

APLICAȚIE MOBILE PENTRU RESTAURANTE - FOOD MENU

Autor: Cotei Ovidiu-Alin

Coordonatori: Lect. Univ. Dr. Muntean Maria, Asist. Univ. Drd. Boca Liana

Universitatea "1 Decembrie 1918" din Alba Iulia, Facultatea de Stiințe Exacte și Inginerești

Abstract

Food Menu is a restaurant serving menu application. The Food Menu application provides customer can view the menu in 3D before ordering. The application is based on AR (Augmented Reality) technology, offering the user`s customer all the possible menus in an Augmented Reality on a mobile device.

1. Introducere

Dacă în urmă cu aproximativ 20 de ani omul folosea telefonul mobil doar pentru apeluri și mesaje, odată cu progresul tehnologic acesta a ajuns să combine caracteristici diferite și tot mai multe funcții, astfel încât s-a ajuns la telefoane cu o mare putere de procesare și de stocare. Astfel de dispozitive permit în prezent utilizatorilor să păstreze melodii, imagini, amintiri, informații personale, corespondență, un program personal (al întâlnirilor – de exemplu), date de sănătate și financiare într-un singur loc.

Industria telefoanelor s-a dezvoltat fantastic și a avut un impact mare pe piață, generând concurență între marile companii. Odată cu evoluția incredibilă a tehnologiei, dezvoltarea aplicațiilor mobile a "explodat". Aplicațiile mobile sunt în prezent în mijlocul existenței umane, iar viața oamenilor devine mult mai confortabilă și mai simplă datorită acestora. Utilizatorul are astăzi la dispoziție o aplicație pentru orice, de exemplu pentru contorizarea pașilor, învățare de limbi străine, limbaje de programare, comandare de produse, etc.

Bineînțeles, aplicații mobile au fost dezvoltate și pentru sectorul alimentar și al băuturilor, firmele intuind repede importanța acestora în relația cu clienții și utilitatea lor în prezentarea cât mai multor informații relevante pentru aceștia.

O altă tehnologie care, în ultima perioadă, a depășit toate așteptările este tehnologia Augmented Reality (Realitatea Augmentată). Aceasta are o tendință crescătoare în rândul companiilor IT, al aplicațiilor mobile și al aplicațiilor de afaceri. Realitatea Augmentată este o versiune îmbunătățită a lumii fizice prin folosirea mediilor vizuale și audio. Cu alte cuvinte, realitatea augmentată integrează informațiile digitale cu mediul utilizatorului în timp real. [1]

2. Realitatea augmentată

Realitatea augmentată utilizează mediul natural existent și suprapune informații virtuale deasupra acestuia. Principalul beneficiu al utilizării realității augmentate constă în faptul că utilizatorii

intră în contact cu o lume naturală nouă și îmbunătățită, în care informațiile din lumea virtuală sunt folosite ca instrument de asistență în viața reală. Informațiile furnizate de aplicațiile de realitate augmentate sunt actualizate și foarte relevante pentru ceea ce sunt utilizate. [1]

Așa cum se va observa pe parcursul acestei lucrări, Realitatea Augmentată sta la baza aplicației de față, integrându-se în subiectul și conceptul acesteia.

3. Framework pentru dezvoltarea aplicațiilor. Stadiul actual

Internetul pune la dispoziție o gamă variată de framework-uri pentru dezvoltarea aplicațiilor pentru Mobile, fiecare având avantajele și dezavantajele lor. Printre ele amintim Android Studio (<https://developer.android.com/studio>) dezvoltat de compania Google și JetBrains. Acest framework este oferit gratuit și pune la dispoziție întregul cod sursă Java.

Un alt framework cu ajutorul căruia se pot dezvolta aplicații mobile cât și jocuri mobile este Unity, dezvoltat de compania Unity Technologies. Acesta pune la dispoziție întregul cod sursă C#.

Pentru dezvoltarea aplicațiilor se pot menționa și alte soluții cross-platform cum ar fi: Xamarin, ReactNative, NativeScript, Ionix și Appcelerator.

