

**УНАПРЕЂЕЊЕ ПОСТОЈЕЋИХ АПЛИКАЦИЈА СА ПРОШИРЕНОМ РЕАЛНОШЋУ У ОБЛАСТИ ПНЕУМАТИКЕ****IMPROVEMENT OF THE AUGMENTED REALITY APPLICATIONS FOR EDUCATION IN PNEUMATICS**

Димитрије Милићевић, Слободан Дудић, *Факултет техничких наука, Нови Сад*

**Област – МЕХАТРОНИКА**

**Кратак садржај** - У овом раду приказане су и објашњене основне поставке развоја апликација са проширеном реалношћу са посебним освртом на препознавање слике и QR кода. Унапређене су две постојеће апликације са проширеном реалношћу од којих је једна намењена за олакшано учење пнеуматских компоненти и система а друга за олакшано учење пнеуматског управљања. У том смислу, исте су ажуриране додавањем QR кодова за пнеуматске управљачке шеме, додавањем нових задатака (развој нових 3Д модела, анимација и слично) и нових слика компоненти које ће бити имплементирани у новом издању Збирке решених задатака са теоријским основама из пнеуматског управљања. Анализирани су добијени резултати и извучени су потребни закључци.

**Кључне речи:** препознавање слике, проширена реалност, апликација

**Abstract** - In this paper is presented and explained the basic of development of augmented reality application with special emphasis on image and QR code recognition. Two existing augmented reality applications have been upgraded. One of them is used for easy learning of pneumatic components and systems and the other for easy learning of pneumatic control. In that sense, they have been updated by adding QR codes for pneumatic control schemes, adding new tasks (development of new 3D models, animations, etc.) and new images of components that will be implemented in the new edition of the workbook of solved tasks with theoretical bases of pneumatic control. The obtained results were analyzed and the necessary conclusions were drawn.

**Keywords:** image recognition, augmented reality, application

**1. УВОД**

Вековима је човечанство вртоглаво напредовало како у технолошком, тако и у осталим системима. технолошким напретком расла је потреба да се све више савремених технологија користи у образовне сврхе, ради олакшавања процеса учења и схватања наученог.

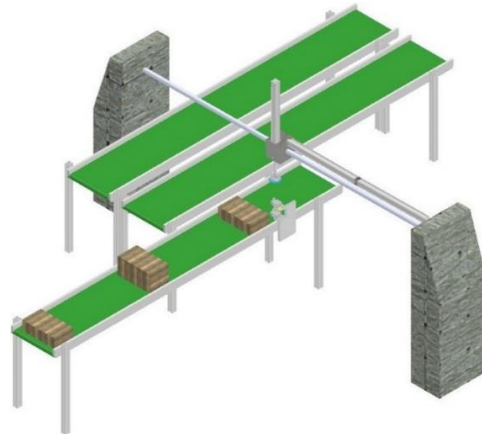
**НАПОМЕНА:**

Овај рад проистекао је из мастер рада чије ментор је био др Слободан Дудић, ванр. проф.

Стога је и у овом раду описан поступак надградње постојећих апликација са проширеном реалношћу под називом „Pneumatika 1“ и „Pneumatika 2“, које су креиране са циљем да олакшају студентима разумевање пнеуматских компоненти и система, као и пнеуматског управљања. Потребно је напоменути да се поменуте апликације користе заједно са Збирком задатака са теоријским основама из пнеуматског управљања [1].

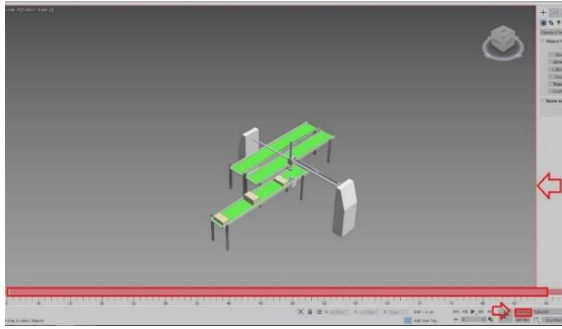
**2. КРЕИРАЊЕ 3Д МОДЕЛА И ЊИХОВА СИМУЛАЦИЈА**

Моделу уређаја за нове задатаке за збирку нацртани су у програмском окружењу Autodesk Inventor Professional 2020, и потом су симуларани у програмском окуржењу Autodesk 3ds Max Design 2020. Ради лакшег разумевања урађеног, у наставку је приказан један пример чији 3Д модел је дат на слици 1.



