

Електронски систем за управљање возилима и возачким дозволама

Марко Митровић
Факултет техничких наука, Нови Сад
mitrovic.sr2.2020@uns.ac.rs

Сажетак

У овом раду детаљно се описује имплементација система за управљање возилима и возачким дозволама. За реализацију овог електронског система коришћен је Laravel фрејмворк у комбинацији са Docker контејнером и MySQL базом података. Применом овог електронског система омогућава се ефикаснији процес регистрације возила, издавање нових возачких дозвола и обнављање постојећих возачких дозвола. Ефикасност је постигнута увођењем аутоматизације и дигитализације процеса, што доводи до значајног смањења времена за обраду захтева. Овај систем такође омогућава приступ информацијама о возилима и возачким дозволама. Потенцијалне мане које би се могле испољити током коришћења овог система се огледају у сигурности података и у неусклађености са сличним системима који су тренутно у употреби. Коришћењем овог система, добијамо брже, ефикасније и сигурније управљање информацијама о возилима и возачкој дозволи, а резултат тога је унапређење рада министарства унутрашњих послова, што доприноси већем задовољству грађана који користе овај систем.

Кључне речи

Веб апликација; министарство унутрашњих послова; информације о возилима и возачким дозволама; подношење захтева за возила и возачке дозволе.

1. Увод

Ефикасно управљање и обрада информација у министарству унутрашњих послова су од изузетне важности за обезбеђивање сигурности и ефикасног рада осталих државних органа. Увођењем модерног информационог система, унапређење се рад министарства унутрашњих послова, кроз аутоматизацију кључних процеса и дигитализацију.

Проблем који би без система за управљање возилима и возачким дозволама био тешко решив, односи се на неефикасно управљање са подацима о возилима и возачким дозволама. Претходно постојећи процеси за издавање возачких дозвола и регистрације возила су временски захтевни, неефикасни и подложни су људским грешкама. Такође приступ информацијама о возилима и возачким дозволама је такође отежан, што отежава доношење судских одлука и управљање евиденцијама.

У овом раду прво се даје увид у тему система за управљање возилима и возачким дозволама у оквиру министарства унутрашњих послова. Затим се детаљније разматрају изазови и проблеми постојећег система. Након тога, описује се предложено решење. Затим се дефинише спецификација и начин имплементације система. Након завршене имплементације система, врши се демонстрација, а затим се приказују резултати током извршавања система. Потом се врши дискусија о предностима и могућим унапређењима. На крају, доносе се закључна разматрања и предлажу се будући правци истраживања.

2. Сродна решења и преглед коришћених технологија

У овом поглављу пружен је пример апликације која је намењена дигитализацији и аутоматизацији службе полиције у Министарству унутрашњих послова Републике Србије. Такође су наведене и технологије које су коришћене за имплементирање система са сличним функционалностима.

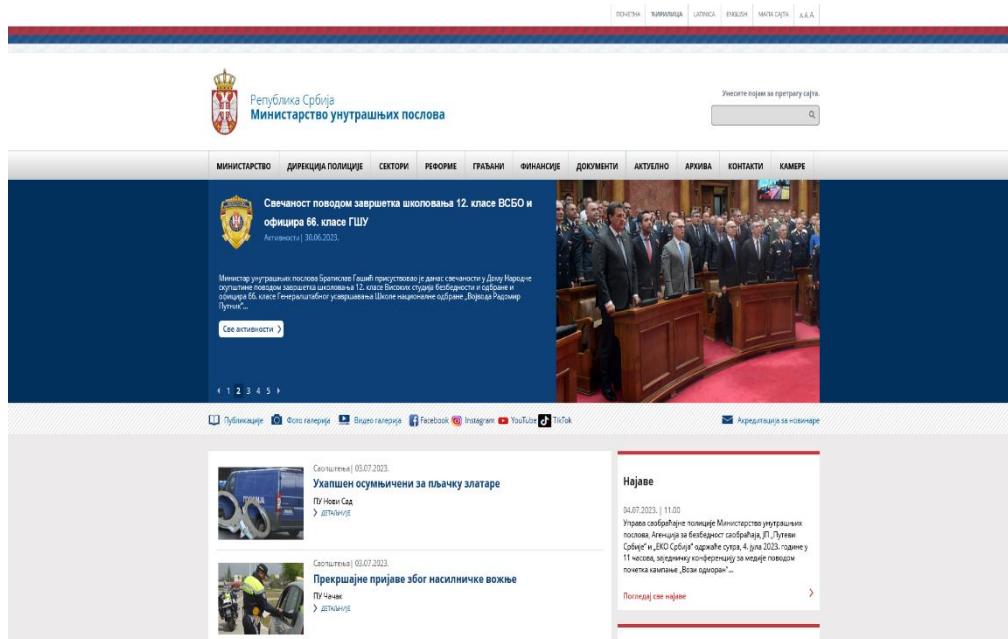
2.1. Министарство унутрашњих послова Републике Србије

Министарство унутрашњих послова Републике Србије [1] пружа грађанима широк спектар услуга путем својих дигиталних платформи, којима се приступа путем интернет странице. Неке од услуга обухватају могућност подношења захтева за различите врсте личних докумената (попут возачких дозвола и личних карти), проверавање информација о возачкој дозволи и о регистрованим возилима, као и приступ информацијама о безбедности и законима.

Међутим, постоје и одређене мане овог система. Неки грађани могу се суочити са недостатком интуитивности и комплексности навигације кроз интернет страницу, нарочито ако нису упознати како се употребљавају савремене технологије. Такође могу постојати и технички проблеми попут недоступности система у одређеним тренуцима или са пријављивањем на исти.

Упркос горе наведеним манама, овај дигитални систем грађанима је донео значајне предности. Овај систем је омогућио брз и једноставан приступ услугама, што ће смањити потребу за физичким одласком на шалтере и чекањем у редовима. Осим тога, овај систем је допринео ефикасности и транспарентности рада Министарства унутрашњих послова, олакшавајући комуникацију и размену информација између грађана и државних институција.

На *слици 1* приказан је изглед интернет странице Министарства унутрашњих послова:



Слика 1 – изглед интернет портала Министарства унутрашњих послова [2]

2.2. Коришћене технологије

У овом одељку наводе се технологије које су употребљене за решавање проблема електронског система за управљање возилима и возачким дозволама.

