|  |
| --- |
| AEGON Emeklilik ve Hayat |
| PDF-Service Web API Açıklamaları |
| İş Dokümanı |



TechSiN Bilişim Çözümleri

Ali Nihat Tarlan Cad. Ertaç Sok. No:10 Kat:4 34572 İçerenköy Ataşehir/İSTANBUL

Tel: +90 216 573 01 84 | Fax: +90 216 573 01 87

[www.techsin.com.tr](http://www.techsin.com.tr)

|  |  |  |
| --- | --- | --- |
| Doküman Kodu | : | Aegon-PDF-Service-Manual |
| Tarih | : | 07.11.2018 |
| Sürüm | : | 1.0.0 |
| Hazırlayan(lar) | : | Suat İMAMOĞLU-Gökhan YENER |
| Onaylayan(lar) | : | İsmail Tellioğlu, FUAT SATIR |

Doküman Sürüm Geçmişi

|  |  |  |  |
| --- | --- | --- | --- |
| **Tarih** | **Sürüm** | **Değişiklikleri Yapan** | **Sürüm Açıklaması** |
| 17.05.2018 | 1.0.0 | Suat İMAMOĞLU | Doküman oluşturuldu. |
| 07.11.2018 | 1.0.1 | Gökhan YENER | Doküman güncellendi. |

İçerik

[Doküman Sürüm Geçmişi 2](#_Toc529455616)

[1 GENEL TANIMLAR 5](#_Toc529455617)

[1.1 Amaç 5](#_Toc529455618)

[1.2 Altyapı 5](#_Toc529455619)

[1.3 RESTful API 5](#_Toc529455620)

[1.4 Docker Konfigürasyonu 5](#_Toc529455621)

[1.5 Symfony Konfigürasyonu 5](#_Toc529455622)

[1.6 Sigorta Türleri 6](#_Toc529455623)

[1.7 PDF Dosya Türleri 6](#_Toc529455624)

[1.8 PDF Dosyaları 7](#_Toc529455625)

[1.9 Database Yapısı 7](#_Toc529455626)

[1.9.1 aegon\_pdf Tablosu 7](#_Toc529455627)

[1.9.2 aegon\_pdf\_log Tablosu 7](#_Toc529455628)

[1.9.3 Database Tablolarını Oluşturmak için SQL 8](#_Toc529455629)

[2 KURULUM 9](#_Toc529455630)

[2.1 Ön Hazırlık 9](#_Toc529455631)

[2.2 Start 9](#_Toc529455632)

[2.3 Stop 9](#_Toc529455633)

[2.4 Moreum Arşivleme Komutu 9](#_Toc529455634)

[2.5 API Dokümanı (Swagger) Erişimi 10](#_Toc529455635)

[3 WEBSERVİS KULLANIMI 11](#_Toc529455636)

[3.1 KONFİGÜRASYON BİLGİLERİNİ SORGULAMA 11](#_Toc529455637)

[3.1.1 Sigorta Türleri 11](#_Toc529455638)

[3.1.2 PDF Dosya Tipleri 12](#_Toc529455639)

[3.2 PDF DOSYASI LİSTELEME 13](#_Toc529455640)

[3.3 PDF OLUŞTURMA VE GÜNCELLEME 14](#_Toc529455641)

[3.3.1 Genel Açıklamalar 14](#_Toc529455642)

[3.3.2 İhtiyaç Teyit Formu 16](#_Toc529455643)

[3.3.3 Teklif Özeti 17](#_Toc529455644)

[3.3.4 Başvuru Formu 18](#_Toc529455645)

[3.3.5 Bilgilendirme Formu 21](#_Toc529455646)

[3.3.6 Genel ve Özel Şartlar 21](#_Toc529455647)

[3.3.7 Poliçe 22](#_Toc529455648)

[3.3.8 Poliçe Okuma Kılavuzu 23](#_Toc529455649)

[4 Dikkat Edilmesi Gerekenler ve Alınabilecek Hatalar 23](#_Toc529455650)

[4.1.1 Teklif Özeti Pdf Oluşturma Hatası : 23](#_Toc529455651)

[4.1.2 400 Bad Request Hatası : 23](#_Toc529455652)

[4.1.3 500 Internal Server Error Hatası : 23](#_Toc529455653)

[4.1.4 Cronjob log hataları 23](#_Toc529455654)

[4.2 Loglama: 24](#_Toc529455655)

[4.2.1 Database Loglarından Tekrardan Pdf Üretme 25](#_Toc529455656)

[5 Web Servis Topoloji ve Mimarisi 26](#_Toc529455657)

# GENEL TANIMLAR

## Amaç

Projenin amacı, “Aegon Kolay Katılım” süreci esnasında müşterinin girmiş olduğu bilgileri derleyip ihtiyaç duyulan PDF dokümanlarının webservis aracılığıyla oluşturulmasıdır.

## Altyapı

Altyapı olarak PHP7, MySQL, NGINX ve Symfony Framework v.3.4 kullanılmıştır. Sistem “docker-compose” desteği sayesinde herhangi bir Linux makinesinde kolayca çalıştırılabilir durumdadır (bk. 2. Kurulum). Docker üzerinden PHP ve NGINX container’leri çalıştırılır; veritabanı (MySQL) ise docker yapısı içerisinde olmayıp, mevcut olan harici bir veritabanı kullanılmıştır.

## RESTful API

Webservis RESTful API olarak tasarlanmıştır. API dokümantasyonu Swagger formatında olup, sistemin çalıştırıldığı makinede erişilebilir durumdadır.

API Dokümantasyonu URL: http://localhost:8888/doc

## Docker Konfigürasyonu

Docker konfigürasyonu *docker-compose.yml* dosyası üzerinden yapılır. Özellikle Nginx ayarı olarak hangi portun kullanılacağı önem taşıyabilir. Varsayılan port olarak 8888 ayarlanmıştır. Ancak aynı sunucuda bu port başka bir uygulama için kullanılmaktyasa *docker-compose.yml* dosyası içinde Nginx ayarlarından değiştirilebilir:

nginx:

ports:

- "8888:80"

## Symfony Konfigürasyonu

Symfony projesinin temel konfigürasyonu *app/config/parameters.yml* dosyası içinde yapılır:

parameters:

database\_driver: pdo\_mysql

database\_host: 10.184.33.111

database\_port: 3306

database\_name: kolaykatilim

database\_user: kolaykatilim

database\_password: inomera

moreum\_archival\_wsdl: 'http://10.184.33.34/TerraDocsWS/Archival.svc?wsdl'

moreum\_retrieval\_wsdl: 'http://10.184.33.34/TerraDocsWS/Retrieval.svc?wsdl'

Öncelikle database\_host, database\_name, database\_user ayarları veritabanı bağlantısı için önemlidir ve kontrol edilimelidir.

