

# Sınıf İçi Etkinlikler

Web Tabanlı Programlama Dersi | HTML • CSS • JavaScript

Bitlis Eren Üniversitesi

Mühendislik-Mimarlık Fakültesi | Bilgisayar Mühendisliği Bölümü

## 1-Ürün Paylaşma Sitesi(BEUShareBox)

**SINIF İÇİ UYGULAMA-1:** Mini Sosyal Medya Ürün Paylaşım Platformu

2025-2026 Bahar Dönemi

### PROJE TANIMI

**BEUShareBox**, kullanıcıların keşfettikleri veya sahip oldukları ürünleri toplulukla paylaşabilecekleri bir **mini sosyal medya platformudur**. Instagram'ın görsel paylaşım mantığını, bir ürün kataloğunun düzeni ile birleştiren bu uygulama; HTML, CSS ve JavaScript'in temel ve orta düzey becerilerini tek bir projede bir araya getirir.

Bu proje **sınıf içi uygulama** olarak tasarlanmıştır. Öğrenciler temel gereksinimleri karşılayarak **tam puan (10)** alabilir. İleri düzey özellikler ekleyen öğrencilere ise ders içinde **serbest (bonus) duyuru** yapılacaktır.

Ders: <b>Web Tabanlı Programlama</b>	Tür: <b>Bireysel Sınıf İçi Uygulama</b>
--------------------------------------	---

Teknoloji: <b>HTML5 + CSS3 + JavaScript</b>	Teslim: <b>Ders saati sonuna kadar (.zip dosyası)</b>
---	---

Tam Puan: <b>10 puan (temel gereksinimler)</b>	Bonus: <b>İleri düzey özellikler = Serbest duyuru</b>
--	---

### Senaryo

Üniversite kampüsünde öğrenciler sürekli birbirlerine ürün tavsiyesi yapıyor: "Şu kulaklığı aldım harika!", "Bu kitabı mutlaka oku!", "Yeni aldığım ayakkabıya bak!" BEUShareBox, bu organik paylaşım kültürünü dijital bir platforma taşıyor.

Siz bu platformun frontend geliştiricisisiniz. Kullanıcıların ürün paylaşabildiği, beğenebildiği, yorum yapabildiği ve kategorilere göre filtreleyebildiği bir tek sayfalık web uygulaması (SPA) geliştirmeniz bekleniyor.

### TEMEL GEREKSİNİMLER (10 Puan)

Aşağıdaki tüm gereksinimleri eksiksiz karşılayan her öğrenci **10 tam puan** alır. Eksik kalan her kriter puandan düşürülür.

#### 1. HTML Yapısı (Semantik & Erişilebilir)

✓	Özellik	Teknik İpucu
<input type="checkbox"/>	<b>Sayfa iskeleti</b> header, main, footer bölümleri	<code>&lt;header&gt;</code> , <code>&lt;main&gt;</code> , <code>&lt;footer&gt;</code>
<input type="checkbox"/>	<b>Ürün ekleme formu</b> Başlık, açıklama, fiyat, kategori alanları	<code>&lt;form&gt;</code> , <code>&lt;input&gt;</code> , <code>&lt;select&gt;</code> , <code>&lt;textarea&gt;</code>
<input type="checkbox"/>	<b>Ürün kartları bölgesi</b> Kartların listeleneceği container	<code>&lt;section&gt;</code> + <code>&lt;article&gt;</code> per card
<input type="checkbox"/>	<b>Kategori filtreleme alanı</b> Kategori butonları veya sekmeleri	<code>&lt;nav&gt;</code> veya <code>&lt;div role="tablist"&gt;</code>
<input type="checkbox"/>	<b>Arama çubuğu</b> Ürün adı/açıklamada arama	<code>&lt;input type="search"&gt;</code>

#### 2. CSS Tasarım (Responsive & Modern)

✓	Özellik	Teknik İpucu
<input type="checkbox"/>	<b>Kart bazlı grid layout</b> Ürünler kart görünümünde, grid veya flexbox ile	<code>display: grid;</code> <code>grid-template-columns: repeat(auto-fill, minmax(300px, 1fr))</code>
<input type="checkbox"/>	<b>Responsive tasarım</b> Mobil (1 sütun) ve masaüstü (2-3 sütun)	<code>@media (max-width: 768px)</code>

<input type="checkbox"/>	<b>Hover efektleri</b> Kartlarda shadow, scale veya renk geçişi	<i>transition + :hover pseudo-class</i>
<input type="checkbox"/>	<b>Tutarlı renk paleti</b> En az 1 ana renk + 1 vurgu rengi	<i>CSS Custom Properties (--primary, --accent)</i>
<input type="checkbox"/>	<b>Form tasarımı</b> Input'lar, butonlar stilize edilmiş	<i>:focus pseudo-class ile geri bildirim</i>

### 3. JavaScript İşlevsellik (Dinamik & İnteraktif)

✓	Özellik	Teknik İpucu
<input type="checkbox"/>	<b>Ürün ekleme</b> Form submit ile yeni ürün kartı oluşturma	<i>createElement + appendChild veya innerHTML</i>
<input type="checkbox"/>	<b>Beğeni sistemi</b> Her kartta beğeni butonu ve sayacı	<i>Event delegation + data attribute</i>
<input type="checkbox"/>	<b>Yorum sistemi</b> Her ürüne yorum ekleme ve görüntüleme	<i>Array.push + DOM güncelleme</i>
<input type="checkbox"/>	<b>Kategori filtreleme</b> Seçilen kategoriye göre kartları göster/gizle	<i>Array.filter + classList.toggle</i>
<input type="checkbox"/>	<b>Arama fonksiyonu</b> Ürün adı veya açıklamada anlık arama	<i>input event + String.includes()</i>
<input type="checkbox"/>	<b>localStorage ile kalıcılık</b> Sayfa yenilendiğinde veriler korunsun	<i>JSON.stringify / JSON.parse</i>
<input type="checkbox"/>	<b>Ürün silme</b> Her kartta silme butonu (onaylı)	<i>confirm() + splice/filter</i>

### 4. Kullanıcı Deneyimi (UX)

✓	Özellik	Teknik İpucu
<input type="checkbox"/>	<b>Boş durum mesajı</b> Hiç ürün yokken bilgilendirici mesaj	<i>"Henüz ürün yok, ilk paylaşımı sen yap!"</i>
<input type="checkbox"/>	<b>Form doğrulama</b> Boş alan kontrolü ve görsel uyarı	<i>required attribute + custom validation</i>
<input type="checkbox"/>	<b>Toplam istatistik</b> Toplam ürün, beğeni, yorum sayıları	<i>Array.reduce ile hesaplama</i>

### DEĞERLENDİRME TABLOSU

Her kriter aşağıdaki puan dağılımına göre değerlendirilecektir:

Kriter	Puan	Açıklama
<b>HTML Yapısı</b>	<b>2</b>	Semantik etiketler (header, main, section, article). Doğru iç içe yapı. Erişilebilirlik özellikleri (alt, aria-label).
<b>CSS Tasarım</b>	<b>2</b>	Flexbox/Grid kullanımı. Responsive tasarım (en az 2 breakpoint). Tutarlı renk paleti ve tipografi. Hover/focus durumları.
<b>JavaScript İşlevsellik</b>	<b>3</b>	Ürün ekleme formu çalışıyor. Beğeni/yorum sistemi aktif. Kategori filtreleme ve arama fonksiyonu. localStorage ile veri kalıcılığı.
<b>Kullanıcı Deneyimi (UX)</b>	<b>1.5</b>	Boş durum mesajları. Yükleme/hata durumları. Tutarlı ve sezgisel arayüz. Geri bildirim animasyonları.
<b>Kod Kalitesi</b>	<b>1.5</b>	Temiz ve okunabilir kod. Anlamlı değişken/fonksiyon isimleri. Yorum satırları. Modüler yapı (fonksiyon ayrımı).
<b>TOPLAM</b>	<b>10</b>	<b>Tam puan = Tüm temel gereksinimleri eksiksiz karşılamak</b>

## Önemli Not

Temel gereksinimlerin tamamını karşılayan her öğrenci 10 tam puan alır. Kısmi tamamlamalar orantılı puanlanır. Çalışmayan kod, ilgili kriterden 0 puan alır. Kopyala-yapıştır tespit edilen projeler değerlendirmeye alınmaz.

## İLERİ DÜZEY ÖZELLİKLER

### 🏆 Serbest Duyuru Ödülü

Temel gereksinimlerin üzerine aşağıdaki ileri düzey özelliklerden en az 3 tanesini başarıyla uygulayan öğrenciler, ders içinde serbest duyuru hakkı kazanır. Bu öğrencilerin çalışmaları sınıfa örnek olarak sunulur ve ders portalında vitrine çıkarılabilir.

1	<b>Karanlık / Açıklık Mod</b> ★★ Kullanıcı tercihini localStorage'da saklayan tema geçiş sistemi. CSS custom properties ile tüm renklerin dinamik değişimi.
2	<b>Drag &amp; Drop Sıralama</b> ★★★ Ürün kartlarını sürük-le-bırak ile yeniden sıralama. HTML5 Drag and Drop API kullanımı.
3	<b>Görsel Yükleme (Base64)</b> ★★★ FileReader API ile kullanıcının cihazından görsel seçip karta ekleme. Küçük görseller Base64 olarak localStorage'da saklanabilir.
4	<b>Animasyonlu Geçişler</b> ★★ Kart ekleme/silme sırasında CSS animasyonları. Filtreleme geçişlerinde fade/slide efektleri. @keyframes kullanımı.
5	<b>Sıralama Seçenekleri</b> ★★ Fiyata göre (artan/azalan), beğeni sayısına göre, tarihe göre sıralama. Array.sort() ile dinamik sıralama.
6	<b>Ürün Detay Modal</b> ★★★ Karta tıklayınca açılan detay penceresi. Yorumların tamamı, büyük görsel, paylaşım seçenekleri.
7	<b>Kullanıcı Profili</b> ★★★ Basit profil sistemi: isim, avatar seçimi, paylaştığım ürünlerim sayısı.
8	<b>Dışa/İçe Aktarma</b> ★★ Tüm verileri JSON olarak indirme ve yükleme. Blob API + download link oluşturma.
9	<b>PWA Desteği</b> ★★★★★ manifest.json + Service Worker ile uygulamayı ana ekrana ekleme. Temel offline destek.
10	<b>Bildirim Sistemi</b> ★★ Ürün eklendiğinde, beğenildiğinde toast notification gösterimi. setTimeout ile otomatik kapanma.

## ÇALIŞMA ADIMLARI

Projenizi aşağıdaki sırayla geliştirmeniz önerilir:

#	Adım	Süre	Detay
1	<b>Proje İskeleti</b>	15 dk	index.html, style.css, app.js dosyalarını oluşturun. HTML'de semantik yapıyı kurun (header, main, footer). CSS'de temel reset ve font ayarlarını yapın.
2	<b>CSS Grid Layout</b>	20 dk	Sayfa düzenini oluşturun: üstte navbar, ortada ürün grid'i, formun konumu. Responsive breakpoint'leri ayarlayın. Kart tasarımını tamamlayın.

3	Veri Modeli	10 dk	JavaScript'te ürün dizisini (array) oluşturun. Her ürün nesnesi: id, title, desc, price, category, likes, comments, createdAt. Örnek 2-3 ürünü hardcode olarak ekleyin.
4	Kartları Render Etme	20 dk	Veri dizisinden DOM'a kart oluşturan renderProducts() fonksiyonunu yazın. Template literal ile HTML string oluşturun veya createElement kullanın.
5	Form & Ürün Ekleme	20 dk	Form submit event'ini dinleyin. Input değerlerini alın, validation yapın, diziye ekleyin, kartları yeniden render edin.
6	Beğeni & Yorum	20 dk	Event delegation ile kart içi butonları yönetin. Beğeni toggle ve sayaç güncellemesini yapın. Yorum ekleme fonksiyonunu yazın.
7	Filtreleme & Arama	15 dk	Kategori butonlarına click event ekleyin. Array.filter ile filtrelenmiş listeyi render edin. Arama input'una keyup event ile anlık filtreleme ekleyin.
8	localStorage Entegrasyonu	10 dk	Her veri değişikliğinde localStorage'a kaydedin. Sayfa yüklendiğinde localStorage'dan verileri çekin.
9	Son Kontrol & Polish	20 dk	Tüm özellikleri test edin. Edge case'leri kontrol edin (boş form, uzun metin, çok fazla kart). Görsel son rötuşları yapın.

🕒 **Tahmini Toplam Süre:** ~150 dakika (2.5 saat) | İleri düzey özellikler için ek ~60 dakika

## TESLİM KOŞULLARI

- Proje dosyaları tek bir klasörde olmalı: index.html, style.css, app.js (ve varsa ek dosyalar).
- Klasör ismi: OgrenciNo\_BEUShareBox (Örnek: 2210101\_BEUShareBox)
- Klasörü .zip olarak sıkıştırıp ders portalına yükleyin.
- Teslim süresi: Ders saati bitimine kadar.
- Geç teslimler her saat başı -10 puan ceza ile değerlendirilir.
- Harici kütüphane/framework kullanımı YASAKTIR. Sadece vanilla HTML/CSS/JS.

### 💡 AI Araç Kullanımı Hakkında

Bu projede ChatGPT, Copilot, Claude vb. AI araçlarını kullanabilirsiniz. Ancak üretilen kodu anlamanız ve sınıf içinde açıklayabilmeniz gerekmektedir. Rastgele seçilen öğrencilerden kodlarını açıklamaları istenecektir. Açıklayamadığınız kod, değerlendirmede dikkate alınmaz.

### 🚀 Haydi Başlayalım!

Bu proje, gerçek dünyada kullanılan sosyal medya platformlarının temel yapı taşlarını öğrenmeniz için tasarlandı. Temel gereksinimlere odaklanarak başlayın, zamanınız kalırsa ileri düzey özelliklere geçin. Unutmayın: çalışan basit bir proje, yarım kalmış karmaşık bir projeden her zaman daha iyidir. Başarılar!

Dileyen Öğrenci aşağıdaki **Promptları** kullanabilir.

- 🎯 **Prompt 1** → **Minimum Requirements (10/10 almak için)**
- 🚀 **Prompt 2** → **Advanced / Full Version (Showcase seviyesi)**

Bunlar öğrencinin ChatGPT / Claude / Copilot'a direkt verebileceği şekilde optimize edildi.

# PROMPT 1 — MINIMUM REQUIREMENTS (Full 10 Points)

You are a senior frontend developer.

Build a Single Page Application (SPA) called "BEUShareBox".

This is a classroom project. The goal is to implement ONLY the core required features using:

- HTML5 (semantic structure)
- CSS3 (responsive modern design)
- Vanilla JavaScript (no frameworks, no libraries)

⚠ Do NOT use React, Vue, Bootstrap, Tailwind, or any external library.

Only pure HTML, CSS, and JavaScript.

=====

## PROJECT DESCRIPTION

=====

BEUShareBox is a mini social media product sharing platform.

Users can:

- Add products
- Like products
- Comment on products
- Filter by category
- Search products
- Delete products
- Persist data using localStorage

=====

## REQUIRED FEATURES

=====

### 1) HTML Structure (Semantic & Accessible)

- Use <header>, <main>, <footer>
- Product form with:
  - Title
  - Description
  - Price
  - Category (select dropdown)
- Product list container using <section>
- Each product card must be inside <article>
- Category filter area
- Search input (type="search")

### 2) CSS Requirements

- Responsive grid layout using CSS Grid or Flexbox
- Mobile: 1 column
- Desktop: 2-3 columns
- Hover effects on product cards
- Clean modern color palette
- Styled form inputs with focus effects
- Use CSS variables for primary and accent colors

### 3) JavaScript Features

- Add product dynamically (form submit)
- Like button with counter
- Comment system per product
- Category filter using Array.filter()
- Search functionality using input event
- Delete product with confirmation()

- Persist products using localStorage
- Render products dynamically using DOM manipulation

#### 4) UX Features

- Show empty state message if no products
- Validate form inputs
- Display total product count
- Display total likes count

=====

#### TECHNICAL RULES

=====

- Use modular JavaScript (functions separated clearly)
- Keep code clean and readable
- Use meaningful variable names
- Add comments explaining logic
- All data stored as array of objects:

```
{
  id,
  title,
  description,
  price,
  category,
  likes,
  comments: [],
  createdAt
}
```

=====

#### OUTPUT FORMAT

=====

Provide:

- 1) index.html
- 2) style.css
- 3) app.js

Code must be complete and working.



## PROMPT 2 — ADVANCED VERSION (Showcase Level)

Bu prompt full feature isteyen öğrencilere.

You are a senior frontend engineer building a professional-grade SPA.

Upgrade the BEUShareBox application into a feature-rich, production-style frontend app.

Use only:

- HTML5
- CSS3
- Vanilla JavaScript

No frameworks or libraries allowed.

=====

#### CORE FEATURES

=====

Implement all minimum required features:

- Product add
- Like system
- Comment system
- Category filter
- Search
- Delete
- localStorage persistence
- Responsive layout

=====

## ADVANCED FEATURES

=====

Add at least 5 of the following advanced features:

### 1) Dark / Light Mode

- Toggle theme button
- Store user preference in localStorage
- Use CSS custom properties

### 2) Drag & Drop Reordering

- Use HTML5 Drag & Drop API
- Update product order dynamically
- Save order in localStorage

### 3) Image Upload

- Use FileReader API
- Convert image to Base64
- Store small images in localStorage
- Display preview on product card

### 4) Animated Transitions

- Smooth card add/remove animations
- Fade/slide filter transitions
- Use CSS @keyframes

### 5) Sorting Options

- Sort by price
- Sort by likes
- Sort by newest
- Use Array.sort()

### 6) Product Detail Modal

- Click card to open modal
- Show full description
- Show all comments
- Close with ESC key

### 7) User Profile System (Basic)

- Set username
- Upload avatar

- Show “My Products” filter

#### 8) Import / Export Data

- Export JSON file (Blob API)
- Import JSON file
- Merge safely with existing data

#### 9) Toast Notification System

- Show animated toast when:
  - Product added
  - Product liked
  - Product deleted
- Auto close after 3 seconds

#### 10) Statistics Dashboard

- Total products
- Most liked product
- Category distribution
- Use Array.reduce()

### =====

### ARCHITECTURE REQUIREMENTS

- Use modular architecture
- Separate rendering logic
- Separate state management
- Avoid inline event handlers
- Use event delegation
- Keep code scalable

### =====

### GOAL

=====

Make it look like a modern SaaS product.

Focus on:

- Clean UI
- Smooth interactions
- Professional code structure
- Performance optimization

Provide full project code:

- index.html
- style.css
- app.js

The result should feel like a mini real-world social product platform.

**Alternatif Olarak Aşağıdaki Uygulamayı da yapabilirsiniz?**

**2-Loop Studio**



demo url: <https://deep-state-00019630.figma.site/>

Sitedeki temel muzik yeteneklerini yapması gerekir. Arayüz ve stiller serbest aynı olmak zorunda değil.