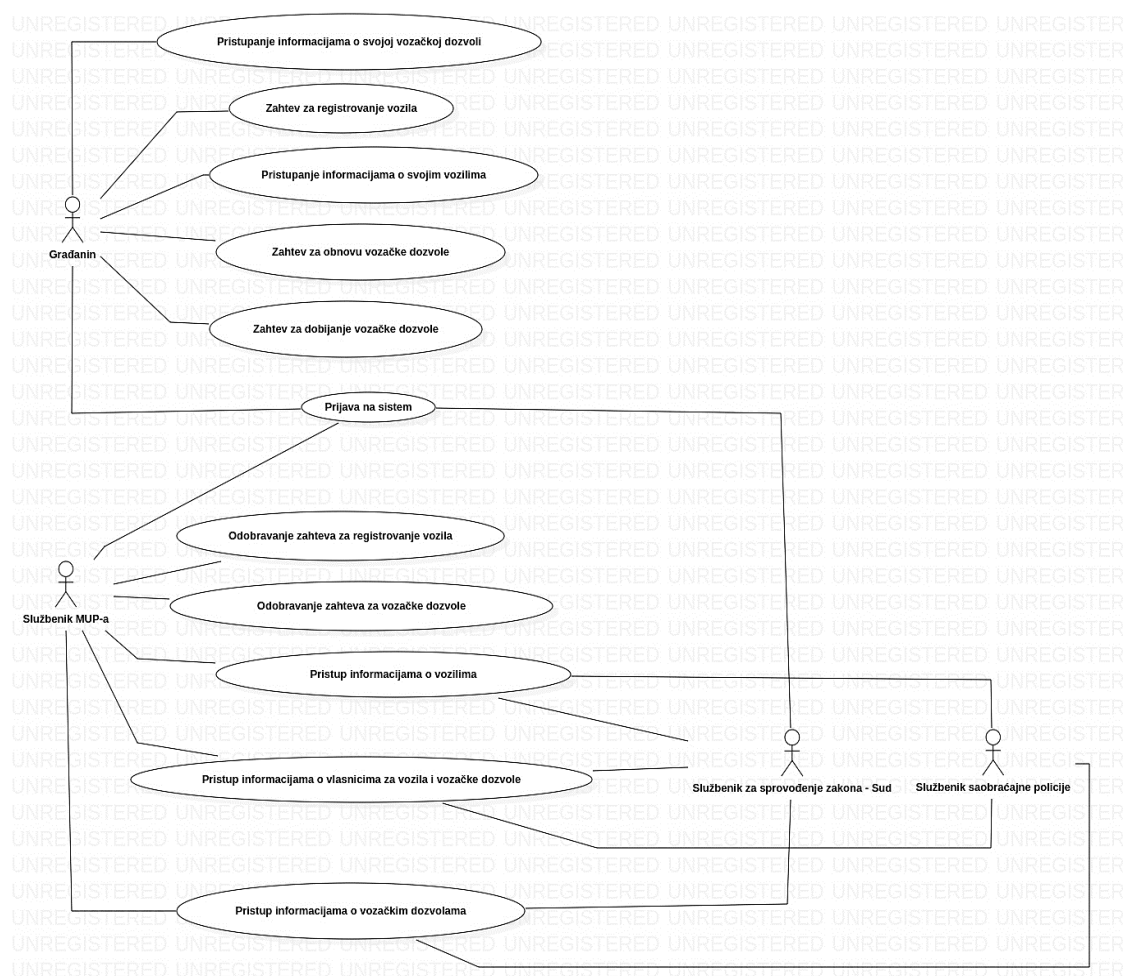
За израду овог електронског система коришћен је PHP [3] програмски језик у комбинацији са Laravel [4] фрејмворком. Архитектура овог система је таква да се садржај интернет странице генерише на серверској страни. За чување података користи се MySQL [5] база података. Комуникација између сервиса је реализована путем REST [6] архитектонског стила за комуникацију.

3. Спецификација захтева

У овом поглављу, детаљно су описани функционални и нефункционални захтеви софтверског решења које је представљено.

3.1. Функционални захтеви

У овом одељку су описани функционални захтеви који се односе на софтверско решење за електронски систем за управљање возилима и возачким дозволама. Захтеви су представљени путем UML дијаграма случаја коришћења, приказаног на слици 2:



Слика 2 – UML дијаграм случаја коришћења

У табели 1 је описан случај коришћења „Приступање информацијама о својој возачкој дозволи“:

Назив	Приступање информацијама о својој возачкој дозволи
Учесници	Грађанин, Службеник за спровођење закона, Службеник саобраћајне полиције
Предуслови	<ol style="list-style-type: none"> 1. Корисник има приступ интернету 2. Корисник поседује SSO лозинку и корисничко име 3. Корисник је успешно аутентификован као грађанин 4. Корисник поседује возачку дозволу
Кораци	<ol style="list-style-type: none"> 1. Након успешне пријаве, корисник бира опцију „Прикажи информације о својој возачкој дозволи“ 2. Систем даје кориснику приступ информацијама о његовој возачкој дозволи
Резултат	<ol style="list-style-type: none"> 1. Корисник има приступ информацијама о својој возачкој дозволи
Изузеци	<ol style="list-style-type: none"> 1. Корисник нема издату возачку дозволу

Табела 1 – Опис случаја коришћења „Приступање информацијама о својој возачкој дозволи“

У табели 2 је описан случај коришћења „Захтев за регистравање возила“:

Назив	Захтев за регистравање возила
Учесници	Грађанин
Предуслови	<ol style="list-style-type: none"> 1. Корисник има приступ интернету 2. Корисник поседује SSO лозинку и корисничко име 3. Корисник је успешно аутентификован као грађанин 4. Корисник поседује све неопходне информације о возилу
Кораци	<ol style="list-style-type: none"> 1. Након успешне пријаве, корисник бира опцију „Поднеси захтев за регистравање возила“ 2. Систем приказује форму за унос неопходних информација о возилу за које се подноси захтев 3. Корисник попуњава форму са информацијама о возилу 4. Систем потврђује пријем захтева 5. Систем бележи информације о времену и датуму подношења захтева 6. Корисник добија потврду да је захтев примљен
Резултат	<ol style="list-style-type: none"> 1. Систем бележи информацију о времену и датуму подношења захтева за регистрацију возила 2. Корисник добија потврду о пријему захтева за регистрацију возила путем интернет странице
Изузеци	<ol style="list-style-type: none"> 1. Неуспешна валидација попуњене форме за регистрацију возила

Табела 2 – Опис случаја коришћења „Захтев за регистравање возила“

У табели 3 је описан случај коришћења „Приступање информацијама о својим возилима“:

Назив	Приступање информацијама о својим возилима
Учесници	Грађанин
Предуслови	<ol style="list-style-type: none"> 1. Корисник има приступ интернету 2. Корисник поседује SSO лозинку и корисничко име 3. Корисник је успешно аутентификован као грађанин 4. Корисник има најмање једно регистровано возило
Кораци	<ol style="list-style-type: none"> 1. Након успешне пријаве, корисник бира опцију „Приступање информацијама о својим возилима“ 2. Систем приказује информације о возилима који се воде на уложеног корисника
Резултат	<ol style="list-style-type: none"> 1. Корисник је успешно приступио информацијама о својим возилима
Изузеци	<ol style="list-style-type: none"> 1. Корисник нема ниједно регистровано возило

Табела 3 – Опис случаја коришћења „Приступање информацијама о својим возилима“

У табели 4 је описан случај коришћења „Захтев за обнову возачке дозволе“:

Назив	Захтев за обнову возачке дозволе
Учесници	Грађанин
Предуслови	<ol style="list-style-type: none"> 1. Корисник има приступ интернету 2. Корисник поседује SSO лозинку и корисничко име 3. Корисник је успешно аутентификован као грађанин 4. Корисник поседује возачку дозволу
Кораци	<ol style="list-style-type: none"> 1. Након успешне пријаве, корисник бира опцију „Обнова возачке дозволе“ 2. Затим систем приказује форму где корисник може изменити (уколико жели) категорије возила за које жели да добије возачку дозволу 3. Систем обрађује захтев и аутоматски продужава возачку дозволу 4. Систем обавештава корисника да је захтев успешно обрађен
Резултат	<ol style="list-style-type: none"> 1. Корисник је успешно обновио своју возачку дозволу
Изузеци	<ol style="list-style-type: none"> 1. Корисник не поседује возачку дозволу

Табела 4 – Опис случаја коришћења „Захтев за обнову возачке дозволе“

У табели 5 је описан случај коришћења „Захтев за добијање возачке дозволе“:

