

**RAZVOJ AR APLIKACIJE ZA POSTAVLJANJE DIGITALNOG NAMEŠTAJA U FIZIČKO OKRUŽENJE KORISNIKA****DEVELOPMENT OF AN AR APPLICATION FOR INTEGRATING DIGITAL FURNITURE INTO USER'S PHYSICAL SURROUNDINGS**Luka Vukanović, Stefan Đurđević, *Fakultet tehničkih nauka, Novi Sad***Oblast – GRAFIČKO INŽENJERSTVO I DIZAJN**

**Kratak sadržaj** – Predstavljen je proces razvoja mobilne AR aplikacije, korišćenjem programa Unity Engine, namenjene za postavljanje digitalnog nameštaja u fizičko okruženje korisnika. Takođe obuhvata kreiranje korisničkog interfejsa, trodimenzionalnih modela i uspostavljanje online baze podataka i skladišta radi osiguranja funkcionalnosti aplikacije u realnom vremenu.

**Ključne reči:** Proširena stvarnost, Unity Engine, korisničko okruženje, 3D modelovanje, backend tehnologije

**Abstract** – This paper presents the development process of a mobile AR application using Unity Engine, designed for placing digital furniture in user's physical surroundings. It also encompasses the creation of the user interface, three-dimensional models, and the establishment of an online database and storage to ensure real-time application functionality.

**Keywords:** Augmented Reality, Unity Engine, User Interface, 3D modelling, backend technologies

**1. UVOD**

Augmented Reality (AR), ili proširena stvarnost, je tehnologija koja je transformisala naše svakodnevne aktivnosti. Omogućila je korisnicima interaktivnu zabavu, poboljšala iskustvo kupovine putem virtuelnog probanja proizvoda i postala neizostavan alat u obrazovanju i obuci. Kroz AR aplikacije, korisnici mogu da dožive nove načine interakcije sa stvarnim svetom, olakšavajući im uobičajene zadatke i pružajući im neverovatna iskustva.

U budućnosti, očekuje se značajan napredak u proširenoj stvarnosti, što će uticati na različite aspekte ljudskih života. Predviđa se da će AR blagovremeno postati neizostavan deo svakodnevnice, integrišući se u pametne uređaje i unapređujući interakcije sa digitalnim i stvarnim svetom.

Fokus istraživanja je upotreba razvojnog okruženja Unity Engine za sam razvoj aplikacije koristeći se njegovim alatima i skriptama pisanim u C# programskom jeziku. Takođe predstavljen je proces dizajna korisničkog okruženja koristeći se programom Adobe XD. Generisanje trodimenzionalnih modela manuelnom metodom izvršeno

**NAPOMENA:**

Ovaj rad proistekao je iz master rada čiji mentor je bio dr Stefan Đurđević, docent.

je upotrebom programa Blender, dok je generisanje trodimenzionalnih modela fotogrametrijskom metodom urađeno putem mobilne aplikacije Polycam. Kreiranje baze podataka i skladišta je omogućeno putem Google Firebase servisa.

Cilj rada jeste kreiranje aplikacije pod nazivom „ARrange“, koja pruža korisnicima intuitivan i zabavan način za planiranje i vizualizaciju enterijera. Koristeći AR tehnologiju „ARrange“ omogućava korisnicima da virtuelno postavljaju nameštaj u okolni prostor putem svojih mobilnih uređaja.

**2. PROŠIRENA STVARNOST (AR)**

Proširena stvarnost, ili *Augmented Reality*, je tehnologija koja omogućava digitalnim informacijama da se preklapaju s fizičkim svetom u realnom vremenu. Razvoj AR predstavlja plod dugog niza inovacija i tehnoloških napredaka koji sežu decenijama unazad [1].

**2.1. Istorija AR tehnologije**

Proširena stvarnost je razvijena 1967. godine od strane naučnika i profesora sa Harvarda, Ivana Suderlanda (*Ivan Sutherland*) koji je, zajedno sa svojim učenikom Bobom Sproulom (*Bob Sproull*), proizveo uređaj pod nazivom „Damoklov Mač“ (*The Sword of Damocles*). Ovaj uređaj je predstavljen u vidu displeja (*display*) okačenog o plafon, dok bi drugi kraj bio lociran na korisnikovoj glavi. Korisnik bi putem tog uređaja vizuelno doživeo računarsku grafiku, što bi im pružalo osećaj alternativne stvarnosti [2].

