Универзитет у Београду

Електротехнички факултет



**СОФТВЕРСКИ ДОДАТАК ВЕБ ПРЕГЛЕДАЧА ЗА ОДЛОЖЕНО ЧИТАЊЕ ПОРУКА ЕЛЕКТРОНСКЕ ПОШТЕ**

**Д И П Л О М С К И Р А Д**

|  |  |
| --- | --- |
| Ментор: | Кандидат: |
| Доц. др Дражен Драшковић | Александра Богићевић 0390/17 |

Београд, децембар 2021.

**Садржај**

[1. Увод 3](#_Toc90763129)

[2. Екстензије електронске поште 4](#_Toc90763130)

[3. Начин рада екстензија 7](#_Toc90763131)

[**3.1.** **Манифест датотетка** 7](#_Toc90763132)

[***3.1.1.*** ***Обавезна поља*** 8](#_Toc90763133)

[***3.1.2.*** ***Препоручена поља*** 8](#_Toc90763134)

[***3.1.3.*** ***Опциона поља*** 9](#_Toc90763135)

[**3.2.** **Позадинска скрипта** 10](#_Toc90763136)

[**3.3.** **Скрипта садржаја** 10](#_Toc90763137)

[**3.4.** **Страна за опције** 10](#_Toc90763138)

[4. Опис система 11](#_Toc90763139)

[**4.1.** **Инсталација система** 11](#_Toc90763140)

[***4.1.1.*** ***Из угла програмера*** 11](#_Toc90763141)

[***4.1.2.*** ***Из угла корисника*** 13](#_Toc90763142)

[**4.2.** **Функционална спецификација** 13](#_Toc90763143)

[5. Имплементација 18](#_Toc90763144)

[**5.1.** **Опис коришћених технологија и структура система** 18](#_Toc90763145)

[**5.2.** **Реализација система** 19](#_Toc90763146)

[6. Закључак 29](#_Toc90763147)

[Литература 30](#_Toc90763148)

[Списак скраћеница 31](#_Toc90763149)

[Списак слика 31](#_Toc90763150)

[Списак табела 31](#_Toc90763151)

[Прилози 32](#_Toc90763152)

# Увод

Процењује се да око 4,66 милијарди људи на свету активно користи интернет, што чини 56,7% светског становништва [1]. Водећи веб прегледач са приближно 65% корисника интернета је *Google Chrome* [2]. Током свог развоја, док се борио да достигне титулу водећег веб прегледача, *Google Chrome* је децембра 2009. године покренуо своју галерију софтверских додатака веб прегледачу. Софтверски додаци веб прегледачу (енг. *web extensions*, у даљем наставку екстензије) представљају алатку која корисницима треба да омогући да прилагоди рад веб прегледача, као и да им обезбеди одређене олакшице у раду и додатне функционалности. Иако није први који је увео могућност инсталирања екстензија, то је свакако имало велики утицај на популарност. Већ годину дана касније, до децембра 2010. године, било је више од 8.500 екстензија, и тај број је временом само растао. Данас постоји преко 135.000 екстензија за *Google Chrome*. [3][4]

Екстензије нуде мноштво могућности, као што су алати за продуктивност, обогаћивање садржаја веб странице, а такође и забаву и игру. У овом раду ће бити приказана израда екстензије која омогућава корисницима одложено читање порука електронске поште (енг. *e-mail*).

У другом поглављу овог рада биће више речи о екстензијама електронске поште, као и о тренутном стању екстензија на тржишту.

У трећем поглављу биће речи о начину рада сваке екстензије.

У четвртом поглављу ће бити детаљно описан наш систем и начин на који систем ради.

У петом поглављу ће бити објашњена реализација нашег система.

# Екстензије електронске поште

Електронска пошта (е-пошта у наставку) у облику у каквом је сада познајемо, иако настала 1971. године, и даље представља важну улогу у друштву и један је од облика комуникације који и даље није превазиђен. Основна идеја електронске поште била је да се обезбеди комуникација између два корисника која се налазе на различитим локацијама (и уређајима), пружајући могућност да корисник који прима поруку може да ту поруку прочита када он то пожели. Данас је много више од тога.

Сада, када смо у 21. веку пред крај 2021. године, када се мноштво ствари дешава у онлајн простору, број корисника е-поште само расте. Е-пошта представља неопходно средство пословне комуникације, а такође и неопходан је део налога на свим друштвеним мрежама и различитим платформама. Сваког дана се пошаље и прима више од 319 милијарди порука е-поште, а очекује се да ће се тај број само повећавати. [7] Е-пошта је доступна на свим уређајима који могу да се повежу на интернет, као што су рачунари, телефони, таблети и сл.

Постоји више платформа за е-пошту. Разлика између платформи се огледа у разилици у сигурности, као и у цени – нечије услуге су бесплатне, док нечије нису. Тренутно, убедљиво најпопуларнија и најзаступљенија платформа је продукт компаније *Google*, платформа *Gmail*, са преко 1,8 милијарди активних корисника . [8] [9]-da li doraditi ovaj deo istorijom o gmailu

Модерне платформе дају кориснику скоро потпуну контролу над својим поштанским сандучетом помоћу разних функционалности. *Gmail* се издваја од осталих са највише интегрисаних функционалности, као што су:

* филтрирање нежељене поште
* опције претраге и напредне претраге
* слање великих прилога помоћу платформе *Google Drive*
* директно превођење порука е-поште
* помоћ при састављању поруке
* сигурност
* велико складиште

и много друих опција.

Растом и побољшавањем платформи јавила се жеља за додатним функционалностима које нису уграђене. Тада се јављају прве екстензије.

На тржишту постоји мноштво екстензија веб прегледача које побољшавају функционалности е-поште, а у наставку ће бити издвојене неке од популарнијих.