Назив	Захтев за добијање возачке дозволе
Учесници	Грађанин
Предуслови	<ol style="list-style-type: none"> 1. Корисник има приступ интернету 2. Корисник поседује SSO лозинку и корисничко име 3. Корисник је успешно аутентификован као грађанин
Кораци	<ol style="list-style-type: none"> 1. Након успешне пријаве, корисник бира опцију „Поднеси захтев за добијање возачке дозволе“ 2. Затим систем приказује форму где корисник бира категорије возила за које жели да добије возачку дозволу 3. Систем генерише захтев за издавање возачке дозволе и бележи га у систем 4. Корисник добија потврду да је захтев примљен и да ће бити обрађен
Резултат	<ol style="list-style-type: none"> 1. Корисник је успешно обновио своју возачку дозволу
Изузеци	<ol style="list-style-type: none"> 1. Корисник не поседује возачку дозволу која је претходно истекла

Табела 5 – Опис случаја коришћења „Захтев за добијање возачке дозволе“

У табели 6 је описан случај коришћења „Пријава на систем“:

Назив	Пријава на систем
Учесници	Грађанин, Службеник Министарства унутрашњих послова (МУП)
Предуслови	<ol style="list-style-type: none"> 1. Корисник поседује исправне креденцијале 2. Корисник има приступ интернету
Кораци	<ol style="list-style-type: none"> 1. Корисник отвара интернет портал 2. Систем приказује форму за унос креденцијала 3. Систем проверава валидност унетих података 4. Ако су креденцијали исправни, систем преусмерава корисника на одговарајућу страницу 5. Ако су креденцијали неисправни, систем шаље поруку о грешци и враћа корисника на форму за пријаву
Резултат	<ol style="list-style-type: none"> 1. Корисник је успешно пријављен на систем и има приступ функционалностима система 2. У случају неуспешне пријаве на систем, корисник добија поруку о грешци и враћа корисника на форму за пријаву
Изузеци	

Табела 6 – Опис случаја коришћења „Пријава на систем“

У табели 7 је описан случај коришћења „Одобравање захтева за регистравање возила“:

Назив	Одобравање захтева за регистравање возила
Учесници	Службеник Министарства унутрашњих послова (МУП)
Предуслови	<ol style="list-style-type: none"> 1. Корисник има приступ интернету 2. Корисник поседује SSO лозинку и корисничко име 3. Корисник је успешно аутентификован као службеник 4. Службеник има приступ систему за обраду захтева за регистрацију возила 5. Постоји најмање један захтев за регистрацију возила у систему 6. Потребне информације су доступне
Кораци	<ol style="list-style-type: none"> 1. Након успешне пријаве, корисник бира опцију „Пристигли захтеви за регистрацију возила“ 2. Службеник врши преглед доступних захтева за регистрацију возила 3. Службеник проверава да ли су све потребне информације доступне и исправне 4. Службеник бира захтеве које жели да одобри, односно одбије 5. Уколико је службеник одобрио захтев, корисник ће имати могућност да види информације о тек регистрованом возилу 6. Уколико је службеник одбио захтев, систем обавештава корисника о одбијеном захтеву
Резултат	<ol style="list-style-type: none"> 1. Возило је успешно регистровано 2. Службеник завршава процес одобравања захтева за регистрацију возила
Изузеци	

Табела 7 – Опис случаја коришћења „Одобравање захтева за регистравање возила“

У табели 8 је описан случај коришћења „Одобравање захтева за возачке дозволе“:

Назив	Одобравање захтева за возачке дозволе
Учесници	Службеник Министарства унутрашњих послова (МУП)
Предуслови	<ol style="list-style-type: none"> 1. Корисник има приступ интернету 2. Корисник поседује SSO лозинку и корисничко име 3. Корисник је успешно аутентификован као службеник 4. Службеник има приступ систему за обраду захтева за возачке дозволе 5. Постоји најмање један захтев за возачку дозволу у систему 6. Потребне информације су доступне
Кораци	<ol style="list-style-type: none"> 1. Након успешне пријаве, корисник бира опцију „Обради захтеве за возачке дозволе“ 2. Службеник отвара листу захтева за возачке дозволе које чекају на одобрење 3. Службеник проверава да ли су сви потребни подаци исправни 4. Ако се утврде одређени проблеми, службеник може да одбије захтев уз образложење зашто је захтев одбијен 5. Ако је све у реду, службеник одобрава захтев за издавање возачке дозволе
Резултат	<ol style="list-style-type: none"> 1. Захтев за возачку дозволу је одобрен и корисник добија могућност провере информација о својој возачкој дозволи 2. Ако је захтев одбијен, корисник је обавештен о разлогу одбијања
Изузеци	

Табела 8 – Опис случаја коришћења „Пријава на систем“

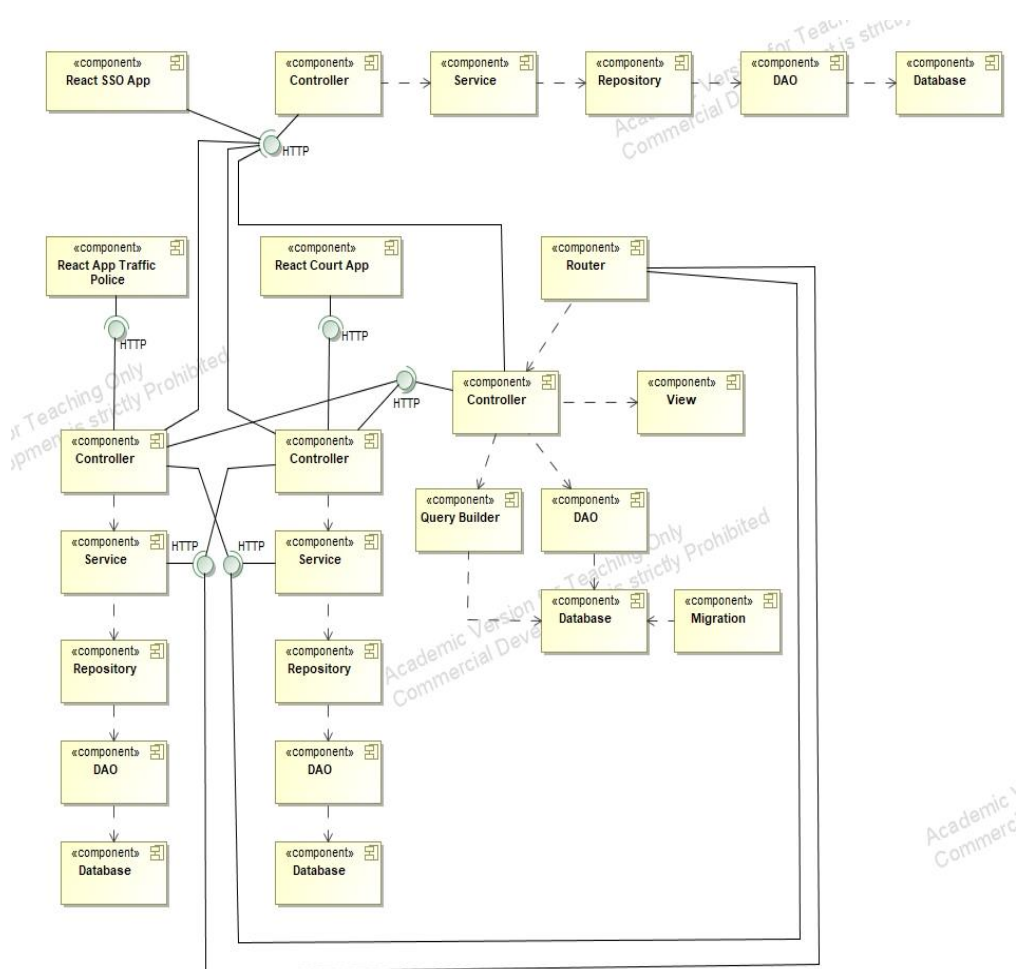
3.2. Нефункционални захтеви

У овом одељку су наведени нефункционални захтеви које електронски систем за управљање возилима и возачким дозволама треба да задовољи.

