**O‘ZBEKISTON RESPUBLIKASI RAQAMLI TEXNOLOGIYALAR VAZIRLIGI**

**MUHAMMAD AL-XORAZMIY NOMIDAGI TOSHKENT AXBOROT TEXNOLOGIYALARI UNIVERSITETI**

**1-Mustaqil ish**

Mavzu:“Restoran boshqaruv tizimi”

|  |  |  |
| --- | --- | --- |
| Bajardi | \_\_\_\_\_\_\_\_\_\_\_\_ | Eshonqulov O.SH  Shokirov J.A  Shukurov R.SH |
| Tekshirdi | \_\_\_\_\_\_\_\_\_\_\_\_ | Ro’zibayev O.B |

**Toshkent – 2024**

**Restoran boshqaruv tizimi**

1. **Ishtirokchi(actor) va obyekt(subject/system) o’rtasidagi bog’liqliklar foydalanish holatlar(use cases)ini ko’rsatish.**

Use case diagramma — bu tizimda mavjud jarayonlarni va foydalanuvchilarning tizim bilan o‘zaro aloqalarini grafik tarzda ko‘rsatuvchi diagrammadir. U aksariyat hollarda UML (Unified Modeling Language) standartidan foydalanadi va asosan dasturiy ta’minotni loyihalash jarayonida qo‘llaniladi. Use case diagramma orqali tizim foydalanuvchilari qaysi funksiyalarni ishlatishini va ularning tizim bilan qanday munosabatda bo‘lishini oson tushunib olish mumkin.

### Use Case Diagrammaning Asosiy Unsurlari

1. **Aktorlar (Actors)**

* Tizim bilan bevosita aloqa qiladigan tashqi subyektlar (foydalanuvchilar yoki boshqa tizimlar) bo‘lib, odatda odam yoki boshqa dasturiy ta’minot tizimi sifatida ifodalanadi.
* Ular diagrammada odam shaklidagi belgi yoki oddiy belgi yordamida ko‘rsatiladi.

1. **Use caselar (Use cases)**:

* Foydalanuvchi yoki aktor tomonidan bajarilishi mumkin bo‘lgan operatsiyalarni, vazifalarni ifodalaydi. Bu operatsiyalar tizimning funksionalligini ko‘rsatadi va oval shaklda tasvirlanadi.

1. **Aloqalar (Relationships)**:

* **Chiziqlar**: Aktor va use case orasidagi chiziqlar ularning o‘zaro aloqasini bildiradi.
* **Include** va **Extend**: Bu tushunchalar use case'lar orasidagi bog‘liqlikni ko‘rsatadi. Include boshqa use caseni majburiy tartibda qo‘shish vazifasini bajaradi, Extend esa qo‘shimcha yoki opsional operatsiyalarni bajarishi mumkin.