* ***Checker Plus for Gmail*** [10]

Олакшава коришћење и управљање више налога. Без потребе за отварањем сајта *Gmail*, пружа могућност прегледа порука као и њихово брисање. Постоји и опција читања порука наглас, што је велика предност и добра опција за особе са оштећеним видом.

* ***Gmail reverse conversation*** [11]

Екстензија која окреће редослед конверзације, тако да се прво види најновија порука.

* ***FlowCrypt: Encrypt Gmail with PGP*** [12]

Екстензија користи *PGP*(*Pretty Good Privacy*) енкрипцију за шифровање и потписивање порука које се шаљу путем е-поште.

* ***Sortd for Gmail*** [13]

Функционише као апликација за управљање пројектима, али је дизајниран да помогне при организазији поштанског сандучета. Омогућава сортирање по различитим колонама на основу критеријума по жељи.

Додати још.

Посматрањем тржишта дошли смо до закључка да је људима потребна помоћ у организацији, при чему им управо ови софтверски додаци и помажу. Према истраживању, просечан број порука електронске поште који запослена особа дневно прими износи око 120 порука.[6] Узимајући у обзир количину примљених порука, као и количну посла којом је просечна особа дневно окупирана, честа је појава да неке поруке буду прочитане с недовољно пажње или заборављене. Управо ту се јавља инспирација за овај рад, а о томе ће више бити речено у четвртом поглављу.

# Начин рада екстензија

Као што смо већ рекли, софтверски додаци веб прегледача, тј. екстензије, представљају алат која корисницима треба да омогући да прилагоде рад веб прегледача, као и да им обезбеди одређене олакшице у раду и додатне функционалности. Екстензије су изграђене помоћу веб технологија као што су *HTML (HyperText Markup Language)*, *CSS(Cascading Style Sheet)* и *JavaScript*.

Екстензије раде помоћу корисничког интерфејса, на страни крајњег корисника, и АПИ-ја (*API – Application Programming Interface*), на страни програмера. Кориснички интерфејс обезбеђује начин за кориснике да управљају својим екстензијама, док АПИ омогућава коду екстензије да приступи функцијама самог претраживача.

Екстензија је изграђена од више компоненти. Компоненте могу бити:

* Манифест датотека
* Позадинска скрипта
* Скрипта садржаја
* Страна за опције

У наставку ће детаљније бити објашњена свака компонента.

## **Манифест датотетка**

Манифест датотека (манифест у наставку) је неизоставни део сваке екстензије. То је датотека у *JSON (JavaScript Object Notation)* формату која мора да носи назив *manifest.json* и садржи одређене информације које су неопходне веб прегледачу за процесуирање екстензије. Чине је одређена поља, од којих су нека обавезна, нека препоручена, а нека опциона, зависно од намене екстензије. У наставку следи опис могућих поља.

### ***Обавезна поља***

Обавезна поља се морају наћи у свакој екстензији. То су поља:

* ***manifest\_version***

Целобројно поље које специфира верзију манифеста. Дозвољене вредности су 2, за манифест 2, који представља старију верзију и 3, за манифест 3, који представља актуелну верзију.

* ***name***

Текстуално поље које идентификује екстензију и представља њено име. Може садржати максимално 45 карактера.

* ***version***

Поље попуњено једном или четири целобројним вредностима међусобно одвојеним тачкама. Целобројна вредност мора да буде број између 0 и 65535, инклузивно. Представља тренутну верзију екстензије која је инсталирана.

### ***Препоручена поља***

Препоручена поља су поља која нису обавезна, али је добра пракса да постоје. То су поља:

* ***action***

Дефинисањем овог поља омогућавамо коришчење АПИ-ја *chrome.action* помоћу ког можемо да контролишемо понашање иконице екстензије на траци са алаткама (енгл. *toolbar*) у веб прегледачу *Google Chrome*.

* ***description***

Текстуално поље које представља опис екстензије. Максимално може да садржи 132 карактера.

* ***icons***

Низ слика у *PNG* формату, различитих димензија, које репрезентују екстензију. Препорука је да се наведе више димензија, од тога се издвајају као најбитније величине 128х128 и 48х48.

### ***Опциона поља***

Опциона поља су поља која могу постојати, али и не морају, зависно од улоге и потребе екстензије. Постоји мноштво опционих поља. У наставку ће бити описана поља која имају највећу употребу.

* ***background***

Поље у ком се дефинише позадинска скрипта, која ће бити описана у тачки 3.2.

* ***content\_scripts***

Поље у ком се дефинише тзв. скрипта садржаја, која ће бити описана у тачки 3.3.

* ***options\_page***

Поље у ком се дефинишу стране за опције, које ће бити описане у тачки 3.4.

* ***permissions***

Поље у ком се дефинишу дозволе које су неопходне за рад екстензије.

## **Позадинска скрипта**

Екстензије су програми засновани на догађајима унутар веб прегледача. Екстензија ослушкује догађаје користећи скрипте у свом тзв. сервисеру у позадини (енг. *background service worker*). Сервисер у позадини се учитава по потреби и представља главног извршиоца кода.

## **Скрипта садржаја**

Скрипте садржаја су датотеке које се покрећу у контексту веб страница. Коришћењем *DOM (Document Object Model)* стандарда, скрипте су у могућности да читају детаље веб страница, уносе промене и прослеђују информације позадинској скрипти путем асинхроног прослеђивања порука.

Углавном се активирају на догађај који је регистровала позадинска скрипта.

## **Страна за опције**

Десним кликом на иконицу екстензије и одабиром опције „*Options*“ кориснику се отвара страна са опцијама. На овој страни корисницима је дозвољено да прилагоде понашања екстензије.

# Опис система

У овом поглављу биће детаљно описана екстензија за одложено читање порука електронске поште. Циљ рада је био да се направи софтверски додатак веб прегледачу који ће кориснику, у моменту када отвори поруку е-поште, понудити опцију да му за неко одређено време стигне обавештење за исту ту поруку, као подсетник да је касније прочита.

