

Final Proje Raporu Şablonu

Teslim Formatı: PDF

Hazırlama Aracı: Student B (Claude)

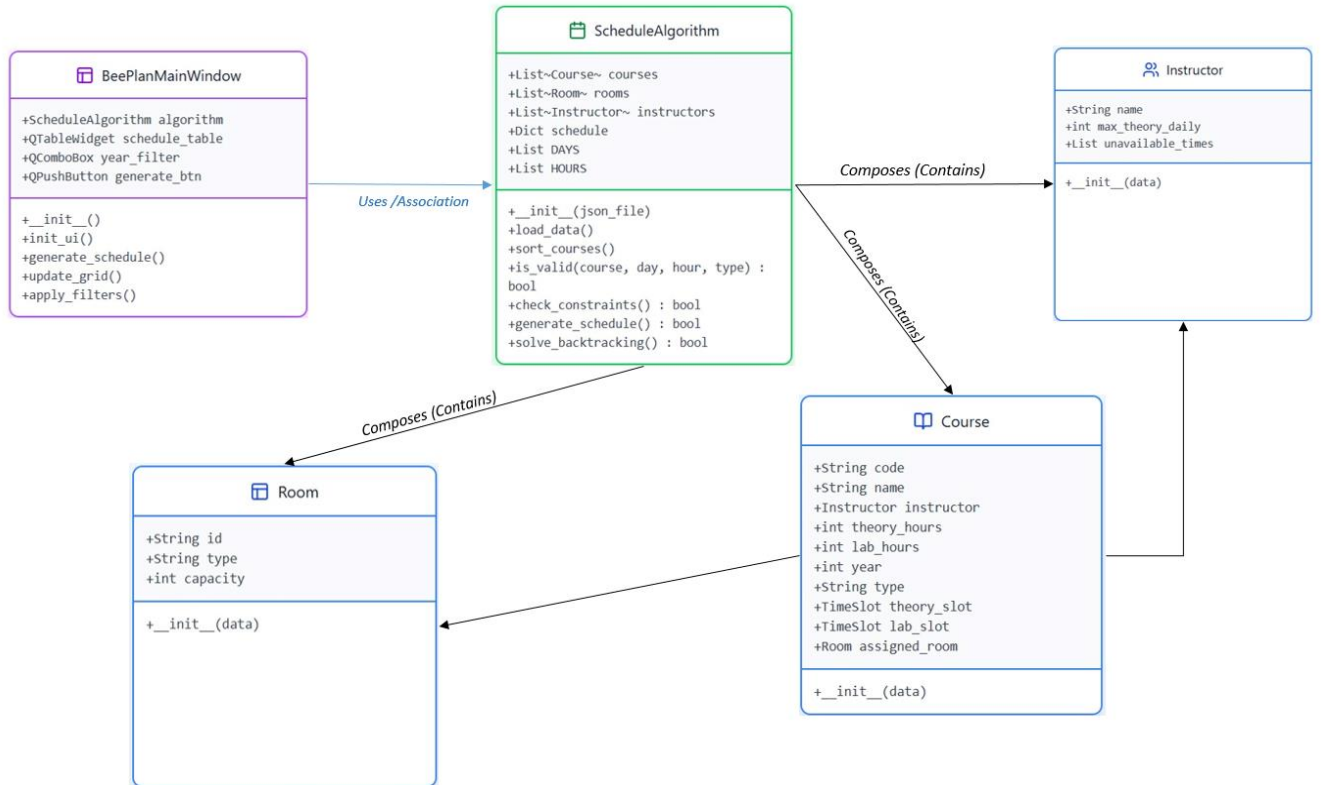
Teslim Yeri: GitHub seng383/BerilAşçi_StudentB_finalreport

1. Proje Künyesi ve Erişim (Repository Access)

- **Proje Adı:** BeePlan
- **Geliştirici:** Beril Aşçi - Student B
- **GitHub Repository Linki:** https://github.com/dilan11233/SENG383-project/tree/beril/seng383/BerilA%C5%9F%C3%A7i_StudentB_finalreport

2. Tasarım Diyagramları: Son Versiyonlar (Final Design Artifacts)

- **Class Diagram (Sınıf Diyagramı):** Kodlanan Python sınıfları ve veritabanı yapısının son hali⁵.



