**УНИВЕРЗИТЕТ У КРАГУЈЕВЦУ**

**ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА У ЧАЧКУ**

Logo

Description automatically generated

ПРОЈЕКТНИ РАД

из предмета

**Савремене софтверске архитектуре**

**Тема: Софтверске архитектуре за праћење попуњености канти за смеће**

**Професор: Студенти:**

Проф. др Владимир Младеновић Данијела Станковић 80/2018

Урош Трифуновић 49/2018

**Aсистенти:**

мр Младен Јањић

др Фарук Селимовић

Чачак, 2022.

Садржај

[1 Увод 3](#_Toc104927062)

[1.1 Опис система 3](#_Toc104927063)

[2 Спецификације носилаца и интереса 3](#_Toc104927064)

[2.1 Спецификација носилаца 4](#_Toc104927065)

[2.2 Спецификацијa интереса 5](#_Toc104927066)

[2.3 Асоцијативна матрица носилаца и интереса 6](#_Toc104927067)

[3 Спецификације софтверских захтева 6](#_Toc104927068)

[3.1 Функционални захтеви 7](#_Toc104927069)

[3.2 Нефункционални захтеви 7](#_Toc104927070)

[3.3 Процесни захтеви 8](#_Toc104927071)

[4 УМЛ дијаграми модела 4+1 9](#_Toc104927072)

[4.1 Логичко гледиште 9](#_Toc104927073)

[4.1.1 Дијаграми класа 9](#_Toc104927074)

[4.2 Процесно гледиште 10](#_Toc104927075)

[4.2.1 Дијаграм активности 11](#_Toc104927076)

[4.3 Развојно гледиште 12](#_Toc104927077)

[4.3.1 Дијаграм компонената 13](#_Toc104927078)

[4.4 Физичко гледиште 13](#_Toc104927079)

[4.5 Случајеви коришћења 14](#_Toc104927080)

[4.5.1 Дијаграм случајева коришћења 14](#_Toc104927081)

[5 Презентациони слој 15](#_Toc104927082)

[5.1 Опис апликације 15](#_Toc104927083)

[6 Закључак 16](#_Toc104927084)

# Увод

У датом тексту, приказан је начин развоја и пројектовања информационог система за праћење попуњености канти. Овај систем омогућава организацијама да користе стратегију за одлагање, смањење, поновну употребу и контролисање отпада.

Отпад може бити чврст, течан или гасовит и свака врста има различите методе прикупљања и одлагања. Циљ управљања отпадом је смањење опасних утицаја таквог отпада на животну средину и здравље људи.

Редовни систем управљања отпадом укључује камионе за сакупљање отпада и њихове возаче који прате унапред дефинисану руту без испитивања нивоа попуњености канти. Пошто овај систем не може да мери нивое попуњености канти, полупуне канте могу да се испразне, а за разлику од њих, претходно напуњене канте морају да сачекају до следећег периода прикупљања. Возачи понекад сакупљају празне канте, па сходно томе унапред дефинисане руте прикупљања система изазивају губљење времена, повећање потрошње горива и прекомерно коришћење ресурса. Праћење попуњености канти за смеће у великој мери олакшава комуналним предузећима брзо и ефикасно уклањање отпада.

Правилно и контролисано управљање отпадом је битно за изградњу одрживих градова и очување животне средине људи.

## Опис система

Систем за праћење попуњености канти за смеће комбинује хардверске и софтверске елементе, где податке о попуњености канти из окружења добија путем сензора за мерење попуњености. Подаци добијени овим сензорима, шаљу се у базу података и путем одговарајуће софтверске подршке на интернету пружају информације о нивоу попуњености канти у било ком тренутку.

Како је систем првенствено намењен и прилагођен комуналним предузећима, омогућен им је приступ уз могућности директног приказа канти чији је проценат попуњености већи од 75%. Након организације пражњења попуњених канти, радници комуналних служби у могућности су да у софтверском систему измене проценат попуњености, односно ресетују вредности на подразумеване, како би хардверски део могао да настави да обавља функцију за коју је и намењен.