## **Инсталација система**

У овом делу биће објашњено како инсталирати екстензију на свом веб прегледачу. Биће приказана два случаја – инсталирање екстензије из угла програмера и из угла корисника.

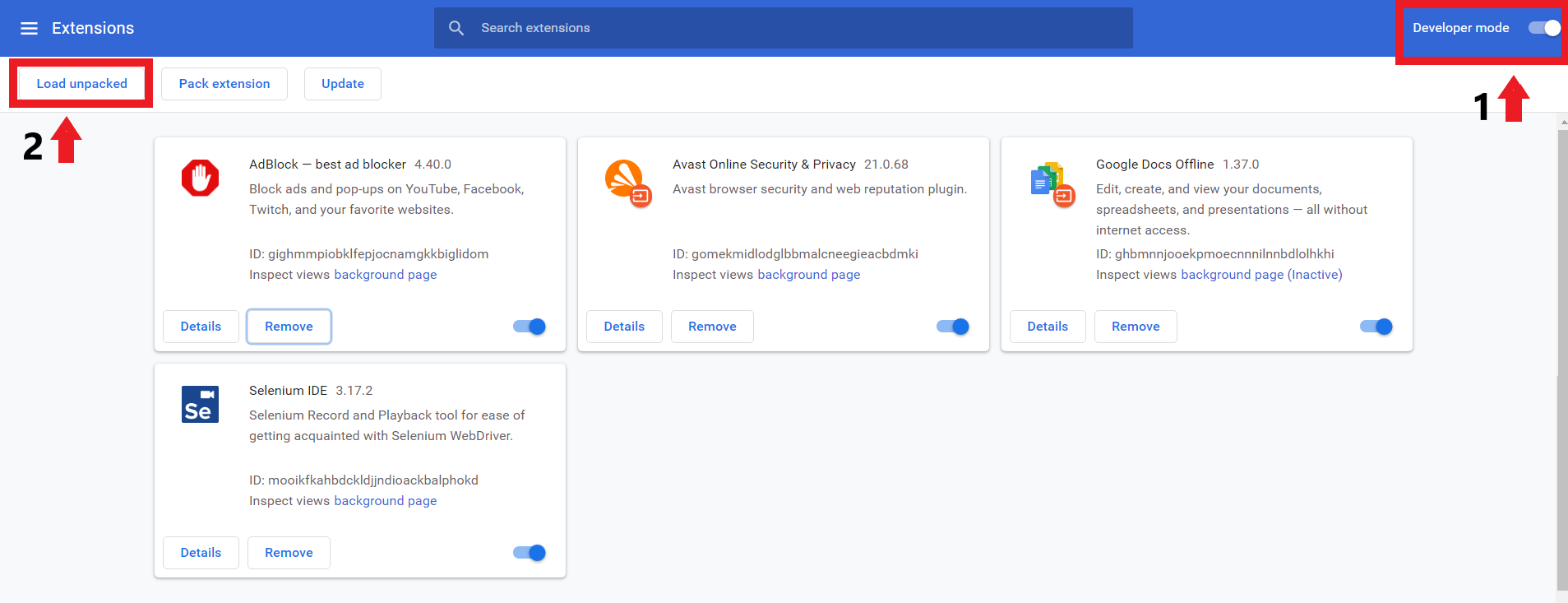
### ***Из угла програмера***

Инсталирање екстензије из угла програмера се односи на инсталирање екстензије која није објављена на платформи *Google Web Store*.

За почетак потребно је отворити *Chrome* мени. У одељку „*More tools*“ одабрати „*Еxtensions*“.

Тиме се отвара страница за управљање екстензијама.

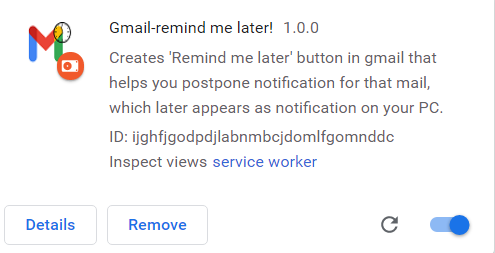
Када се налазимо на страници за управљање екстензијама, следеће што треба урадити је омогућити „*Developer mode*“ (слика 4.1.1, корак 1), а потом кликнути „*Load unpacked*“ (слика 4.1.1, корак 2).



Слика 4.1.1. Инсталирање екстензије, кораци 1 и 2

Након тога треба изабрати директоријум у којем се налази изворни код наше екстензије.

Одрадивши ове кораке, екстензија је успешно инсталирана на веб прегледачу и то је видљиво на страници за управљање екстензијама (слика 4.1.2).



Слика 4.1.2. Успешно инсталирана екстензија

### ***Из угла корисника***

Инсталирање екстензије из угла корисника се односи на инсталирање екстензије која је објављена на платформи *Google Web Store*.

За почетак потребно је отворити *Google Web Store*. Уношењем имена екстензије у претрагу биће приказана наша екстензија. Кликом „*Add to Chrome*“ корисник успешно инсталира екстензију.

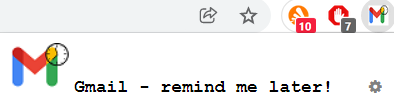
## **Функционална спецификација**

По инсталирању екстензије у свој веб прегледач, кориснику се у траци са алаткама приказује иконица са слике 4.2.1.



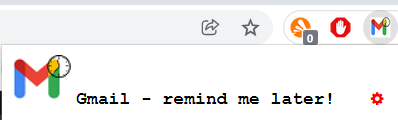
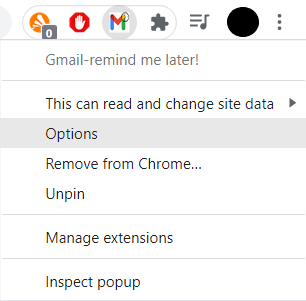
Слика 4.2.1. Трака са алаткама

Притиском на иконицу отвара се искачући прозор са слике 4.2.2.