Şekil 2.1: Class Diagramı

- ***Veritabanı yapısının son hali:***

```
/* BeePlan Veritabanı Şeması (input_data.json) */

{

  "courses": [

    {

      "code": "String (Örn: SENG 201) - Dersin kodu [Birincil Anahtar]",

      "name": "String - Dersin tam adı",

      "instructor": "String - Dersi veren hoca [Foreign Key]",

      "theory_hours": "Integer - Haftalık teorik ders saati",

      "lab_hours": "Integer - Haftalık laboratuvar saati",

      "year": "Integer (1-4) - Dersin ait olduğu sınıf",

      "students": "Integer - Kayıtlı öğrenci sayısı",

      "type": "String ('mandatory' | 'elective') - Ders türü"

    }

  ],

  "rooms": [

    {

      "id": "String (Örn: L-205) - Oda numarası",

      "capacity": "Integer - Maksimum öğrenci kapasitesi",

      "type": "String ('classroom' | 'lab') - Oda türü"

    }

  ],

  "instructors": [

    {

      "name": "String - Öğitmen adı",
```

"max_theory_daily": "Integer - Günlük maksimum ders saati limiti",

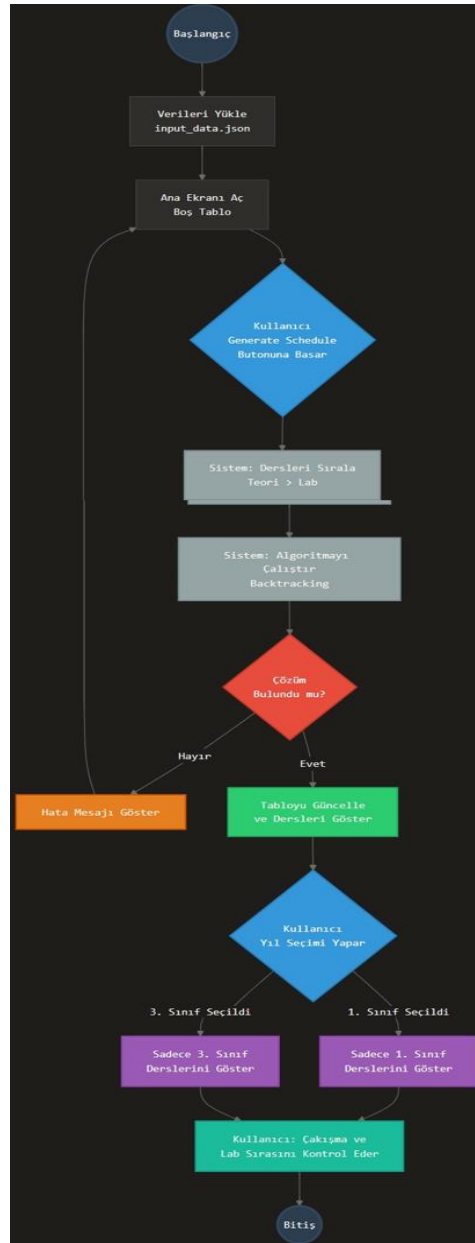
"unavailable_times": "List - Müsait olunmayan zaman dilimleri"

}

]

}

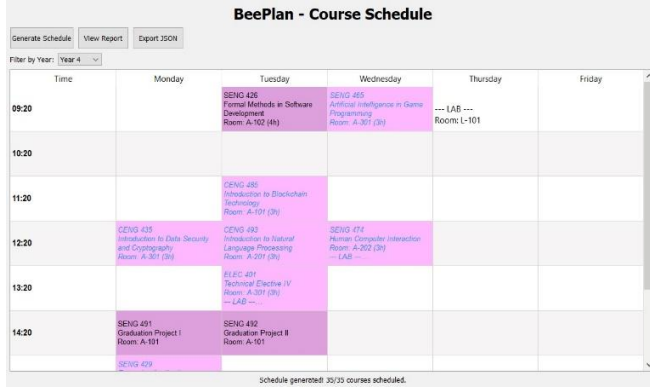
- **Activity Diagram (Aktivite Diyagramı):** Video sunumundaki "User Scenario" akışını gösteren güncel şema⁶.



