

Ekonomski fakultet u Subotici

Univerzitet u Novom Sadu

Projekat iz predmeta Objektni softverski inženjering

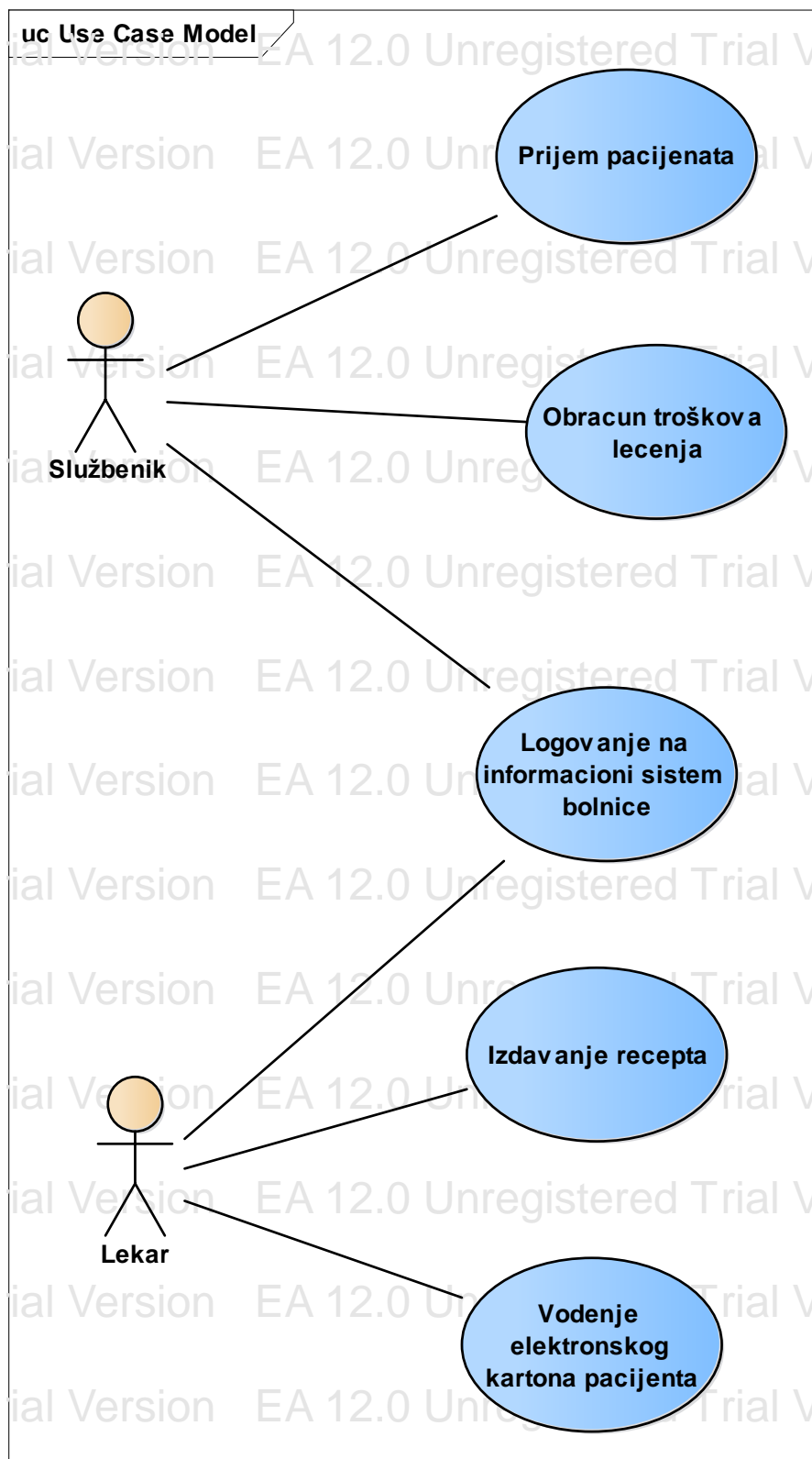
Student: Gaborov Sava

G013\11

SADRŽAJ

1. DIJAGRAM SLUCAJA UPOTREBE	3
2. OPIS SLUČAJA PRIJEM PACIJENATA	4
2.2 KORISNIČKI INTERFEJS ZA SLUČAJ UPOTREBE PRIJEM PACIJENATA	4
2.2.1 Obaveštenje službenika o slobodnim mestima	4
2.2.2 Unos podataka o pacijentu	5
2.3 DIJAGRAM SEKVENCE ZA SLUČAJ UPOTREBE PRIJEM PACIJENATA	6
2.4 CRC KARTICE ZA SLUCAJ UPOTREBE PRIJEM PACIJENATA	7
2.5 DIJAGRAM KLASA ZA SLUČAJ UPOTREBE PRIJEM PACIJENATA	8