Proširena stvarnost je zvanično dobila svoje ime 1990. godine od strane Toma Kodela (*Tom Caudell*). Kao zaposlen u Boingovom istražnom centru (*Boeing Computer Services Research*) Tom biva zadužen da osmisli novi sistem koji bi pomogao radnicima prilikom projektovanja konstrukcije aviona. Kodel i njegov kolega Dejvid Mizel (*David Mizell*) su predložili uređaj koji bi radnici nosili tokom projektovanja i sastavljanja aviona. Taj uređaj bi projektovao poziciju kablova kroz naočare i prikazivao ih na višenamenskim tablama. Po uspešnoj implementaciji Kodel i njegov kolega zajedno imenuju ovu tehnologiju, *Augmented Reality*, to jest proširena stvarnost [3].

**2.2. AR tehnologija danas i njene platforme**

Nakon više od pola veka od začeća ove tehnologije, AR je doživela komercijalni uspeh pa samim tim i komercijalnu dostupnost. Implementacijom AR tehnologije na mobilne

uređaje doprinelo se njenoj masivnoj popularizaciji, što je rezultovalo velikim investiranjem od strane svetskih tehnoloških kompanija poput kompanija Apple, Google i Microsoft. Najveći korak ka modernizaciji i svakodnevnoj upotrebljivosti proširene stvarnosti je pružio Google 2013. godine, razvojem uređaja *Google Glass* [4].

### 3. RAZVOJ AR APLIKACIJA

AR integriše digitalne informacije kao što su grafika, zvuk, lokacija, i ostale senzorske karakteristike u realan svet korisnika - to se postiže korišćenjem uređaja poput pametnih telefona, tableta ili AR naočara, koji koriste senzore poput kamere, ili akcelerometara, da bi detektovali korisnikovo okruženje.

#### 3.1. Setovi alata za razvoj AR aplikacija

Kako bi kompanije koje su zastupljene na tržištu proširene stvarnosti pomogle, i podstakle, programere u kreiranju određenih aplikacija za njihove platforme ili operativne sisteme, one im pružaju SDK-ove (*Software Development Kit*). SDK predstavlja set alata za razvoj softvera, kao i skup alata, biblioteka, dokumentacije i uzoraka koda. SDK-ovi obično obuhvataju sve što je potrebno programeru da napravi softver za određenu platformu, uključujući API-je (*Application Programming Interface*) - interfejsse za programiranje aplikacija radi pristupa platformskim funkcijama, razvojna okruženja, alate za dibagovanje (*debugging*) i često uputstva ili dokumentaciju koja pomaže programerima da efikasno koriste upravo taj SDK [5].

Najpopularniji setovi alata su:

- *Arkit* (iOS),
- *ARCore* (Android),
- *Vuforia*,
- *Easy AR i*
- *Snapchat Lens Studio*

### 4. RAZVOJ APLIKACIJE „ARRANGE“

Razvoj aplikacije obuhvata korake od pripreme okruženja za razvoj u programu Unity Engine, implementacije ključnih funkcionalnosti, sve do dizajniranja korisničkog interfejsa, pripreme trodimenzionalnih modela i povezivanja same aplikacije sa bazom podataka koja joj omogućava rad u realnom vremenu.

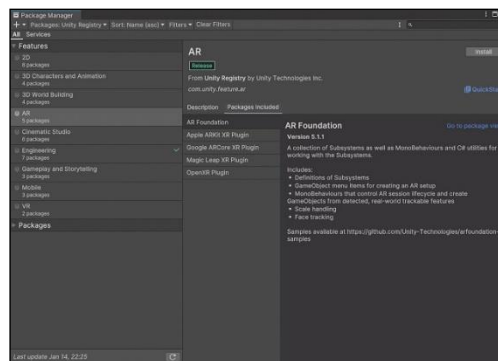
#### 4.1. Priprema AR okruženja

Nakon kreiranja novog projekta, neophodno je izvršiti implementaciju određenih paketa kako bi projekat omogućio razvoj aplikacije proširene stvarnosti.

Otvaranjem prozora *Package Manager* (Slika 1.) dobija se uvid u različite pakete i podgrupe paketa koji su dostupni u ovom razvojnom okruženju.