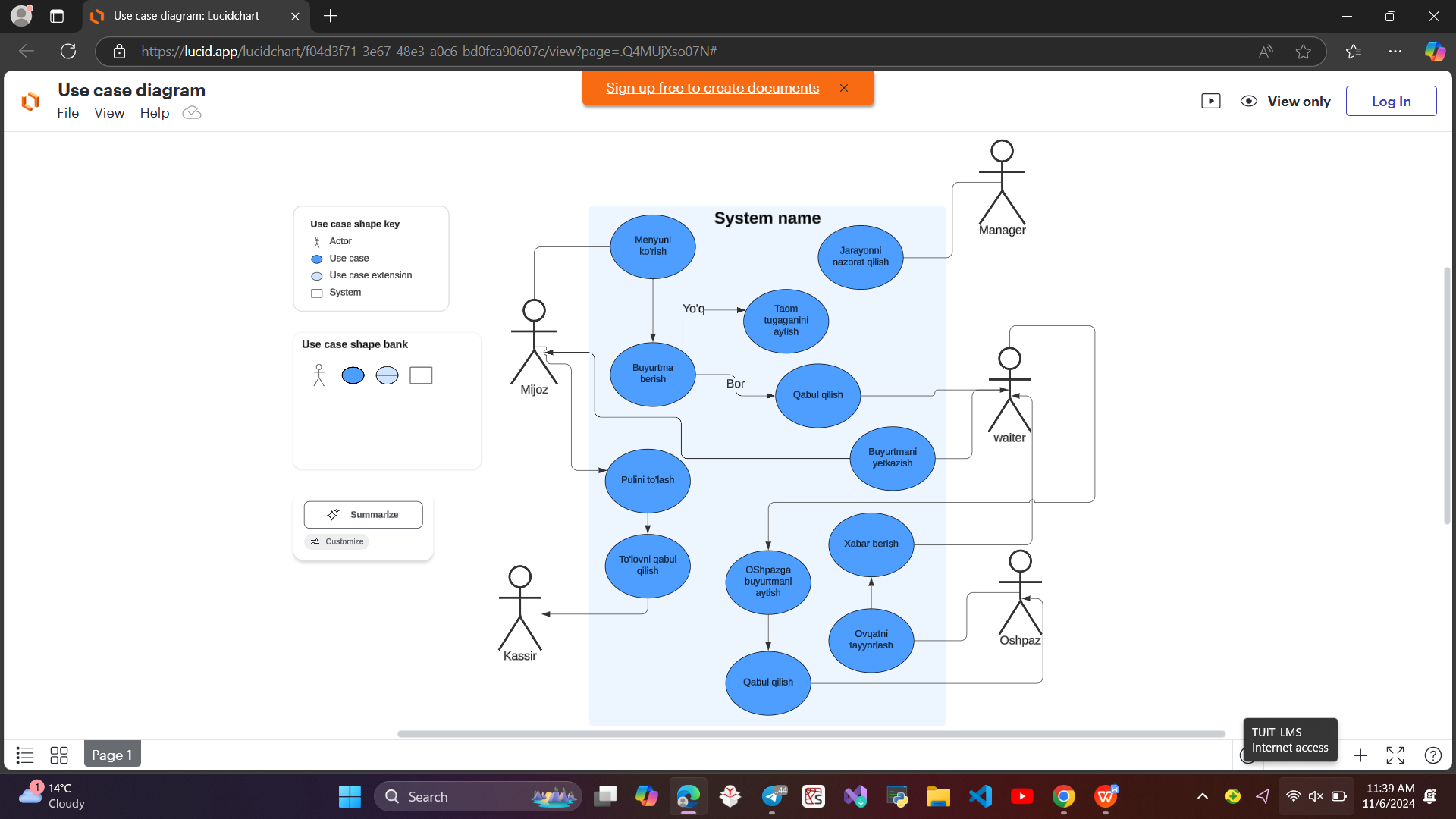
### Use Case Diagrammaning Foydalari

* **Talablarni aniqlash**: Tizim qanday funktsiyalarni taqdim etishi va foydalanuvchilarning qaysi funksiyalarni ishlatishi kerakligini tushunish uchun yaxshi vosita.
* **Kommunikatsiya**: Loyiha ishtirokchilariga tizim qanday ishlashi haqida umumiy tushuncha beradi, shu orqali texnik va no-texnik xodimlar o‘zaro tushunishni yaxshilaydi.
* **Loyihalashni osonlashtirish**: Dasturiy ta'minotni rejalashtirish va loyihalash jarayonida yordam beradi, chunki tizimning asosiy jarayonlari vizual tarzda ko‘rsatilgan bo‘ladi.