2.6 PROGRAMSKI KOD ZA SLUČAJ UPOTREBE PRIJEM PACIJENATA	9
3 OPIS SLUČAJA OBRAČUN TROŠKOVA LEČENJA	16
3.1 KORISNIČKI INTERFEJS ZA SLUČAJ UPOTREBE TROŠKOVA LEČENJA	16
3.1.1 Unos podataka o pacijentu	16
3.1.2 Štampanje i izdavanje fakture pacijentu	17
3.2 DIJAGRAM SEKVENCE ZA SLUČAJ UPOTREBE OBRAČUN TROŠKOVA LEČENJA	18
3.3 CRC KARTICE ZA SLUCAJ UPOTREBE OBRAČUN TROŠKOVA LEČENJA	19
3.4 DIJAGRAM KLASA ZA SLUČAJ UPOTREBE OBRAČUN TROŠKOVA LEČENJA	20
.....	20
3.5 PROGRAMSKI KOD ZA SLUČAJ UPOTREBE OBRAČUN TROŠKOVA LEČENJA	21

1. DIJAGRAM SLUCAJA UPOTREBE

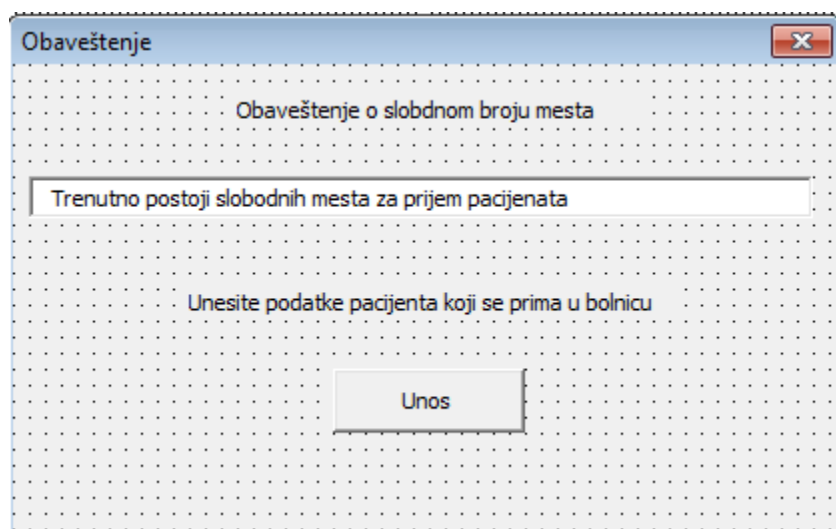
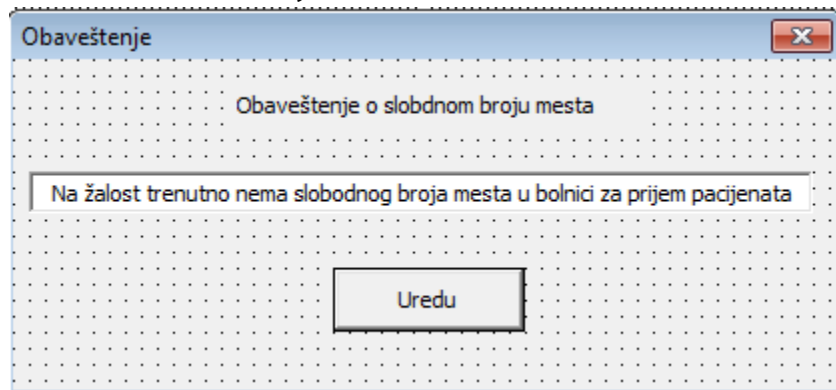