Систем мора бити у потпуности усклађен са системима Саобраћајне полиције и Суда, а да би се то постигло неопходно је устоставити стандардизован начин комуникације. Систем мора имати имплементирану контролу приступа, а такође мора бити компатибилан и са SSO системом еУправе, како би се на ефикасан и безбедан начин омогућио приступ и размена података између система. Софтверско решење мора бити једноставно за коришћење из разлога да се корисницима овог система олакша извршавање функционалних захтева. Овај систем такође мора да обезбеди и изузетно високу доступност, да би се постигло веће задовољство грађана.

4. Спецификација дизајна

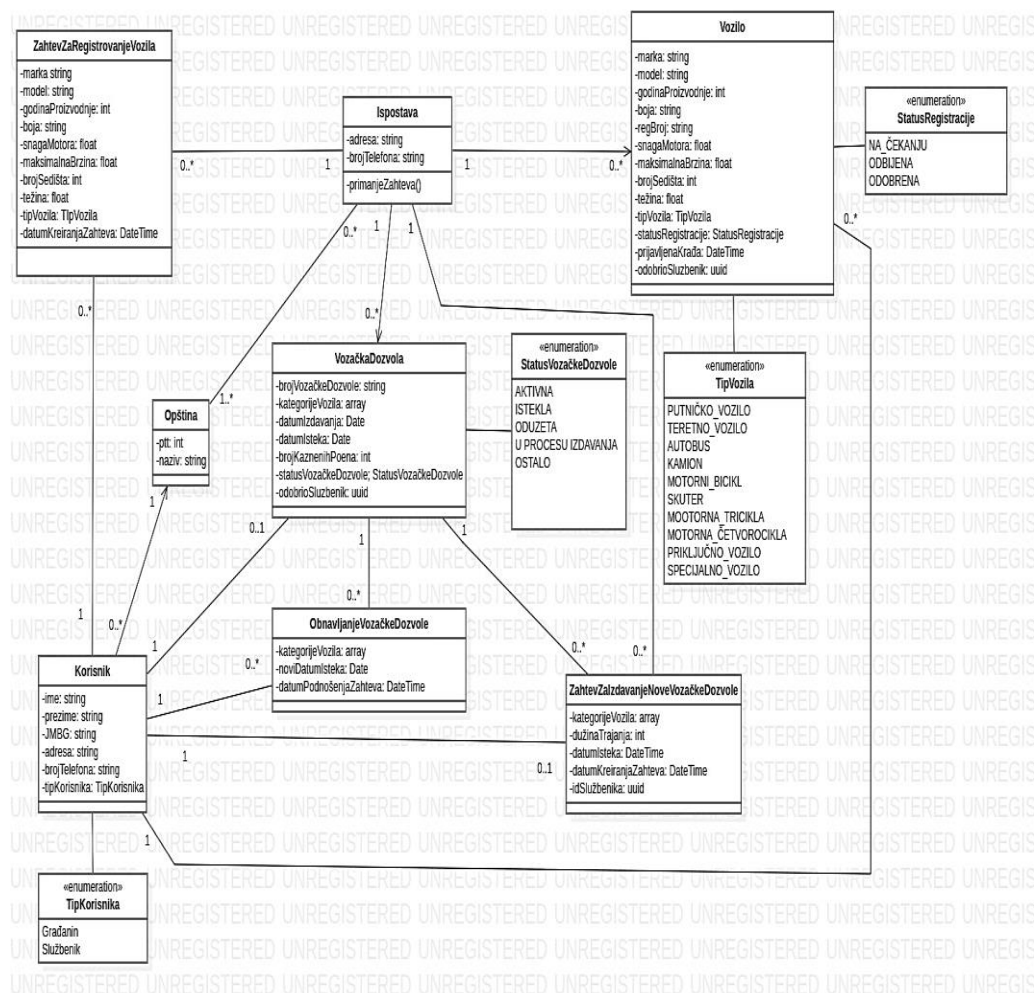
У овом одељку се објашњава дизајн софтверског решења за интернет портал Министарства унутрашњих послова. Архитектура овог софтверског решења је представљена UML дијаграмом компоненти. На том дијаграму такође можемо видети начин комуникације са осталим системима, као што је приказано на *слици 3*:



Слика 3 – UML дијаграм компоненти

Систем Министарства унутрашњих послова (МУП) комуницира са сервисима Саобраћајне полиције и Суда у циљу што ефикасније размене података. За аутентификацију и ауторизацију користи се SSO сервер са којим се комуницира путем REST архитектуре. Архитектура је таква да се садржај интернет странице директно генерише на серверу, пружајући бољу контролу и постојаност података.

На слици 4 приказан је дијаграм класа који илуструје структуру софтверског решења за управљање возилима и возачким дозволама:



Слика 4 – UML дијаграм класа

Класа „Корисник“ представља корисника система. Она садржи атрибуте који се користе за аутентификацију и ауторизацију. Ова класа се користи од стране SSO сервиса. У систему Министарства унутрашњих послова, ова класа се користи за пријаву службеника на интернет порталу Министарства унутрашњих послова. Ова класа садржи методе за пријаву и одјаву на систем.

Класа „Возачка дозвола“ представља возачку дозволу коју корисник може имати. Она садржи атрибуте као што су број возачке дозволе, категорије возила, датум издавања и датум истека, број казних поена, који службеник је одобрио и какав је тренутни статус возачке дозволе. Ова класа има методе за добијање информација о возачкој дозволи за улогованог грађанина, обнављање постојеће возачке дозволе, подношење захтева за нову возачку дозволу и управљање пристиглим захтевима за возачке дозволе.

Класа „Возило“ представља скуп возила које корисник може регистровати. Ова класа садржи атрибуте о возилу, о статусу регистрације, који службеник је одобрио или одбио захтев за регистрацију и кад је поднет захтев за регистровање возила. Она има методе за добављање информација о регистрованим возила за тренутно пријављеног грађанина, подношење захтева за регистровање новог возила и управљање пристиглим захтева за регистровање возила.

Класа „ЗахтевЗаРегистровањеВозила“ представља структуру података која се користи за захтев за регистровање возила. Класа садржи поља или својства која одговарају пољима форме. Ови атрибути омогућавају прикупљање потребних података за процес регистровања новог возила.

Класа „ЗахтевЗаИздавањеВозачкеДозволе“ представља структуру података која се користи за захтев за издавање нове возачке дозволе. Класа садржи поља и својства која одговарају пољима форме. Ови атрибути омогућавају прикупљање потребних података за процес издавања возачке дозволе.

Класа „ОбнављањеВозачкеДозволе“ представља структуру података која се користи за захтев приликом обнављања возачке дозволе. Класа садржи поља и својства која одговарају пољима форме. Ови атрибути омогућавају прикупљање потребних података за процес обнављања возачке дозволе.

5. Имплементација

Ово поглавље приказује на који начин су имплементиране функционалности за представљено софтверско решење (електронски систем за управљање возилима и возачким дозволама). За имплементацију овог софтверског решења коришћен је PHP програмски језик у комбинацији са Laravel фрејмворком. За имплементацију корисничког интерфејса коришћен је Blade шаблон, који је саставни део Laravel фрејмворка.

