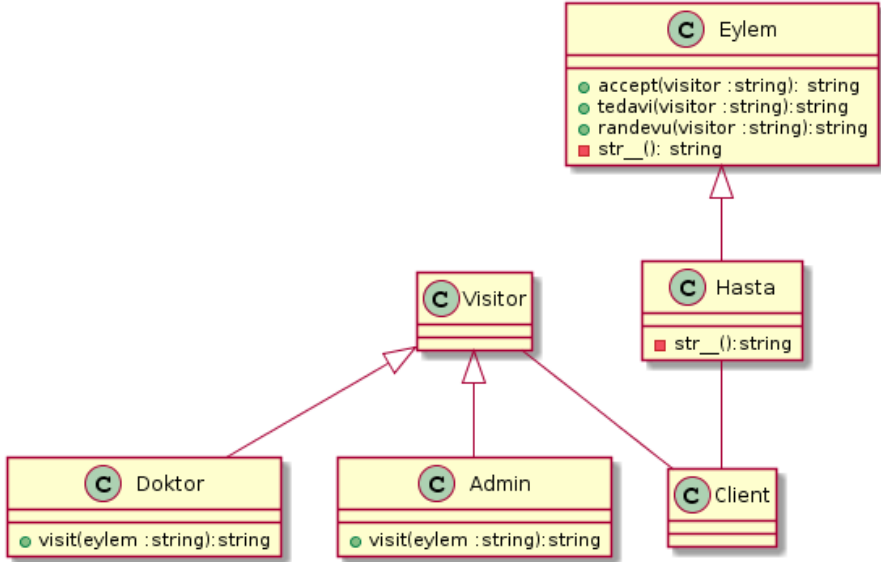


# VİSİTOR DESIGN PATTERN

Uygulamada ki sınıflara yeni metotlar eklenmesini düzenleyen bir tasarım desendir. Uygulamada ki bir yapıda ki bazı sınıflara bir işlevsellik eklenmesi gerektiğinde genelde ya yapıdaki sınıfların uyguladığı arayüzde metot tanımlanır ve bu arayüzü uygulayan sınıflarda metot yazılır veya eklenmesi gereken sınıflara eklenir. Arayüze metot tanımlarsak metodu kullanmayacak olan sınıflarda da bu metot uygulanacağı için tasarım prensiplerinden “Interface Segregation Principle - ISP” a uymamış oluruz. İlgili alt sınıflara ayrı ayrı metot eklersek de ileride bakımı zorlaşacak ve kodun karışmasına neden olacak bir yapı kurmuş oluruz. Böyle durumlarda bu hatalara düşmemek için visitor tasarım deseni kullanılabilir.

Classes - Class Diagram



```
@startuml
```

```
title Classes - Class Diagram
```

```
class Hasta{
    -str_():string
}
class Eylem{
```

```
+accept(visitor :string): string
+tedavi(visitor :string):string
+randevu(visitor :string):string
-str__(): string

}
class Visitor {

}
class Doktor{
    +visit(eylem :string):string
}
class Admin{
    +visit(eylem :string):string
}
class Client {

}
Eylem<|-- Hasta
Visitor<|-- Doktor
Visitor<|-- Admin
Visitor -- Client
Hasta -- Client

@enduml
```