4. Tehnologii folosite în dezvoltarea aplicației

Având în vedere aplicația care s-a dorit a fi dezvoltată, s-au ales pentru implementare framework-ul Unity, pluginul Vuforia și limbajul de programare C#.

Motivele pentru care s-au ales aceste tehnologii sunt punctate în subcapitolele următoare.

4.1 Descrierea softului utilizat în dezvoltarea aplicației

Mediul de dezvoltare al aplicației este Unity (www.unity3d.com), care este un motor de jocuri (sau motor graphic) dezvoltat de compania Unity Technologies și este oferit atât în variantă gratuită cât și cu plată. Versiunea utilizată este Unity 2018.3, deși a fost lansată deja versiunea Unity 2019.1 beta.

Unity dispune de o interfață grafică intuitivă, prietenoasă, destul de ușor de folosit și personalizabilă. Interfața grafică oferă utilizatorului libertatea de a-și alege după propria dorință spațiul de lucru. Selecția fereastră se face ușor, cu click stânga și apoi poate fi poziționată în oricare zonă. În figura de mai jos este prezentată o reprezentare a spațiului de lucru *default*, care se încarcă la crearea unui nou proiect 3D. [2]

Acest mediu de dezvoltare dispune de două moduri de creare a proiectelor: 2D sau 3D. Pe lângă aceste moduri, softul mai dispune de exportul proiectelor pentru uzul pe mai multe platforme, fiind necesară modificarea codului sursă al proiectului pentru a corespunde controlerului de joc specific fiecărei platforme. [2]

Platformele pentru care se pot dezvolta jocuri sau aplicații folosind Unity sunt: PC, MAC, Linux Standalone, WebPlayer, iOS, Android, Windows Store, Windows Phone, Xbox One, PS4, PS Vita, Steam OS, Wii U, Nintendo 3D S, Oculus Rift, Playstation VR, Gear VR, Android TV, Facebook Gameroom și multe altele.

4.2 Descrierea plugin-ului utilizat în dezvoltarea aplicației

Vuforia Engine este cea mai răspândită platformă pentru dezvoltarea AR, cu suport pentru telefoane și tablete. Dezvoltatorii pot adăuga cu ușurință funcționalitate avansată a viziunii pe computer la aplicațiile Android, iOS și UWP, pentru a crea experiențe AR care interacționează realist cu obiectele și mediul. [3]

Vuforia Developer Library pune la dispoziția dezvoltatorilor o gamă largă de plugin-uri. În cazul acestei aplicații unul dintre pluginurile folosite este *“Image Targets”*.

Image Targets reprezintă imagini pe care Vuforia Engine le poate detecta și urmări. Spre deosebire de marcatorii fiduciar tradiționali, codurile matricei de date și codurile QR, obiectivele imagine nu au nevoie de regiuni sau coduri speciale alb-negru pentru a fi recunoscute. Motorul detectează și urmărește caracteristicile care se găsesc în mod natural în imagine, prin compararea acestor caracteristici naturale cu o bază de date cunoscută a resurselor țintă. Odată ce obiectivul imagine este detectat, Vuforia Engine va urmări imaginea atât timp cât este cel puțin parțial în câmpul vizual al camerei. [3]

4.3 Limbajul de programare C#

Lansat publicului în iunie 2000 și oficial în primăvara anului 2002, C# este un limbaj de programare care combină facilități testate de-a lungul timpului cu inovații de ultim moment. Creatorii acestui limbaj au fost o echipă de la firma Microsoft condusă de Anders Hejlsberg. Deși limbajul este creat de Microsoft, acesta nu este destinat doar platformelor Microsoft. Compilatoarele C# există și pentru alte sisteme precum Linux sau Macintosh. Creat ca instrument de dezvoltare pentru arhitectura .NET, limbajul oferă o modalitate facilă și eficientă de a scrie programe pentru sistemul Windows, internet, componente software etc. [4]

Deși Java a rezolvat cu succes problema portabilității, există unele aspecte care îi lipsesc. Una dintre acestea este interoperabilitatea limbajelor diferite, sau programarea în limbaj mixt (posibilitatea codului scris într-un limbaj de a lucra în mod natural cu codul scris în alt limbaj). Interoperabilitatea limbajelor diferite este esențială la realizarea sistemelor software de dimensiuni mari.

