

KARABÜK ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ BİLGİSAYAR MÜHENDİSLİĞİ



CME426 – YAZILIM MÜHENDİSLİĞİ DERSİ
PROJE ÖDEVİ

Grup Proje Sorumlusu: Ökkeş KILINÇ	2014210217010
Emrah BAYINDIR	2013010217062
Burak Han KAPTAN	2013010217010

PROJE KONUSU:

S.14. Bir Test Merkezinde konular, alt konular, hazırlanan sorular, soru yazarları ve soru denetleyicileri ile ilgili bilgilerin yer aldığı bir Test Veri Tabanı oluşturulmak isteniyor. Oluşturulacak veri tabanı ile ilgili kısa açıklamalar aşağıda yer almaktadır.

- Konular (tarih, coğrafya, matematik, vb): her konunun numarası (biricik) ve adı.
- Her konunun alt konuları (tarih için: Türk tarihi, Osmanlı tarihi. Kurtuluş savaşı ve TC tarihi, vb) : her alt konunun numarası ve adı. Alt konu numaraları konu içinde biricik olan sıra numaralarıdır.
- Test merkezinde görev yapan her soru yazarı ve denetleyicinin: TC kimlik numarası (biricik), adı, soyadı, cinsiyeti ve adresi. Merkezde görevli kişiler ya soru yazarı ya da denetleyicidir. Soru yazarı olarak öğretmenler, denetleyici olarak da öğretim üyeleri görev yapmaktadır. Kimlik ve adres bilgilerine ek olarak veri tabanında soru yazarlarının mezuniyet yılı, denetleyicilerin de akademik ünvan bilgilerinin bulunması istenmektedir.
- Test merkezinde görevli her kişinin uzmanlık alanı (her soru yazarı ve denetleyici bir ve yalnız bir konuda uzmandır).
- Kurslar: her kursun numarası (biricik), adı, başlangıç tarihi ve süresi.
- Her soru yazarının hangi kurslara katıldığı ve katıldığı her kursta elde ettiği başarı. Soru yazarı olabilmek için en az bir kursa katılmak gereklidir. İsteyenler birden çok kursa da katılabilmektedir.
- Üniversiteler: her üniversitenin numarası (biricik) ve adı.
- Her denetleyicinin hangi üniversitede çalıştığı.
- Sorular: yazılan her test sorusunun numarası (biricik), soru metni, doğru cevap, zorluk derecesi, yazarı (her soru bir ve yalnız bir soru yazarı tarafından yazılmaktadır) ve alt konusu (her soru bir ve yalnız bir alt konu ile ilgilidir).
- Yazılan her sorunun hangi denetleyiciler tarafından hangi tarihlerde denetlendiği ve her denetleyicinin soruyla ilgili görüşü. Soruların denetlenmeden veri tabanına girilmesi; daha sonra denetimler yapıldıkça da bu bilgilerin eklenmesi istenmektedir. Her soru en çok 3 denetleyici tarafından denetlenmektedir.

Yukarıdaki açıklamalar ışığında (açıklamalarda yer almayan konular için, gerekiyorsa, uygun varsayımlar yaparak) projenin model ve tasarımını oluşturunuz.

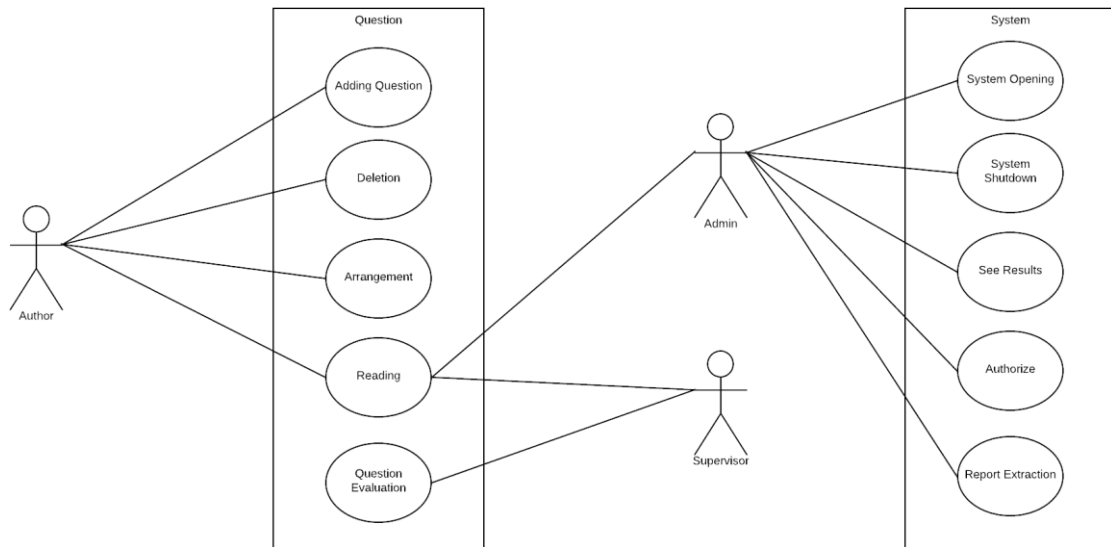
-Proposed process model for the project: For this project; "*The V-Model*" is suitable. Because we can control the project with The V-Model. We can know how to test the stages. In this way it is easier to catch and correct errors. Because of the requirements in this project are quite evident, it would be appropriate to prefer the V model used in this projects where uncertainties are low.