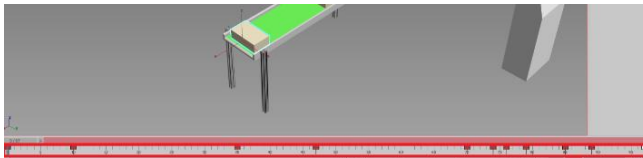
Слика 1. 3Д модел уређаја за један нови задатак за збирку

Након креирања модела, неопходно га је упамтити као степ фајл и потом додати у горе наведени програм за креирање анимације. Након што је додат модел, неопходно је додати материјал одређеним деловима како би изгледали што је могуће реалистичније и помоћу операције „Attach“ спојити поједине делове модела, ради олакшаног креирања симулације. Креирање симулације врши се тако што се притисне дугме „Toggle Auto Key Mode“, које се налази у лабели „Auto Key“ или притиком на дугме N (слика 2).



Слика 2. Притискање думета „Auto Key“ и промене на екрану

Као што се са слике изнад види, индикатор да смо активирали дугме „Auto Key“ је промена боје истог у црвену, као и промена позадине временског клизача и активни поглед. Након тога, ако се селекује било који објекат, тј. део у моделу и изврши било какву промену над њим, она ће бити забележена на временском клизачу у виду црвеног правоугаоника (слика 3).



Слика 3. Промена на временском клизачу

Подешавање дужине трајања анимације врши се у прозору „Time Configuration“ (слика 4).



Слика 4. Дугме за отварање прозора „Time Configuration“

Након завршетка креирања анимације, потребно ју је убацити у програм Unity, а затим ју је потребно упамтити као FBX фајл.

### 3. РЕАЛИЗАЦИЈА AR АПЛИКАЦИЈЕ

За надградњу Андроид апликација за збирку изграђене су горе наведене апликације као и неопходан видео материјал, који је интегрисан унутар поменутих апликација. Коришћени су следећи софтверски алати:

1. Autodesk Inventor Professional 2020,
2. Autodesk 3ds Max Design 2020,
3. FluidSim,
4. OBS Studio
5. Movavi Video Editor Plus 2020,
6. Visual Studio,
7. Android Studio,
8. Unity.

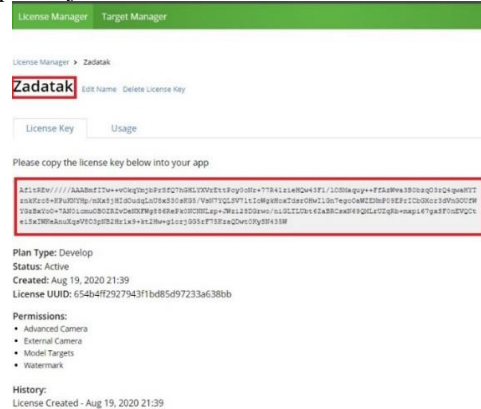
### 4. ИНСТАЛАЦИЈА НЕОПХОДНИХ ПРОГРАМСКИХ АЛАТА

Да би се кренуло у реализацију AR апликације неопходно је успешно инсталирати следеће ствари:

1. инсталирати JDK,
2. инсталирати Android studio,
3. инсталирати Unity и
4. искористити Vuforia окружење за креирање кључа за лиценцу и објекте препознавања

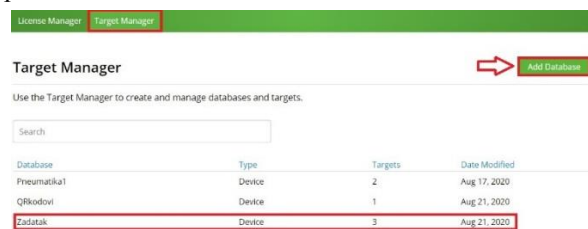
**Напомена:** Код програма Android studio треба водити рачуна да се у оквиру „SDK Manager“-а изабере варијанта Андроида „4.1 Jelly Bean“.