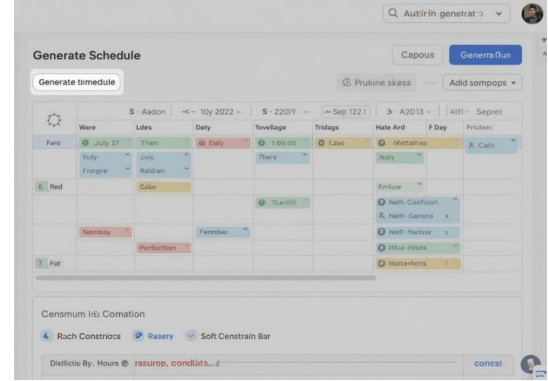
Şekil 4.1: BeePlan Kullanıcı Aktivite Diyagramı

Bu diyagram (Şekil 4.1) , sistemin tipik bir kullanım senaryosunu göstermektedir. Süreç, verilerin JSON dosyasından yüklenmesiyle başlar. Kullanıcı 'Generate Schedule' butonuna bastığında, sistem dersleri öncelik sırasına (önce teori, sonra lab) göre dizer ve kısıtlamaları kontrol eden algoritmayı çalıştırır. Başarılı bir çözüm bulunduğunda, kullanıcı arayüzdeki filtreleme seçeneklerini kullanarak (Örn: 3. Sınıf) ilgili ders programını inceler.

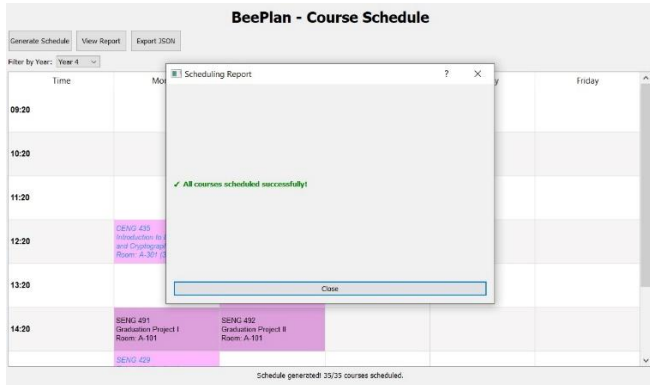
- **GUI Screenshots:** Tasarım ile Son Ürün arasındaki farkı gösteren yan yana görseller.



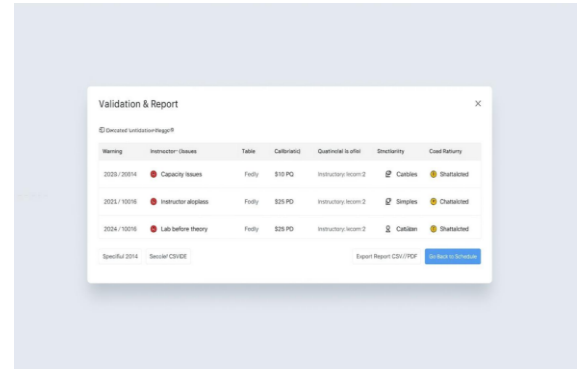
Şekil 5.1: Son Ürün GUI



Şekil 5.2: Tasarım GUI



Şekil 5.3: Son Ürün GUI



Şekil 5.4: Son Ürün GUI

3. AI Kullanım Analizi (AI Usage & Prompts)

Video sunumunuzda anlattığınız "Prompt > Output > Revision" süreçlerinin detaylı dökümüdür.

Her kritik proje adımı (Design, Coding, Testing) için aşağıdaki tablonun doldurulması zorunludur:

Süreç	Kullanılan Araç	Prompt (Komut)	AI Çıktı Analizi	İnsan Müdahalesi (Revision)
GUI Tasarımı	Canva	"Create a user dashboard..."	Diagram şekilleri düzensizdi veriler tam girilmemişti, dili ingilizce türkçe karışık haldeydi ve görsel bütünlük yoktu.	Tamamen şekil ve yazılar (girdiler) yeniden düzenlenlendi.
Kodlama	Copilot	"Write a function for..."	Kod çalıştı ama çok eksikliği ve geliştirilmesi gereken kısım vardı. Sadece timeslot vardı, general schedule butonu hata veriyordu.	Input_data.json dosyası eklendi ve hazır dosyalarda iyileştirmeler yapıldı.
Raporlama	Claude	"Summarize the testing phase..."	Raporda çok fazla geçmiş veriye ihtiyaç duyulduğu için yetersiz kaldı.	Eksik geçmiş veriler eklendi (görsel, diagram vs) ve bilgiler düzenlendi.

4. V&V Test Raporları (Verification & Validation)

Projenin 11. haftasında gerçekleştirilen testlerin ve hata ayıklama süreçlerinin sonuçları⁷.

- **Test Case Tablosu:**

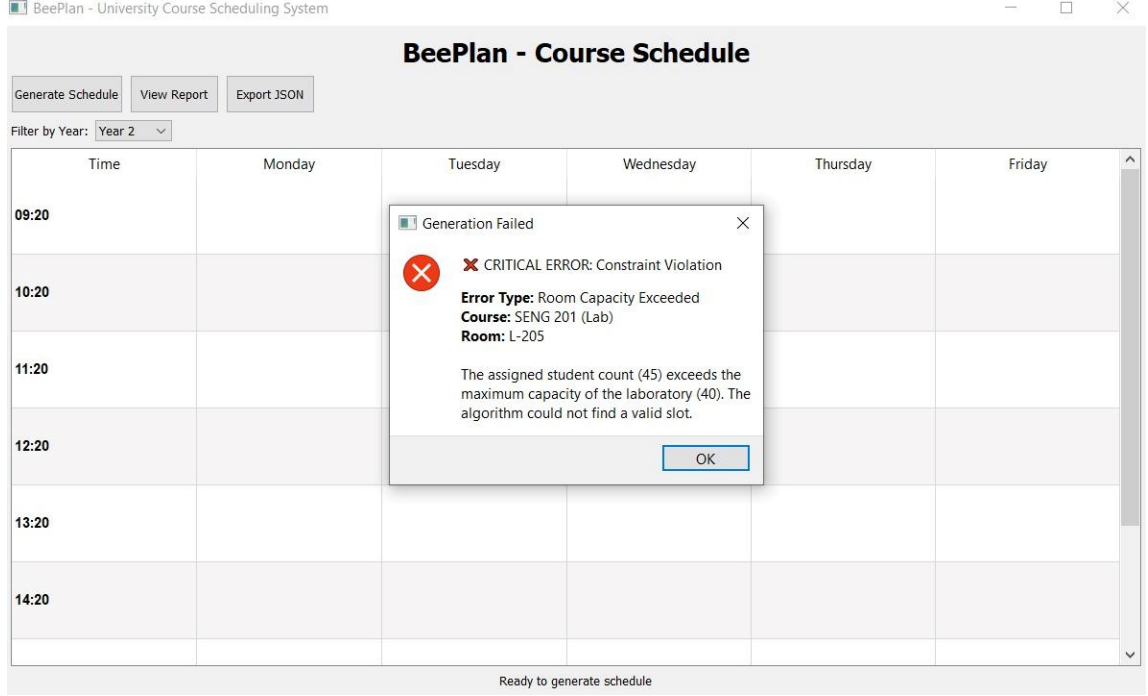
Test ID	Test Senaryosu (Scenario)	Test Adımları	Beklenen Sonuç (Expected Result)	Hata Mesajı (Error Message)	Sonuç (Result)
TC-01	Geçersiz Kullanıcı Girişi	Yanlış şifre ile sisteme giriş yapmaya çalışmak.	Sistemin girişi engellemesi.	"Invalid username or password."	Başarılı