Or "*Incremental Model*" bu can be selected for this project. Because the risk analysis and installation were deemed appropriate. This is because, first, a core system is developed and requirements are determined together with the customers. One of these two models can be preferred.

-Requirements:

1. The Writer should be able to see the status of the questions, add or delete questions according to the situation.
2. Admin should be able to access the questions written by the authors and read the questions.
3. Admin should be able to control the system. In other words, it should be able to switch the system on and off.
4. Admin should authorize Writer and Supervisor actors when necessary.
5. The Writer should be able to make arrangements to correct the errors in the questions.
6. The Supervisor should be able to check and read the questions at all times in order to fulfill its duties.
7. The Supervisor must have access to topics and sub-topics.
8. Admin must contain all the information of the Writer and Supervisor.
9. The university information of the Supervisors and the graduation years of the Writers should be known.
10. Admin should be able to contact the Supervisor and approve the questions.

-Use-case Diagram:



System Opening: Open access to add questions or evaluate questions.

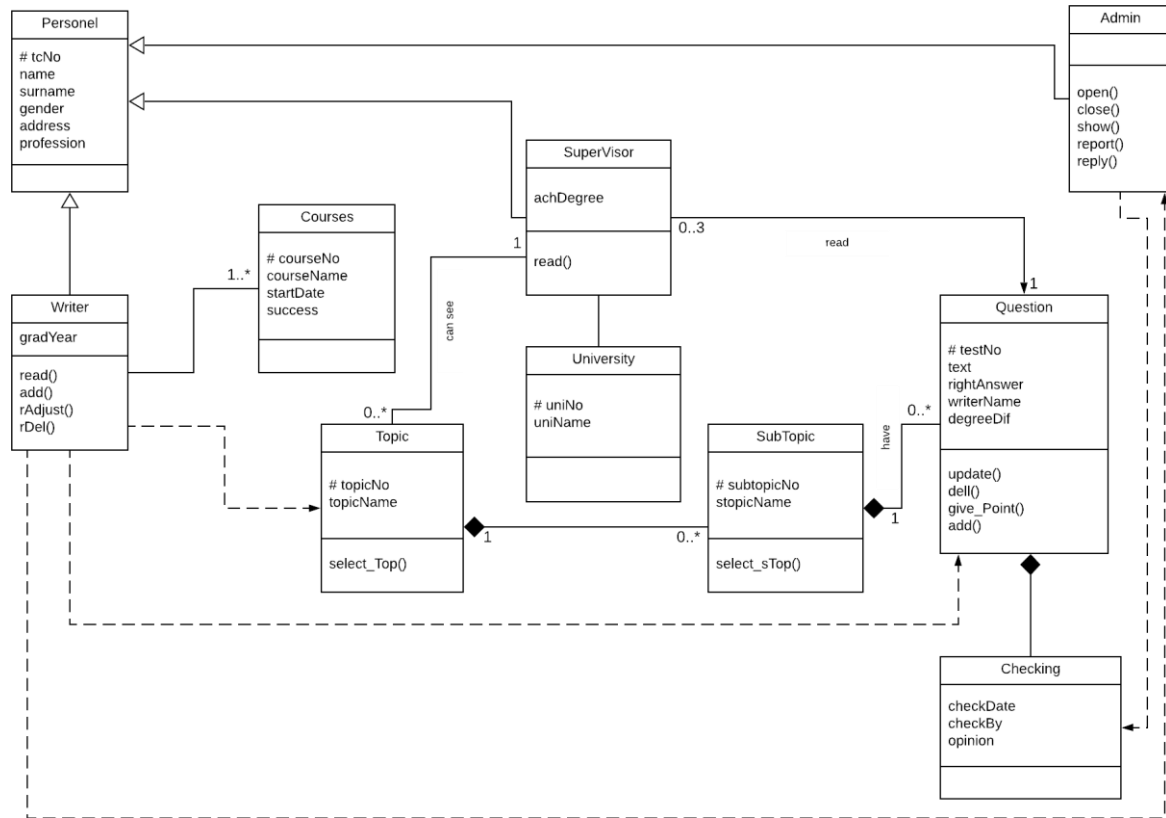
System Shutdown: Prevents the system from adding questions or evaluating questions. It can only read.

Deletion, Arrangement: for these operations, you need the admin approval first.