# Спецификације носилаца и интереса

Носилац је особа или организација која има права, акције, потраживања или интересе у вези са системом или његовим својствима која задовољавају њихове потребе и очекивања.

Често се на питање за кога се пројекат обично креира, добија одговор за крајње кориснике. Крајњи корисници су такође заинтересоване стране у пројекту, али можда нису најважнији. Стога, приликом развоја софтвера, потребно је формирати такав систем који ће јасно идентификовати ко су носиоци и какви типови носилаца су у систему прихватљиви, какви интереси су од значаја и које су везе између интереса и носилаца.

## Спецификација носилаца

Међу носиоцима система свакако би требало размотрити ко су и какве интересе имају:

* корисници система,
* власници система,
* девелопери система,
* одржаваоци система.

Као носиоци се могу идентификовати и друга заинтересована лица, као што су медицинске установе, транспортери, организације за заштиту животне средине, предузећа која својом производњом стварају велике количине отпада и слично.

Појединац, тим, организација или класа неког од њих, са неким интересом у односу на

предметни систем представља носиоца.

У носиоце система спадају они субјекти:

* који су одговорни за пројектовање и развој система,
* који имају финансијски интерес, који су одговорни за продају или набавку,
* који су одговорни за увођење и одржавање,
* који оперишу системом,
* који имају било какву корист од система (функционалну, политичку, финансијску, друштвену, материјалну или духовну),
* који регулишу разне аспекте система (финансијске, безбедносне и друге),
* на чије функционисање и живот система утиче у било ком смислу, позитивном или негативном

У табели 1 приказане су спецификације носилаца који су у овој архитектури приказани

кроз власника, девелопера, корисника, одржавање система.

Табела . Спецификација носилаца

|  |  |  |
| --- | --- | --- |
| Назив | Опис | Интереси |
| Власник | Особа или предузеће у чијем власништву се налазе канте и које је одговорно за њихово одржавање и организовање | Квалитет услуга, ефикасност пословања |
| Девелопер | Особа која врши пројектовање и развој система и његових функциионалности, пратећи при том неопходне спецификације. | Зарада |
| Корисник | Особа која има приступ и користи апликацију како би имала увид у стање попуњености канте и извршавала њихово пражњење. | Квалитет услуга, функционалност система |
| Одржаваоци система | Особа која обавља редовно ажурирање и одржавање система ради обезбеђивања функционалног окружења. | Одржавање система |

## Спецификацијa интереса

Интерес представља заинтересованост или одговорност једног или више носилаца за предметни систем.

Могу се посматрати и интереси као што су обезбеђење квалитета, понашање, трошкови, приступ подацима, интегритет података, дистрибуција, флексибилност, отвореност, приватност, безбедност, структура, интеграција подсистема и други.

Када је реч о интересима у односу на предметни систем, треба идентификовати, описати

и алоцирати са носиоцима следеће:

* циљеви система или његова мисија,
* погодност архитектуре система за постизање циљева,
* изводљивост изградње и размештаја система,
* потенцијални ризици и ефекти система,
* могућност одржавања система.

Табела . Спецификација интереса

|  |  |  |
| --- | --- | --- |
| Назив | Опис | Интереси |
| Квалитет услуга | Квалитетна услуга подразумева испуњење главних задатака власника, али и корисника | Власник, корисник |
| Функционалност система | Апликација је једноставна и лака за коришћење од стране корисника | Корисник |
| Одржавање система | Како би апликација остала функционална и унапређивала своје функционалности кроз захтеве корисника и власника, потребно је њено одржавање | Одржаваоци система |
| Зарада | Један од битнијих фактора којем тежи девелопер – продаја свог система. | Девелопер |
| Ефикасност пословања | Обезбеђивање извршавања свих потребних циљева и постављених задатака, циљ је власника. | Власник |

## Асоцијативна матрица носилаца и интереса

Кроз асоцијативну матрицу носилаца и интереса приказане су везе између носилаца и интереса (табела 3).

