Öncekile sözlük yazılımı gelişmiş ilişkisel veritabanı yapısına sahip olacak. İki farklı arayüzü olacak. Bir admin girişi olacak. Bu kişi aynı bütün yetkilere sahip olacak. Bu kişi verişi girişleri için şifreyle giriş yapabilen yeni kullanıcılar tanımlayabilecek. Bir de kullanıcı arayüzü olacak. Buraya da girişler kayıt yapılarak girilebilecek. Ad, soyad, şifre ve eposta bilgisiyle kayıt olan kullanıcılar sözlüğün kullanıcı arayüzünü görebilecek. Arayüzler hem kullanışlı hem de estetik olarak cazibeli olacak.

Sözlük yazılımı modüller şeklinde olacak. Yani aşağıda vereceğim parametreler birer modül olacak. Bir kere sürükleyip bıraktığım ve dizilimini yaptığım sözlük veri tabanı sonraki madde girişleri için de belirlenmiş olacak. Yani istediğim modülleri sürükleyip bırakacağım ve sözlükte hangi parametrelerin olduğuna ilk maddebaşı girilmeden önce karar vereceğim. Sonrakilerinde aynen devam edecek. Kullanıcı arayüzünde ve admin yani veri girişi arayüzünde istediğim gibi bu modülleri sıralayabileceğim. Daha önceden tanımlanmış bir modül yoksa istek hâlinde yeni bir modül de sonradan ekleyebileceğim. Adını ve girdisini, sırasını diğer parametreler gibi belirleyebileceğim.

**Sözlük programı parametreleri.**

**Maddebaşı**:

Maddebaşı için 3 şekilde veri girişi olabilecek. 1. Yöntemde belirli bir derleme bağlanarak otomatik bir şekilde maddebaşları oluşturulacak. 2. Yöntemde sistem yapay zeka destekli bir algoritma kullanarak web ortamından seçilen dille ilgili sözcükleri derleyecek ve maddebaşı oluşturacak. 3. Yöntemde ise klasik bir şekilde admin olan kişi veya kişiler elle maddebaşı girebilecek. Bütün bu seçenekler modüller sürükle bırak yöntemi ile hazırlandıktan sonra veri girişi sayfasında butonlarla seçime sunulacak.

Maddebaşı kutucuğu anahtar olacak. Aynı sözcük tekrar girilemeyecek. Girildiğinde sistem uyarı verecek. Aynı sözcük bu uyarıya rağmen tekrar girilirse sistem önceki girilen Romen rakamı ile 1’den başlatılarak numaralandırma ne kadar tekrar ederse o kadar devam edecek. Maddebaşıyla ilişkili sözcükler kavram alanı, kavram haritası için işaretlenecek. Daha sonra bununla ilgili visualthesarus.com sitesindeki yapıya benzer bir gösterimle kullanıcıya sunulacak. Aynı zamanda Madde başı sözcük hem yapay zeka ile hem de elle köklerine ve eklerine ayrılacak. Bu bilgi de başka bir alanda kaydedilecek. Sonra istendiğinde kavram visualthesarus.com sitesinde olduğu gibi üzeri tıklanan sözcükle ilgili çeşitli bilgiler sağ tarafta gösterilecek. Burada gösterilecek bilgilerden biri de ek-kök bilgisi. Burada gösterilecek bilgileri de manuel olarak belirlenecek. Maddebaşı kullanıcı arayüzünde koyu yazılacak. Metin küçük büyük karakterde nasıl girildiyse o şekilde kullanıcı arayüzünde görülecek.

Maddebaşı sözcük başka bir sözcüğe gönderme yapacaksa (cross referense) unikod simgelerinden birisi sonradan tercih edilebilecek. Şayet gönderme yapıldıysa ve o sözcük maddebaşı olarak girilmediyse henüz sistem farklı bir renkle uyaracak. Ne zaman ki gönderme yapılan sözcük maddebaşı olarak girildi o zaman uyarı sonlanacak.

Maddebaşı modülü diğer bütün modüllerin yani sözlük bileşenlerinin bağlı olduğu yapı. Sonraki girilecek bütün bilgiler hangi maddebaşı için girildiyse onunla eşleştirilecek.

