

ECODATION  
Hamit Mızrak  
DevOPS Specialist, Advisor , Java  
Senior Full Stack Developer  
JAVA SE  
ECODATION

# Sunumda olacaklar

- ▶ Projemizin ekran videosu (youtube)
  - ▶ GitHub linki
  - ▶ Kaynak Kodlar
- 
- ▶ örneğin: Java SE-11 Staj Sunumu Bilgisayar Müh. Hamit Mızrak [hamitmizrak@gmail.com](mailto:hamitmizrak@gmail.com)
  - ▶ **Proje linki:** <https://github.com/hamitmizrak/otomasyon.git>

# Java SE

- ▶ Java SE bütün konular
- ▶ Database Sql bilmek
- ▶ MVC yapısını mantığını bilmek
- ▶ Design pattern ile proje geliştirme mantığını bilmek. (Singleton design pattern)

# KURULACAK PROGRAMLAR

- ▶ Eclipse - STS
- ▶ Postgresql - MariaDb - Mysql ( server , workbench)

# Java SE

- ▶ Java SE Bütün konuları kapsacak şekilde kod yazmak çok önemli özellikle;
- ▶ Clean code , S.O.L.I.D
- ▶ data structure = Set,Map,List
- ▶ OOP: abstract,inheritance,interface
- ▶ enum
- ▶ Design pattern yapıları gereken yerlerde kullanılacak.
- ▶ stream,optional

# Git /Github

- ▶ CVS olarak Git repository olarak GitHub kullanacağız yapılan projeyi GitHub üzerinden repository public proje olarak açılarak paylaşılabilecek.
- ▶ README.md olmalı

# Console Bankamatik Programı Tanıtımı

- ▶ Login ve Register console ekranı olup database kontrol ederek kişi önceden üye ise database giriş yapabilecek kayıt olunması gerekecek.
- ▶ Register database attributes: emailadres,password,kullanıcı adı
- ▶ Projemizde 2 tane rol olacak
- ▶ 1-) Banka (admin) 2-) Müşteri (user)
- ▶ Database bağlanarak CRUD ( Create -Read -Update -Delete ) işlemler yapabileceğimiz bir proje yapacağız.
- ▶ Her rolün kendisine ait console ekranı olacaktır. sadece süper admin(Banka) müşterinin yaptığı değişiklikleri loglama olarak görebilecek.

# Müşteri ekranı

- ▶ Müşteri Şifrenini doğru girdikten sonra açılan ekranda;
  - ▶ Para yatırma
  - ▶ Para Çekme
  - ▶ Havale
- 
- ▶ NOT: her CRUD (Database işlemlerimizde mutlaka) transaction işlemini yapalım.



# Banka ekranı

- ▶ Banka Şifrenini doğru girdikten sonra açılan ekranda;
- ▶ Müşterinin günlük hesap kontrollerini yapabilecek
- ▶ Kayıt olan müşterilerini aktif edebilir veya pasif duruma getirebilir.
- ▶ NOT: her CRUD (Database işlemlerimizde mutlaka) transaction işlemini yapalım.

# Araştırma Ödevleri

- ▶ Agile nedir ? avantajları ?
- ▶ Scrum nedir ? avantajları ?
- ▶ Mvc nedir ? avantajları ?
- ▶ Design Pattern nedir ? avantajları ?
- ▶ Software prensible ?

# Yazılımda en çok kullanılan terimler (25)

- ▶ delete=silmek
- ▶ create=oluşturmak
- ▶ subkategori=alt kategori