Након успешне инсталације прве три ставке, неопходно је отићи на званични сајт Vuforia, где се потребно регистровати и креирати лиценцу. У наредном кораку, кликом на име лиценце добијају се информације као на слици 5.



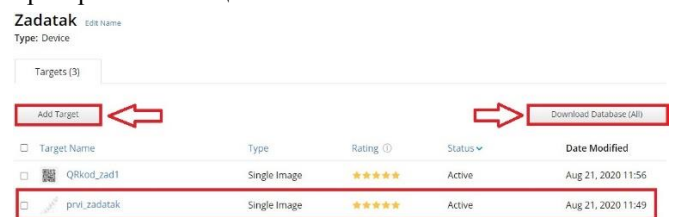
Слика 5. Информација о лиценци

Следеће што је потребно урадити је отворити таб „Target Manager“ (слика 6). Кликком на дугме „Add Database“ могуће је креирати базу са објектима препознавања.



Слика 6. Приказ Target Manager-a

Као што се види са слике изнад, присутне су три базе. Кликком на базу са именом „Zadatak“ појављује се прозор као на слици 7.



Слика 7. Приказ свих таргета везаних за базу изнад

За додавање нових таргета, неопходно је притиснути дугме „Add Target“ након чега се отвара нови прозор (слика 8). Таргет може бити слика, квадар, цилиндар или 3Д објекат. Како је задатак овог рада препознавање слика одабран је тип „Single Image“. Након тога треба притиснути дугме „Browse“ и позиционрати се на жељено место на рачунару те извршити одабир слике, која не сме бити већа од 2 Mb и мора бити .jpg или .png екстензије. У наредном кораку, неопходно је дефинисати ширину слике, док се висина аутоматски поставља. Овај параметар је једнак 100. Последња, али не и нужна опција, је промена имена. Овај поступак неопходно је поновити онолико пута колико постоји таргета за препознавање.



Слика 8. Додавање таргета у базу „Zadatak“

Када су сви таргети унети у базу, неопходно је преузети базни фајл притиском на дугме „Download Database (All)“ након чега се појављује прозор у коме треба одабрати „Unity Editor“ и потом притиснути думе „Download“.

Пре него што се крене на рад са Unity окружењем, треба напоменути да сваки објекат препознавања у бази података има своју оцену (енг. *Rating*) која се дефинише бројем звездица, као што се може видети на једном од примера, на слици 9.



Слика 9. Приказ информација о квалитету слике пнеуматског прекидача

Препознавање слика се може унапредити побољшавањем видљивости критичних детаља на самој слици, односно вршењем подешавања дизајна саме слике, оштрине, величине и начина на који је штампана. Такође се може побољшати детекција и праћење слике контролишући сам фокус камере апарата којим се снима. Сlike које се скенирају пожељно је да буду обасјане умереним и добро постављеним светлосним извором. С обзиром да препознавање пнеуматских шема у збирци [1] није увек успешно, уведени су специјално генерисани кодови, који се називају QR кодови (енг. *Quick Response Code*). Разлог увођења ових кодова је лака препознатљивост AR камером, добар контраст и оштре ивице, као и много тачака препознавања, што резултује брзим одзивом приликом препознавања (слика 10).



Слика 10. Детектованост QR кодова

## 5. РАЗВОЈ АПЛИКАЦИЈЕ УНУТАР UNITY ОКРУЖЕЊА

У Unity ожуењу потребно је креирати нови пројекат након чега је потребно избршити следећа подешавања [2]:

1. Потребно је из падајућег менија изабрати „File/Build Settings“ или притиском на Ctrl+Shift+B како би се ожуење подесило за рад са Андроид уређајима.
2. Притиском на „Other Settings/Identification“ отвара се прозор у коме је неопходно одабрати минималну верзију андроид оперативног система, која треба да буде „4.1 Jelly Bean“ и променити назив пакета. Такође, потребно је селектовати „Vuforia Augmented Reality Supported“ као и деселектовати опцију „Android TV Compatibility“.
3. Потом је потребно подесити SDK и JDK путање, убацити преузету базу и обрисати постојећу камеру, те убацити AR камеру, а унутар ње додати Vuforia лиценцу.