Ca parte a ansamblului strategiei .NET, dezvoltată de Microsoft, la finele anilor '90 a fost creat limbajul C#. C# este direct înrudit cu C, C++ și Java. "Bunicul" limbajului C# este C-ul. De la C, C# moștenește sintaxa, multe din cuvintele cheie și operatorii. De asemenea, C# construiește peste modelul de obiecte definit în C++. Relația dintre C# și Java este mai complicată. Java derivă la rândul său din C și C++. Ca și Java, C# a fost proiectat pentru a produce cod portabil. Limbajul C# nu derivă din Java. Între C# și Java există o relație similară celei dintre "veri", ele derivă din același strămoș, dar deosebindu-se prin multe caracteristici importante. [4]

Limbajul C# conține mai multe facilități inovatoare, dintre care cele mai importante se referă la suportul încorporat pentru componente software. C# dispune de facilități care implementează direct elementele care alcătuiesc componentele software, cum ar fi proprietățile, metodele și evenimentele. Poate cea mai importantă facilitate de care dispune C# este posibilitatea de a lucra într-un mediu cu limbaj mixt. [4]

5. Descrierea aplicației

Food Menu este o aplicație destinată restaurantelor, cărora le pune la dispoziție posibilitatea de a-și prezenta preparatele și băuturile disponibile din meniu, clienților, dintr-o perspectivă mai detaliată și mult mai atractivă. Această perspectivă este mult mai atractivă datorită faptului că aplicația are suport pentru dispozitivele mobile, fiind bazată pe tehnologia Augmented Reality.

Aplicația este capabilă să pună la dispoziția utilizatorului o gamă mare de produse (preparate și băuturi) în 3D care pot fi vizualizate cu ajutorul camerei telefonului într-o lume reală. Pune la dispoziție o vizualizare cu foarte multe unghiuri de rotire astfel încât produsul se poate vizualiza la 360 de grade.

În același timp, aplicația pune la dispoziție și informații despre produs cum ar fi: prețul acestuia, ingredientele folosite pentru prepararea lui și informații nutriționale astfel încât clientul este informat despre cantitatea de calorii, carbohidrați, proteine, grăsimi, precum și alergenii existenți în mâncarea preparată care, conform legislației actuale (intrată în vigoare în anul 2014) este obligatoriu ca fiecare

business din domeniul alimentației să furnizeze lista completă a acestora - conținuți de fiecare aliment sau mâncare.

Aplicația are ca scop crearea unei legături între un restaurant și clienții acestuia prin această perspectivă realistă și atrăgătoare, fiind în același timp și o modalitate de publicitate și vizibilitate în online.

Obiectivul primar al aplicației este de a veni atât în întâmpinarea restaurantelor cât și a clienților cu o soluție actuală, atractivă și ușor de folosit, care să fie avantajoasă atât pentru restaurant cât și pentru client.

Structura aplicației

5.1 Meniul de start

Meniul de start al aplicației pune la dispoziția utilizatorului două butoane care folosesc comenzi separate: *Alegere Meniu* și *Ieșire*.

Butonul “*Alegere Meniu*” duce utilizatorul către meniul principal al aplicației.

Butonul “*Ieșire*” părăsește aplicația.

5.2 Meniul principal

Meniul principal al aplicației vine cu o listă în care se află toate preparatele și băuturile din meniul restaurantului, lista fiind poziționată în partea stângă a ecranului.

Butonul “*Meniu*” din partea dreaptă duce utilizatorul către meniul de start al aplicației.