2. OPIS SLUČAJA PRIJEM PACIJENATA

Prilikom pokretanja ovog slučaja upotrebe službeniku se prosleđuje forma koja ga obaveštava o mogućnosti slobodnog mesta kako bi se pridolio pacijent smestio. Ukoliko ne postoji slobodan broj mesta službeniku se prosleđuje nova forma koja ga obaveštava da trenutno nema slobodnih mesta za prijem pacijenata. Ukoliko postoji slobodan broj mesta u bolnici službeniku se prosleđuje forma za unos podataka o pacijentu koji se prima u bolnicu. Na ovoj formi je navedeno (Ime i prezime pacijenta, jmbg, datum rođenja, datum prijema, pol, oboljenje, terapija) . Službenik mora da odabere željeni datum prijema pacijenta ovo je bitno zbog obračuna troškova pacijenta kada bude napuštao bolnicu. Kada ga kada potvri unos pacijenta, treba zvesti njegove podatke kao i povećati broj trenutno zalezih mesta u bolnici za 1.

2.2 Korisnički interfejs za slučaj upotrebe prijem pacijenata

2.2.1 Obaveštenje službenika o slobodnim mestima



2.2.2 Unos podataka o pacijentu

Unos podataka o pacijentu

Šifra pacijenta	<input type="text"/>	Pol	<input type="text"/>
Jmbg	<input type="text"/>	Oboljenje	<input type="text"/>
Ime	<input type="text"/>	Datum prijema	<input type="text"/>
Prezime	<input type="text"/>	Terapija	<input type="text"/>
Datum rođenja	<input type="text"/>		