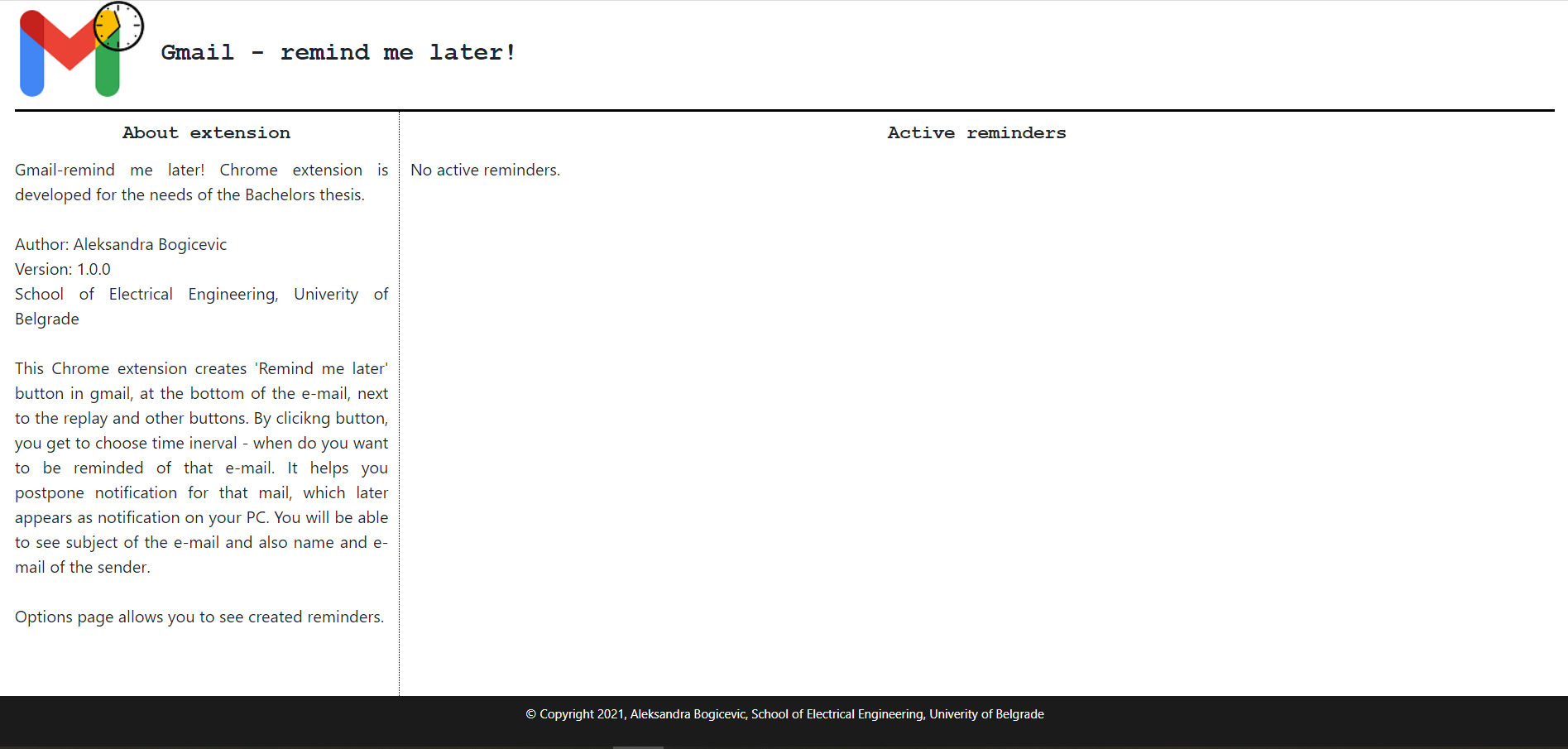


Слика 4.2.2. Искачући прозор

Кликом на точкић са десне стране на искачућем прозору (слика 4.2.3.а), или десним кликом на иконицу и одабиром опције „*Options*“ (слика 4.2.3.б), одлази се на страницу са опцијама. Страна са опцијама је приказана на слици 4.2.4.

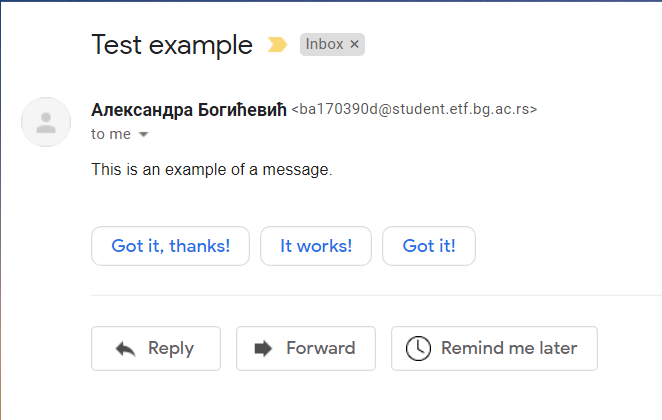
Слика 4.2.3.а. Одабир опција, начин 1 Слика 4.2.3.б. Одабир опција, начин 2



Слика 4.2.4. Страница са опцијама

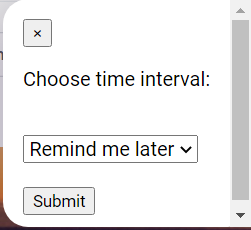
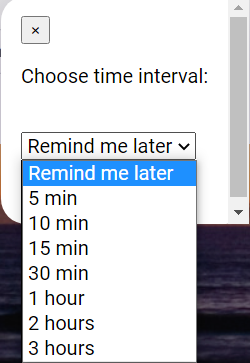
На страници са опцијама, са леве стране, кориснику се приказују основне информације о екстензији. Са десне стране, кориснику се приказује уколико има актуелних подсетника. Уколико нема, стоји порука обавештења да не постоје актуелни подсетници.