Након свих ових корака, програм Unity је подешен за рад. Следеће што треба урадити је да се за формира онолико „Image target“-а колико има таргета у бази Vuforia и за сваки везати одговарајућу слику. У наредном кораку, неопходно је формирати раван тако да постане дете сликовитог таргета, а потом за њега везати одговарајући видео. Наиме, када се од стране апликације препозна слика, преко ње се појављује ова раван и у њој се репродукује видео. Да би се ово и остварило, неопходно је за сваки „Image target“ везати „Ground Plane“ из датотеке под називом „Prefabs“. Ово се остварује једноставним превлачењем „Ground Plane“ преко „Image target“. Унутар „Ground Plane“ неопходно је креирати видео компоненту и за њу везати одговарајући видео. Када је све ово успешно одрађено, неопходно је покренути скрипту која се налази у „Image target“ под називом „DefaultTrackableEventHandler“.

У њој је потребно формирати променљиве типа *string* под називом *imeSlike* и типа *bool* под називом *ponasaoSliku*, као и променљиву типа *VideoPlayer*. Прва променљива служи за смештање пронађеног таргета а друга даје информацију да ли је слика пронађена. Потом унутар методе *Start()*, која се извршава на почетку, треба за сваки формирану променљиву додати *VideoPlayer* (код испод):

```
GameObject ZAD2GP = GameObject.Find("ZAD2GP");
ZAD2 = ZAD2GP.GetComponent<VideoPlayer>();
```

Овим је креиран „Game“ објекат који је везан за одговарајући „Ground Plane“. Затим се креираном „Game“ објекату додаје видео компонента и све смешта унутар променљиве *VideoPlayer*.

Унутар методе *OnTrackableStateChanged()* потребно је унутар *if* сегмета додати следећу функцију (код испод) у којој се смешта име таргета.

```
imeSlike = mTrackableBehaviour.TrackableName;
Debug.Log("Trackable " + imeSlike + " found");
```

У методи *OnTrackingFound()* потребно је поставити променљиву *ponasaoSliku* на *true* и формирати *switch* сегмет који ће на основу назива уоченог таргета покренути одговарајући видео или отићи на одређену интернет страницу (код на следећој страни):



```
switch(imeSlike){
    case"Zad2":ZAD2.Play(); break;
    case"Slika39c":
Application.OpenURL("https://www.festo.com/cat/sr_rs/pro
ducts_010204"); break;
... }

```

Приликом нестанка таргета из фокуса камере, неопходно је зауставити видео. Ово се врши унутар методе *OnTrackingLost()* постављањем променљиве *pronasaoSliku* на *false* и унутар *switch* сегмента треба додати следеће:

```
switch(imeSlike){
    case "QR12": QR12.Play(); break;...}

