

5. Payout

- Intro
- High Level Plan
- Diagrams
 - High level overview
 - Detailed view
- Components responsibilities
 - Payout orchestrator
 - Payout executor
 - Payout bank adapter
 - Balance (payout context)
 - Merchant context (payout context)
- Payout status
- Payout API
 - Current Version
 - Orchestrator
 - Execution
- Payloads for integration between Falcon-P and other parts of the system
 - Balance Snapshot
 - Payout Prepared
 - Confirmed Payout
- API for integration between Falcon-P and Merchant Context
 - Auto payout
 - API call

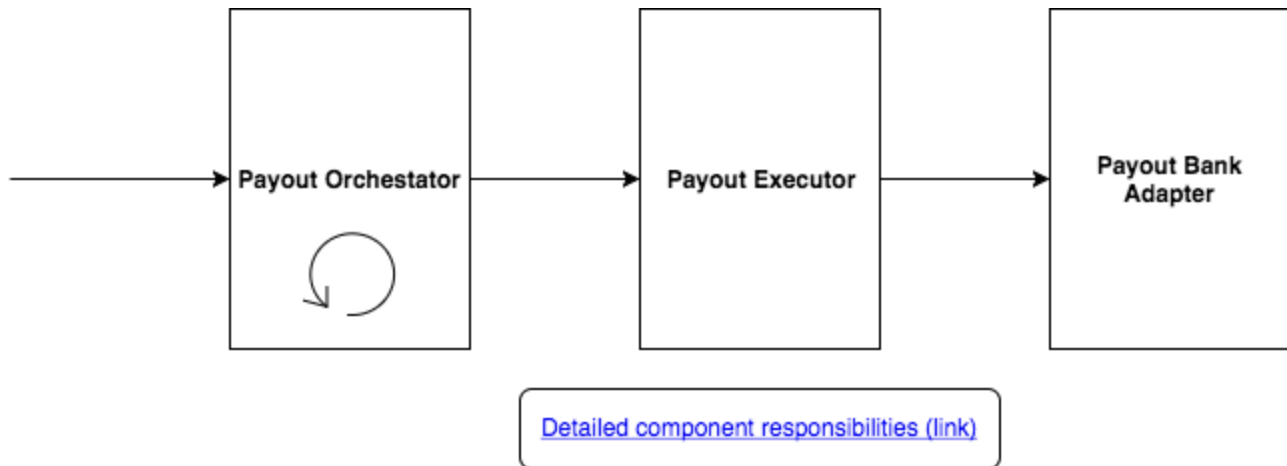
Intro

Being part of the Falcon product, Payout piece is responsible for withdrawal execution that can be triggered by a merchant, automatic scheduler or manually for individual payer. Its main responsibilities is to handle and validate payout request, verify it against any business rules that may apply, pack, send to bank and get the response from that bank (for more detailed description, see the below section on components responsibilities).