5.1. Приказ информација о возачкој дозволи

У овом одељку се објашњава на који начин је имплементиран приказ информација о возачкој дозволи на серверском делу апликације. За ову функционалност је коришћена метода која је приказана у *секцији 1*:

```
public function findById($jmbg): JsonResponse
{
    $drivingLicense = DrivingLicense::where('korisnik', $jmbg)->first();

    if (!$drivingLicense) {
        return response()->json(['error' => 'Vozacka dozvola nije pronadjena'], 404);
    }

    return response()->json($drivingLicense);
}
```

Секција 1 – Добављање информација о возачкој дозволи

Метода омогућава приказ различитих информација о возачкој дозволи на основу улазног параметра ЈМБГ (Јединствени матични број грађанина). Информације које се приказују укључују податке о власнику дозволе, категорије возачке дозволе, датум издавања и истека, и друге релевантне информације. Подаци неопходни за приказ се преузимају из базе података, а затим се форматирају у жељени приказ за корисника.

5.2. Захтев за регистравање новог возила

У овом одељку је објашњен начин имплементације функционалности захтева за регистравање новог возила на серверском делу апликације. За ову функционалност је коришћена метода која је приказана у *секцији 2*:

```

public function vehicleRegistrationRequest(Request $request): View|RedirectResponse
{
    $token = session()->pull('token', '');
    $authController = new AuthController();
    $isValidToken = $authController->validateToken($token);

    $user = session()->pull('user', '');
    if ($isValidToken) {

        $newVehicleRegistrationRequest = new Vehicle();

        $newVehicleRegistrationRequest->fill([
            'marka' => $request->brand,
            'model' => $request->model,
            'godina' => $request->year,
            'boja' => $request->color,
            'regBroj' => $request->registration_number,
            'snagaMotora' => $request->engine_power,
            'maksimalnaBrzina' => $request->max_speed,
            'brojSedista' => $request->num_of_seats,
            'tezina' => $request->weight,
            'tipVozila' => $request->vehicle_type,
            'statusRegistracije' => "NA_CEKANJU",
            'prijavljenaKradja' => null,
            'odobrioSluzbenik' => null,
            'korisnik' => $user['jmbg']
        ]);

        $newVehicleRegistrationRequest->save();

        return Redirect::back();
    } else {
        return view('authorization_failed');
    }
}

```

Секција 2 – Додављање информација о возачкој дозволи

Ова метода служи за обраду захтева за регистрацију новог возила. Процес укључује проверу валидности токена, преузимање информација о кориснику, креирање нове инстанце возила и унос података из захтева. Нови захтев се чува у бази података, а кориснику се враћа одговор или редирекција на претходну страницу. Уколико дође до неуспешне аутентификације, кориснику се приказује одговарајућа страница која га обавештава о томе. Ова метода осигурава успешно чување захтева за регистрацију новог возила на серверској страни апликације.

5.3. Приказ информација о возилима

У овом одељку је објашњен начин на који је имплементирана функционалност за приказ информација о возилима. Метода за ову функционалност је приказана у секцији 3:

```
/**
 * @throws ContainerExceptionInterface
 * @throws NotFoundExceptionInterface
 */
public function findById():
View|Application|Factory|\Illuminate\Contracts\Foundation\Applika-
tion
{
    $token = session()->pull('token', '');
    $authController = new AuthController();
    $isValidToken = $authController->validateToken($token);
    $user = session()->get('user', '');

    if ($isValidToken) {
        $vehicles = Vehicle::where('korisnik', $user['jmbg'])->
>get();
        return view('vehicle_registration_requests',
            ['vehiclesRegistration' => $vehicles, 'isOfficial' =>
false]);
    } else {
        return view('authorization_failed');
    }
}
```

Секција 3 – Приказ информација о возилима

Ова метода се користи за проналажење возила на основу јединственог матичног броја грађанина. Приликом извршавања ове методе, прво се проверава валидност токена који је претходно сачуван у сесији. Након тога се преузима информација о тренутној сесији корисника.

Уколико је токен валидан, врши се претрага возила која припадају грађанину са јединственим матичним бројем тог грађанина. Затим се резултати прослеђују одговарајућем приказу заједно са информацијом да корисник није службеник, тј. да није ауторизован за управљање захтевима за регистрацију возила.

У супротном, уколико токен није валидан, корисник се преусмерава на страницу која приказује поруку о неуспешној ауторизацији.

5.4. Захтев за обнављање возачке дозволе

У овом одељку је објашњен начин на који је имплементирана функционалност за обнављање возачке дозволе. Имплементација методе за ову функционалност је приказана у *секцији 4*:

```
public function renewDrivingLicenseRequest(Request $request): View|RedirectResponse
{
    $token = session()->pull('token', '');

    $authController = new AuthController();
    $officialController = new OfficialController();
    $drivingLicenseController = new DrivingLicenseController();

    $isValidToken = $authController->validateToken($token);
    $user = session()->pull('user', '');

    if ($isValidToken) {

        if ($user && isset($user['jmbg'])) {

            $drivingLicense = DrivingLicense::where("korisnik",
$user['jmbg']->first();

            if (!$drivingLicense) {
                return view('error');
            }

            if($drivingLicense->statusVozackeDozvole == "ODUZETA"
||
            $drivingLicense->statusVozackeDozvole == "OD-
BIJENA") {

                return view('error');

            } else {

                $drivingLicense->kategorijeVozila = serial-
ize($request->categories);
                $drivingLicense->datumIzdavanja = Carbon::today();
                $drivingLicense->datumIsteka = Carbon::today()-
>addYears(10);

                $drivingLicense->save();
            }
        }
    }
}
```

Секција 4 – Обнављање возачке дозволе

Ова метода се користи за обраду захтева за обнову возачке дозволе. Прима HTTP захтев и враћа приказ или преусмерава корисника на основу резултата обраде захтева.

У методи се прво преузима токен из сесије и врши се валидација токена позивом одговарајуће методе. Затим се добавља информација о тренутној сесији корисника.

Уколико је токен валидан и постоји информација о кориснику, врши се претрага возачке дозволе на основу уложеног корисника. Ако возачка дозвола не постоји, враћа се приказ са поруком грешке.

Уколико је статус возачке дозволе "ODUZETA" или "ODBIJENA", такође се враћа приказ са поруком грешке. У супротном, врши се ажурирање информација о возачкој дозволи са подацима из захтева. Категорије возила се серијализују, а датум издавања и истека се поставља на текући датум и датум за 10 година у будућности. Након ажурирања, информације о возачкој дозволи се чувају у бази података.

На крају, у зависности од резултата валидације токена и постојања информације о кориснику, враћа се приказ са поруком грешке или страница за успешну обнову возачке дозволе.

5.5. Захтев за издавање нове возачке дозволе

У овом одељку је објашњено на који начин је имплементирана функционалност подношења захтева за издавање возачке дозволе. Метода за ову функционалност је приказана у *секцији 5*:

```
public function drivingLicenseRequest(Request $request): View|RedirectResponse
{
    $token = session()->pull('token', '');
    $authController = new AuthController();
    $isValidToken = $authController->validateToken($token);
    $user = session()->pull('user', '');
    if ($isValidToken) {
        if ($user && isset($user['jmbg'])) {
            if(DrivingLicense::where('korisnik', $user['jmbg'])->first()){
                return view('error');
            } else {
                $newDrivingLicenseRequest = new DrivingLicense();

                $newDrivingLicenseRequest->fill([
                    'brojVozackeDozvole' => substr(Str::uuid()->toString(), 0, 8),
                    'kategorijeVozila' => serialize($request->categories),
                    'datumIzdavanja' => null,
                    'datumIsteka' => null,
                    'brojKaznenihPoena' => 0,
                    'statusVozackeDozvole' => "U PROCESU IZDAVANJA",
                    'odobrioSluzbenik' => null,
                    'korisnik' => $user['jmbg']
                ]);

                $newDrivingLicenseRequest->save();
                return Redirect::back();
            }
        } else {
            return view('error');
        }
    } else {
        return view('authorization_failed');
    }
}
```


Ова метода се користи за обраду захтева за возачку дозволу. Прима HTTP захтев и враћа приказ или преусмерава корисника на основу резултата обраде захтева.