Уколико корисник жели да креира подсетник, то може урадити отварањем поруке е-поште и кликом на дугме „*Remind me later*“ (слика 4.2.5).



Слика 4.2.5. Креирање подсетника

Одабиром опције „*Remind me later*“,кориснику се приказује искачући прозор који захтева унос информације за колико времена треба да се огласи обавештење (слике 4.2.6.а и 4.2.6.б).

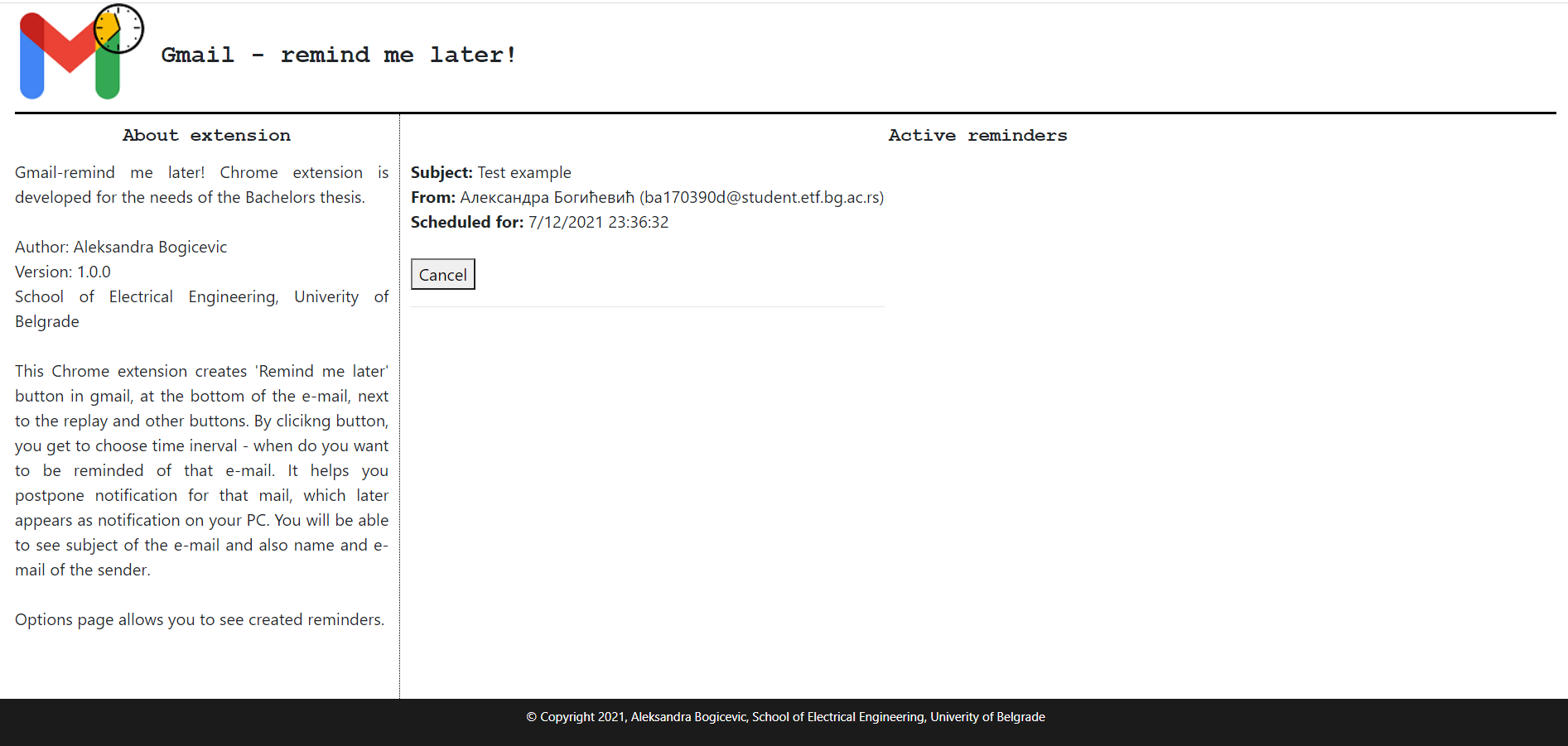
Слика 4.2.6.а Слика 4.2.6.б

Одабир временског интервала, део 1 Одабир временског интервала, део 2

Након што је корисник одабрао временски интервал, могуће је на страници са опцијама проверити да ли је заказано обавештење и за када је заказано (слика 4.2.7). Кориснику се приказују све информације у вези са обавештењем, а то су:

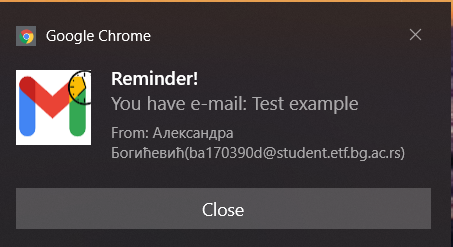
* наслов поруке
* ко је пошиљалац поруке – име и адреса е-поште
* за када је заказан подсетник

Одабиром опције „*Cancel*“, корисник је у могућности да откаже обавештење.



Слика 4.2.7. Преглед подсетника

Када дође време за оглашавање обавештења, кориснику се појављује обавештење на рачунару (слика 4.2.8).



Слика 4.2.8. Обавештење

Да би екстензија радила и да би обавештење могло да се прикаже, неопходно је да *Google Chrome*буде укључен.

Уколико је корисник заказао подсетник, а потом је време за подсетник истекло док *Google Chrome* није био активан, кориснику се неће у том моменту приказати обавештење. Међутим, први следећи пут када корисник буде приступио свом веб прегледачу, подсетник ће се активирати и кориснику ће се приказати обавештење.