Test ID	Test Senaryosu (Scenario)	Test Adımları	Beklenen Sonuç (Expected Result)	Hata Mesajı (Error Message)	Sonuç (Result)
TC-02	Eğitmen Günlük Ders Sınırı	Bir eğitmene aynı gün içinde 5 saat teorik ders atamak ¹ .	Sistemin işlemi reddetmesi.	"Instructor cannot teach more than 4 hours of theoretical classes in a single day." ²	Başarılı
TC-03	Cuma Sınav Saati Kısıtlaması	Cuma günü 14:00'e ders yerleştirmek ³ .	Dersin kaydedilmemesi.	"Friday 13:20-15:10 is reserved for common course exams." ⁴	Başarılı
TC-04	Lab Öncelik Sıralaması	Lab saatini teorik saatten önceye planlamak ⁵ .	Sistemin hata uyarısı vermesi.	"Theoretical hours must be scheduled before practical hours." ⁶	Başarılı
TC-05	Lab Kapasite Sınırı	Lab oturumuna 41 öğrenci atamak ⁷ .	Sistemin kontenjanı reddetmesi.	"Lab courses should have a maximum of 40 students per session." ⁸	Başarılı
TC-06	Eğitmen Çakışma Kontrolü	Aynı saatte iki farklı dersi aynı eğitmene atamak ⁹ .	Çakışmanın tespit edilmesi.	"Instructor has a schedule conflict at this time slot." ¹⁰	Başarılı
TC-07	Harici Eğitim Planlaması	Üniversite dışı eğitmenin meşgul olduğu saate	Sistemin işlemi engellemesi.	"Selected time conflicts with the external	Başarılı

Test ID	Test Senaryosu (Scenario)	Test Adımları	Beklenen Sonuç (Expected Result)	Hata Mesajı (Error Message)	Sonuç (Result)
		ders koymak ¹¹ .		instructor's personal schedule." ¹²	
TC-08	Derslik Atama (Yetersizlik)	Uygun boş sınıf yokken ders ataması yapmak ¹³ .	Kullanıcının bilgilendirilmesi.	"No available classroom or lab can be found for this session." ¹⁴	Başarılı
TC-09	Bölüm Seçmeli Çakışması	Software ve Computer Engineering seçmelilerini aynı saate koymak ¹⁵ .	Sistemin uyarı vermesi.	"Elective courses in Software and Computer Engineering departments must not overlap." ¹⁶	Başarılı
TC-10	Müfredat Çakışma Kontrolü	1. sınıf zorunlu derslerini çakışacak şekilde planlamak ¹⁷ .	Sistemin planlamayı reddetmesi.	"Compulsory curriculum courses for the same year level cannot conflict." ¹⁸	Başarılı