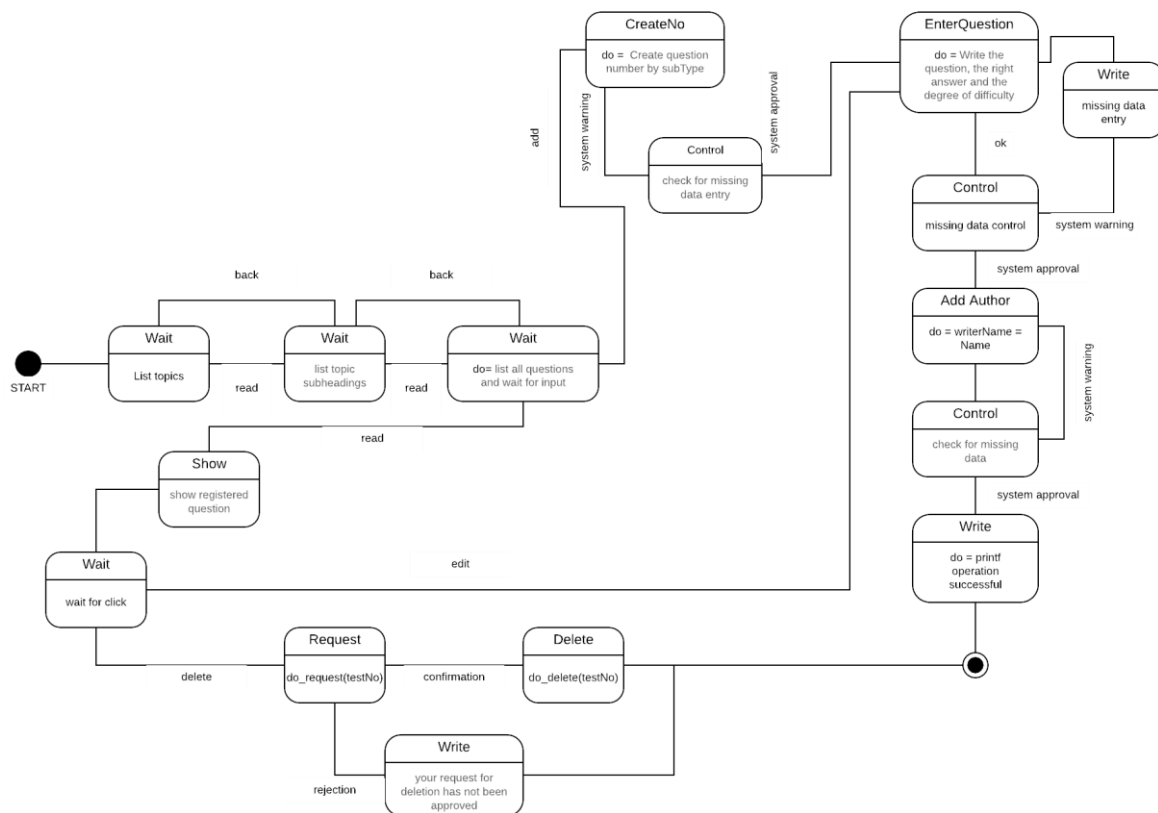
Authorize: It provides authorization for editing or deletion when necessary.

Report Extract: A collective report of each author's questions and the values it receives is issued.

-Class Diagram:



-State Diagram for “Writer”:



State	Description
-------	-------------

Wait:	The system waits for an input and lists the desired thing.
--------------	--

Show:	The system lists the desired content and expects input according to the need.
--------------	---

CreateNo:	The system automatically generates a question number based on the sub-topic.
------------------	--

Write:	The system prints a text on the screen.
---------------	---

Contol:	The system checks if important places are full.
----------------	---

Enter Question:	The author enters question, the correct answer and the degree of difficulty on the screen.
------------------------	--

Add Author:	The system adds the person who created the question and the creation date.
--------------------	--

Request:	Admissions are requested from the administrator.
-----------------	--

Delete:	The system executes the deletion function with the testNo it receives for the deletion.
----------------	---

Stimulus

Description

Add: User presses the add button.

Read: User presses the read button.

OK: User presses the ok button.

System Approval: The system gives the approval for the next system as a result of the check.

System Warning: The system does not transmit the operation due to missing data.

Deletion: The system performs the deletion.

Edit: User presses the edit button.