Табела 3. Асоцијативна матрица носилаца и интереса

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Носиоци  Интереси | Власник | Девелопер | Корисник | Одржаваоци система |
| Квалитет услуга | P |  | О |  |
| Функционалност система |  | Р | П |  |
| Одржавање система |  | P | П | О |
| Зарада | K | K | K | K |
| Ефикасност пословања | P |  | О |  |

К – носилац има корист од остварења асоцираног интереса (материјалну, политичку, професионалну итд)

О – носилац је одговоран за остваривање асоцираног интереса

Р – носилац регулише, прописује, уређује како се остварује асоцирани интерес

П – носилац је погођен остваривањем асоцираног интереса (у позитивном или негативном смислу)

# Спецификације софтверских захтева

Захтеви представљају својства и услове које систем мора да задовољи. Захтеви се описују помоћу УМЛ модела случајева коришћења. Такав модел састоји се од скупа случајева коришћења, актера (спољних корисника система) и веза између случајева коришћења и актера.

Случај коришћења описује скуп сценарија, тј. скуп жељених коришћења система од стране

актера. Стога можемо рећи да је сценарио описан преко:

* секвенце акције,
* интеракција актер-систем.

Случај коришћења састоји се од главног и алтернативних сценарија. Спецификација корисничких захтева подразумева опис потреба или жеља које софтверски производ треба да омогући. Захтев корисника представља опис потреба или жеља за неким производом. Неопходно је идентификовати и документовати потребе ради лакше комуникације клијента са тимом за развој софтверског производа.

Врсте софтверских захтева:

* Функционални
* Нефункционални
* Процесни

## Функционални захтеви

Функционални захтеви софтверског система идентификују се проучавањем пословне и/или техничке документације реалног система, поступцима као што су разговор са корисницима, наручиоцима итд. Циљ је сагледавање функција које реални систем обавља и да се направи опис функција које треба да обавља софтверски систем за потребе реалног система.

Табела 4. Функционални захтеви за систем праћења попуњености канти

|  |  |  |  |
| --- | --- | --- | --- |
| Редни број | Захтев | Сегмент | Приоритет |
| ФЗ.1 | Систем ће омогућити преглед канти заузетих више од 75% | Преглед | О |
| ФЗ.2 | Систем ће омогућити ажурирање испражњених канти | Ажурирање | О |
| ФЗ.3 | Систем ће омогућити приказ процента попуњености | Приказ | В |

* обавезни (О)
* важни (В)
* могући (М)
* будући (Б)

## Нефункционални захтеви

Нефункционални захтеви софтверског система су идентификовани као ограничења или захтеване вредности атрибута квалитета софтвера (скалабилност, перформансе, доступност, безбедност, поузданост, погодност за употребу, итд.) и системске карактеристике које произилазе из захтева корисника, прописа или стандарда (преносивост, модуларност, интеграбилност, погодност за одржавање и др.) Сваком захтеву треба доделити идентификациони број (рецимо НФ.1, НФ.2, ...).

Сваки нефункционални захтев треба исказати јасном формулацијом и квантификованим или на други начин прецизираним ограничењима (на пример: Време чекања одговора на захтев корисника ће бити мање од 5 секунди или систем ће подржавати најновију као и старије верзије „web browser“- а и сл.) Класификација нефункционалних захтева врши се по приоритету и важности (на исти начин као и функционални захтеви).

Табела 5. Нефункционални захтеви за систем праћења попуњености канти

|  |  |  |  |
| --- | --- | --- | --- |
| Редни број | Захтев | Сегмент | Приоритет |
| НФ.1 | Систем ће омогућити приступ само власницима и корисницима које су власници одредили | Безбедност | О |
| НФ.2 | Систем ће бити редовно ажуриран | Одржавање | О |
| НФ.3 | Време чекања на захтев корисника биће мање од 5 секунди | Перформансе | О |
| НФ.4 | Систем ће бити подржан на Windows (7,8,10) платформама | Перформансе | В |
| НФ.5 | Систем ће бити доступан 24h | Доступност | М |
| НФ.6 | Систем ће бити подржан на Linux системима | Перформансе | Б |
| НФ.7 | Систем ће бити омогућен за коришћење путем мобилног телефона | Перформансе | Б |
| НФ.8 | Систем ће омогућити безбедно чување података у бази података | Безбедност | О |

