

WebPush Sistemi Veri Modeli

Temel Varlıklar ve İlişkiler

Aşağıda WebPush bildirim sistemimizde kullanılan veri modeli ve temel varlıklar arasındaki ilişkiler detaylandırılmıştır. Bu bilgiler, sistem tasarımı ve veritabanı yapısını anlamak için kullanılabilir.

1. Kullanıcı (User)

- Alanlar:

- user_id (PK)
- email
- first_name
- last_name
- created_at
- updated_at
- status (active, inactive)
- preferences_json
- last_login

- İlişkiler:

- Bir kullanıcının birden çok push aboneliği olabilir (1:n)
- Bir kullanıcı birden çok segmente ait olabilir (n:m)
- Bir kullanıcı birden çok role sahip olabilir (n:m)
- Bir kullanıcının birden çok etkileşim kaydı olabilir (1:n)

2. Push Abonelik (WebPush Subscription)

- Alanlar:

- subscription_id (PK)
- user_id (FK)
- endpoint
- p256dh_key
- auth_key
- user_agent
- created_at
- updated_at
- last_push_status
- last_push_time
- is_valid

- İlişkiler:

- Her abonelik bir kullanıcıya aittir (n:1)
- Bir aboneliğin birden çok bildirim gönderimi olabilir (1:n)

3. Rol (Role)

- Alanlar:

- role_id (PK)

- name
- description
- created_at
- updated_at

- **İlişkiler:**

- Bir rol birden çok kullanıcıya atanabilir (n:m)
- Bir rolün birden çok izni olabilir (n:m)

4. İzin (Permission)

- **Alanlar:**

- permission_id (PK)
- name
- code
- description
- resource_type
- action_type
- created_at
- updated_at

- **İlişkiler:**

- Bir izin birden çok role atanabilir (n:m)

5. Segment

- **Alanlar:**

- segment_id (PK)
- name
- description
- type (static, dynamic, imported)
- filter_criteria_json
- created_at
- updated_at
- created_by_user_id (FK)
- last_calculated_at
- total_users_count

- **İlişkiler:**

- Bir segment birden çok kullanıcıyı içerebilir (n:m)
- Bir segment birden çok kampanyada kullanılabilir (1:n)
- Bir segment bir kullanıcı tarafından oluşturulur (n:1)

6. Şablon (Template)

- **Alanlar:**

- template_id (PK)
- name
- title
- body

- image_url
- action_url
- version
- created_at
- updated_at
- created_by_user_id (FK)
- variables_json
- category_tags

- **İlişkiler:**

- Bir şablon birden çok kampanyada kullanılabilir (1:n)
- Bir şablon bir kullanıcı tarafından oluşturulur (n:1)
- Bir şablonun birden çok versiyonu olabilir (1:n)

7. Kampanya (Campaign)

- **Alanlar:**

- campaign_id (PK)
- name
- description
- type (one-time, recurring, trigger-based, ab_test)
- status (draft, active, paused, completed, failed)
- schedule_time
- recurrence_pattern_json
- created_at
- updated_at
- created_by_user_id (FK)
- segment_id (FK)
- template_id (FK)
- personalization_settings_json

- **İlişkiler:**

- Bir kampanya bir segmenti hedefler (n:1)
- Bir kampanya bir şablonu kullanır (n:1)
- Bir kampanya bir kullanıcı tarafından oluşturulur (n:1)
- Bir kampanyanın birden çok bildirim gönderimi olabilir (1:n)
- Bir kampanyanın birden çok A/B test varyantı olabilir (1:n)
- Bir kampanya birden çok tetikleyiciye sahip olabilir (1:n)

8. Tetikleyici (Trigger)

- **Alanlar:**

- trigger_id (PK)
- name
- event_type
- conditions_json
- delay_minutes
- created_at
- updated_at
- created_by_user_id (FK)

- campaign_id (FK)
- segment_id (FK)
- template_id (FK)
- status (active, inactive)
- last_triggered_at

- **İlişkiler:**

- Bir tetikleyici bir kampanyaya bağlıdır (n:1)
- Bir tetikleyici bir segmenti hedefleyebilir (n:1)
- Bir tetikleyici bir şablon kullanır (n:1)
- Bir tetikleyicinin birden çok tetiklenme kaydı olabilir (1:n)

9. Bildirim (Notification)

- **Alanlar:**

- notification_id (PK)
- user_id (FK)
- subscription_id (FK)
- campaign_id (FK)
- template_id (FK)
- trigger_id (FK)
- scheduled_time
- sent_time
- status (scheduled, sent, delivered, failed, canceled)
- personalized_content_json
- error_details
- retry_count
- created_at
- updated_at

- **İlişkiler:**

- Her bildirim bir kullanıcıya gönderilir (n:1)
- Her bildirim bir aboneliğe gönderilir (n:1)
- Her bildirim bir kampanyaya bağlıdır (n:1)
- Her bildirim bir şablonu kullanır (n:1)
- Her bildirim bir tetikleyici tarafından başlatılabilir (n:1)
- Bir bildirimin birden çok etkileşim kaydı olabilir (1:n)