У методи се прво преузима токен из сесије и врши се валидација токена позивом одговарајуће методе. Затим се врши преузимање информација о тренутној сесији корисника.

Уколико је токен валидан и постоје информације о кориснику, врши се провера да ли корисник већ поседује возачку дозволу. Ако корисник већ поседује возачку дозволу, враћа се преглед са поруком грешке.

Уколико корисник нема постојећу возачку дозволу, креира се нови захтев за возачку дозволу. Нова возачка дозвола се попуњава са информацијама из захтева.

Након креирања нове возачке дозволе, подаци се чувају у бази података.

На крају, у зависности од резултата валидације токена и постојања информација о кориснику, враћа се приказ са поруком грешке или страница за успешан захтев за возачку дозволу.

5.6. Управљање захтевима за возачке дозволе

У овом одељку је објашњено на који начин је имплементирана функционалност управљање пристиглим захтевима за возачке дозволе. Имплементација метода за ову функционалност је приказана у *секцијама 6 и 7*:

```
public function getPendingDriverLicenseRequests(): View|Application|Factory|\Illuminate\Contracts\Foundation\Application
{
    $token = session()->pull('token', '');
    $authController = new AuthController();
    $isValidToken = $authController->validateToken($token);

    if ($isValidToken) {
        $driverLicenseRequests = DrivingLicense::whereNull('odobrioSluzbenik')->
            orWhere('odobrioSluzbenik', '')->get();
        return view('driving_license_requests',
            ['driverLicenseRequests' => $driverLicenseRequests]);
    } else {
        return view('authorization_failed');
    }
}
```

Секција 6 – Добављање пристиглих захтева за возачке дозволе

Ова метода се користи за преузимање свих захтева за возачку дозволу који су на чекању. Примањем НТТР захтева враћа се приказ свих захтева за возачку дозволу који тренутно нису прегледани од стране службеника.

У методи прво преузимамо токен из сесије и вршимо валидацију токена позивом одговарајуће методе.

Ако је токен валидан, претражујемо базу података за све захтеве за возачку дозволу који још нису прегледани од стране службеника. Резултат претраге се прослеђује погледу.

Уколико токен није валидан, приказује се страница која обавештава корисника о неуспешној аутентикацији.

```
public function manageDrivingLicenseRequest(Request $request,
$id):
    View|Application|Factory|\Illuminate\Contracts\Foundation\Applika-
    tion
    {
        $action = $request->input('action', '');

        $token = session()->pull('token', '');
        $authController = new AuthController();
        $isValidToken = $authController->validateToken($token);
        $user = session()->pull('user', '');

        if ($isValidToken) {
            $drivingLicense = DrivingLicense::find($id);
            if ($drivingLicense) {
                $drivingLicense->odobrioSluzbenik = $user['jmbg'];
                if ($action === 'approve') {
                    $drivingLicense->statusVozackeDozvole = "AKTIVNA";
                    $drivingLicense->datumIzdavanja = Carbon::today();
                    $drivingLicense->datumIsteka = Carbon::today()-
                    >addYears(10);
                } elseif ($action === 'reject') {
                    $drivingLicense->statusVozackeDozvole = "OD-
                    BIJENA";
                }

                $drivingLicense->save();

                return view('driving_license_requests',
                    ['driverLicenseRequests' => DrivingLi-
                    cense::whereNull('odobrioSluzbenik')->
                    orWhere('odobrioSluzbenik', '')->get()]);
            } else {
                return view('error');
            }
        } else {
            return view('authorization_failed');
        }
    }
}
```

Ова метода се користи за управљање захтевима за возачку дозволу. Прима HTTP захтев и идентификациони број захтева над којим се управља. Као резултат враћа приказ са ажурираним списком захтева за возачке дозволе.

У методи прво преузимамо акцију (approve или reject) из HTTP захтева. Затим добављамо токен из сесије, затим вршимо валидацију токена позивом одговарајуће методе.

Ако је токен валидан, претражујемо базу података за захтев за возачку дозволу са одговарајућим идентификационим бројем. Ако постоји такав захтев, ажурирамо информације о захтеву у складу са изабраном акцијом.

Ако је акција "approve", постављамо статус возачке дозволе на "AKTIVNA", постављамо датум издавања на тренутни дан и датум истека на 10 година од тренутног дана. Ако је акција "reject", постављамо статус возачке дозволе на "ODBIJENA".

Након ажурирања информација, чувамо измене у бази података и враћамо преглед са ажурираним списком захтева за возачку дозволу који чекају на преглед.

Ако захтев са датим идентификационим бројем не постоји, приказује се страница која враћа грешку.

Уколико токен није валидан, приказује се страница која обавештава корисника о неуспешној аутентикацији.

5.7. Управљање захтевима за регистравање возила

У овом одељку је објашњено на који начин је имплементирана функционалност за управљање захтевима за регистравање возила. Метода која имплементира ову функционалност је приказана у *секцијама 8 и 9*:

```
public function getPendingVehicleRegistrationRequests():
View|Application|Factory|\Illuminate\Contracts\Foundation\Applica-
tion
{
    $token = session()->pull('token', '');
    $authController = new AuthController();
    $isValidToken = $authController->validateToken($token);

    if ($isValidToken) {
        $vehicleRegistrationRequests = Vehicle::whereNull('odobrio-
odobrioSluzbenik')->orWhere('odobrioSluzbenik', '')
        ->get();
        return view('vehicle_registration_requests',
        ['vehiclesRegistration' => $vehicleRegistrationRe-
quests, 'isOfficial' => true]);
    } else {
        return view('authorization_failed');
    }
}
```

Секција 8 – Добављање пристиглих захтева за возила

Ова метода се користи за добављање свих захтева за регистрацију возила који су тренутно на чекању. Метода враћа приказ са захтевима за регистрацију возила који још нису прегледани.

У методи прво преузимамо токен из сесије и вршимо валидацију токена позивом одговарајуће методе.

Ако је токен валидан, претражујемо базу података за захтеве за регистрацију возила који нису још прегледани. Преузимамо све такве захтеве и прослеђујемо их одговарајућем приказу. Приказу прослеђујемо и информацију да ли је корисник службеник или обичан грађанин.

Ако токен није валидан, приказује се страница која обавештава корисника о неуспешној аутентикацији.

```

public function manageVehicleRegistrationRequest(Request $request,
$id):
    View|Application|Factory|\Illuminate\Contracts\Foundation\Applica-
tion
    {
        $action = $request->input('action', '');

        $token = session()->pull('token', '');
        $authController = new AuthController();
        $isValidToken = $authController->validateToken($token);
        $user = session()->pull('user', '');

        if ($isValidToken) {
            $vehicle = Vehicle::find($id);
            if ($vehicle) {
                $vehicle->odobrioSluzbenik = $user['jmbg'];

                if ($action === 'approve') {
                    $vehicle->statusRegistracije = "ODOBRENA";
                } elseif ($action === 'reject') {
                    $vehicle->statusRegistracije = "ODBIJENA";
                }

                $vehicle->save();

                return view('vehicle_registration_requests',
                    ['vehiclesRegistration' => Vehicle::whereN-
null('odobrioSluzbenik')->
                        orWhere('odobrioSluzbenik', '')->get(), 'isOf-
ficial' => true]);
            } else {
                return view('error');
            }
        } else {
            return view('authorization_failed');
        }
    }
}

```

Секција 9 – Управљање захтевима за регистравање возила

Ова метода се користи за управљање над захтевима за регистрацију возила. Прима захтев и идентификатор захтева који се обрађује.