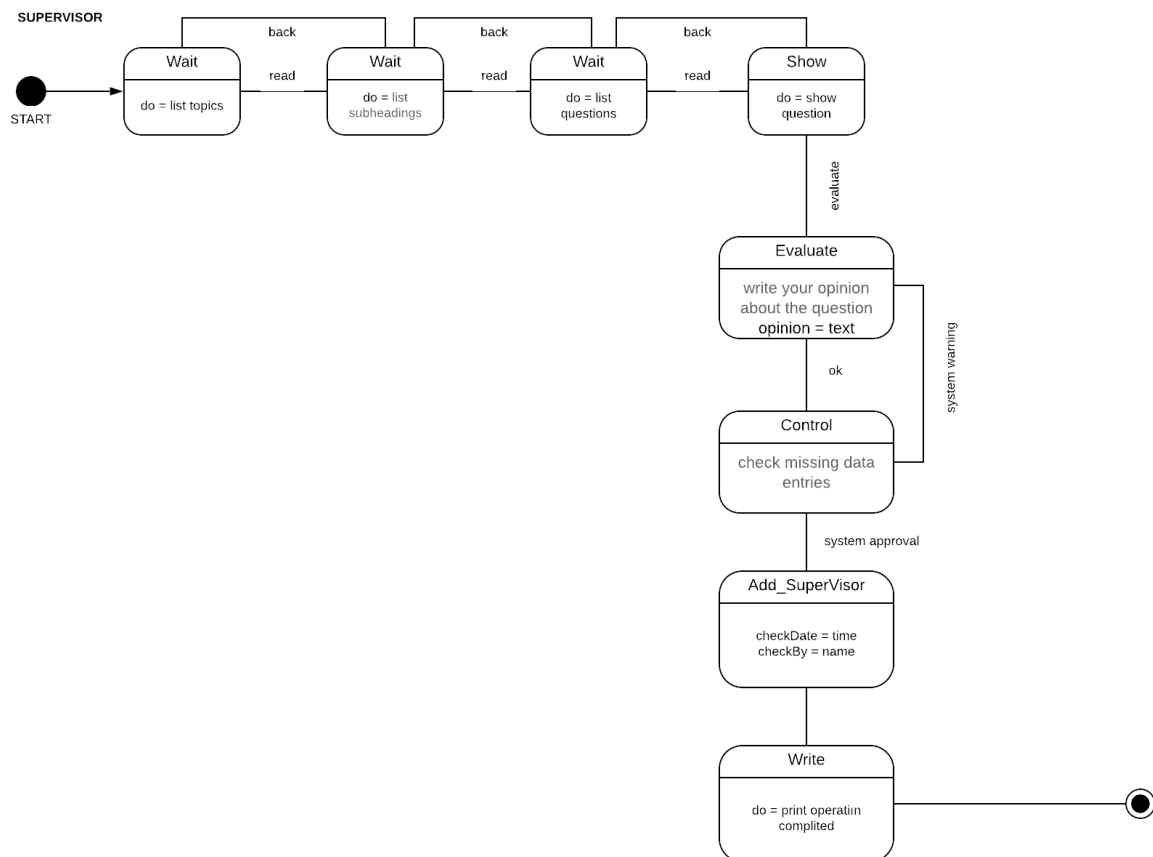
Return: User presses the return button.

Approval: Admin authorizes the approval.

Rejection: Admin has not authorized the operation.

Back: User presses the back button.

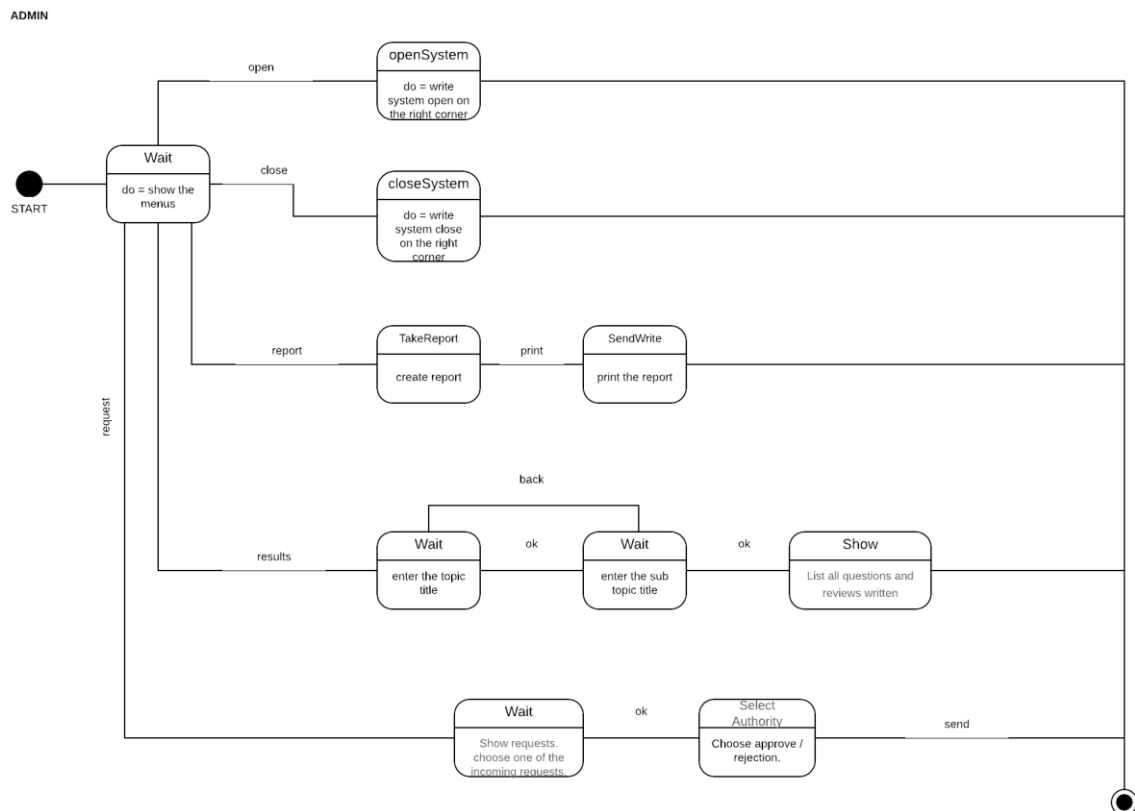
-State Diagram for "Supervisor":



<u>State</u>	<u>Description</u>
Wait:	The system waits for an input and lists the desired thing.
Show:	The system lists the desired content and expects input according to the need.
Evaluation:	The Supervisor enters his Opinion on the question into the system.
Control:	The system checks if the data is entered where the entries are required.
Add_Supervisor:	The system adds the name of the Supervisor that makes the Opinion to the control time.
Write:	The system prints text on the screen.

