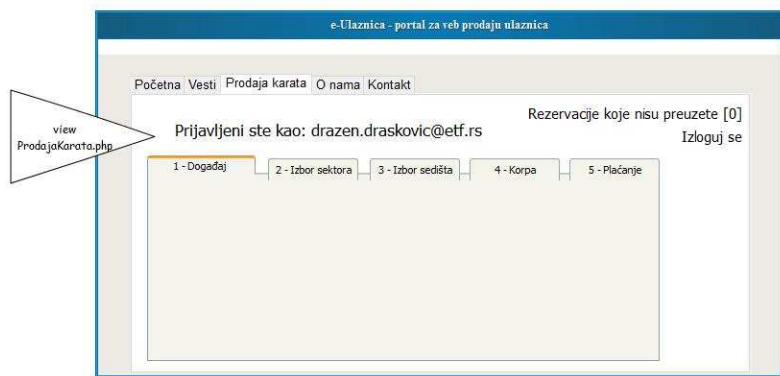


Испит траје 3 сата. Није дозвољена употреба литературе.

- [12] Мапирање класа из хијерархије наслеђивања у табеле релационе базе може да се изведе на један од три начина:
 - Све класе са дијаграма се мапирају у једну једину релациону табелу.
 - Свака конкретна класа добија сопствену релациону табелу. Апстрактне класе не добијају табеле него се подаци из њих чувају у оквиру одговарајућих конкретних класа које их наслеђују.
 - Свака класа (било апстрактна или конкретна) добија сопствену релациону табелу.

Нека су дате класе **Osoba** (апстрактна, са атрибутима: име, презиме), **Student** (конкретна, наслеђује класу *Osoba*, атрибут: индекс), **Nastavnik** (конкретна, наслеђује класу *Osoba*, атрибут: звање) и **Honorarac** (конкретна, наслеђује класу *Nastavnik*, атрибут: проценат ангажовања).

Представити табеле у бази (назив табеле, списак назива атрибута; уз атрибуте назначити да ли се ради о примарном и/или спољном кључу) за сваку од варијанти А, Б и В. Такође, за сваку од варијанти написати SQL упит којим се исписују сви подаци о сваком хонорарном наставнику.
- [6] Објаснити *SQL injection* тип напада на веб апликације и како се заштити од ове врсте напада.
- [20] На слици је дат приказ структуре веб странице портала "е-Улазница" за продају улазница путем Интернета. Дражен и пар асистената Катедре за РТИ желе да купе улазнице за фудбалску утакмицу Србија-Велс. На основу описа корисничких захтева датих у наставку од стране спортских савеза Србије, потребно је да:
 - [5] урадите прототип свих релевантних делова ове веб странице за продају карата методом *storyboard*;
 - [10] реализуете УМЛ дијаграм секвенце за Драженову успешну куповину улазница, уз приказ свих интеракција и порука;
 - [5] написати класе еквиваленције за тестирање корака 1 и 5 ове веб стране.



У кораку 1 корисник треба да има могућност уноса података које претражује (унос назива догађаја и/или унос места одржавања). Испод форме за претрагу треба да постоји табела са приказом резултата, који се добијају након притиска на дугме. Иницијално у табели приказати све резултате, при чему табела може прикати највише 20 резултата претраге. Списак догађаја треба да садржи датум и време одржавања догађаја.

Корак 2: Корисник на слици стадиона/дворане треба да одабере сектор који жели.

У кораку 3 корисник бира из падајуће листе број улазница, а највише 4. Уколико је погрешно, треба му омогућити да поново одабере сектор. Корисник има могућност да сам одабере места или да му систем аутоматски додели места. Ако сам одабере места, онда их бира избором одговарајућих седишта са слике тог сектора и потврђује притиском на дугме. Ако узме аутоматски избор, онда бива информисан која седишта и ред је добио и ручно убацује у корпу притиском на дугме.

Корак 4: Корисник добија приказ корпе и ако има неке улазнице, он може да испразни корпу или да обрише само одређену појединачну улазницу.

У кораку 5 корисник улази у процедуру плаћања улазница и на тој страни добија: основне информације о купцу/себи (које се читају из базе: име, презиме, адреса, место, телефон), чекбокс да мора да прихвати услове е-куповине и избор картице са основним информацијама (Мастер или Виза, број картице, датум истека и троцифрени заштитни код). Име на картици мора бити као име власника корисничког налога (из основних инфо.). Корисника апликација даље води на интерфејс банке за завршетак плаћања (то није део ове апликације) и тај интерфејс враћа нашем систему успешно или неуспешно обрађену трансакцију.

Напомена за тачку б):

Претпоставити да се све провере врше на серверској страни. За интерфејс банке користити класу *Bank*, а за интеракцију са базом користити класу *TicketsDB*.

На дијаграму јасно назначити улогу учесника (*Model, View, Controller*), путем стереотипа (<<model>>, <<view>>, <<controller>>). При позивима метода других учесника, јасно назначити који се аргументи прослеђују.