Paketi koji se uvoze u program i instaliraju su:

- *AR Foundation*,
- *Google ARCore XR Plugin*,
- *Apple ARKit XR Plugin*,
- *XR Interaction Toolkit*,
- *Universal Render Pipeline i*
- *TextMeshPro*



Slika 1. Izgled Package Manager prozora

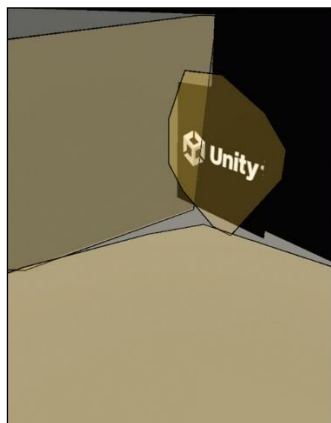
Po integraciji pomenutih paketa, konfigurise se projekat definisanjem određenih vrednosti i specifikacija koje namenski uređuju rad programa na mobilnim uređajima. Što se tiče Android platforme, u prozoru *Player Settings*, neophodno je onemogućiti *Multithreaded Rendering* ukoliko se targetiraju i stariji uređaji. Takođe, onemogućiti *Autographics API*, a u grupi *Rendering* odabrati *OpenGL ES3* opciju. Kao *Scripting Backend* opciju postaviti *IL2CPP*, omogućiti *ARM64* i podesiti minimalnu verziju Android operativnog sistema za koji je aplikacija predviđena. Što se tiče Apple uređaja, jedino što se zahteva jeste definisanje minimalne verzije operativnog sistema, to jest iOS-a. Po kompletiranju prethodnih koraka, vrši se aktivacija *Google ARCore* i *Apple ARKit* paketa u kartici *XR Plug-in Management*.

Nakon konfiguracije bitnih parametara aplikacije, vrši se integracija *Universal Render Pipeline* (URP) paketa. URP predstavlja sistem za renderovanje i generisanje grafike u igrama i aplikacijama. Kreiran je na taj način da bude kompatibilan i sa računarima i mobilnim uređajima. Osim fleksibilnosti, nudi odlične grafičke rezultate obezbeđujući optimalne performanse na samom uređaju [6].

Po optimizaciji aplikacije i primenjivanja URP sistema renderovanja, kreira se scena (*Scene*) i time se stvara radni prostor u kojem se definišu i primenjuju neophodni elementi za rad aplikacije, poput skripti ili modela.

Osnovno svojstvo AR aplikacija jeste upravo to da se koriste fizičke kamere, to jest kamere korisnikovog uređaja. Kako bi se to omogućilo kreiraju se AR sesija (*AR Session*) i XR izvor (*XR Origin*), čije su funkcije definisanje programu da je u pitanju AR aplikacija, posmatranje realnog sveta kroz objektiv kamere XR izvora kao i detektovanje vertikalnih i horizontalnih površina. Da bi pomenuto detektovanje ravni bilo moguće, dodaje se komponenta *AR Plane Manager* na objekat *XR Origin* (XR izvor). Po definisanju izgleda detektovanih ravni putem objekta *AR Plane* i njegovog materijala, i nakon njegovog referenciranja u *AR Plane Manager* komponenti, XR izvor, to jest kamera korisnikovog uređaja, vizuelno prikazuje detektovane horizontalne i vertikalne ravni generisanjem mreže poligona po površini prepoznatih fizičkih ravni. Izgled pomenute mreže je definisan pomenutim materijalom na *AR Plane* objektu što je u ovom slučaju prozirno žuti

materijal sa naglašenim ivicama crne boje kao što je prikazano na slici 2.



Slika 2. Prepoznavanje vertikalnih i horizontalnih površina

#### 4.2. Implementacija osnovnih funkcionalnosti manipulacije nameštaja

Osnovne funkcionalnosti manipulacije nameštaja, to jest objekata na sceni, koje je potrebno da ovakav tip aplikacije poseduje su:

- Instanciranje objekata na scenu,
- pomeranje instanciranih objekata i
- rotacija instanciranih objekata

Instanciranje objekata predstavlja kreiranje instanci, to jest kopija predefinisanih objekata, i ono se vrši putem skripti pisanih u C# programskom jeziku, metodom *Instantiate()*, dok se lokacija instanciranja definiše putem *Raycast* tehnike. *Raycast* tehnika predstavlja tehniku „pucanja“ usmerenih zrakova u definisanom pravcu, pri čemu se kao rezultat dobijaju informacije o pogođenom objektu, što su u ovom slučaju detektovane površine ili prethodno postavljen nameštaj na sceni [7]. Komponenta *AR Raycast Manager* se postavlja na *XR Origin* objekat kako bi izvor zraka bila sama kamera korisnikovog uređaja. Definiše se skriptom da se vrši konstantno zračenje u pravcu sredine ekrana uređaja i da se na samoj tački pogotka detektovanih horizontalnih površina vrši prikaz identifikatora (Slika 3.), to jest same lokacije gde će se stvoriti izabrani nameštaj.