<u>Stimulus</u>	<u>Description</u>
OK:	User presses the OK button.
Back:	User presses the back button.
Read:	User presses the read button.
Evaluate:	User presses the evaluate button.
System Approval:	The system gives confirmation if the required places are full.
System Warning:	The system does not give approval if the required places are not full.

-State Diagram for “Admin”:



State	Description
-------	-------------

Wait: The system waits for an input.

OpenSystem: The system should be open to make a question or evaluate the system.

CloseSystem: The system is not allowed to add or evaluate questions.

TakeReport: Questions and assessments added to the system are collected.

SendWrite: Output is sent.

Show: The requested data is listed.

SelectAuthority: A confirmation or rejection message is sent to the incoming Request.

Stimulus	Description
----------	-------------

Open: Admin clicks the open button.

Close: Admin clicks the close button.

Report: Admin clicks the report button.

Results: Admin clicks the results button.

Request: Admin clicks the request button.

Print: Admin clicks the print button.

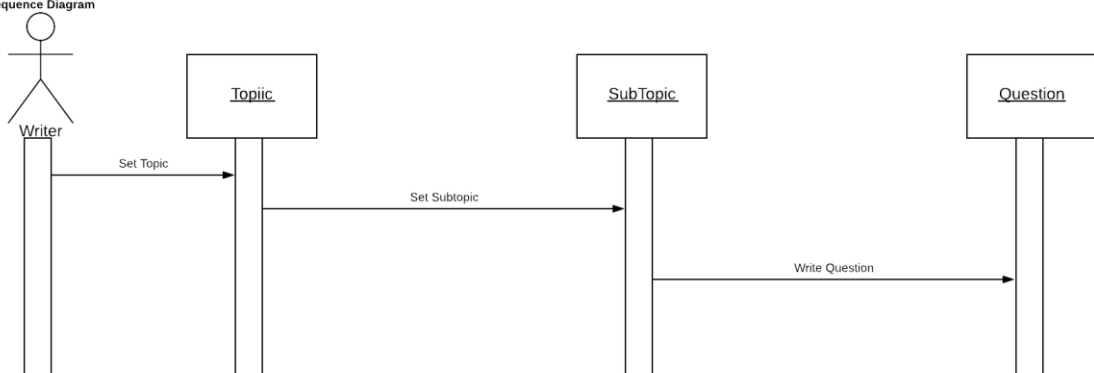
OK: Admin clicks the OK button.

Back: Admin clicks the back button.

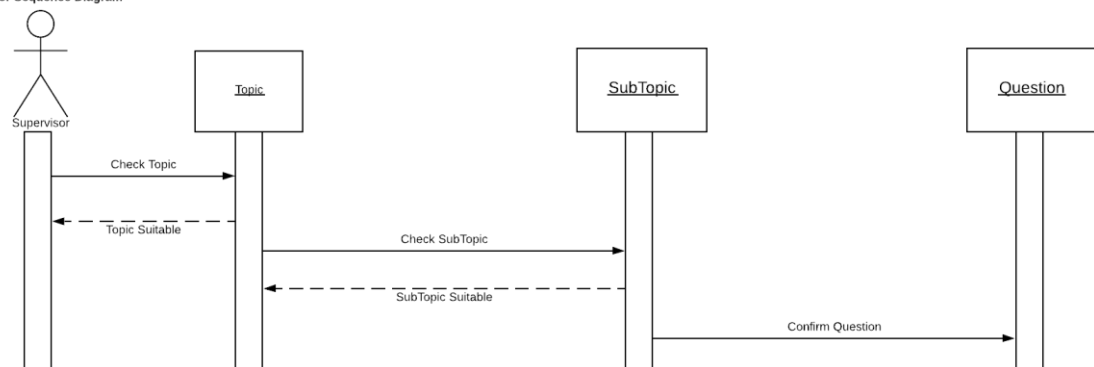
Send: Admin clicks the send button.

