

AZƏRBAYCAN RESPUBLİKASI TƏHSİL NAZİRLİYİ
AZƏRBAYCAN TEXNİKİ UNİVERSİTETİ
İNFORMASIYA VƏ TELEKOMMUNİKASIYA TEXNOLOGİYALARI
FAKÜLTƏSİ

Əlyazması hüququnda

İxtisasın şifrə və adı: M060509-Kompüter elmləri

İxtisaslaşmanın adı: Optimal idarəetmənin riyazi-iqtisadi üsulları

Mühəndis riyaziyyatı və süni intellekt kafedrasının magistrantı

Muradov Anar Səyavuş oğlu

(Magistrantın S.A.A)

magistr dərəcəsi almaq üçün

“Restoranlarda müştərilərə online xidmət üçün mobil tətbiqin yaradılması”

mövzusunda

DİSSERTASIYA İŞİ

Kafedra müdiri:

t.e.d.,prof Ağayev N.B.
(elmi dərəcəsi,elmi adı, S.A.A)

Elmi rəhbər:

r.f.d, dosent Əmirova R.A
(elmi dərəcəsi,elmi adı, S.A.A)

MÜNDƏRİCAT

Giriş	3
--------------------	---

FƏSİL I

İnteqrasiya edilmiş mobil tətbiq sisteminin nəzəri aspektləri	5
--	---

1.1. Mobil tətbiqlərin inkişafı və effektivliyinin analizi	5
---	---

1.2. Mobil tətbiqlərin texniki aspektləri	7
--	---

1.3. Mobil tətbiqlərin günümüzdəki rolu	11
--	----

FƏSİL II

Mobil tətbiqinin dizaynı və mobil proqramın hazırlanması	18
---	----

2.1. Restoranlar mobil tətbiqinin interfeys və qrafik dizaynı	18
--	----

2.2. Mobil tətbiq üçün proqramlaşdırma dili və texnologiyaların seçimi	28
---	----

2.3. Mobil tətbiq üçün Firebase məlumat bazasının hazırlanması	38
---	----

FƏSİL III

Mobil tətbiqin analizi və effektivliyi	45
---	----

3.1. Restoranlar mobil tətbiqinin əsas funksiyalarının təhlili	45
---	----

3.2. Sifarişlərin sərbətlə əlavə olunması və qiymətləndirilməsi	56
--	----

3.3. Mobil tətbiqlərin restoran iş sahəsinə təsiri	63
---	----

Nəticə	67
---------------------	----

Ədəbiyyat	68
------------------------	----

GİRİŞ

Mövzunun aktuallığı: Müasir dövrdə insan mobil cihazlarla sürətli məlumat əldə etməyə üstünlük verir. Hər il saytların mobil versiyalarının istifadəçilərinin sayı artır. Azərbaycanda əhalinin 98%-nin mobil telefonu var və rəqəmsal əsrdə bir çox proqram məhsulları, mobil telefonlar üçün hazırlanmışdır. Elektron tətbiqlərin ən son qabaqcıl modellərinin əksəriyyəti funksionallıq və performans baxımından böyük mobil telefonlar və kompüterlərdən geri qalmır.

İndi dünyada demək olar ki, mobil tətbiqlərdən istifadə etməyən insanın sayı azdır. Mobil tətbiqlərin populyarlığının səbəbləri bu tətbiqlərin funksionallığı, bütün ünsiyyət üsullarına çıxışı olmasıdır. Mobil texnologiyanın artması səbəbindən əhali bütün tətbiqlərdə məlumat axınlarını izləməkdə müəyyən çətinliklər yaşasalar da, restoran üçün mobil tətbiqlərin yaradılması çox aktualdır və günün tələblərinə uyğundur.

İşin məqsədi: Dissertasiya işinin məqsədi Restoran üçün mobil tətbiqin yaradılması üçün proqram təqdim etməkdir. Bu məqsədlə aşağıdakı şəkildə tətqiqatlar aparılmışdır.

- 1) İnternet marketinq modelinin formalaşması və idarəedilmə olunması problemini yeniləmək;
- 2) Mobil tətbiqin effektivliyinin qiymətləndirilməsi meyarlarını müəyyən etmək;
- 3) İnternet üzərindən marketinq üçün mövcud olan qaydalarla təyin olunmuş inteqrasiya edilmiş İnternet marketinq modelinin üstünlüyünün qiymətləndirmək üsulunu təkmilləşdirmək;
- 4) Mobil proqramların təqdim olunması texnologiyalarını təhlil etmək;
- 5) Proqram paketi hazırlamaq.

Elmi yenilik: Elmi yenilik mobil idarəetmə sistemində inteqrasiya olunmuş İnternet marketinq modelinin formalaşdırılması üçün metodoloji yanaşmalar işlənilib hazırlanmasıdır. Verilmiş obyektin mobil mühidə restoran və kafe fəaliyyətini təşviq etmək və təqdim etməkdir. Təqdim olunan işdə restoran və kafelərdə funksional prosesin işləmə mexanizmi təqdim olunmuşdur.

Tədqiqat işinin metodikası: Tədqiq olunan işin nəzəri və metodoloji əsasını xarici və yerli alimlərin elmi işləri, rəqəmsal marketinqin qaydaları təşkil edir.

Nəzəri və praktiki əhəmiyyəti: Tədqiq olunan dissertasiya işin məzmununa uyğun olaraq nəzəri və praktiki əhəmiyyəti uyğun olaraq aşağıdakı şəkildə müəyyən edilmişdir:

1. İnternetdə Restoran üçün mobil tətbiqlərin yaradılması, tətbiqin təhlil edilməsi,
2. Verilmiş restoran və kafenin saytında ilkin məlumatların, eskizlərin və məhsul dizaynlarının hazırlanması.
3. Əməliyyat sistemi ilə işləyən mobil tətbiqin android sistemlərində realizasiyası.

Dissertasiya işinin strukturu və həcmi: Dissertasiya işi girişdən, üç fəsildən, nəticə və ədəbiyyat siyahısından ibarətdir.

Birinci fəsildə inteqrasiya edilmiş mobil tətbiq sisteminin nəzəri aspektləri, mobil tətbiqlərin inkişafı və effektivliyinin analizi, mobil tətbiqlərin texniki aspektləri, mobil tətbiqlərin günümüzdəki rolu haqqında məlumat verilmişdir.

İkinci fəsildə mobil tətbiqinin dizaynı və mobil proqramın hazırlanması, restoranlar mobil tətbiqinin interfeys və qrafik dizaynı, restoranlar mobil tətbiqi üçün proqramlaşdırma dili və texnologiyaların seçimi, restoranlar mobil tətbiqi üçün Firebase məlumat bazasının hazırlanması haqqında ətraflı məlumatlar verilmişdir.

Ü

ç

ü

n

c

ü

f

ə

s

i

l

d

ə

FƏSİL I

İnteqrasiya edilmiş mobil tətbiq sisteminin nəzəri aspektləri

1.1 Mobil tətbiqlərin inkişafı və effektivliyinin analizi

Mobil tətbiqlərin inkişafı, mobil telefonların və smartfonların yayılması ilə birlikdə başlayıb sürətlənmişdir. İlk mobil tətbiqlər, əvvəlcə əsas funksiyalar, məsələn, telefon zəngləri, SMS, kalendər və çalar saat kimi funksiyaları təklif edirdi. İlk mobil tətbiqlər əsasən mobil operatorların tərəfindən təklif olunan xidmətlərə məqsədəuyğunluq və təşkilatçılıq əsasında inkişaf etdi. Əvvəlcə, mobil oyunlar, mobil musiqi və rəsmi xidmətlər, məsələn, bank xidmətləri və məlumatların mobil şəbəkə vasitəsilə əldə edilməsi ilə əlaqədar tətbiqlər məşhur oldu. 2007-ci ildə Apple iPhone-un piyasaya çıxması ilə mobil tətbiqlərin inkişafı tam anlamı ilə sürətlənmişdir. iOS əməliyyat sistemində çalışan tətbiqlər, App Store vasitəsilə geniş yayılmağa başlamış və bir çox fərqli kateqoriyalarda məşhur tətbiqlər ortaya çıxmışdır. 2008-ci ildə Google Android əməliyyat sistemini təqdim etdi. Bu, açıq mənbəli bir platforma əsaslanan və çox sayda mobil cihazda işləyən tətbiqlərin inkişafını daha da artırdı [9].

tətbiqləri, Google Play Store vasitəsilə endirilə bilər və məşhur tətbiqlər, oyunlar, sosial media platformaları, məlumat və əyləncə tətbiqləri ilə zənginləşdirildi. Son illərdə,

ə

ə

ə

ə

ər bir mobil platformanın inkişafı ilə birlikdə, mobil tətbiqlər üçün dillərin və texnologiyaların inkişafı da yaşanmışdır. Əvvəlcədən, Java və Objective-C kimi dillər effektivliyinin analizi, mobil texnologiyaların artan populyarlığı ilə birlikdə önəmli bir

ə

ə

ə

ə

r

tətbiqlərin inkişafı və effektivliyinin analizi, bir neçə mərhələni əhatə edir:

n

k

i

ək və ehtiyacların müəyyənləşdirilməsi: Bir mobil tətbiq inkişaf prosesinin əvvəlində, açılışın sorğularının və geribildirimlərinin analizi, rəqabət təhlili, istifadəçi təcrübəsi araşdırmaları və s. ilə aparıla bilər.

lanlaşdırma və dizayn: Mobil tətbiqlərin effektivliyini artırmaq üçün planlaşdırma və inteqrasiya edilmiş funksiyalar və optimal performans bu mərhələdə nəzərə alınmalıdır.

inkişaf və proqramlaşdırma: Mobil tətbiqlərin inkişaf prosesi, proqramlaşdırma funksiyaları test etmək və geribildirimlərdən istifadə etmək bu mərhələdə əhəmiyyətlidir. Proqramlaşdırma dilləri və texnologiyaları, məsələn, Java, Objective-C əhəmiyyətlidir. Müxtəlif mobil platformalar, məsələn, iOS və Android üçün fərqli proqramlaşdırma dilləri və texnologiyalar tələb edir.

ə səhmlər: Mobil tətbiqlərin effektivliyini qiymətləndirmək üçün geniş test və səhmlər aparılmalıdır. Bu, tətbiqin fəaliyyətinin istənilən funksiyalarını və performansını İstifadəçi təcrübəsi testləri və kompüterləşdirilmiş test prosedurları da tətbiqlərin

ayımlanma və monitorinq: Mobil tətbiqlərin effektivliyi, yayımlanma prosesindən əhəmiyyətlidir. Tətbiqlərin yenilənməsi, təhlükəsizlik güncəlləşdirmələri və yeni funksiyaların əlavəsi belə monitorinq prosesi ilə əlaqəlidir [4].

1.2 Mobil tətbiqlərin texniki aspektləri

ə optimizasiya: Mobil tətbiqlər istifadəçilərə sürətli, məlumatları effektiv şəkildə işlədən və enerji istehlakını minimal saxlayan bir təcrübə təklif olunması və batareyaya istehlakının idarə olunması kimi məsələlər mobil tətbiqlərin

ənlik və məxfilik: Mobil tətbiqlər, istifadəçilərin məxfiliyini və əlumatlarının təhlükəsizliyini təmin etməlidir. Verilənlər əmanətliliyi, şifrələmə, autentifikasiya, kullanıcı izləməsi və digər tədbirlər, mobil tətbiqlərin güvənliyi və

ə
əstək və yeniləmələr əhəmiyyətli bir məsələdir. Mobil tətbiqlərin səhvkarlıqlarını

ə
ə
ə
ə
ənmə və tətbiq mağazaları: Mobil tətbiqlər, müştərilərə tətbiqlərin rahatlıqla yüklənməsi və güncəllənməsi imkanı təklif etməlidir. Mobil tətbiqlərin tətbiq mağazalarında (App Store və ya Google Play Store kimi) mövcud olması, yükləmə prosesini asanlaşdırır və daha çox istifadəçi tərəfindən əldə olunmağa imkan verir.

ə
ək dərəcəli məsuliyyət və təhlükəsizlik: Mobil tətbiqlər müştərilər üçün əhəmiyyətli məlumatları saxladığından, məsuliyyət və təhlükəsizlik aspekti çox əhəmiyyətli haldır.

- **Analitika və performans izləmə:** Mobil tətbiqlər, istifadəçilərin davranışını
 - **Yerli və bulud integrasiya:** Mobil tətbiqlər, yerli cihaz funksionallığından
 - **İstifadəçi geri bildirimləri:** İstifadəçilərin mobil tətbiqlərlə bağlı geri
- əri, tətbiqin inkişaf prosesində və yeniləmələrində əhəmiyyətli bir rola malikdir. Mobil yaxşı bir təcrübə təklif etmək üçün əhəmiyyətli addımlardır.**

əhlükəsizlik: Mobil tətbiqlərin təhlükəsizlik məsələləri, müştərilərin

ə
işləmə təcrübəsi təklif etməlidir.

- ə s
- ə ə ya bulud bazalı mühit: Mobil tətbiqlərin yerli (native) və ya bulud bazalı
- ə a
- ə nələr bazası idarəsi: Mobil tətbiqlər, istifadəçilərə dair məlumatları idarə
- ək üçün bir verilənlər bazası sistemi ilə əlaqəli olmalıdır. Məlumatların saxlanması,
- n l
- k
- ri • Yerli qurğulara inteqrasiya: Mobil tətbiqlər, cihazın yerli qurğuları ilə
- ş
- ə və paylaşma funksiyaları: Mobil tətbiqlər, məlumatları çap etmə və
- paylaşma funksiyalarını dəstəkləməlidir. İstifadəçilərə məlumatları sənaye
- standartlarına uyğun şəkildə çap etmə, e-poçt ilə göndərmə, sosial mediya
- platformalarında paylaşma və digər kommunikasiya forması ilə paylaşma imkanı təklif
- ə ə
- ə nələr və API inteqrasiyası: Mobil tətbiqlər, çeşitli yerli xidmətlər və
- ərlə inteqrasiya olunmalıdır. Məsələn, ödəniş gateway-ləri, məlumat analitika
- b ə
- a
- k ə
- ə ə
- h • Push bildirişləri: Mobil tətbiqlər, push bildirişləri vasitəsilə istifadəçilərə
- s
- ə ə təşviqat kampaniyaları: Mobil tətbiqlər, reklam və təşviqat
- kampaniyalarını dəstəkləməlidir. Bu, tətbiqlərin daha geniş bir istifadəçi qrupuna
- ş l
- ə ə ya Google Assitant kimi səsləndirilən köməkçilərə inteqrasiya: Mobil
- ə ğ
- h 1
- ə ə

ətbiqlər, Siri, Google Assistant və ya digər səsəndirilən köməkçilərlə enteqrasiya edilə

- İnkişaf dəstəyi və əlavə tədris resursları: Mobil tətbiqlər, inkişafçılara əlavə ədris resursları, məlumat bazası və dəstək təqdim etməlidir. Bu, inkişafçıların tətbiqləri daha effektiv şəkildə inkişaf etdirmələrini və problemləri həll etmələrini təmin edir [6].

ə

ə

1.3 Mobil tətbiqlərin günümüzdəki rolu

ətbiqlər günümüzün vəziyyətində digər hər hansı texnologiya vasitələri ilə müqayisədə, əbul edilir.

ə

ə

ətbiqlər, həm iş, həm də həyat təcrübəmizi ciddi mənada dəyişdirmiş və yaradır. İş dünyasında, şirkətlər müştərilərə və çalışanlara məhsul və xidmətlərini daha müştərilərə təcrübəli alış-veriş imkanları, kollektiv iş fəaliyyətləri, məhsul və

ə

ə

ə

ə

ə

ə

ə

ə

əmçinin, mobil tətbiqlər müxtəlif sahələrdəki iş proseslərini effektivləşdirir və avtomatlaşdırır. Məsələn, bankacılıq sektorunda mobil bankçılıq tətbiqləri vasitəsilə rahatlıqla həyata keçirə bilirlər. Bu, müştərilər üçün daha çox müəyyənlik, mobil əsaslı

i

ş

ə

ə

ə

ə

Sağlamlıq və tibb sahəsində mobil tətbiqlər də çox vacib bir rol oynayır. İnsanlar artıq sağlamlıq monitorinqi, idman və mədəniyyət proqramları, əhali sağlamlığı

ə

ə

ə

ə

ə

tətbiqlər, iş yerində kommunikasiya və iş fəaliyyətləri üçün tətbiq edilə bilər. Onlar toplantılar təşkil etməyə, proqramları izləməyə, e-poçtları oxumağa və iş tapşırıqlarını qurmağı, onların ehtiyaclarını qarşılamağı və məhsul və xidmətləri daha sürətli və effektiv şəkildə təklif etmək imkanı verir. Mobil tətbiqlər, təhsil və öyrənmə sahəsində materiallarına asan giriş imkanı verir və həm də tələbələrin interaktiv təhsil

a

ğ

h

ü

n

ti

d

ə

h

ə

Əzi mobil tətbiqlər, istifadəçilərin diqqətini dağıtabilər və onların iş performansını ehtimalların sərflənməsinə səbəb olaraq iş effektivliyini azalda bilər. Mobil tətbiqlərin psixoloji sağlamlığı təhlükə altına ala bilər. Sosial şəbəkələr və əyləncə tətbiqləri,

h

b

d

q

ü

h

y

h

ə izolyasiyanın artmasına səbəb ola bilər. Bəzi mobil tətbiqlər, sürətli və təhlükəsiz sürüşə nəzarətə təsir edə bilər. Sürücülər əyləncə tətbiqləri ilə məşğul olduqunda,

ə

əzi mobil tətbiqlər, məhsul satışı və reklam platformaları üçün əhəmiyyətli bir təsire onların görə reklamların hədəfli şəkildə təklif edilməsi, mobil tətbiqlərin mənfi bir şəkildə satılmasına yönəldilə bilər, bu da alış-veriş həvəsinə dayalı impulsiv