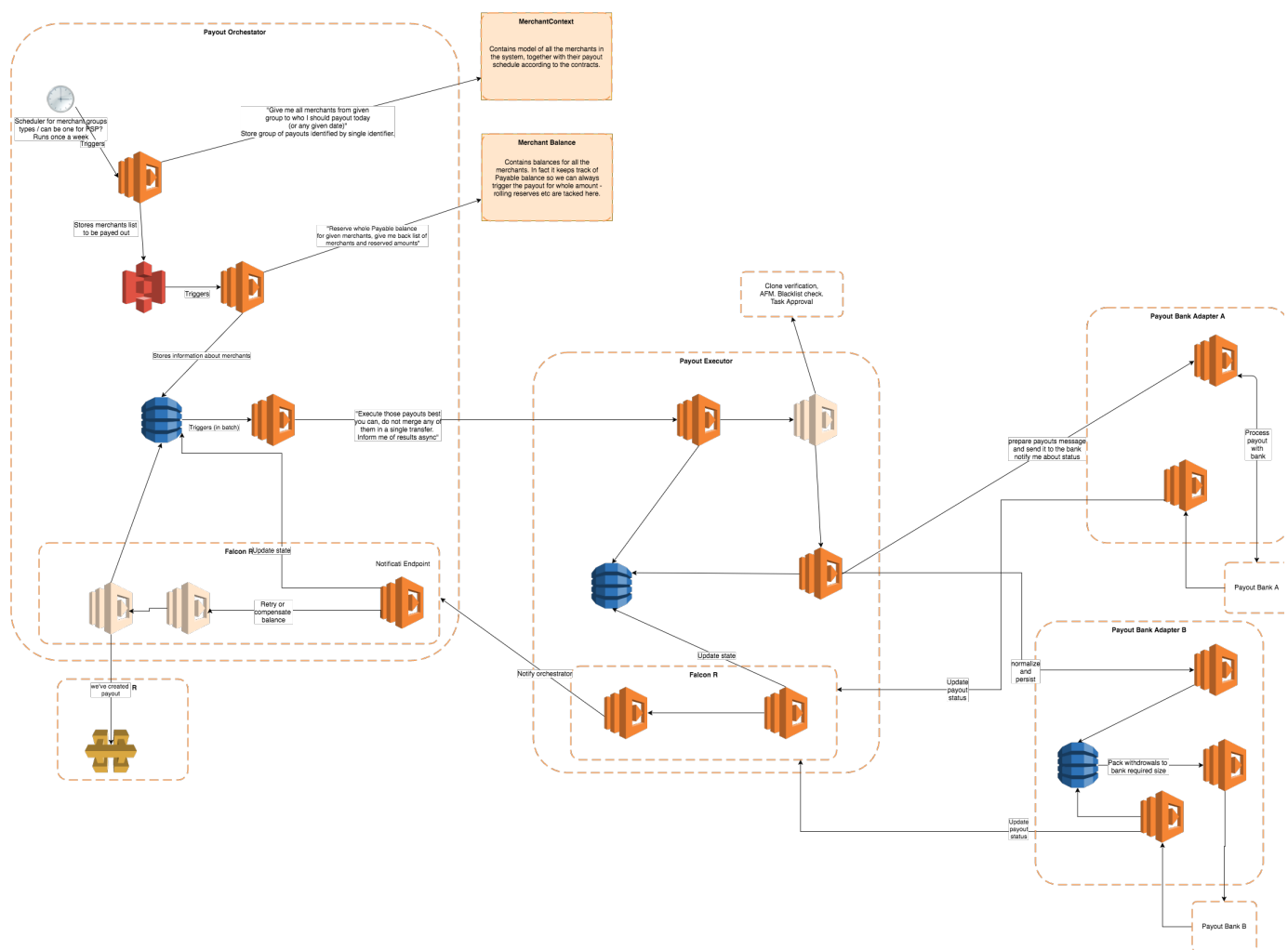
High Level Plan

Stage 1	Stage 2	Stage 3	Stage 4
simple excel as an execution outcome	enriching executor	payouts reconciliation	Handling payers
<ul style="list-style-type: none">• list/sheet of payouts:<ul style="list-style-type: none">• based on balance• with data driven from Merchant context• generates a file <p>DoR:</p> <ul style="list-style-type: none">• balance and merchant context in place• balances opening balances inputted based on Gecad• agreed flow on how to make and release reservation on balances	<ul style="list-style-type: none">• e2e flow, with bank interaction<ul style="list-style-type: none">• accepts requests• verifying business rules• checks against AFM/white-/blacklists• packaging• sending requests to bank• receiving call backs from banks and verifying payouts statuses against that information	<ul style="list-style-type: none">• reconciliation of withdrawal entries on bank statement against payout records from our side confirming the execution of payables for merchants	<ul style="list-style-type: none">• ability to handle and pay out payer's money and not only merchant's

High level overview



Detailed view



Components responsibilities

Payout orchestrator

1. Handles external calls (exposes APIs)
2. Init auto-payout process (cron jobs)
3. Gets information from Merchant Context about data needed to make a payout
4. Gets information from Balances about balance that we need payout
5. Initiates Payable funds blockade in Balances
6. Validates incoming Payout Requests
7. Provides views/actions to Payout Admin
8. Notifies Falcon Reconciliation about successful payouts
9. Notifies Balances about failure in executing Payout so that it can lift the Payable funds blockade
10. Schedules delayed Payouts*

Payout executor

1. Verifies clones (duplicates)
2. Verifies payout account blacklist/whitelist
3. Initiates and acts upon Task Approval process
4. Packages incoming Payout Requests according to business rules (for example by currency)
5. Decides which Acquirer Adapter to use for given Payout
6. Sends Payouts to appropriate Acquirer Adapters

Payout bank adapter

1. Prepares bank specific payout
2. Sends payouts to bank in appropriate batches
3. Notifies executor about payout status from the bank

Balance (payout context)

1. Holds the Payable balance for all Merchants/Submerchants (the smallest unit of payout)
2. Takes care of all fees and rolling reserve - Payouts just payout requested or full amount
3. Exposes batch API for blocking Payable funds for set of Merchants for Payout
4. Lifts/finalizes Payable funds blockade based on notification from Payout Orchestrator about success/failure of payout

Merchant context (payout context)

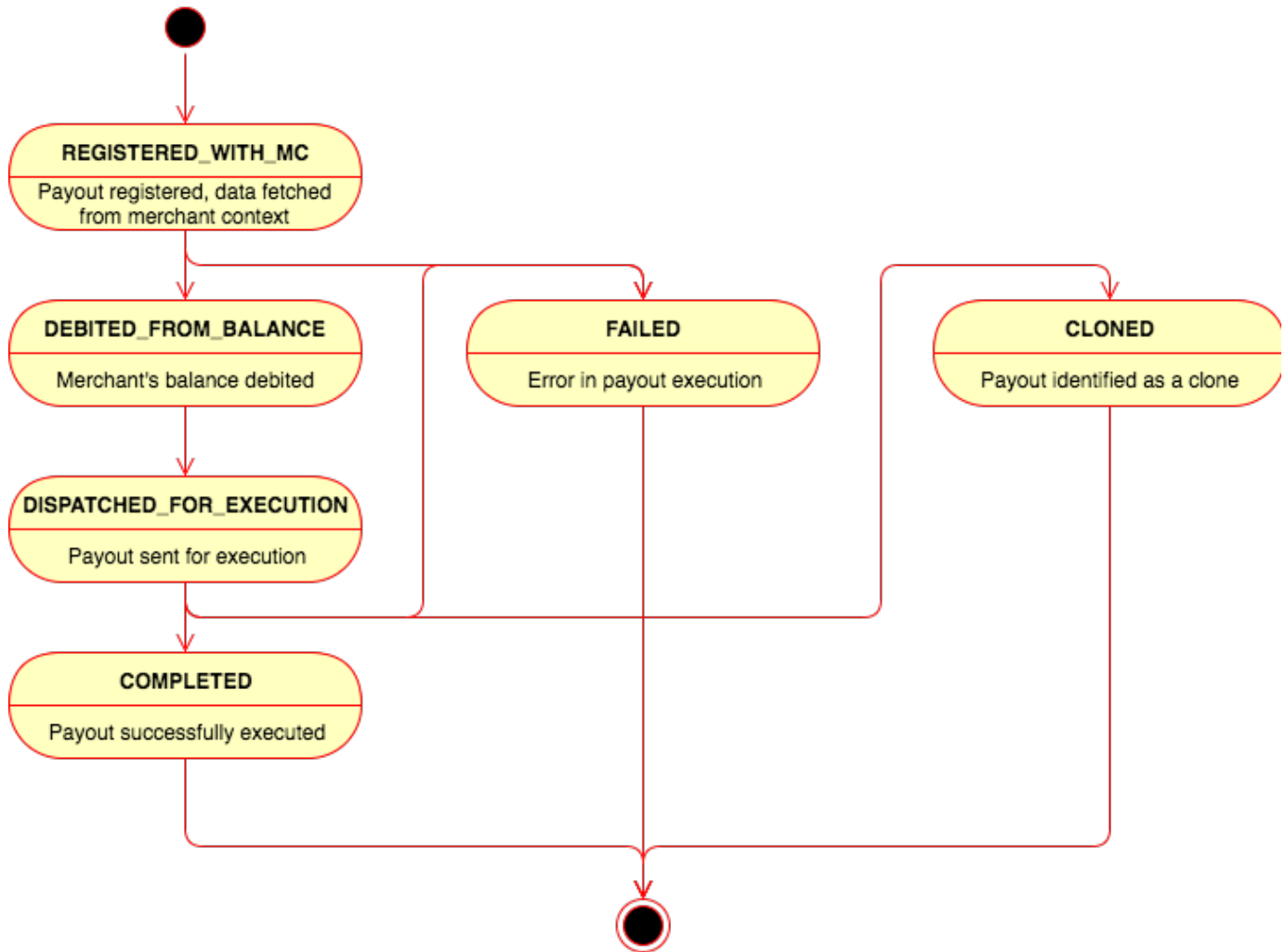
1. Holds the model of autopayout contracts to Merchants (the schedule)
2. Hold all Merchant data needed to execute a payout (names, bank accounts, address etc)
3. Holds settlementGroups information
4. Exposes API that answers the question "For given date and settlementGroup what are the Merchants should I autopayout to"