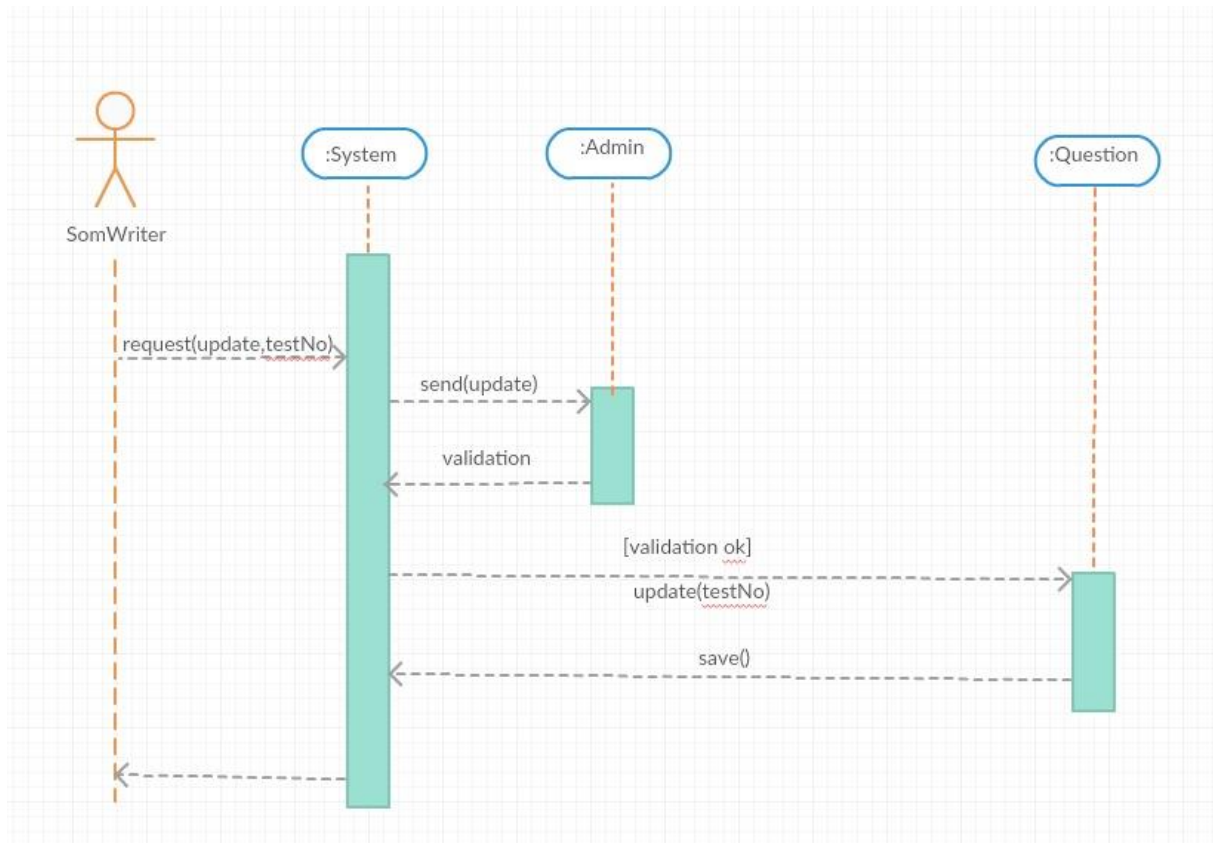
-Sequence Diagram for “Writer” and “Supervisor”:

Writer Sequence Diagram



Supervisor Sequence Diagram





-Code of the project:

```

import os
import tkinter
import sqlite3

index=999

sqlconnection=sqlite3.connect('softwarehomeworkdb.db')
sqlcursor=sqlconnection.cursor()

if sqlcursor:
    print("connection successfull")
else:
    print("connection failed")

def guncelle(zorluk,dTarih,dGorus,denetID,sorulId):

    sqlcursor.execute("UPDATE soru SET zorluk=? , denetleme_tarihi=? , denetleyici_id=? ,
denetleyici_gorus=? WHERE Id=?" ,(zorluk,dTarih,denetID,dGorus,sorulId))

    sqlconnection.commit()
  
```

def soruekleme():

metin=input("soru metnini giriniz: ")

yazar=str(input("yazar adi:"))

tc=sqlcursor.execute("SELECT tc_no FROM yazar WHERE adi=?", (yazar,))

tc = tc.fetchone()[0]

cevap=input("cevap giriniz: ")

zorluk=int(input("1-5 zorluk seviyesi giriniz: "))

altkonu=str(input("altkonu adi:"))

alt_konu_id=sqlcursor.execute("SELECT id FROM alt_konu WHERE adi=?", (altkonu,))

alt_konu_id = alt_konu_id.fetchone()[0]

denet_tarih=input("Denetleme tarihi: ")

denetleyici=str(input("Denetleyici : "))

denet_id=sqlcursor.execute("SELECT tc_no FROM denetleyici WHERE adi=?", (denetleyici,))

denet_id = denet_id.fetchone()[0]

denet_gorus=input("Denetleyici gorusu: ")

```
sqlcursor.execute("INSERT INTO soru
(metin,yazar_id,cevap,zorluk,alt_konu_id,denetleme_tarihi,denetleyici_id,denetleyici_gorus)"
" VALUES
(?,?,?,?,?,?,?,?)", (metin,tc,cevap,zorluk,alt_konu_id,denet_tarih,denet_id,denet_gorus))
```

def konuekleme():