ə

b

a

ğ

l

ı

đ

đ

z

đ

k

i

ü

ş

ä

ğ

l

ä

m

f

o

ø

m

a

FƏSİL II

Mobil tətbiqinin dizaynı və mobil proqramın hazırlanması

tətbiqinin interfeys və qrafik dizaynı

ə

ə

ə

ə

n

d

f

ö

k

d

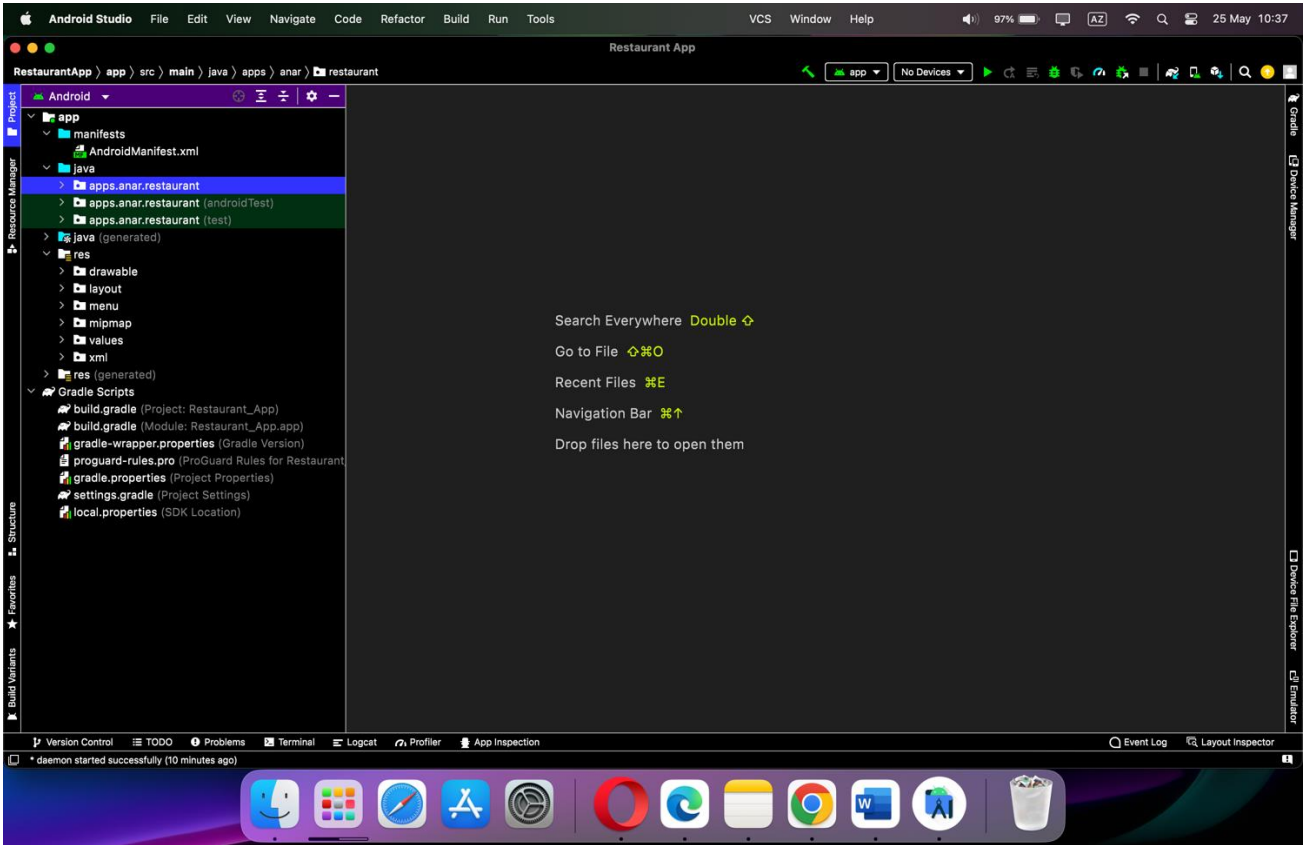
d

ş

z

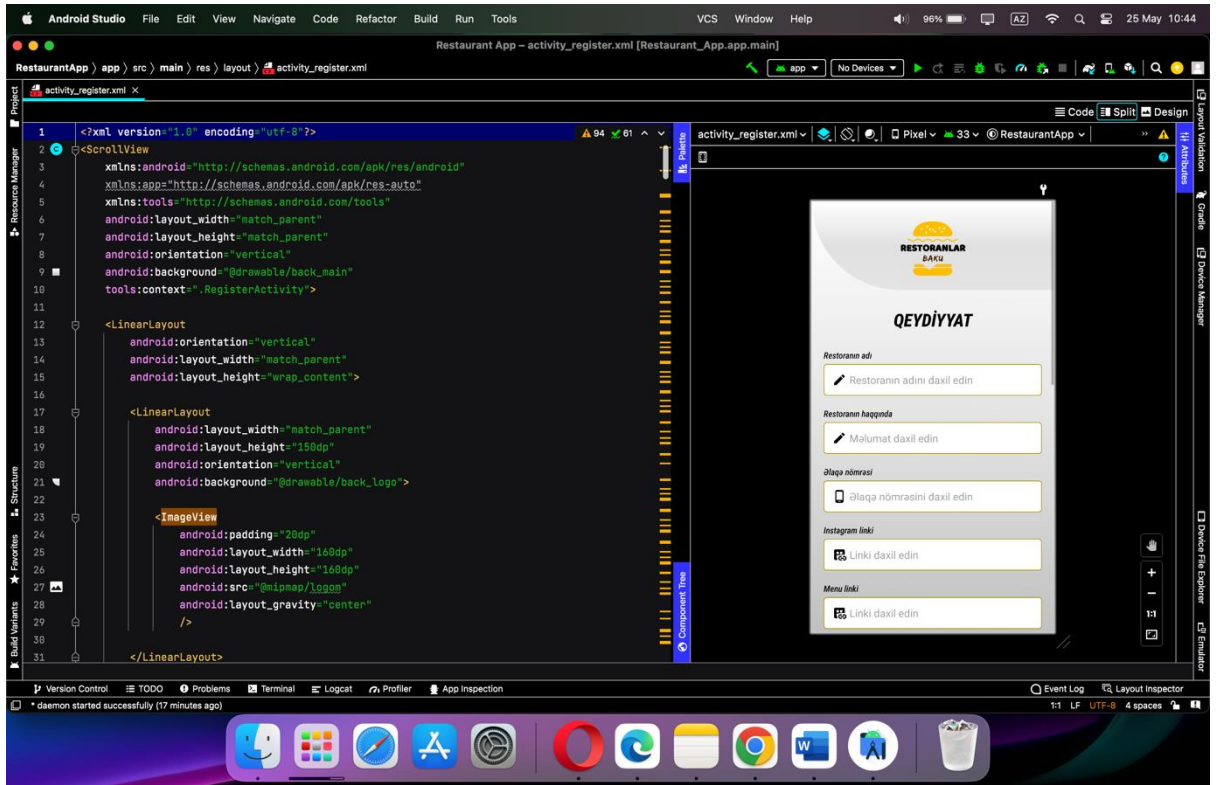
u

d



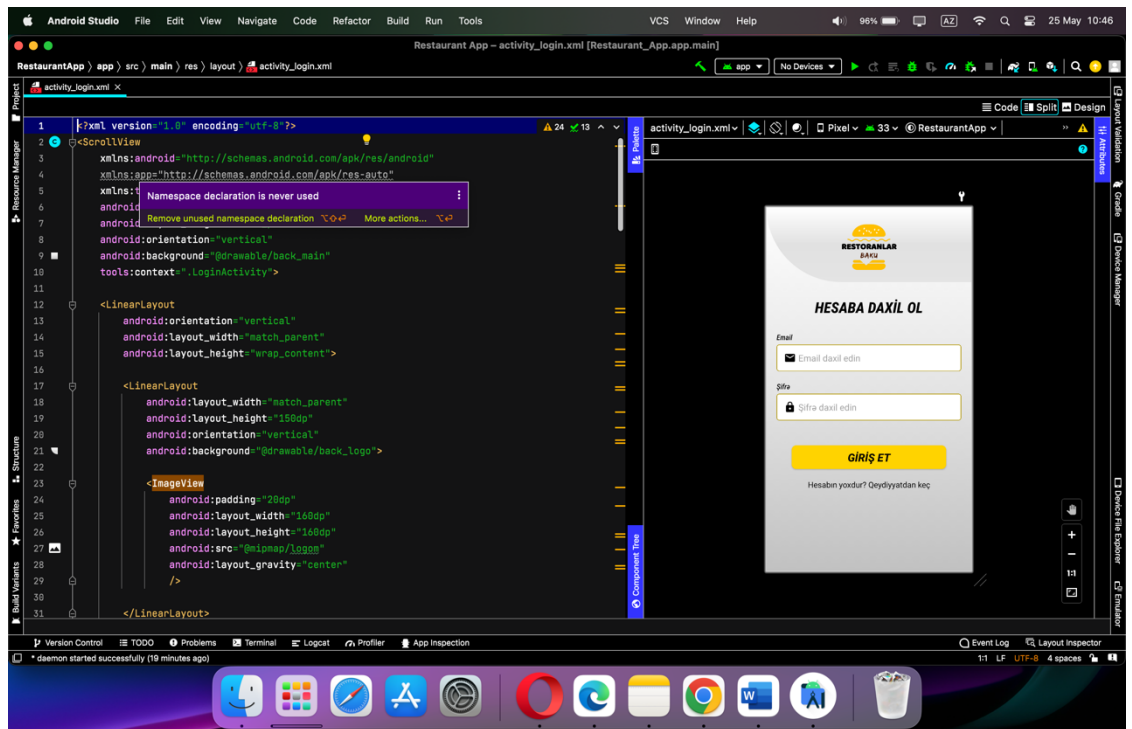
Şəkil 1. Android studio Restoranlar proyeği

Qeydiyyatdan keçmək üçün ekranın dizaynı və dizayn kodları aşağıdakı şəkildə göstərilmişdir. Bu səhifəyə restoran haqqında məlumatları yazaraq restoranınızı mobil tətbiqdə qeydiyyatdan keçirə bilərsiniz.



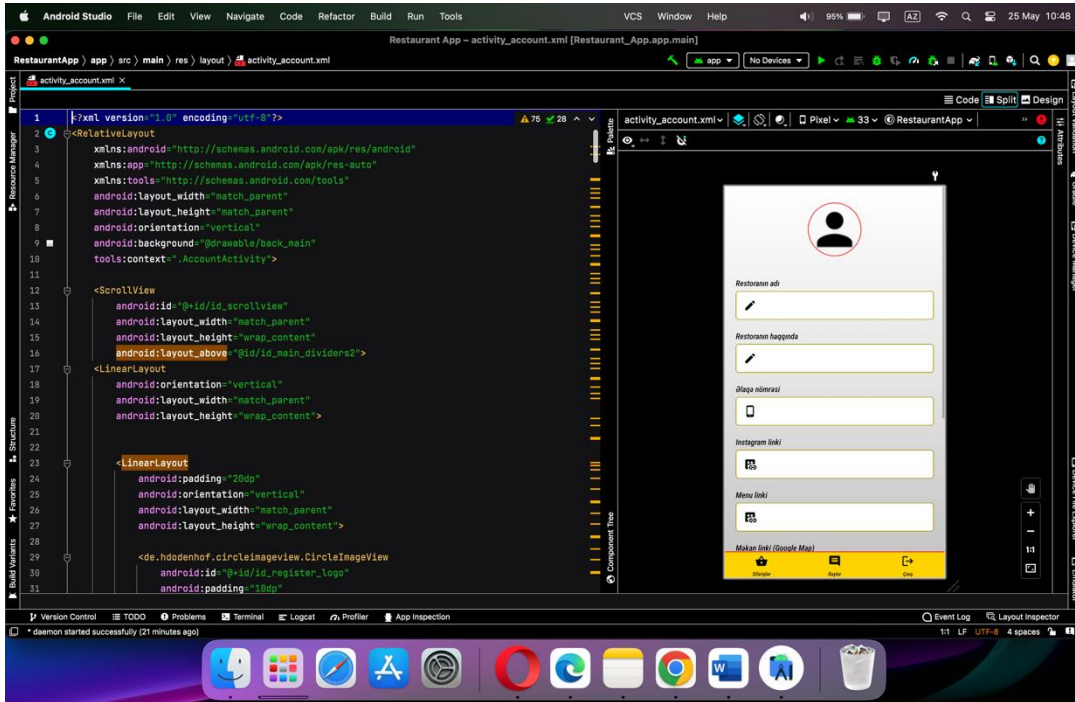
Şəkil 2. Qeydiyyat ekranı üçün dizayn

Hesaba daxil olmaq ekranının dizaynı və dizayn kodları aşağıdakı şəkildə göstərilmişdir.



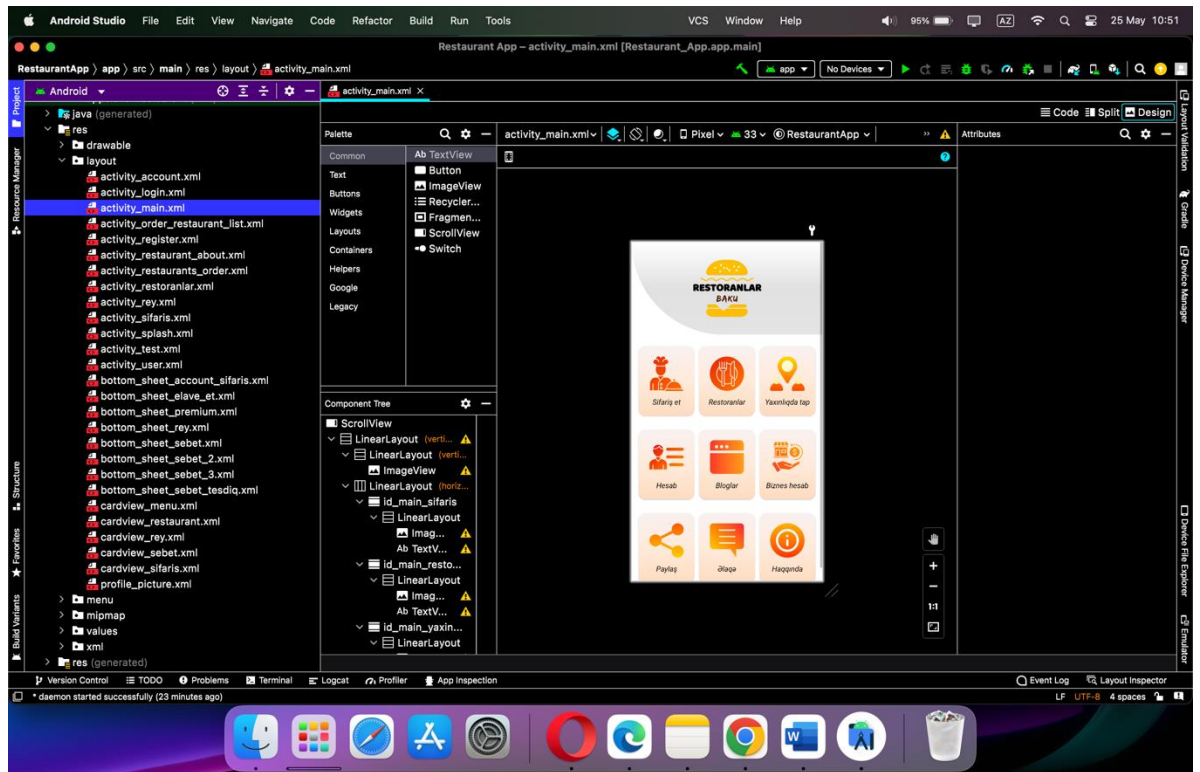
Şəkil 2. Hesaba daxil ol ekranı üçün dizayn

Hesabı idarə etmək üçün ekranın dizaynı və dizayn kodları aşağıdakı şəkildə göstərilmişdir.



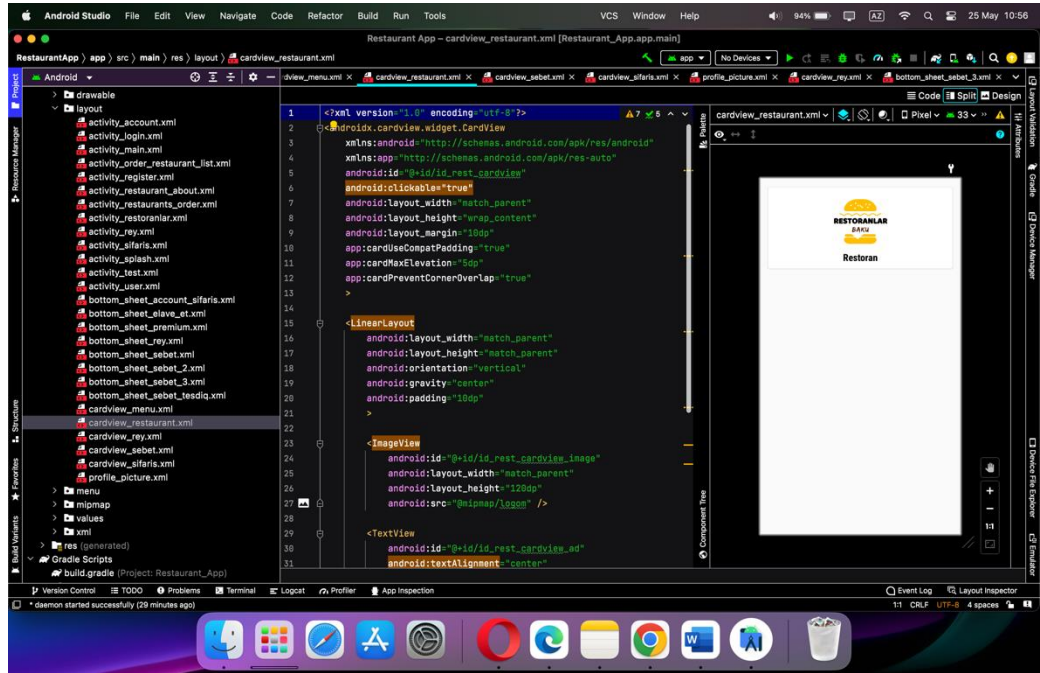
Şəkil 3. Hesab ekranı üçün dizayn

Əsas ekranın dizaynı və dizayn kodları aşağıdakı şəkildə göstərilmişdir.



Şəkil 4. Əsas ekran üçün dizayn

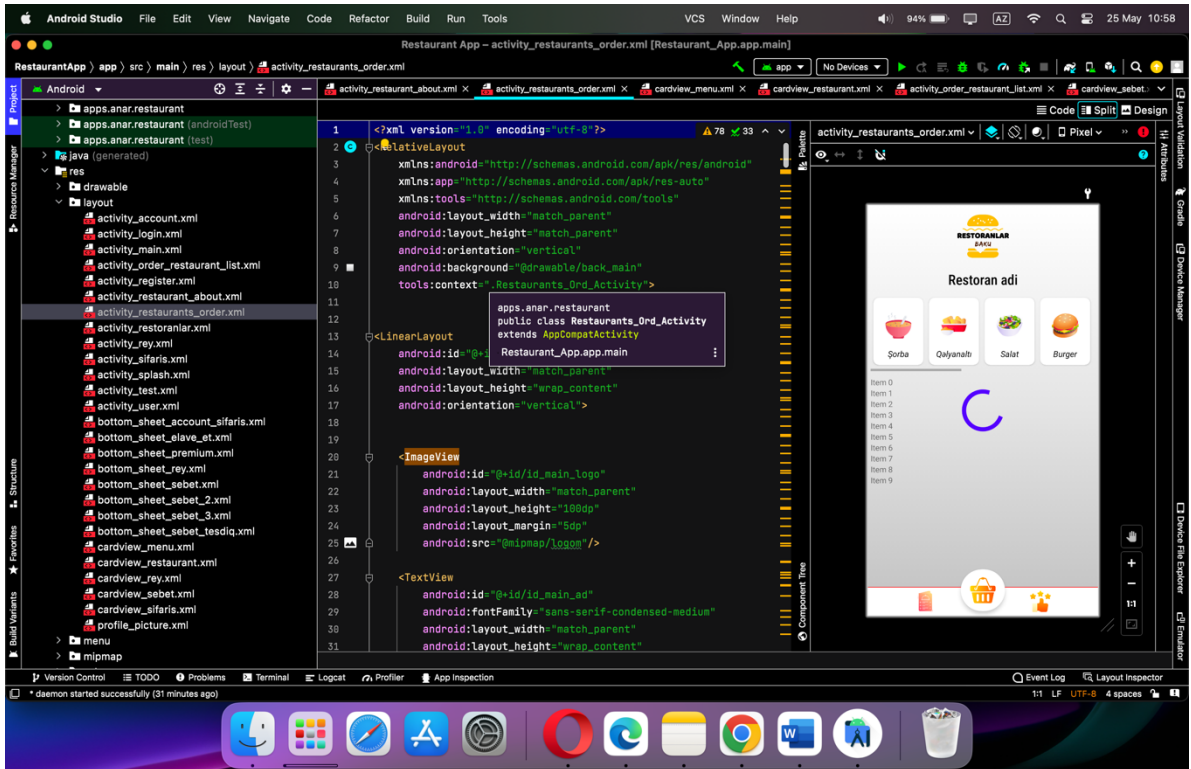
Qeydiyyatdan keçdikdən sonra restoran bu siyahı üzrə sıralanır. Həmin ekranın dizaynı və dizayn kodları aşağıdakı şəkildə göstərilmişdir.



Şəkil 5. Restoranlar siyahısı ekranı üçün dizayn

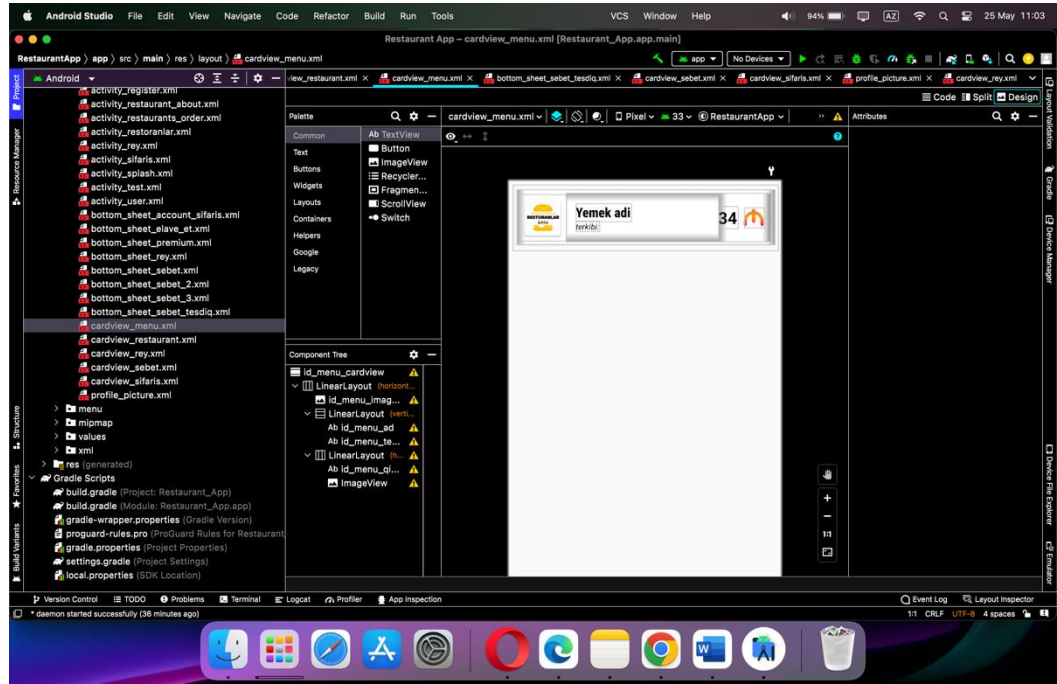
Restoran əgər premium hesabdırsa həmin restoranda yeməklər sifariş etmək olar.

Aşağıdakı dizayn bunu əks etdirir.



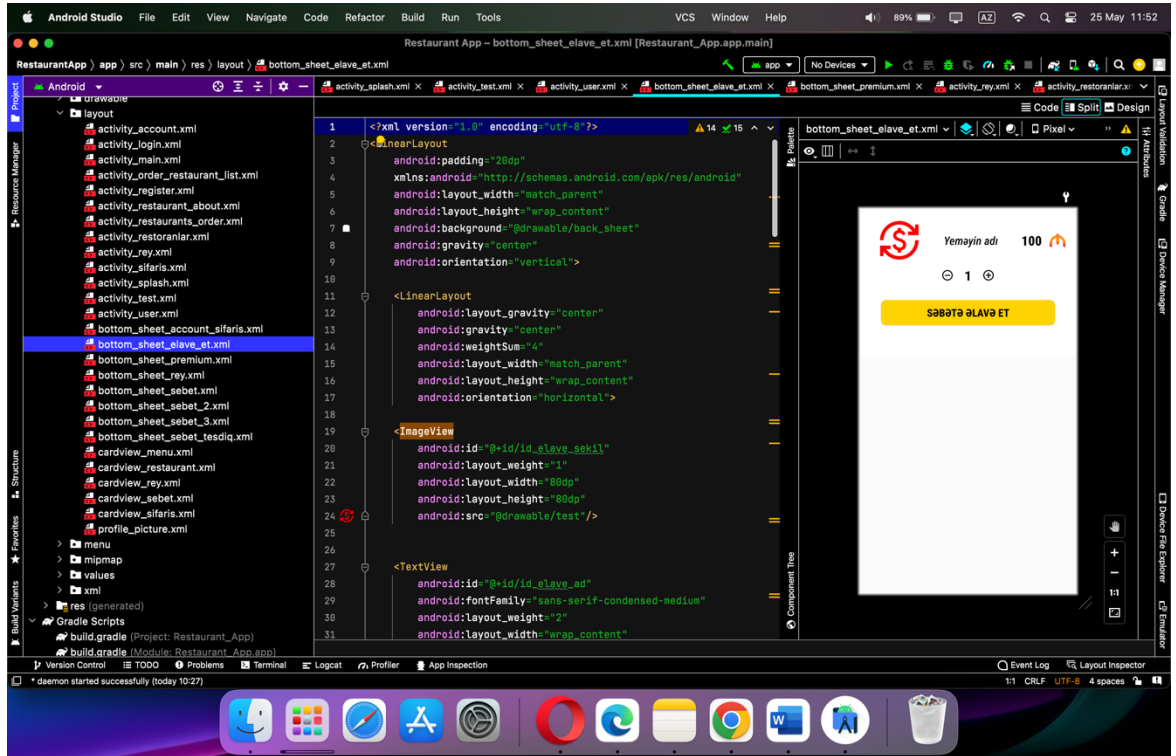
Şəkil 6. Sifariş et ekranı üçün dizayn

Menyudakı yeməklər aşağıdakı dizayn kimi mövcud olacaqdır.