**Anlamlar** (sense):

Bir maddebaşına birden fazla tanım yapılabilecek. Maddebaşının 1 tane anlamı varsa anlam numaralandırılmayacak. Şayet birden fazla anlam varsa 1., 2., 3., …. Diye sınırsız sayıda devam edebilecek. Numaralandırılmış her anlamın önünde varsa sözcük türü bilgisi, kullanım alanı gibi bilgiler girilebilecek. Bu bilgiler girilirse farklı font ve karakter büyüklüğünde olacak. Anlamların arkasından örnek cümleler girilecek. Örnek cümlelerin arkasından şayet varsa maddebaşıyla ilgili deyimler ve atasözleri iç maddebaşı olarak girilecek ve anlamı verilecek. Birden fazla ise 1., 2, 3., … şeklinde sıralanacak. İç maddebaşları küçük harflerle, koyu bir şekilde dizilecek. İç maddebaşlarına örnek cümleler varsa hemen ardından onlar girilecek. Bir sözlük maddesinin en altında maddebaşı olan sözcükle ilgili sözcükler koyu bir şekilde sadece dizilecek. Paragrafın başında da ok işareti simgesi olacak. Hem maddebaşının anlamları için hem de iç maddebaşlarının anlamları için bir veri girişi olduktan sonra yanında bir seçenekle ikinci bir anlam veya ikinci bir iç maddebaşı girilmesi imkanı verilecek. Bu bir artı işareti olabilir.

Girilen maddebaşları hem sözdizimsel, hem morfolojik çalışmalar için raporlanabilecek, üzerinde araştırma yapılabilecek veriler sunabilecek. Mesela bir başka arayüzde aratılan sözcüklerin sözdizimsel kullanımları sıralanabilecek. Sayısı kullanıcıya sorularak.

Diğer sözlük modüllerini burada girmiyorum. Bu modüller yani sözlük bileşenleri manuel olarak girilsin. Bu modüller şayet kendi içinde otomatik önceden yüklenmiş seçenekleri sunmak gerektiği durumlarda combobox gibi yine admin tarafından seçenekleri girilebilecek bir yapıda tasarlansın.

Sonuç olarak istediğim kadar modülü, istediğim gibi tasarlayabileceğim bir serbestlik sağlayan web tabanlı profesyonel sözlük hazırlama programı yazılımı kodlanmanı istiyorum.

**🧠 1. GENEL MİMARİ PLANI**

**✅ Web tabanlı olacak**

* **Frontend:** React + TailwindCSS (modüler arayüz + sürükle-bırak desteği)
* **Backend:** Flask (Python) + REST API
* **Veritabanı:** SQLite (başlangıçta), PostgreSQL’e geçilebilir

**✅ İki Arayüz:**

* **Admin Paneli:**
  + Admin girişli
  + Kullanıcı ekleme, modül tanımlama
  + Sürükle-bırak ile modül sıralama
  + Modül oluşturma / silme / yeniden sıralama
  + Derlemeye bağlama, AI destekli madde üretimi
* **Kullanıcı Arayüzü:**
  + Kayıt ve giriş sistemi
  + Modüllerin belirlenmiş sırayla görüntülenmesi
  + Arama, görsel-morfolojik görüntüleme, kök-ek, örnek cümle, kavram haritası gösterimi

**🔄 2. MODÜLER YAPI**

**✨ Modüller sürükle-bırak arayüzle tanımlanır:**

Örnek modül listesi (her biri tabloya karşılık gelir):

* Maddebaşı
* Anlam
* Sözcük türü
* Kullanım alanı
* Örnek cümle
* İç maddebaşı
* Atasözleri / deyimler
* Kök / ek bilgisi
* Cross-reference
* Kavram haritası işaretleyicisi
* Görsel / ses dosyaları

**🔧 Admin, modül ekleyebilir:**

* Yeni modül adı
* Veri tipi: Metin / Sayı / Dosya / Çoklu seçim / Tarih vs.
* ComboBox içeriği (isteğe bağlı)
* Giriş kontrolü (örneğin: maddebaşı benzersiz olacak mı?)