konu=input("Lutfen konu basligini giriniz: ")

sqlcursor.execute("INSERT INTO konu (adi) VALUES (?)", (konu,))

def kursekleme():

ad=input("Kursun Adi: ")

baslangic=input("Baslangic tarihi: ")

sure=int(input("Kursun suresi: "))

```
sqlcursor.execute("INSERT INTO kurs (adi,baslangic_tarih,sure) VALUES (?,?,?)",(ad,baslangic,sure))
```

```
def uniekleme():
```

```
    adi=input("Universite Adi: ")
```

```
    sqlcursor.execute("INSERT INTO universite (adi) VALUES (?)",(adi,))
```

```
def altkonuekleme():
```

```
    sql = "SELECT * FROM konu"
```

```
    listeleme(sql)
```

```
    konuadi=str(input("konu basligi seciniz: "))
```

```
    alt_konu=input("alt konu basligi: ")
```

```
    id=sqlcursor.execute("SELECT id FROM konu WHERE adi=?", (konuadi,))
```

```
    id = id.fetchone()[0]
```

```
    sqlcursor.execute("INSERT INTO alt_konu (adi,konu_id) VALUES (?,?)",(alt_konu,id))
```

```
def yazarekleme():
```

```
    tcNo=int(input("TC kimlik: "))
```

```
    mezuniyet=int(input("Mezuniyet: "))
```

```
    ad=input("Adi:")
```

```
    soyad=input("Soyadi:")
```

```
    adres=input("adres:")
```

```
    cinsiyet=input("Cinsiyet:")
```

```
    uzmanlik=input("Uzmanlik:")
```

```
    sqlcursor.execute("INSERT INTO yazar  
(tc_no,adi,soyadi,cinsiyet,adres,mezuniyet_yili,uzmanlik_alani) VALUES  
(?,?,?,?,?,?,?)",(tcNo,ad,soyad,cinsiyet,adres,mezuniyet,uzmanlik))
```

```
def denetleyiciekleme():
```

```
    tcNo=int(input("TC kimlik: "))
```

```
    ad=input("Adi:")
```

```
    soyad=input("Soyadi:")
```

```
    cinsiyet=input("Cinsiyet:")
```

```
adres=input("adres:")
akademik=input("Akademik: ")
uzmanlik=input("Uzmanlik:")
```

```
uni=str(input("Universite adi:"))
id=sqlcursor.execute("SELECT id FROM universite WHERE adi=?", (uni,))
id = id.fetchone()[0]
```

```
sqlcursor.execute("INSERT INTO denetleyici
(tc_no,adi,soyadi,cinsiyet,adres,akademik_unvan,uzmanlik_alani,uni_id) VALUES
(?,?,?,?,?,?,?,?)", (tcNo,ad,soyad,cinsiyet,adres,akademik,uzmanlik,id))
```

```
def listeleme(sql):
    sqlcursor.execute(sql)
    result = sqlcursor.fetchall()
    for rs in result:
        print(rs)
```

```
def sorudenetleme():
    sql = "SELECT * FROM soru"
    sql1 = "SELECT * FROM denetleyici"
```

```
listeleme(sql)
sld=int(input("id sini giriniz: "))
sqlcursor.execute("SELECT zorluk,denetleme_tarihi,denetleyici_id,denetleyici_gorus FROM soru
WHERE Id=?", (sld,))
rs=sqlcursor.fetchone()
print(rs)
zorluk=int(input("1-5 arasinda zorluk seviyesi: "))
dTarih=input("Denetleme Tarihi: ")
dGorus=input("Denetleyici Gorus: ")
listeleme(sql1)
```

```
denetleyici = str(input("Denetleyici adi:"))  
denetID = sqlcursor.execute("SELECT tc_no FROM denetleyici WHERE adi=?", (denetleyici,))  
denetID = denetID.fetchone()[0]
```

```
guncelle(zorluk,dTarih,dGorus,denetID,sld)
```

```
def yazarislemleri():
```

```
    print("***50)  
    print("Yazar Listele ->1 \nYazar Ekleme -> 2")  
    print("***50)  
    yazarsecim=int(input("islem seciniz: "))  
    if yazarsecim==1:  
        sql = "SELECT * FROM yazar"  
        listeleme(sql)  
    elif yazarsecim==2:  
        yazarekleme()
```

```
def denetlemeislemleri():
```

```
    print("***50)  
    print("Denetleyici Listele ->1 \nDenetleyici Ekleme -> 2 \nSoru Denetleme -> 3")  
    print("***50)  
    denetlemesecim=int(input("islem seciniz: "))  
    if denetlemesecim==1:  
        sql = "SELECT * FROM denetleyici"  
        listeleme(sql)  
    elif denetlemesecim==2:  
        denetleyiciekleme()  
    elif denetlemesecim==3:  
        sorudenetleme()  
    else:  
        print("Yanlis secim!!!")
```

```
def soruislemleri():  
    print("*" * 50)  
    print("Soru Listele ->1 \nSoru Ekleme -> 2 \nKonu Ekleme -> 3 \nAlt Konu Ekleme -> 4")  
    print("*" * 50)  
    soruSecim=int(input("Secim yapiniz: "))  
    if soruSecim==1:  
        sql = "SELECT * FROM soru"  
        listeleme(sql)  
    elif soruSecim==2:  
        soruekleme()  
    elif soruSecim==3:  
        konuekleme()  
    elif soruSecim==4:  
        altkonuekleme()  
    else:  
        print("Yanlis secim!!!")
```

```
while True:  
    clear=lambda:os.system('cls')  
    clear()  
    print("*"*50)  
    print("Soru islemleri -> 1 \nYazar Islemleri -> 2 \nDenetleme Islemleri -> 3 \nKurs Islemleri -> 4  
    \nCIKIS -> 0")  
    print("*" * 50)  
    secim=int(input("Seciminizi yapiniz: "))  
    if secim==1:  
        soruislemleri()  
    elif secim==2:  
        yazarislemleri()
```