**Biz mustaqil ish mavzuyimiz use case diagrammasini Lucidchart da bajardik.**

**Quyidagi link orqali ham uni ko’rishingiz mumkin:**

<https://lucid.app/lucidchart/f04d3f71-3e67-48e3-a0c6-bd0fca90607c/view>



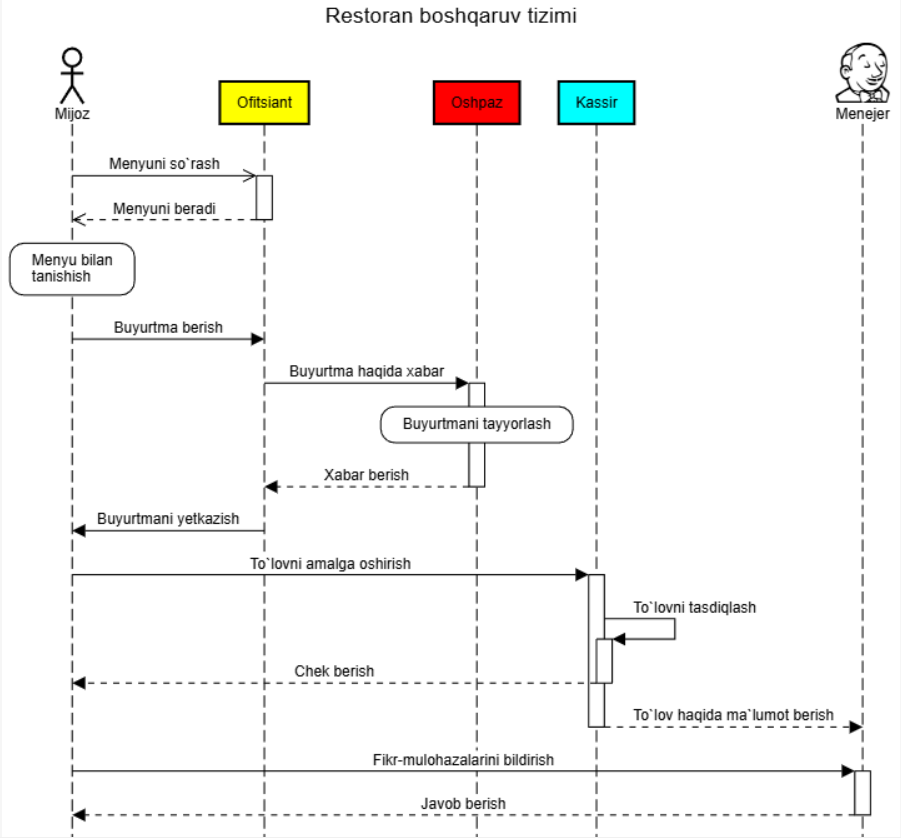
**2)Ketma-ketlik(Sequence) diagrammasi-bu xatti-harakatlarning vaqt**

**oralig’ida tuzilgan ketma-ket qadamlar tasviri.**

Restoran boshqaruv tizimi uchun Sequence Diagram ni tasvirlaydi. Diagrammada turli rollar va ular o'rtasida vaqt bo'yicha qanday interaktsiyalar sodir bo'lishi ko'rsatilgan. Quyida diagrammadagi asosiy jarayonlar bayoni keltiriladi:

### Rollar:

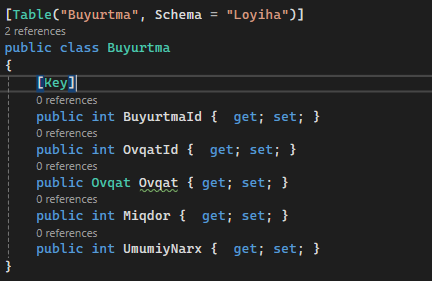
* **Mijoz → Ofitsiant**
* Mijoz menyuni ko'rib, ovqatlar va ichimliklar haqida ma'lumot oladi.
* Mijoz o‘z tanlagan ovqat va ichimliklar uchun buyurtma beradi.
* Mijoz buyurtma uchun to‘lov qiladi.
* Mijoz xizmat sifati yoki ovqat haqida fikr bildirishi mumkin.
* **Ofitsiant → Oshpaz**
* Ofitsiant mijozga menyu taklif qiladi va tanlashda yordam beradi.
* Ofitsiant mijozning buyurtmasini qabul qiladi.
* Ofitsiant qabul qilingan buyurtmani oshxonaga yoki oshpazga yetkazadi.
* Ofitsiant oshpaz tayyorlagan ovqatni mijoz stoliga olib boradi.
* **Oshpaz → Ofitsiant**
* Oshpaz ofitsiant yoki tizimdan kelgan buyurtmalarni qabul qiladi.
* Oshpaz buyurtma asosida ovqatni retseptga muvofiq tayyorlaydi.
* Oshpaz tayyor bo‘lgan ovqatlarni ofitsiantga tayyor ekanini bildiradi.
* **Kassir → Mijoz**
* Kassir mijoz yoki ofitsiant tomonidan keltirilgan to‘lovni qabul qiladi va tasdiqlaydi.
* Kassir to‘lov amalga oshirilganidan so‘ng kvitansiyani mijozga beradi.
* **Menejer → Xodimlar**
* Menejer xodimlarning jadvali va ishlash rejimini boshqaradi.
* Menejer oshxona uchun kerakli mahsulotlarni inventarizatsiya qiladi va zarurat bo‘lsa, buyurtmalarni amalga oshiradi.
* Menejer mijozlarning fikrlari va takliflarini tahlil qiladi va xizmat sifatini oshirish uchun choralar ko‘radi.

