

APİLER VE ÇIKTILARI

Students

GET	https://localhost:7262/api/students	Send
POST	https://localhost:7262/api/students	Send
PUT	https://localhost:7262/api/students	Send
DELETE	https://localhost:7262/api/students	Send

```
{
  "studentID": 10001,
  "firstName": "Ahmet",
  "lastName": "Yılmaz",
  "email": "ahmet.yilmaz@ogr.ktu.edu.tr",
  "advisorID": 1,
  "advisor": {
    "fullName": "Prof. Dr. Hamdi Tolga Kahraman",
    "title": "Bölüm Başkanı",
    "department": "Yazılım Mühendisliği Bölümü"
  },
  "courses": [
    {
      "courseID": 23,
      "courseName": "Yapay Zeka",
      "selectionDate": "2024-12-20T00:00:00"
    },
    {
      "courseID": 3,
      "courseName": "Bilgisayar Mimarisi ve Organizasyonu",
      "selectionDate": "2024-12-20T00:00:00"
    },
    {
      "courseID": 1,
      "courseName": "Algoritma ve Programlama",
      "selectionDate": "2024-12-20T00:00:00"
    }
  ]
}
```

Advisors

GET	https://localhost:7262/api/advisors	Send
POST	https://localhost:7262/api/advisors	Send
PUT	https://localhost:7262/api/advisors	Send
DELETE	https://localhost:7262/api/advisors	Send

```
{
  "advisorId": 1,
  "fullName": "Prof. Dr. Hamdi Tolga Kahraman",
  "title": "Bölüm Başkanı",
  "department": "Yazılım Mühendisliği Bölümü",
  "email": "tolgakahraman@ktu.edu.tr",
  "students": [
    {
      "studentID": 10001,
      "firstName": "Ahmet",
      "lastName": "Yılmaz"
    },
    {
      "studentID": 10002,
      "firstName": "Mehmet",
      "lastName": "Kaya"
    },
    {
      "studentID": 10003,
      "firstName": "Ayşe",
      "lastName": "Demir"
    },
    {
      "studentID": 10004,
      "firstName": "Fatma",
      "lastName": "Celik"
    }
  ]
}
```

Courses

GET	https://localhost:7262/api/courses	Send
POST	https://localhost:7262/api/courses	Send
PUT	https://localhost:7262/api/courses	Send
DELETE	https://localhost:7262/api/courses	Send

```
{
  "courseId": 1,
  "courseCode": "YAZ101",
  "courseName": "Algoritma ve Programlama",
  "isMandatory": true,
  "credit": 4,
  "department": "Yazılım Mühendisliği",
  "studentCourseSelections": [
    {
      "studentID": 10002,
      "selectionDate": "2024-12-19T00:00:00"
    },
    {
      "studentID": 10001,
      "selectionDate": "2024-12-20T00:00:00"
    },
    {
      "studentID": 10001,
      "selectionDate": "2024-12-20T00:00:00"
    },
    {
      "studentID": 10001,
      "selectionDate": "2024-12-20T00:00:00"
    }
  ]
}
```

StudentCourseSelections

GET	https://localhost:7262/api/studentCourseSelections	Send
POST	https://localhost:7262/api/studentCourseSelections	Send
PUT	https://localhost:7262/api/studentCourseSelections	Send
DELETE	https://localhost:7262/api/studentCourseSelections	Send

```
{
  "studentID": 10001,
  "firstName": "Ahmet",
  "lastName": "Yılmaz",
  "courses": [
    {
      "courseID": 23,
      "selectionDate": "2024-12-20T00:00:00",
      "courseName": "Yapay Zeka"
    },
    {
      "courseID": 3,
      "selectionDate": "2024-12-20T00:00:00",
      "courseName": "Bilgisayar Mimarisi ve Organizasyonu"
    },
    {
      "courseID": 1,
      "selectionDate": "2024-12-20T00:00:00",
      "courseName": "Algoritma ve Programlama"
    },
    {
      "courseID": 2,
      "selectionDate": "2024-12-20T00:00:00",
      "courseName": "Veritabanı Yönetim Sistemleri"
    }
  ]
}
```

PendingSelections

GET	https://localhost:7262/api/pendingSelections	Send
POST	https://localhost:7262/api/studentCourseSelections	Send
PUT	https://localhost:7262/api/pendingSelections	Send
DELETE	https://localhost:7262/api/pendingSelections	Send

```
{
  "id": 92,
  "studentId": 10001,
  "studentFirstName": "Ahmet",
  "studentLastName": "Yılmaz",
  "courseName": "Bilgisayar Ağları",
  "courseId": 4
},
{
  "id": 93,
  "studentId": 10001,
  "studentFirstName": "Ahmet",
  "studentLastName": "Yılmaz",
  "courseName": "Nesneye Yönelik Programlama",
  "courseId": 5
},
{
  "id": 94,
  "studentId": 10001,
  "studentFirstName": "Ahmet",
  "studentLastName": "Yılmaz",
  "courseName": "Bilgisayar Ağları",
  "courseId": 4
},
```