elif secim==3:

denetlemeislemleri()

elif secim==4:

kursekleme()

else:

break

sqlconnection.commit()

sqlconnection.close()

```
connection successfull
*****
Soru islemleri -> 1
Yazar Islemleri -> 2
Denetleme Islemleri -> 3
Kurs Islemleri -> 4
CIKIS -> 0
*****
Seciminizi yapiniz: 1
*****
Soru Listele ->1
Soru Ekleme -> 2
Konu Ekleme -> 3
Alt Konu Ekleme -> 4
*****
Secim yapiniz: 4
(1, 'matematik')
(2, 'edebiyat')
(3, 'tarih')
(4, 'biyoloji')
(5, 'cografya')
(6, 'turkce')
konu basligi seciniz:
```

h 6: TODO Terminal Python Console

Table: alt_konu

	id	adi	konu_id
	Filtre	Filtre	Filtre
1	1	dilbilgisi	6
2	2	sayilar	1
3	3	denklemler	1
4	4	Osmanli Edebi...	2
5	5	TC edebiyati	2
6	6	TC tarihi	3
7	7	Osmanli tarihi	3
8	8	Tek hucreliler	4
9	9	Cok hucreliler	4
10	10	Harita bilgisi	5
11	11	Toprak tipleri	5
12	12	Fiil	6
13	13	Zamir	6

Table: konu

	id	adi
	Filtre	Filtre
1	1	matematik
2	2	edebiyat
3	3	tarih
4	4	biyoloji
5	5	cografya
6	6	turkce

Table: **kurs**

	id	adi	baslangic_tarih	sure
	Filtre	Filtre	Filtre	Filtre
1	1	Tarih Kursu	01.06.2019	120
2	2	Matematik kursu	2.06.2019	240
3	3	Cografya Kursu	2.06.2019	180

Database Structure Browse Data Edit Pragmas Execute SQL

Table: **yazar**

	tc_no	adi	soyadi	cinsiyet	adres	mezuniyet_yili	uzmanlik_alani
	Filtre	Filtre	Filtre	Filtre	Filtre	Filtre	Filtre
1	123	ahmet	burdami	e	antep	1234	Tarih
2	1234567890	Emrah	Bayindir	E	Istanbul	2019	Edebiyat
3	1111111111	Burak	Kaptan	E	Karabuk	2019	Biyoloji
4	2222222222	Okkes	Kilinc	E	Gaziantpe	2019	Edebiyat

Table: **universite**

	id	adi
	Filtre	Filtre
1	1	Karabuk
2	2	Gaziantep
3	3	Ankara
4	4	Gebze

	id	metin	yazar_id	cevap	zorluk	alt_konu_id	denetleme_tarihi	denetleyici_id	denetleyici_gorus
	Filtre	Filtre	Filtre	Filtre	Filtre	Filtre	Filtre	Filtre	Filtre
1	1	MATEMATİK ...	22222222222	Evet mat sorusu	0	2	02.06.2019	1234	bu matematik sorusu çok güzel olmuş eline sağlık :D
2	2	TARİH SORUS...	1234567890	evet tarih sorusu	0	6	02.06.2019	1234	Bu tarih sorusu acayip
3	3	COĞRAFYA S...	11111111111	evet coğrafya sor...	0	10	02.06.2019	1234	bu kadarbu kadar kus bakisi coğrafya sorusu hazırlanamazdı

	tc_no	adi	soyadi	cinsiyet	adres	akademik_unvan	uzmanlik_alani	uni_id
	Filtre	Filtre	Filtre	Filtre	Filtre	Filtre	Filtre	Filtre
1	1234	ali	DenetleGelOglu	E	DenetçilerBoyu	Ogr. Uyesi	Tarih	1
2	33333333333	Hazal	Denetleyici	K	Denetleyiciler sitesi	Ogr. Uyesi	Coğrafya	4
3	44444444444	Halil	Denetleyici Oglu	E	Urfa / Denetleyiciler	Dr	Matematik	1
4	55555555555	Mehmet	DenetlemezOglu	E	Denetleyiciler karsisi	Ogr. Uyesi	Biyoloji	3