```

## 6. РАД СА АНИМАЦИЈАМА

Убацавање FBX фајлова у програм Unity врши се тако што се креира нова фасцикла у „*Hierarchy window*“ унутар фасцикле „*Assets*“. Убацавање се врши једноставним превлачењем анимационих фајлова у нову фасциклу. Након завршеног превлачења, потребно је сваки од тих фајлова селектовати и унутар „*Inspector window*“ а потребно је притиснути дугме „*Animation*“. Након тога, отвара се мени у коме је потребно преименовати видео клип за сваки видео на следећи начин *MODEL\*\**, где \*\* представља број задатка и опцију „*Loop Time*“ треба оставити нечекирану. Након што је урађено све до сад дефинисано, треба створити услове да у једном тренутку буде активан само једна апликација, како би се спречило беспотребно пуњење меморије и обезбедио њен нормалан рад. Ово се остварује креирањем фајла у коме се дефинише хијерархија извршавања анимационих видеа. Специјална врста фајла која ово омогућава назива се „*Animator controller*“ којег је потребно креирати у фолдеру где се налазе анимације. Све анимације треба превући у овај прозор и креирати празно стање одабиром „*Create State/Empty*“, и потом га преименовати. У наредном кораку, треба селектовати видео фајлове унутар овог прозора и одабиром опције „*Make Transition*“ извршити повезивање видео блокова. Потом је потребно креирати променљиве типа *bool* и преименовати их да асоцирају на одређену анимацију. Ово се врши кликом на + унутар прозора „*Parameters*“. Затим је потребно за сваку превучену анимацију у одговарајућем „*Image target*“ за контролер изабрати контролер анимације. У наставку рада, потребно је креирати нов фолдер унутар програма Unity и потом унутар њега креирати нову скрипту коју је потребно преименовати и затим покернути. Унутар те скрипте потребно је креирати две променљиве типа *string* назива *imeSl* и *bool* назива *stanje*. У наредном кораку, унутар методе *Update()* неопходно је додати *switch* петљу и променљиве унутар којих се чува назив пронађеног таргета као и информација да ли је пронађен (код испод).

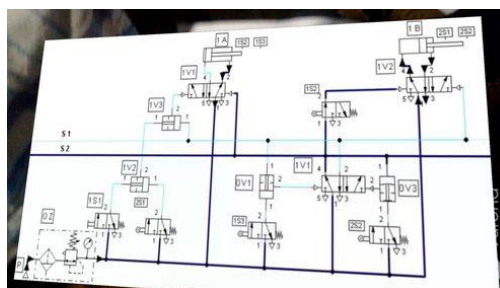
```
void Update() {
    stanje = DefaultTrackableEventHandler.nasao;
    imeSl = DefaultTrackableEventHandler.imeSlike;
    switch (imeSl) {
        case"MODEL11":
GameObject.Find("MODEL11").GetComponent<Animator>
(). SetBool("Var11", stanje); break;}

```

Апликација се извршава на следећи начин. Њеним стартовањем из празног блока прелази се у први повезани блок. У случају да се било који таргет, који је везан за анимацију, нађе у кругу камере, врши се прелазак из прво повезаног блока у блок у коме је анимација модела. Идентификовани модел постаје видљив али се не креће тј. кретање се обавља у случају када је препознат таргет. Након што се заврши анимација враћа се опет у празан блок.

## 7. КРЕИРАЊЕ АПЛИКАЦИЈЕ И ЊЕНО ТЕСТИРАЊЕ

Када је све успешно урађено неопходно је креирати инсталациони *.apk* фајл и потом тестирати апликацију на неком уређају. Пример тестирања апликације на пнеуматској шеми једног задатка из збирке [1] дата је на слици 11.



Слика 11. Тестирање апликације на пнеуматској шеми једног задатка у збирци [1]

## 8. ЗАКЉУЧАК

У овом раду описан је поступак креирања андроид апликација са проширеном реалношћу „*Pneumatika 1*“ и „*Pneumatika 2*“, које се користе као додатак уз Збирку задатака са теоријским основама из пнеуматског управљања. Након тестирања апликација, закључено је да у потпуности задовољавају намену и да је постављени задатак успешно решен.

## 9. ЛИТЕРАТУРА

- [1] С. Дудић, Д. Шешлија, И. Миленковић, Ј. Шулиц, В. Рељић, Б. Бајчи, *Збирка решених задатака са теоријским основама из пнеуматског управљања*, ФТН, Нови Сад, 2017.
- [2] К. Takoordyal, *Beginning Unity Android Game Development From Beginner to Pro*, Apress, Eau Coulee, 2020.

### Кратка биографија:



**Димитрије Милићевић** рођен је у Новом Саду 1996. год. Дипломирао је 2019. год. на Факултету техничких наука, смер Мехатроника, роботика и аутоматизација, на којем је исте године уписао мастер студије.



**Слободан Дудић** рођен је у Ваљеву 1970. год. Докторирао је 2012. године. Од 2017. године ради као ванредни професор на Факултету техничких наука у Новом Саду.