања и државе у коју се путује. Није дозвољено да корисник има два пута плаћен исти дан за исту државу, као и да плаћа путно осигурање за државу у којој живи. Неке државе не подржавају све типове осигурања.

Осигурање аутомобила се ради на нивоу календарске године. Полиса осигурања за аутомобиле не мора да се води на власника аутомобила, али не сме постојати више полиса осигурања за исту годину и аутомобил.

Регистровани корисници поред тога што могу да виде типове и износе осигурања, могу и да електронски пријаве које осигурање желе да купе, као да да виде сва њихова досадашња осигурања укључујући и колико су их платили.

Радник у осигуравајућој кући, поред тога што има све могућности као и осигураник, има могућност да издаје осигурања која су електронски пријављена и она која се купују на лицу места. У случају да осигураник жели да купи осигурање, а нема кориснички налог у систему, радник му може креирати нови налог, са ЈМБГ који не постоји у систему и у том случају поставља том осигуранику празну лозинку. У случају да осигураник има свој кориснички налог, тај налог се користи за куповину осигурања.

- a) [14] На нивоу релационе базе података, дате у *SQL skriptajun.sql*, утврдити да ли је база и да ли су подаци у бази у складу са горе описаном спецификацијом. Урадити све измене, како би база података одговарала овим корисничким захтевима. Модификовану базу података попунити подацима, неопходним за тестирање тачке ц).

Излаз задатка 4а): скрипт модификоване базе података *skriptajunGGGGBBBB.sql*

- b) [16] У алату *NetBeans IDE*, реализовати *MVC (Model-View-Controller)* апликацију коришћењем програмског језика *PHP*. Имплементацију је могуће радити коришћењем објектно-оријентисаног *PHP* кода или коришћењем *CodeIgniter* или *Laravel* оквира. Потребно је имплементирати искључиво део система који се односи на креирање Осигурања аутомобила од стране радника у осигуравајућем друштву и приказ свих важећих полиса осигурања за аутомобиле уз могућност претраге по ЈМБГ-у корисника. У овој тачки се подразумева да корисник већ постоји у систему (радник не треба да креира нови кориснички налог осигуранику).

Излаз задатка 4б): пројекат *PHP* имплементације описаног дела апликације (уколико се ради у радном оквиру, прекопирати такође читав пројекат)

- c) [14] Нацртати *WAE* дијаграм који приказује серверске компоненте (М,С) и клијентске компоненте (V) и *HTML* форме само дела веб апликације реализованог под тачком б).

Излаз задатка 4ц): нацртати реализован дијаграм

Име и презиме:	Број индекса:
----------------	---------------

Решење задатка 4ц: