# 

|  |  |  |
| --- | --- | --- |
| tekno6 |  | Selcuk_SONNN |

**T.C.**

**SELÇUK ÜNİVERSİTESİ**

**TEKNOLOJİ FAKÜLTESİ**

**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

|  |  |  |
| --- | --- | --- |
|  | **EBEVEYN SİTESİ VERİTABANI**  **SEMİH ÜNAL**  **ADEM ENES POLAT** |  |

**MAYIS-2019**

**KONYA**

**Her Hakkı Saklıdır**

**İÇİNDEKİLER**

[1.İHTİYAÇ ANALİZİ 1](#_Toc514105328)

[1.1 Sistem Hizmet Alanları 1](#_Toc514105329)

[1.2 Sistem Hangi İhtiyaçları Karşılar? 1](#_Toc514105330)

[1.3 Sistem Hangi Verileri Depolar? 1](#_Toc514105331)

[1.3 Sistem Hangi Tablolardan Oluşuyor? 1](#_Toc514105332)

[2.VARLIK İLİŞKİ ŞEMASI 2](#_Toc514105333)

[3.TABLOLARIN OLUŞTURULMASI 3](#_Toc514105334)

# 1.İHTİYAÇ ANALİZİ

## 1.1 Sistem Hizmet Alanları

Ebeveyn sitesi veritabanı, ebeveynlerin siteye üye olmaları için gerekli bilgileri, siteye kayıtlı ebeveynlerin kişisel bilgilerini, post paylaşım bilgilerini, postların like, dislike sayısı bilgilerini, diğer kişilerin postlarına yaptıkları yorumların bilgileri tutan sistemdir.

## 1.2 Sistem Hangi İhtiyaçları Karşılar?

Ebeveyn sitesi veritabanı sistemi siteye kayıtlı ebeveynlerin paylaşımlarının bilgilerini, bu paylaşımlara yapılan yorumların içeriklerinin bilgilerini, sakıncalı içeriklerin spam durumlarının bilgilerini tutar. Bu sistem sayesinde sitenin daha kullanışlı olmasını sağlamaktadır.

## 1.3 Sistem Hangi Verileri Depolar?

Ebeveyn sitesi veritabanı, ebeveynlerin bilgilerini, oluşturdukları tartışma ve bilgilendirme postlarını, postlarını faydalı bulan veya faydalı bulmayan kullanıcıları ve bu kullanıcıların sayısını, yetkili kişiler tarafından paylaşılan çocuk ebeveyn ilişkisi hakkındaki bilgileri depolamaktadır. Bu bilgilerin yanı sıra siteye ait admin bilgileri tutulmaktadır ve bu adminler tarafından gerçekleştirilen işlemlerin bilgileri tutulmaktadır.

## 1.3 Sistem Hangi Tablolardan Oluşuyor?

Ebeveyn sitesi veritabanı; admin, comments, commentsreply, news, posts, spam, users, likes tablolarından oluşmaktadır.

admin ( adminId, userId, status, authority )

comments ( id,userId, postId, text, addedDate, modifiedDate )

commentsreply ( id, commentId, userId, text, addedDate, modifiedDate, userBy )

news ( newsId, userId, title, text, comments, state, datetime)

posts ( postId, userId, text, usefulNumber , notusefulNumber, newId, datetime )

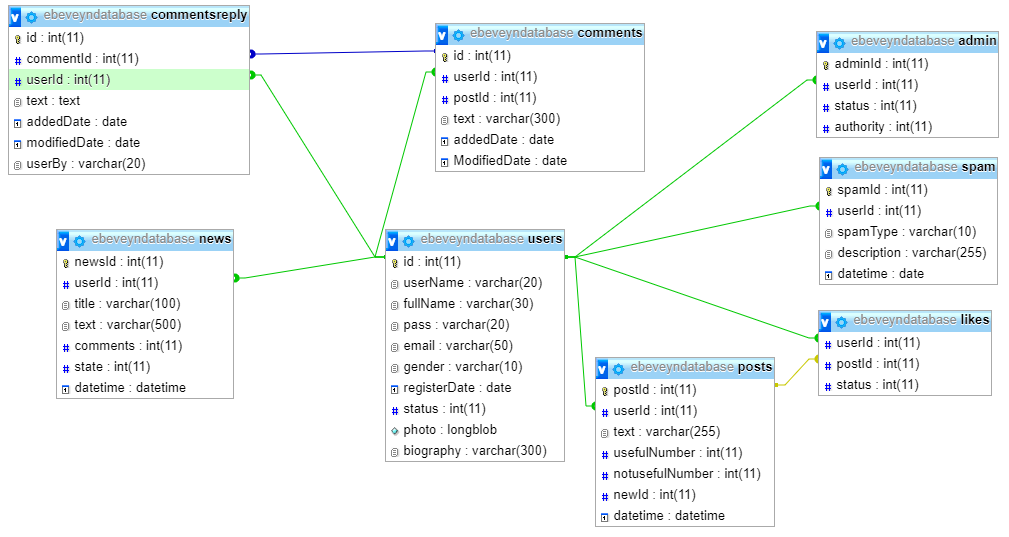
spam ( spamId, userId, spamType, description, datetime)

users ( id, userName, fullName, pass, email, gender, registerDate, status, photo, biography)

likes ( userId, postId, status)

# 2.VARLIK İLİŞKİ ŞEMASI

Ebeveyn Sitesi veritabanının varlık ilişki şeması aşağıdaki gibidir.



**Şekil 2.1.** Varlık İlişki Şeması

# 3.TABLOLARIN OLUŞTURULMASI

Veritabanında Users, Comments, CommentsReply, Posts, News, Admin ve Spam isimlerinde tablolar oluşturuldu. Tabloları oluşturmak için şu kodlar kullanıldı.

CREATE TABLE users( id int(11) PRIMARY KEY NOT NULL AUTO\_INCREMENT,

userName varchar(20) NOT NULL,

fullName varchar(30) NOT NULL,

pass varchar(20) NOT NULL,

email varchar(50) NOT NULL,

gender varchar(10),

registerDate date NOT NULL,

status int(11)

photo longblob,

biography varchar(300));

CREATE TABLE comments( id int(11) PRIMARY KEY NOT NULL AUTO\_INCREMENT

Foreign Key (userId) REFERENCES users(id),

Foreign Key(postId) REFERENCES posts(postId),

text varchar(300),

addedDate date,

modifiedDate date);

CREATE TABLE commentsreply( id int(11) PRIMARY KEY NOT NULL AUTO\_INCREMENT, Foreign Key (comment) REFERENCES comments(id),

Foreign Key (userId) REFERENCES users(id),

text varchar(300),

addedDate date,

modifiedDate date,

userBy varchar(20));

CREATE TABLE spam( spamId int(11) PRIMARY KEY NOT NULL AUTO\_INCREMENT,

Foreign Key (userId) REFERENCES users(id),

spamType int(10),

description varchar(255),

datetime date);

CREATE TABLE news( newsId int(11) PRIMARY KEY IDENTITY(1,1),

Foreign Key (userId) References users(id),

title varchar(100),

text varchar(500),

comments int(11),

state int(11),

datetime datetime);

CREATE TABLE admin( adminId int(11) PRIMARY KEY NOT NULL AUTO\_INCREMENT,

userId int(11),

status int(11),

authority int(11));

CREATE TABLE posts( postId int(11) PRIMARY KEY NOT NULL AUTO\_INCREMENT,

Foreign Key (userId) References users(id),

text varchar(255),

usefulNumber int(11),

notusefulNumber int(11),

newId int(11),

datetime datetime);

CREATE TABLE likes( Foreign Key (userId) References users(id),

Foreign Key (postId) References posts(postId),

status int(11));