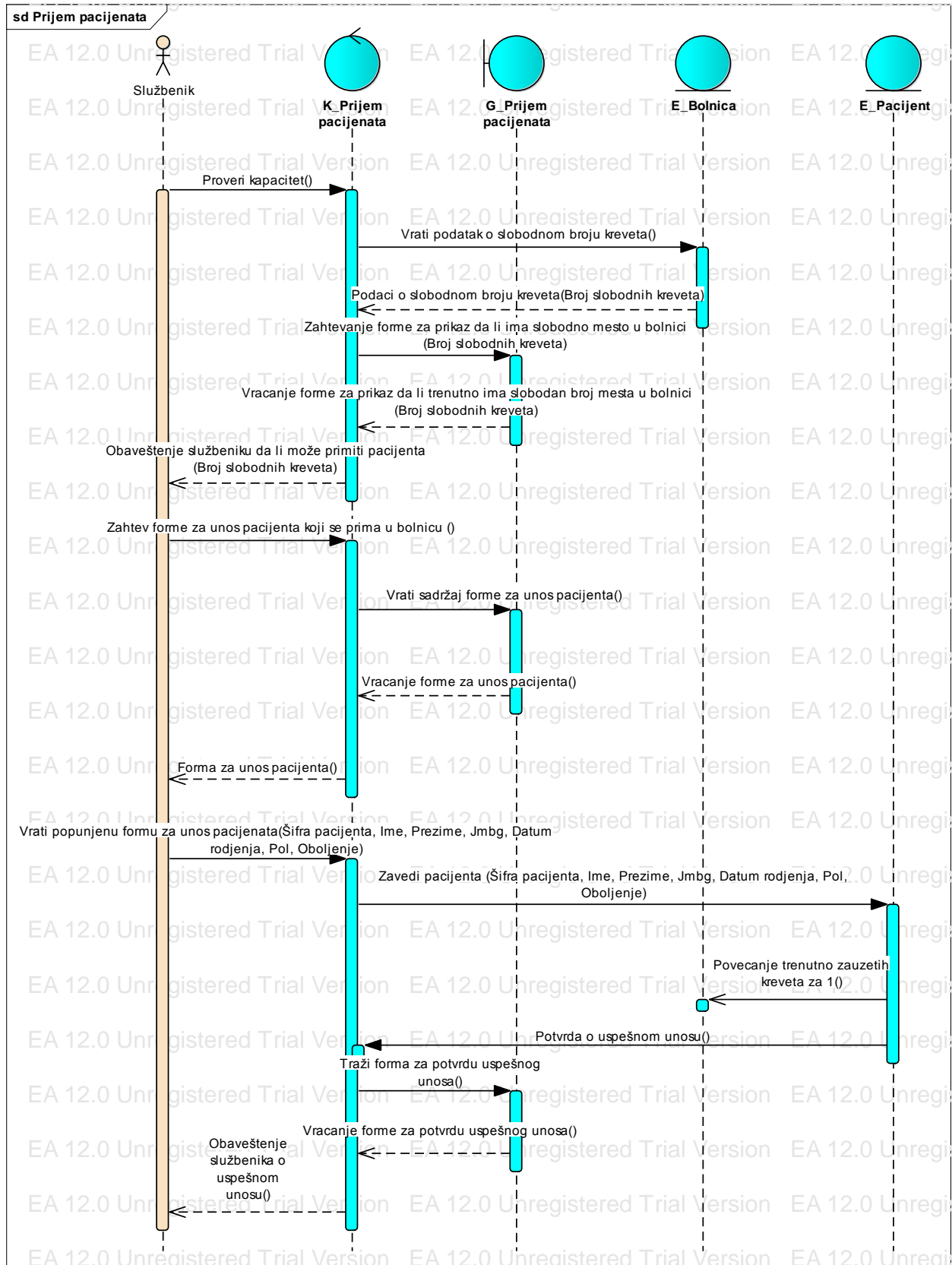
Unesi podatke

Uspešan unos

Uspešno ste uneli pacijenta

Uredi

2.3 Dijagram sekvence za slučaj upotrebe prijem pacijenata



2.4 CRC kartice za slučaj upotrebe prijem pacijenata

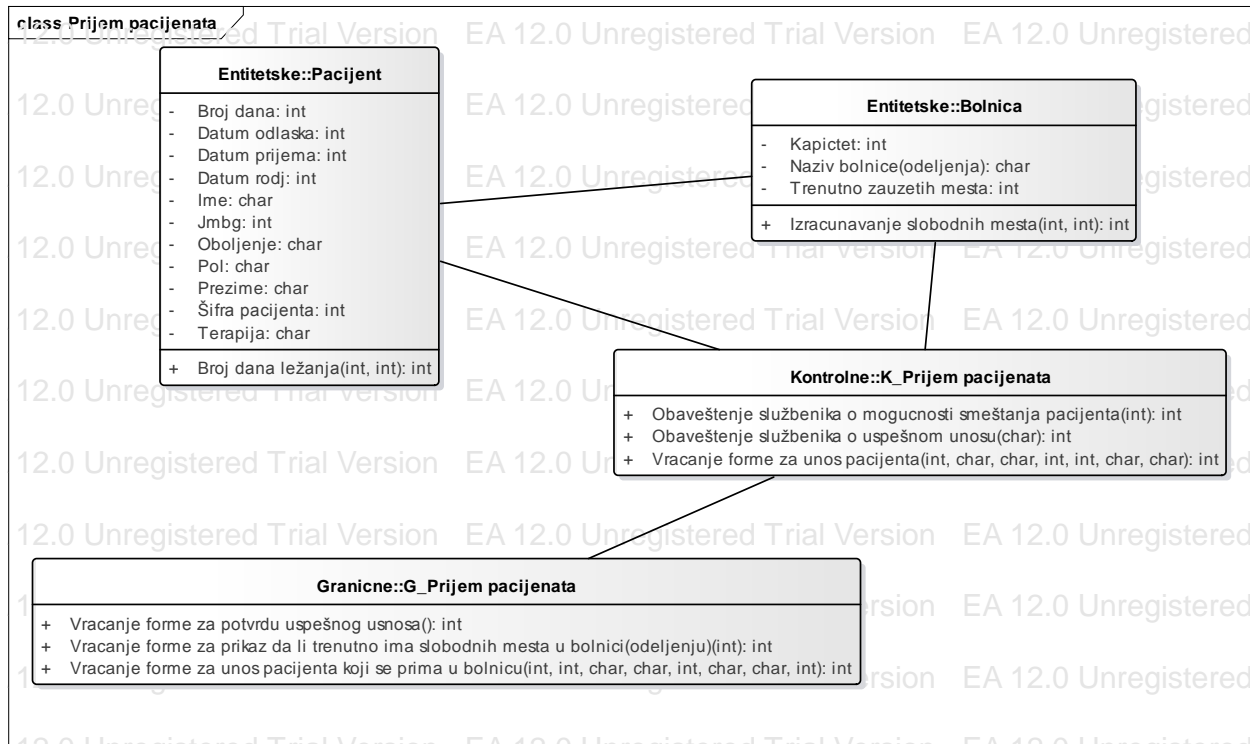
K_Prijem pacijenata	
Odgovornosti	Saradnja
Obaveštavanje službenika o mogućnosti smeštanja pacijenta	G_Prijem pacijenata
Obaveštavanje službenika o uspešnom unosu	E_Bolnica
Vraćanje forme za unos pacijenata	E_Pacijent

