

ECGM

Dokumentace

OBSAH

Úvod	4
1 Základní pojmy	5
2 Instalace	6
2.1 Požadavky	6
2.2 Instalace	6
2.3 Po instalaci	6
2.4 \ECGM\ECGM	7
3 Modely	9
3.1 \ECGM\Model\BaseArray	9
3.2 \ECGM\Model\AssociativeBaseArray	10
3.3 \ECGM\Model\Product	11
3.4 \ECGM\Model\ProductComplement	12
3.5 \ECGM\Model\CurrentProduct	13
3.6 \ECGM\Model\Parameter	13
3.7 \ECGM\Model\Customer	14
3.8 \ECGM\Model\CustomerGroup	15
3.9 \ECGM\Model\Order	16
3.10 \ECGM\Model\StrategyProduct	17
3.11 \ECGM\Model\OrderProduct	18
4 Controllery	19
4.1 \ECGM\Controller\ CustomerParametersMergeController	19
4.2 \ECGM\Controller\ CustomerParametersCleaningController	19
4.3 \ECGM\Controller\CustomerGroupingController	20
4.4 \ECGM\Controller\DealerStrategyController	21
4.5 \ECGM\Controller\CustomerStrategyController	22

4.6	\ECGM\Controller\StrategyController	22
4.7	\ECGM\Controller\PassiveStrategyTypeController	23
4.8	\ECGM\Controller\ConservativeStrategyTypeController	24
4.9	\ECGM\Controller\AggressiveStrategyTypeController	25
5	Pomocné třídy	27
5.1	\ECGM\Util\DistanceFunctions	27
5.2	\ECGM\Util\MathFunctions	27
5.3	\ECGM\Util\MiscFunctions	27
5.4	\ECGM\Util\KmeansPlusPlus	28
5.5	\ECGM\Util\SilhouetteAnalysis	29

ÚVOD

ECGM je nástroj, který má za úkol zpracovat strukturovaná statistická data a na základě nich doporučí produkty/doplňky k produktům, které by bylo nejvhodnější danému zákazníkovi v danou dobu zobrazit. Dále je možné specifikovat i automatické navrhování slev a to buď agresivním, nebo konzervativním způsobem, tak aby byl s co největší pravděpodobností prodán produkt, který je výhodný pro prodejce.

Knihovna a její kompletní zdrojové kódy jsou pod MIT licencí dostupné na github <https://github.com/okomarek/ECGM> , popřípadě na <https://packagist.org/packages/okomarek/ecgm>.

1 ZÁKLADNÍ POJMY

- PPC – PPC neboli Product Payoff Coefficient je numerická hodnota, která vyjadřuje jakou má produkt v daném stavu (sleva, výše skladových zásob, doba před expirací) pro prodejce hodnotu. Čím vyšší PPC je tím vyšší je hodnota tohoto produktu pro prodejce.
- Shluk – Shluk je skupina zákazníků, která byla vytvořena an základě shlukové analýzy. Shluky jsou určeny pouze pro interní použití v knihovně.
- Parametry zákazníka – Pro správnou funkci shlukování je nutné každému zákazníkovi přidat parametry. Ty reprezentují hodnoty, podle kterých dochází ke shlukování. Je nutné, aby měl každý zákazník stejný počet parametrů.
- `*@param type $parameter` – Požadovaný parametr.
- `@param type $parameter = value` – Volitelný parametr s výchozí hodnotou.
- `@return type` – Návrátová hodnota funkce, pokud není specifikována, je vráceno `void`.
- `@throws exception` – Specifikace výjimky, kterou může funkce vyvolat.
- Pasivní strategie (`StrategyType::PASSIVE`) – U pasivní strategie dojde pouze k navržení pořadí produktů v jakém bude pro prodejce nejvýhodnější je zákazníkovi prodat.
- Konzervativní strategie (`StrategyType::CONSERVATIVE`) – U konzervativní strategie bude při řazení zároveň docházet k navržení minimální slevy, tak aby došlo k prioritizování daného produktu zákazníkem.
- Agresivní strategie (`StrategyType::AGGRESIVE`) – U agresivní strategie bude při řazení zároveň docházet k navržení maximální slevy, tak aby došlo k prioritizování daného produktu zákazníkem, ale zároveň nedošlo k poklesu daného produktu ve strategii prodejce.

2 INSTALACE

2.1 Požadavky

- PHP verze 5.6 a vyšší
- phpunit verze 5 pro vývoj

2.2 Instalace

- Pomocí composer: require okomarek/ecgm

2.3 Po instalaci

Po instalaci je nutné implementovat rozhraní:

`\ECGM\MainInterface`

- **getCustomers** – Má vrátit všechny zákazníky, kteří budou zahrnuti do strategie. Musí vracet `BaseArray` s `\ECGM\Model\Customer` jako `requiredClass`.
 - @return `BaseArray`
- **getUngroupedCustomers** – Má vrátit všechny zákazníky, kteří budou zahrnuti do strategie a nemají přiřazený žádný shluk. Musí vracet `BaseArray` s `\ECGM\Model\Customer` jako `requiredClass`.
 - @return `BaseArray`
- **getCustomerGroups** – Má vrátit všechny již vytvořené shluky zákazníků. Musí vracet `BaseArray` s `\ECGM\Model\CustomerGroup` jako `requiredClass`.
 - @return `BaseArray`
- **getProducts** – Má vrátit všechny produkty, které jsou momentálně v prodeji a mají být zahrnuty do strategie. Musí vracet `AssociativeBaseArray` s `\ECGM\Model\CurrentProduct` jako `requiredClass`.
 - @return `AssociativeBaseArray`
- **setProductPPC** – Nastaví požadované PPC (Product Payoff Coefficient) podle stavu vkládaného produktu.
 - *@param `CurrentProduct $product`
 - @return `CurrentProduct`

Toto rozhraní je nutné implementovat na základě daného e-shopu. Poté již stačí volat třídu `\ECGM\ECGM`, která zapouzdřuje všechny funkce knihovny.

2.4 `\ECGM\ECGM`

Pokud by Vám nevyhovovala jakákoliv funkce, je pro většinu tříd umožněno `DependencyInjection`. Jediným požadavkem je, aby bylo implementováno příslušné rozhraní. V tomto případě je silně doporučeno dědit již existující třídy a pouze přetěžovat potřebné funkce.

- `___construct`
 - `*@param \ECGM\MainInterface $mainInterface`
 - `*@param int $strategyMultiplierCoefficient` – Hodnota této proměnné je využita pro zvýšení váhy nákupů zákazníka oproti nákupům jeho skupiny a ke zvýšení váhy doplňků. Čím vyšší, tím více budou tyto hodnoty .
 - `*@param int $dimension` – Určuje dimenzi parametrů zákazníka. Je nutné počítat s tím, že po očištění cirkulární hodnoty se její dimenze zdvojnásobí. Například pro hodnoty [Skupina(necirkulární), DevVRoce(cirkulární)] by `$dimension = 3`.
 - `*@param int $initialClusterNumber` – Tento parametr určuje minimální množství shluků, se kterými bude shluková analýza pracovat. Pokud bude `$autoClusterNumberAdjustment` nastaveno na `false`, nebude toto číslo dále měněno. Správný počáteční odhad počtu shluků bude mít výrazný vliv na časovou náročnost shlukování.
 - `@param bool $autoClusterNumberAdjustment = true` – Pokud je nastaven na `true`, bude počet shluků automaticky upraven na základě siluetové analýzy. Siluetová analýza je značně výpočetně náročná, proto je u velkého počtu zákazníků doporučeno tuto funkci vypnout.
 - `@param int $maxProductsInStrategy = 40` – Určuje maximální počet produktů, které budou zařazeny do výpočtu strategie. Tyto produkty jsou vybrány na základě zákaznickovi strategie. V základě bude do výpočtu strategie zařazeno maximálně 40 produktů, které si zákazník s největší pravděpodobností koupí. Zvláště u Konzervativní a Agresivní strategie bude vysoký počet produktů způsobovat zpomalení výpočtu.

- @throws \ECGM\Exceptions\InvalidArgumentException
- @throws \ECGM\Exceptions\UndefinedException
- **getStrategyController**
 - @return \ECGM\Int\StrategyInterface
- **setStrategyController**
 - *@param \ECGM\Int\StrategyInterface \$strategyController
- **getGroupingController**
 - @return \ECGM\Int\CustomerGroupingInterface
- **setGroupingController**
 - *@param \ECGM\Int\CustomerGroupingInterface \$groupingController
- **getParameterCleaningController**
 - @return \ECGM\Int\CustomerParametersCleaningInterface
- **setParameterCleaningController**
 - *@param \ECGM\Int\CustomerParametersCleaningInterface \$parameterCleaningController
- **groupCustomers**
 - @return \ECGM\Model\BaseArray
- **getStrategy**
 - *@param \ECGM\Model\Customer \$customer – Zákazník pro kterého má dojít k výpočtu strategie.
 - @param \ECGM\Model\Order|null \$currentOrder = null – Aktuální objednávka zákazníka, pokud má již nějaké produkty v košíku.
 - *@param \ECGM\Enum\StrategyType \$strategyType = \ECGM\Enum\StrategyType::CONSERVATIVE
 - @return \ECGM\Model\AssociativeBaseArray – Produkty jsou vráceny seřazeně od nejlepšího po nejhorší, společně s navrženými slevami.