Payout status

Possible payout statuses:

- REGISTERED_WITH_MC - initial status - payout has been registered and data has been fetched from merchant context
- DEBITED_FROM_BALANCE - merchant's balance has been debited
- DISPATCHED_FOR_EXECUTION - payout has been sent for execution
- COMPLETED - final status - payout has been successfully executed
- FAILED - final status - there has been an error in payout execution
- CLONED - final status - payout has been identified as a clone

The diagram below represents possible transitions between statuses:



Payout API

Current Version

Current version of API is stored in repository: <https://payu-sicarios.github.io/falcon/api/blob/master/payouts.yml>

Orchestrator

PayoutCreateRequest

```
{
  "requestId": "ad63a696-c9a4-4001-94f5-27a9f83ecf0b",
  "reference": {
    "externalId": "GECAD-12345"
  },
  "settlementGroup": "GECAD_TR",
  "balanceCurrency": "EUR",
  "withdrawal": {
    "description": "Payout from PayU RO to eMag",
    "amount": {
      "value": 10,
      "currency": "EUR"
    },
    "type": "TRANSFER",
    "recipient": {
      "id": "1212121-dadasdsad-31231321",
      "name": "eMag",
      "address": {
        "street": "Grunwaldzka 182",
        "city": "Poznan",
        "countryCode": "PL"
      },
      "account": {
        "number": "RO49AAAA1B31007593840000",
        "bankSwiftCode": "XXXXXXX",
        "bankCountryCode": "PL"
      }
    },
    "additionalProperties": {
      "hint": "VAL"
    }
  }
}
```

Field	Required (Mandatory/Optional)	Description
"requestId": "ad63a696-c9a4-4001-94f5-27a9f83ecf0b",	O	UUID for request
"reference": {	M	This section contains references
"externalId": "GECAD-12345" }	M	This is a withdrawal id from client.
"settlementGroup": "GECAD_TR",	O	Identifies settlement group of payout (can be one or a few per client/PSP)

"balanceCurrency": "EUR",	O	Currency of balance that payout should be made out of. Not needed if only one balance exists for a given shop, required if there are more than once balance.
"withdrawal": {	O (when we want to pay all with default config)	Withdrawal data
"description": "Payout from PayU RO to eMag",	O	Withdrawal description
"amount": { "value": "10", "currency": "EUR" },	O	Withdrawal amount
"type": TRANSFER POSTAL_ORDER	O (with default)	Type of withdrawal (wire transfer, postal order)
"recipient": {	O	Withdrawal recipient data (only for direct withdrawals for custom accounts)
"id": 1212121-dadasdsad-31231321,	O	Identifies recipient (for flows with sensitive data)
"name": "eMag", "address": { "street": "Grunwaldzka 182", "city": "Poznan", "countryCode": "PL" },	M (when recipient specified)	Recipient identification
"account": {	O	Target account details
"number": "R049AAAA1B31007593840000",	M	Account number
"bankSwiftCode": "XXXXXXX",	M	Swift code
"bankCountryCode": "PL"	M	Bank country
} } "additionalProperties": { "hint": "VAL"	O	Additional parameters.