# Имплементација

У овом поглављу биће детаљно описана имплементација екстензије за одложено читање порука електронске поште.

## **Опис коришћених технологија и структура система**

Екстензија је прављена помоћу *HTML, CSS JavaSctipt* технологија. Састоји из следећих датотека:

* *manifest.json*
* дадтотеке које чине страницу са опцијама:
  + *options.html*
  + *options.css*
  + *options.js*
* датотеке које чине кориснички интерфејс екстензије:
  + *popup.html*
  + *popup.css*
  + *popup,js*
* позадинска скрипта (*background.js*)
* скрипта садржаја (*content.js*)
* додаци
  + директоријум *images* са сликама неопходним за рад екстензије
  + директоријум са датотекама за подршку технологије *bootstrap*
  + директоријум *font-awesome-4.7.0*, са датотекама за подршку приказивања одређених иконица
  + *jquery-1.11.3.min*

## **Реализација система**

Најбитнији део екстензије је фајл манифест. У њему су прецизирана сва подешавања неопходна за сам рад екстензије.

На почетку је потребно дефинисати име екстензије и дати њен опис и верзију, као и верзију манифеста.

"name": "Gmail-remind me later!",

"description": "Creates 'Remind me later' button in gmail that helps you postpone notification for that mail, which later appears as notification on your PC.",

"version": "1.0.0",

"manifest\_version": 3

Потом, потребно је навести фајл позадинске скрипте, дозволе које екстензија захтева и скрипте садржаја.

Од дозвола, наша екстензија захтева дозволу за коришћењем складишта, за приказивање обавештења, за праћење која картица у прегледачу је активна, као и коришћење уграђеног аларма унутат прегледача*.* Накнадно ће бити објашњено зашто нам је неопходна свака од ових дозвола.

За скрипту садржаја наводе се неопходно фајлови у којима се налази изворни код, али такође и облик *URL (Uniform Resource Locators)* адресе на којој желимо да се наша скрипта примењује. У овом случају наводи се регуларни израз који обухвата све *URL* адресе које у себи садрже путању ка *Gmail* платформи.

"background": {

"service\_worker": "background.js"

},

"permissions": ["storage", "notifications", "activeTab", "tabs", "scripting", "alarms"],

"host\_permissions": [

"\*://mail.google.com//\*"

],

"content\_scripts": [

{

"matches": [

"\*://mail.google.com//\*"

],

"js": [

"content.js", "jquery-1.11.3.min.js"

],

"run\_at": "document\_end"

}

]

С обзиром да ће наша екстензија имати и корисничи интерфејс, као и страну са опцијама, потребно је навести и фајлове које то представљају. Поред тога наводимо и путање ка сликама које представљају иконице екстензије које ће се кориснику приказивати.

"action": {

"default\_popup": "popup.html",

"default\_icon": {

"16": "/images/icons/logo\_16.png",

"32": "/images/icons/logo\_32.png",

"48": "/images/icons/logo\_48.png",

"96": "/images/icons/logo\_96.png",

"128": "/images/icons/logo\_128.png"

}

},

"icons": {

"16": "/images/icons/logo\_16.png",

"32": "/images/icons/logo\_32.png",

"48": "/images/icons/logo\_48.png",

"96": "/images/icons/logo\_96.png",

"128": "/images/icons/logo\_128.png"

},

"options\_page": "options.html"

За крај се наводе и јавно доступни фајлови, којима је могуће приступити како би се приказао њихов садржај на одређеној интернет страници. У овом случају то је путања ка слици која ће бити приказана на страници поруке е-поште.

"web\_accessible\_resources": [{

"resources": ["/images/clock.png"],

"matches": ["<all\_urls>"]

}]

Објаснивши манифест датотеку која представља „подлогу“ за рад екстензије, можемо се фокусирати на позадинску скрипту која представља главног извршиоца кода, али такође упоредо обратити пажњу и на код који се извршава помоћу скрипте садржаја, на позив из позадинске скрипте.

Као што је већ споменуто у поглављу 3.2, позадинска скрипта ослушкује догађаје и у зависности од њих извршава или не извршава неки код. Главни догађај који се ослушкује у нашој позадинској скрипти је да ли је страница отвореног веб прегледача ажурирана, тј. да ли је дошло до неког поновног учитавања странице. Уколико је дошло де неких промена извршава се скрипта садржаја. С обзиром да смо скрипту садржаја подесили да се извршава само на страницама платформе *Gmail*, долазимо до закључка да ће се скипта садржаја извршавати при свакој промени на страници платформе *Gmail*.

chrome.tabs.onUpdated.addListener( function (tabId, changeInfo, tab) {

//...

})

Код који се извршава у скрипти садржаја представљају две функције.

removeAll**();**

addButton**();**

Прва функција која се позива је функција *removeAll*. Она је задужена да пре икаквих радњи ресетује вредност помоћних променљивих унутар локалног складишта (енг. *local storage)*, као и да уклони све измене са странице које су можда додате неким претходним учитавањем скрипте садржаја. Улога споменутих помоћних променљивих унутар локалног складишта ће бити касније објашњена.