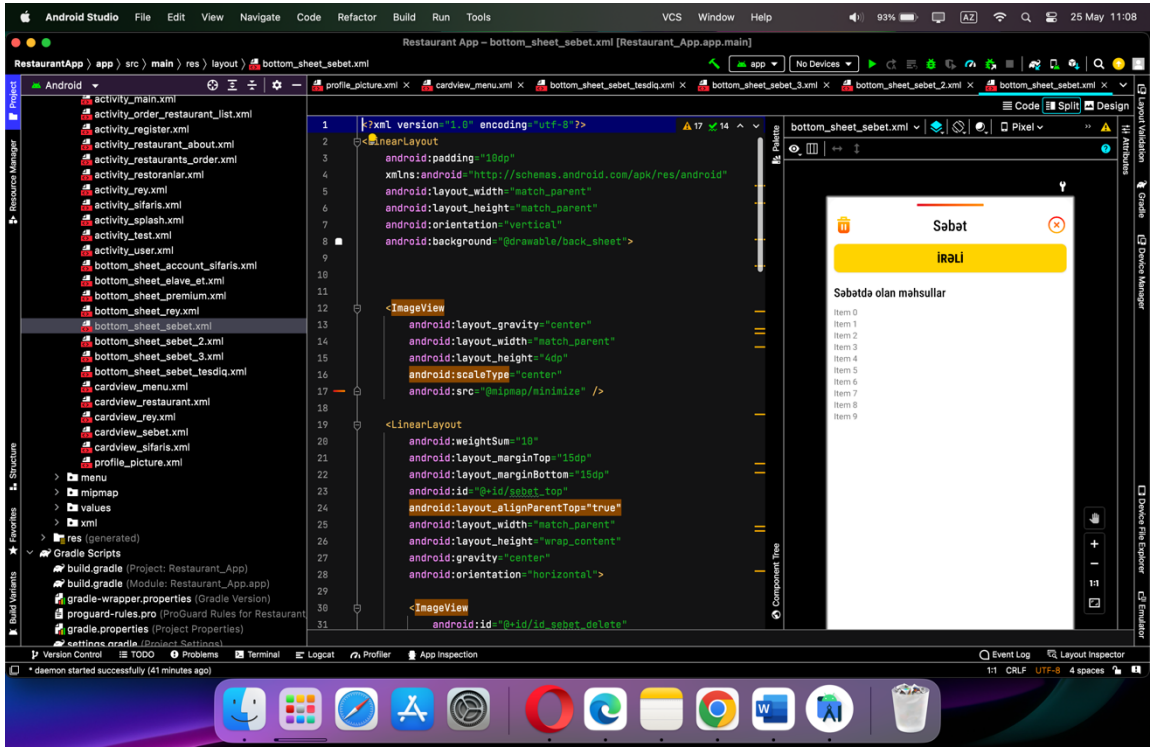
Şəkil 7. Menü üçün dizayn

Menyudakı yeməkləri sərbətə əlavə etmək mümkün. Bu ekranın dizaynı aşağıda göstərilmişdir.



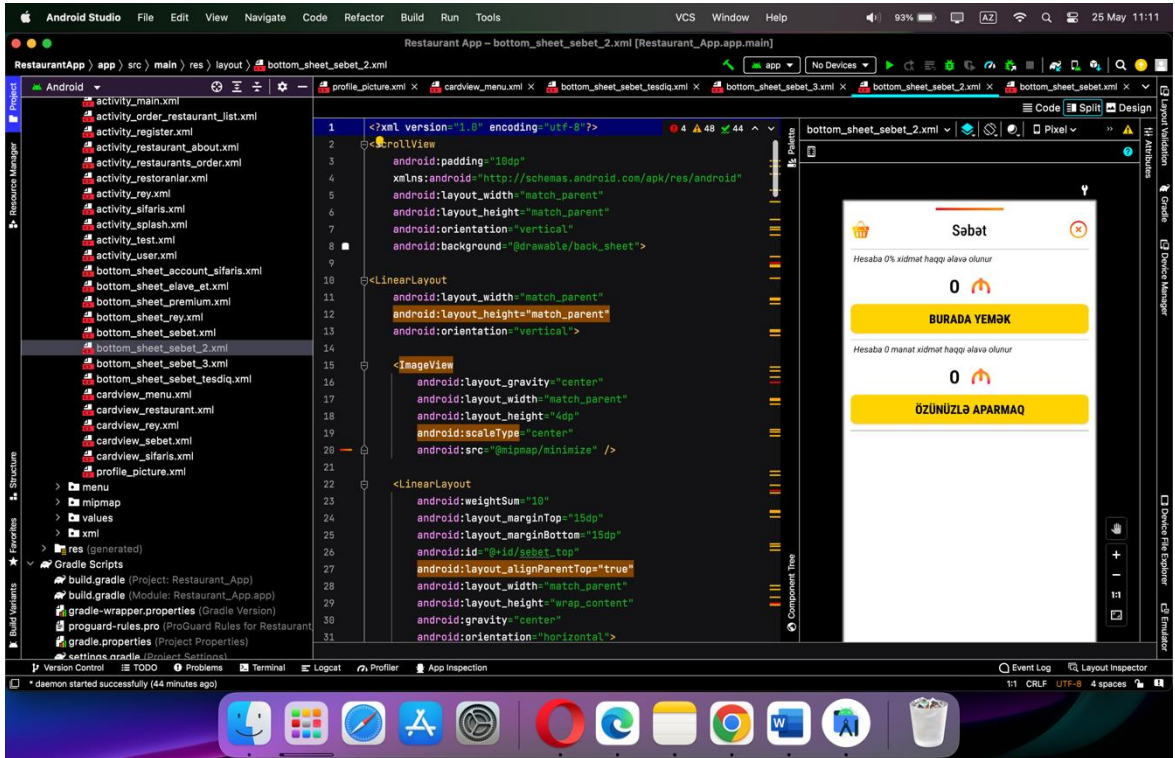
Şəkil 8. Yeməkləri sərbətə əlavə etmək üçün dizayn

Menyudaki yeməkləri sərbət əlavə etmək mümkündür. Bu ekranın dizaynı aşağıda göstərilmişdir.



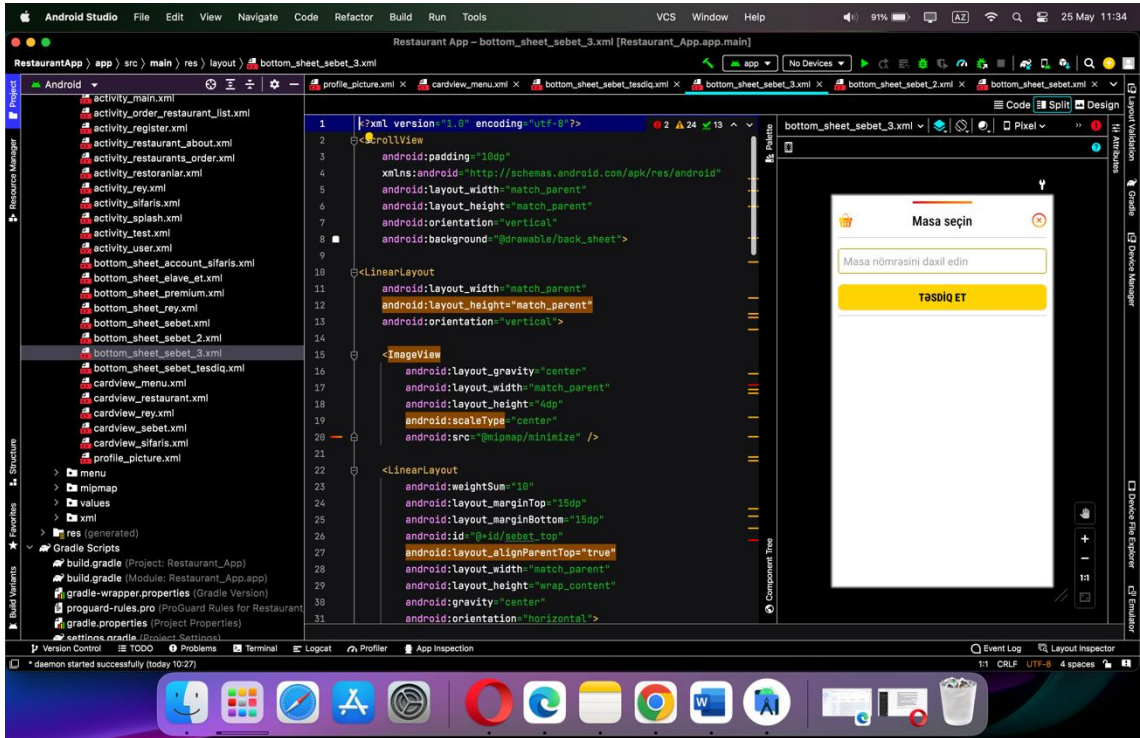
Şəkil 9. Səbət üçün dizayn

İrəli düyməsinə basdıqdan sonra açılan ekranın dizaynı aşağıda göstərilmişdir.



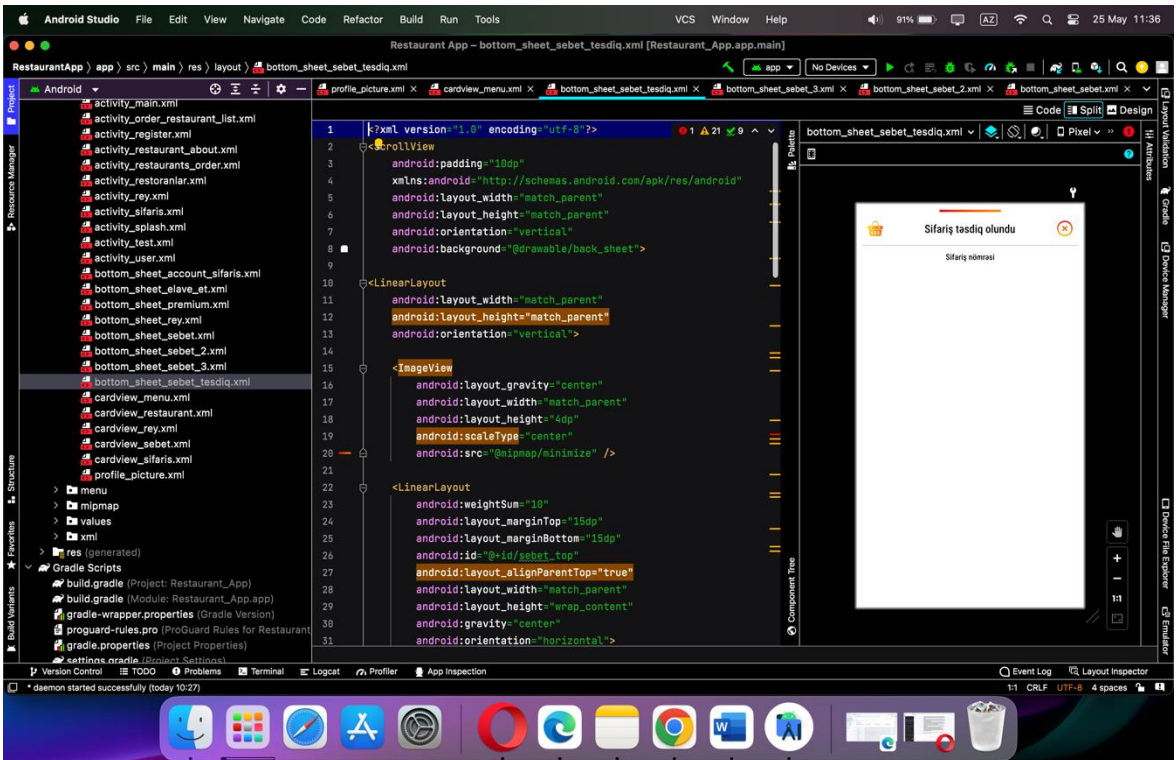
Şəkil 10. Səbət üçün dizayn

Burada yemek seçimi etdikdən sonra aşağıdakı ekran dizaynı görünəcəkdir.



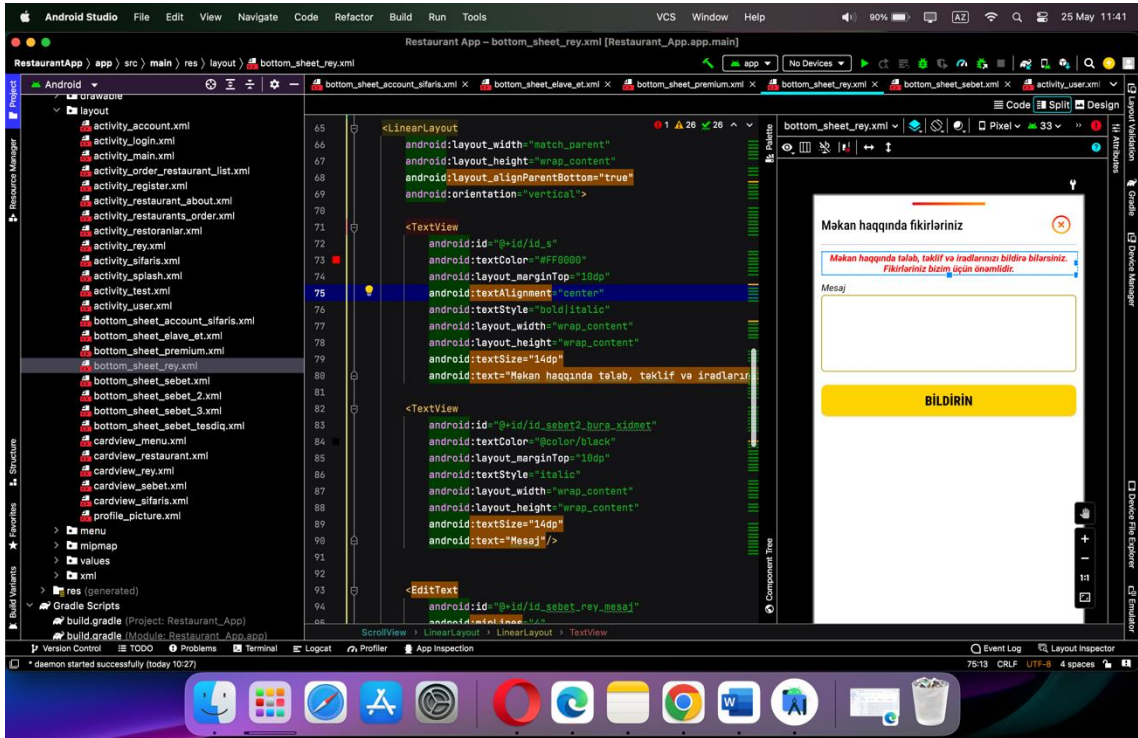
Şəkil 11. Masa nömrəsi ekranı üçün dizayn

Sifariş kodu ekranı üçün dizayn aşağıdakı dizayn kimi qurulacaq.



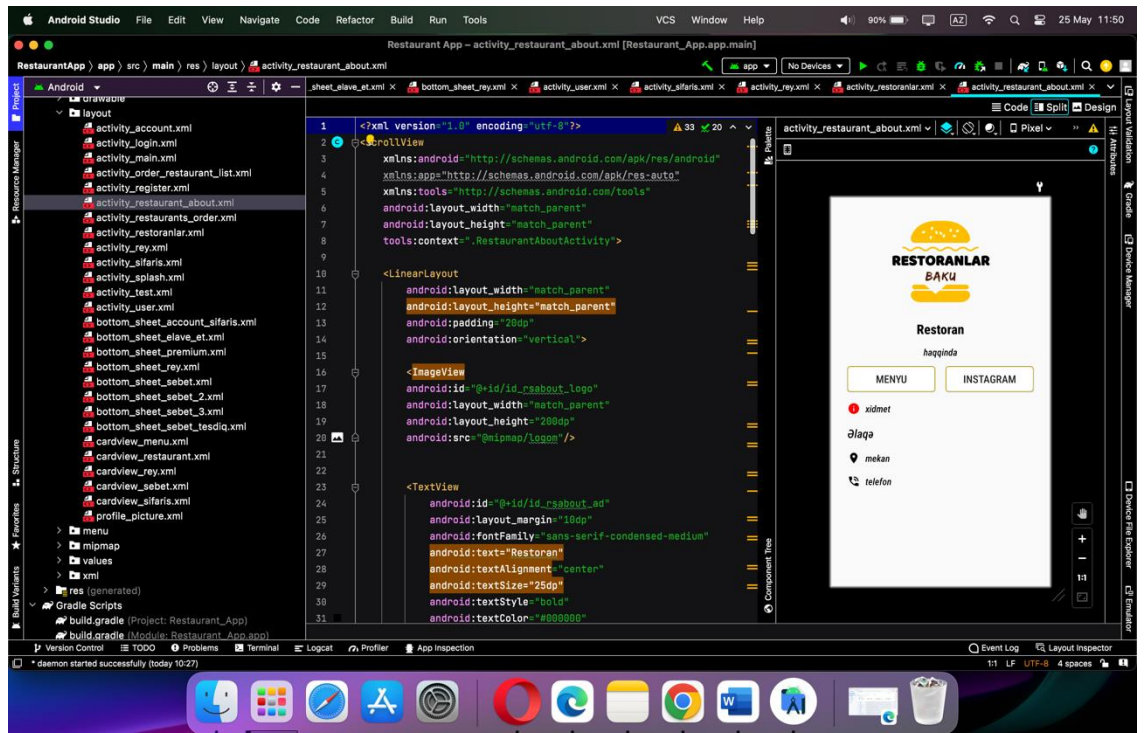
Şəkil 12. Sifariş kodu ekranı

Hər bir müştərinin restoran haqqında fikir bildirməsi üçün ekranın dizaynı aşağıdakı kimi olacaq



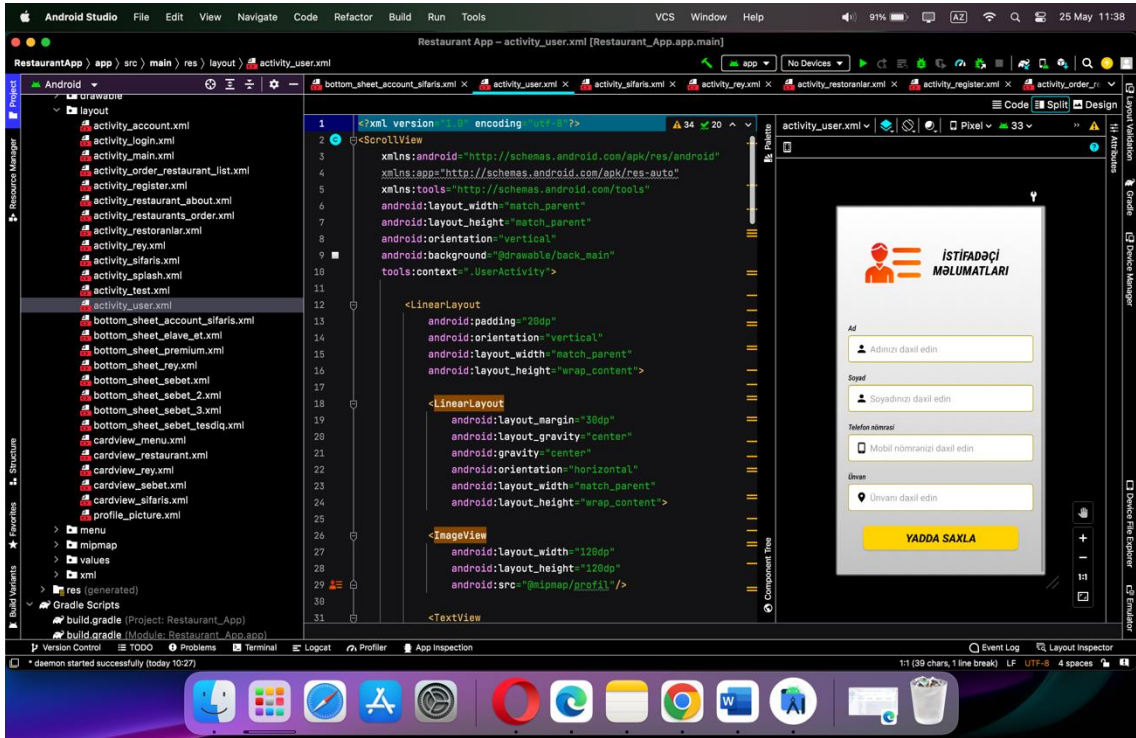
Şəkil 13. Rəy ekranı üçün dizayn

Restoran haqqında ümumi məlumat ekranının dizaynı aşağıdakı kimi olacaqdır.