У методи прво преузимамо акцију из захтева (ако је дефинисана) користећи метод `input()` на захтеву, иначе сетујемо акцију на празан стринг. Затим преузимамо токен из сесије и вршимо валидацију токена позивом одговарајуће методе.

Уколико је токен валидан, претражујемо базу података за захтев за регистравањем возила са датим идентификатором. Ако такав захтев постоји, ажурирамо поље са информацијом који службеник је одобрио захтев.

На основу изабране акције, мењамо статус регистрације возила. Ако је акција "approve" (одобри), постављамо статус на "ODOBRENA", а ако је акција "reject" (одбиј), постављамо статус на "ODBIJENA".

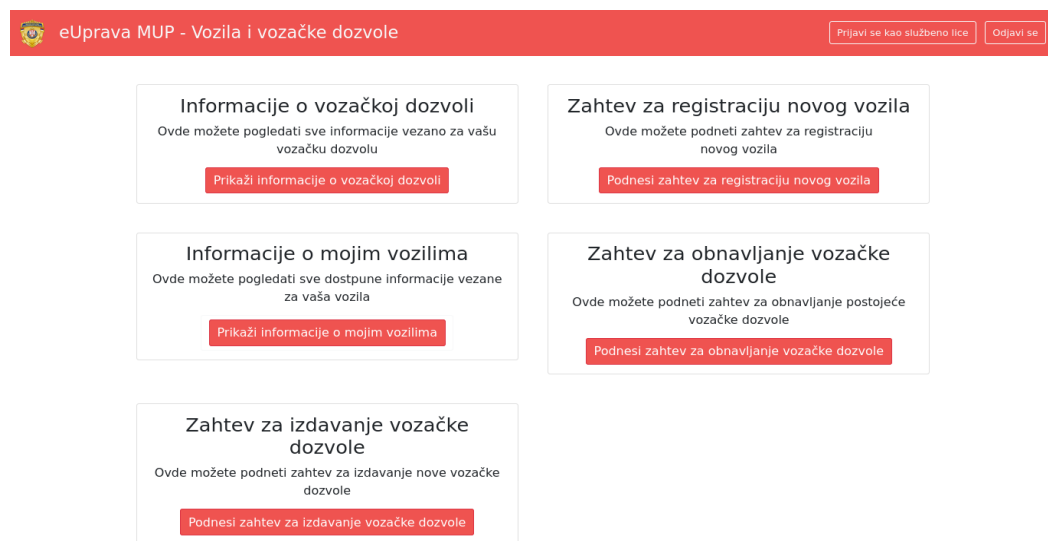
Након ажурирања, чувамо измене у базу података и на крају, враћамо приказ са свим захтевима за регистрацију возила који још нису прегледани.

Уколико захтев са датим идентификатором не постоји, враћа се приказ који враћа грешку.

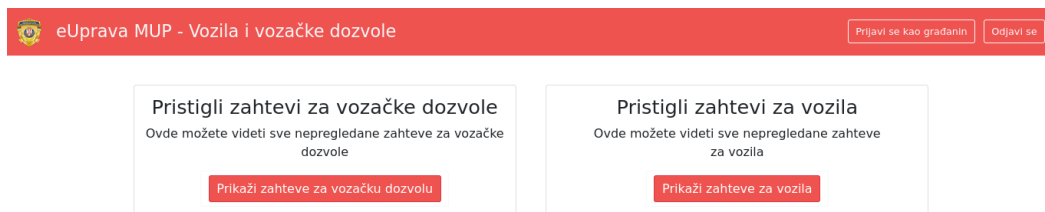
Уколико токен није валидан, приказује се страница која обавештава корисника о неуспешној аутентикацији.

6. Демонстрација

У овом поглављу се демонстрира начин употребе електроног система за управљање возилима и возачким дозволама (Министарства унутрашњих послова - МУП). Након успешне аутентификације кроз SSO сервис, кориснику се приказује почетни екран, чији изглед можемо видети на *сликама 5 и 6*:



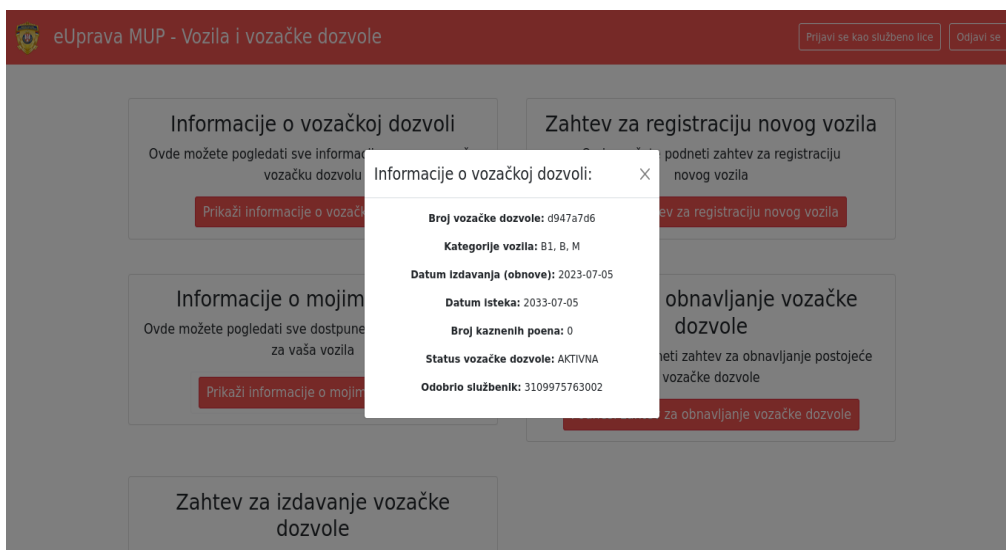
Слика 5 – Изглед почетне странице за грађане



Слика 6 – Изглед почетне странице за службена лица

У горњем делу екрана се налази навигациона трака која има опције за одјаву са система и могућност пријаве на систем као службено лице (уколико има додељено право за то). У централном делу странице се налазе картице са описаним услугама које су нам на располагању у систему. Кликком на дугме у некој од тих картица, кориснику се приказује форма или страница (у зависности од изабране радње) на којој се извршава та радња.

Бирањем опције „Информације о возачкој дозволи“, кориснику се приказују информације о возачкој дозволи (уколико је поседује) као што је приказано на слици 7:



Слика 7 – Изглед странице за приказ информација о возачкој дозволи

Бирањем опције „Захтев за регистрацију новог возила“, кориснику се приказује форма као на *слици 8* коју је неопходно попунити са валидним подацима и кликом на дугме поднети захтев и чекати на његово прегледање од стране службеника.