Slika 3. Izgled identifikatora lokacije za instanciranje nameštaja

Pomeranje pomenutih objekata funkcioniše na sličan način, pri dodiru ekrana „ispaljuju“ se zraci u pravcu tačke dodira, ukoliko zrak pogađa i prepozna prethodno instanciran objekat, isti taj objekat se usidrava za dodirnu tačku sve do samog momenta otpuštanja ekrana. Što se

tiče rotacije, vrši se pri dodiru dva prsta na osnovu njihove relativne pozicije. Rotacija se vrši isključivo oko vertikalne ose objekta.

#### 4.3. Dizajn i implementacija korisničkog interfejsa

*User Interface*, UI ili korisnički interfejs, i sam dizajn istog, predstavlja proces vizuelnog kreiranja naličja i izgleda programa, softvera, ili generalno uređaja koji se služe ekranom kao sredstvom za interakciju između korisnika i uređaja [8]. Dizajn interfejsa se vrši sveobuhvatnim setom alata koje nudi program Adobe XD. Vizuelne celine koje čine ovu aplikaciju su:

- ekran dobrodošlice i učitavanja aplikacije,
- ekran početne scene aplikacije,
- ekran menija za izbor kategorije proizvoda,
- ekran menija sa potkategorijama proizvoda i prikaz proizvoda tog tipa,
- ekran pojedinačnog proizvoda,
- ekran nakon postavljanja proizvoda u fizičko okruženje, sa funkcionalnostima manipulacije istog i
- ekran menija sa podešavanjima

Po pojedinačnom izvozu grafičkih elemenata koji čine ove celine iz programa Adobe XD, uvoz istih u Unity Engine se vrši jednostavnim prevlačenjem u projektni prozor aplikacije i definisanjem njihove namene u programu, to jest podešavanje teksture da bude tipa *Sprite (2D and UI)*. Proces integracije ovih elemenata u programu Unity Engine se vrši unutar *Canvas* objekta koji obuhvata ceo korisnički interfejs aplikacije. U njemu se definišu svi meniji, dugmad, slike i tekstualni elementi kao i njihove funkcije i metode koje izvršavaju.

Unity Engine podržava i uvoz eksternih, nesistemskih, fontova, koji se pre korišćenja obrađuju putem paketa *TextMeshPro* koji generiše *font atlas* i time omogućava implementaciju novih fontova.

#### 4.4. Kreiranje i priprema 3D modela

Kreiranje i dizajn trodimenzionalnih modela u ovom istraživačkom radu se izvršava na dva načina:

- manuelnim modelovanjem putem programa Blender
- fotogrametrijskom metodom putem programa Polycam

U programu Blender, kao primer, se vrši dizajniranje himalajske slane lampe putem osnovnih primitiva i modifikatora.



Slika 4. Himalajska slana lampa, rezultat manualnog 3D modelovanja

Kao baza se uzima jednostavan 3D objekat kocke na kojoj se primenjuju modifikatori *Subdivision Surface modifier* s ciljem generisanja kompleksnije mreže poligona, kao i modifikator *Displace* radi obrade same mreže čime se dobija nepravilan, prirodan oblik 3D modela. Primenjuje

se funkcija *Decimate Geometry* radi optimizacije mreže poligona. Generiše se postolje lampe kao i materijal na osnovu slike teksture himalajske slane lampe i za rezultat se dobija 3D model sa slike 4. Prilikom izvoza 3D modela iz programa Blender, definiše se to da se osim modela, zajedno s njim izvezu i primenjene teksture i materijali.

Što se tiče fotogrametrijske metode kreiranja 3D modela, fotografiše se željeni stvarni objekat, u slučaju ovog rada to je sto za kafu, iz različitih uglova. Zabeležene fotografije se obrađuju u programu Polycam i kao rezultat se dobija 3D model velikih deformacija koje ne ispunjavaju uslove reprezentativnog 3D modela. Korisnici ove aplikacije očekuju verodostojnu reprodukciju digitalnog modela prilikom izbora i pregleda artikla pri kupovini istog.

#### 4.5. Integracija Firebase baze podataka i skladišta

Kako bi se izbeglo veliko zauzimanje memorije od strane aplikacije, učitavaju se samo podaci i modeli koje korisnik odabere. „ARrange“ koristi *Firestore Database* za osnovne podatke o proizvodu i *Firestore Storage* za slike i modele nameštaja. Pre korišćenja ovih usluga, potrebno je integrisati Firebase u Unity projekat i kreirati bazu i skladište na Firebase platformi, te uvezati projekat sa Firebase servisima putem Firebase konzole.