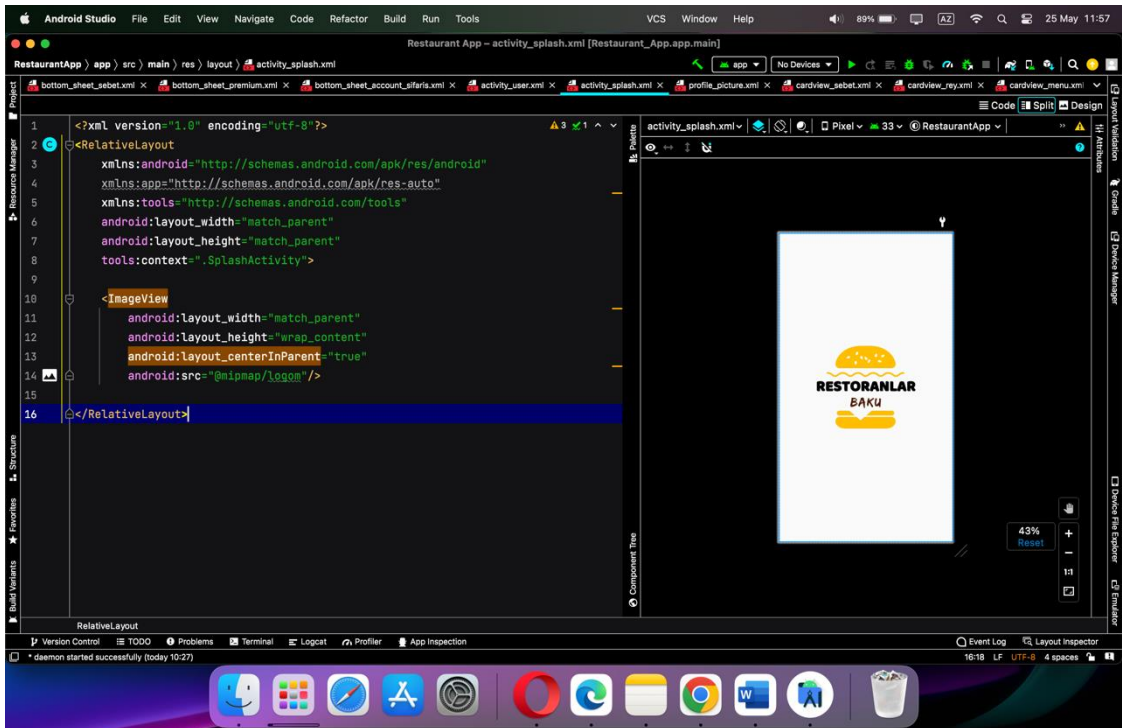
Şəkil 14. Restoran haqqında ekran üçün dizayn

Müştəri məlumatları ekranının dizaynı aşağıdakı dizayn kimi olacaqdır.



Şəkil 15. Müştəri məlumatları ekran dizaynı

Splash ekranının dizaynı aşağıdakı dizayn kimi olacaqdır.



Şəkil 16. Splash ekran dizaynı

2.2 Mobil tətbiq üçün proqramlaşdırma dili və texnologiyaların seçimi

Restoranlar mobil tətbiqi Android studioda Java proqramlaşdırma dilində yazılmışdır. İndi isə kodları nəzərdən keçirək. Qeydiyyat ekranının kodlarının bir qismi aşağıda qeyd edilmişdir.

```
import ...

public class RegisterActivity extends AppCompatActivity {

    EditText ad, haqqinda, nomre, instagram, menu, mekan, sahib, email, sifre, tekrar_sifre, xidmet_haqqi, unvan_soz;
    Button hesabYarat;
    TextView giris;
    ImageView ima;
    ProgressBar p;

    private Uri MyimageUri;
    FirebaseAuth auth;
    final private StorageReference storageReference = FirebaseStorage.getInstance().getReference();

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_register);
        getSupportActionBar().hide();

        auth = FirebaseAuth.getInstance();

        ad = findViewById(R.id.id_register_rest_adi);
        haqqinda = findViewById(R.id.id_register_rest_haqqinda);
        nomre = findViewById(R.id.id_register_telefon);
        instagram = findViewById(R.id.id_register_instagram);
        menu = findViewById(R.id.id_register_menu);
        mekan = findViewById(R.id.id_register_mekan);
        sahib = findViewById(R.id.id_register_sahibi);
        email = findViewById(R.id.id_register_mail);
        sifre = findViewById(R.id.id_register_sifre);
    }
}
```

```
ActivityResultLauncher<Intent> activityResultLauncher = registerForActivityResult(
    new ActivityResultContracts.StartActivityForResult(),
    new ActivityResultCallback<ActivityResult>() {
        @Override
        public void onActivityResult(ActivityResult result) {
            if (result.getResultCode() == Activity.RESULT_OK) {
                Intent data = result.getData();
                MyimageUri = data.getData();
                imageView.setImageURI(MyimageUri);
            }
            else {
                Toast.makeText(context, RegisterActivity.this, text: "Səkil seçilməyib", Toast.LENGTH_SHORT).show();
            }
        }
    });

imageView.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        Intent photoPicker = new Intent();
        photoPicker.setAction(Intent.ACTION_GET_CONTENT);
        photoPicker.setType("image/*");
        activityResultLauncher.launch(photoPicker);
    }
});

hesabYarat.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        String text_ad = ad.getText().toString();
        String text_haqqinda = haqqinda.getText().toString();
    }
});
```

Hesaba daxil ekranının kodlarının bir hissəsi aşağıda göstərilib.

```

public class LoginActivity extends AppCompatActivity {

    EditText mail, sifre;
    Button button;
    TextView qeydiyyat;
    FirebaseAuth auth;
    ProgressBar progressBar;
    SharedPreferences sharedPreferences = new SharedPreferences();

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_login);
        getSupportActionBar().hide();

        mail = findViewById(R.id.id_login_mail);
        sifre = findViewById(R.id.id_login_sifre);
        button = findViewById(R.id.id_login_btn);
        qeydiyyat = findViewById(R.id.id_login_qeydiyyat);
        progressBar = findViewById(R.id.id_login_progress);
        auth = FirebaseAuth.getInstance();

        button.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String txt_mail = mail.getText().toString();
                String txt_sifre = sifre.getText().toString();

                if (txt_mail.isEmpty()){
                    mail.setError("Tələb olunur");
                    return;
                }

                return;

                progressBar.setVisibility(View.VISIBLE);

                auth.signInWithEmailAndPassword(txt_mail, txt_sifre).addOnCompleteListener(new OnCompleteListener<AuthResult>() {
                    @Override
                    public void onComplete(@NonNull Task<AuthResult> task) {
                        if (task.isSuccessful()){
                            progressBar.setVisibility(View.GONE);
                            startActivity(new Intent(getApplicationContext(), LoginActivity.this, AccountActivity.class));

                            sharedPreferences.infoWriteBusiness(context: LoginActivity.this, key: "login1", txt_mail);
                            sharedPreferences.infoWriteBusiness(context: LoginActivity.this, key: "login2", txt_sifre);
                        }else {
                            progressBar.setVisibility(View.GONE);
                            Toast.makeText(context: LoginActivity.this, task.getException().getMessage(), Toast.LENGTH_SHORT).show();
                        }
                    }
                });
            }
        });

        qeydiyyat.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                startActivity(new Intent(getApplicationContext(), RegisterActivity.class));
            }
        });
    }
}

```

Hesab idarəetmə ekranının kodlarının bir hissəsi aşağıda göstərilib.


```

public class Restaurants_Ord_Activity extends AppCompatActivity {

    RecyclerView recyclerView;
    DatabaseReference databaseReference;
    ArrayList<MenuModel> list;
    RecyclerViewAdapterMenu myAdapter;
    ProgressBar progressBar;

    CardView sebet;
    TextView restAd;
    ImageView logo, rey, info;
    CardView sorba, qelyanalti, salat, burger, steyk, pizza, isti_yemek, kabab,sirniyyat, icki;

    String rest_id;
    int xidmetHaqqi, catdirilmaHaqqi;

    SharedPreferences sharedPreferences = new SharedPreferences();

    double umumimeblegFloat;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_restaurants_order);
        getSupportActionBar().hide();

        findViewById();

        Intent intent = getIntent();
        rest_id = intent.getExtras().getString(key: "rest_id");

        restAbout();

```

```

ImageView delete = sheetview.findViewById(R.id.id_sebet_delete);

RecyclerViewAdapterSebet myAdapterx = new RecyclerViewAdapterSebet(MyContext: Restaurants_Ord_Activity.this, list);
recyclerViewtesdiq.setHasFixedSize(true);
recyclerViewtesdiq.setLayoutManager(new LinearLayoutManager(context: Restaurants_Ord_Activity.this));
recyclerViewtesdiq.setAdapter(myAdapterx);
recyclerViewtesdiq.setLayoutManager(new GridLayoutManager(context: Restaurants_Ord_Activity.this, spanCount: 1));

close.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) { bottomSheetDialog.dismiss(); }
});

delete.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View v) {
        listSebet.clear();
        bottomSheetDialog.dismiss();

        Toast.makeText(context: Restaurants_Ord_Activity.this, text: "Sebet sıfırlandı", Toast.LENGTH_SHORT).show();
    }
});

ireli.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        bottomSheetDialog.dismiss();
        BottomSheetDialog bottomSheetDialog2;
        bottomSheetDialog2 = new BottomSheetDialog(context: Restaurants_Ord_Activity.this, R.style.BottomSheetDialog);
        View sheetview2 = LayoutInflater.from(Restaurants_Ord_Activity.this.getApplicationContext()).inflate(R.layout.sheetview2, bottomSheetDialog2.findViewById(R.id.container), true);

```

Restoranlar ekranının kodlarının bir hissəsi aşağıda göstərilib.

```

public class OrderRestaurantListActivity extends AppCompatActivity {

    ProgressBar progressBar;
    RecyclerView recyclerView;
    DatabaseReference databaseReference;
    ArrayList<RestaurantsModel> list;
    RecyclerViewAdapterRestaurants_Ord myAdapter;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_order_restaurant_list);

        recyclerView = (RecyclerView) findViewById(R.id.id_rey_recycleview);
        progressBar = findViewById(R.id.id_rey_progress);
        recyclerView.setHasFixedSize(true);
        recyclerView.setLayoutManager(new LinearLayoutManager(context: OrderRestaurantListActivity.this));

        databaseReference = FirebaseDatabase.getInstance().getReference(path: "RestoranSifaris");
        list = new ArrayList<RestaurantsModel>();

        myAdapter = new RecyclerViewAdapterRestaurants_Ord(MyContext: this, list);
        recyclerView.setAdapter(myAdapter);
        recyclerView.setLayoutManager(new GridLayoutManager(context: this, spanCount: 2));

        databaseReference.addValueEventListener(new ValueEventListener() {
            @Override
            public void onDataChange(@NonNull DataSnapshot snapshot) {
                list.clear();
                for (DataSnapshot dataSnapshot: snapshot.getChildren()){

```

```

                switch (item.getItemId()){
                    case R.id.menu_search:

                        SearchView searchView = (SearchView) item.getActionView();
                        searchView.setQueryHint("Axtar");

                        searchView.setOnQueryTextListener(new SearchView.OnQueryTextListener() {
                            @Override
                            public boolean onQueryTextSubmit(String query) {return false;}

                            @Override
                            public boolean onQueryTextChange(String newText) {
                                searchdata(newText);
                                return true;
                            }
                        });
                        return true;
                }
            }
        });
        return super.onOptionsItemSelected(item);
    }

    private void searchdata(String newText) {

        ArrayList<RestaurantsModel> searchlist = new ArrayList<>();
        for (RestaurantsModel model: list){

            String ad = model.getAd();
            if (ad.toLowerCase().contains(newText.toLowerCase())){
                searchlist.add(model);
            }
        }
        myAdapter.searchlist(searchlist);
    }
}

```

İstifadəçi məlumatları ekranının kodlarının bir hissəsi aşağıda göstərilib.

```
public class UserActivity extends AppCompatActivity {

    EditText ad, soyad, nomre, unvan;
    Button btn;
    SharedPreferences sharedPreferences = new SharedPreferences();

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_user);
        getSupportActionBar().hide();

        ad = findViewById(R.id.id_user_ad);
        soyad = findViewById(R.id.id_user_soyad);
        nomre = findViewById(R.id.id_user_nomre);
        btn = findViewById(R.id.id_user_button);
        unvan = findViewById(R.id.id_user_unvan);

        ad.setText(sharedPreferences.infoRead(context: UserActivity.this, key: "ad"));
        soyad.setText(sharedPreferences.infoRead(context: UserActivity.this, key: "soyad"));
        nomre.setText(sharedPreferences.infoRead(context: UserActivity.this, key: "nomre"));
        unvan.setText(sharedPreferences.infoRead(context: UserActivity.this, key: "unvan"));

        btn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
```

```
                String txt_ad = ad.getText().toString();
                String txt_soyad = soyad.getText().toString();
                String txt_nomre = nomre.getText().toString();
                String txt_unvan = unvan.getText().toString();

                if (txt_ad.isEmpty()){
                    ad.setError("Boş saxlanıla bilməz");
                    return;
                }

                if (txt_soyad.isEmpty()){
                    soyad.setError("Boş saxlanıla bilməz");
                    return;
                }

                if (txt_nomre.isEmpty()){
                    nomre.setError("Boş saxlanıla bilməz");
                    return;
                }

                if (txt_unvan.isEmpty()){
                    unvan.setError("Boş saxlanıla bilməz");
                    return;
                }

                sharedPreferences.infoWrite(context: UserActivity.this, key: "ad", txt_ad);
                sharedPreferences.infoWrite(context: UserActivity.this, key: "soyad", txt_soyad);
                sharedPreferences.infoWrite(context: UserActivity.this, key: "nomre", txt_nomre);
                sharedPreferences.infoWrite(context: UserActivity.this, key: "unvan", txt_unvan);
                Toast.makeText(context: UserActivity.this, text: "Dəyişikliklər yadda saxlanıldı", Toast.LENGTH_SHORT).show();
            }
        });
```

Sifarişlər ekranının kodlarının bir hissəsi aşağıda göstərilib.

```
public class SifarisActivity extends AppCompatActivity {
    ProgressBar progressBar;
    RecyclerView recyclerView;
    DatabaseReference databaseReference;
    ArrayList<SifarisModel> list;
    RecyclerViewAdapterSifarisler myAdapter;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_sifaris);

        FirebaseUser user;
        FirebaseAuth auth;
        auth = FirebaseAuth.getInstance();
        user = auth.getCurrentUser();

        // setTitle("Tədqiqatçılar");
        recyclerView = (RecyclerView) findViewById(R.id.id_rey_recycleview);
        progressBar = findViewById(R.id.id_rey_progress);
        recyclerView.setHasFixedSize(true);
        recyclerView.setLayoutManager(new LinearLayoutManager(context, SifarisActivity.this));

        databaseReference = FirebaseDatabase.getInstance().getReference(path: "RestoranSifaris").child("HQA84phB2Qb3yLc");
        list = new ArrayList<SifarisModel>();

        myAdapter = new RecyclerViewAdapterSifarisler(MyContext: this, list);
        recyclerView.setAdapter(myAdapter);
        recyclerView.setLayoutManager(new GridLayoutManager(context, this, spanCount: 1));
    }
}
```

```
databaseReference.addValueEventListener(new ValueEventListener() {
    @Override
    public void onDataChange(@NonNull DataSnapshot snapshot) {
        list.clear();
        for (DataSnapshot dataSnapshot: snapshot.getChildren()){
            SifarisModel models = dataSnapshot.getValue(SifarisModel.class);
            list.add(models);
        }
        myAdapter.notifyDataSetChanged();
        progressBar.setVisibility(View.GONE);
    }

    @Override
    public void onCancelled(@NonNull DatabaseError error) {
        Toast.makeText(context, SifarisActivity.this, error.getMessage(), Toast.LENGTH_SHORT).show();
    }
});
}
```

Müştəri rəyləri ekranının kodlarının bir hissəsi aşağıda göstərilib.

```

import android.os.Bundle;
import android.view.View;
import android.widget.ProgressBar;
import android.widget.Toast;

import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.auth.FirebaseUser;
import com.google.firebase.database.DataSnapshot;
import com.google.firebase.database.DatabaseError;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.database.ValueEventListener;

import java.util.ArrayList;

import apps.anar.restaurant.adapter.RecyclerViewAdapterReyler;
import apps.anar.restaurant.adapter.RecyclerViewAdapterSifarisler;
import apps.anar.restaurant.model.ReyModel;
import apps.anar.restaurant.model.SifarisModel;

public class ReyActivity extends AppCompatActivity {
    ProgressBar progressBar;
    RecyclerView recyclerView;
    DatabaseReference databaseReference;
    ArrayList<ReyModel> list;
    RecyclerViewAdapterReyler myAdapter;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_rey);

        progressBar = findViewById(R.id.id_rey_progress);
        recyclerView.setHasFixedSize(true);
        recyclerView.setLayoutManager(new LinearLayoutManager(context, RecyclerViewAdapterReyler.this));

        databaseReference = FirebaseDatabase.getInstance().getReference("RestoranSifaris").child("HQ484phB2Qb3y");
        list = new ArrayList<ReyModel>();

        myAdapter = new RecyclerViewAdapterReyler(MyContext.this, list);
        recyclerView.setAdapter(myAdapter);
        recyclerView.setLayoutManager(new GridLayoutManager(context, this, spanCount: 1));

        databaseReference.addValueEventListener(new ValueEventListener() {
            @Override
            public void onDataChange(@NonNull DataSnapshot snapshot) {
                list.clear();
                for (DataSnapshot dataSnapshot: snapshot.getChildren()){
                    ReyModel models = dataSnapshot.getValue(ReyModel.class);
                    list.add(models);
                }
                myAdapter.notifyDataSetChanged();
                progressBar.setVisibility(View.GONE);
            }
        });

        @Override
        public void onCancelled(@NonNull DatabaseError error) {
            Toast.makeText(context, ReyActivity.this, error.getMessage(), Toast.LENGTH_SHORT).show();
        }
    }
};

```

Menu model sinfinin kodları aşağıda göstərilib.

```
public class MenuModel {

    String sekil, ad, terkibi;
    int qiymet;

    public MenuModel() {
    }

    public MenuModel(String sekil, String ad, String terkibi, int qiymet) {
        this.sekil = sekil;
        this.ad = ad;
        this.terkibi = terkibi;
        this.qiymet = qiymet;
    }

    public String getSekil() { return sekil; }

    public void setSekil(String sekil) { this.sekil = sekil; }

    public String getAd() { return ad; }

    public void setAd(String ad) { this.ad = ad; }

    public String getTerkibi() { return terkibi; }

    public void setTerkibi(String terkibi) { this.terkibi = terkibi; }

    public int getQiymet() { return qiymet; }

    public void setQiymet(int qiymet) { this.qiymet = qiymet; }
}
```

Restoran model sinfinin kodları aşağıda göstərilib.

```
public RestaurantsModel(String restoran_id, String ad, String haqqinda, String elaqe, String instagram, String menu, String mekan, String profil_sekil, String hesap_novu, String mekan_soz, String xidmet_haqqi) {
    this.restoran_id = restoran_id;
    this.ad = ad;
    this.haqqinda = haqqinda;
    this.elaqe = elaqe;
    this.instagram = instagram;
    this.menu = menu;
    this.mekan = mekan;
    this.profil_sekil = profil_sekil;
    this.hesab_novu = hesap_novu;
    this.mekan_soz = mekan_soz;
    this.xidmet_haqqi = xidmet_haqqi;
}

public String getRestoran_id() { return restoran_id; }

public void setRestoran_id(String restoran_id) { this.restoran_id = restoran_id; }

public String getAd() { return ad; }

public void setAd(String ad) { this.ad = ad; }

public String getHaqqinda() { return haqqinda; }

public void setHaqqinda(String haqqinda) { this.haqqinda = haqqinda; }

public String getElaqe() { return elaqe; }

public void setElaqe(String elaqe) { this.elaqe = elaqe; }

public String getInstagram() { return instagram; }
}
```

Menyuya uygun RecyclerView sınıfının kodları aşağıda gösterilib.

```

public class RecyclerViewAdapterMenu extends RecyclerView.Adapter<RecyclerViewAdapterMenu.MyViewHolder> {

    private List<MenuModel> list;
    private Context MyContext;

    ArrayList<SebetModel> listsebet = new ArrayList<>();

    public RecyclerViewAdapterMenu(Context MyContext, List<MenuModel> list) {
        this.MyContext = MyContext;
        this.list = list;
    }

    @NonNull
    @Override
    public MyViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
        View v = LayoutInflater.from(MyContext).inflate(R.layout.cardview_menu, parent, attachToRoot: false);

        return new MyViewHolder(v);
    }

    @Override
    public void onBindViewHolder(@NonNull MyViewHolder holder, @SuppressWarnings("RecyclerView") final int position) {

        MenuModel menuModel = list.get(position);
        holder.ad.setText(menuModel.getAd());
        holder.terkib.setText(menuModel.getTerkibi());
        holder.qiymet.setText(Integer.toString(menuModel.getQiymet()));
        Glide.with(MyContext).load(menuModel.getSekil()).into(holder.imageView);

        qiymet.setText(Integer.toString(menuModel.getQiymet()));
        ad.setText(menuModel.getAd());

        plus.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                int int_say = Integer.parseInt(say.getText().toString());
                say.setText(Integer.toString(int_say+1));
                qiymet.setText(Integer.toString(int_say*menuModel.getQiymet()));
            }
        });

        minus.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                int int_say = Integer.parseInt(say.getText().toString());
                say.setText(Integer.toString(int_say-1));
                qiymet.setText(Integer.toString(int_say*menuModel.getQiymet()));
            }
        });

        elaveet.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                listsebet.add(new SebetModel(menuModel.getAd(), menuModel.getSekil(), Integer.parseInt(say.getText().toString())));
                bottomSheetDialog.dismiss();
                Toast.makeText(MyContext, text: "Sebete əlavə edildi", Toast.LENGTH_SHORT).show();
            }
        });

        bottomSheetDialog.setContentView(sheetview);
        bottomSheetDialog.show();
    }
}

```

Restoran modelinə uyğun RecyclerView sinfinin kodları aşağıda göstərilib.

```
public class RecyclerViewAdapterRestaurants_Ord extends RecyclerView.Adapter<RecyclerViewAdapterRestaurants_Ord.MyViewH

    private List<RestaurantsModel> list;
    private Context MyContext;

    public RecyclerViewAdapterRestaurants_Ord(Context MyContext, List<RestaurantsModel> list) {
        this.MyContext = MyContext;
        this.list = list;
    }

    @NonNull
    @Override
    public MyViewHolder onCreateViewHolder(@NonNull ViewGroup parent, int viewType) {
        View v = LayoutInflater.from(MyContext).inflate(R.layout.cardview_restaurant, parent, attachToRoot: false);

        return new MyViewHolder(v);
    }

    @Override
    public void onBindViewHolder(@NonNull MyViewHolder holder, @SuppressWarnings("RecyclerView") final int position) {

        RestaurantsModel model = list.get(position);
        holder.name.setText(model.getAd());
        Glide.with(MyContext).load(model.getProfil_sekil()).into(holder.imageView);
    }
}
```

```
        intent.putExtra(name: "rest_id", model.getRestoran_id());
        MyContext.startActivity(intent);
    }

    });
}

@Override
public int getItemCount() { return list.size(); }

public void searchlist(ArrayList<RestaurantsModel> searchlist){
    list = searchlist;
    notifyDataSetChanged();
}

public static class MyViewHolder extends RecyclerView.ViewHolder{

    TextView name;
    CardView cardView;
    ImageView imageView;

    public MyViewHolder(@NonNull View itemView) {
        super(itemView);
        name = (TextView) itemView.findViewById(R.id.id_rest_cardview_ad);
        imageView = (ImageView) itemView.findViewById(R.id.id_rest_cardview_image);
        cardView = (CardView) itemView.findViewById(R.id.id_rest_cardview);
        cardView.setBackgroundResource(R.drawable.back_cardview_restaurant );
        cardView.setRadius(20);
    }
}
```


2.3 Mobil tətbiq üçün Firebase məlumat bazasının hazırlanması

Google Firebase, Google tərəfindən hazırlanan və istifadəçilərə qurğularında, veb və mobil tətbiqlərdən pulsuz istifadə etmək üçün hazırlanan bir bulut kompüter platformasıdır. Firebase, istifadəçilərə tətbiq inkişafı üçün məlumat bazası, serverless işlətmə, istifadəçi autentifikasiyası, depolama, anlık bildirişlər və daha bir çox xidmətlər təklif edir.