G_Prijem pacijenata	
Odgovornosti	Saradnja
Vraćanje forme za potvrdu uspešnog uosa	K_Prijem pacijenata
Vraćanje forme da li trenutno ima slobodnih mesta	
Vraćanje forme za unos pacijenata koji se prima	

E_Pacijent	
Odgovornosti	Saradnja
Unos podataka o pacijentu	G_Prijem pacijenata

E_Bolnica	
Odgovornosti	Saradnja
Izračunavanje slobodnog broja mesta	K_Prijem pacijenata

2.5 Dijagram klasa za slučaj upotrebe prijem pacijenata



2.6 Programski kod za slučaj upotrebe prijem pacijenata

Generisani kod

```
public class Bolnica {

    private int Kapictet;

    private char Naziv bolnice(odeljenja);

    private int Trenutno zauzetih mesta;

    public Bolnica(){

    }

    public void finalize() throws Throwable {

    }

    /**
     *
     * @param Kapacitet
     * @param Trenutno zauzetih mesta
     */
    public int Izracunavanje slobodnih mesta(int Kapacitet, int Trenutno zauzetih mesta){

        return 0;

    }

}

} //end Bolnica

package Entitetske;
```

```
/**  
 * @author Win7  
 * @version 1.0  
 * @created 22-Jun-2015 10:28:59 PM  
 */
```

```
public class Pacijent {  
  
    private int Broj dana;  
  
    private int Datum odlaska;  
  
    private int Datum prijema;  
  
    private int Datum rođj;  
  
    private char Ime;  
  
    private int Jmbg;  
  
    private char Oboljenje;  
  
    private char Pol;  
  
    private char Prezime;  
  
    private int Šifra pacijenta;  
  
    private char Terapija;  
  
    public Troškovi m_Troškovi;  
  
    public Bolnica m_Bolnica;  
  
  
    public Pacijent(){  
  
  
    }  
}
```

```

        public void finalize() throws Throwable {

        }

        /**
         *
         * @param Datum prijema
         * @param Datum odlaska
         */
        public int Broj dana ležanja(int Datum prijema, int Datum odlaska){
            return 0;
        }
    } //end Pacijent

```

```

    /**
     * @author Win7
     * @version 1.0
     * @created 22-Jun-2015 10:36:06 PM
     */
    public class G_Prijem pacijenata {

        public G_Prijem pacijenata(){

        }

    }

```

```

public void finalize() throws Throwable {

}

public int Vracanje forme za potvrdu uspešnog usnosa(){

    return 0;

}

/**
 *
 * @param Slobodna_mesta
 */

public int Vracanje forme za prikaz da li trenutno ima slobodnih mesta u bolnici(oddeljenju)(int
Slobodna_mesta){

    return 0;

}

/**
 *
 * @param Šifra pacijenta
 * @param Jmbg
 * @param Ime
 * @param Prezime
 * @param Datum rodjenja
 * @param Pol
 * @param Oboljenje

```

```

        * @param Datum prijema
        */

        public int Vracanje forme za unos pacijenta koji se prima u bolnicu(int Šifra pacijenta, int Jmbg,
char Ime, char Prezime, int Datum rođenja, char Pol, char Oboljenje, int Datum prijema){

            return 0;

        }

    }//end G_Prijem pacijenata

import Entitetske.Bolnica;

import Entitetske.Pacijent;

/**
 * @author Win7
 * @version 1.0
 * @created 22-Jun-2015 10:38:42 PM
 */
public class K_Prijem pacijenata {

    public G_Prijem pacijenata m_G_Prijem pacijenata;

    public Bolnica m_Bolnica;

    public Pacijent m_Pacijent;

    public K_Prijem pacijenata(){

    }

    public void finalize() throws Throwable {

```

```

}

/**
 *
 * @param Slobodan broj mesta
 */
public int Obaveštenje službenika o mogućnosti smeštanja pacijenta(int Slobodan broj mesta){
    return 0;
}

/**
 *
 * @param Poruka
 */
public int Obaveštenje službenika o uspešnom unosu(char Poruka){
    return 0;
}

/**
 *
 * @param Šifra pacijenta
 * @param Ime
 * @param Prezime
 * @param Jmbg
 * @param Datum rođenja