Nakon uvezivanja, generišu se i baza i skladište, a podaci se organizuju po sledećoj hijerarhiji:

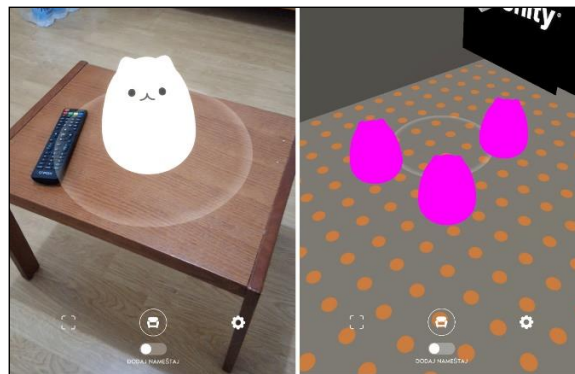
1. Svi proizvodi
2. Prostorija
3. Vrsta proizvoda
4. Pojedinačni proizvod
5. Podaci o proizvodu

Učitavanje podataka, modela i fotografija proizvoda vrši se u realnom vremenu pri izboru proizvoda, koristeći predefinisane metode Firebase paketa.

## 5. REZULTAT

U ovom istraživačkom radu prikazan je razvoj mobilne aplikacije za proširenu stvarnost pod imenom "ARrange", koja omogućava postavljanje digitalnog nameštaja u fizički prostor korisnika. Aplikacija je razvijena koristeći Unity Engine, dok su 3D modeli napravljeni pomoću programa Blender i Polycam. Za *backend* tehnologije korišćen je Google Firebase servis.

Rezultat rada je funkcionalna AR aplikacija koja omogućava korisnicima da virtuelno postavljaju i manipulišu nameštajem u stvarnom prostoru koristeći mobilne uređaje, kao i radni projekat koji sadrži sve neophodne pakete i podešavanja za razvoj aplikacije ovog tipa (Slika 5.). Glavni koraci u razvoju aplikacije uključuju pripremu razvojnog okruženja, implementaciju funkcionalnosti za postavljanje i pomeranje nameštaja, dizajn korisničkog interfejsa, kreiranje 3D modela i povezivanje sa Firebase bazom podataka i skladištem. Aplikacija "ARrange" nudi intuitivan i zabavan način za planiranje i vizualizaciju enterijera. Korisnici mogu lako da vide kako bi određeni komad nameštaja izgledao u njihovom prostoru, što im pomaže pri odlučivanju o kupovini.



Slika 5. Prikaz rada aplikacije na uređaju i u projektu

## 6. ZAKLJUČAK

Može se zaključiti da je razvoj AR aplikacije kompleksan i multidisciplinarni proces koji zahteva kombinaciju programerske stručnosti, kreativnog dizajna i tehničkog znanja o 3D modelovanju i *backend* tehnologijama. Detaljnim prikazom svih koraka, od konceptualizacije do implementacije i optimizacije, rad nudi uvid, uputstva i praktične savete za razvoj AR aplikacije. Kroz ovakav pristup, moguće je postići balans između tehničke izvedbe i korisničkog iskustva, što je ključno za uspeh modernih aplikacija.

## 7. LITERATURA

- [1] P. Ciproso, I.A.C. Giglioli, M.A. Raya, and G. Riva, "The Past, Present, and Future of Virtual and Augmented Reality Research: A Network and Cluster Analysis of the Literature," *Front. Psychol.*, Vol. 9, pp. 4, November 2018.
- [2] <https://colocationamerica.com/blog/history-of-augmented-reality> (pristupljeno u februaru 2024.)
- [3] <https://dummies.com/article/technology/programming-web-design/general-programming-web-design/the-history-of-virtual-and-augmented-reality-256104> (pristupljeno u februaru 2024.)
- [4] <https://techtaraget.com/iotagenda/definition/Google-Glass> (pristupljeno u februaru 2024.)
- [5] <https://www.ibm.com/blog/sdk-vs-api> (pristupljeno u februaru 2024.)
- [6] <https://docs.unity3d.com/Packages/com.unity.render-pipelines.universal@17.0/manual/index.html> (pristupljeno u maju 2024.)
- [7] <https://docs.unity3d.com/Packages/com.unity.xr.arfoundation@4.0/manual/raycast-manager.html> (pristupljeno u maju 2024.)
- [8] <https://www.browserstack.com/guide/what-is-user-interface> (pristupljeno u februaru 2024.)

### Kratka biografija:



**Luka Vukanić** rođen je u Senti 1997. god. Master rad na Fakultetu tehničkih nauka iz oblasti Grafičkog inženjerstva i dizajna odbranio je u junu 2024. godine.

*kontakt:*

**cimbur13@gmail.com**