* обавезни (О)
* важни (В)
* могући (М)
* будући (Б)

## Процесни захтеви

Процесни захтеви се односе на процес развоја софтвера (нпр. треба да буде написан у програмском језику C#). Поставља их сама организација – произвођач софтвера или њен купац или нека трећа заинтересована страна. Могу се генерисати имплицитно од неких других софтверских захтева, као нпр. када је реч о избору верификационе технике или технике анализе за смањење грешака које могу угрозити поузданост, итд. Сваком захтеву треба доделити идентификациони број (нпр. ПЗ.1, ПЗ.2, итд). Ови захтеви би требало да буду исказани јасно и конкретно и да постоји могућност да се класификују по приоритету и важности на исти начин као и функционални захтеви.

Табела . Процесни захтеви

|  |  |  |  |
| --- | --- | --- | --- |
| Редни број | Захтев | Карактеристика | Приоритет |
| ПЗ.1 | Систем ће бити пројектован програмским језиком C# и Ангулар оквиру | Развојно окружење | О |
| ПЗ.2 | Подаци о попуњености канти чување се у Microsoft SQL бази | Развојно окружење | О |
| ПЗ.3 | База ће бити тестирана реалним подацима | Тестирање | О |
| ПЗ.4 | Систем ће имати сигурносну валидацију | Сигурност | Б |
| ПЗ.5 | Систем ће захтевати мало ресурса за покретање | Ресурси | М |

* обавезни (О)
* важни (В)
* могући (М)
* будући (Б)

# УМЛ дијаграми модела 4+1

Graphical user interface

Description automatically generated with low confidence

Слика . УМЛ модел 4+1

## Логичко гледиште

Гледиште дефинише начин на који се посматра предметни систем, поглед је оно што се види таквим посматрањем. Логичко гледиште подржава функционалне захтеве. Логички поглед специфицира сервисе које систем треба да обезбеди крајњим корисницима.

Систему може приступити администратор који врши измену над системом.

Систем треба да обезбеди функционалности овлашћеном кориснику система брисање података о канти, капацитету и попуњености.

### Дијаграми класа

Дијаграм класа је статички модел који приказује класе и везе између класа у систему. Елементи дијаграма класа су: класа, атрибути и методе. Класа је опис скупа објекта са сличним атрибутима, заједничким операцијама (методама) и везама са другим објектима и представља се подељеним правоугаоником.

Атрибути су својства класе која описују карактеристике објеката и може имати свој тип, као што су нпр. интегер, реал, поинт, боолеан. Методе су акције или функције које класа може изводити и може манипулисати атрибутима.

Релације представљају везу између објеката једне или више класа и могу бити: асоцијација, агрегација, генерализација, зависност и реализација. Представља се линијом. Класа такође може да наследи атрибуте и операције од изворне класе.

Diagram

Description automatically generated

Слика . Дијаграм класа

## Процесно гледиште

Процесно гледишта подржава нефункционалне захтеве који су у надлежности девелопера и интегратора. Процесни поглед специфицира процесну архитектуру система.

Програмер треба да испланира систем тако да испуњава следеће захтеве:

* систем ће онемогућити неовлашћеним лицима приступ тако што ће само утврђени корисници и власници имати приступ
* систем редовно мора бити ажуриран,
* сви подаци ће бити записани у бази података,
* систем ће подржавати Windows и Linux системе
* систем ће бити доступан 24h,
* систем ће бити доступан и са мобилних уређаја

### Дијаграм активности

Дијаграми активности су намењени да моделирају динамичке аспекте система. Активност је спецификација понашања система. Приказују секвенцијалне и конкурентне кораке у процесу обраде, а њима се може приказати процедурална логика, пословни процес или ток посла. Слични су блок дијаграмима за опис алгоритама (додатно, подржавају паралелно понашање). Могу се третирати као специјалан случај дијаграма стања.