```

```
* @param Pol
* @param Oboljenje
*/

public int Vracanje forme za unos pacijenta(int Šifra pacijenta, char Ime, char Prezime, int Jmbg,
int Datum rodjenja, char Pol, char Oboljenje){

    return 0;

}

} //end K_Prijem pacijenata
```

3 OPIS SLUČAJA OBRAČUN TROŠKOVA LEČENJA

Ovaj slučaj upotrebe se odnosi na obračun troškova lečenja pacijenta za dane boravka koje je proveo u bolnici. Službeniku se prikazuje prvo forma za unos osnovnih podataka pacijenta (Ime, Prezime, jmbg, Datum odlaska) da bi se znalo za kog pacijenta se vrši obračun troškova odnosno da se utvrdi terapija koju je pacijent primao tokom boravka. Službenik takođe unosi datum odlaska da bi se utvrdio broj dana boravka pacijenta u bolnici. Zatim se pritiskom na dugme obračunaj službeniku se prikazuje nova forma u kojoj su sadržani osnovni podaci o pacijentu kao i iznos koji je pacijent dužan da plati za usluge lečenja. Pritiskom na dugme odštampaj fakturu, faktura se izdaje i štampa pacijentu.

3.1 Korisnički interfejs za slučaj upotrebe troškova lečenja

3.1.1 Unos podataka o pacijentu

Unos podataka pacijenta za kog se obracunavaju troškovi

Ime

Prezime

Datum odlaska

Jmbg

Obračunaj troškove za datog pacijenta

Obračunaj

3.1.2 Štampanje i izdavanje fakture pacijentu

Štampanje i izdavanje fakture pacijentu

IME:

PETAR

PREZIME:

PETROVIĆ

JMBG:

1987228770005

IZNOS:

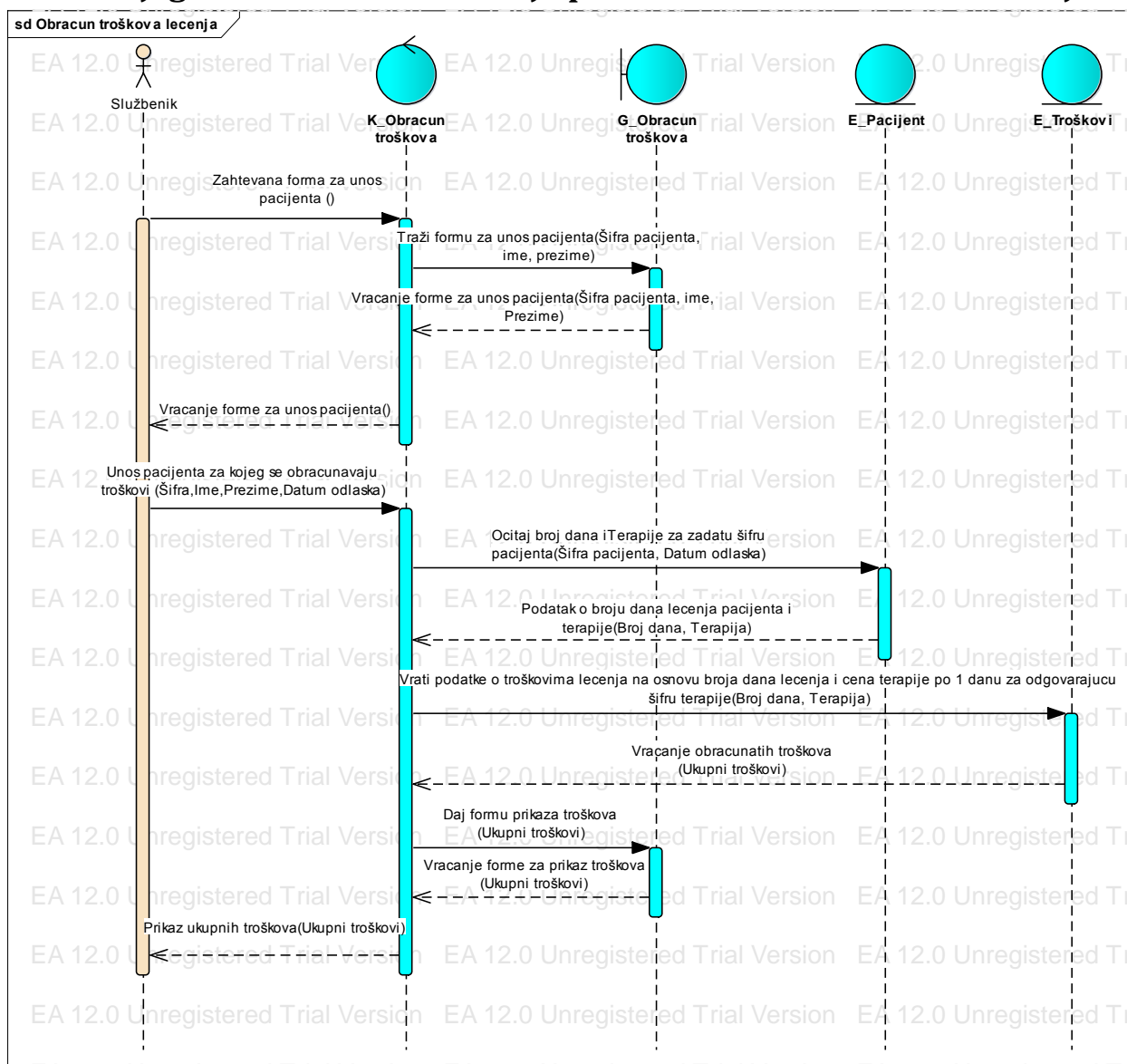
3543,70.00

DATUM IZDAVANJE FAKTURE:

22.6.2015

Odštapaj fakturu

3.2 Dijagram sekvence za slučaj upotrebe obračun troškova lečenja



3.3 CRC kartice za slučaj upotrebe obračun troškova lečenja

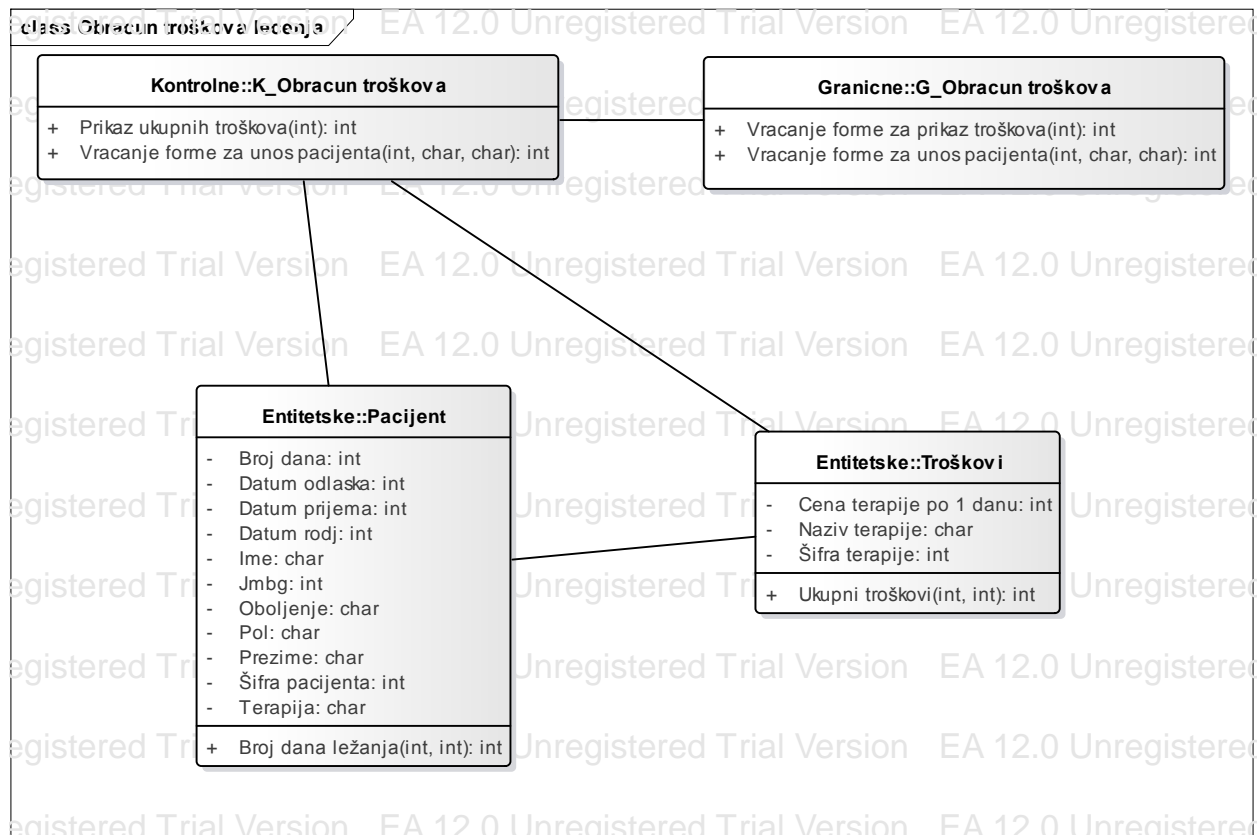
K_Obračun troškova	
Odgovornosti	Saradnja
Prikaz ukupnih troškova	G_ obračun troškova
Unos pacijenta	E_ Pacijent
Vraćanje forme za prikaz ukupnih troškova	E_ troškovi