10. Etkileşim (Interaction)

- **Alanlar:**

- interaction_id (PK)
- notification_id (FK)
- user_id (FK)
- type (delivery, open, click, conversion)
- timestamp
- metadata_json
- device_info
- location_info
- created_at

- **İlişkiler:**

- Her etkileşim bir bildirime bağlıdır (n:1)
- Her etkileşim bir kullanıcıya aittir (n:1)

11. A/B Test

- **Alanlar:**

- test_id (PK)
- campaign_id (FK)
- name
- description
- status (setup, active, analyzing, completed)
- start_time
- end_time
- success_metric
- created_at
- updated_at
- created_by_user_id (FK)
- winning_variant_id
- is_statistically_significant

- **İlişkiler:**

- Her test bir kampanyaya aittir (n:1)
- Bir testin birden çok varyantı olabilir (1:n)
- Bir test bir kullanıcı tarafından oluşturulur (n:1)

12. Test Varyantı (Test Variant)

- **Alanlar:**

- variant_id (PK)
- test_id (FK)
- template_id (FK)
- name
- distribution_percentage
- performance_metrics_json
- created_at
- updated_at

- **İlişkiler:**

- Her varyant bir teste aittir (n:1)
- Her varyant bir şablonu kullanır (n:1)
- Bir varyantın birden çok bildirim gönderimi olabilir (1:n)

13. Webhook

- **Alanlar:**

- webhook_id (PK)
- name
- description

- endpoint_url
- event_types_json
- secret_key
- status (active, inactive)
- created_at
- updated_at
- created_by_user_id (FK)
- headers_json
- retry_settings_json
- last_triggered_at

- **İlişkiler:**

- Bir webhook bir kullanıcı tarafından oluşturulur (n:1)
- Bir webhook'un birden çok tetiklenme kaydı olabilir (1:n)

14. Webhook Tetiklenme (Webhook Trigger)

- **Alanlar:**

- webhook_trigger_id (PK)
- webhook_id (FK)
- event_type
- event_data_json
- triggered_at
- status (pending, success, failed, retrying)
- response_code
- response_body
- retry_count
- next_retry_at
- completed_at

- **İlişkiler:**

- Her tetiklenme bir webhook'a aittir (n:1)

15. CDP Entegrasyon (CDP Integration)

- **Alanlar:**

- integration_id (PK)
- user_id (FK)
- cdp_user_id
- sync_status
- last_sync_time
- sync_direction (inbound, outbound, bidirectional)
- profile_data_json
- created_at
- updated_at

- **İlişkiler:**

- Her CDP entegrasyonu bir kullanıcıya bağlıdır (n:1)
- Bir CDP entegrasyonunun birden çok senkronizasyon kaydı olabilir (1:n)

16. CEP Karar (CEP Decision)

- Alanlar:

- decision_id (PK)
- user_id (FK)
- communication_id
- timestamp
- channels_evaluated_json
- selected_channel
- decision_factors_json
- performance_data_json
- created_at

- İlişkiler:

- Her CEP kararı bir kullanıcıya ilişkindir (n:1)
- Her CEP kararı bir bildirime bağlı olabilir (1:1)

Harici Sistemlerle Entegrasyonlar

Kimlik Doğrulama Sistemi

- Kullanıcı kimlik doğrulama ve yetkilendirme harici bir sistemden sağlanır
- Her API isteği bir token ve müşteri ID'si içerir
- Token doğrulaması harici sistem tarafından gerçekleştirilir
- Sistem rollerle eşleştirilerek yetkilendirme yapılır

CDP Sistemi

- Kullanıcı profil verileri CDP sistemi ile senkronize edilir
- Segment bilgileri CDP'den alınabilir veya CDP'ye gönderilebilir
- Etkileşim verileri CDP'ye aktararak müşteri profilleri zenginleştirilir

CEP Sistemi

- İletişim kanalı kararları CEP sistemi tarafından sağlanır
- WebPush bildirimleri CEP tarafından belirlenen optimal zamanlarda gönderilir
- Performans verileri CEP algoritmasını iyileştirmek için geri gönderilir

Veri Akışı

1. Kullanıcı web sitesini ziyaret eder ve bildirimlere abone olur
2. Kullanıcı segmentlere eklenir (manuel veya otomatik olarak)
3. Pazarlama ekibi şablonlar ve kampanyalar oluşturur
4. Kampanyalar zamanlanır veya tetikleyiciler ayarlanır
5. Bildirimler hedef kitleye gönderilir
6. Kullanıcı etkileşimleri kaydedilir ve analiz edilir
7. Performans verileri raporlanır ve webhook'lar aracılığıyla dış sistemlere aktarılır
8. CDP ve CEP sistemleriyle veri senkronizasyonu yapılır

Bu veri modeli, WebPush bildirim sistemimizin temel yapı taşlarını ve aralarındaki ilişkileri göstermektedir. Wireframe ve sistem tasarımı çalışmalarında bu model referans alınarak ilerlenmelidir.