

E-Grading Mapleme Süreç Dokümantasyonu

1. Giriş

Bu doküman, **E-Grading Eğitim Formlarının** ön yüzden (frontend) doldurulmasının ardından, arka planda (backend) **TMS 1.0 veritabanı tablolarına mapleme sürecini** kapsamaktadır. Süreç; JSON formatındaki form verilerinin **EGRADING_FORM_DATA** tablosuna kaydedilmesi, instructor tarafından **Complete** butonuna basılmasıyla devreye giren servis üzerinden tabloların beslenmesi ve transaction mantığıyla güvence altına alınmasını içerir.

2. Süreç Akışı

- Form Girişi** → Instructor veya trainee, e-grading formunu doldurur.
 - Geçici Kayıt** → Veriler JSON formatında **EGRADING_FORM_DATA.JsonData** kolonuna yazılır. Burada her kayıt **TrainingId + TraineeId + InstructorId + FormId** dört anahtar üzerinden tutulur.
 - Complete Butonu** → Instructor, Complete butonuna bastığında **Training Completion Service** devreye girer ve tüm ilgili trainee/form kayıtları işlenir.
 - Deserialize** → JSON camelCase formatında parse edilir. Veri tipleri (string/int) kontrol edilir; tip uyumsuzluğunda hata oluşur.
 - Transaction** → Tüm insertler transaction içinde yapılır. Hata alınırsa rollback edilir.
 - Finalizasyon** → Başarılı işlem sonunda **TRAINING.StatusId = 2 (Complete)** güncellenir, **EGRADING_COMPLETED_TRAINING** tablosuna kayıt düşülür.
-

3. JSON Veri Yapısı ve Kurallar

Form verileri JSON formatında aşağıdaki şekilde iletilir:

```
{  
  "lessonCode": "OPC - Lesson 3",  
  "lessonId": 157,  
}
```

```
"trainingId": 280555,
"traineeId": 1,
"traineeFullName": "HÜS*****",
"traineeCrewId": "ERDSR",
"formId": 17,
"EgradingFormParts": {
  "OverallResult": {...},
  "LessonSubjects": [...],
  "Competencies": [...],
  "TemMetricGroups": [...]
}
}
```

Validasyon Kuralları:

- **CamelCase** kullanılmalı; aksi halde alan deserialize edilmez ve null işlenir.
- Veri tipleri frontend ve backend arasında uyumlu olmalıdır. Örn: **lessonId** integer iken string gönderilirse hata oluşur.
- Zorunlu alanlar boş gönderilmemelidir.

4. Tablo Bazlı Mapleme

4.1 Training Form

- **Tablo:** **TRAINING_FORM**
- **Kaynak:** DTO alanları (lessonId, trainingId, formId, traineeName vs.)
- **Ek Alanlar:**
 - **Location Info** → **TRAINING.LocationId** join **LOCATION.Id**
 - **Contact Info** → **T_CONTACTS.WinUserName** join → LicenseType, LicenseName, Rank

4.2 Lesson Subjects

- **Tablolar:**
 - **TRAINING_FORM_DETAIL** (LessonSubjectId, Text, Order, IsMandatory)
 - **TRAINING_FORM_RESULTDETAIL** (Pass/Fail veya Complete/Incomplete)
 - **TRAINING_FORM_DETAILRESULT** (LessonDetailId, ParentId)
 - **Referans:** **T_LESSONDETAIL**, **T_LESSONRESULT** (tanım tabloları).
-

4.3 Competencies & Observational Behaviours (OB)

- **Tablolar:**
 - **TRAINING_FORM_COMPETENCY_RESULT** → CompetencyId, ResultText, Comment
 - **TRAINING_FORM_OB_RESULT** → OBId, ResultText (0, 1, -1)
 - **İlişki:** One-to-many (her Competency'ye bağlı çok sayıda OB olabilir).
-

4.4 TEM Metrics

- **Tablo:** **TRAINING_FORM_TEM_METRIC**
 - **Kayıt:** TemMetricGroup + TemMetricText + ResultText.
 - **Not:** **T_TEM_METRIC_GROUP** ve **T_TEM_METRICGROUPDETAIL** sadece tanım tablolarıdır, insert yapılmaz.
-

4.5 Completed Training

- **Tablo:** **EGRAIDING_COMPLETED_TRAINING**
- **Alanlar:** TrainingId, CompletedDate, IsPDFCreated (false → cron job → true).

- **İşlem:** Complete sonrası insert edilir, tekrar Complete yapılırsa update edilir (soft delete + yeniden insert).
-

5. Transaction ve Rollback

- Tüm süreç **transaction** içinde yürütülür.
 - Hata durumunda rollback devreye girer, hiçbir tabloya kalıcı insert yapılmaz.
 - Örnek: integer beklenen alana string gelmesi durumunda tüm kayıtlar geri alınır.
-

6. Update (Tekrar Complete) Senaryosu

- Instructor aynı eğitimi tekrar **Complete** ederse:
 1. Önceki kayıtlar **IsDeleted=1** olarak soft delete edilir.
 2. Tüm tablolar yeniden insert edilir.
 - Veri çoklaması engellenir, güncel bilgiler işlenir.
-

7. Test Senaryoları

1. **Training Form:** DTO alanlarının doğru işlendiği ve join edilen bilgilerin eksiksiz geldiği doğrulanmalı.
2. **Lesson Subjects:** Subject sayısı kadar kayıt oluşmalı. Pass/Fail veya Complete/Incomplete doğru eşlenmeli.
3. **Competency & OB:** One-to-many ilişkisi doğru test edilmeli.
4. **TEM Metrics:** Yalnızca **TRAINING_FORM_TEM_METRIC**'e insert edilmeli.
5. **Transaction:** Yanlış tip gönderilerek rollback mekanizması test edilmeli.

6. **Update:** Eski kayıtların soft delete edilip yeni kayıtların yeniden insert edildiği doğrulanmalı.
 7. **Completed Training:** **IsPDFCreated** başlangıçta false olmalı; job sonrası true güncellenmeli.
-

8. SQL Kontrol Örnekleri

-- Training & Form

```
SELECT * FROM TRAINING_FORM WHERE TRAININGID=285431 AND LESSONID=157;  
SELECT * FROM EGRADING_FORM_DATA WHERE TrainingId=285431;
```

-- Lesson Subjects

```
SELECT * FROM TRAINING_FORM_DETAIL WHERE TRAININGFORMID IN  
(88878,88879);  
SELECT * FROM TRAINING_FORM_RESULTDETAIL WHERE TRAININGID=285431 AND  
LESSONID=157;
```

-- Competency & OB

```
SELECT * FROM TRAINING_FORM_COMPETENCY_RESULT WHERE  
TRAINING_FORM_ID IN (88878,88879);  
SELECT * FROM TRAINING_FORM_OB_RESULT WHERE  
TRAINING_FORM_COMPETENCY_RESULT_ID IN (...);
```

-- TEM Metrics

```
SELECT * FROM TRAINING_FORM_TEM_METRIC WHERE TRAINING_FORM_ID IN  
(88878,88879);
```

-- Completed Training

```
SELECT * FROM EGRAIDING_COMPLETED_TRAINING WHERE TrainingId=285431;
```

9. Sonuç

Bu doküman;

- **JSON formatı,**
- **Tablo ilişkileri,**
- **Transaction & rollback mantığı,**

- Update/soft delete senaryosu,
- Test yöntemleri ve SQL kontrolleri

ile **E-Grading Mapleme Süreci'nin uçtan uca resmi bir tanımıdır.**