PayoutCreateResponse

```
{
  "requestId": "ad63a696-c9a4-4001-94f5-27a9f83ecf0b",
  "registeredWithdrawalReference": {
    "id": "e7298396-1ddd-49d7-9c60-4156439b40cf",
    "externalId": "GECAD-12345"
  },
  "result": "OK",
  "errorCode": 0,
  "message": "Payout created."
}
```

Field	Required (Mandatory/Optional)	Description
"requestId": "ad63a696-c9a4-4001-94f5-27a9f83ecf0b",	M	UUID for request
"registeredWithdrawalReference": {	M	This section contains references
"id": "e7298396-1ddd-49d7-9c60-4156439b40cf",	M	Internal ID of withdrawal
"externalId": "GECAD-12345" }	M	This is a withdrawal id from client.
"result": "OK",	M	Result state
"errorCode": 0,	M	Error code
"message": "Payout created."	O	This section contains merchant data.

Execution

PayoutExecuteRequest

```
{
  "requestId": "ad63a696-c9a4-4001-94f5-27a9f83ecf0b",
  "reference": {
    "externalId": "PAYOUT_ORCHIESTER-12345"
  },
  "settlementGroup": "GECAD_TR",
  "withdrawal": {
    "description": "Payout from PayU RO to eMag",
    "amount": {
      "value": 10,
      "currency": "EUR"
    },
    "type": "TRANSFER",
    "recipient": {
      "id": "1212121-dadasdsad-31231321",
      "name": "eMag",
      "address": {
        "street": "Grunwaldzka 182",
        "city": "Poznan",
        "countryCode": "PL"
      },
      "account": {
        "number": "RO49AAAA1B31007593840000",
        "bankSwiftCode": "XXXXXXX",
        "bankCountryCode": "PL"
      }
    },
    "additionalProperties": {
      "hint": "VAL"
    }
  }
}
```

Field	Required (Mandatory/Optional)	Description
"requestId": "ad63a696-c9a4-4001-94f5-27a9f83ecf0b",	M	UUID for request
"reference": {	M	This section contains references
"externalId": "GECAD-12345" }	M	This is a withdrawal id from Orchester.
"settlementGroup": "GECAD_TR",	M	Identifies settlement group of payout (can be one or a few per client/PSP)
"withdrawal": {	M	Withdrawal data
"description": "Payout from PayU RO to eMag",	M	Withdrawal description

"amount": { "value": "10", "currency": "EUR" },	M	Withdrawal amount
"type": TRANSFER POSTAL_ORDER,	M	Type of withdrawal (wire transfer, postal order)
"recipient": {	M	Withdrawal recipient data (only for direct withdrawals for custom accounts)
"id": 1212121-dadasdsad-31231321,	O	Identifies recipient (for flows with sensitive data)
"name": "eMag", "address": { "street": "Grunwaldzka 182", "city": "Poznan", "countryCode": "PL" },	M	Recipient identification
"account": {	M	Target account details
"number": "R049AAAA1B31007593840000",	M	Account number
"bankSwiftCode": "XXXXXXX",	M	Swift code
"bankCountryCode": "PL"	M	Bank country
} }, "additionalProperties": { "hint": "VAL"	O	Additional parameters.

PayoutExecuteResponse

```
{
  "requestId": "ad63a696-c9a4-4001-94f5-27a9f83ecf0b",
  "registeredWithdrawalReference": {
    "id": "e7298396-1ddd-49d7-9c60-4156439b40cf",
    "externalId": "PAYOUT_ORCHIESTER-12345"
  },
  "result": "OK",
  "errorCode": 0,
  "message": "Payout execution started."
}
```

Field	Required (Mandatory/Optional)	Description
"requestId": "ad63a696-c9a4-4001-94f5-27a9f83ecf0b",	M	UUID for request
"registeredWithdrawalReference": {	M	This section contains references
"id": "e7298396-1ddd-49d7-9c60-4156439b40cf",	M	Internal ID of withdrawal
"externalId": "PAYOUT_ORCHIESTER-12345" }	M	This is a withdrawal id from client.
"result": "OK",	M	Result state
"errorCode": 0,	M	Error code
"message": "Payout execution started."	O	This section contains merchant data.

