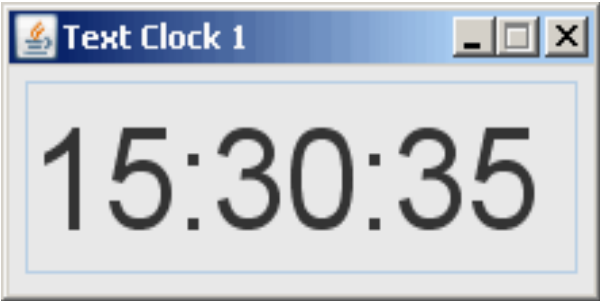
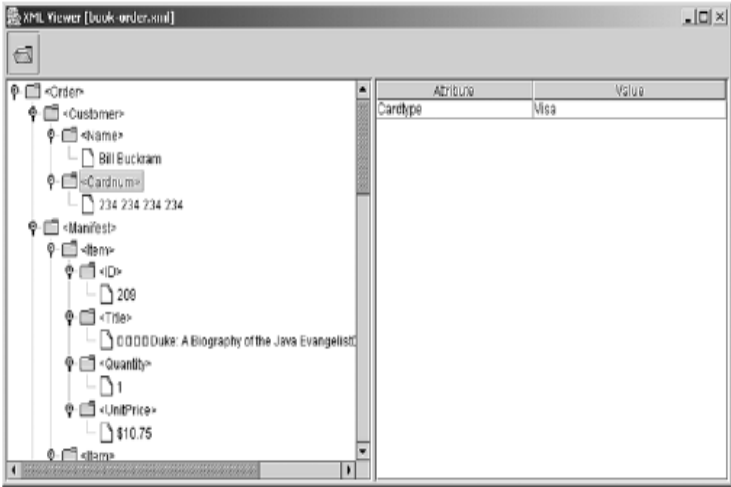


Uvod u softversko inženjerstvo: 03.10.2016.g.

Zadatak 1. (20)	<p>Neka je data delimična definicija Python klase Time kao:</p> <pre>class Time: def __init__(self, hour, minute): self._hour = hour self._minute = minute def advance(self): self._minute += 1 if self._minute > 60: self._minute = 0 self._hour += 1 if self._hour > 12: self._hour = 1 def getHour(self): return self._hour def getMinute(self): return self._minute</pre> <p>a. (05) Napisati str metodu specificirane klase koja nema parametre i vraća string predstavu vremena (time).</p> <p>b. (15) Metoda advance nije korektna. Specificirajte Gerkin predstavu njene dinamike i na osnovu te specifikacije definišite jedinični test koji otkriva unetu grešku.</p>
Zadatak 2. (35)	<p>Na slici je dat izgled korisničkog interfejsa koji je neophodno implementirati uz oslonac na PYTHON programski jezik, QT biblioteku i MVC arhitektonski šablon.</p>  <p>Proširiti osnovni model i programskog rešenje mogućnošću izbora načina vizualizacije vremena (analogni/digitalni časovnik).</p> <p>Proširiti osnovni model i aplikaciju datumskom komponentom tako da konačno rešenje obezbeđuje mogućnost izbora: samo datuma, samo vremena i kombinaciju datuma i vremena.</p>

<p>Zadatatak 3.</p> <p>(20)</p>	<p>Neophodno je specificirati AlarmClock Python klasu koja predstavlja tipičan budilnik.</p> <ol style="list-style-type: none"> Navedite najmanje tri atributa koje je moguće koristiti za predstavljanje stanja budilnika. Detaljno obrazložite svaki od definisanih atributa. Opišite najmanje tri ponašanja koja je neophodno podržati od strane budilnika. Koristiti Gerkin sintaksu. Konstruišite specificiranu klasu sa potrebnim atributima i metodama uz oslonac na Python programski jezik. Za svako od definisanih ponašanja (b) napisati metodu sa parametrima koja implementira specificirano ponašanje i kao parametra vraća specificirani rezultat.
<p>Zadatak 4.</p> <p>(25)</p>	<p>Na slici je dat izgled kompozitnog korisničkog interfejsa koji kombinuje mogućnost prikaza strukture proizvoljnog stabla (<i>levi deo</i>) i tabelarnog prikaza sadržaja fokusnog elementa (<i>desni deo</i>).</p>  <ol style="list-style-type: none"> (10) Definirati QT komponente koje je potrebno koristiti za implementaciju prikazanog GUI. (15) Napisati PYTHON kod koji omogućava implementaciju prikazanog GUI, uz pretpostavku da se struktura stabla i sadržaj elementa čitaju iz tekstualnih datoteka.

Ispit traje 3 (tri) sata

Rezultati do 04.10.2016.g.