G_ Obračun troškova	
Odgovornosti	Saradnja
Vraćanje forme za prikaz ukupnih troškova	K_ Obračun troškova
Vraćanje forme za unos pacijenta	

E_ Pacijent	
Odgovornosti	Saradnja
Broj dana ležanja	K_ Obračun troškova

E_ Troškovi	
Odgovornosti	Saradnja
Ukupni troškovi	K_ Obračun troškova

3.4 Dijagram klasa za slučaj upotrebe obračun troškova lečenja



3.5 Programski kod za slučaj upotrebe obračun troškova lečenja

package Entitetske;

/**

* @author Win7

* @version 1.0

* @created 22-Jun-2015 10:28:59 PM

*/

public class Pacijent {

private int Broj dana;

private int Datum odlaska;

private int Datum prijema;

private int Datum rodj;

private char Ime;

private int Jmbg;

private char Oboljenje;

private char Pol;

private char Prezime;

private int Šifra pacijenta;

private char Terapija;

public Troškovi m_Troškovi;

public Bolnica m_Bolnica;

public Pacijent(){

```
}
```

```
public void finalize() throws Throwable {
```

```
}
```

```
/**
```

```
*
```

```
* @param Datum prijema
```

```
* @param Datum odlaska
```

```
*/
```

```
public int Broj dana ležanja(int Datum prijema, int Datum odlaska){
```

```
    return 0;
```

```
}
```

```
}//end Pacijent
```

```
package Entitetske;
```

```
/**
```

```
* @author Win7
```

```
* @version 1.0
```

```
* @created 22-Jun-2015 10:28:59 PM
```

```
*/
```

```
public class Troškovi {
```

```
    private int Cena terapije po 1 danu;
```

```
    private char Naziv terapije;
```

```

private int Šifra terapije;

public Troškovi(){

}

public void finalize() throws Throwable {

}

/**
 *
 * @param Cena terapije po 1 danu
 * @param Broj dana lecenja
 */
public int Ukupni troškovi(int Cena terapije po 1 danu, int Broj dana lecenja){
    return 0;
}

} //end Troškovi

/**
 * @author Win7
 * @version 1.0
 * @created 22-Jun-2015 10:36:03 PM
 */
public class G_Obracun troškova {

```

```

public G_Obracun troškova(){

}

public void finalize() throws Throwable {

}

/**
 *
 * @param Ukupni troškovi
 */
public int Vracanje forme za prikaz troškova(int Ukupni troškovi){
    return 0;
}

/**
 *
 * @param Šifra pacijenta
 * @param Ime
 * @param Prezime
 */
public int Vracanje forme za unos pacijenta(int Šifra pacijenta, char Ime, char Prezime){
    return 0;
}
} //end G_Obracun troškova

```



```
import Entitetske.Pacijent;

import Entitetske.Troškovi;

/**
 * @author Win7
 * @version 1.0
 * @created 22-Jun-2015 10:38:39 PM
 */
public class K_Obracun troškova {

    public G_Obracun troškova m_G_Obracun troškova;

    public Pacijent m_Pacijent;

    public Troškovi m_Troškovi;

    public K_Obracun troškova(){

    }

    public void finalize() throws Throwable {

    }

    /**
     *
     * @param Ukupni troškovi
     */
    public int Prikaz ukupnih troškova(int Ukupni troškovi){
```

```
        return 0;
    }

    /**
     * @param Šifra
     * @param Ime
     * @param Prezime
     */
    public int Vracanje forme za unos pacijenta(int Šifra, char Ime, char Prezime){
        return 0;
    }
} //end K_Obracun troškova
```