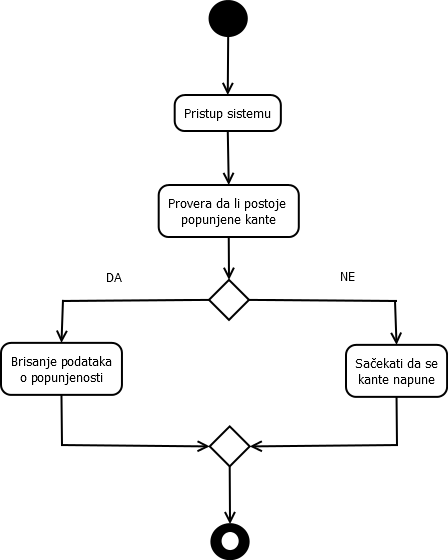
У чворовима овог дијаграма приказане су акције. Акције се имплементирају као методи класа или неке подактивности. Може се придружити класи или имплементацији случаја коришћења, и описује акције и активности које се извршавају у оквиру њих. Акција је основна јединица спецификације понашања и основни извршни елемент активности. Дијаграми активности садрже активности, транзиције између активности, тачке одлуке и синхронизационе сабирнице. Код UML-а активности се представљају као заобљени правоугаоници, транзиције се цртају као усмерене стрелице, тачке одлуке као ромбови, а синхронизациони барови као дебеле хоризонталне или вертикалне линије. На слици 1 су приказане: почетна тачка, завршна тачка, две активности и кретање процеса.

Diagram

Description automatically generated

Слика . Дијаграм активности

#### Дијаграм активности за проверу попуњености канте



Слика . Дијаграм активности – преглед и брисање података о попуњеним кантама

На овом дијаграму приказан је процес приступа систему, тако да када овлашћени корисници, односно власници треба да провере да ли постоје попуњене канте које је потребно испразнити, прво приступају систему. Следећи корак је провера да ли постоје канте које треба испразнити, ако постоје, прави се план пражњења и након што су канте испражњене, следи брисање података о попуњености канти, односно сетовања вредности испражњених канти на почетне. Уколико не постоје попуњене канте, корисници могу сачекати да се канте напуне и поново приступити систему.

## Развојно гледиште

Развојно гледиште подржава декомпозицију система на подсистеме. Развојни поглед: специфицира организацију софтверских модула и компонената, може да специфицира вишеслојну архитектуру система, представља основ за управљање софтверским пројектом.

Процесни захтеви које треба испунити при процесу развоја софтвера су:

* Систем ће бити кодиран у програмском језику C# и Аngular oквиру ,
* Подаци о загађењу као и све друге информације чуваће се Мicrosoft SQL бази,
* База ће бити тестирана над реалним подацима
* Систем ће бити комерцијалан

### Дијаграм компонената

Дијаграм компонената је специјалан случај дијаграма класе употребљен за описивање компонената унутар софтверског система. Компонента је физички и заменљиви део система који реализује скуп интерфејса. Овај дијаграм служи да прикаже физичку организацију и зависности између компонената. Садржи ствари и релације. У ствари спадају компоненте, интерфејси, портови, класе, пакети и др. док у релације спадају асоцијације, генерализације, зависности и реализације.

На приказу се налази дијаграм компонената који се односи на систем праћења попуњености канти.

Diagram

Description automatically generated

Слика . Дијаграм компоненти

## Физичко гледиште

Може се користити неколико различитих физичких конфигурација:

* за развој и тестирање система,
* за прилагођавање система различитим услугама или за различите клијенте.