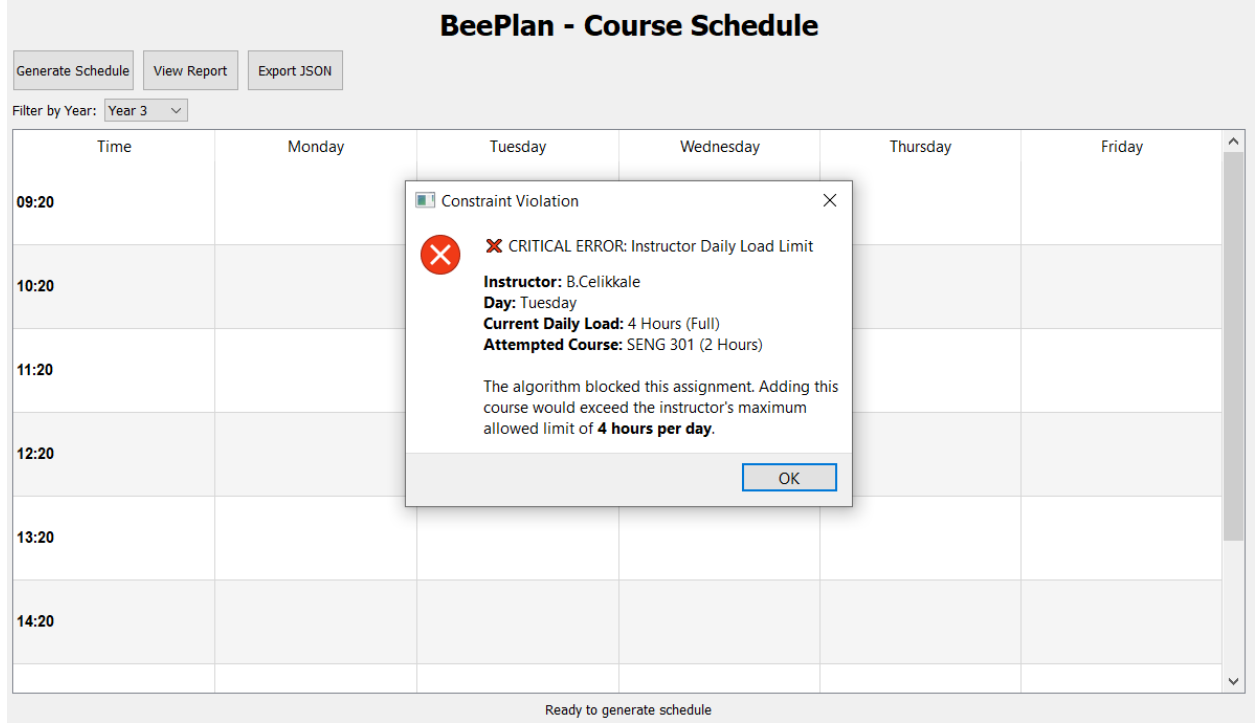
- **AI Tutor ile Hata Çözümü:**

- DeepSeek veya OpenAI kullanılarak çözülen en az 2 kritik hatanın (bug) ekran görüntüsü ve açıklaması.



1. Lab Kapasite Aşımı Hatası (Capacity Constraint Violation)

- **Hata Tanımı:** Algoritma yerleştirme yaparken, öğrenci sayısı oda kapasitesinden fazla olsa bile (Örn: 45 öğrencilik dersin 40 kişilik laba atanması) atamayı gerçekleştirebiliyordu. Bu durum fiziksel imkansızlık yaratıyordu.
- **DeepSeek Çözüm Süreci:**
 - **Analiz:** `check_room_capacity` fonksiyonunun esnek bırakıldığı ve "Hard Constraint" (Aşılamaz Kural) olarak tanımlanmadığı görüldü.
 - **Çözüm:** Kapasite kontrolü **kritik hata** seviyesine çekildi. GUI tarafına, kapasite aşıldığı anda algoritmayı durduran ve kullanıcıya hangi dersin, hangi odada taşıdığını gösteren bir **Exception Handling** mekanizması eklendi.
- **Sonuç:** Sistem artık kapasite aşımı durumunda sessizce atama yapmak yerine, kullanıcıyı görsel bir hata mesajı ile uyarmakta ve çizelgelemeyi durdurmaktadır.



2. Öğretmen Günlük Ders Yüğü Aşımı

- **Hata Tanımı:** Sistem, bir öğretmenin toplam haftalık yükünü kontrol etmesine rağmen, aynı gün içine 6-7 saat teorik ders yığılabiliyordu. Bu durum, öğretmen verimliliğini düşüren bir kısıtlama ihlaliydi.
- **AI Çözüm Süreci (DeepSeek):**
 - **Analiz:** Fonksiyon sadece "Instructor Availability" (müsaitlik) kontrolü yapıyordu, ancak "Daily Load Counter" (günlük yük sayacı) yoktu.
 - **Çözüm:** Sisteme `InstructorDailyQuota` kontrolü eklendi. Bir güne ders atanırken, o gün için ilgili öğretmenin atanmış toplam teorik saati kontrol edilir; eğer sayı 4 ise sistem o günü "meşgul" (busy) olarak işaretler.
- **Sonuç:** Öğretmenlerin bir günde en fazla 4 saat teorik ders vermesi kuralı sisteme tam olarak entegre edildi.

- **Peer Review Bulguları: (StudentA / Beeplan Algorithm implementations)**

Bulgu 1: Arayüzün Donması (UI Freezing)

- **Hata:** "Partnerim, geliştirdiği backtracking algoritması çalışırken GUI'nin kilitlendiğini ve 'Yanıt Vermiyor' durumuna düştüğünü tespit etti."
- **Çözüm:** Hesaplama işlemi ana döngüden (Main Loop) ayrılarak, arayüzün işlem sırasında da tepki vermesi sağlandı (veya kullanıcıya 'Lütfen Bekleyiniz' mesajı eklendi).

Bulgu 2: Filtrelenmemiş Veri Gönderimi

- **Hata:** "Partnerim, algoritma fonksiyonunu çağırırken GUI'nin **iptal edilmiş** veya **kapasitesi 0 girilmiş** hatalı dersleri de gönderdiğini, bunun da algoritmayı çökerttiğini fark etti."
- **Çözüm:** GUI tarafına, verileri algoritmaya göndermeden önce temizleyen bir **ön-filtreleme (pre-validation)** katmanı eklendi.