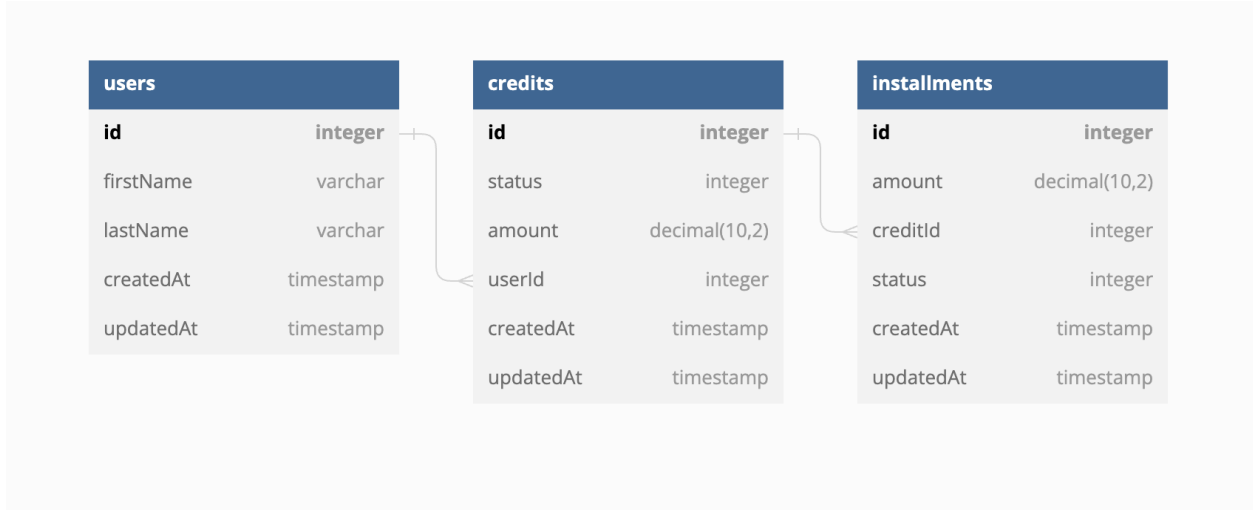


Aşağıdaki ilişkisel şeması verilen tablolardan basit bir seviyede finansal uygulama oluşturulması, bu kapsamda temel bankacılığın finansal araçlarından olan kredi kullandırım ve kredi geri ödeme fonksiyonlarının geliştirilmesi beklenmektedir.

Tablolar ve örnek istekler sadece yol gösterme amaçlıdır. ekleme ve çıkarmalar yapılabilir. uygulama kapsamı istenildiği kadar genişletilebilir.



- Programlama dili olarak Java veya Node.js tercih edilmelidir.
- Kullanıcı bazlı kredi oluşturulacaktır. **(Zorunlu)**
  - Örnek istek

```
{
  "userId": 1,
  "amount": 100.00,
  "installmentCount": 3
}
```

- Taksitlerin son ödeme tarihleri, kredi oluşturulma gününe 30'ar gün eklenerek hesaplanmalı. Taksit günü hafta sonuna denk geldiği durumda bir sonraki hafta içine denk gelen gün olmalıdır.
- Taksit tutarı = Kredi tutarı / taksit sayısı olarak hesaplanabilir. Faiz olmayacaktır.
- Örnek cevap

```
{
  "creditId": 1,
  "installments": [
    {
      "id": 1,
      "dueDate": "2023-06-02",
      "amount": 33.33
    },
    {
      "id": 2,
      "dueDate": "2023-07-03",

```

```

        "amount": 33.33
      },
      {
        "id": 3,
        "dueDate": "2023-08-01",
        "amount": 33.34
      }
    ]
  }
}

```

- Kullanıcı bazlı tüm krediler listelenebilmeli (**Zorunlu**)
- Kullanıcıya ait krediler filtrelenerek listelenebilmeli, pagination olmalı. (*Opsiyonel*)
  - Filtreler
    - status (Kredi Durumu)
    - date (Oluşturulma Tarihi)
- Krediye ait taksitler ödenebilmeli (**Zorunlu**)
  - Sadece toplam tutar ile ödeme yapılabilmesi yeterlidir. Opsiyonel olarak geliştirme yapılmak istenirse parçalı ödemeye de izin verilebilir (Önce 10 lira öde, sonra geri kalan tutarı öde gibi)
  - Örnek istek:
 

```

{
  "installmentId" : 1,
  "amount": 33.33
}

```
- Gecikmeye düşen krediler için gecikme faizi hesaplaması yapılması (*Opsiyonel*)
  - Bir endpoint üzerinden tetiklenmemeli, arkaplanda çalışmalı.
  - $(\text{Gecikmeye Düşen Gün Sayısı} * (\text{Faiz Oranı} / 100) * \text{Tutar}) / 360$
- Uygulama docker ile ayağa kaldırabilmeli (Opsiyonel)
- Test yazılması (Opsiyonel)
- Geliştirme yaparken git kullanılmalı ve branch /commit yapısına dikkat edilmeli (Opsiyonel)