User

GET	https://localhost:7262/api/users	Send
POST	https://localhost:7262/api/users	Send
PUT	https://localhost:7262/api/users	Send
DELETE	https://localhost:7262/api/users	Send

```
[
  {
    "userID": 11,
    "username": "tolgakahraman",
    "passwordHash": "3",
    "role": "Advisor",
    "relatedID": 1,
    "email": "tolgakahraman@ktu.edu.tr"
  },
  {
    "userID": 12,
    "username": "hozyurt",
    "passwordHash": "hashedpassword2",
    "role": "Advisor",
    "relatedID": 2,
    "email": "hozyurt@ktu.edu.tr"
  },
  {
    "userID": 13,
    "username": "geyup",
    "passwordHash": "hashedpassword3",
    "role": "Advisor",
    "relatedID": 3,
    "email": "geyup@ktu.edu.tr"
  },
  {
    "userID": 14,
```

FRONT END BÖLÜMÜ

User Login

Username

Password

Role

Student

Login

[Forgot Password?](#)

Login sayfası,kullanıcıların sisteme güvenli bir şekilde giriş yapmasını sağlar. Kullanıcı adı ve şifre bilgileri doğrulanarak ,doğru bilgileri giren kullanıcılar yetkilendirilmiş alanlara yönlendirilir. Giriş işlemi sırasında hatalı veya eksik bilgi girilmesi durumunda kullanıcılara geri bildirim sağlanır. Ayrıca oturum yönetimi gerçekleştirilerek kullanıcıların sisteme sürekli erişimi kontrol altında tutulur.

Öğrenci rolüne sahip kullanıcıların eriştiği bir ana kontrol paneli

Öğrenci Sayfası

ID: 10001

Ad Soyad: Ahmet Yılmaz

Ders Seçim

Çıkış Yap

Öğrencinin ID'si ve adı-soyadı gibi temel bilgileri, kullanıcının doğruluğu ve oturumla ilişkilendirilmesi için görüntülenir.

"Ders Seçim" butonu, öğrencilerin sistemde mevcut dersleri görüntüleyerek seçim yapmasını sağlar.

"Çıkış Yap" butonu, öğrencinin oturumunu sonlandırarak güvenli bir şekilde sistemden çıkış yapmasını sağlar.

Ders seçim sayfası

Ders Seçimi

Grup

- ☐ Algoritma ve Programlama
- ☐ Veritabanı Yönetim Sistemleri
- ☐ Bilgisayar Mimarisi ve Organizasyonu
- ☐ Bilgisayar Ağları
- ☐ Nesneye Yönelik Programlama
- ☐ Veri Yapıları ve Algoritmalar
- ☐ Matematiksel Modelleme ve Simülasyon
- ☐ İşletim Sistemleri
- ☐ Yazılım Mühendisliği Prensipleri
- ☐ Yazılım Test ve Kalite Güvencesi
- ☐ İnsan-Bilgisayar Etkileşimi
- ☐ Veri İletişimi ve Ağ Güvenliği
- ☐ Yazılım Projesi Yönetimi
- ☐ Bitirme Projesi I
- ☐ Bitirme Projesi II
- ☐ Mikroişlemciler ve Gömülü Sistemler
- ☐ Sayısal Mantık Tasarımı

Öğrenciler, kayıtlı oldukları program için uygun olan derslerin bir listesini görüntüler. Bu dersler, genellikle veri tabanından dinamik olarak getirilir.

Öğrenciler, almak istedikleri dersleri seçim kutularını işaretleyerek belirler. Her bir seçim, öğrencinin ilgili derse kaydolma isteğini ifade eder.

Öğrenci seçimleri, "Kaydet" butonu yardımıyla gönderilir. Bu işlem, öğrencinin seçtiği dersleri veri tabanına kaydeder ve ileride danışman veya yönetici tarafından onaylanması için işlenir.

İŞLEMİN BAŞARIYLA TAMAMLANMASI

Ders Seçiminiz Başarıyla Kaydedildi!

Seçimleriniz sistemimize başarıyla eklendi. Daha fazla işlem yapmak için menüyü kullanabilirsiniz.

[Ders Seçim Sayfasına Dön](#)

Öğrenci, ders seçimini tamamladıktan sonra, bu sayfa sayesinde seçimlerinin sisteme başarıyla kaydedildiğini öğrenir. Bu, kullanıcı deneyimi açısından önemlidir çünkü işlemin başarılı olup olmadığını kullanıcıya açıkça iletir.

Sayfada yer alan "Ders Seçim Sayfasına Dön" butonu, öğrencinin gerekirse tekrar ders seçim yapabilmesini sağlar. Böylece, kullanıcı yaptığı seçimleri yeniden kontrol edebilir veya düzenleyebilir.

Yönetici rolüne sahip kullanıcıların eriştiği bir ana kontrol paneli



Danışman girişi sonrası açılan ekran, danışmanın ID ve ad-soyad bilgilerinin görüntülenmesiyle giriş doğrulamasını sağlar. Ekranda bulunan "Ders Onay" ve "Öğrenciler" butonları, danışmana yetkili olduğu işlemleri hızlıca gerçekleştirme imkânı tanır. Ayrıca, "Çıkış Yap" butonu ile güvenli bir şekilde oturum kapatma seçeneği sunulur. Bu ekran, kullanıcı dostu bir arayüzle danışmanın akademik süreçleri yönetmesini kolaylaştırır.

Danışmana bağlı olan öğrencilerin listesi

Öğrencilerim

#	STUDENT ID	ÖĞRENCİ ADI	ÖĞRENCİ SOYADI
1	10013	Onur	Kurt
2	10014	Gizem	Aydın
3	10015	Furkan	Boz
4	10016	Ceren	Erdoğan
5	10031	Kağan	Aktaş

"Öğrenciler" butonuna basıldığında, danışmana bağlı olan öğrencilerin listesi görüntülenir. Bu sayfa, öğrencilerin kimlik bilgileri ve akademik durumları hakkında bilgi edinilmesini sağlar. Danışman, bu ekran aracılığıyla öğrencileriyle ilgili detaylı bilgiye ulaşabilir ve gerekirse akademik süreçleri yönetebilir. Bu özellik, danışmanın sorumluluğu altındaki öğrencilerle etkili iletişim kurmasına ve işlemlerini hızlı bir şekilde gerçekleştirmesine olanak tanır.

Ders Onaylama İşlemleri

Derslerim

Seç	Student ID	Student Name	Course ID	Course Name
<input type="checkbox"/>	10001	Ahmet Yılmaz	3	Bilgisayar Mimarisi ve Organizasyonu
<input type="checkbox"/>	10001	Ahmet Yılmaz	5	Nesneye Yönelik Programlama
<input type="checkbox"/>	10001	Ahmet Yılmaz	6	Veri Yapıları ve Algoritmalar
<input type="checkbox"/>	10001	Ahmet Yılmaz	8	İşletim Sistemleri
<input type="checkbox"/>	10001	Ahmet Yılmaz	28	Blockchain Teknolojileri

Seçilenleri Onayla

"Ders Onay" butonuna basıldığında, öğrenciler tarafından yapılan ders seçimlerinin listelendiği bir ekran açılır. Bu ekran, danışmanın öğrencilerin ders seçimlerini incelemesini ve onaylamasını sağlar. Danışman, her dersin öğrencinin akademik programına uygunluğunu kontrol ederek onaylama veya reddetme işlemini gerçekleştirir. Bu özellik, öğrencilerin doğru ders seçimleri yapmasını sağlarken, danışmanın akademik süreçlerdeki rolünü etkin bir şekilde yerine getirmesine olanak tanır.

ONAYLAMA İŞLEMİN BAŞARIYLA TAMAMLANMASI

Ders Onayınız Başarıyla Kaydedildi!

Seçimleriniz sistemimize başarıyla eklendi. Daha fazla işlem yapmak için menüyü kullanabilirsiniz.

[Ders Seçim Sayfasına Dön](#)

"Seçilenleri Onayla" butonuna basıldığında, danışman tarafından seçili olarak işaretlenen dersler sistemde onaylanır. Bu işlem, ilgili derslerin öğrencinin ders programına resmi olarak eklenmesini sağlar. Onaylanan dersler veritabanında "aktif" olarak güncellenir ve öğrenciler bu derslere katılmaya hak kazanır.

E-R DİYAGRAMI

Advisor ve Student:

Bir Advisor bir veya birden fazla Student ile ilişkilidir (birden çok öğrenci bir danışman olabilir).

Bir Student yalnızca bir Advisor ile ilişkilidir.

Course ve StudentCourseSelection:

Bir Course birçok StudentCourseSelection ile ilişkilidir (bir ders birçok öğrenci tarafından seçilebilir).

Bir StudentCourseSelection yalnızca bir Course ile ilişkilidir.

Student ve StudentCourseSelection:

Bir Student birçok StudentCourseSelection ile ilişkilidir (bir öğrenci birçok ders seçebilir).

Bir StudentCourseSelection yalnızca bir Student ile ilişkilidir.

Student ve Transcript:

Bir Student bir veya daha fazla Transcript ile ilişkilidir (bir öğrenci birçok dersten not alabilir).

Bir Transcript yalnızca bir Student ile ilişkilidir.

Course ve Transcript:

Bir Course bir veya daha fazla Transcript ile ilişkilidir (bir dersin birden fazla öğrencisi olabilir).

Bir Transcript yalnızca bir Course ile ilişkilidir.

PendingSelection:

PendingSelection hem Student hem de Course ile ilişkilidir. Bu, öğrenci ders seçimi yapmadığında bekleyen bir seçim durumudur.

User ve diğer varlıklar:

User varlığı, öğrenciler, öğretmenler veya diğer sistem kullanıcıları gibi farklı varlıklarla ilişkilendirilebilir. RelatedID bu ilişkiyi belirtir.