3 MODELÝ

3.1 \ECGM\Model\BaseArray

Implementuje \Iterator, \Countable

Jedná je o třídu obalující jednoduché PHP pole (array).

- **__construct**
 - @param BaseArray|null \$baseArray = null – Pokud je hodnota dojde k počátečnímu naplnění pole obsahem tohoto BaseArray.
 - @param string|null \$requiredBaseClass = null – Pokud je hodnota naplněna neumožní vložení jiné hodnoty, než té, která je nebo má jako předka danou třídu.
 - @throws \ECGM\Exceptions\InvalidArgumentException
- **requiredBaseClass**
 - @return string – Vrací aktuální hodnotu požadované třídy.
- **add** – Vloží hodnotu nakonec pole.
 - *@param mixed \$obj
 - @throws \ECGM\Exceptions\InvalidArgumentException
- **setList** – Přepíše všechny aktuální hodnoty v pole obsahem proměnné \$list.
 - *@param \$list
 - @throws \ECGM\Exceptions\InvalidArgumentException
- **mergeList** – Obsah proměnné \$listy přidá na konec aktuálního pole.
 - *@param \$list
 - @throws \ECGM\Exceptions\InvalidArgumentException
- **mergeList** – Přepíše všechny aktuální hodnoty obsahem pole v \$baseArray.
 - *@param BaseArray \$baseArray
 - @throws \ECGM\Exceptions\InvalidArgumentException
- **mergeList** – Obsah pole v \$baseArray přidá na konec aktuálního pole.
 - *@param BaseArray \$baseArray
 - @throws \ECGM\Exceptions\InvalidArgumentException
- **clear** – Smaže všechny aktuální hodnoty.
- **size** – Vrací aktuální velikost pole.

- @return int
- **removeAll** – Smaže všechny hodnoty, které se nachází v průniku aktuálního pole a pole které obsahuje parametr \$baseArray.
 - *@param BaseArray \$baseArray
 - @throws @throws \ECGM\Exceptions\InvalidArgumentException
- **remove** – Smaže objekt, na pozici parametru \$key.
 - *@param int \$key
- **removeByObject** – Smaže objekt, který je roven parametru \$obj, pokud se v poli nachází.
 - *@param mixed \$obj
- **isEmpty**
 - @return bool
- **getObj** – Vráti objekt, které se nachází na pozici dané hodnotou \$key. Vrací null, pokud daná pozice neexistuje, nebo je parametr \$key nesprávně zadáný.
 - *@param int \$key
 - @return mixed|null

3.2 \ECGM\Model\AssociativeBaseArray

Rozšiřuje \ECGM\Model\BaseArray

Jedná se o rozšíření třídy \ECGM\Model\AssociativeBaseArray, které místo indexového pole využívá pole asociativní. Všechny vkládané hodnoty musí implementovat \ECGM\Int\KeyableValue. Dále budou zmíněny pouze přetížené funkce.

- **__construct**
 - @param AssociativeBaseArray|null \$baseArray = null – Pokud je hodnota dojde k počátečnímu naplnění pole obsahem tohoto BaseArray.
 - @param string|null \$requiredBaseClass = null – Pokud je hodnota naplněna neumožní vložení jiné hodnoty, než té, která je nebo má jako předka danou třídu.
 - @throws InvalidArgumentException
- **requiredBaseClass**
 - @return string – Vrací aktuální hodnotu požadované třídy.
- **add** – Vloží hodnotu na pozici určenou jejím klíčem.

- `*@param \ECGM\Int\KeyableValue $obj`
- `@throws \ECGM\Exceptions\InvalidArgumentException`
- **setList** – Přepíše všechny aktuální hodnoty v pole obsahem proměnné `$list`.
 - `*@param \ECGM\Int\KeyableValue[] $list`
 - `@throws \ECGM\Exceptions\InvalidArgumentException`
- **mergeList** – Obsah proměnné `$listy` vloží do aktuálního pole. Pokud by hodnota v poli a hodnota v proměnné `$list` sdílely klíč, přepisuje hodnota v `$list` aktuální hodnotu.
 - `*@param \ECGM\Int\KeyableValue[] $list`
 - `@throws \ECGM\Exceptions\InvalidArgumentException`
- **remove** – Smaže objekt, na pozici parametru `$key`.
 - `*@param mixed $key`
- **removeByObject** – Smaže objekt, který je roven parametru `$obj`, pokud se v poli nachází.
 - `*@param \ECGM\Int\KeyableValue $obj`
- **getObj** – Vrátil objekt, které se nachází na pozici dané hodnotou `$key`. Vrací null, pokud daná pozice neexistuje.
 - `*@param mixed $key`
 - `@return mixed|null`