Ayrıca moreum\_archival\_wsdl ve moreum\_retrieval\_wsdl ayarları Moreum arşivleme sistemi için önem taşır ve kontrol edilmelidir.

## Sigorta Türleri

Webservis, Kolay Katılım projesi kapsamında oluşturulabilen 14 farklı sigorta çeşidi için PDF dosyası üretebilmektedir:

|  |  |
| --- | --- |
| **Sigorta Türü** | **Sigorta Kodu (slug)** |
| 5+5 Ödüllü Birikim Hayat Sigortası | 5-5-odullu-birikim |
| Eğitim İçin Hayat Sigortası | egitim-icin-hayat-sigortasi |
| Ferdi Kazalı Prim İadeli Hayat Sigortası | ferdi-kazali-prim-iadeli |
| Ferdi Kazalı Prim İadeli Hayat Sigortası (Euro) | ferdi-kazali-prim-iadeli-euro |
| Grup Turuncu Elma Hayat Sigortası | grup-turuncu-elma |
| Ödeme Güvence Sigortası | odeme-guvence |
| Ödüllü Birikim Hayat Sigortası | odullu-birikim |
| Prim İadeli Hayat Sigortası | prim-iadeli |
| Prim İadeli Hayat Sigortası (Euro) | prim-iadeli-euro |
| 3 Yıllık Prim İadeli Hayat Sigortası | prim-iadeli-3-yillik |
| 5 Yıllık Prim İadeli Hayat Sigortası | prim-iadeli-5-yillik |
| Tek Prim Ödemeli Hayat Sigortası | tek-prim-odemeli |
| 5 Yılıık Tek Prim Ödemeli Hayat Sigortası | tek-prim-odemeli-5-yillik |
| Turuncu Elma Hayat Sigortası | turuncu-elma |

## PDF Dosya Türleri

Her sigorta çeşidi için 7 farklı PDF dosyası üretilebilmektedir:

|  |  |
| --- | --- |
| **PDF Türü** | **PDF Kodu (slug)** |
| İhtiyaç Teyit Formu | requirements-confirmation |
| Teklif Özeti | offer-summary |
| Başvuru Formu | application-form |
| Bilgilendirme Formu | information-form |
| Genel ve Özel Şartlar | terms |
| Poliçe | insurance-policy |
| Poliçe Okuma Kılavuzu | policy-manual |

## PDF Dosyaları

PDF dosyaları proje içerisindeki “web/download” ana klasörü altındaki alt klasörlere kaydedilir. Alt klasör ismi ve dosya ismi veritabanına kaydedilmektedir (“dir\_name” ve “file\_name” alanlarında).

## Database Yapısı

Webservise ait iki tablo mevcuttur: “aegon\_pdf” ve “aegon\_pdf\_log”.

### aegon\_pdf Tablosu

***aegon\_pdf*** tablosu güncel/geçerli olan PDF dosyalarının bilgilerini tutmaktadır.

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Örnek** | **Açıklama** |
| id | int(11) | 1 | İntern ID (autoincrement) |
| moreum\_document\_id | int(11) | 83905 | Dosya Moreum’a kaydedildiyse, Moreum’un “DocumentId” değeri, aksi takdirde NULL |
| personal\_id\_number | varchar(11) | 12345678901 | TCKN |
| pol\_id | varchar(64) | 2223524 | POLID |
| policy\_number | varchar(64) | ROP\_AD\_3017060001 | Poliçe No |
| insurance\_slug | varchar(128) | 5-5-odullu-birikim | Sigorta türü kodu (slug) |
| pdf\_type\_slug | varchar(128) | 01-ihtiyac-teyit | PDF türü kodu (slug) |
| dir\_name | varchar(128) | 1234567/89 | Dosyanın kaydedildiği alt klasör |
| file\_name | varchar(255) | aegon\_2223524\_5-5-odullu-birikim\_01-ihtc.pdf | Dosya ismi |
| json\_data | longtext | - | Webservise gönderilen JSON objesi |
| json\_processed | longtext | - | JSON objesinin derlenip PDF oluşturmaya uygun hali |
| created\_at | datetime |  | Oluşturma tarihi |
| updated\_at | datetime |  | Güncelleme tarihi |

### aegon\_pdf\_log Tablosu

***aegon\_pdf\_log*** tablosu, *aegon\_pdf* tablosunun birebir kopyasıdır. *aegon\_pdf* tablosunda her yeni kayıt oluşturulduğunda ve/veya var olan bir kayıt güncellendiğinde, ilgili kaydın birebir kopyası *aegon\_pdf\_log* tablosuna da yeni bir kayıt olarak kaydedilmektedir.

Böylelikle yeni oluşturulan veya güncellenen PDF dosyalarının geçmişi kayıt altına alınmaktadır.

### Database Tablolarını Oluşturmak için SQL

MySQL database tablolarını oluşturmak için aşağıdaki SQL kodları çalıştırılabilir:

CREATE TABLE `aegon\_pdf` (

`id` int(11) NOT NULL AUTO\_INCREMENT,

`personal\_id\_number` varchar(11) COLLATE utf8\_unicode\_ci NOT NULL,

`insurance\_slug` varchar(128) COLLATE utf8\_unicode\_ci NOT NULL,

`pdf\_type\_slug` varchar(128) COLLATE utf8\_unicode\_ci NOT NULL,

`dir\_name` varchar(128) COLLATE utf8\_unicode\_ci NOT NULL,

`file\_name` varchar(255) COLLATE utf8\_unicode\_ci NOT NULL,

`json\_data` longtext COLLATE utf8\_unicode\_ci,

`json\_processed` longtext COLLATE utf8\_unicode\_ci,

`created\_at` datetime DEFAULT NULL,

`updated\_at` datetime DEFAULT NULL,

`moreum\_document\_id` int(11) DEFAULT NULL,

`pol\_id` varchar(64) COLLATE utf8\_unicode\_ci NOT NULL,

`policy\_number` varchar(64) COLLATE utf8\_unicode\_ci DEFAULT NULL,

PRIMARY KEY (`id`),

KEY `idx\_pdf\_personal\_id\_number` (`personal\_id\_number`),

KEY `idx\_pdf\_insurance\_slug` (`insurance\_slug`),

KEY `idx\_pdf\_pdf\_type\_slug` (`pdf\_type\_slug`),

KEY `idx\_pdf\_dir\_name` (`dir\_name`),

KEY `idx\_pdf\_file\_name` (`file\_name`),

KEY `idx\_pdf\_created\_at` (`created\_at`),

KEY `idx\_pdf\_updated\_at` (`updated\_at`),

KEY `idx\_pdf\_moreum\_document\_id` (`moreum\_document\_id`),

KEY `idx\_pdf\_pol\_id` (`pol\_id`),

KEY `idx\_pdf\_policy\_number` (`policy\_number`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci;

CREATE TABLE `aegon\_pdf\_log` (

`id` int(11) NOT NULL AUTO\_INCREMENT,

`personal\_id\_number` varchar(11) COLLATE utf8\_unicode\_ci NOT NULL,

`insurance\_slug` varchar(128) COLLATE utf8\_unicode\_ci NOT NULL,

`pdf\_type\_slug` varchar(128) COLLATE utf8\_unicode\_ci NOT NULL,

`dir\_name` varchar(128) COLLATE utf8\_unicode\_ci NOT NULL,

`file\_name` varchar(255) COLLATE utf8\_unicode\_ci NOT NULL,

`json\_data` longtext COLLATE utf8\_unicode\_ci,

`json\_processed` longtext COLLATE utf8\_unicode\_ci,

`created\_at` datetime DEFAULT NULL,

`updated\_at` datetime DEFAULT NULL,

`moreum\_document\_id` int(11) DEFAULT NULL,

`pol\_id` varchar(64) COLLATE utf8\_unicode\_ci NOT NULL,

`policy\_number` varchar(64) COLLATE utf8\_unicode\_ci DEFAULT NULL,

PRIMARY KEY (`id`),

KEY `idx\_pdf\_log\_personal\_id\_number` (`personal\_id\_number`),

KEY `idx\_pdf\_log\_insurance\_slug` (`insurance\_slug`),

KEY `idx\_pdf\_log\_pdf\_type\_slug` (`pdf\_type\_slug`),

KEY `idx\_pdf\_log\_dir\_name` (`dir\_name`),

KEY `idx\_pdf\_log\_file\_name` (`file\_name`),

KEY `idx\_pdf\_log\_created\_at` (`created\_at`),

KEY `idx\_pdf\_log\_updated\_at` (`updated\_at`),

KEY `idx\_pdf\_moreum\_document\_id` (`moreum\_document\_id`),

KEY `idx\_pdf\_pol\_id` (`pol\_id`),

KEY `idx\_pdf\_policy\_number` (`policy\_number`)

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci;

# KURULUM

## Ön Hazırlık

Uygulamanın çalıştırılacağı sunucuda Docker altyapısının kurulmuş olması gerekir.

Uygulama ZIP dosyası halinde teslim edilir. Bu ZIP dosyası sistemin çalışacağı sunucuya yüklenip öngörülen klasör içinde açılmalıdır (“extract” edilmelidir).

Örn.

cp pdf-service-1.0.0.zip /var/www  
cd /var/www/   
unzip pdf-service-1.0.0.zip

## Start

Uygulamayı çalıştırmak için konsol (“command line” - CLI) ile önceki adımda extract edilmiş olan uygulamanın ana klasörüne geçip *app-start.sh* scripti çalıştırılmalıdır.

Örn.

cd /var/www/pdf-service-1.0.0/  
./app-start.sh

Bu şekilde iki tane Docker container’i çalıştırılmış olacaktır (nginx ve php).

## Stop

Uygulamayı durdurmak için yine konsol ile uygulamanın ana klasörüne geçip, *app-stop.sh* scripti çalıştırılmalıdır:

Örn.

cd /var/www/pdf-service-1.0.0/  
./app-stop.sh

## Moreum Arşivleme Komutu

Servis üzerinden oluşturulan tüm PDF dosyalarının ayrıca Moreum Arşivleme sistemine aktarılması gerekmektedir.

Bu işlem için belirli aralıklarla bir komutun çalıştırılması gerekmektedir. Bunun için servisin çalıştırıldığı makinede (“docker host”) bir cronjob’un ayarlanması gerekli. Bu cronjob örn. 5 veya 10 dakikada bir çalışmalıdır.

Bu şekilde ayarlanmış olan cronjob, şu komutları çalıştırmalıdır:

cd /var/www/pdf-service-1.0.0/  
docker-compose exec php bin/console app:moreum:archival –env=prod

Bu komut henüz arşivlenmemiş olan tüm PDF dosyalarını Moreum’a aktarır ve aktarılan her PDF dosyası için veritabanındaki aegon\_pdf tablosunda moreum\_document\_id alanına Moreum’dan dönen ID’yi kaydeder.

## API Dokümanı (Swagger) Erişimi

Detaylı API dokümantasyonu Swagger formatında olup, sistemin çalıştırıldığı makinede erişilebilir durumdadır:

http://localhost:8888/doc

# WEBSERVİS KULLANIMI

Webservis kullanımı öncelikle API Dokümantasyonu sayfasından deneyimlenebilir ve test edilebilir:

http://localhost:8888/doc

Bu sayfada tüm API endpoint’leri ve kullanılan model yapısı, örnek sorgular vs. açıklanmıştır. Webservisin sunduğu tüm endpointler için detaylı örnek sorgular mevcut olup, bu sayfa üzerinden doğrudan test edilebilir.

## KONFİGÜRASYON BİLGİLERİNİ SORGULAMA

İlk aşamada webservisin sunduğu “Sigorta Türlerini” ve “PDF Dosya Tiplerini” sorgulamak gerekebilir. Bu bilgiler diğer sorguları oluştururmak için gereklidir.

### Sigorta Türleri

Sigorta türlerini sorgulamak için kullanılan endpoint:

|  |  |
| --- | --- |
| **Metod** | **URL** |
| GET | http://localhost:8888/api/configuration/insurance-slugs |
| Response | {  "5-5-odullu-birikim": "5+5 Ödüllü Birikim Hayat Sigortası",  "egitim-icin-hayat-sigortasi": "Eğitim İçin Hayat Sigortası",  "ferdi-kazali-prim-iadeli": "Ferdi Kazalı Prim İadeli Hayat Sigortası",  "grup-turuncu-elma": "Grup Turuncu Elma Hayat Sigortası",  "odeme-guvence": "Ödeme Güvence Sigortası",  "odullu-birikim": "Ödüllü Birikim Hayat Sigortası",  "prim-iadeli": "Prim İadeli Hayat Sigortası",  "prim-iadeli-3-yillik": "3 Yıllık Prim İadeli Hayat Sigortası",  "prim-iadeli-5-yillik": "5 Yıllık Prim İadeli Hayat Sigortası",  "tek-prim-odemeli": "Tek Prim Ödemeli Hayat Sigortası",  "tek-prim-odemeli-5-yillik": "5 Yılıık Tek Prim Ödemeli Hayat Sigortası",  "turuncu-elma": "Turuncu Elma Hayat Sigortası"  } |

### PDF Dosya Tipleri

Webservisin oluşturabildiği PDF dosya tiplerini sorgulamak için:

|  |  |
| --- | --- |
| **Metod** | **URL** |
| GET | http://localhost:8888/api/configuration/pdf-type-slugs |
| Response | {  "requirements-confirmation": "İhtiyaç Teyit Formu",  "offer-summary": "Teklif Özeti",  "application-form": "Başvuru Formu",  "information-form": "Bilgilendirme Formu",  "terms": "Genel ve Özel Şartlar",  "insurance-policy": "Poliçe",  "policy-manual": "Poliçe Okuma Kılavuzu"  } |

## PDF DOSYASI LİSTELEME

Herhangi bir müşteriye ait daha önceden oluşturulmuş olan PDF dosyalarını listelemek için kullanılır.

|  |  |
| --- | --- |
| **Metod** | **URL** |
| GET | http://localhost:8888/api/pdf/get-by-customer/{personalIdNumber}/{polId}/{insuranceSlug} |

|  |  |
| --- | --- |
| **Parametre** | **Açıklama** |
| personalIdNumber | Sigortalının TC Kimlik No’su |
| polId | Sigorta POLID’si |
| insuranceSlug | Sigorta türünün kodu |

|  |  |
| --- | --- |
| Örnek | http://localhost:8888/api/pdf/get-by-customer/12345678901/2223524/5-5-odullu-birikim |
| Response | [  {  "id": 73,  "moreumDocumentId": 84249,  "personalIdNumber": "12345678901",  "polId": "2223524",  "policyNumber": "ROP\_AD\_3017060001",  "insuranceSlug": "5-5-odullu-birikim",  "pdfTypeSlug": "06-police",  "dirName": "1234567/89",  "fileName": "aegon\_2223524\_5-5-odullu-birikim\_06-polc.pdf",  "url": "/download/1234567/89/aegon\_2223524\_5-5-odullu-birikim\_06-polc.pdf"  }  ] |

Bu örnekte oluşturulmuş olan PDF dosyasını şu URL’den indirmek mümkündür:

*http://localhost:8888/download/1234567/89/aegon\_2223524\_5-5-odullu-birikim\_06-polc.pdf*

Dosyalara ayrıca webservisin çalıştırıldığı sunucuda (docker host) şu klasörden de ulaşılabilir:

Örn.:

/var/www/pdf-service-1.0.0/web/  
 download/1234567/89/aegon\_2223524\_5-5-odullu-birikim\_06-polc.pdf

## PDF OLUŞTURMA VE GÜNCELLEME

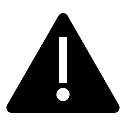
### Genel Açıklamalar

#### Endpointler ve POST / PUT Metodları

PDF dosyası oluşturmak için her PDF Dosya Türüne özel ayrı bir endpoint bulunmaktadır.

PDF oluşturma veya güncelleme için yapılan tüm istekler POST veya PUT metoduyla yapılmalıdır.

Her endpoint, request metoduna (POST veya PUT) bağlı olarak, RESTful API yapısına uygun olacak şekilde, yeni bir dosya oluşturur (POST = create) veya var olan bir dosyayı günceller (PUT = update).

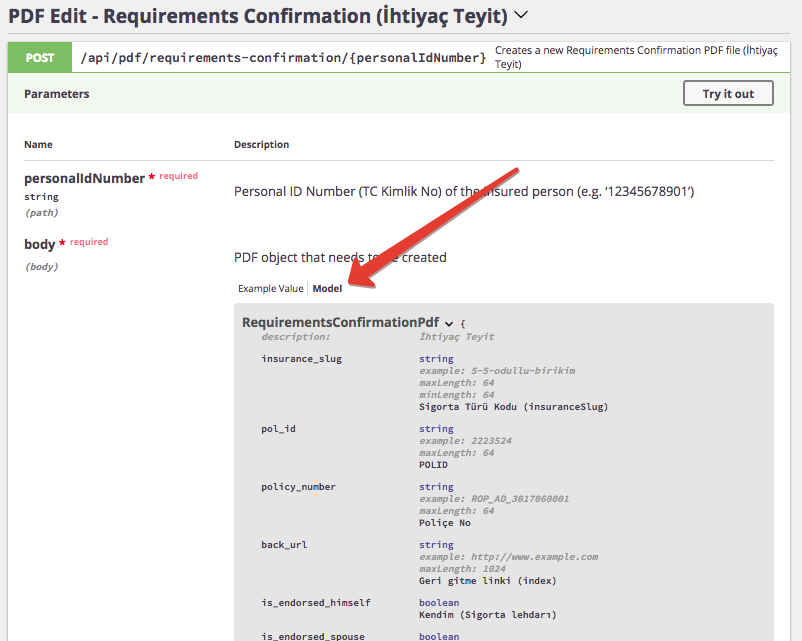
Henüz oluşturulmamış olan bir PDF dosyası için PUT (update) metodu kullanılamaz, aksi takdirde sistem hata verir. Aynı şekilde, var olan bir PDF dosyası için POST (create) metodu kullanılamaz, yoksa sistem hata verir.

Tüm endpointlerin detayları örnekleriyle beraber API dokümantasyon sayfasında (http://localhost:8888/doc) belirtilmiştir.

Ağağıdaki açıklamalar, her PDF türü için özellikle dikkat edilmesi gereken ayrıntıları içerir.

#### Model Yapısı

Her isteğin request body kısmında, PDF dosyasının içeriğini belirleyen JSON objesi bulunmalıdır. Bu JSON objeleri, her PDF türü için farklıdır ve API dokümantasyon sayfasında ilgili endpointun altında örnekleriyle beraber detaylıca açıklanmıştır:



Tüm JSON objelerinin (Model) ortak ve zorunlu alanları şunlardır:

|  |  |
| --- | --- |
| insurance\_slug | Sigorta türü kodu (slug), örn. “5-5-odullu-birikim” |
| pol\_id | Sigortanın POLID’si |
| policy\_number | Sigortanın Poliçe No’su |
| back\_url | Geri gitme linki (PDF dosyasının her sayfasının alt kısmında “İndex” yazılı bir imaj linki basılır). Kullanıcı PDF dosyasını tarayıcı içinde açtığında bu linke tıklayınca geldiği sayfaya geri gidecektir. |

Tüm endpoint URL’lerinin ortak parametresi şudur:

|  |  |
| --- | --- |
| personalIdNumber | Sigortalı kişinin TC Kimlik No’su |

### İhtiyaç Teyit Formu

İhtiyaç Teyit Formu oluştururken gönderilen JSON objesinde özellikle dikkat edilmesi gereken alanlar:

|  |  |
| --- | --- |
| POST/PUT | /api/pdf/requirements-confirmation/{personalIdNumber} |
| insurer\_sign\_image\_url | Sigorta ettiren kişinin imzasının bulunduğu bir imajın URL’sidir. Bu URL’ye webservisin kurulu olduğu sunucudan erişilebilmelidir. Bu imaj, PDF’in imza kısmına basılacaktır. |



### Teklif Özeti

Teklif Özeti endpoint’u, aslında var olan ve farklı bir webservis tarafından üretilmiş olan teklif özeti PDF dosyalarını *Kolay Kullanım* projesi tasarımıyla süslemektedir. Dolayısıyla bu endpoint’e var olan PDF dosyasının URL’si bildirilmelidir (offer\_summary\_pdf\_url).

Dikkat edilmesi gereken alanlar:

|  |  |
| --- | --- |
| POST/PUT | /api/pdf/offer-summary/{personalIdNumber} |
| offer\_summary\_pdf\_url | Var olan teklif özeti PDF dosyasının URL’si bu alana yazılmalıdır. İlgili URL’ye webservisin kurulu olduğu sunucudan erişilebilmelidir. |
| insurer\_initials\_image\_url | Sigorta ettiren kişinin parafının bulunduğu bir imajın URL’sidir. Bu URL’ye webservisin kurulu olduğu sunucudan erişilebilmelidir. Bu imaj, tüm PDF sayfalarının alt kısmına basılacaktır. |



### Başvuru Formu

Başvuru Formu diğer PDF türlerine nazaran daha karmaşık olan bir PDF dosyası tipidir. Gönderilen JSON objesinin yapısına göre PDF’in sayfa sayısı, görsel yapısı ve içeriği büyük oranda değişmektedir. Bu yüzden aşağıdaki açıklamaların özenle dikkate alınması gerekir.

|  |  |
| --- | --- |
| POST/PUT | /api/pdf/application-form/{personalIdNumber} |
| guarantees | Array halinde 1 veya daha fazla teminat bilgisini içerir (*ApplicationFormGuarantee*). |
| credit\_cards | Array halinde 1 veya (maks.) 2 kredi kartı bilgisini içerir (*ApplicationFormCreditCard*). PDF dosyasına en fazla 2 tane kredi kartı bilgisi basılabilir. Bunlardan birinin is\_preferred bilgisi true (boolean) olmalıdır. |
| endorseds | Array halinde 1 veya daha fazla lehdar bilgisini içerir (*ApplicationFormEndorsed*). |
| mediator | Aracı bilgilerini içerir (*ApplicationFormMediator*). |
| insured\_person (1) | Sigortalanan kişinin bilgilerini içerir (*ApplicationFormInsuredPerson*). Null olamaz (1). |
| insurer\_person (1) | Sigortalayan kişinin (eğer gerçek kişiyse) bilgilerini içerir (*ApplicationFormInsurerPerson*).  Null olabilir (1). |
| insurer\_company (1) | Sigortalayan kişinin (eğer tüzel kişiyse, yani şirket ise) bilgilerini içerir (*ApplicationFormInsurerPerson*).  Null olabilir (1). |
| lifestyle | Yaşam tarzi bilgilerini içerir (*ApplicationFormLifestyle*). health\_declaration bilgisi (“Sağlık Beyan Metni”) null olduğu takdirde bu alan ve başlığı basılmaz. |
| health\_basic (2)(3) | 5 sorudan oluşan sağlık bilgilerini içerir (*ApplicationFormHealthBasic*). Null olabilir (2). |
| health\_extended (2)(3) | 12 sorudan oluşan sağlık bilgilerini içerir (*ApplicationFormHealthExtended*). Null olabilir (2). |
| confirmation (4) | *ApplicationFormConfirmation* objesini içerir (bk. aşağıdaki açıklamalar) |

1. insured\_person, insurer\_person, insurer\_company Kombinasyonları:  
   Bu alanların kombinasyonlarına göre “Sigortalayan” (gerçek/tüzel şahıs) bilgileri oluşturulur:
   1. insurer\_person = null AND insurer\_company = null  
      Sigortalayan = Sigortalanan kabul edilir (yani sigortalayan kişi, kendisi için sigorta oluşturmuştur)
   2. insurer\_person != null AND insurer\_company = null  
      Sigortalayan kişi gerçek şahıs ve sigortalanan şahıstan farklı biridir
   3. insurer\_person = null AND insurer\_company != null  
      Sigortalayan tüzel kişidir (şirket)
2. health\_basic, health\_extended Kombinasyonları:  
   Bu alanların kombinasyonlarına göre 5 sorulu veya 12 sorulu sağlık bilgileri basılır:
   1. health\_basic != null   
      5 sorudan oluşan “Sağlık Bilgileri” basılır
   2. health\_basic = null AND health\_extended != null  
      12 sorudan oluşan “Sağlık Bilgileri” basılır
3. “Sağlık Bilgileri” sayfasının altına bazı durumlarda *sürprim hesaplamaları* basılmaktadır.  
   Bu hesaplamaların basılması için, health\_basic (*ApplicationFormHealthBasic*) veya health\_extended (*ApplicationFormHealthExtended*) objesinin şu alanlarının boş bırakılmaması veya null olmaması gerekir:

|  |  |
| --- | --- |
| additional\_guarantees\_explanation | Sürprim Teminat Tablosu Açıklaması |
| additional\_guarantees\_name\_title | Sürprim Teminat Tablosu (1. Başlık): Teminat İsmi |
| additional\_guarantees\_before\_title | Sürprim Teminat Tablosu (2. Başlık): Uygulama Öncesi |
| additional\_guarantees\_percent\_title | Sürprim Teminat Tablosu (3. Başlık): Sürprim Oranı |
| additional\_guarantees\_after\_title | Sürprim Teminat Tablosu (4. Başlık): Uygulama Sonrası |
| additional\_guarantees\_table\_rows | Array şeklinde 1 veya daha fazla teminat bilgisi içerir (*HealthAdditionalGuaranteesTableRow*). |



1. **ApplicationFormConfirmation** objesi için dikkat edilmesi gereken alanlar:

|  |  |
| --- | --- |
| insurer\_sign\_image\_url | Sigorta ettiren kişinin imzasının bulunduğu bir imajın URL’sidir. Bu URL’ye webservisin kurulu olduğu sunucudan erişilebilmelidir. Bu imaj, PDF’in son sayfasına basılacaktır.  Null olabilir veya boş bırakılabilir. |
| insurer\_initials\_image\_url | Sigorta ettiren kişinin parafının bulunduğu bir imajın URL’sidir. Bu URL’ye webservisin kurulu olduğu sunucudan erişilebilmelidir. Bu imaj, tüm PDF sayfalarının alt kısmına basılacaktır. |
| company\_sign\_image\_url | Sigorta ettiren şirketin imzasının bulunduğu bir imajın URL’sidir. Bu URL’ye webservisin kurulu olduğu sunucudan erişilebilmelidir. Bu imaj, PDF’in son sayfasına basılacaktır.  Null olabilir veya boş bırakılabilir. |

insurer\_sign\_image\_url ve company\_sign\_image\_url boş/null olmadığı durumda PDF’in son sayfası şu şekilde basılır:



insurer\_sign\_image\_url ve insurer\_sign\_image\_url boş/null olduğu durumda PDF’in son sayfası şu şekilde basılır:



### Bilgilendirme Formu

Bilgilendirme Formu oluştururken gönderilen JSON objesinde özellikle dikkat edilmesi gereken alanlar:

|  |  |
| --- | --- |
| POST/PUT | /api/pdf/information-form/{personalIdNumber} |
| insurer\_sign\_image\_url | Sigorta ettiren kişinin imzasının bulunduğu bir imajın URL’sidir. Bu URL’ye webservisin kurulu olduğu sunucudan erişilebilmelidir. Bu imaj, PDF’in son sayfasına basılacaktır. |
| insurer\_initials\_image\_url | Sigorta ettiren kişinin parafının bulunduğu bir imajın URL’sidir. Bu URL’ye webservisin kurulu olduğu sunucudan erişilebilmelidir. Bu imaj, tüm PDF sayfalarının alt kısmına basılacaktır. |
| company\_sign\_image\_url | Sigorta ettiren şirketin imzasının bulunduğu bir imajın URL’sidir. Bu URL’ye webservisin kurulu olduğu sunucudan erişilebilmelidir. Bu imaj, PDF’in son sayfasına basılacaktır. |

### Genel ve Özel Şartlar

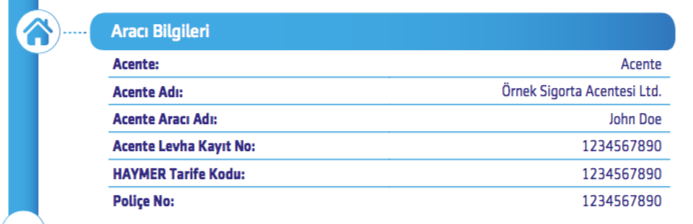
Genel ve Özel Şartlar PDF dosyasını oluştururken default değerlerin dışında özellikle dikkat edilmesi gereken alanlar bulunmamaktadır.

### Poliçe

Poliçe dosyası oluştururken JSON objesinde özellikle dikkat edilmesi gereken alanlar:

|  |  |
| --- | --- |
| POST/PUT | /api/pdf/policy/{personalIdNumber} |
| details | Array şeklinde 1. sayfaya basılan poliçe detaylarını içerir (*PolicyDetay*). |
| guarantees | Array şeklinde poliçe teminatlarını içerir (*PolicyGuarantee*). |
| payback\_years | Array şeklinde yıllık geri ödeme tarihlerini içerir (*PolicyPaybackYear*).  Sadece “Eğitim İçin Hayat Sigortası” için geçerlidir. |
| redemption\_fees | Array şeklinde “Matematik Karşılık ve İştira Bedeli” tablosunun satırlarını içerir (*PolicyRedemptionFee*).  Sadece “Ferdi Kazalı Prim İadeli Hayat Sigortası” ve “Prim İadeli Hayat Sigortası” için geçerlidir. |
| endorseds | Array şeklinde poliçe lehdarlarının bilgilerini içerir (*PolicyEndorsed*). |
| mediator\_details (1) | Array şeklinde poliçe aracı bilgileri tablosunun satırlarını içerir (*PolicyMediatorDetail*) (1). |
| is\_signed | Boolean olarak true gönderilirse, aşağıdaki imajlar için geçerli URL’ler gönderilmelidir. Aksi takdirde aşağıdaki imza resimleri PDF’e basılmaz. |
| company\_sign\_image\_url | Sigorta ettiren kişinin imzasının bulunduğu bir imajın URL’sidir. Bu URL’ye webservisin kurulu olduğu sunucudan erişilebilmelidir. Bu imaj, PDF’in son sayfasına basılacaktır. |
| aegon\_sign\_image\_url | Aegon şirket yetkilisinin imzasının bulunduğu bir imajın URL’sidir. Bu URL’ye webservisin kurulu olduğu sunucudan erişilebilmelidir. Bu imaj, PDF’in son sayfasına basılacaktır. |

1. **PolicyMediatorDetail** objeleri mediator\_details alanında array şeklinde gönderilir ve sonuç olarak şu şekilde PDF’e basılır:



### Poliçe Okuma Kılavuzu

Poliçe Okuma Kılavuzu PDF dosyasını oluştururken default değerlerin dışında özellikle dikkat edilmesi gereken alanlar bulunmamaktadır.

# Dikkat Edilmesi Gerekenler ve Alınabilecek Hatalar

### Teklif Özeti Pdf Oluşturma Hatası :

Teklif özeti pdf oluşturulurken, İnomera'ya ait olan offer\_summary\_pdf\_url kısmındaki linkin doğru olduğu kontrol edilmelidir. Bir örneği aşağıda yer almaktadır.

"offer\_summary\_pdf\_url":"[https://teklif.aegon.com.tr//Teklif/ROL5\_5/OdulluBirikim5Arti5Sonuc.aspx?  
TeklifID=ROL5\_5\_24102018\_88&Hash=774C440204F269D75C8EE4783963FA  
2226AD303E79CAED0CD6EA1C80245F64F4&Mode=4](https://teklif.aegon.com.tr//Teklif/ROL5_5/OdulluBirikim5Arti5Sonuc.aspx?TeklifID=ROL5_5_24102018_88&Hash=774C440204F269D75C8EE4783963FA2226AD303E79CAED0CD6EA1C80245F64F4&Mode=4)",

### 400 Bad Request Hatası :

Bu hata oluştuysa, oluşturulmak istenen pdf dosyası, daha önceden oluşturulmuş anlamına gelmektedir. Bu işlemin doğru bir şekilde çalışması için, pdf’i tekrardan oluşturmak yerine (POST), güncelleme(PUT) yapılması gerekmektedir.

- POST metoduyla yeni kayıt ve PDF oluşturulur

- PUT metoduyla var olan kayıt/PDF güncellenir

(Symfony\\Component\\HttpKernel\\Exception\\BadRequestHttpException(co  
> de: 0): A PDF file for the given personalIdNumber \"18053100110\",   
> polId \"2290676\", insuranceSlug \"prim-iadeli\" and pdfTypeSlug   
> \"06-police\" already exists. Please update it instead of creating a   
> new one. at

### 500 Internal Server Error Hatası :

Bu hata, POST yada PUT metoduyla, pdf servisine gönderilen parametrelerle ilgili bir hatadır. Dolayısıyla gönderilen parametreleri kontrol ediniz. Bu konuda detaylı bilgiye pdf servisi dökümanından ulaşabilirsiniz.

API Dokümantasyonu URL: <http://localhost:8888/doc>

### Cronjob log hataları

1. Aşağıdaki gibi , “No such file or directory in” hatası oluştuğunda, pdf dosyasının kaydının database de oluştuğunu, fakat pdf dosyasının web/download (diskte) klasöründe olmadığını göstermektedir. Dolayısıyla Moreum arşivleme sistemine pdf dosyasının aktarılamadığını göstermektedir. Çözüm için, database den olmayan pdf lerin kaydı silinebilir yada aegon\_pdf tablosundaki is\_archival\_required alanı 0 yapılabilinir.

Warning: file\_get\_contents(/var/www/symfony/web/download/2324612/85/aegon\_2289448\_turuncu-elma\_01-ihtc.pdf): failed to open stream: No such file or directory in /var/www/symfony/src/AppBundle/Command/AppMoreumArchivalCommand.php on line 176

1. Aşağıdaki gibi “Undefined Index” hatası durumunda,

PHP Notice: Undefined index: Poliçe Okuma Kılavuzu in /var/www/symfony/src/AppBundle/Command/AppMoreumArchivalCommand.php on line 183

Moreum tarafında gönderilmek istenen arşivleme sistemi ile aşağıdaki kısımda yer alan eşleşmelerde hata olduğunu göstermektedir. Bu durumda Moreum arşivleme sisteminde bir değişikliğe gidilmiş olabilinir. Sağ tarafdaki parametreler Moreum tarafından kontrol edilmelidir.

'01-ihtiyac-teyit' : 'IHTIYAC\_TEYIT'  
'02-teklif-ozet': 'TEKLIF\_OZET'  
'03-basvuru-formu': 'HAYAT\_BASVURU'  
'04-bilgilendirme-formu': 'BILGILENDIRME'  
'05-ozel-ve-genel-sartlar': 'OZELGENELSARTLAR'  
'06-police': 'POLICE'  
'07-police-okuma-kilavuzu': 'POLICEOKUMAKLAVUZU'

1. Karşılaşılabilecek bir diğer problem, Arşivleme sistemi için Moreum bağlantı parametreleridir. Herhangi bir hata durumunda filesystem deki app/config/parameters.yml da yer alan aşağıdaki bilgiler kontrol edilmelidir.

#Moreum Test Sunucusu  
#moreum\_archival\_wsdl: 'http://moreum-test.aegon.com.tr/TerraDocsWS/Archival.svc?wsdl'  
#moreum\_retrieval\_wsdl: 'http://moreum-test.aegon.com.tr/TerraDocsWS/Retrieval.svc?wsdl'  
  
# Moreum Production Sunucusu  
moreum\_archival\_wsdl: 'http://10.184.33.34/TerraDocsWS/Archival.svc?wsdl'  
moreum\_retrieval\_wsdl: 'http://10.184.33.34/TerraDocsWS/Retrieval.svc?wsdl'

## Loglama:

Log dosyaları 3 farklı şekilde loglanmaktadır.

1. **Database**: Database de, bugüne kadar Moreum arşivleme sistemine aktarılmış, aktarılmamış bütün pdf lerin log ları **aegon\_pdf\_log** tablosunda yer almaktadır.
2. **Filesistem**: var/log klasöründe yer almaktadır. Oluşturulmak istenen her pdf dosyası için başarılı olduğunda info mesajı, başarısız olduğunda ise başarısızlıkla ilgili error mesajı, log dosyasına işlenmektedir.
3. **Cronjob** : Cronjob çalıştığında Moreum arşivleme sistemine başarılı ve başarısız olarak gönderilen pdf sayısı, ayrıca pdf gönderilemediği takdirde hata mesajı, 4.1.4 de ifade edildiği şekilde loglarda işlenmektedir.

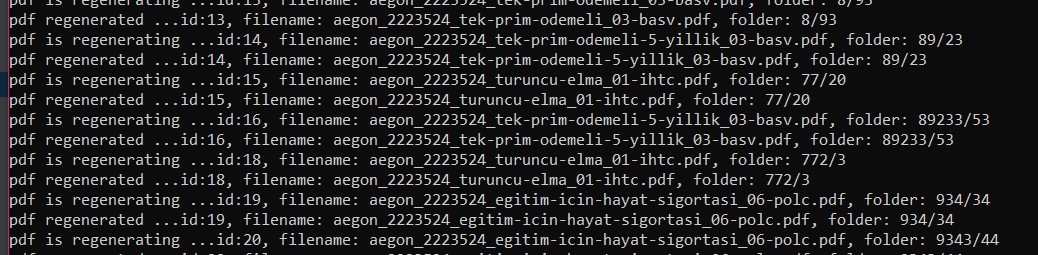
### Database Loglarından Tekrardan Pdf Üretme

Database logları aegon\_pdf\_log tablosunda tutulmaktadır. Bu tabloda servise gelen bütün request ler kayıt edilmektedir. Bu log kayıtlarından tekrardan pdf üretebilmek için, belirli aralıklarla bir komutun çalıştırılması gerekmektedir. Bunun için servisin çalıştırıldığı makinede (“docker host”) bir cronjob’un ayarlanabilinir. Bu cronjob örn. 12 veya 24 saatte bir çalışabilir.

Bu şekilde ayarlanmış olan cronjob, şu komutları çalıştırmalıdır:

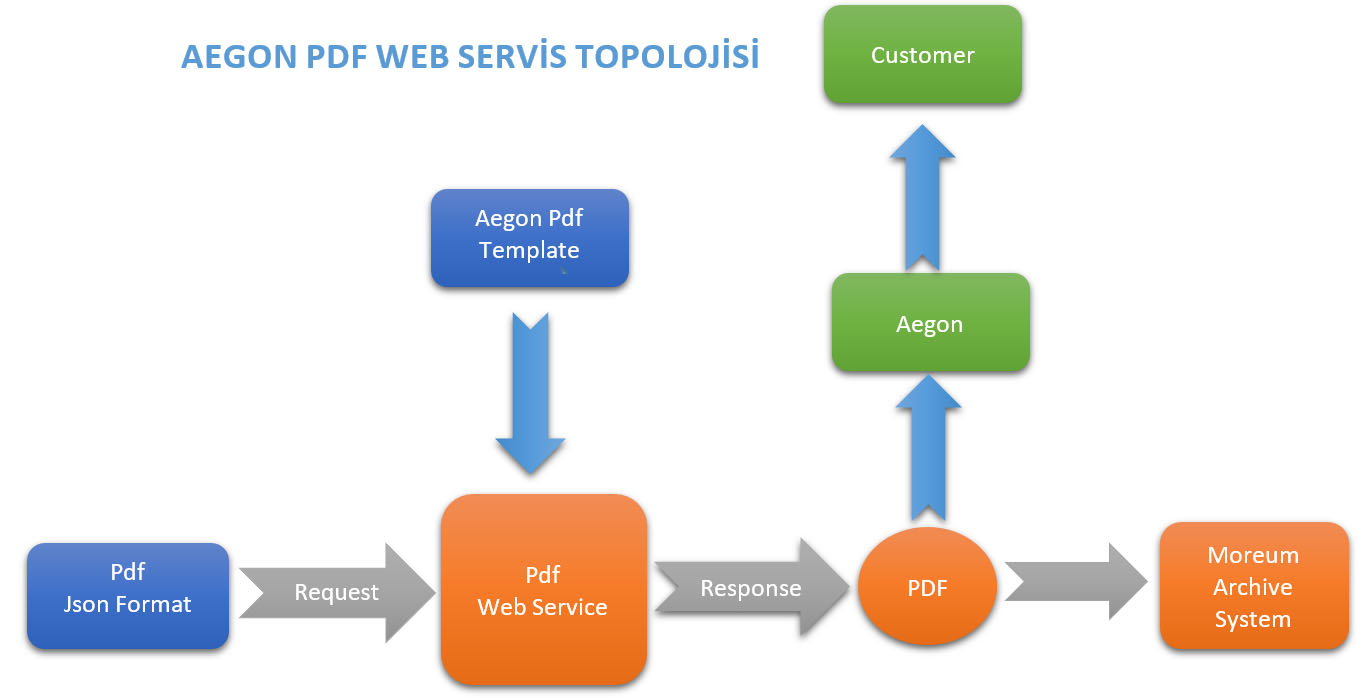
cd /var/www/pdf-service-1.0.0/  
docker-compose exec php bin/console app:pdf-regenerate –env=prod

Konsol ekranından aşağıdaki şekilde oluşturulan pdfler izlenebilinir.



Hata durumunda üretilemeyen pdf ekrana yansıyacaktır, bu durumda databaseden id numarasına göre kaydın parametreleri kontrol edilmelidir.

# Web Servis Topoloji ve Mimarisi



PDF Web Servisi, RESTful API olarak tasarlanmıştır. API dokümantasyonu Swagger formatında olup, altyapı olarak;

* PHP7
* SYMFONY FREAMWORK
* MYSQL
* DOCKER
  + NGNIX

kullanlmıştır.

Servis, gelen json formatındaki requestleri , aegon’a ait olan pdf template üzerindeki karşılıklarına yerleştirerek yeni bir pdf oluşturmaktadır. Ayrıca oluşturulan bu pdf ler, cronjob ile belirli zaman aralığında, pdf in sigorta türüne göre Moreum arşivleme sistemine eklenmektedir.