

## Todo List Uygulaması Test Case Dokümanı

Bu test case de steplerin **sırasıyla** bitirilmesi beklenmektedir. Hepsinin bitirilmesine ihtiyaç **yoktur** ancak değerlendirme **hepsi** üzerinden yapılacaktır.

Projenin tüm kodlarını bir git reposuna yükleyerek <https://github.com/moviwo> kullanıcısına erişim izni vermeniz gerekmektedir.

### STEP 1

#### Frontend

Bu proje, kullanıcıların görevleri (tasks) oluşturmaya, her görev altında alt görevler (subtasks) oluşturmaya ve görevleri sürükleyip bırakmasına olanak tanıyan bir Todo List uygulamasıdır. Uygulama, yan yana olan iki kolondan oluşacaktır: birinci kolon "Backlog" ve ikinci kolon "In Progress" olacaktır. **Projede tailwind harici 3. parti bir teknoloji kullanılmaması gerekiyor. Tasarım konusunda tamamen özgür ilerleyebilirsiniz.**

#### Teknolojiler

- React/NextJs
- TypeScript
- CSS (veya bir UI kütüphanesi)

#### Kullanıcı Hikayeleri

1. **Görev Ekleme:** Kullanıcı yeni bir görev eklemelidir.
2. **Alt Görev Ekleme:** Kullanıcı bir görevin altında alt görevler ekleyebilmelidir.
3. **Görevleri Sürükleyip Bırakma:** Kullanıcı görevleri sürükleyip bırakma yeteneğine sahip olmalıdır.
4. **Görevleri Silme:** Kullanıcı mevcut görevleri ve alt görevleri silebilmelidir.

#### Test Case'ler

##### Test Case 1: Görev Ekleme

- **Açıklama:** Kullanıcı yeni bir görev eklemelidir.
- **Adımlar:**
  1. Uygulama açılır.
  2. "Backlog" kolonundaki görev ekleme alanına "Yeni Görev" yazılır.
  3. "Ekle" butonuna tıklanır.
- **Beklenen Sonuç:** "Yeni Görev" "Backlog" kolonunda görünmelidir.

#### Test Case 2: Alt Görev Ekleme

- **Açıklama:** Kullanıcı bir görevin altında alt görev ekleyebilmelidir.
- **Adımlar:**
  1. "Backlog" kolonundaki ana görev seçilir.
  2. Alt görev ekleme alanına "Yeni Alt Görev" yazılır.
  3. "Ekle" butonuna tıklanır.
- **Beklenen Sonuç:** "Yeni Alt Görev" seçilen ana görev altında görünmelidir.

#### Test Case 3: Görevleri Sürükleyip Bırakma

- **Açıklama:** Kullanıcı görevleri sürükleyip bırakabilmelidir.
- **Adımlar:**
  1. "Backlog" kolonundaki bir görev, fare ile seçilir.
  2. Görev sürüklenerek "In Progress" kolonuna bırakılır.
- **Beklenen Sonuç:** Görev "In Progress" kolonunda görünmelidir.

#### Test Case 4: Görev Silme

- **Açıklama:** Kullanıcı mevcut bir görevi silebilmelidir.
- **Adımlar:**
  1. "Backlog" kolonunda silinmek istenen görev bulunur.
  2. "Sil" butonuna tıklanır.
- **Beklenen Sonuç:** Seçilen görev "Backlog" kolonundan kaldırılmalıdır.

#### Test Case 5: Alt Görev Silme

- **Açıklama:** Kullanıcı mevcut bir alt görevi silebilmelidir.
- **Adımlar:**
  1. "Backlog" kolonundaki bir görev seçilir ve altında bulunan alt görevler görüntülenir.
  2. Silinmek istenen alt görev bulunur.
  3. "Sil" butonuna tıklanır.
- **Beklenen Sonuç:** Seçilen alt görev, ana görev altında görünmemelidir.

#### Tasarım ve Kullanıcı Arayüzü

- **Backlog Kolonu:** Ana görevler için.
- **In Progress Kolonu:** Programa alınmış işler için.
- Görevler ve alt görevler, sürüklenabilir olmalıdır.

#### Ek Notlar

- Kullanıcı deneyimini artırmak için uygun hata mesajları ve onay mesajları eklenmelidir.
- Uygulama, responsive olmalı ve farklı ekran boyutlarında iyi görünmelidir.

## STEP 2

### Backend

Frontend projesi için gerekli olan tüm endpointleri karşılayacak REST API projesi.

### Teknolojiler

- NodeJs
- SQLite

## STEP 3

### DevOps

Frontend, Backend ve DB projelerini dinamik, environment bazlı configurasyonlar olacak şekilde projeyi çalışır hale getirme. Burada beklentimiz **app.localhost** adresine girdiğimizde projenin doğru bir şekilde çalışması.

### Teknolojiler

- Docker
- İlave istenilen her teknoloji kullanılabilir.