Butonul “*Informații*” pune la dispoziția utilizatorului informațiile despre produsul pe care dorește să îl comande.

5.3 Tutorial disponibil utilizatorului

Aplicația pune la dispoziția utilizatorului un tutorial în care acesta este informat despre funcționalitățile acesteia, astfel încât să se familiarizeze foarte ușor cu funcționarea meniului precum și tot ceea ce oferă aceasta.

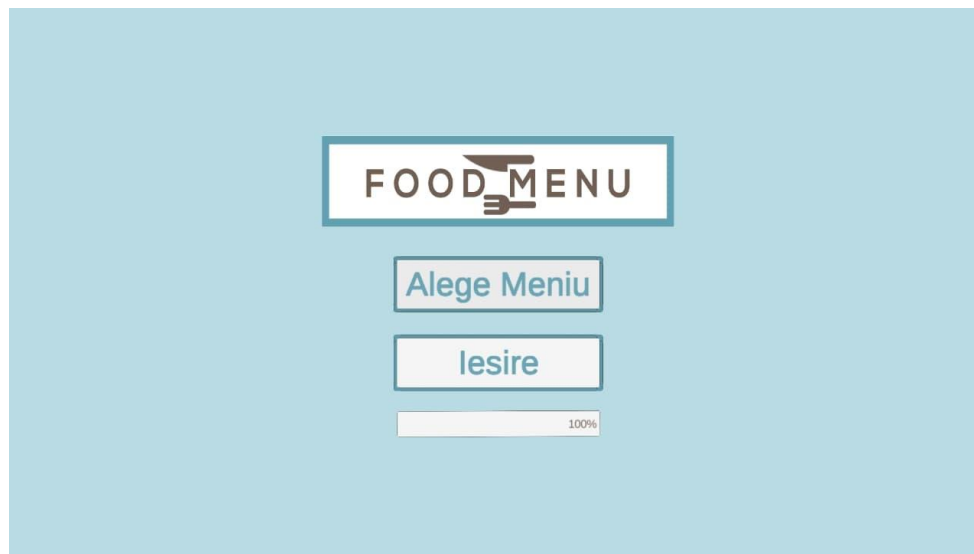


Figura 1. Meniul de start

Pentru funcționalitatea butonului de “*Alegere Meniu*” s-a implementat următorul script:
`public void ChooseMenu(int sceneIndex)`

```

{
    soundManager.SendMessage("PlayClick");
    StartCoroutine(LoadAsynchronously(sceneIndex));
}

```

Pentru funcționalitatea butonului de “Ieșire” a fost implementat următorul script:

```

public void ExitApp()
{
    soundManager.SendMessage("PlayClick");
    Application.Quit();
}

```



Figura 2. Meniul principal

Pentru funcționalitatea listei de preparate a fost implementat următorul script:

```

using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;

// -----
// Class: AutoResizeGridElement
// Desc: Contine toate metodele pentru funcționalitatea
//       sistemului de Auto Resize asupra elementelor
//       aflate in parintele Grid Layout.
// -----
public class AutoResizeGridElement : MonoBehaviour {

    private Canvas canvas;
    private GridLayoutGroup gridLayout;

    // -----
    // Method: Start
    // Desc: Folosita pentru initializarea referintelor atunci
    //       cand aplicatia porneste.
    // -----
    private void Start()
    {
        canvas = this.GetComponentInParent<Canvas>(); // Accesam (facem referinta)
        componentei Canvas
    }
}

```

```

        GridLayout = this.GetComponent<GridLayoutGroup>(); // Accesam (facem referinta)
componentei GridLayoutGroup
    }

    // -----
    // Method: Update
    // Desc: Folosita pentru actualizarea frecventa la fiecare
    //       cadru pe secunda la rularea aplicatiei.
    // -----
    private void Update()
    {
        AutoResize(); // Apelam functia AutoResie pentru actualizare frecventa.
    }

    // -----
    // Method: AutoResize
    // Desc: Folosita pentru a calcula size-ul canvas-ului.
    // -----
    private void AutoResize()
    {
        float cellX = canvas.pixelRect.width * 0.1916f;
        float cellY = canvas.pixelRect.height * 0.82f / 3;
        GridLayout.cellSize = new Vector2(cellX, cellY);
    }
}