Мапирање софтвера на чворове мора бити веома флексибилно и треба да има минималан утицај на изворни код. Систем се састоји од следећих компоненти:

* сервер на којим ће да буде инсталирана апликација
* сервер на ком ће да буде покренут SQL систем

## Случајеви коришћења

|  |
| --- |
| UseCase: Унос података о попуњености канти |
| **Кратак опис:** Овлашћени корисник врши унос података о кантама  **Актер:** Овлашћени корисник  **Предуслов:** Да не постоји канта у систему  **Последице:** Подаци о новим кантама су унети |

|  |
| --- |
| UseCase: Унос и брисање података о попуњености канти |
| **Кратак опис:** Овлашћени корисник врши брисање података о попуњености  **Актер:** Овлашћени корисник  **Предуслов:** Да је испражњена претходно попуњена канта  **Последице:** Сетоване су вредности попуњених канти на подразумеване. |

### Дијаграм случајева коришћења

Дијаграм случајева коришћења приказује скуп случајева коришћења и актера. Дијаграм визуализује понашање система, подсистема или чак класе или интерфејса. Такође служи да кориснику прикаже шта систем ради, а верификатору да провери како систем функционише.

Елементи дијаграма су:

* Учесник – актер (eng Actor)
* Случајеви коришћења (eng. Use cases)
* Везе
* Везе придруживање (eng. Association)
* Везе проширивање (eng. Extend)
* Везе укључивање (eng. Include)
* Везе генерализације (eng. Generalization)

Везе могу постојати између: актера и случајева коришћења (асоцијације), самих случајева коришћења (генерализација, проширење, укључивање) и актера (генерализација).

Diagram

Description automatically generated

Слика . Дијаграм случајева

# Презентациони слој

Крајњи циљ пројектног рада било је пројектовање и реализовање система који ће користити комуналним предузећима за организовање плана пражњења канти прегледом њихове попуњености.

Овлашћени корисници могу вршити потребне измене у апликацији и прегледати податке.

Коришћење апликације у велкој мери би допринело смањењу загађења животне средине и допринело здрављу људи.

Као што је већ поменуто, апликација је израђена помоћу програмског језика у C# и Angular оквира.

## Опис апликације

Приликом покретања апликације, овлашћеном кориснику се приказује страна која му омогућава да прегледа податке о кантама чија је попуњеност већа од 75% . Корисник на основу прегледаних података, може утврдити које канте се морају хитно испразнити и на тај начин саставити план по којем ће се оријентисати на пражњење канти. Након што испразни канту, корисник може обрисати податке о попуњености, односто поставити их на подразумеване, нулте вредности како би хардверски део који се налази у канти, могао да обавља несметано своју функцију и шаље податке о попуњености.

A picture containing graphical user interface

Description automatically generated

Слика . Графички нтерфејс за преглед података – први део

Chart, treemap chart

Description automatically generated

Слика . Графички интерфејс за преглед података – други део

# Закључак

Овај информациони систем може користити организацијама у циљу праћења попуњености канти. Веома је корисан, јер помаже у очувању животне средине и формирању добрих планова унутар организација.

Како је овај информациони систем развијен за потребе студирања, склон је надоградњи и побољшању функционалности. Оно што је информациони систем могао да садржи односи се на бољи кориснички доживљај, више функционалности за манипулисање подацима о кантама и више информација о начину коришћења. Међутим, иако је склон проширивању, систем је употребљив.

**Саржај слика**

[Слика 1. УМЛ модел 4+1 9](#_Toc104927135)

[Слика 2. Дијаграм класа 10](#_Toc104927136)

[Слика 3. Дијаграм активности 11](#_Toc104927137)

[Слика 4. Дијаграм активности – преглед и брисање података о попуњеним кантама 12](#_Toc104927138)

[Слика 5. Дијаграм компоненти 13](#_Toc104927139)

[Слика 6. Дијаграм случајева 14](#_Toc104927140)

[Слика 7. Графички нтерфејс за преглед података – први део 15](#_Toc104927141)

[Слика 8. Графички интерфејс за преглед података – други део 16](#_Toc104927142)

**Садржај табела**

[Табела 1. Спецификација носилаца 4](#_Toc104927168)

[Табела 2. Спецификација интереса 5](#_Toc104927169)

[Табела 3. Асоцијативна матрица носилаца и интереса 6](#_Toc104927170)

[Табела 4. Функционални захтеви за систем праћења попуњености канти 7](#_Toc104927171)

[Табела 5. Нефункционални захтеви за систем праћења попуњености канти 7](#_Toc104927172)

[Табела 6. Процесни захтеви 8](#_Toc104927173)