Firebase, hazırda mobil tətbiqlərin inkişaf prosesini sadələşdirmək və server pərəstişkarlığına ehtiyac qoymadan çox funksional və performanslı tətbiqlər yaratmağa imkan verir. Aşağıdakı əsas xidmətləri təklif edir:

ə

irebase Authentication: İstifadəçilərin tətbiqə daxil olmaq üçün e-poçt, sosial media hesabları və ya telefon nömrələri ilə autentifikasiya etməyə imkan verir.

irebase Cloud Messaging: Mobil cihazlara anlık bildirişlər göndərmək üçün

tətbiqlərinizi vebdə paylaşmaq üçün həssas və sürətli veb hosting təklif edir.

irebase Storage: İstifadəçilərinizin şəkillər, videolar və digər faylları yükləməsi

6. Firebase Cloud Functions: Server pərəstişkarlığı olmadan serverless funksiyonlar yazmaq və çalışdırmaq üçün bir platformadır.

Bu yalnız bir neçə Firebase xidmətləri nümunələridir. Firebase, inkişafçıların

1

Z

Tətbiqdəki bütün məlumatlar, hesaba giriş, çıxış məlumatları Google firebase-də saxlanılır. Google firebase mobil tətbiqlə əlaqələndirilməsi üçün aşağıdakı kodlardan istifadə olunur.

ə

ə

ə

ə

a

n

1

```
dependencies {
    implementation 'androidx.appcompat:appcompat:1.6.1'
    implementation 'com.google.android.material:material:1.8.0'
    implementation 'androidx.constraintlayout:constraintlayout:2.1.4'
    implementation 'com.google.firebase:firebase-database-ktx:20.2.0'
    implementation 'com.google.firebase:firebase-auth:21.3.0'
    implementation 'com.google.firebase:firebase-auth-ktx:21.3.0'
    implementation 'com.google.firebase:firebase-storage-ktx:20.1.0'
    testImplementation 'junit:junit:4.13.2'
    androidTestImplementation 'androidx.test.ext:junit:1.1.5'
    androidTestImplementation 'androidx.test.espresso:espresso-core:3.5.1'

    implementation 'com.github.bumptech.glide:glide:4.14.2'
    annotationProcessor 'com.github.bumptech.glide:compiler:4.14.2'
    implementation 'de.hdodenhof:circleimageview:3.1.0'
    implementation 'de.hdodenhof:circleimageview:3.1.0'
}
```

Profil şəkillərinin firebase -ə əlavə olunması və dəyişdirilməsi üçün kod aşağıdakı kimi olacaqdır.

```
@Override
public void onClick(View view) {
    mydialog = new Dialog(context, AccountActivity.this);
    mydialog setContentView(R.layout.profile_picture);
    mydialog.setCancelable(true);
    Button saveBtn = mydialog.findViewById(R.id.id_profile_change_btn);

    profileChangeimageView = mydialog.findViewById(R.id.id_profile_change_pic);
    progressBarProfile = mydialog.findViewById(R.id.id_profile_change_progress);

    profileChangeimageView.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            Intent photoPicker = new Intent();
            photoPicker.setAction(Intent.ACTION_GET_CONTENT);
            photoPicker.setType("image/*");
            activityResultLauncher.launch(photoPicker);
        }
    });

    saveBtn.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            if (MyimageUri != null) {
                uploadToFirebase(MyimageUri);
            } else {
                Toast.makeText(context, AccountActivity.this, text: "Zəhmət olmasa səkil seçin", Toast.LENGTH_SHORT).s
            }
        }
    });

    mydialog.show();
}
```

Məlumatların firebase -dən tətbiqə göndərilməsi üçün kod aşağıdakı kimi olacaqdır.

```

auth = FirebaseAuth.getInstance();

user = auth.getCurrentUser();
reference = FirebaseDatabase.getInstance().getReference().child("Restoranlar");
reference.child(user.getUid()).addValueEventListener(new ValueEventListener() {
    @Override
    public void onDataChange(@NonNull DataSnapshot snapshot) {

        RestaurantsModel models = snapshot.getValue(RestaurantsModel.class);
        Glide.with(activity: AccountActivity.this).load(models.getProfil_sekil()).into(imageView);

        sharedPreferences.infoWriteBusiness(context: AccountActivity.this, key: "hesab_novu", models.getHesab_novu());
        sharedPreferences.infoWriteBusiness(context: AccountActivity.this, key: "restID", models.getRestoran_id());
        ad.setText(models.getAd());
        haqqinda.setText(models.getHaqqinda());
        nomre.setText(models.getElage());
        instagram.setText(models.getInstagram());
        menu.setText(models.getMenu());
        mekan.setText(models.getMekan());
        unvan_soz.setText(models.getMekan_soz());
        xidmet_haqqi.setText(Integer.toString(models.getXidmet_haqqi()));
    }
    @Override
    public void onCancelled(@NonNull DatabaseError error) {
        Toast.makeText(context: AccountActivity.this, error.getMessage(), Toast.LENGTH_SHORT).show();
    }
});

```

Məlumatların firebase vasitəsilə yenilənməsi üçün kod aşağıdakı kimi olacaqdır.

```

progressBar.setVisibility(View.VISIBLE);

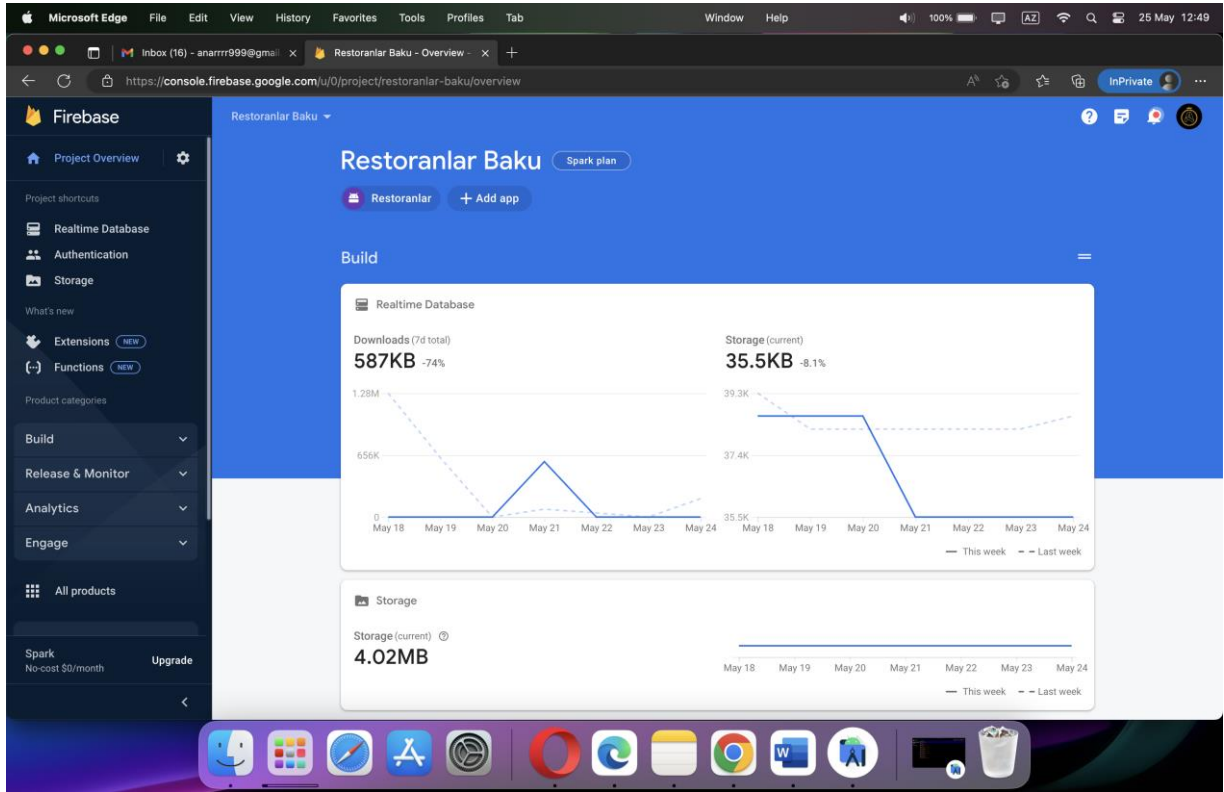
reference.child(user.getUid()).addListenerForSingleValueEvent(new ValueEventListener() {
    @Override
    public void onDataChange(@NonNull DataSnapshot snapshot) {
        RestaurantsModel models = snapshot.getValue(RestaurantsModel.class);

        HashMap<String, Object> map = new HashMap<>();
        map.put("ad", text_ad);
        map.put("haqqinda", text_haqqinda);
        map.put("elage", text_nomre);
        map.put("instagram", text_instagram);
        map.put("menu", text_menu);
        map.put("mekan", text_mekan);
        map.put("mekan_soz", text_unvan_soz);
        map.put("xidmet_haqqi", Integer.parseInt(text_xidmet_haqqi));

        reference.child(user.getUid()).updateChildren(map);
        Toast.makeText(context: AccountActivity.this, text: "Məlumatlar dəyişdirildi", Toast.LENGTH_SHORT).s
        progressBar.setVisibility(View.GONE);
    }
    @Override
    public void onCancelled(@NonNull DatabaseError error) {
        Toast.makeText(context: AccountActivity.this, error.getMessage(), Toast.LENGTH_SHORT).show();
    }
});

```

Restoranlar mobil tətbiqinə uyğun Firebase yaradılması aşağıdakı şəkildə göstərilib.



Şəkil 17. Google Firebase

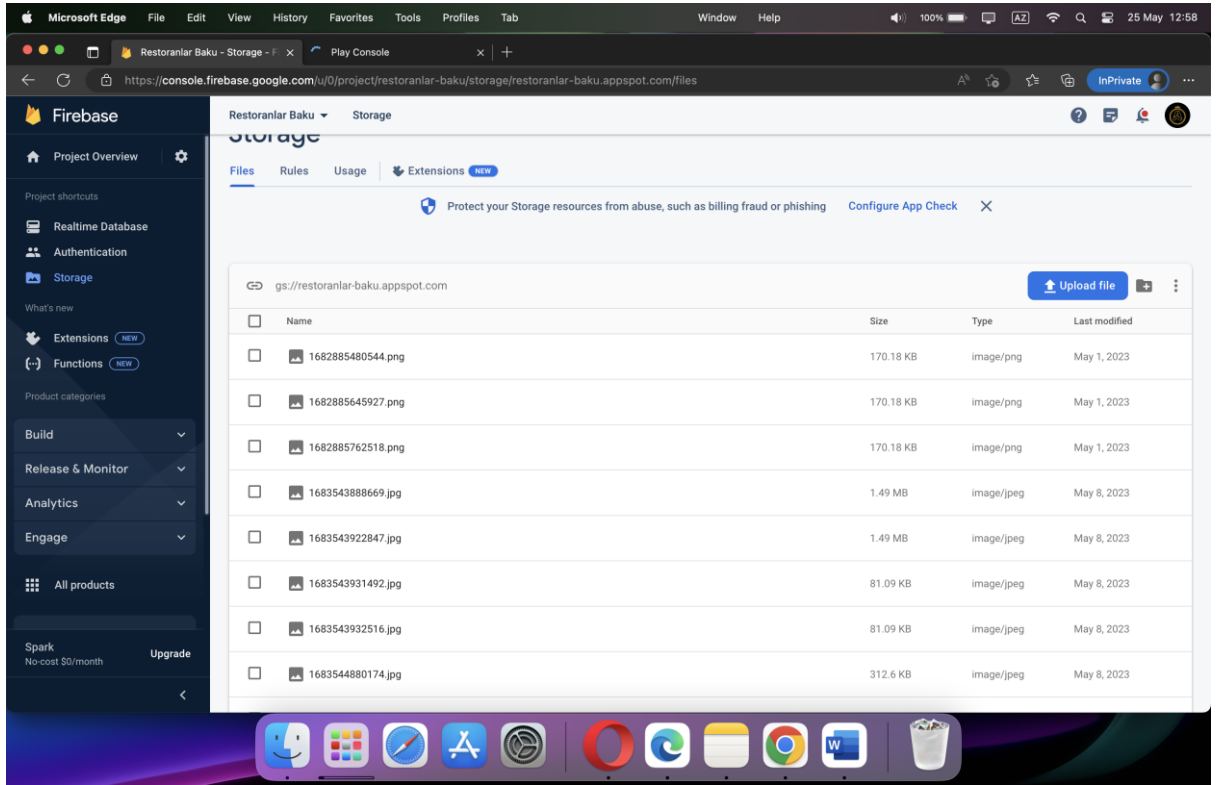
Restoranlar mobil tətbiqində qeydiyyatdan keçmiş istifadəçilər aşağıdakı şəkildə göstərilib. Burada istifadəçi ID və mailləridə göstərilib.

The screenshot shows the 'Authentication' page in the Firebase console. The 'Users' tab is selected, and a table lists the registered users:

Identifier	Providers	Created	Signed in	User UID
anar29364@gmail.com	📧	May 1, 2023	May 21, 2023	HQA84ph82Qb3yIyDGH0dnH2zdJ3

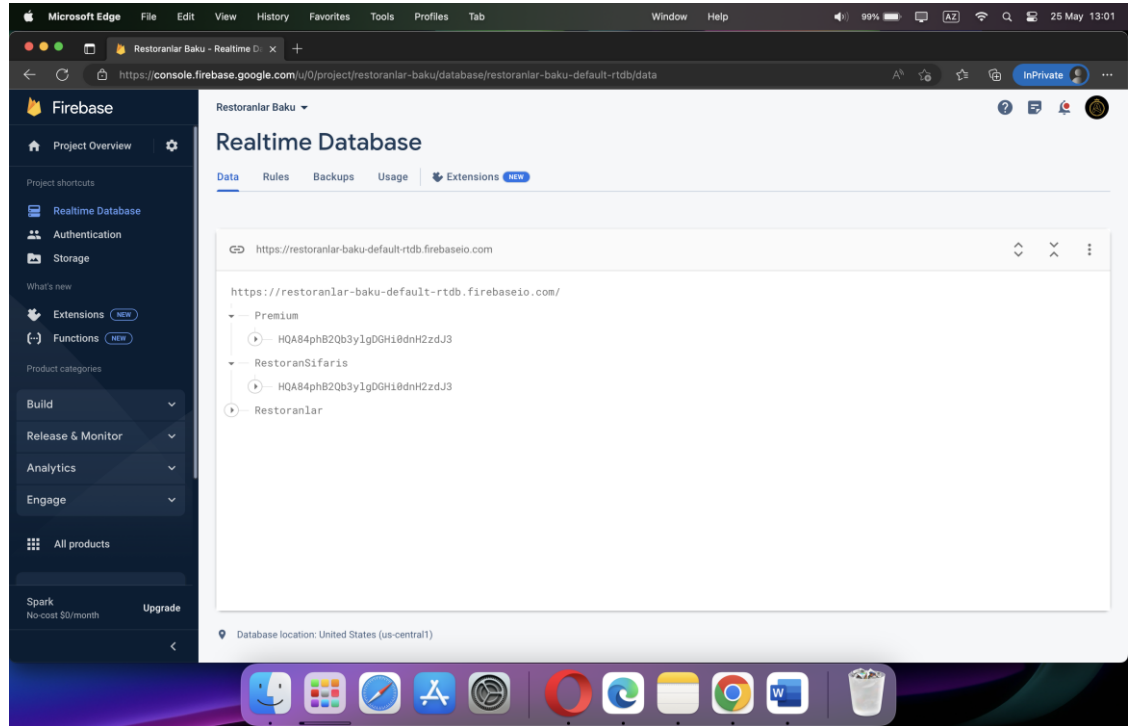
Şəkil 18. Google Firebaseda hesablar

Restoranlar mobil tətbiqində qeydiyyatdan keçmiş istifadəçilərin profil şəkillərinin Google Firebase necə depolanması aşağıdakı şəkildə göstərilib.



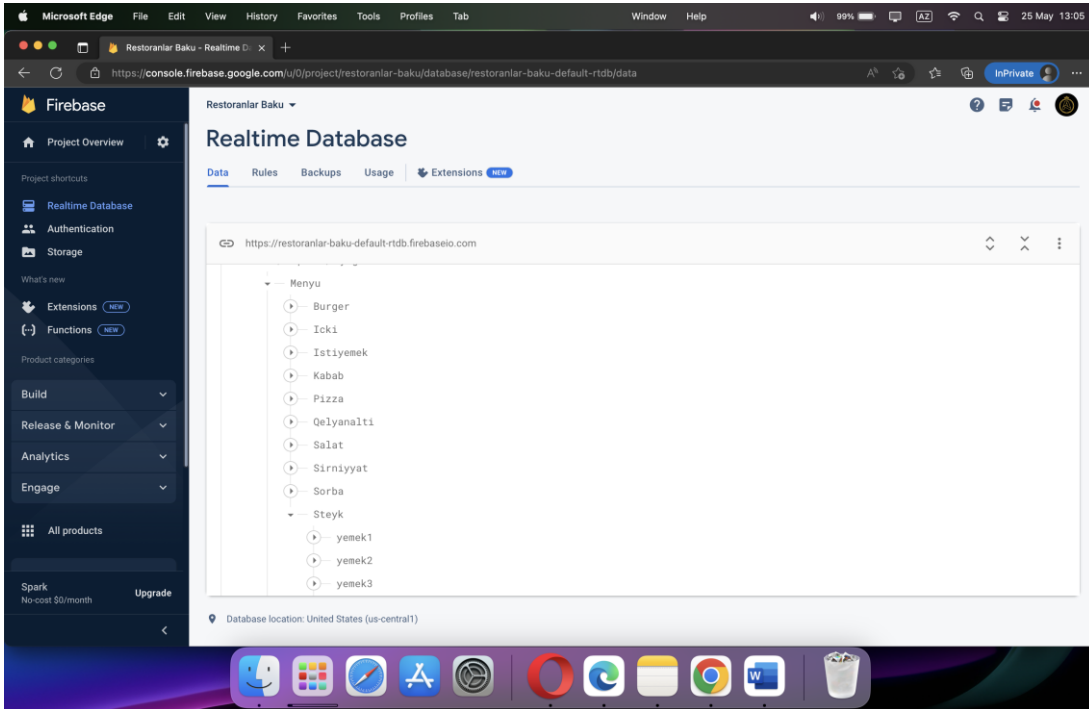
Şəkil 19. Google Firebasədə profil şəkillərinin depolanması

Google Firebasədə istifadəçilərin necə strukturlaşması aşağıdakı şəkildə göstərilib.



