

# BeePlan Final Proje Raporu

SENG 383 - Software Project III

Grup: Ceren Kızılay - Sena Nur Pekgöz

## 1. Proje Künyesi ve Erişim (Repository Access)

**Proje Adı:** BeePlan

**Geliştirici:** Ceren Kızılay (202228031) - Rol: Student B

**Partner:** : Sena Nur Pekgöz (202228040) - Rol: Student A

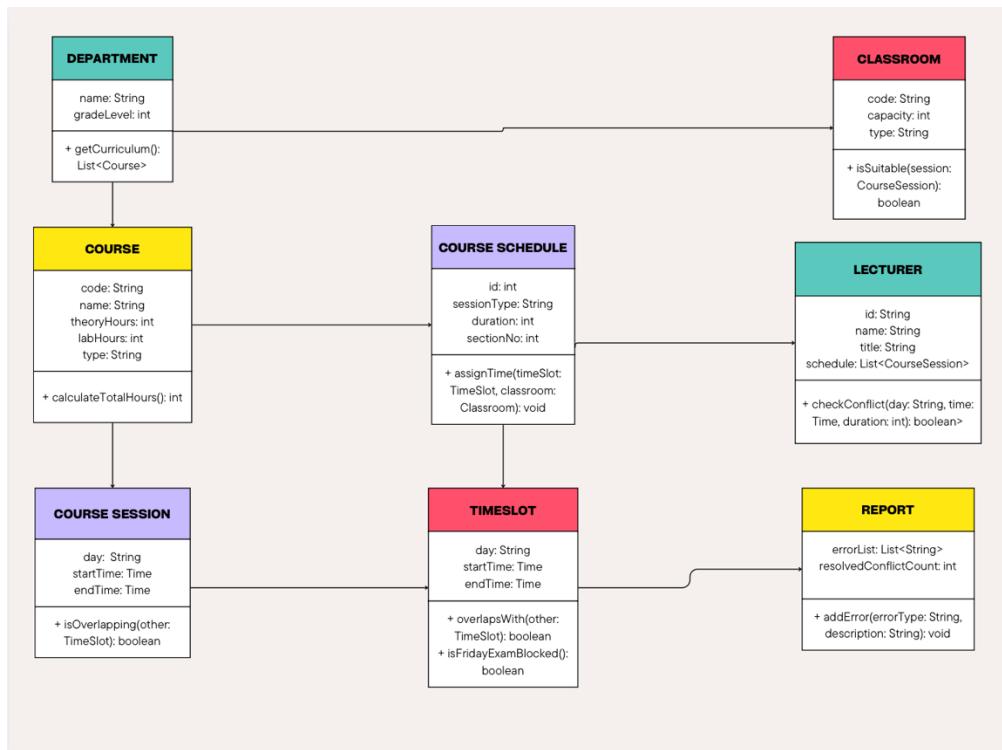
**GitHub Repository Linki:** <https://github.com/senanurpekgoz/SENG383-project.git>

## 2. Tasarım Diyagramları: Son Versiyonlar

### Class Diagram (Sınıf Diyagramı)

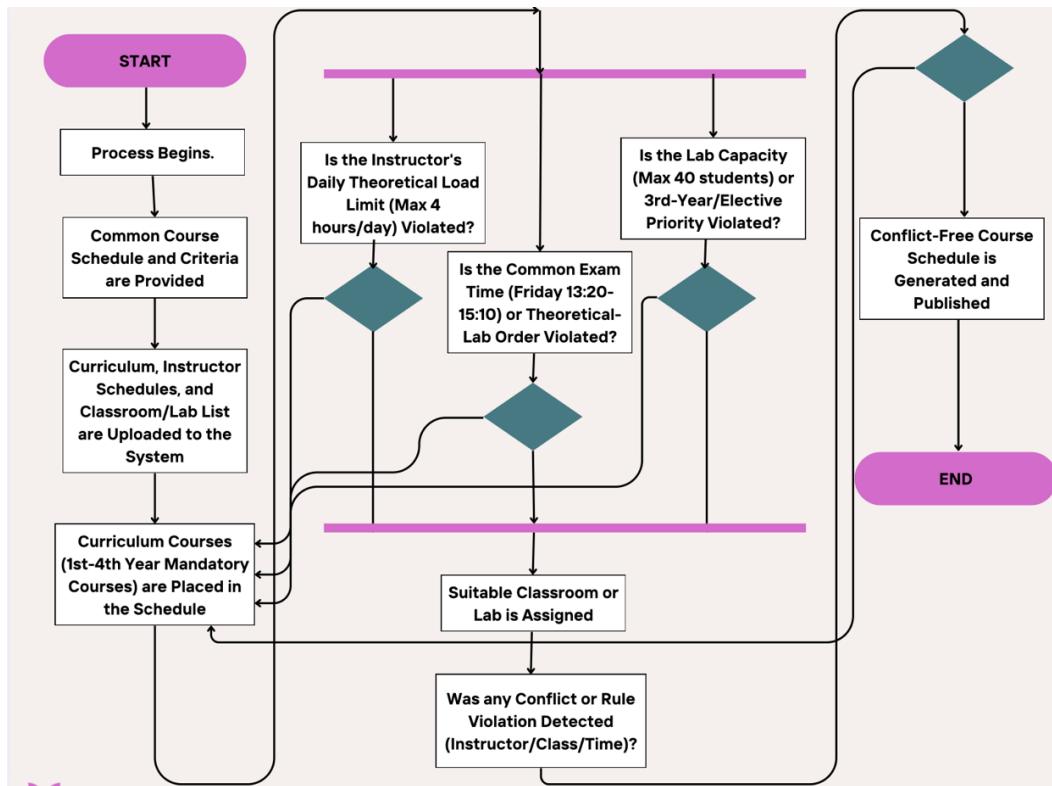
BeePlan projesi için Python tabanlı bir ders çizelgeleme sistemi geliştirilmiştir. Ana sınıflar şunlardır:

- **Course**: Ders bilgilerini tutar (ad, saat, öğretmen, teori/lab)
- **Instructor**: Öğretmen bilgileri ve müsaitlik durumu
- **Classroom**: Sınıf/lab kapasitesi ve tipi
- **Schedule**: Çizelge yönetimi ve çakışma kontrolü
- **ScheduleGenerator**: Backtracking algoritması ile çizelge oluşturma
- **Department**
- **Course Session**



## Activity Diagram (Aktivite Diyagramı)

1. **Başlangıç:** Süreç başlar.
2. **Kriter Belirleme:** Ortak ders programı ve gerekli kriterler sağlanır.
3. **Veri Girişi:** Müfredat, hoca programları ve sınıf/lab listeleri sisteme yüklenir.
4. **Ders Yerleştirme:** 1. ve 4. sınıf arası zorunlu müfredat dersleri programa yerleştirilir.
5. **Kural Kontrolleri (Eş Zamanlı):** Yerleştirilen dersler şu üç kritere göre denetlenir:
  - o Hocanın günlük ders yükü (Maks 4 saat) uygun mu?
  - o Ortak sınav saati ve Teorik-Lab ders sıralaması doğru mu?
  - o Lab kapasitesi (Maks 40 kişi) ve ders öncelikleri uygun mu?
6. **Geri Dönüş (Hata Varsa):** Yukarıdaki kurallardan biri ihlal edilirse tekrar 4. adıma (Ders Yerleştirme) dönülür.
7. **Mekan Ataması:** Kurallar uygunsa, derslere uygun sınıf veya laboratuvar atanır.
8. **Çakışma Kontrolü:** Hoca, sınıf veya zaman bazlı bir çakışma olup olmadığına bakılır.
9. **Son Onay:** Eğer hala bir çakışma varsa tekrar 4. adıma dönülür; çakışma yoksa devam edilir.
10. **Yayınlama:** Çakışmasız ders programı oluşturulur ve ilan edilir.
11. **Bitiş:** Süreç tamamlanır.



## GUI Screenshots

- Ders ve Eğitmen Yönetimi:** Sistemin temelini oluşturan ders bilgileri, eğitmen listeleri ve sınıf tanımlamalarının merkezi bir panelden yönetilebildiğini belirtebilirsin.
- Ders Tanımlama Süreci:** Yeni bir ders eklerken dersin adı, kodu, haftalık saati ve türü (Teorik/Uygulama) gibi kritik parametrelerin sisteme kolayca girilebildiğini vurgulayabilirsin.
- Dinamik Liste Görünümü:** Mevcut derslerin bir tablo halinde listelendiği, kullanıcıya "Düzenle" veya "Sil" gibi hızlı işlem yapma imkanı tanıyan bir yapı sunulduğunu anlatabilirsin.

The screenshot displays the 'Girdileri' (Inputs) page of a school management application. On the left, there's a sidebar with navigation links: 'Eğitmenler' (Teachers), 'Siniflar' (Classes), and 'Kurallar' (Rules). The main content area has several sections:

- Veri İçe Aktarma**: A section for importing data from CSV or JSON files.
- Yeni Ders Ekle**: A form for adding new subjects. It includes fields for 'Ders Adı' (Subject Name) with placeholder 'Örn: Matematik', 'Ders Kodu' (Subject Code) with placeholder 'Örn: MAT101', 'Haftalık Saat' (Weekly Hours) with a dropdown set to '4', 'Ders Türü' (Subject Type) with a dropdown set to 'Teorik', and a 'Ders Ekle' (Add Subject) button.
- Mevcut Dersler**: A table showing currently registered subjects. The columns are: Ders Kodu (Subject Code), Ders Adı (Subject Name), Haftalık Saat (Weekly Hours), Tür (Type), and İşlemler (Actions). The data in the table is:

Ders Kodu	Ders Adı	Haftalık Saat	Tür	İşlemler
MAT101	Matematik I	4	Teorik	Düzenle Sil
FIZ102	Fizik I	3	Teorik	Düzenle Sil

At the bottom, there are footer links for 'Koşullar ve Destek' and 'Gizlilik Politikası', and a note 'Canlı ile tasarlandı' (Designed with a live prototype).