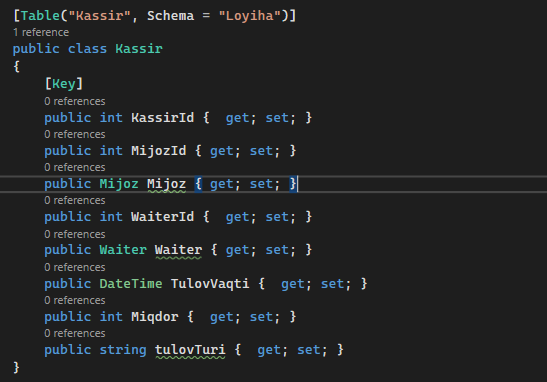


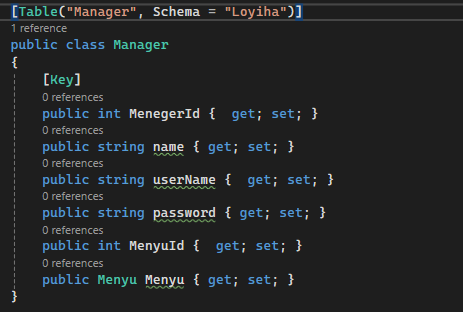
**Database architecture haqida**

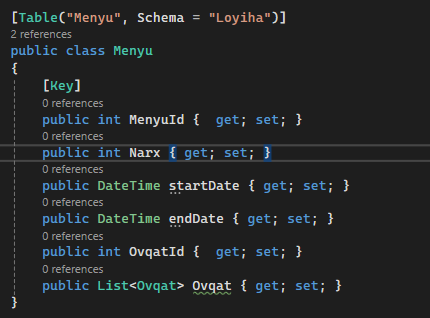
Ma'lumotlar bazasi arxitekturasi (database architecture) - bu ma'lumotlarni saqlash, boshqarish va ulardan foydalanish uchun tuzilgan tizimni loyihalash usuli. Ma'lumotlar bazasi arxitekturasi ko'pincha ma'lumotlar bazasining qanday tashkil etilganligi va foydalanuvchilar hamda dasturlar qanday qilib unga kirishi mumkinligini belgilaydi.

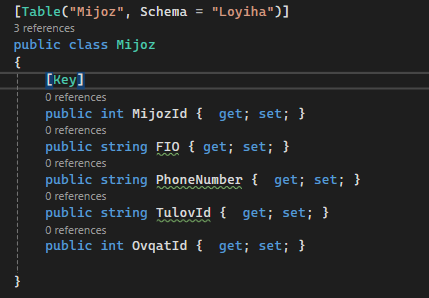
Ushbu surat ma'lumotlar bazasi modelini ko'rsatadi, unda quyidagi jadvallar mavjud:

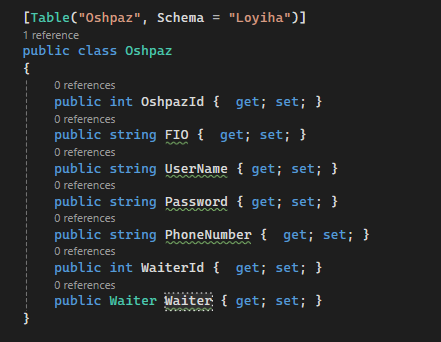


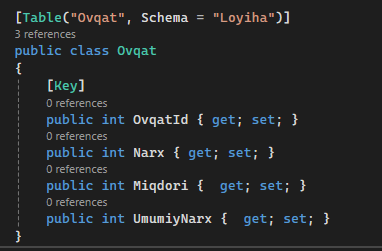


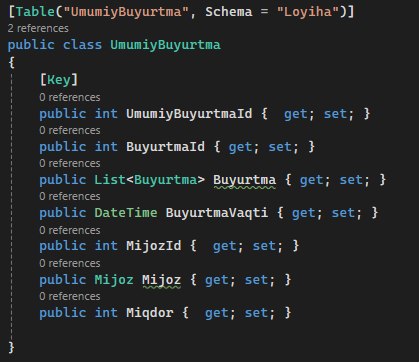


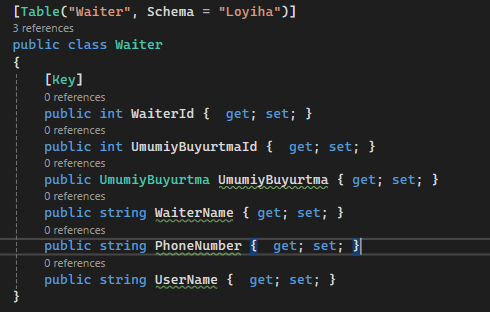


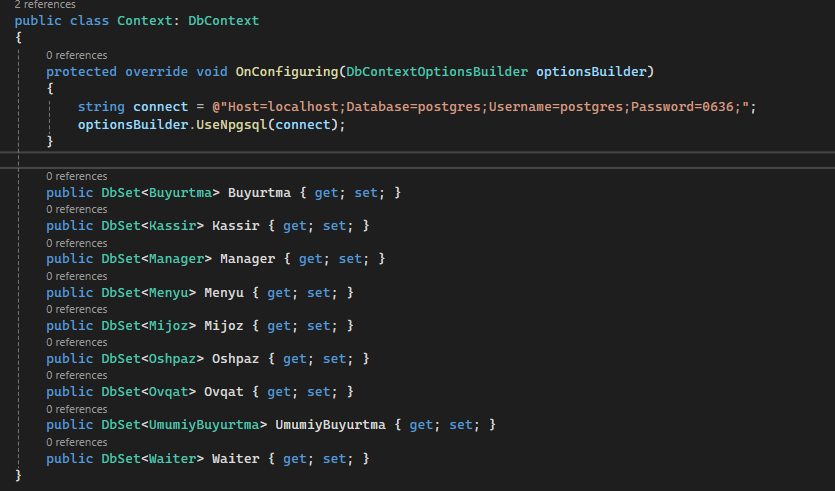