**function** removeAll**(){**

**let** elem **=** **document.**getElementById**(**"remindme"**);**

**if** **(**elem **!==** undefined **&&** elem **!=null){**

elem**.**remove**();**

**}**

chrome**.**storage**.**local**.**set**({**"timer"**:** **-**1**},** **function()** **{});**

chrome**.**storage**.**local**.**remove**(**"subject"**);**

chrome**.**storage**.**local**.**remove**(**"name"**);**

chrome**.**storage**.**local**.**remove**(**"email"**);**

chrome**.**storage**.**local**.**get**(**"notifArray"**,** **function(**result**)** **{**

**if** **(**result**.**notifArray **==** undefined**){**

**var** arr **=** **new** Array**();**

chrome**.**storage**.**local**.**set**({**"notifArray"**:** arr**},** **function()** **{});**

**}**

**});**

**}**

Друга функција која се позива је функција *addButton*. Уз помоћ ове функције успевамо да, коришћењем *DOM* стандарда, у предео где се налазе већ уграђена поља која представљају дугме за одговор, проследи, проследи свима и сл, придодамо и ново дугме, са натписом „*Remind me later“*. То је одрађено инспекциом странице поруке *Gmail* е-поште, и проналаском класе која је додељена елементу коју окружује споменута поља. По проналаску класе, треба да дохватимо тај елемент, креирамо ново дугме, и додамо га уграђеном елементу који смо претходно дохватили.

Новом дугмету је придодат исти стил као и уграђеним. То је одрађено такође помоћу инспекције странице поруке *Gmail* е-поште и проналаском којим класама припадају уграђене опције, и потом додељивањем тих класа нашем новом дугмету.

function addButton(){

var div\_elem = document.getElementsByClassName("amn")[0];

if (div\_elem != undefined){

let newSpan = document.createElement("span");

newSpan.id = "remindme";

newSpan.tabIndex = "0";

newSpan.className = "ams remindMeClass";

newSpan.onclick = () => remindMeFunc();

newSpan.innerText = "Remind me later";

let extensionId = chrome.runtime.id;

let src = "url('chrome-extension://"+extensionId + "/images/clock.png')"

newSpan.style.backgroundImage = src;

newSpan.style.backgroundOrigin = "content-box";

newSpan.style.paddingLeft = " 5 px";

newSpan.style.backgroundPosition = "left";

newSpan.style.backgroundRepeat = "no-repeat";

newSpan.style.backgroundSize = "20 px";

div\_elem.append(newSpan);

}

}

Када смо додали ново дугме на страницу, време је да му доделимо и неку функцију. Постављамо да се притиском на наше дугме позива функција *remindMeFunc.*

Функција *remindMeFunc* се састоји из три целине.

Прва целина представља дохватање свих неопходних информација о поруци е-поште, које ће касније бити приказане кориснику у обавештењу. У те информације се сврставају наслов поруке е-поште, име пошиљаоца и адреса е-поште пошиљаоца. Као и претходно додавање дугмета, и ово је одрађено инспекцијом странице поруке *Gmail* е-поште и проналаском којим класама припадају одређене информације, и потом њиховим дохватањем и смештањем у помоћне локалне променљиве. Осим што се смештају у помоћне локалне променљиве, такође се смештају у помоћне променљиве унутар локалног складишта. Локалном складишту се приступа помоћу *chrome.storage API*, који омогућава чување, преузимање и праћење података.

//fetching e-mail subject

**let** subjectToSend **=** **document.**getElementsByClassName**(**"hP"**)[**0**].**innerText**;**

chrome**.**storage**.**local**.**set**({**"subject"**:** subjectToSend**},** **function()** **{});**

console**.**log**(**"Subject: " **+** subjectToSend**);**

//fetching e-mail sender

**let** senderObj **=** **document.**getElementsByClassName**(**"gD"**)[**0**];**

**let** senderNameToSend **=** senderObj**.**getAttribute**(**"name"**);**

**let** senderMailToSend **=** senderObj**.**getAttribute**(**"email"**);**

console**.**log**(**"Name: " **+** senderNameToSend**);**

chrome**.**storage**.**local**.**set**({**"name"**:** senderNameToSend**},** **function()** **{});**

console**.**log**(**"Email: " **+** senderMailToSend**);**

chrome**.**storage**.**local**.**set**({**"email"**:** senderMailToSend**},** **function()** **{});**

Друга целина функције *remindMeFunc* представља давање могућности кориснику да одабере за који временски период жели да му се прикаже обавештење за одређену поруку. Кориснику се та могућност пружа избацивањем искачућег прозора на којем има опцију да из падајућег менија одабере временски период, и потом кликом на дугме *Submit* потврђује свој избор. Та функционалност је одрађена креирањем модала који се појављује преко тренутне странице.

**const** modal **=** **document.**createElement**(**"dialog"**);**

modal**.**setAttribute**(**

"style"**,**`

height:150px;

border: none;

top:150px;

border-radius:20px;

background-color:white;

position: fixed; box-shadow: 0px 12px 48px rgba(29, 5, 64, 0.32);

`**);**

modal**.**innerHTML **=** `

<div class="modal-content">

<button class="close closeBtn" style:"align:right;">&times;</button>

</br>

<p>Choose time interval:</p> </br>

<div class="selectMenuDiv>

</div>

</div>

`**;**

Унутар модала се налази елемент који представља падајући мени са више опција. Свака опција поседује вредност, која представља временски интервал изражен у минутима.

**let** newSelect **=** **document.**createElement**(**"select"**);**

newSelect**.**className **=** "selectClass"**;**

**let** option0 **=** **document.**createElement**(**"option"**);**

option0**.**value **=** "0"**;**

option0**.**innerText **=** "Remind me later"**;**

newSelect**.**appendChild**(**option0**);**

...

**let** option1 **=** **document.**createElement**(**"option"**);**

option1**.**value **=** "1"**;**

option1**.**innerText **=** "5 min"**;**

newSelect**.**appendChild**(**option1**);**

...