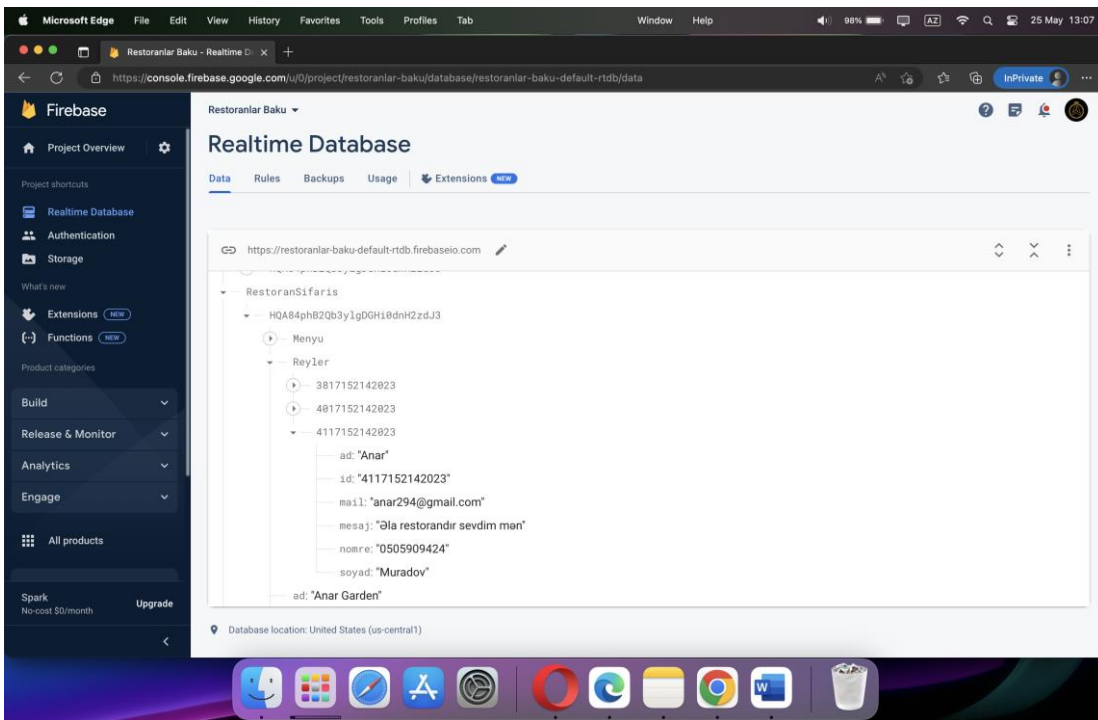
Şəkil 20. Google Firebase strukturu

Google Firebasedə restoran menyusunun necə saxlanması aşağıdakı şəkildə göstərilmişdir.



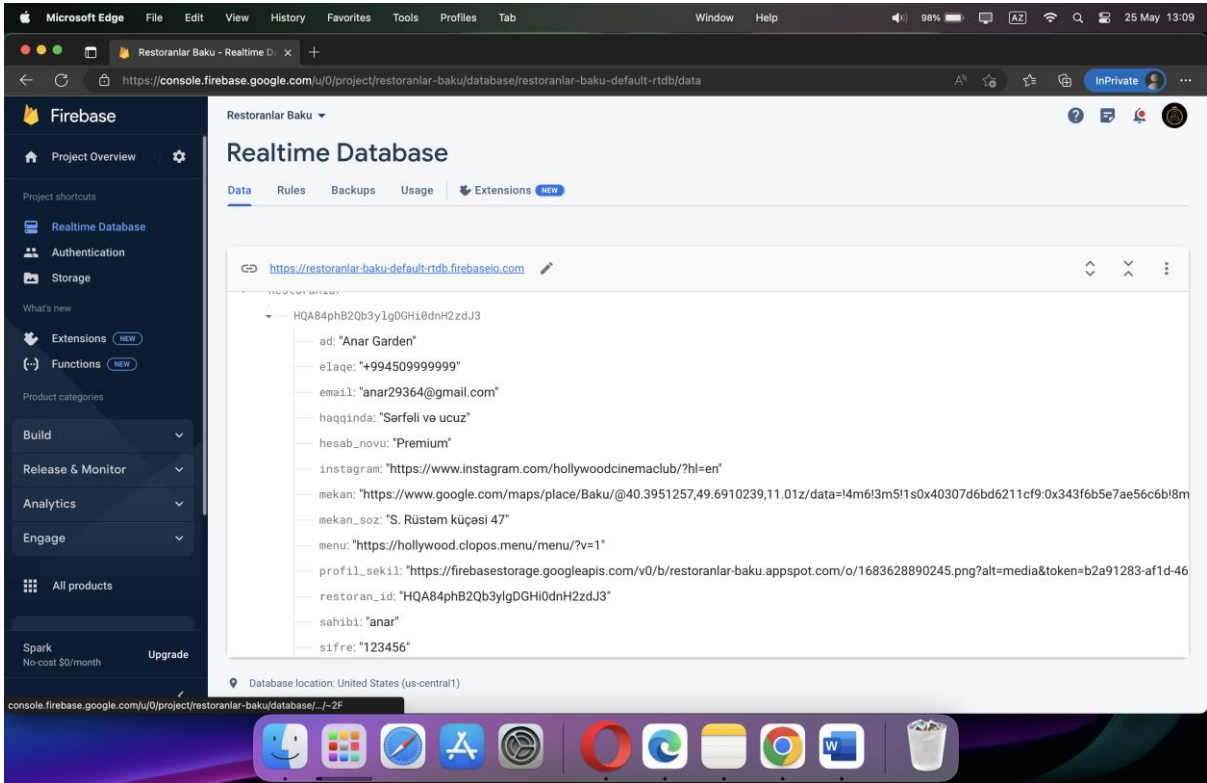
Şəkil 21. Google Firebase yemek meyuları

Google Firebasedə restorana istifadəçilər tərəfindən verilmiş rəylərin necə depolanması aşağıdakı şəkildə göstərilmişdir.



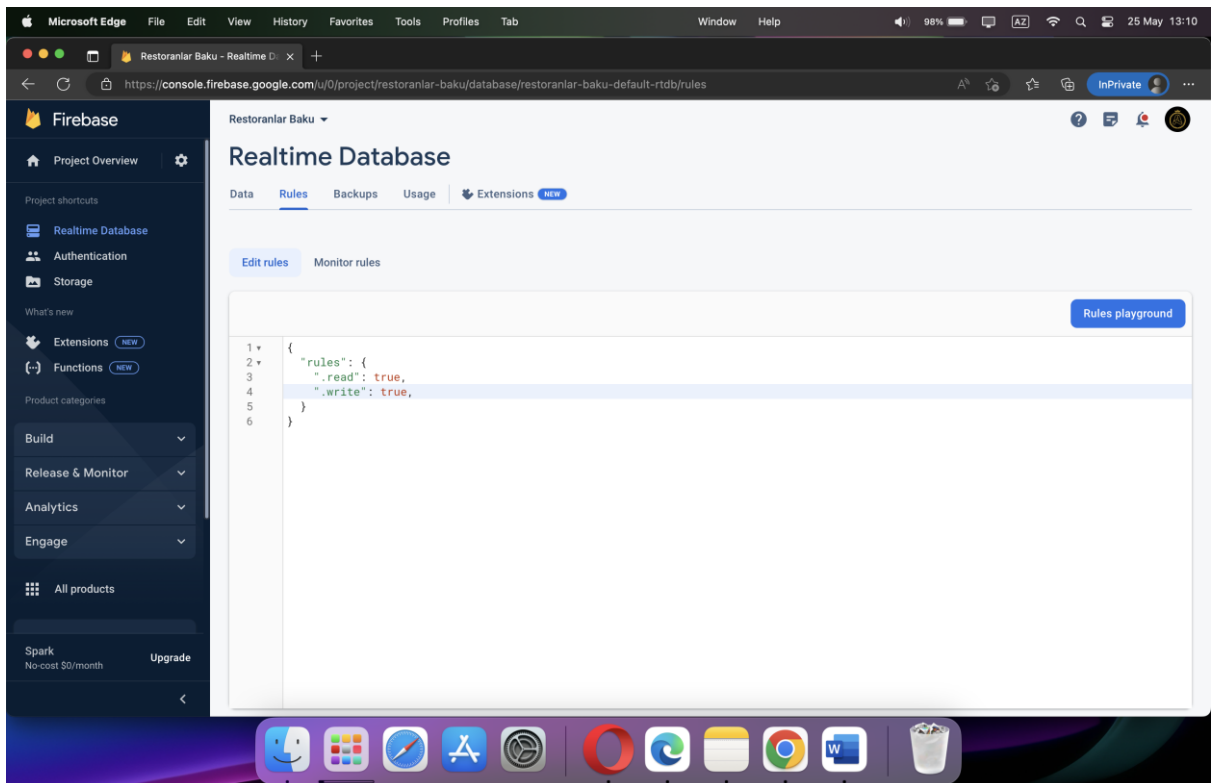
Şəkil 22. Google Firebase rəylər

Restoran məlumatları aşağıdakı kimi Google Firebasedə saxlanılır.



Şəkil 23. Google Firebase restoran məlumatları

Google Firebasedə qoyuluş qaydalar aşağıdakı kimidir.



Şəkil 24. Google Firebase təhlükəsizlik qaydaları

FƏSİL III

Mobil tətbiqin analizi və effektivliyi

3.1 Restoranlar mobil tətbiqinin əsas funksiyalarının təhlili

ə

Şəkil 25. Splash ekran

i

l

k

a

ç

ı

ə

n

b

i

z

i

S

p

l

a

s

h

S

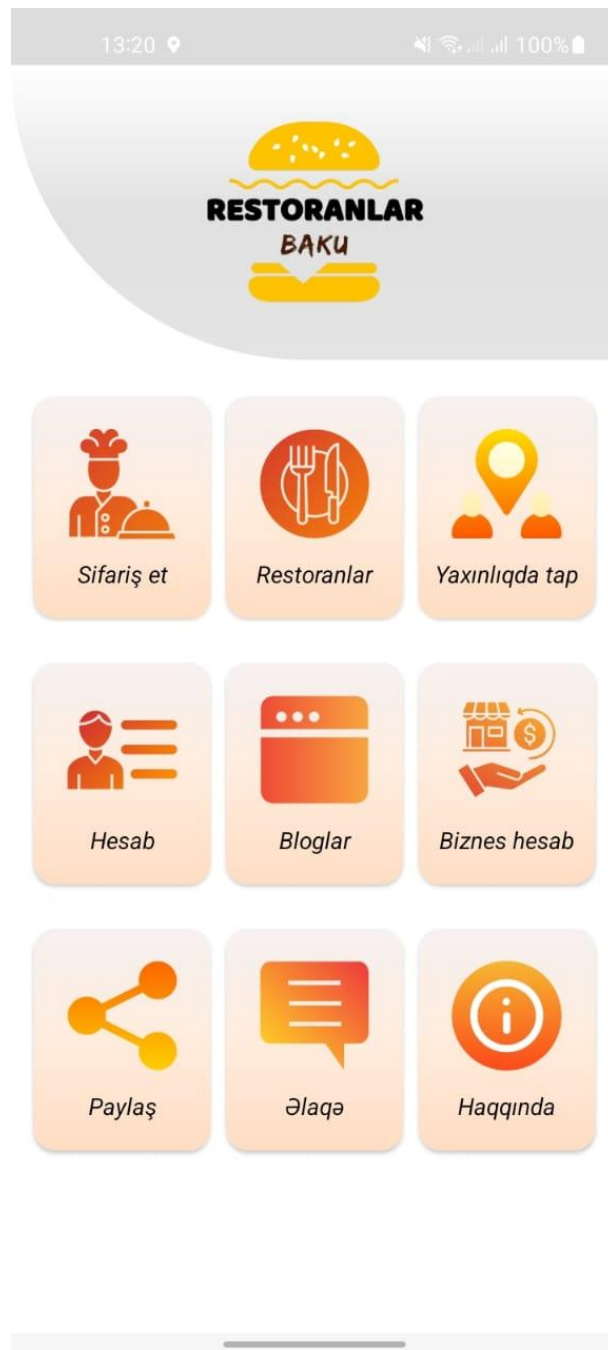
Əsas ekranda Sifariş et, Restoranlar, Yaxınlıqda restoran tap, Hesab, Bloglar, Biznes hesab, Paylaş, Əlaqə, Haqqında kimi menyular vardır.

e

e

n

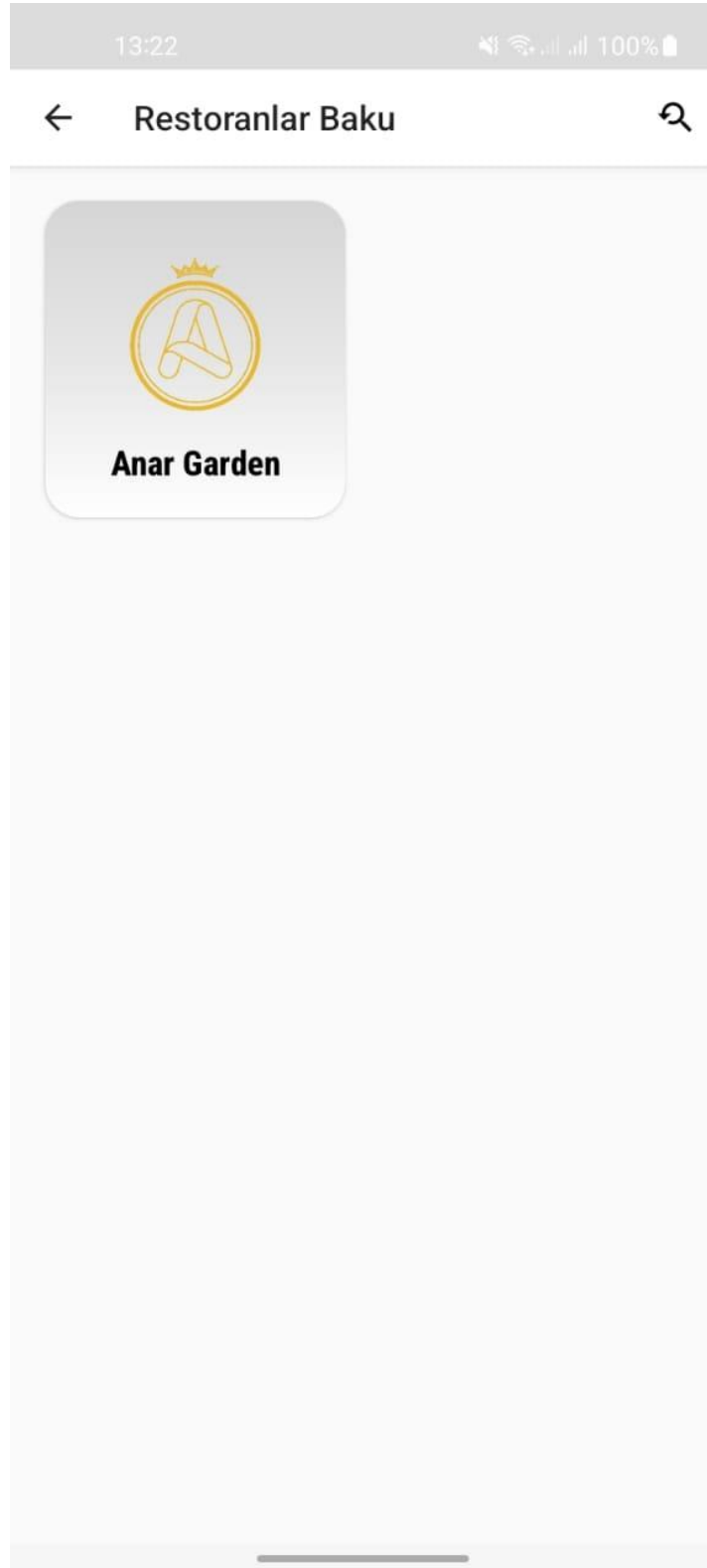




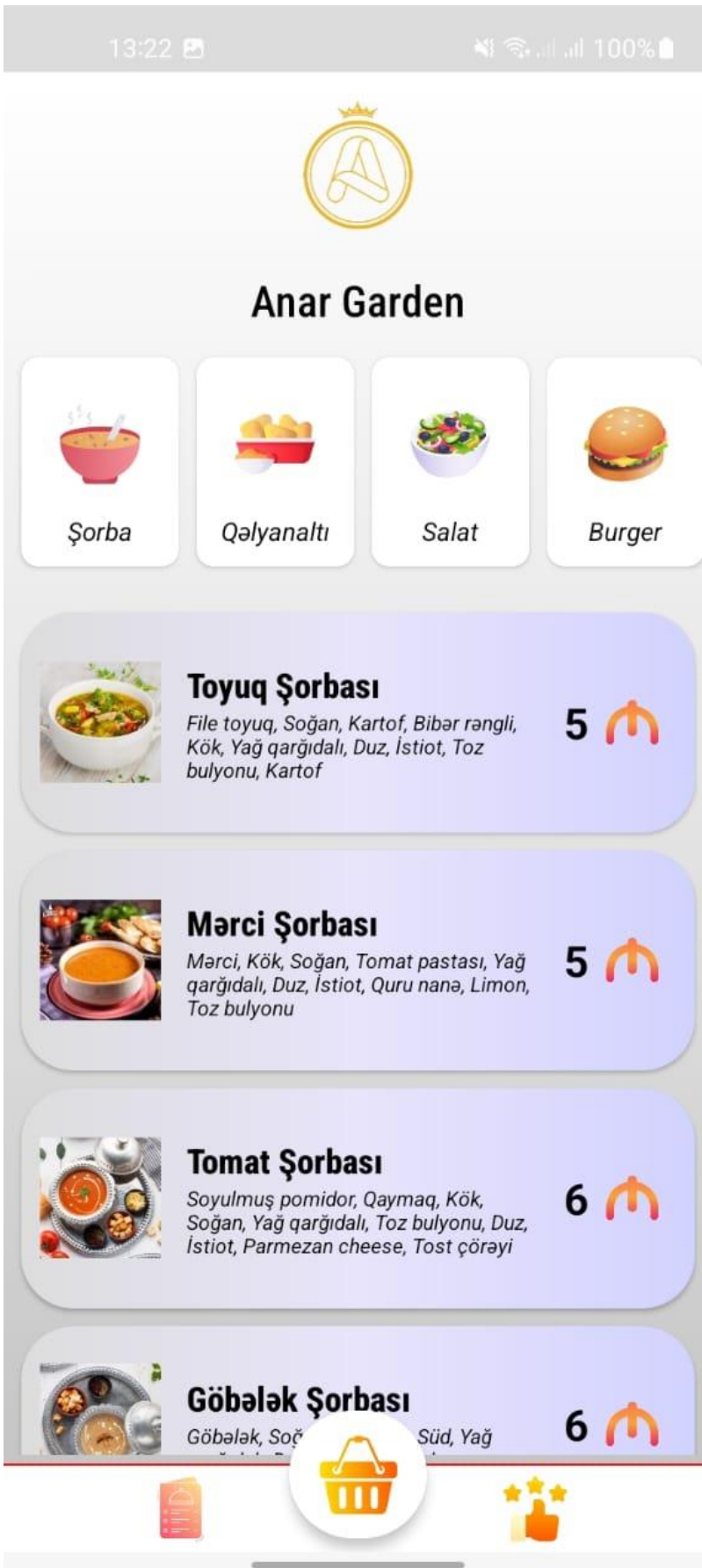
Şəkil 26. Əsas ekran

Sifariş et ekranında Premium hesab statusuna malik olan restoranlar yerləşdirilmişdir. Restoranlara daxil olaraq menyularla tanış ola və sifariş edə bilərsiniz.

İstifadəçi restoran seçdikdən sonra menyu ekranı açılacaqdır.



Şəkil 27. Restoran siyahısı ekranı



Şəkil 28. Menu ekranı

İstifadəçi rəy düyməsinə basdıqda isə restoran haqqında fikir bildirə bilər.



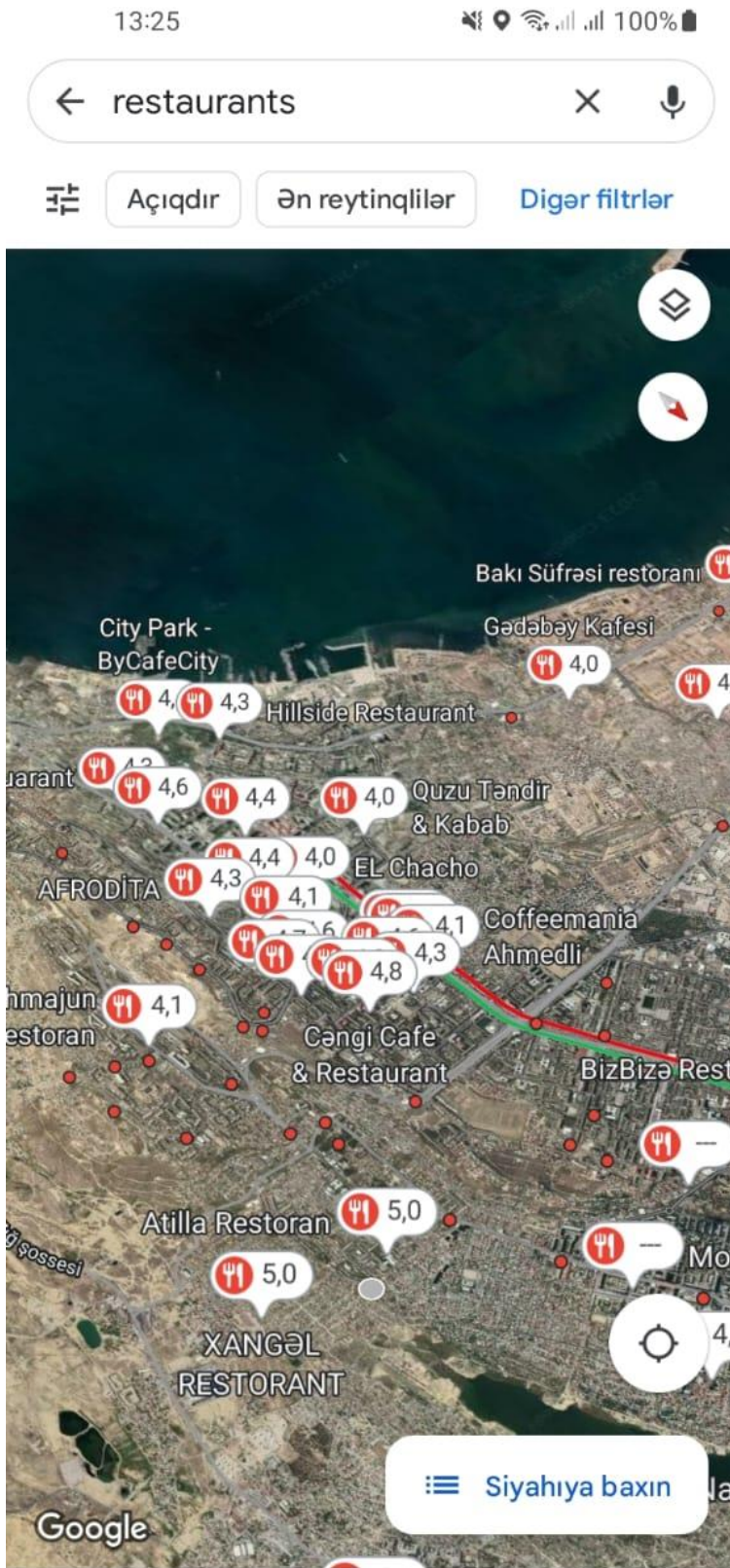
Şəkil 29. Rəy bildir ekranı

İstifadəçi haqqında düyməsinə kliklədikdə isə isə restoran haqqında məlumatları görə bilər.