****

****

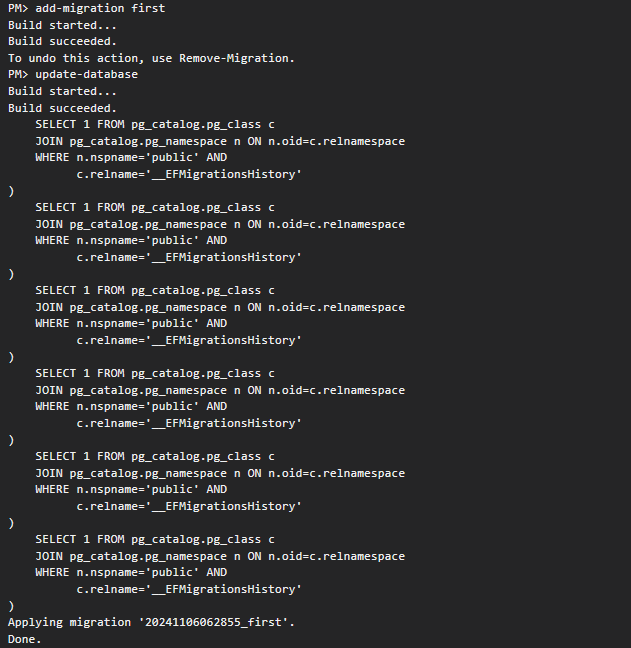
****

****

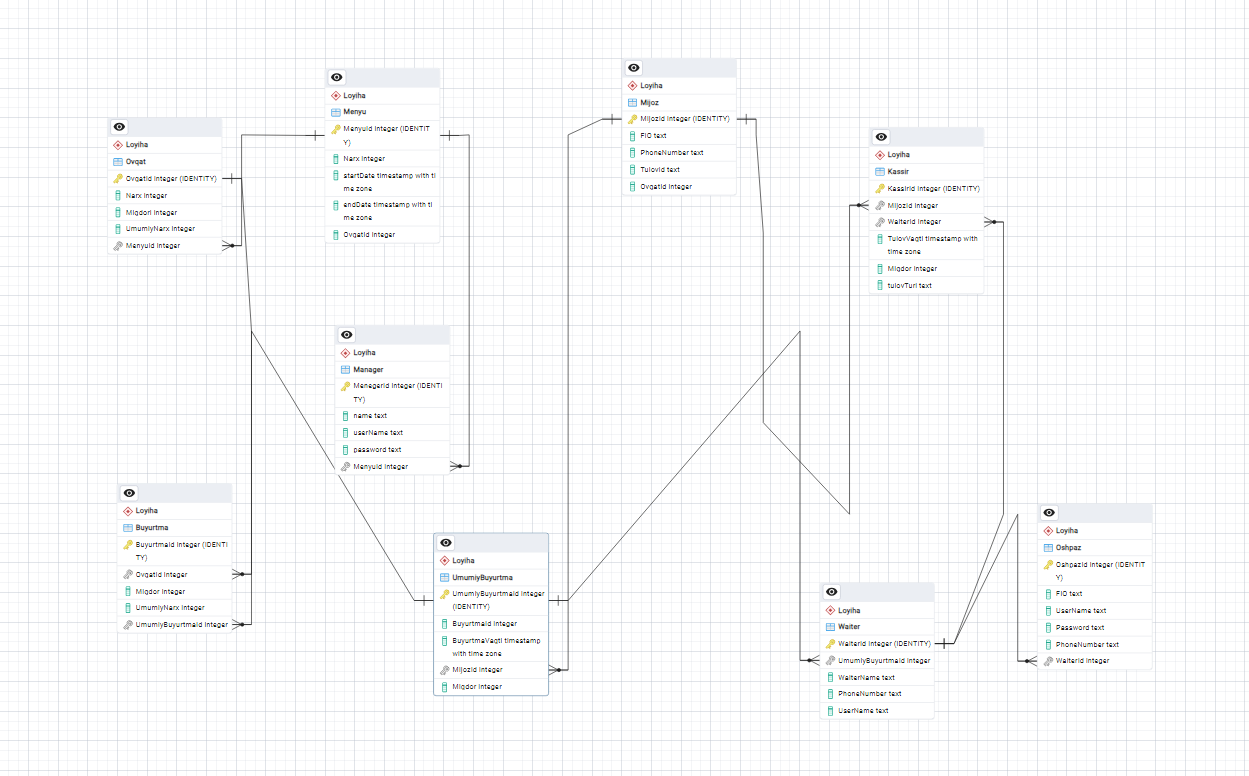
****

****

**C# va postrsql(database) ni bog’lanish jarayoni**



**Databasega jadvallarni yaratish jarayoni**



**Jadvallarning bog’lanishi**