```



Figura 3. Vizualizare 3D

Figura de mai sus prezintă vizualizarea 3D a unui meniu selectat. Odată selectat meniul dorit, în partea Jos – Dreapta a ecranului apare un buton de informații, astfel încât dacă acesta este selectat pot fi vizualizate informații despre preparat / produs. Acesta poate conține atât informațiile nutriționale ale preparatului / produsului cât și informațiile obligatorii pe care firmele de profil au obligația de a le aduce la cunoștința clienților.

Pentru toate acestea este prezentat mai jos o mică parte din scriptul implementat:

```

public void SelectModel(int index)
{
    soundManager.SendMessage("PlayClick");
    modelsManager.SendMessage("SetModel", index);
    // Setam toate imaginile sa fie selectate
}

```

```

DeselectAll();
sliderImages[index].sprite = selectedSprites[index];
if(index != 6)
{
    informationCards.sprite = cardsSprites[index];
    textCard.sprite = textSprites[index];
    ToogleInformationIcon(true);
    StartCoroutine(ShowHideInformationUserInfo());
}
else
{
    ToogleInformationIcon(false);
}
}
public void ShowInformation()
{
    informationPanel.SetActive(true);
    soundManager.SendMessage("PlayClick"); // Apeleaza functia PlayClick din scriptul
SoundManager.
}

```



Figura 4. Instrucțiuni



Figura 5. Informații despre produs / preparat

6. Direcții de extindere

Prezenta aplicație a fost gândită ca o soluție practică și utilă și prietenoasă pentru restaurante și clienți.

Se dorește ca aplicația să fie dezvoltată astfel încât să conțină o bază de date, iar prin intermediul unui cont securizat orice restaurant să își creeze un profil și să își poată adăuga / edita / șterge propriul meniu compus din toate preparatele / produsele / băuturile disponibile și la care să adauge orice informații dorite sau obligatorii.

Se dorește implementarea unei soluții prin care să fie afișată o listă cu toate restaurantele din zona în care se află utilizatorul, iar acesta să poată vizualiza profilul restaurantelor.

De asemenea, se dorește implementare unui sistem de rating, astfel încât utilizatorul să poată nota un anumit restaurant și astfel se va crea o listă cu cele mai bune și vizitate restaurante ceea ce va duce la creșterea de vizitatori pentru un anume restaurant.

CONCLUZII

Aplicația de față a fost creată pentru a crea o legătură între restaurant și clienții lor și se dorește ca odată cu dezvoltarea acesteia să devină o soluție dedicată și utilă acestora.

În urma implementării și testării componentelor descrise în subcapitolele anterioare am observat că implementarea unor componente într-un limbaj de nivel înalt cum ar fi C# se integrează perfect cu mai puține dificultăți de implementare.

Una dintre dificultățile implementării poate fi cea a pluginului Vuforia deoarece acesta nu se adaptează mereu cu Unity din cauza schimbărilor de versiuni.

BIBLIOGRAFIE

- [1] ***, „What is Augmented Reality, explanations and various types,” 2018. [Interactiv]. Available: https://www.academia.edu/37588379/What_is_Augmented_Reality_explanations_and_various_types.pdf.
- [2] ***, „Unity User Manual (2018.3),” [Interactiv]. Available: <https://docs.unity3d.com/Manual/index.html>.
- [3] ***, „Getting Started with Vuforia Engine in Unity,” [Interactiv]. Available: <https://library.vuforia.com/articles/Training/getting-started-with-vuforia-in-unity.html>.
- [4] C. Galeș, „Programare C#,” [Interactiv]. Available: <https://www.math.uaic.ro/~cgales/csharp/Curs1.pdf>.