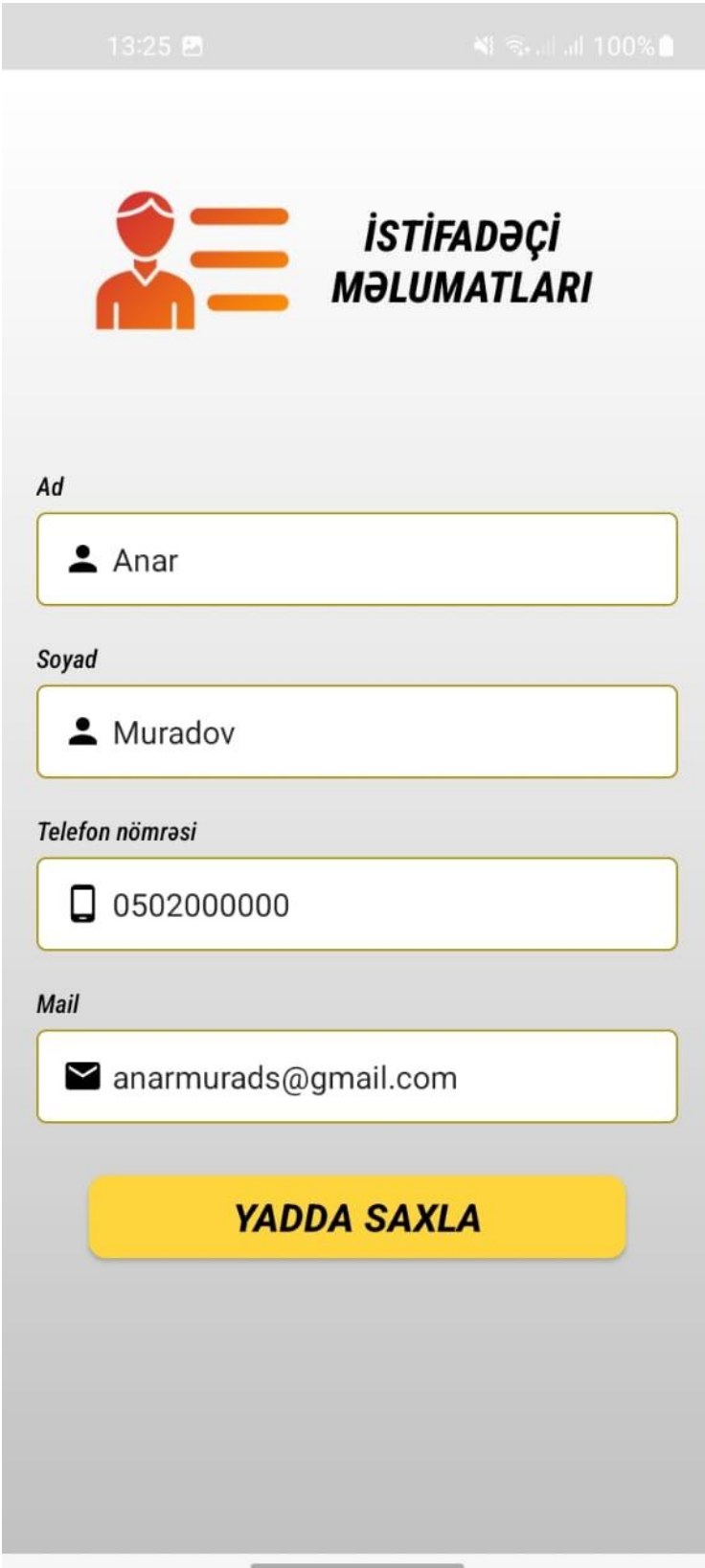
Şəkil 30. Restoran haqqında məlumat ekranı

Menyudan yaxınlıqda rstoran tap menyusuna kliklədikdə isə google mapda qeyd edilmiş və sizə yaxın olan restoranlar xəritədə sizə göstərilir.



Şəkil 30. Yaxınlıqda restoran tap ekranı

Müştəri məlumatları ekranında sifarişçinin məlumatları qeyd olunmalıdır.



The screenshot shows a mobile application interface for user registration. At the top, the status bar displays the time 13:25, signal strength, Wi-Fi, and 100% battery. The main header features an orange icon of a person with three horizontal lines to the right, followed by the text "İSTİFADƏÇİ MƏLUMATLARI" in bold black font. Below this, there are four input fields, each with a label and a corresponding icon: "Ad" (Name) with a person icon, "Soyad" (Surname) with a person icon, "Telefon nömrəsi" (Phone number) with a phone icon, and "Mail" (Email) with an envelope icon. The values entered in the fields are "Anar", "Muradov", "0502000000", and "anarmurads@gmail.com" respectively. At the bottom, there is a prominent yellow button with the text "YADDA SAXLA" (Save).

13:25 100%

**İSTİFADƏÇİ
MƏLUMATLARI**

Ad

Anar

Soyad

Muradov

Telefon nömrəsi

0502000000

Mail

anarmurads@gmail.com

YADDA SAXLA

Şəkil 31. İstifadəçi məlumatları ekranı

Restoran sahibləri menyudan qeydiyyat bölməsinə keçid edərək restoranlarını sistemə qeyd edə bilirlər. Bunun üçün göstərilən məlumatları düzgün doldurmaqları lazımdır.

13:25 100%

RESTORANLAR BAKU

QEYDİYYAT

Restoranın adı

Restoranın adını daxil edin

Restoranın haqqında

Məlumat daxil edin

Əlaqə nömrəsi

Əlaqə nömrəsini daxil edin

Instagram linki

Linki daxil edin

Menu linki

Linki daxil edin

Məkan linki (Google Map)

Linki daxil edin

Ünvani daxil edin

Xidmət haqqı (%)

% Daxil edin

Logo

Məkan sahibinin adı, soyadı

Ad, soyad daxil edin

Email

Email daxil edin

Şifrə təyin edin

Şifrəni daxil edin

Şifrəni təkrar daxil edin


Şifrəni daxil edin

HESAB YARAT

Hesabın var? Giriş et

Şəkil 32. Qeydiyyat ekranı

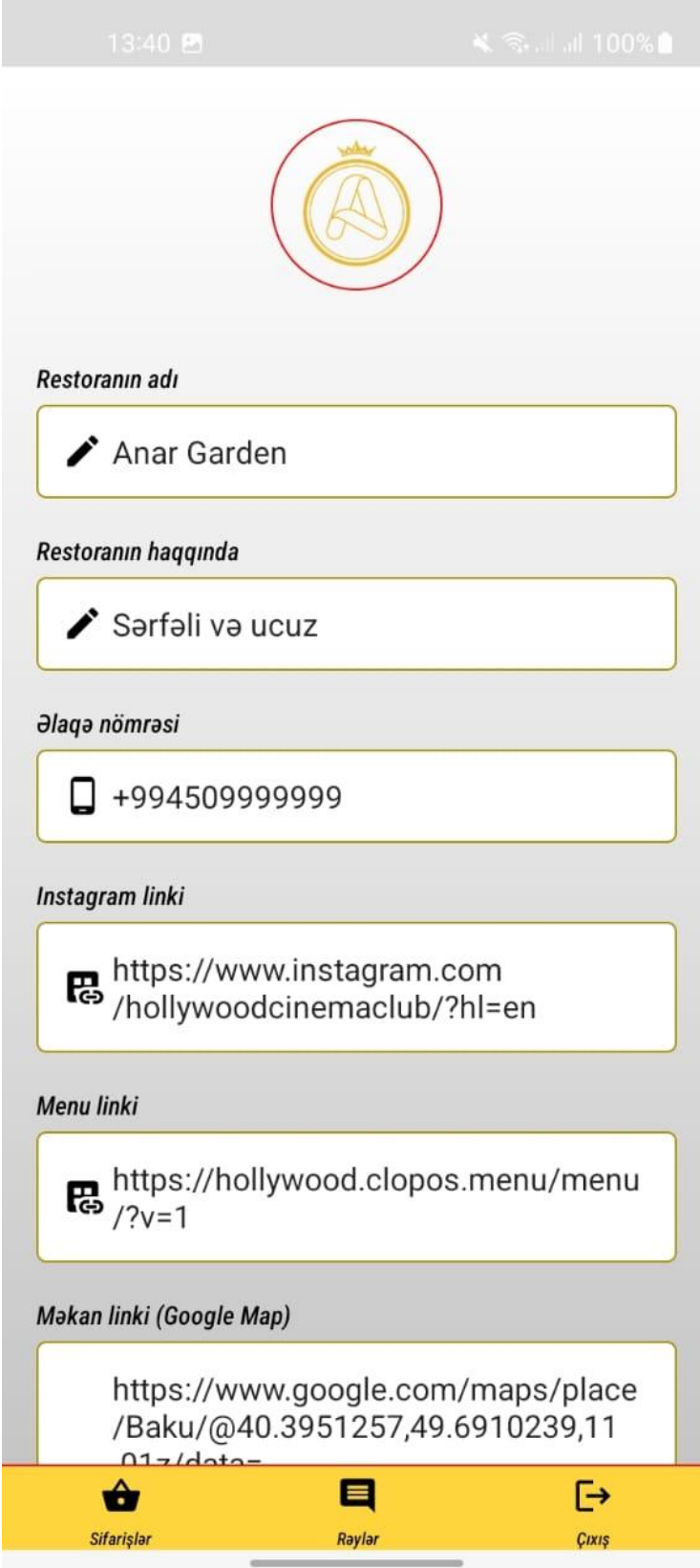
Hesaba daxil olmaq üçün email və şifrələrini yazmaq lazımdır. İlkin açılmış hesablar sadə hesab kimi qeyd edilir.




The image shows a mobile application login screen. At the top, the status bar displays the time 13:25, signal strength, Wi-Fi, and 100% battery. Below the status bar is the app's logo, which consists of a yellow burger icon with the text 'RESTORANLAR BAKU' underneath. The main heading is 'HESABA DAXİL OL' in bold black letters. There are two input fields: the first is labeled 'Email' and contains the placeholder text 'Email daxil edin' with an envelope icon; the second is labeled 'Şifrə' and contains the placeholder text 'Şifrə daxil edin' with a lock icon. Below the input fields is a large yellow button with the text 'GİRİŞ ET' in bold black letters. At the bottom, there is a link that says 'Hesabın yoxdur? Qeydiyyatdan keç'.

Şəkil 32. Hesaba giriş ekranı


Hesaba daxil olduqdan sonra idarəetmə paneli açılır və məlumatları dəyişdirmək mümkün olur.




13:40 100%




Restoranın adı

 Anar Garden


Restoranın haqqında

 Sərfəli və ucuz


Əlaqə nömrəsi

 +994509999999

Instagram linki


 <https://www.instagram.com/hollywoodcinemaclub/?hl=en>


Menu linki


 <https://hollywood.clopos.menu/menu/?v=1>

Məkan linki (Google Map)

<https://www.google.com/maps/place/Baku/@40.3951257,49.6910239,1101z/data=>

 Sifarişlər

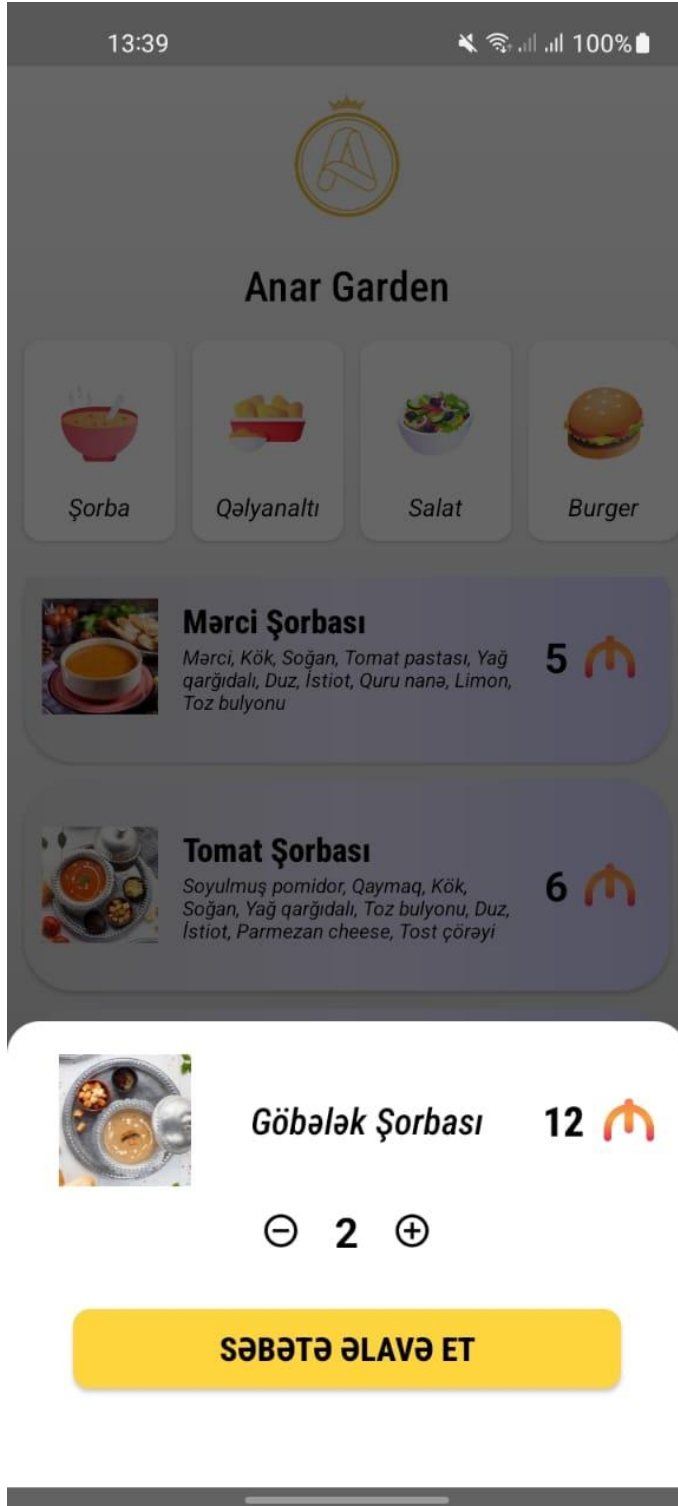
 Raylar

 Çıxış

Şəkil 33. İdarəetmə ekranı

3.2 Sifarişlərin sərbətə əlavə olunması və qiymətləndirilməsi

İstifadəçi yemək növlərindən istədiyini seçə və sərbətə əlavə edə bilər. Yemək növünün üzərinə bəzə sayı tənzimləyə bilər. Bu zaman qiymətlər avtomatik olaraq sərə uyğun hesablanır.



Şəkil 34. Sərbətə əlavə etmək

Səbət düyməsinə kliklədikdə əlavə olunmuş məhsullar görünəcəkdir. Həm sayı azaltmaq, artırmaq həm də məhsulu silmək mümkündür.



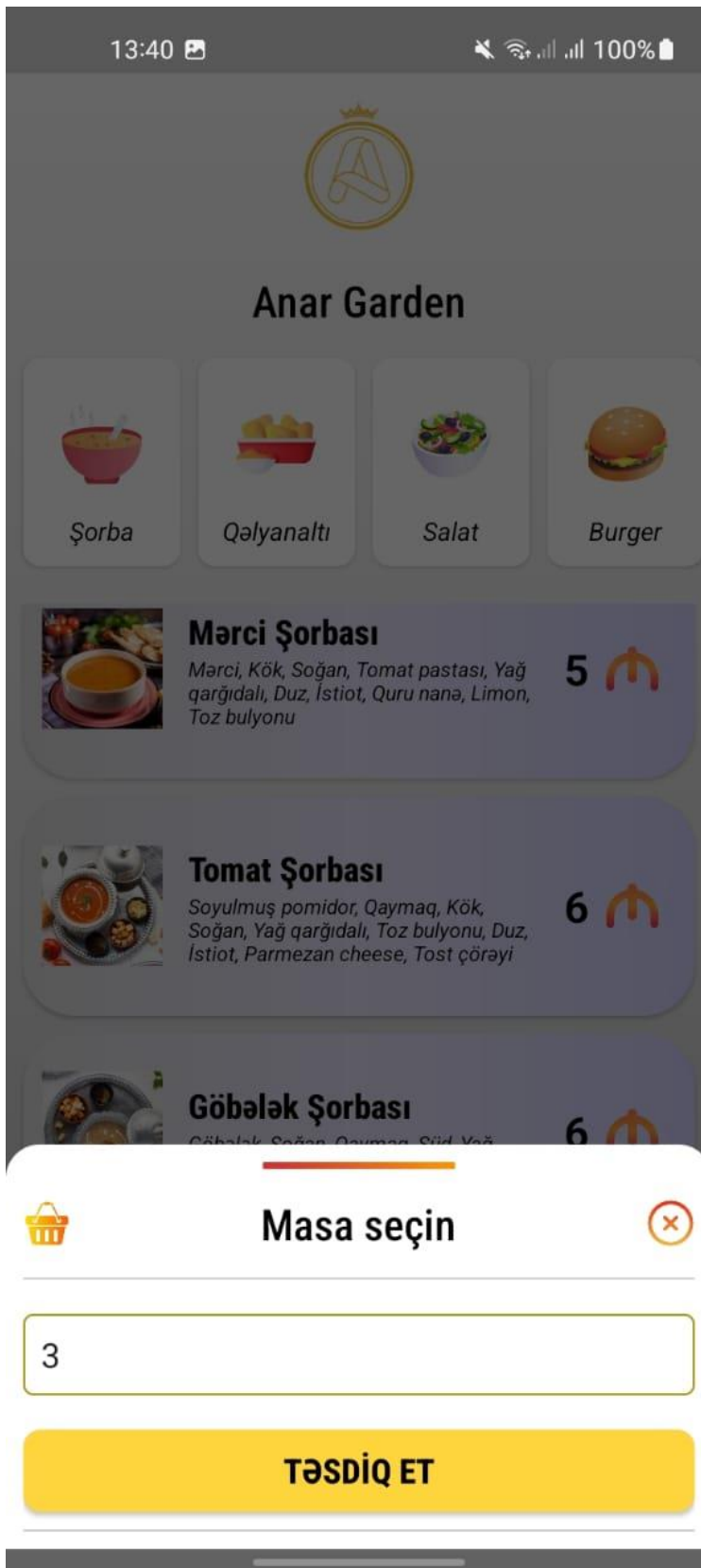
Şəkil 35. Səbət

İrəli düyməsinə klik etdikdə qiymət görünəcəkdir.



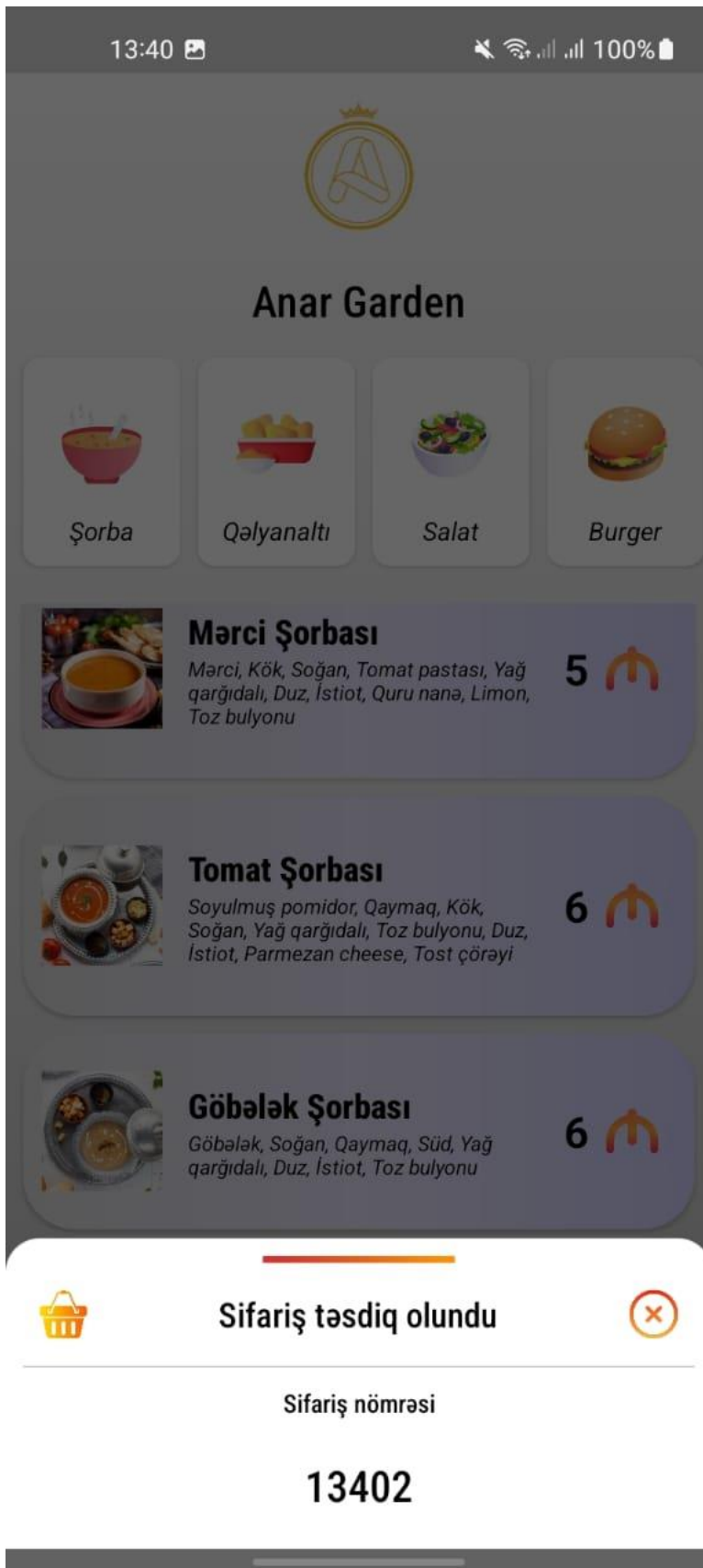
Şəkil 36. Ümumi qiymətlər

Əgər burada yemək seçiminə klikləsəniz masa nömrəsini və ya adını qeyd etməlisiniz.



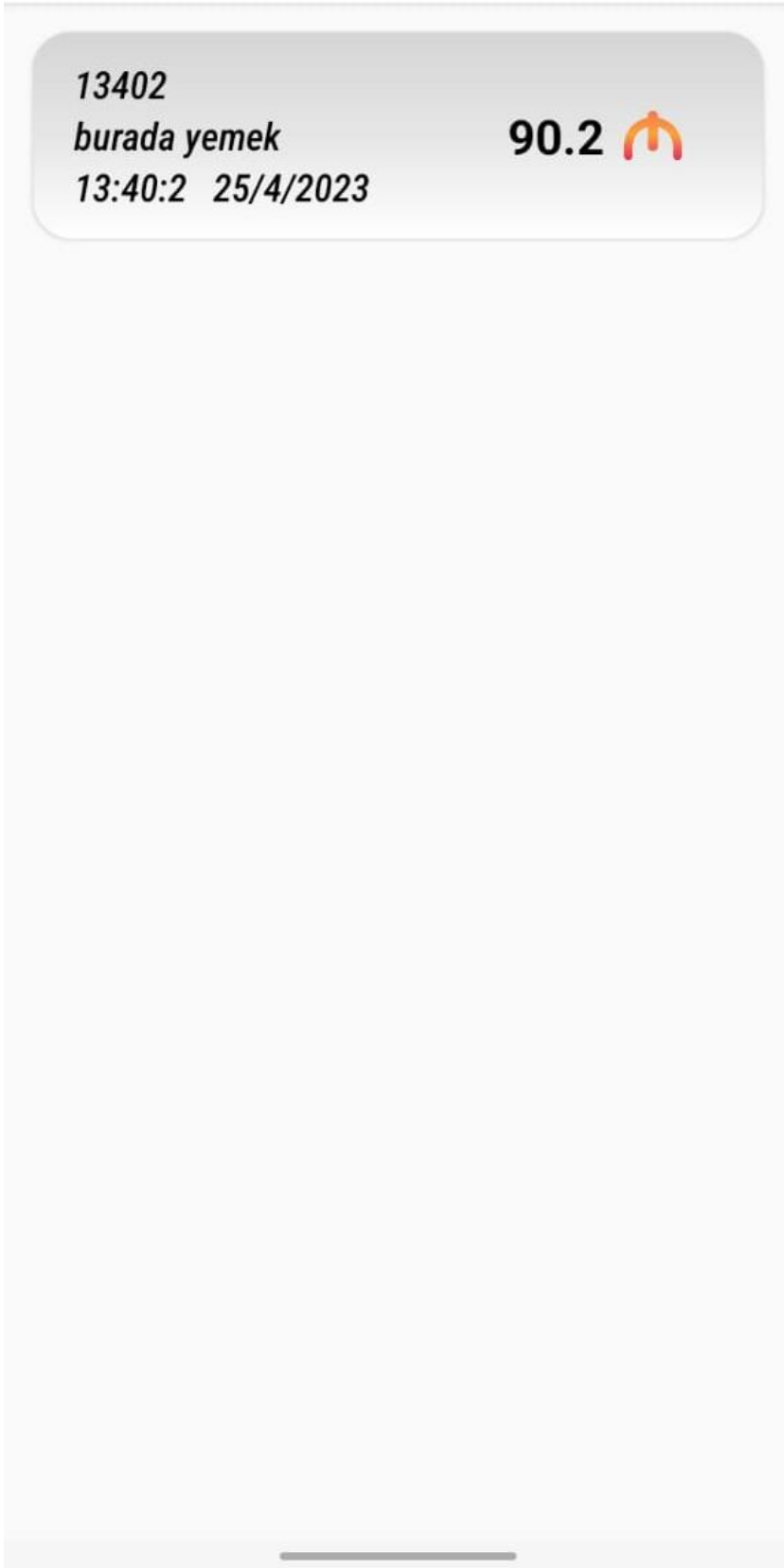
Şəkil 37. Masa seçimi

Daha sonra isə sifariş qeydə alınır və sifariş kodunuz qeyd olunur.



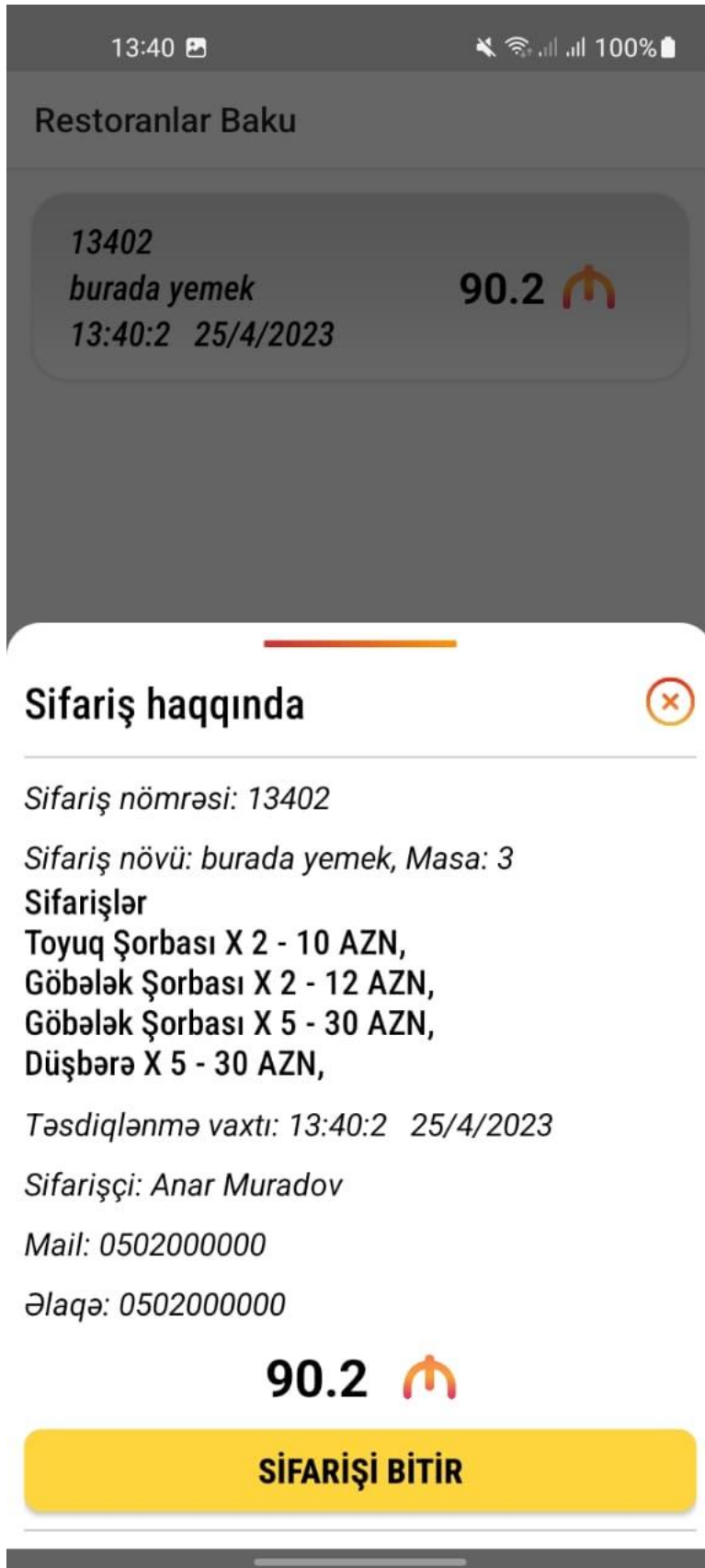
Şəkil 38. Sifariş kodu

Bu sifarişlər isə avtomatik olaraq restoran sahibinin hesabında görünür.



Şəkil 38. Sifarişlərin restoran sahibində görünməsi

Bu sifarişlərin üzərinə basdıqda isə ətraflı məlumatlar görünəcəkdir.



Şəkil 38. Sifarişin ətraflı izahı

3.3 Mobil tətbiqlərin restoran iş sahəsinə təsiri

Mobil tətbiqlər restoran iş sahəsinə dəyərli təsir göstərə bilirlər. İstifadəçilər üçün mobil tətbiqlər, restoranların sifarişləri qəbul etmələrini, menyunu göstərmələrini, masaların rezervasiyasını və xidmət keyfiyyətini yaxşılaşdırmağa kömək edir. İstehlakçılar üçün isə mobil tətbiqlər, restoran işinin idarəetməsini, müştərilərin statistikasını, finansal hesabatları və stokları idarə etmələrini asanlaşdırır. İşlənən tərəfdən baxdıqda, mobil tətbiqlər, restoranın görünürlüyünü artırmaq, markalaşmasını gücləndirmək və müştərilərə daha çox təşəkkür vermək üçün effektiv bir vasitədir. İstifadəçilər üçün, mobil tətbiqlər restoranların menularını, qiymətlərini və xüsusi təkliflərini görmək imkanı verir. Bu, müştərilərin sərbəst şəkildə restoran menusu ilə tanış olmasına və daha doğru seçimlər etməsinə kömək edir. Sifarişin mobil tətbiq vasitəsilə verilməsi, müştərilərə daha sürətli və daha dəqiq xidmət təmin etməyə kömək edir. Ayrıca, mobil tətbiqlər masanın rezervasiyasını və yerləşməyə hazırlanmış sifarişləri idarə etmək imkanı verir, bu da restorandakı effektivliyi artırır. Restoran işinin idarəçiləri üçün mobil tətbiqlər, işin idarə edilməsini asanlaşdırır. Mobil tətbiqlər, sifarişlərin idarə edilməsini avtomatlaşdıraraq insan faktorundan kaynaklanan yanlışlıqları azaldır. Ayrıca, finansal hesabatları və stokları idarə etmək üçün mobil tətbiqlərdən istifadə etmək, iş sahiblərinə effektivliyi artırmaq, məsuliyyətini izləmək və maliyyəvi məlumatları daha asanlıqla əldə etmək imkanı verir [21].

Mobil tətbiqlər, restoranların müştərilərinə xüsusi təkliflər, endirimlər və promosiyalar təqdim etmək imkanı verir. Bu, müştəriləri restorana cəlb etmək və onları təkrar gəlməyə məhv etmək üçün effektiv bir yoldur. Mobil tətbiqlər aynı zamanda müştərilərə sərəncam və qiymətləndirmələr təqdim edərək, müştərilərin rəylərini və təşəkkürlərini bildirmələrini təmin edir. Bu, müştərilərə daha şəffaf və interaktiv bir təcrübə yaşadır və onları restorana bağlamaq üçün daha çox əsas təşkil edir. Mobil tətbiqlər ayrıca restoranların sosial media platformaları ilə daha bərabər işləməsinə imkan verir. Müştərilər restoranı paylaşaraq, rəylərini və təcrübələrini sosial media kanallarında paylaşaraq restoranın tanınmışlığını artırmağa kömək edirlər. Bu, restoranın ətrafında bir mətbəx məntəqəsi yaratmaq və müştərilərin gələcəkdəki

potensial müştərilərə təsir etmək üçün güclü bir araçdır. Ümumi olaraq deyə bilərik ki,

ə

ə

ə Ayrıca, mobil tətbiqlər restoranlar üçün effektivliyi və verimliliyi artırmağa kömək edir. İşletmələr, məhsulların sifarişlərinin elektronik olaraq qəbul edilməsi sayəsində, əvvəlcədən hazırlıq edilmiş sifarişlərə vaxt və ehtiyat hissəsi ayırmaqla bilir. Bu, restoranın bir rol oynayır. İndi restoranlar, mobil tətbiqlər vasitəsilə müştərilərə endirimlər, xüsusi aracılığı ilə müştərilərin rəylərini, şəkillərini və qiymətləndirmələrini paylaşmalarını restoranın sosial media ətrafında bir cəmiyyət yaratmasına kömək edir və daha çox insanın restoran haqqında məlumat almalarına və onu sınaqlamağa maraqlanmalarına əlaqələrini gücləndirə bilər, fərqli bir təcrübə yaratmaq üçün innovativ yollar tapa bilər dönüşmələrinə kömək edə bilər [22].

r

a

t

m

a

q

,

i

ş

ə

ə

ə

ə

ə

ə

əmçinin, mobil tətbiqlər restoranların verilənlər bazasını artırmaq və müştərilər haqqında ətraflı məlumat toplamaq üçün də dəyərli bir vasitədir. Müştərilərin mobil

ə

ə

ətbiclərdə qeydiyyatdan keçməsi, restoranın onlar haqqında məlumat toplamağa imkan verir. Bu, müştərilərin səvədlərini, istəklərini və mövqeyini daha yaxşı anlamağa ya restoran-daxili sifarişləri də idarə etməyə kömək edir. Restoranlar, müştərilərə vaxtda restoranın məhsullarını sifariş etməsinə imkan verir. Bu, müştərilərə daha yaxşı planlama imkanı verir və restoranın sifarişləri daha effektiv şəkildə idarə etməsinə

ə

ə

r

r

e

s

t

o

r

a

n

i

ş

ə

ə

ə

ə

NƏTİCƏ

Dissertasiya işində restoranlarda müştərilərə online xidmət üçün mobil tətbiqin yaradılması, riyazi və kompüter modelləşdirilməsi tədqiq edilmişdir. Bu məqsədlə aşağıdakı işlər görülmüşdür.

İntegrasiya edilmiş mobil tətbiq sisteminin nəzəri aspektləri, mobil tətbiqlərin texniki aspektləri, Mobil tətbiqlərin günümüzdəki rolu, mobil tətbiqin dizaynı və mobil proqramın hazırlanması, restoranlar mobil tətbiqi üçün proqramlaşdırma dili və texnologiyaların seçimi, mobil tətbiqin analizi və effektivliyi, restoranlar mobil tətbiqinin əsas funksiyalarının təhlili, Sifarişlərin sərbətlə əlavə olunması və qiymətləndirilməsi, mobil tətbiqlərin restoran iş sahəsinə təsiri.

Dissertasiya işinin tapşırığına uyğun olaraq:

- a) Metod və modellər verilmişdir.
- b) Verilmiş metod və modellərin alqoritmi və onun analizi aparılmışdır.
- c) İnternet marketinq modelinin idarə edilən olması problemini yenillənmişdir.
- d) Mobil tətbiqin effektivliyinin qiymətləndirilməsi meyarlarını müəyyən edilmişdir.
- e) İnternet üzərindən marketinq üçün mövcud olan qaydalarla təyin olunmuşdur
- f) Mobil proqramların təqdim olunması texnologiyalarını təhlil edilmişdir.

Ədəbiyyat

1. Grove J. Service theater: an analytical framework for service marketing / J.Grove. – Lovelock, 2014. – 126 с
2. Всемирный банк – URL: <https://datatopics.worldbank.org/world-development-indicators/> (дата обращения 25.05.2022).
3. Попова Е. А., Носаева О. В., Пащенко П. О. Особенности маркетинговой среды Японских предприятий // Вопросы экономики и управления. – URL: <https://moluch.ru/th/5/archive/38/1084/> (дата обращения: 25.05.2022).
4. Голубкова, Е. Н. Интегрированные маркетинговые коммуникации: учебник и практикум для академического бакалавриата / Е. Н. Голубкова. – 3-е изд., перераб. и доп. – Москва: Издательство Юрайт, 2019. – 363 с.
5. Федеральная служба государственной статистики. Социально-экономическое положение России – URL: https://gks.ru/bgd/regl/b20_01/Main.htm (дата обращения 21.05.2022).
6. Социально-экономическое положение России – URL: <https://rosstat.gov.ru/storage/mediabank/osn-12-2021.pdf> (дата обращения: 21.05.2022).
7. Розничная торговля, услуги населению, туризм // Федеральная служба государственной статистики – URL: http://www.gks.ru/wps/wcm/connect/rosstat_main/rosstat/ru/statistics/enterprise/retail/# (дата обращения: 17.05.2022 г.).
8. Рынок общественного питания в России – URL: <https://cyberleninka.ru/article/n/rynok-obschestvennogo-pitaniya-v-rossii-osobennosti-i-tendentsii-razvitiya/viewer> (дата обращения: 18.05.2022).
9. Средний чек в ресторанах среднего ценового сегмента упал, несмотря на рост цен – URL: <https://marketing.rbc.ru/articles/13075/> (дата обращения: 14.05.2022).
10. Аналитический журнал бизнеса – URL: <https://www.tadviser.ru/index.php/R-Keeper> (дата обращения: 19.06.2022)

- 11.Маркетинговые коммуникации ресторанного бизнеса – URL: https://www.tadviser.ru/index.php/#cite_note-4 (дата обращения: 20.06.2022)
12. Рестораны не вернутся на докризисный уровень в 2021 году – URL: <https://www.vedomosti.ru/business/articles/2021/02/10/857472-restorani-vernutsya> (дата обращения: 20.06.2022).
- 13.Трофимова Е. В. Оценка эффективности рекламной деятельности. Обзор методов расчёта рекламного бюджета // Молодой ученый. – 2018. – № 6. – С. 471–473. 22.
- 14.Прокопивный С.Ф. Результаты и эффективность // Экономика предприятия: Учебник / С.Ф. Прокопивного. – М.: Финансы, 2015. – С. 204.
- 15.Краевская Н.А. Оценка эффективности рекламы // Актуальные проблемы экономики современной России, 2016. – № 3. – С. 401–405.
- 16.Шуякова М., Сакал П. Концепция устойчивых маркетинговых коммуникаций для идеального промышленного предприятия // Вопросы территориального развития – URL: <http://vtr.isert-ran.ru/article/2419> (дата обращения 17.06.2022).
- 17.Нацубидзе А. С. Внедрение стратегической системы управления конкурентоспособностью в деятельность современных российских предприятий // Аудит и финансовый анализ. – 2016. – № 1. – С. 298–302
- 18.. Парамонова Т.Н., Бикулов В.Ш. Составляющие элементы системы маркетинговых коммуникаций // Маркетинг. – 2014. – № 2. – С. 49–53
- 19.Самые успешные ресторанные концепции последних лет – URL: <https://restoplace.cc/blog/konceptcii-restoranov> (дата обращения: 15.06.2022)
- 20.Синее вино в Испании – URL: <https://www.bbc.com/travel/article/20161104-the-worlds-first-blue-wine> (дата обращения: 20.05.2022).
- 21.Литлфилд Р. Интегрированные маркетинговые коммуникации // Книги Лексингтона, 2021, с. 287
- 22.способа найти информацию о целевой аудитории– URL: <https://vc.ru/marketing/80538-22-sposoba-nayti-informaciyu-o-celevoy-auditorii> (дата обращения: 14.05.2022).

23. "Download Android Studio and SDK tools". Android Developers. Retrieved March 7, 2022.
24. "Download Android Studio and SDK tools". Android Developers. Retrieved March 7, 2022.
25. "Android Studio Dolphin". Android Developers Blog. Retrieved September 16, 2022.
26. "Android Studio Chipmunk". Android Developers Blog. Retrieved May 12, 2022.
27. "Google Go language IDE built using the IntelliJ Platform: go-lang-plugin-org/go-lang-idea-plugin, Go Language support for IDEA based IDEs, February 23, 2019, retrieved February 23, 2019, Supported IDEs [...] Android Studio 1.2.1+
28. "Android Studio Overview". Archived from the original on July 20, 2022. Retrieved July 20, 2022.
29. "Android Studio Hedgehog Canary 4 now available". Android Studio Release Updates. Retrieved May 16, 2023.
30. Chapman, Ben (April 3, 2018). "Google staff protest company's involvement with Pentagon drones programme". The Independent. Archived from the original on February 14, 2021. Retrieved April 21, 2021.
31. Amadeo, Ron (April 4, 2018). "Google employees revolt, say company should shut down military drone project". Ars Technica. Archived from the original on February 2, 2021. Retrieved April 21, 2021.
32. Greenwald, Glenn; MacAskill, Ewen (June 7, 2013). "NSA Prism program taps in to user data of Apple, Google and others". The Guardian. Archived from the original on August 18, 2006. Retrieved April 17, 2021.
33. Grant, Nico (June 13, 2022). "Google Agrees to Pay \$118 Million to Settle Pay Discrimination Case". The New York Times. ISSN 0362-4331. Retrieved June 14, 2022.

34. Complex, Valerie (June 12, 2022). "Google Agrees to Pay Out \$118 Million To Former Employees In Gender Discrimination And Pay Equity Suit". Deadline. Retrieved June 14, 2022.