- [22] На Електријади учествују факултети из региона. Корисници веб система "Електријада 2017" су наставници (селектори екипа и ментори такмичења), студенти, који могу бити учесници (у знању и у спорту) или туристи, и администратор система, који није ни наставник ни студент. Корисници попуњавају веб форму чиме се региструју на систем. Након што се успешно региструју, корисници могу да се улогују на систем. Они у форми могу извршити резервацију смештаја и организованог превоза. Могу резервисати и само једно од та два или ниједно. За сваку резервацију чува се време резерације и ко је резервисао. Приликом резервације може се одабрати и тип собе (2-, 3- или 4-креветна).

Наставници и студенти могу припадати само једном факултету. Селектор је наставник који може да прегледа своје екипе и да креира екипу за одређену дисциплину у знању. Селектор не сме бити ментор било ког такмичења (администратор додељује менторе пре почетка Електријаде).

Свака екипа има свој назив и максимално 3 члана. Један факултет може имати највише 2 екипе у једној дисциплини у знању. Такмичење у знању има своје место одржавања, датум и време, и везује се за одређену дисциплину. Студенти на такмичењу учествују у појединачној конкуренцији и екипно, тако што представљају свој факултет. Поени које такав студент освоји, рачунају се и у бодовни салдо те екипе. Може бити студената који учествују на такмичењу само у појединачној конкуренцији. Студенти се сами пријављују за такмичења, а онда их селектори бирају приликом креирања тима. Студент се може пријавити за више тимова (осим ако нису из исте дисциплине) и такмичења (осим ако се временски не преклапају), али може бити одабран у највише 2 тима. Изласком на такмичење студент учествује на истом, остварује одређени број поена и одређени ранг. Када три студента истог факултета, чланова исте екипе, изађу на такмичење, збир њихових поена представља поене те екипе и та екипа се онда може рангирати на такмичењу. Последња екипа са најмање поена добија 1 бод, претпоследња 2, а прва екипа Н бодова, ако је учествовало Н екипа. Бонуси за прве три екипе су 5, 3 и 1 бод.

Од претходне Електријаде преузет је део базе података са следећим табелама:

FAKULTETI_UCESNICI (**IDFakulteta**, naziv, adresa, zemlja, brojpobeda)
SIFRARNIK_DISCIPLINA_ZNANJE (**IDNDisc**, naziv, IDNastavnikaMentora)

а) [4] Написати главни ток корака у ССУ (сценаријима случаја употребе) за функционалност *Пријављивање екипе у знању*, од стране студента. Разматрати успешан сценарио и могуће алтернативне токове. Предуслов је да студент има кориснички налог и да је улогован.

б) [6] Нацртати *IE* модел базе података, на основу описане спецификације система. У свакој табели приказати назив колоне и тип података за ту колону, означити примарне и стране кључеве и све релације између табела.

в) [6] Нацртати WAE дијаграм који приказује серверске странице, клијентске странице и HTML форме ове веб апликације. Део апликације за администратора није детаљно описан у корисничкој спецификацији, али га треба реализовати, сходно опису.

г) [6] Написати *PHP* скриптове који реализују описани случај употребе *Пријављивање екипе у знању* и интеракцију између студента који се пријављује за такмичење и селектора који одабране студенте бира у тим. Написати све клијентске и серверске веб странице, класе и методе које бисте реализовали. Параметре за базу података чувати као константне вредности, у посебном фајлу који се учитава у ове скриптове. Дозвољено је и коришћење фрејмворка *CodeIgniter/Laravel*, односно компонената *Model, View* и *Controller* пројектног узорка MVC.

Пример рангирања у знању (за једну дисциплину):

Такмичење у дисциплини у знању: ИНФОРМАТИКА			
ЕТФ Београд 1	ФЕР Загреб 1	ЕТФ БЛ 1	ЕТФ Београд 2
М.Брзаковић 65	С.Студени 68	П.Петровић 65	Н.Бебић 70
К.Ђорђевић 64	Л.Листопад 72	Ј.Јовановић 60	А.Милаковић 75
Н.Црноглавац 63	П.Просинац 50	М.Марковић 50	И.Гавриловић 70
Укупно поена: 192 Место: 2	Укупно поена: 190 Место: 3	Укупно поена: 175 Место: 4	Укупно поена: 215 Место: 1
Бодова: 3 + 3 = 6	Бодова: 2 + 1 = 3	Бодова: 1	Бодова: 4 + 5 = 9

Напомена: Победник Електријаде је екипа која сакупи највећи број бодова у свим дисциплинама.

Потписи главних функција програмског језика PHP:

```
resource mysqli_connect (string $server, string $username, string $password, string $db_name);  
mixed mysqli_query ( mysqli $link , string $query )  
int mysqli_num_rows ( mysqli_result $result )  
mixed mysqli_fetch_array ( mysqli_result $result )  
array mysqli_fetch_assoc ( mysqli_result $result )
```