Слика 8 – Изглед странице за подношење захтева за регистрацију новог возила

Бирањем опције „Информације о мојим возилима“, кориснику се приказује нова страница са листом свих возила за које је корисник поднео захтеве за регистрацију, као можемо видети на *слици 9*:

Vrati se na početnu stranicu

Marka	Model	Godina proizvodnje	Boja	Registarski broj	Snaga motora	Broj sedišta	Maksimalna brzina	Težina	Tip vozila	Status registracije	Prijavljena krađa	Odobrio službenik
Peugeot	307	2001	Crna	BG1367DD	90	5	140	780	PUTNICKO_VOZILO	ODOBRENA		3109975763002

Слика 9 – Изглед странице за приказ информација о возилима за грађане

Бирањем опције „Захтев за обнављање возачке дозволе“, кориснику се приказује форма где може (уколико жели) да промени категорије у возачкој дозволи, а након тога кликом на дугме врши обнављање возачке дозволе (уколико је претходно поседује и да му није одузета), као што је приказано на *слици 10*:

The screenshot shows the eUprava MUP - Vozila i vozačke dozvole website. A modal window titled "Obnavljanje vozačke dozvole:" is open. It contains a dropdown menu for "Kategorije vozila:" with options AM, A1, A2, and A. Below the dropdown is a red button labeled "Podnesi zahtev". The background shows several service cards: "Informacije o vozačkoj dozvoli", "Zahtev za registraciju novog vozila", "Informacije o mojim vozilima", "obnavljanje vozačke dozvole", and "Zahtev za izdavanje vozačke dozvole".

Слика 10 – Изглед форме за обнављање возачке дозволе

Бирањем опције „Захтев за издавање возачке дозволе“, кориснику се приказује форма у којој бира категорије возила за које жели да добије возачку дозволу. Затим кликом на дугме врши подношење захтева и чека на његово прегледање од стране службеника. Изглед форме за захтев за издавање возачке дозволе приказан је на *слици 11*:

The screenshot shows the eUprava MUP - Vozila i vozačke dozvole website. A modal window titled "Zahtev za izdavanje vozačke dozvole:" is open. It contains a dropdown menu for "Kategorije vozila:" with options AM, A1, A2, and A. Below the dropdown is a red button labeled "Podnesi zahtev". The background shows several service cards: "Informacije o vozačkoj dozvoli", "Zahtev za registraciju novog vozila", "Informacije o mojim vozilima", "obnavljanje vozačke dozvole", and "Zahtev za izdavanje vozačke dozvole".

Слика 11 – Изглед форме за подношење захтева за нову возачку дозволу

Бирањем опције „Пристигли захтеви за возачке дозволе“, службенику се приказује страница са листом поднетих захтева за возачке дозволе које нису још прегледане. Службеник кликом на дугме бира да ли ће одређени захтев за возачку дозволу бити одобрен или одбијен. Након одабира опције, корисник који је поднео захтев има могућност да кроз „Информације о возачкој дозволи“ види коначан исход. На *слици 12* је приказан изглед странице:

Vrati se na početnu stranicu

Broj Vozacke Dozvole	Kategorije Vozila	Korisnik	Akcije
d947a7d6	B1, B, M	3109975763002	Odobri Odbij

Слика 12 – Изглед странице за управљање пристиглим захтевима за возачке дозволе

Бирањем опције „Пристигли захтеви за возила“, службенику се приказује страница са листом поднетих захтева за регистровање возила које нису још прегледане. Службеник кликом на дугме бира да ли ће одређени захтев за регистровање возила бити одобрен или одбијен. Након одабира опције, корисник који је поднео захтев има могућност да кроз „Информације о мојим возилима“ види коначан исход. На *слици 13* је приказан изглед странице:

Vrati se na početnu stranicu

Marka	Model	Godina proizvodnje	Boja	Registarski broj	Snaga motora	Broj sedišta	Maksimalna brzina	Težina	Tip vozila	ID korisnika	Akcije
Peugeot	307	2001	Crna	BG1367DD	90	5	140	780	PUTNICKO_VOZILO	3109975763002	Odobri Odbij

Слика 13 – Изглед странице за управљање пристиглим захтевима за регистровање возила

На *слици 14* можемо видети изглед странице уколико је уложеном кориснику истекла сесија или ако постоји било који други проблем са аутентификацијом:

Neuspešna autorizacija korisnika, prikaz ove stranice je blokirani!

Слика 14 – Изглед дијалога уколико постоје проблеми са аутентификацијом корисника

На *слици 15* можемо видети изглед странице уколико дође до грешке на серверској страни:

Greška - Internal Server Error!

Слика 15 – Изглед дијалога уколико дође до грешке на серверској страни

7. Закључак

У овом раду је представљено софтверско решење за електронско управљање возилима и возачким дозволама. Приказано решење омогућава корисницима да ефикасно обављају административне услуге везане за лична документа (као што су возачке дозволе и регистрација возила). Корисници путем интернет странице могу прегледати информације о својим возилима и возачкој дозволи, поднети захтев за регистровање возила и возачке дозволе без потребе за физичким доласком у испоставу.

Приказано решење електронског система за управљање возилима и возачким дозволама донело је низ предности и унапређења у односу на традиционалне папирне процесе. Омогућено је брже и ефикасније обрађивање и приступање информацијама, у великој мери је смањен број физичких посета испоставама и уштеђено је време грађанима. Такође, систем је пружио транспарентност и лакши приступ подацима грађанима.

С обзиром на то да у развоју већине анализираних решења учествују тимови инжењера, за очекивати је да та решења поседују одређене предности у односу на софтверско решење које је представљено у овом раду.

У будућности, систем би могао да се интегрише са све већим бројем осталих сервиса еУправе и додавањем више корисничких улога. Ово решење представља значајан корак у напретку и ефикасности управљања возилима и возачким дозволама, пружајући брже и ефикасније услуге за грађане.

У односу на постојећа решења, кориснички интерфејс је једноставнији и приступачнији за коришћење. Употреба модерног дизајна и добро организоване навигације чини овај интерфејс приступачнијим за све кориснике система. Поред једноставности и приступачности, систем пружа јасне и концизне кораке корисницима, што олакшава њихово сналажење и брзо обављање жељених радњи. Све ове карактеристике чине корисничко искуство бољим, стварајући предност у односу на постојећа решења.

Наравно, постоје и неке лоше стране које треба истаћи. Неки корисници могу имати потешкоћа у коришћењу електронског система, нарочито ако нису технички образовани или имају проблеме са приступом интернету. Такође, могу постојати и неке техничке грешке или багови у систему који могу утицати на негативно искуство корисника.

За проблеме са коришћењем система од стране корисника који нису технички образовани, могуће је обезбедити додатну обуку и подршку за кориснике, као и смањити број корака приликом коришћења система.

Могуће је развити и мобилну апликацију која би омогућила једноставнији и приступачнији начин за приступ услугама система које се могу извршавати путем интернет странице.

Да би се проблеми техничке природе свели на минимум, неопходно је континуирано надгледање и одржавање система. Тим за развој и техничку подршку треба да редовно отклања грешке и унапређује функционалности система.

8. Литература

[1] – Министарство унутрашњих послова Републике Србије, преузето 04.07.2023. са <http://www.mup.gov.rs/wps/portal/sr/>

[2] – Министарство унутрашњих послова Републике Србије, преузето 04.07.2023. са <http://www.mup.gov.rs/wps/portal/sr/> [снимак екрана]

[3] – PHP: Hypertext Preprocessor званична документација (приступљено 03.07.2023.), преузето са <https://www.php.net/docs.php>

[4] – Laravel: PHP фрејмворк - документација (приступљено 03.07.2023.), преузето са <https://laravel.com/docs/10.x>

[5] – MySQL: релациона база података - документација (приступљено 03.07.2023.), преузето са <https://dev.mysql.com/doc/>

[6] – REST архитектонски стил - документација (приступљено 03.07.2023.), преузето са <https://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm>