

ÇANAKKALE ONSEKİZ MART ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

NOTE OVERFLOW SİTESİ

Figen Kardelen YILDIRIM
Abdul Karim TOURE
Augusto GOMES JUNIOR

Dr. Bora UĞURLU

Aralık, 2019

ÇANAKKALE

Figen Kardelen YILDIRIM, Abdul Karim TOURE ve Augusto GOMES JUNIOR tarafından **Dr.Bora UĞURLU** yönetiminde hazırlanan “**Note Overflow Sitesi**” başlıklı çalışma tarafımızdan okunmuş, kapsamı ve niteliği açısından bir Bitirme Ödevi olarak kabul edilmiştir.

Dr. Bora UĞURLU

NOTE OVERFLOW SİTESİ

ÖZET

Projemiz Çanakkale 18 Mart Üniversitesi'nde bilgisayar mühendisliği okuyan öğrencilere kolaylık sağlamak amacıyla düşünülmüştür. Tasarladığımız uygulamada öğrenciler ders notlarını paylaşabilecek ve diğer öğrenciler tarafından bu ders notlarına kolayca erişim sağlanabilecektir.

Projemizin amacı: ÇOMÜ 'lü bilgisayar mühendisliği öğrencilerinin faydalanabileceği bir web uygulaması yapmayı amaçlıyoruz. Yapmayı planladığımız sitede öğrenciler sahip oldukları ders notlarını arkadaşlarıyla paylaşabilecek ve ihtiyaçları olan ders notlarına kolayca ulaşabileceklerdir.

Projemizin Kapsamı: Sitemizi ÇOMÜ'de bilgisayar mühendisliği okuyan öğrenciler için tasarladık. Sitede ise öğrenciler ders notlarını paylaşabilecekler ve ders notlarına ulaşabileceklerdir.

İÇİNDEKİLER

BİTİRME ÖDEVİ ONAY SAYFASI.....	2
ÖZET.....	3
BÖLÜM 1-GİRİŞ	5
1.1 Amaç Ve Kapsam.....	5
1.2 Hedefler Ve Başarı Kriterleri.....	5
1.3 Genel Bakış.....	5
BÖLÜM 2-GEREKSİNİMLER	6
2.1 Sistem ve Üye Gereksinimleri.....	6
2.2 Fonksiyonel ve Fonksiyonel Olmayan Gereksinimler.....	7
BÖLÜM 3-SİSTEM MODELLEME.....	9
3.1 Actor-Goal Model.....	9
3.2 Use Case 1	10
3.3 Use Case 2.....	11
3.4 Use Case 3.....	12
3.5 Use Case Models.....	13
3.6 SSD Diagrams.....	15
3.7 State Machine Diagrams.....	17
3.8 Domain Model.....	18
BÖLÜM-4 Veritabanı	20
4.1 Tables.....	20
BÖLÜM- 5 Arayüz Tasarımı.....	23
BÖLÜM- 6 SONUÇ.....	28
KAYNAKLAR	29

BÖLÜM- 1

GİRİŞ

1.1 Amaç Ve Kapsam

Projemizin amacı: ÇOMÜ ‘lü bilgisayar mühendisliği öğrencilerinin faydalanabileceği bir web sitesi yapmayı amaçlıyoruz. Yapmayı planladığımız sitede öğrenciler sahip oldukları ders notlarını arkadaşlarıyla paylaşabilecek ve ihtiyaçları olan ders notlarına kolayca ulaşabileceklerdir.

Projemizin Kapsamı: Sitemizi ÇOMÜ’de bilgisayar mühendisliği okuyan öğrenciler için tasarladık. Sitede ise öğrenciler ders notlarını paylaşabilecekler ve ders notlarına ulaşabileceklerdir.

1.2 Hedefler Ve Başarı Kriterleri

Hedefler:

- Projemizin öncelikli hedeflerinden biri öğrencilerin elinde olmayan notlara kolay bir şekilde ulaşabilmeleridir.
- Öğrenciler arası yardımlaşma ve iletişimin güçlenmesi.

Başarı Kriterleri:

- Not paylaşmak isteyen öğrenciler sorunsuz şekilde siteye üye olmalıdırlar ve profil sayfasına doğru bir şekilde bilgilerini girmelidir.
- Notlara ulaşmak isteyen öğrenci başarılı bir şekilde siteye üye olmalıdır ve profil sayfasına doğru bir şekilde bilgilerini girmelidir.
- Notlara ulaşmak isteyen öğrenci istediği notun detaylarını sorunsuz bir şekilde görüntüleyebilmelidir.

1.3 Genel Bakış

Projemiz ÇOMÜ ‘de bilgisayar mühendisliği okuyan öğrencilere kolaylık sağlamak için düşünülmüştür. Tasarladığımız web sitesinde bir yandan öğrenciler arkadaşlarıyla kolayca not paylaşırken bir yandan da elinde olmayan notlara kolay bir şekilde ulaşabileceklerdir.

BÖLÜM-2

GEREKSİNİMLER

2.1 Sistem Gereksinimleri

- Sistem meydana gelecek aksaklıkları gidermelidir.
- Kurum ihtiyaçlarını analiz etmelidir.
- Sistem kurallarına uymayan üyeleri sistemden çıkarmalıdır.
- Sunucu tarafında yapılacak işleri gerçekleştirir.
- Üyelerin paylaşmak istedikleri notları inceler ve paylaşılmaya uygun bulduklarının paylaşılmasına izin verir.

2.1 Üye Gereksinimleri

- Üyeler sisteme giriş yapmalıdır.
- Üyeler paylaşacağı notların bilgilerini girmelidir.
- Üyeler indireceği notların detaylarını görüntüleyebilir.
- Eğer üyeler sisteme not eklemek isterse ilgili bölümden not ekleyebilir.
- Eğer üyeler sistemden not indirmek isterse ilgili bölümden not indirebilir.
- Üyeler ekleyeceği nota uygun dersi seçmelidir.
- Üyeler eklediği not hakkında gelen soruları yanıtlamalı, diğer üyelerle iletişime geçmelidir.
- Üyeler birbirlerinin profillerini ziyaret edip , birbirleri hakkında bilgi sahibi olabilirler.
- Üyeler işlemlerini tamamladıktan sonra sistemden çıkış yapmalıdır.

2.2 Fonksiyonel Gereksinimler

- Bu sistem adı, soyadı, mail adresi, telefon ve belirlenen parola ile sisteme kayıt yapmaya imkan sağlamalıdır.
- Sisteme üye olan birinin kayıt olurken belirlediği parola ve kullanıcı adı ile sisteme giriş yapmasına imkan sağlamalıdır.
- Üyenin profil sayfası oluşturmaya ve gerektiğinde bilgilerini güncellemesine imkan sağlamalıdır.
- Üyenin siteye not eklemesine imkan sağlamalıdır.
- Üyenin siteye yüklediği notu açıklamalarla detaylandırmasına imkan sağlamalıdır.
- Üyenin siteye yüklediği notu uygun kategoriye yerleştirmesine imkan sağlamalıdır.
- Üyenin ihtiyaç duyduğu notu ararken notları belli kriterlere göre(bölümü, sınıfı, ders adı vb.) şeklinde kategorilere ayırmasına imkan sağlamalıdır.
- Üyenin ilgilendiği notun detaylarını görmesine imkan sağlamalıdır.
- Üyenin ilgilendiği notun bulunduğu sayfada notla ilgili merak ettiği başka şeyler için soru sorma imkanı sağlamalıdır.
- Üyenin istediği arkadaşlarıyla ya da herkesle notu paylaşabilmesine imkan verilmelidir.
- Üyenin ulaşmak istediği nota ulaşabilmesine imkan verilmelidir.
- Üyenin diğer profil sayfalarını görmesine imkan sağlamalıdır.
- Üyelerin sistem veya diğer üyelerle ilgili şikayet ve isteklerini bildirmesine imkan sağlamalıdır.
- Üyenin güvenli bir şekilde çıkış yapmasına olanak sağlamalıdır.

2.2 Fonksiyonel Olmayan Gereksinimler

2.2.1.Security (Güvenlik) :

- Sisteme sadece sistem yöneticisi tarafından erişilebilmelidir.
- Şifre politikası uygulanmalıdır. (Minimum bir büyük, küçük harf ve bir sayı.)
- Üye bilgileri kimse ile paylaşılmamalıdır.
- Sistem bakımını sadece sistem yöneticisi yapmalıdır.
- Yetkili kişiler sistemi güvenli bir şekilde kullanmalıdır.

2.2.2.Usability (Kullanılabilirlik) :

- Sistem her kullanıcı türü için anlaşılır, sade ve kullanılabilir olmalıdır.
- Sistemde yapılmak istenen işlemler en hızlı şekilde yapılmalıdır.

2.2.3.Performance (Performans) :

- Sistem, kapatılana kadar sorunsuzca çalışmalıdır.
- Üyelerin sistemde yapacakları işlemler kısa süre içerisinde olmalıdır.
- Sistemin aynı anda birden fazla kullanıcıyı kaldırma kapasitesi olmalıdır.

2.2.4.Supportability (Bakım) :

- Sistemin bakımı üyelerin en az işlem yaptığı vakitlerde yapılmalıdır.
- Sistem bakımı yapılırken oluşabilecek sorunlara karşı, sistemde giriş yapmış üyeler uyarılarak çıkışları yapılmalıdır.
- Bakım yapılacağına yakın mağduriyet olmaması için daha önceden üyelere haber verilmelidir.
- Sisteme yeni donanım eklendiğinde uyumluluk sağlanmalıdır. Implementation

2.2.5.Constraints (Sistem Kısıtlamaları) :

- Sistemde bulunması gereken belgeler ve sertifikalar eksiksiz bir şekilde sistem yöneticisinde bulunmalıdır.
- Belli zamanlarda sistem istatistiği tutulmalıdır.
- Sistemi kullanmak için mutlaka üyelerin interneti olmalıdır.
- Üyeler sistem kurallarına uymalıdır.
- Sistem kurallarına uymayan üyeler yöneticiler tarafından sistemden uzaklaştırılmalıdır.

BÖLÜM-3
SİSTEM MODELLEME

3.1 ACTOR-GOAL MODEL

Sistem Yöneticisi	<ul style="list-style-type: none">• Sistemde meydana gelebilecek aksaklıkları giderir.• Kurum ihtiyaçlarını analiz eder.• Sistem kurallarına uymayan üyeleri sistemden çıkarır.• Sunucu tarafında yapılacak işleri gerçekleştirir.• Üyelerin paylaşmak istediği notları inceler ve paylaşmasına izin verir.
Üye	<ul style="list-style-type: none">• Üyeler sisteme giriş yapar.• Üyeler paylaşacağı notlarının bilgilerini girer.• Üyeler ilgilendikleri notların detaylarını görebilir.• Üyeler ekledikleri notlar hakkında yapılan yorumları görebilir veya kendileri de yorum ekleyebilir.• Üyeler paylaşmak amacı ile sisteme not ekleyebilir.• Üyeler istediği notları indirebilir.• Eklediği notlara yapılan yorumlara ve kendisine sorulan sorulara cevap verebilirler.• Üyeler diğer profilleri ziyaret edebilirler.• Üyeler sistem veya diğer üyelerle ilgili şikayet ve isteklerini bildirebilirler.• Üyeler sistemden çıkış yaparlar

SEÇİLEN EN KRİTİK ÜÇ USE CASE

3.2 USE CASE 1: Üyenin Sisteme Not Eklemesi

Scope: Not Paylaşım Sistemi

Level: User Goal

Primary Actor: Üye

Stakeholders and Interest: Üye sisteme hızlı ve başarılı bir şekilde not eklemek ister.

Precondition: Üye sisteme kayıtlı ve giriş yapmış olmalıdır.

Postcondition: Üye sisteme yeni bir not eklemiş olmalıdır.

Ana Başarı Senaryosu

- 1.Üye not ekleme işlemini başlatır.
- 2.Sistem üyenin notu detaylandırmasını ister.
- 3.Üye sistemin not ile ilgili istediği bilgileri ve notları kimlerle paylaşmak istediğini eksiksiz bir şekilde sisteme girer.
- 5.Sistem girilen bilgileri üyeye gösterir.
- 6.Üye not bilgilerini kaydeder ve notu ekler.

Alternatif Senaryolar

*a.Sistem herhangi bir anda çökerse,

- 1.Sistem yeniden başlatılır.
- 2.Üye işleme 2. adımdan devam eder.

3.a.Üye not bilgilerini eksik girerse,

- 1.Sistem tarafından üyeye hata mesajı gönderilir.
- 2.Üye eksik bilgileri tamamlar ve ana başarı senaryosu 4. adımdan devam eder.

3.3 USE CASE 2:Sistem Yöneticisinin Notların Paylaşılmasına İzin Vermesi

Scope: Not Paylaşım Sistemi

Level: User Goal

Primary Actor: Sistem Yöneticisi

Stekholders and Interest: Sistem Yöneticisi üyelerin eklediği notları inceler ve paylaşmaya uygun olan notları diğer üyelerle paylaşır.

Precondition: Üye sisteme notları eklemiş olmalıdır.

Postcondition: Sistem yöneticisi üyenin eklediği notları paylaşmış olmalıdır.

Ana Başarı Senaryosu

- 1.Sistem yöneticisi not paylaşımına izin verme işlemini başlatır.
2. Sistem paylaşılacak notları gösterir.
- 3.Sistem yöneticisi notları inceler ve uygun olanların paylaşılmasına izin verir.
- 4.Sistem detayları sistem yöneticisine gösterir.
- 5.Sistem yöneticisi yapılan işlemi kaydeder.

Alternatif Senaryolar

*a.Sistem herhangi bir anda çökerse,

- 1.Sistem yeniden başlatılır.
- 2.Üye işleme 2. adımdan devam eder.

3.4 USE CASE 3:Üyenin Sistemdeki Notlara Ulaşması

Scope: Not Paylaşım Sistemi

Level: User Goal

Primary Actor: Üye

Stakeholders and Interest:Üye istediği not doğrultusunda sistemden hızlı ve başarılı bir şekilde notlara ulaşmak ister.Not detaylarını inceleyerek, notlar arasında kıyaslama yapar ve kendisine yardımcı olacak nota ulaşır.

Precondition: Üye sisteme kayıtlı ve giriş yapmış olmalıdır.

Postcondition: Üye istediği nota sistemden ulaşmış olmalıdır.

Ana Başarı Senaryosu

- 1.Üye notları görüntüleme işlemini başlatır.
2. Sistem seçilebilecek kategorileri gösterir.
- 3.Üye ulaşmak istediği notun kategorisini seçer.
- 4.Sistem üyenin seçtiği kategorideki notları görüntüler.
- 5.Üye sistemin görüntülediği notlardan istediğini seçer.
- 6.Sistem seçilen notu görüntüler.
- 7.Üye notların detaylarına ulaşmak ister.
- 8.Sistem notun detaylarını görüntüler.
- 9.Üye seçtiği notun detaylarını görüntüler.
- 10.Üye ulaşmak istediği notu indirir.

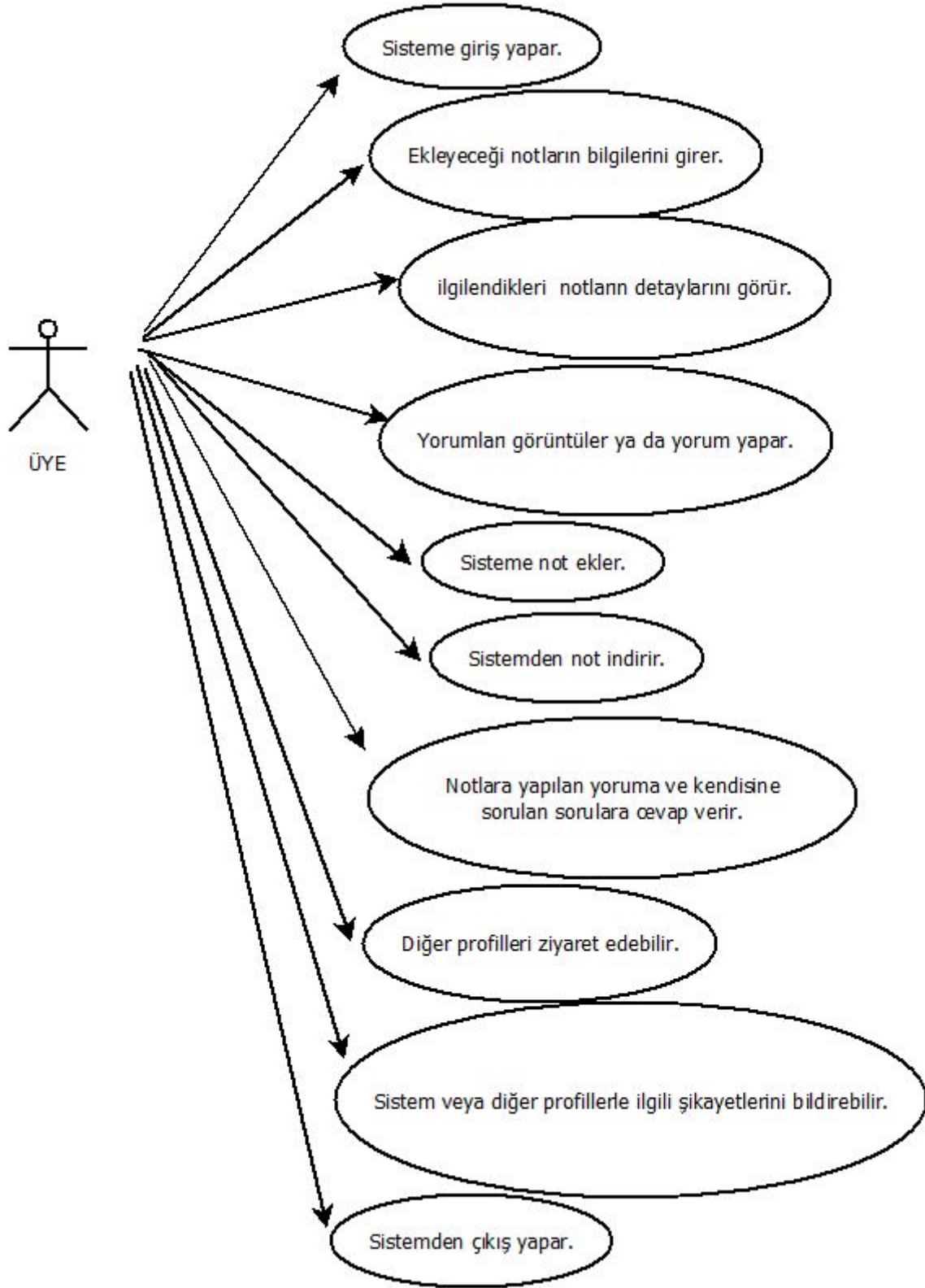
Alternatif Senaryolar

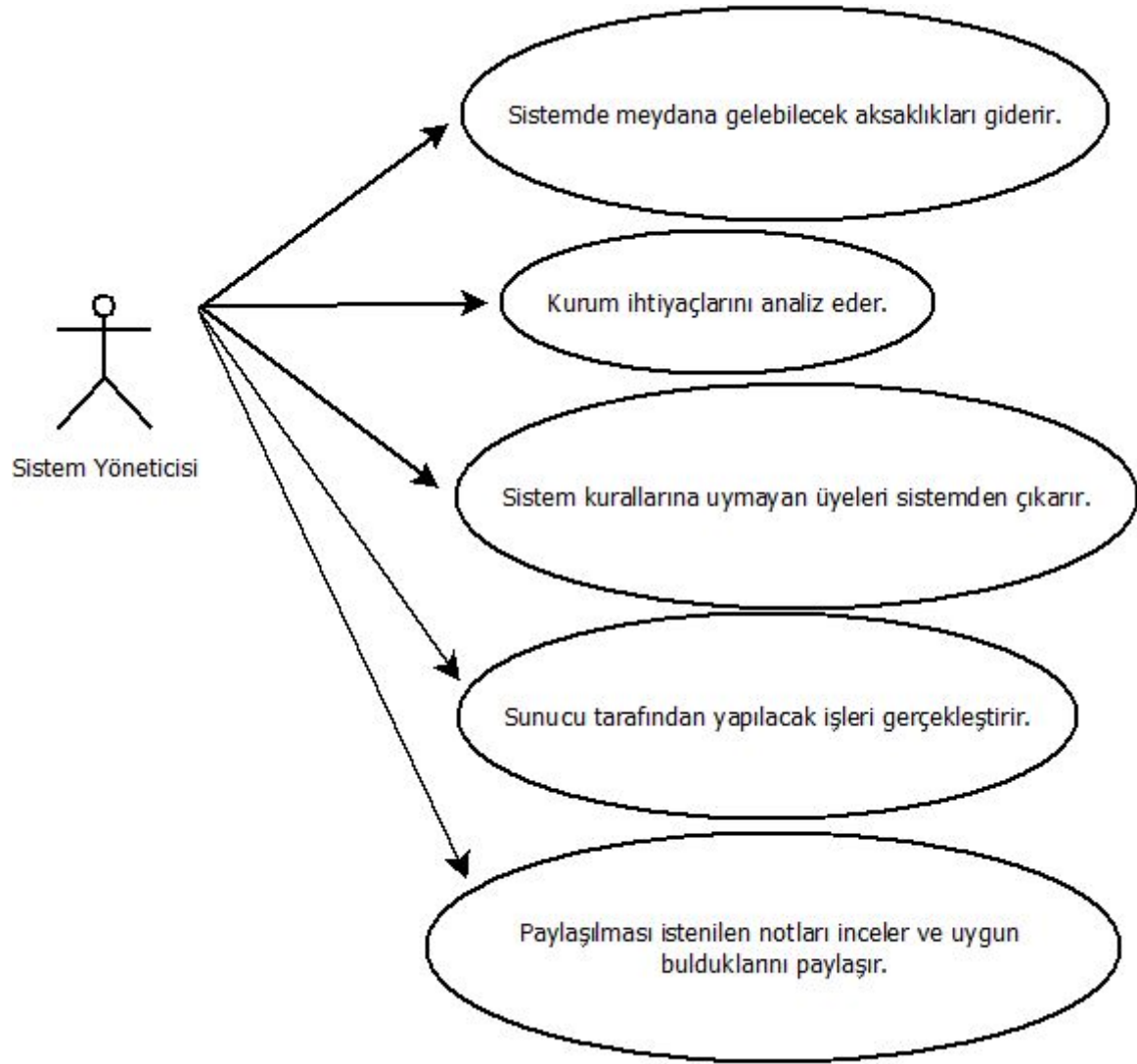
*a.Sistem herhangi bir anda çökerse,

- 1.Sistem yeniden başlatılır.

2.Üye işleme 2. adımdan devam eder.

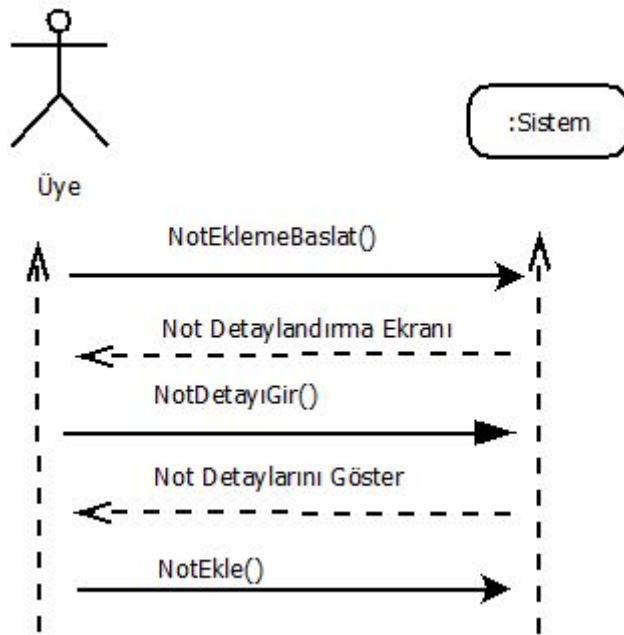
3.5 USE CASE MODELS



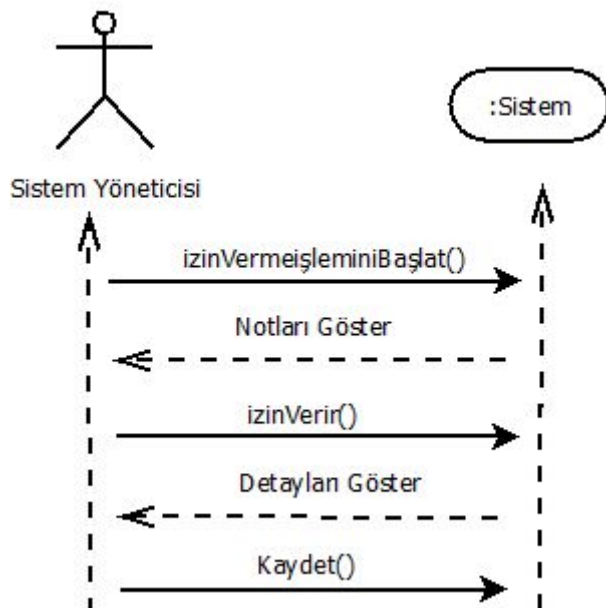


3.6 SSD DIAGRAMS

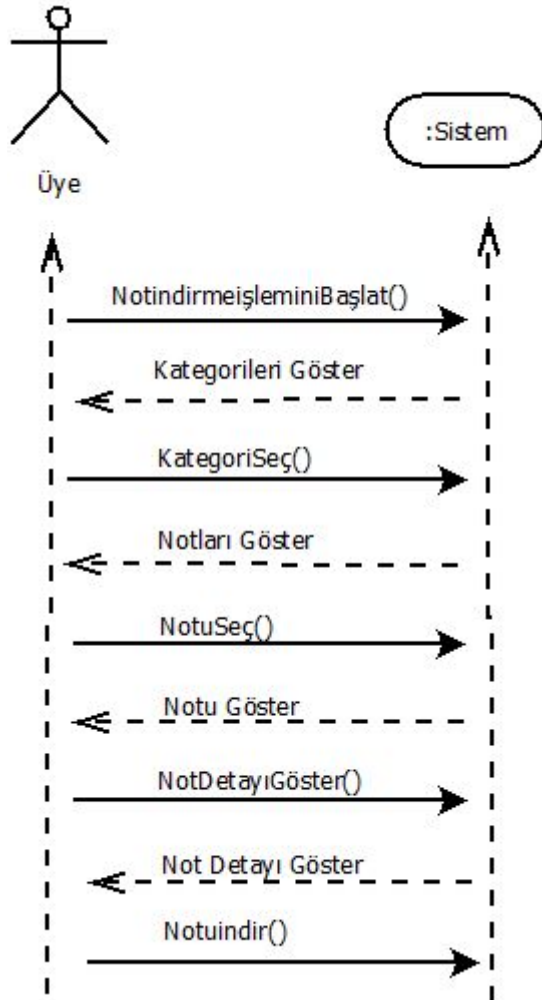
USE CASE 1 SSD MODEL



USE CASE 2 SSD MODEL

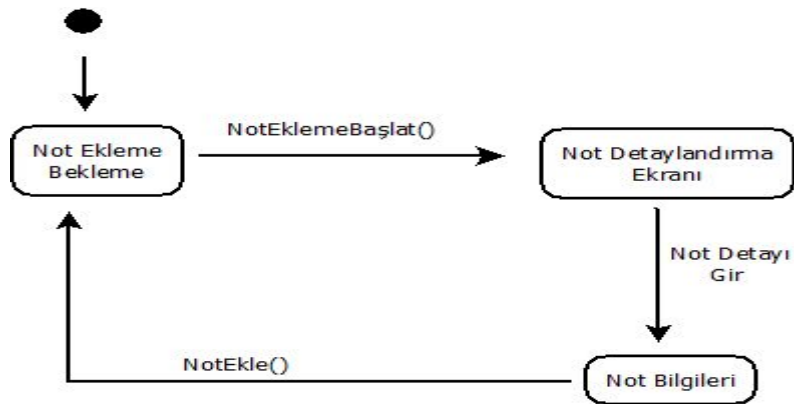


USE CASE 3 SSD MODEL

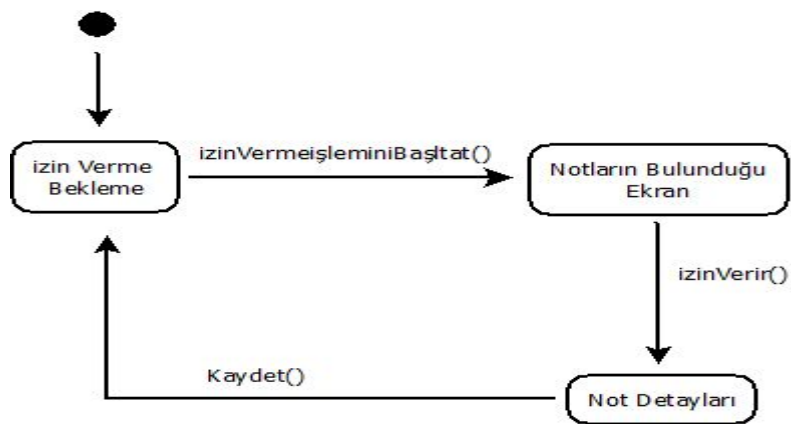


3.7 STATE MACHINE DIAGRAMS

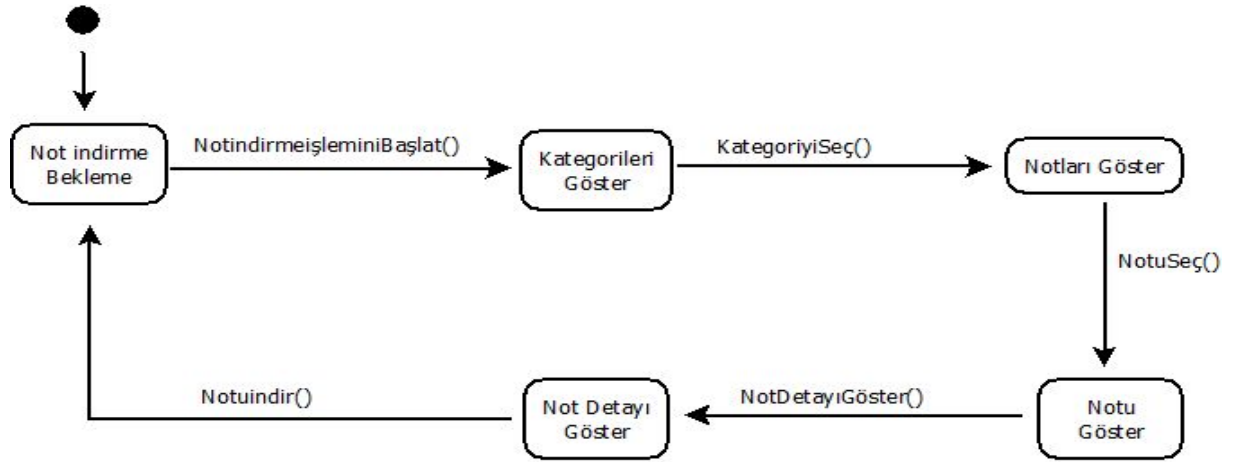
USE CASE 1 SMD



USE CASE 2 SMD

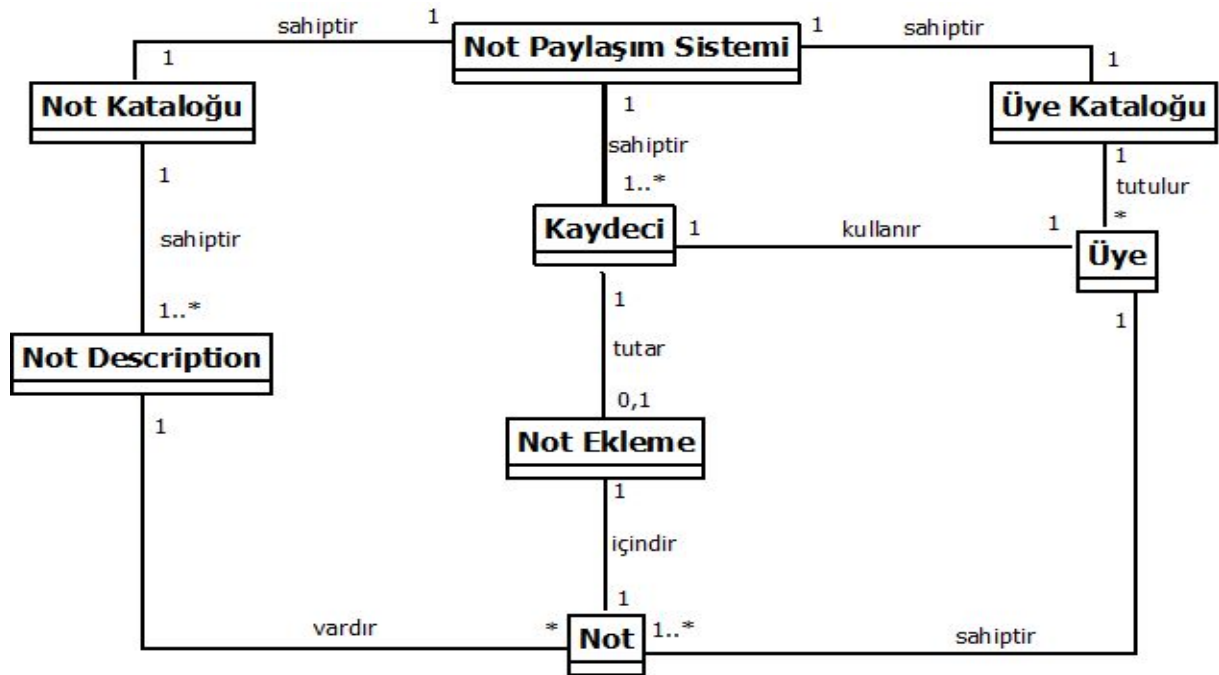


USE CASE 3 SMD

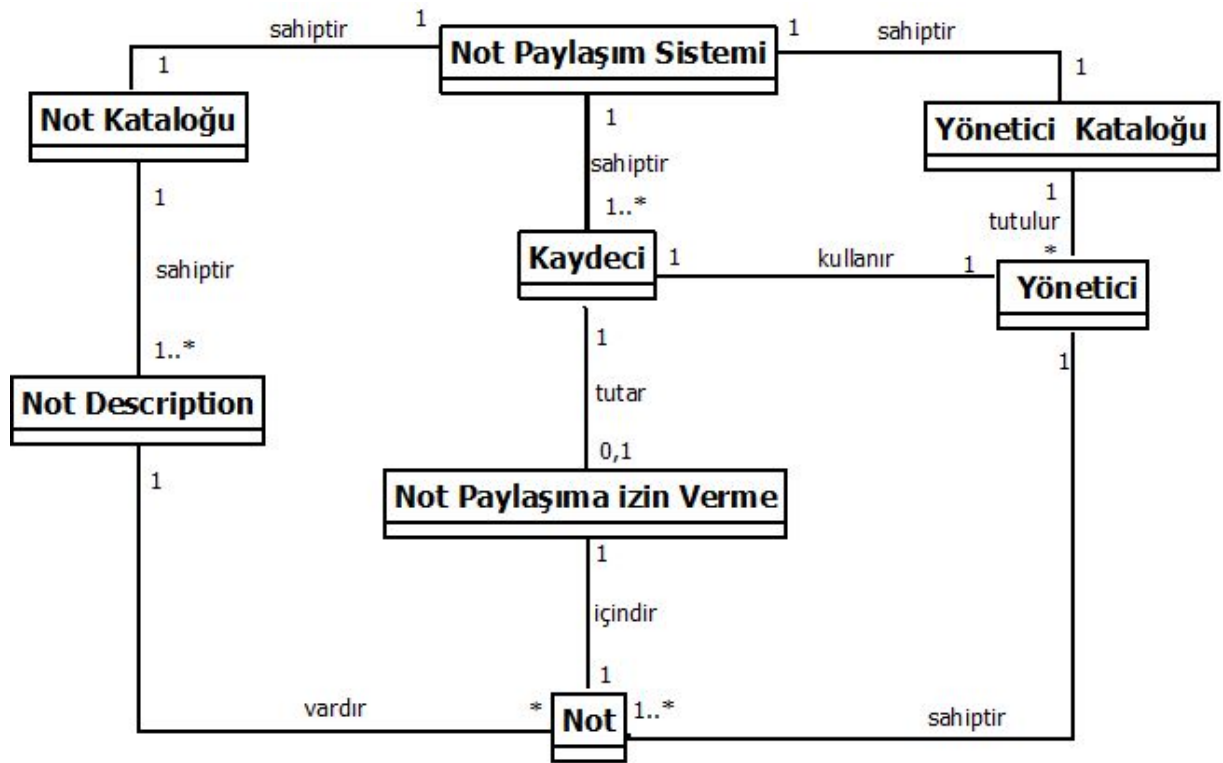


3.8 DOMAIN MODELS

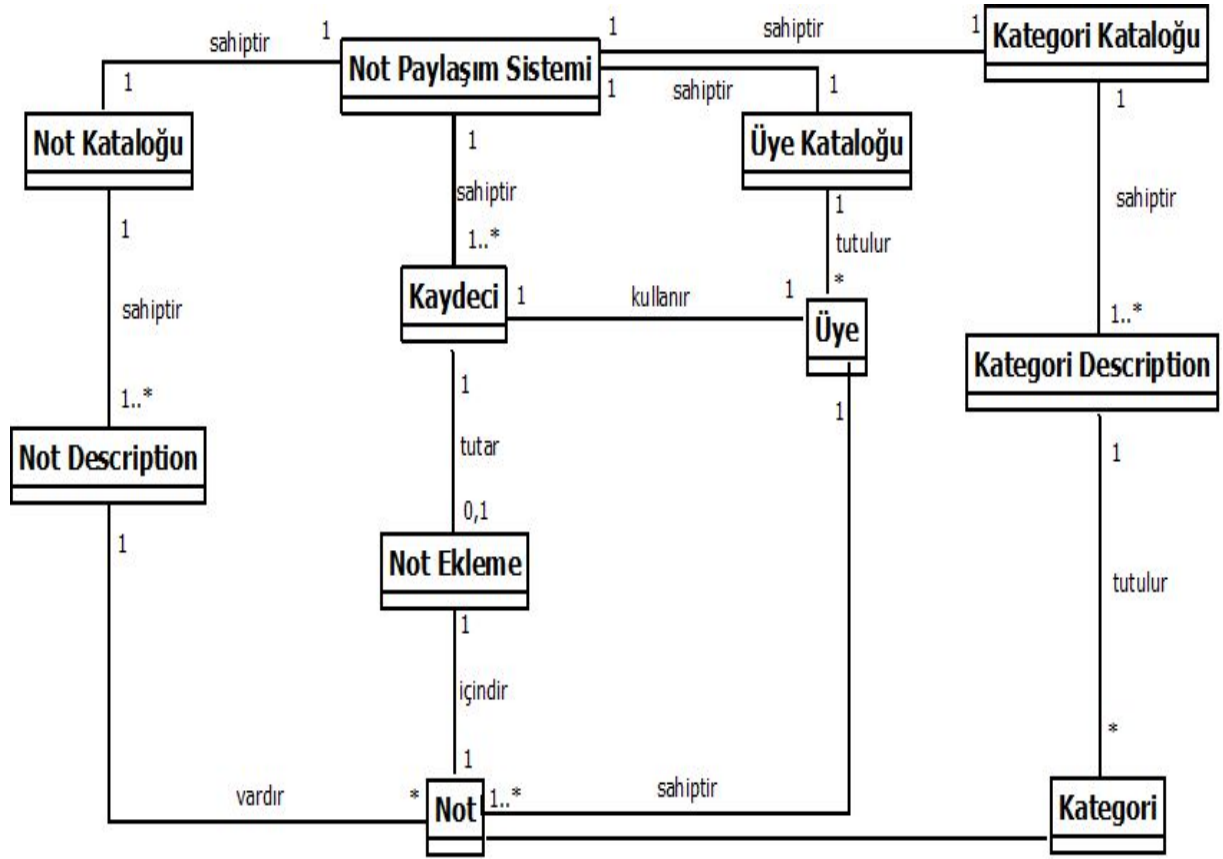
USE CASE 1: Üyenin sisteme not eklemesi



USE CASE 2: Sistem Yöneticisinin Notların Paylaşılmasına İzin Vermesi



USE CASE 3: Üyenin Sistemdeki Notlara Ulaşması



BÖLÜM 4

VERİTABANI

4.1 Tables

Server: localhost:3306 » Database: noteoverflow

StructureSQLSearchQueryExportImportOperationsRoutinesEventsMore

Table	Action	Rows	Type	Collation	Size	Overhead
aktiviteler Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	48 KiB	-	
arkadaslar Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	16 KiB	-	
bengenme Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	32 KiB	-	
dersler Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	16 KiB	-	
mesajlar Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	32 KiB	-	
notlar Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	48 KiB	-	
posts Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	48 KiB	-	
uyeler Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	16 KiB	-	
yorumlar Browse Structure Search Insert Empty Drop	0	InnoDB	utf8mb4_unicode_ci	16 KiB	-	

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	uye_id	int(10)			No	None		AUTO_INCREMENT	Change Drop Primary More
2	uye_adi	varchar(50)	utf8mb4_unicode_ci		No	None			Change Drop Primary More
3	uye_soyadi	varchar(50)	utf8mb4_unicode_ci		No	None			Change Drop Primary More
4	email	varchar(100)	utf8mb4_unicode_ci		No	None			Change Drop Primary More
5	parola	varchar(200)	utf8mb4_unicode_ci		No	None			Change Drop Primary More
6	username	varchar(50)	utf8mb4_unicode_ci		No	None			Change Drop Primary More
7	telefon	varchar(20)	utf8mb4_unicode_ci		No	None			Change Drop Primary More
8	admin	tinyint(1)			No	0			Change Drop Primary More
9	aktif	tinyint(1)			No	0			Change Drop Primary More
10	kayit_tarihi	datetime			No	None			Change Drop Primary More

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	not_id	int(10)			No	None		AUTO_INCREMENT	Change Drop Primary More
2	not_adi	varchar(50)	utf8mb4_unicode_ci		No	None			Change Drop Primary More
3	tarih	datetime			No	None			Change Drop Primary More
4	uye_id	int(10)			No	None			Change Drop Primary More
5	not_durumu	varchar(20)	utf8mb4_unicode_ci		No	None			Change Drop Primary More
6	ders_id	int(10)			No	None			Change Drop Primary More

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	yourum_id	int(10)		No	None		AUTO_INCREMENT	Change Drop Primary More
<input type="checkbox"/>	2	not_id	int(10)		No	None			Change Drop Primary More
<input type="checkbox"/>	3	yorum_metni	varchar(500)	utf8mb4_unicode_ci	No	None			Change Drop Primary More
<input type="checkbox"/>	4	yorum_tarihi	datetime		No	None			Change Drop Primary More
<input type="checkbox"/>	5	uye_id	int(10)		No	None			Change Drop Primary More

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	mesaj_id	int(10)		No	None		AUTO_INCREMENT	Change Drop Primary More
<input type="checkbox"/>	2	uye_id	int(10)		No	None			Change Drop Primary More
<input type="checkbox"/>	3	gonderilecek_kisi	int(10)		No	None			Change Drop Primary More
<input type="checkbox"/>	4	masaj_tarihi	datetime		No	None			Change Drop Primary More
<input type="checkbox"/>	5	mesaj_metni	text	utf8mb4_unicode_ci	No	None			Change Drop Primary More

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	beginme_id	int(10)		No	None		AUTO_INCREMENT	Change Drop Primary Unique More
<input type="checkbox"/>	2	not_id	int(10)		No	None			Change Drop Primary Unique More
<input type="checkbox"/>	3	beginme_tarihi	datetime		No	None			Change Drop Primary Unique More

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
<input type="checkbox"/>	1	arkadas_id	int(10)		No	None		AUTO_INCREMENT	Change Drop Primary More
<input type="checkbox"/>	2	uye_id	int(10)		No	None			Change Drop Primary More
<input type="checkbox"/>	3	arkadas_olma_tarihi	datetime		No	None			Change Drop Primary More

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	aktivite_id	int(10)			No	None		AUTO_INCREMENT	Change Drop Primary Unique More
2	uye_id	int(10)			No	None			Change Drop Primary Unique More
3	post_id	int(10)			No	None			Change Drop Primary Unique More

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	post_id	int(10)			No	None		AUTO_INCREMENT	Change Drop Primary Unique More
2	not_id	int(10)			No	None			Change Drop Primary Unique More
3	yorum_id	int(10)			No	None			Change Drop Primary Unique More
4	post_tarihi	datetime			No	None			Change Drop Primary Unique More

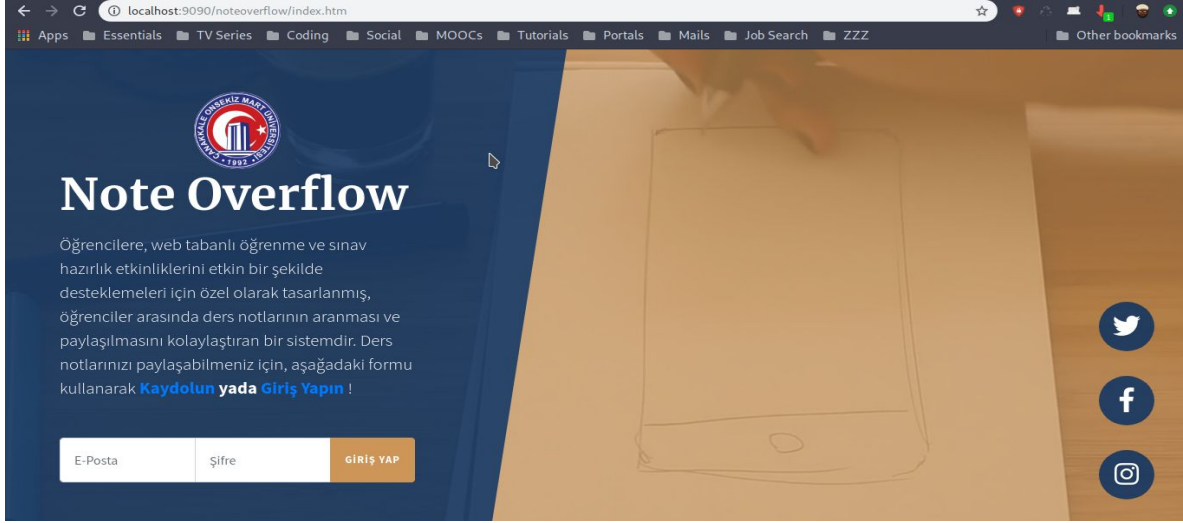
#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra	Action
1	ders_id	int(10)			No	None		AUTO_INCREMENT	Change Drop Primary More
2	ders_adi	varchar(50)	utf8mb4_unicode_ci		No	None			Change Drop Primary More
3	ders_hocasi	varchar(50)	utf8mb4_unicode_ci		No	None			Change Drop Primary More
4	ders_donemi	tinyint(1)			No	None			Change Drop Primary More

BÖLÜM 5

ARAYÜZ TASARIMI

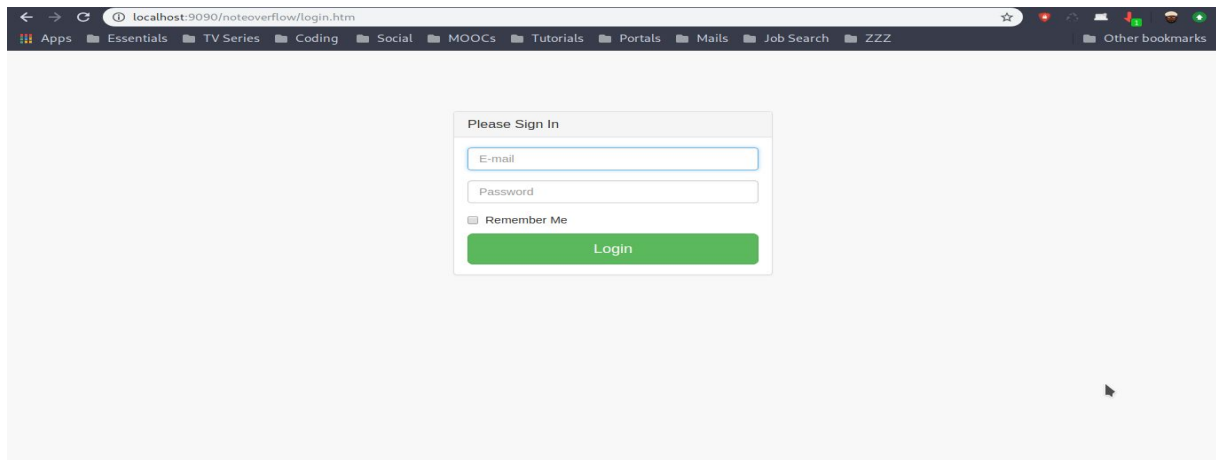
4.1 Arayüzler ve Kodları

Giriş Sayfası



```
34 </video>
35
36 <div class="masthead">
37   <div class="masthead-bg"></div>
38   <div class="container h-100">
39     <div class="row h-100">
40       <div class="col-12 my-auto">
41         <div class="masthead-content text-white py-5 py-md-0">
42           <div class="text-center">
43             
44           </div>
45           <h1 class="mb-3">Note Overflow</h1>
46           <p class="mb-5">
47             Öğrencilere, web tabanlı öğrenme ve sınav hazırlık etkinliklerini
48             etkin bir şekilde desteklemeleri için özel olarak tasarlanmış,
49             öğrenciler arasında ders notlarının aranması ve paylaşılmasını
50             kolaylaştıran bir sistemdir. Ders notlarınızı paylaşabilmeniz
51             için, aşağıdaki formu kullanarak <strong><a href="register.htm">Kaydolun</a>
52             yada <a href="login.htm">Giriş Yapın</a></strong> !
53           </p>
54           <form:form modelAttribute="loginParser" class="input-group input-group-newsletter" action="das
55             <form:input type="email" path="username" id="username" class="form-control" placeholder="E
56             <form:input type="password" path="password" id="password" class="form-control" placeholder
57             <div class="input-group-append">
58               <form:button class="btn btn-secondary" id="submitLogin" type="submit">GİRİŞ YAP</form:l
59             </div>
60           </form:form>
61         </div>
62       </div>
63     </div>
64   </div>
```

Login Sayfası



```
1 <div class="container">
2   <div class="row">
3     <div class="col-md-4 col-md-offset-4">
4       <div class="login-panel panel panel-default">
5         <div class="panel-heading">
6           <h3 class="panel-title">Please Sign In</h3>
7         </div>
8         <div class="panel-body">
9           <form role="form">
10            <fieldset>
11              <div class="form-group">
12                <input class="form-control" placeholder="E-mail" name="email" type="email">
13              </div>
14              <div class="form-group">
15                <input class="form-control" placeholder="Password" name="password" type="password">
16              </div>
17              <div class="checkbox">
18                <label>
19                  <input name="remember" type="checkbox" value="Remember Me">Remember Me
20                </label>
21              </div>
22              <!-- Change this to a button or input when using this as a form -->
23              <a href="index.html" class="btn btn-lg btn-success btn-block">Login</a>
24            </fieldset>
25          </form>
26        </div>
27      </div>
28    </div>
29  </div>
```

Login Parser Model

```

package com.noteoverflow.models;

/**
 *
 * @author trabdlkarim
 */
public class LoginParser {
    private String username;
    private String password;

    public String getUsername() {
        return username;
    }

    public void setUsername(String username) {
        this.username = username;
    }

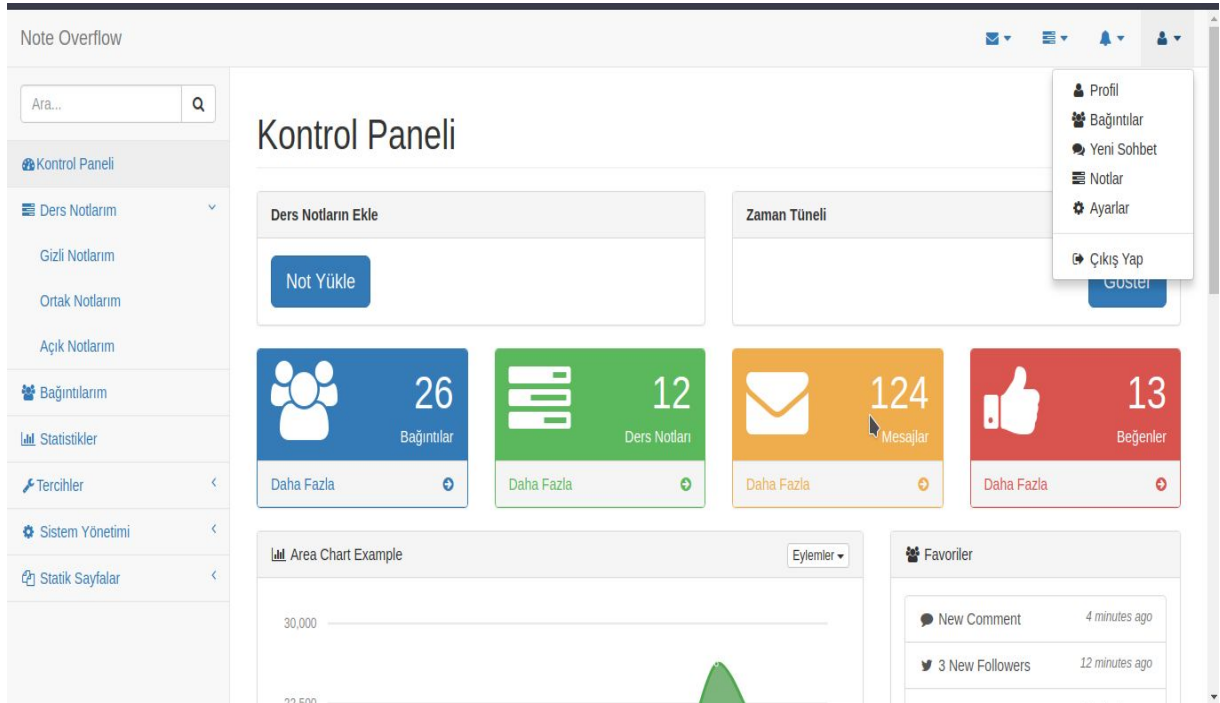
    public String getPassword() {
        return password;
    }

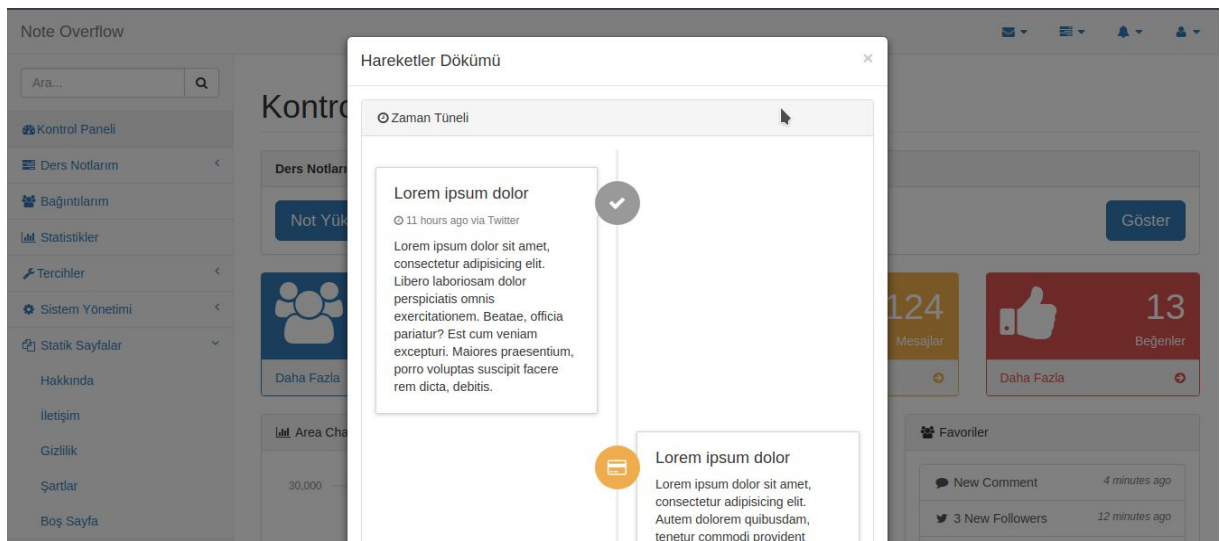
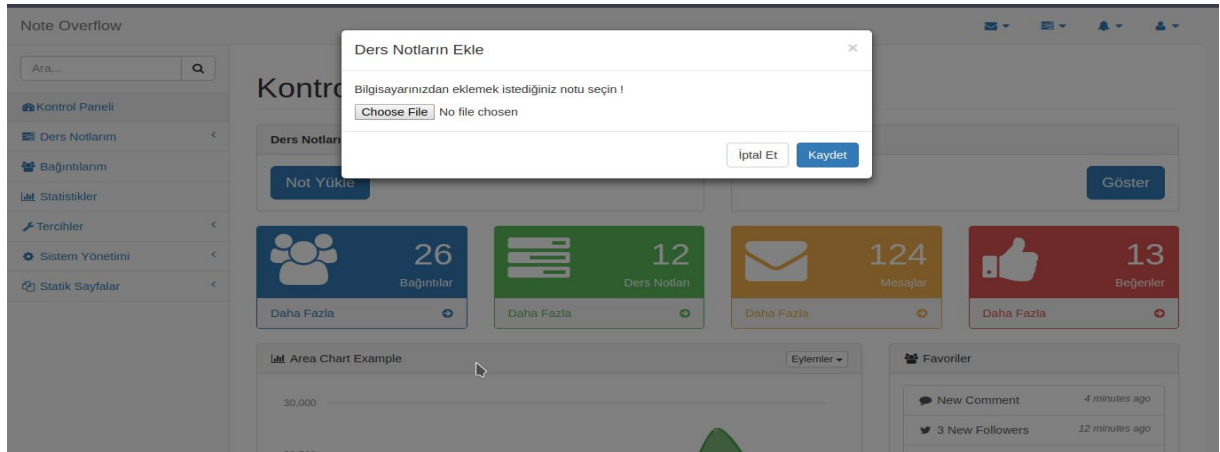
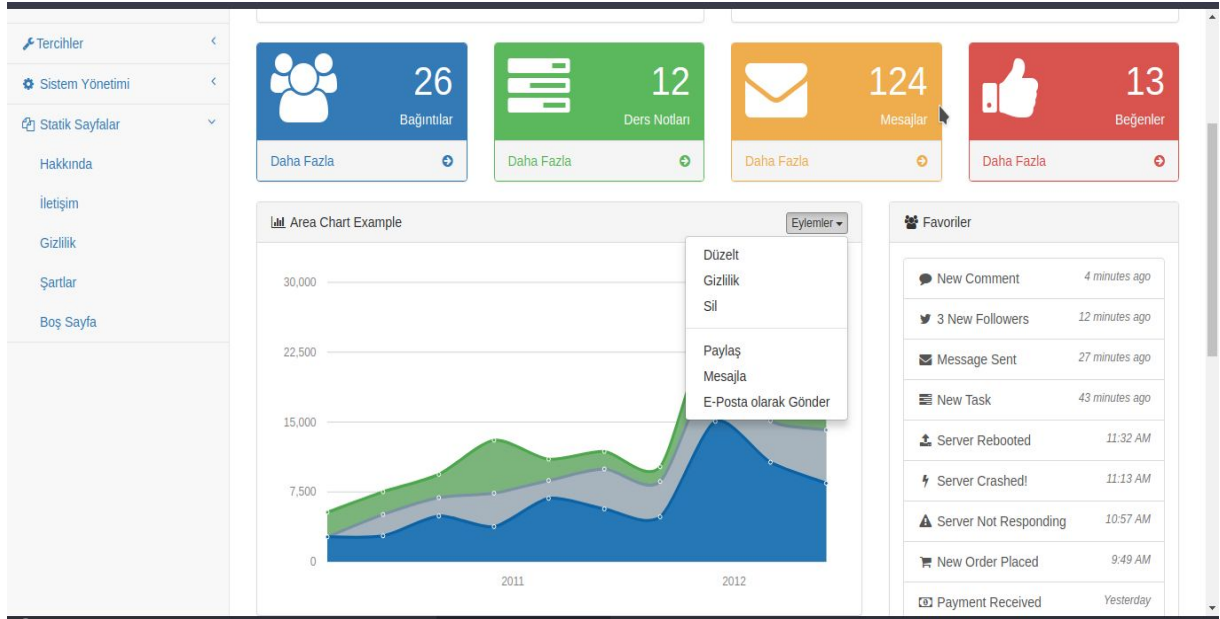
    public void setPassword(String password) {
        this.password = password;
    }
}

```

Anasayfa

Üye sisteme giriş yaptıktan sonra notları paylaşabileceği, başkalarının notlarına ulaşabileceği ve hareketler dökümünü görebileceği bu sayfa açılır.





```

<%@page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<%@taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@taglib uri="http://www.springframework.org/tags/form" prefix="form" %>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="utf-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <meta name="description" content="">
    <meta name="author" content="">

    <title>Note Overflow - Dashboard</title>

    <!-- Bootstrap Core CSS -->
    <link href="${pageContext.request.contextPath}/assets/sb-admin.vendor/bootstrap/css/bootstrap.min.css" rel="stylesheet">

    <!-- MetisMenu CSS -->
    <link href="${pageContext.request.contextPath}/assets/sb-admin.vendor/metisMenu/metisMenu.min.css" rel="stylesheet">

    <!-- Custom CSS -->
    <link href="${pageContext.request.contextPath}/assets/dist/css/sb-admin-2.css" rel="stylesheet">

    <!-- Morris Charts CSS -->
    <link href="${pageContext.request.contextPath}/assets/sb-admin.vendor/morrisjs/morris.css" rel="stylesheet">

    <!-- Custom Fonts -->
    <link href="${pageContext.request.contextPath}/assets/sb-admin.vendor/font-awesome/css/font-awesome.min.css" rel="stylesheet">

    <!-- HTML5 Shim and Respond.js IE8 support of HTML5 elements and media queries -->
    <!-- WARNING: Respond.js doesn't work if you view the page via file:// -->

```

Controllers

```

package com.noteoverflow.controllers;

import com.noteoverflow.models.LoginParser;
import org.springframework.stereotype.Controller;
import org.springframework.ui.ModelMap;
import org.springframework.web.bind.annotation.ModelAttribute;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestMethod;

@Controller
public class HomeController {

    @RequestMapping(value="index", method = RequestMethod.GET)
    public String renderHomeView(ModelMap model, @ModelAttribute("loginParser") LoginParser loginParser) {
        return "index";
    }
}

```



```

package com.noteoverflow.controllers;

import org.springframework.stereotype.Controller;
import org.springframework.ui.ModelMap;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestMethod;

@Controller
public class AccountController {
    @RequestMapping(value="login",method=RequestMethod.GET)
    public String renderLoginView(ModelMap model) {
        return "account/login";
    }
    @RequestMapping(value="dashboard",method=RequestMethod.POST)
    public String renderDashboardView(ModelMap model) {
        return "account/dashboard";
    }
    @RequestMapping(value="register",method=RequestMethod.GET)
    public String renderRegisterView(ModelMap model) {
        return "register";
    }
    @RequestMapping(value="notifications",method=RequestMethod.GET)
    public String renderNotificationView(ModelMap model) {
        return "account/notifications";
    }
}

```

```

package com.noteoverflow.controllers;

import org.springframework.stereotype.Controller;
import org.springframework.ui.ModelMap;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RequestMethod;

/**
 *
 * @author trabdlkarim
 */
@Controller
public class StaticPagesController {
    @RequestMapping(value="about", method = RequestMethod.GET)
    public String renderAboutView(ModelMap model) {
        return "static/about";
    }
    @RequestMapping(value="contact", method = RequestMethod.GET)
    public String renderContactView(ModelMap model) {
        return "static/contact";
    }
}

```

BÖLÜM 6

SONUÇ

Öncelikle projemizin amacını, kapsamını ve başarı kriterlerini belirledik. Daha sonra projemiz için gereksinim analizi yaptık. Gereksinim analizinde sistem ve üye için gereksinimler belirledik. Bu adımdan sonra gereksinimlerimizi fonksiyonel ve fonksiyonel olmayan olarak ikiye ayırdık ve bunları belirledik. Gereksinim analizi aşamasında sonra sistem modellemeye bölümüne geçtik. Burada öncelikle actor-goal modelimizle sistem ve üyenin görevlerini belirledik. Ardından projemiz için en kritik üç senaryoyu belirledik. Bunlar üyenin sisteme not eklemesi, sistem yöneticisinin notların paylaşılmasına izin vermesi ve üyenin sistemdeki notlara ulaşmasıdır . Görsel olarak daha anlaşılır olması için her üç senaryo için de use case modellerimizi oluşturduk. Projemizin daha anlaşılır olması için diagramlar kullanmayı tercih ettik. Bu yüzden use caselerimize SSD ve State Machine diagramlarla gösterdik ve domain modellerle sistem modelleme aşamasını bitirdik. Veri tabanımızı oluşturduk. Buna uygun tablolar oluşturduk. Yaptığımız analizler sonucunda 9 adet tablo oluşturduk . Projemizin ilerleyen zamanlarında tablolar güncellenebilir. Projemizin arayüz tasarımını oluşturduk.

KAYNAKLAR

Bazı Youtube Videoları
Java Kitabı-Mehmet Kirazlı –Kodlab