Payloads for integration between Falcon-P and other parts of the system

Balance Snapshot

The message sent from Falcon-B to Falcon-P about snapshot of a balance (so that the payout can be made from this snapshot) will contain the following information:

BalanceSnapshotPayload

```
[
  {
    "balanceSnapshotId": "2cba9931-46eb-4ed8-961b-0628f0dfb72f",
    "amount": 14848.53,
    "currency": "TRY",
    "shopId": "auto111sfgh",
    "messageTimestamp": 1512547727
  }
]
```

Field	(Required/Optional)	Description
"balanceSnapshotId": "2cba9931-46eb-4ed8-961b-0628f0dfb72f",	R	identifier of balance snapshot
"amount": 14848.53,	R	amount of a balance
"currency": "TRY",	R	currency of a balance
"shopId": "auto111sfgh",	R	identifier of the shop (as of now: identifier currently seen all over Falcon)
"messageTimestamp": 1512547727	R	time when the snapshot has been prepared timestamp will be changed to date in ISO-8601 format v1host.com/PayU/story.mvc/Summary?oidToken=Stor

Payout Prepared

The message sent from Falcon-P to Falcon-B about prepared payout (so that the funds to make a payout can be blocked on the balance) will contain the following information:

PreparedPayoutPayload

```
[
  {
    "balanceSnapshotId": "2cba9931-46eb-4ed8-961b-0628f0dfb72f",
    "amount": "14848.53",
    "currency": "TRY",
    "shopId": "auto111sfgh",
    "operationTimestamp": 1512547734,
    "id": "ab98b42c-7872-4b23-8d83-a6cb3e51a83b"
  }
]
```

Field	(Required /Optional)	Description
"balanceSnapshotId": "2cba9931-46eb-4ed8-961b-0628f0dfb72f",	R	identifier of balance snapshot

"amount": "14848.53",	R	amount of a payout
"currency": "TRY",	R	currency of a payout
"shopId": "auto111sfgh",	R	identifier of the shop (as of now: identifier currently sent by GeC all over Falcon)
"operationTimestamp": 1512547734,	R	time when the snapshot has been used timestamp will be changed to date in ISO-8601 format under http://v1.host.com/PayU/story.mvc/Summary?oidToken=Story:21596
"id": "ad63a696-c9a4-4001-94f5-27a9f83ecf0b"	R	identifier of a payout

Confirmed Payout

The message sent from Falcon-P to Falcon-R about confirmed payout (so that the payout can be later on reconciled against statement) will contain the following information:

ConfirmedPayoutPayload
<pre>[{ "id": "ab98b42c-7872-4b23-8d83-a6cb3e51a83b", "operationTimestamp": 1512547736, "amount": "14848.53", "currency": "TRY", "reference": "PAYOUT-FROM-API-111111", "shopId": "auto111sfgh", "bankId": "GARANTI" }]</pre>

Field	(Required /Optional)	Description
"id": "ad63a696-c9a4-4001-94f5-27a9f83ecf0b",	R	identifier of a payout
"operationTimestamp": 1512547736,	R	time we got confirmation from the bank we successfully submitted for now for Garanti: time when we put the payout to a file timestamp will be changed to date in ISO-8601 format under http://v1.host.com/PayU/story.mvc/Summary?oidToken=Story:21596
"amount": "14848.53",	R	amount of a payout
"currency": "TRY",	R	currency of a payout
"reference": "PAYOUT-FROM-API-111111",	R	reference that shall be used for matching with statement currently sent only for payouts requested through API (not sent
"shopId": "auto111sfgh",	R	identifier of the shop (as of now: identifier currently sent by GeC all over Falcon)

"bankId": "GARANTI"	R	identifier of the bank which was used to send a payout
---------------------	---	--

API for integration between Falcon-P and Merchant Context

Auto payout

Cron job that is executed once a day. In this situation we require a list of merchants that needs to be paid out that particular date.

Request

```
https://apbw36rg1h.execute-api.eu-central-1.amazonaws.com/stage/shops-to-payouts?provider=GECAD_TURKEY&payout_date=2017-12-22&limit=90&after=eyJpbmRlcm5hbFNob3BJZCI6eyJzIjoimTIzNmY0MDAtMzYyMS00YTIxLWE3ZGMtNmU3NTAwYmYyNDhjIn0sImNlcnJlbnN5Ijp7InMiOiJUUlkifX0=
```

Field
"payout_date": "2017-12-22"
"provider": "GECAD_TURKEY"
"limit": 90
"after": eyJpbmRlcm5hbFNob3BJZCI6eyJzIjoimTIzNmY0MDAtMzYyMS00YTIxLWE3ZGMtNmU3NTAwYmYyNDhjIn0sImNlcnJlbnN5Ijp7InMiOiJ

Response

```
{
  "shop_payout_data_list": [
    {
      "id": "df26366d-9658-4419-98de-4f4927b98baf",
      "shop_name": "dreamersfigure.com - Hayalciler Yaynclık Reklam
Film Organizasyon Oyuncak Sanayi ve D Ticaret Lt",
      "street": "Caferaa Mah. Moda Cad. Zuhall Sok. Banu Apt. No:1/3
Kadköy",
      "city": "stanbul",
      "account_no": "TR020013400000502917500001",
      "bank_country_code": "Turkey",
      "bank_swift_code": "XXXXXXX",
      "tax_id": "4600423418",
      "payout_period": "WEEKLY",
      "external_id": "492",
      "currency": "TRY",
      "provider": "GECAD_TURKEY"
    }
  ],
  "page_info": {
    "page_size": 90,
    "after":
"eyJpbmRlcm5hbFNob3BJZCI6eyJzIjoimTIzNmY0MDAtMzYyMS00YTlxLWE3ZGMtNmU3NTAwY
mYyNDhjIn0sImNlcnJlbnN5Ijp7InMiOiJUUlkiX0="
  }
}
```

Field

"id": "df26366d-9658-4419-98de-4f4927b98baf"

"shop_name": "dreamersfigure.com - Hayalciler Yaynclık Reklam Film Organizasyon Oyuncak Sanayi ve D Ticaret

"street": "Caferaa Mah. Moda Cad. Zuhall Sok. Banu Apt. No:1/3 Kadköy"

"city": "stanbul"

"accountNo": "TR020013400000502917500001"

"bankSwiftCode": "XXXXXXX"

"bankCountryCode": "Turkey"

"tax_id": "4600423418"
"payout_period": "WEEKLY"
"external_id": "492"
"currency": "TRY"
"taxId": "1234567896",
"provider": "GECAD_TURKEY"
"page_size": 90
"after": "eyJpbmRlcm5hbFNob3BJZCI6eyJzIjoimTIzNmY0MDAtMzYyMS00YTlxLWE3ZGMtNmU3NTAwYmYyNDhjIn0sImNlcnJlbnN5Ijp7InMiOi

API call

Single payout initiated by a merchant call to API

Request

```
{  
  "shopId": "internal-1234",  
  "currency": "TRY"  
}
```

Field	(Required/Optional)	Description
"shopId": "internal-1234",	R	Shop identifier
"currency": "TRY",	R	Balance currency

Response

```
{
  "name": "eMag",
  "street": "Grunwaldzka 182",
  "city": "Poznan",
  "accountNo": "RO49AAAA1B31007593840000",
  "bankSwiftCode": "XXXXXXX",
  "bankCountryCode": "PL",
  "shopId": "internal-1234",
  "currency": "TRY",
  "taxId": "1234567896",
  "provider": "GECAD_TR"
}
```

Field	(Required/Optional)	Description
"name": "eMag",	R	Merchant name
"street": "Grunwaldzka 182",	R	Merchant street of registration
"city": "Poznan",	R	Merchant city of registration
"accountNo": "RO49AAAA1B31007593840000",	R	Merchant account number
"bankSwiftCode": "XXXXXXX",	R	Merchant bank SWIFT code
"bankCountryCode": "PL",	R	Merchant bank country code
"shopId": "internal-1234",	R	Shop identifier
"currency": "TRY",	R	Balance currency
"taxId": "1234567896",	R	Merchant tax identifier (fiscal code)
"provider": "GECAD_TR"	R	Provider name