На дну модала налази се дугме за потврду избора. Кликом на дугме чува се одабрана вредност временског интервала, а потом се из локалног складишта дохватају сачуване информације о поруци, и заједно са временским интервалом се прослеђују позадинској скрипти. Прослеђивање информација се реализује асинхроном комуникацијом. Скрипта садржаја шаље поруку, док позадинска скрипта све време ослушкује да ли је стигла нека порука. У коду испод приказано је слање поруке помоћу *chrome.runtime.sendMessage* из скрипте садржаја. Треба обратити пажњу да порука носи и информацију шта треба одрадити, под пољем *task.* У овом случају прослеђено је „*start*“, што значи да треба покренути тајмер.

chrome**.**runtime**.**sendMessage**({**

timerValue**:** timerVal**,**

task**:** "start"**,**

subject**:** s**,**

senderName**:** n**,**

senderMail**:** m

**}**

**,**

**function(**response**)** **{**

console**.**log**(**response**.status);**

**}**

**);**

Позадинска скрипта све време ослушкује да ли је стигла нека порука. Када стигне порука од скрипте садржаја, прво се проверава поље *task*, да би видели која порука је стигла, тј. шта треба обавити. У нашем случају прослеђују се само поруке које у пољу *task* имају вредност „*start*“, тако да је могуће само то извршавати.

chrome**.**runtime**.**onMessage**.**addListener**(**

**function(**request**,** sender**,** sendResponse**)** **{**

**if** **(**request**.**task **===** "start"**){**

...

**}**

**}**

**);**

Прво што треба урадити је преузети информације прослеђене поруком – наслов поруке е-поште, име пошиљаоца, адреса е-поште пошиљаоца и вредност временског интервала у минутима. Потом креирамо тајмер помоћу *chrome.alarms API*, који омогућава покретање одређеног кода у будућности у тачно одређено време.

Тајмер се креира помоћу функције *chrome.alarms.create*. Функцији се прослеђује јединствено име тајмера, као и вредност временског интервала за колико тајмер треба да се огласи.

chrome**.**alarms**.**create**(**alarmName**,** **{**delayInMinutes**:** request**.**timerValue**}** **);**

Јединственост имена аларма постигута је конкатенацијом ниске „*myAlarm*“ и ниске који означава временску ознаку (енг. *timestamp*).

**var** timestamp **=** Date**.**now**().**toString**();**

**var** alarmName **=** "myAlarm" **+** timestamp**;**

chrome**.**alarms**.**create**(**alarmName**,** **{**delayInMinutes**:** request**.**timerValue**}** **);**

Након покренутог тајмера потребно је проследити поруку скрипти садржаја да је функција успешно одрађена.

sendResponse**({status:** "started"**});**

Потребно је водити евиденцију о свим заказаним обавештењима, тј. покренутим тајмерима. Због тога чувамо све информације у локалном складишту у виду низа који садржи информације о сваком обавештењу, при том чувајући за свако обавештење исту временску ознаку која је додељена при креирању тајмера.

*Chrome.alarms API* ради у позадини и мери време. Оглашава се по истеку временског интервала, и то оглашавање је могуће регистровати ослушкивањем догађаја у позадинској скрипти. Потребно је за сваки случај проверити да ли се ради о нашем аларму проверавањем имена.

chrome**.**alarms**.**onAlarm**.**addListener**((**alarm**)** **=>** **{**

**if** **(**alarm**.**name**.**substring**(**0**,** 7**)** **===** "myAlarm"**)** **{**

...

**}**

**});**

Уколико је реч о нашем аларму, значи да је истекао тајмер за одређено обавештење. Из локалног складишта се на основу временске ознаке из имена аларма дохвата и уједно и уклања обавештење са истом временском ознаком и све његове информације.

За крај је потребно приказати кориснику обавештење. То је одрађено уз помоћ *chrome.notifications API* који креира обавештења и приказује их на корисниковом рачунару.

Обавештење се креира уз помоћ функције *chrome.notifications.create* којој се прослеђују тип обавештења, слика, наслов, порука и приоритет.

chrome**.**notifications**.**create**(**notifName**,** **{**

type**:** 'basic'**,**

iconUrl**:** 'images/icons/logo\_notif.png'**,**

title**:** "Reminder!"**,**

message**:** "You have e-mail: " **+** notif**.**mailSubject**,**

priority**:** 2**,**

contextMessage**:** "From: "**+** notif**.**senderName **+** "(" **+** notif**.**senderEmail **+** ")"**,**

requireInteraction**:** **true**

**});**

Овиме је објашњена цела логика иза функционисања наше екстензије. За крај остаје објаснити како је направљен кориснички интерфејс екстензије и страна са опцијама.

Кориснички интерфејс екстензије, представљен као искачући прозор кликом на иконицу екстензије, састоји се из једноставног *HTML* фајла *-popup.html* који укључује један *CSS – popup.css*.

Страна са опцијама се састоји такође из једноставног *HTML* фајла-*options.html,* који укључује један *CSS* фајл – *options.css* и један *JavaScript* фајл – *options.js*.

Оно што страна са опцијама омогућава кориснику, осим ишчитавања основнх информација о екстензији, јесте преглед свих креираних тајмера за обавештења и отказивање истих.

Преглед свих тајмера реализован је дохватањем низа обавештења из локалног складишта и његовим динамичким исписивањем на страници. При исписивању, поред сваког обавештења се креира и дугме *Cancel*, које омогућава отказивање.

Притиском на дугме *Cancel* обавештење се уклања из локалног складишта, а тајмер за то обавештење се брише уз помоћ функције:

chrome.alarms.clear(alarm);

По брисању, страница се освежава (поново учитава), и поново се динамички исписује низ обавештења.

# Закључак

# Литература

[1]

[2]

[3]

[4]

[5]

[6]

[7]

[8]

[9]

[10]

[11]

[12]

[13]

# Списак скраћеница

|  |  |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

# Списак слика

|  |  |
| --- | --- |
| Слика 1. |  |
| Слика 2. |  |
| Слика 3. |  |
| Слика 4. |  |
| Слика 5. |  |
| Слика 6. |  |
| Слика 7. |  |
| Слика 8. |  |
| Слика 9. |  |
| Слика 10. |  |
| Слика 11. |  |
| Слика 12. |  |
| Слика 13. |  |
| Слика 14. |  |
| Слика 15. |  |
| Слика 16. |  |