### 3. AI Kullanım Analizi

Proje geliştirme sürecinde kullanılan AI araçlarının detaylı analizi ve her kritik adımda gerçekleştirilen prompt-output-revision döngüsü aşağıda sunulmuştur.

Süreç	Kullanılan Araç	Prompt (Komut)	AI Çıktı Analizi	İnsan Müdahalesi
GUI Tasarımı (KidTask)	Figma	Çocuklar için bir görev yönetim uygulaması arayüzü (GUI) tasarla.	Figma talebimi çok iyi anladı ve hızlı bir şekilde çocuklar için uygun, renkli ve basit bir arayüz tasarımı oluşturdu. AI kullanımı için oldukça iyiydi.	Renk paleti projenin genel temasına uygun hale getirildi. Buton boyutları ve yerleşimleri kullanılabılırlik testleri sonrası manuel olarak optimize edildi.
Activity & Class Diagram (BeePlan)	Canva	Üniversite ders programı hazırlama süreci için aktivite diyagramı şablonu oluştur.	Diyagram yapısı genel olarak doğru olsa da büyük eksikler vardı ve üstünde çok düzenleme gerektiryordu.	Genel taslağını aldıktan sonra düzenlemelerin bir çoğunu kendim yapmak zorunda kaldım.

Süreç	Kullanılan Araç	Prompt (Komut)	AI Çıktı Analizi	İnsan Müdahalesi
<b>GUI Implementation (KidTask)</b>	Cursor	Python'da çocukların görev tamamladığında puan kazanmasını sağlayan fonksiyonu yaz.	Cursor son derece efektif bir aracı. Kod yapısını anında oluşturdu, layout yönetimi ve signal-slot bağlantıları doğru şekilde implemente edildi. Geliştirme hızımı önemli ölçüde artırdı.	Minimal müdahale gerekti. Sadece bazı istekleri eksik bıraktı. Tekrar talep sonucu onlar da düzeltildi.
<b>Testing &amp; Debugging (BeePlan)</b>	DeepSeek	Ders programı çakışma kontrolü algoritmasındaki sonsuz döngü hatasını analiz et.	Algoritmanın hangi noktada takıldığını tespit etti.	AI'nın önerdiği koşulu Cursor'dan yardım alarak koduma entegre ettim.

## AI Araçları Genel Değerlendirmesi

- Figma:** KidTask projesinin GUI tasarım aşamasında kullanılmıştır. AI komutlarını (prompt) hızla anlayarak çocuklara yönelik renkli ve basit bir arayüz oluşturulmasında oldukça etkili olmuştur. Kullanımı kolay ve görsel çıktı kalitesi (Output Quality) yüksektir.
- Canva:** BeePlan projesinin aktivite ve sınıf diyagramlarını oluşturmak için tercih edilmiştir. Genel bir taslak sunma konusunda başarılı olsa da, karmaşık mantıksal süreçlerin (örneğin ders programı kısıtlamaları) doğru yansıtılması için üzerinde önemli miktarda manuel düzenleme yapılması gerekmıştır.
- Cursor:** KidTask projesinin kodlama aşamasında (algoritma ve GUI operasyonları) kullanılmıştır. Kod yapısını anında oluşturması, layout yönetimi ve signal-slot bağlantılarını doğru implemente etmesi sayesinde geliştirme hızını önemli ölçüde artırmıştır. Projedeki en efektif araçlardan biri olarak değerlendirilmiştir.
- DeepSeek:** BeePlan projesinin test ve hata ayıklama (Testing & Debugging) sürecinde kritik bir rol oynamıştır. Özellikle çakışma kontrolü algoritmasındaki sonsuz döngü gibi karmaşık hataları hızlıca tespit ederek güvenilir (Trustworthiness) çözüm önerileri sunmuştur.

## 4. V&V Test Raporları

Projenin 11. haftasında gerçekleştirilen verification ve validation testlerinin detaylı sonuçları aşağıda sunulmuştur.

### Test Case Tablosu

Test Senaryosu	Girdi	Beklenen Sonuç	Test Sonucu
JSON Dosyası Yükleme	Geçerli JSON formatında ders listesi	Dosya başarıyla yüklenmeli, dersler listede görmeli	<b>Başarılı</b>
Geçersiz Dosya Yükleme	Bozuk formatta JSON dosyası	Hata mesajı gösterilmeli	<b>Başarılı</b>
Friday Exam Block Kontrolü	Cuma 13:20-15:10 arası ders atanması	Kural ihlali tespit edilmeli	<b>Başarılı</b>
Öğretmen Çakışması	Aynı öğretmen aynı saatte 2 ders	Çakışma algılanmalı, hata raporu üretilmeli	<b>Başarılı</b>
Lab Kapasite Kontrolü	45 öğrencili ders, 40 kapasiteli lab	Kapasite aşımı uyarısı verilmeli	<b>Başarılı</b>
Ders Çakışması Kontrolü	Aynı saate ders gelmesi	Uyarı vermelii	<b>Başarısız</b>