**🧩 3. MADDEBAŞI YÖNETİMİ**

**Maddebaşı 3 yöntemle oluşturulabilecek:**

1. **Derlemeye bağlı otomatik**
2. **Yapay zeka destekli web kazıması**
3. **Manuel giriş**

Bu seçenekler butonlar olarak görünür olacak. Admin hangi yöntemi seçecekse modülü o şekilde kullanacak.

**Maddebaşı özel kuralları:**

* Anahtar olacak (benzersizlik)
* Tekrar girilirse uyarı → numaralandırma otomatik yapılacak (I, II, III…)
* Cross-reference uyarı mekanizması (varsa ama henüz madde değilse)
* Kavram etiketleri işaretlenebilir (haritalama için)
* Kök / ek bilgisi elle ve AI ile yapılabilir

**📘 4. ANLAM VE ALT ÖZELLİKLER**

* Bir maddebaşına sınırsız anlam girilebilir
* 1’den fazla anlam varsa 1., 2., … diye numaralandırılır
* Anlamla birlikte: *Tür*, *alan*, *örnek cümle* eklenebilir
* Her anlamın sonunda iç maddebaşları (alt sözcük) ve onların anlamları girilebilir
* İç maddebaşları da örnek cümle alabilir
* Her anlam veya iç madde için “+” işaretiyle yenisi eklenebilir

**📊 5. VERİ ANALİZİ VE ARAMA**

* Maddebaşına göre tüm anlamlar, örnek cümleler listelenebilir
* Morfolojik analizler (ek-kök)
* Kavram haritaları (visualthesaurus benzeri dinamik grafikler)
* Arama: Tam metin, etiket, tür, alan, kök vs. ile filtrelenebilir

**🧱 6. VERİTABANI YAPISI (Özet)**

Tabloları sadece özetliyorum, detaylı modelleri istersen birlikte yazabiliriz.

| **Tablo** | **Amaç** |
| --- | --- |
| users | Admin ve kullanıcı hesapları |
| modules | Oluşturulan her sözlük bileşeni |
| module\_order | Modüllerin sürükle-bırak sırası |
| entries | Maddebaşları |
| senses | Anlamlar (maddebaşına bağlı) |
| examples | Örnek cümleler |
| subentries | İç maddebaşları |
| derivations | Kök-ek bilgisi |
| media\_files | Görsel / ses dosyaları |
| relations | Kavram bağlantıları ve cross-ref ilişkileri |

### ✅ 1. Flask Projesi Kurulumu (Backend temel yapı)

### ✅ 2. Veritabanı Yapısının SQLAlchemy ile Kodlanması

### ⏭️ 3. Yönetici (Admin) Paneli Girişi ve Yetkilendirme

### ⏭️ 4. Dinamik Modül Tanımı ve Sürükle-Bırak Modül Sıralama

### ⏭️ 5. Maddebaşı Girişi (3 Yöntem: Elle, Derlem, AI)

### ⏭️ 6. Anlam, Örnek Cümle, İç Maddebaşı Ekleme Arayüzü

### ⏭️ 7. Kavram Haritası ve Ek-Kök Bilgisi Alanları

### ⏭️ 8. Kullanıcı Girişi ve Kullanıcı Arayüzü

### ⏭️ 9. Görsel-Ses Ekleri, Arama ve Filtreleme

### ⏭️ 10. Verilerin Raporlanması (Sözdizimsel, Morfolojik)

sozluk\_backend/

│

├── app/

│ ├── \_\_init\_\_.py # Flask app yaratma, db init vs.

│ ├── models.py # Veritabanı modelleri

│ ├── routes/

│ │ ├── \_\_init\_\_.py

│ │ ├── auth.py # Kullanıcı yönetimi (login/register)

│ │ ├── modules.py # Modül yönetimi API'leri

│ │ ├── entries.py # Maddebaşı API'leri

│ ├── extensions.py # db, migrate, ma gibi uzantılar

│

├── migrations/ # Alembic migration dosyaları

├── config.py # Konfigürasyonlar

├── manage.py # CLI komutları (db migrate, run vs.)

├── requirements.txt

└── run.py # Uygulamayı başlatmak için