3.3 \ECGM\Model\Product

Implementuje `\ECGM\Int\KeyableValue`

- **__construct**
 - `*@param mixed $id` – Unikátní identifikátor produktu.
 - `*@param float $price` – Cena produktu.
 - `*@param int $expiration` – Určuje dobu, za jakou by po nákupu daného produktu, mohl tento produkt znovu chtít koupit (například u spotřebního zboží bude expirace zpravidla v řádech desítek dní).
 - `@param \ECGM\Enum\DateType $expirationDateType = \ECGM\Enum\DateType::DAYS`
 - `@param float = 0.0 $discount` – Aktuální sleva aplikovaná na produkt.
 - `@throws \ECGM\Exceptions\InvalidArgumentException`

- @throws \ECGM\Exceptions\ReflectionException
- **getComplements**
 - @return \ECGM\Model\BaseArray
- **setComplements** – Produkty, které jsou doplňkem tohoto produktu (například židle ke stolu apod.).
 - *@param \ECGM\Model\BaseArray \$complements
 - @throws \ECGM\Exceptions\InvalidArgumentException \$complements
- **getExpirationDateType**
 - @return \ECGM\Enum\DateType
- **setExpirationDateType**
 - *@param \ECGM\Enum\DateType \$expirationDateType
- **getId**
 - @return mixed
- **getPrice**
 - @return float
- **setPrice**
 - *@param float \$price
- **getDiscountedPrice**
 - @return float|int
- **getDiscount**
 - @return float
- **setDiscount**
 - *@param float \$discount
- **getExpiration**
 - @return int
- **setExpiration**
 - *@param int \$expiration

3.4 \ECGM\Model\ProductComplement

Určuje doplněk produktu. Tedy produkt, který nějakým způsobem s daným produktem souvisí a zákazník by ho mohl chtít koupit po nákupu produktu, který tento doplněk má. Například židle ke stolu.

- **___construct**
 - `*@param \ECGM\Model\Product`

getId – Vrací id produktu.

- `@return mixed`

getPrice

- `@return float`

3.5 \ECGM\Model\CurrentProduct

Rozšiřuje \ECGM\Model\Product

- **___construct**
 - `*@param $id`
 - `*@param float $price`
 - `*@param integer $expiration`
 - `*@param mixed $ppc` – Aktuální Product Payoff Coefficient produktu.
 - `@param float $discount=0.0`
 - `@throws \ECGM\Exceptions\InvalidArgumentException`
 - `@throws \ReflectionException`

3.6 \ECGM\Model\Parameter

Jeden zákazníkův parametr.

- **___construct**
 - `*@param mixed $id`
 - `*@param float $value`
 - `@param boolean $isCircular=false` – Specifikuje, že je daný parametr kruhový. Příklad může být hodina ve dni (1-24), nebo měsíc v roce (1-12).
 - `@param float $maxValue=0` – Pokud je \$isCircular nastaveno na true, je nutné zadat maximální velikost, jakou může daný parametr nabývat. U hodiny ve dni by se jednalo o hodnotu 24.
- **getId**
 - `@return mixed`
- **isCircular**

- @return bool
- **getMaxValue**
 - @return float
- **getCustomer**
 - @return \ECGM\Model\Customer
- **setCustomer**
 - *@param \ECGM\Model\Customer \$customer
- **getValue**
 - @return mixed

3.7 \ECGM\Model\Customer

- **___construct**
 - *@param mixed \$id
 - @param CustomerGroup|null \$group = null
 - @throws \ECGM\Exceptions\InvalidArgumentException
- **addParameter** – Je nutné aby parametry byly u všech zákazníků ve stejném pořadí.
 - *@param \ECGM\Model\Parameter \$parameter
 - @throws \ECGM\Exceptions\InvalidArgumentException
- **removeParameter**
 - *@param int \$parameterId
- **removeParameter** – Vrací historii nákupů (\ECGM\Model\Order) daného zákazníka.
 - @return \ECGM\Model\BaseArray
- **setHistory**
 - *@param int \ECGM\Model\BaseArray \$history
 - @throws \ECGM\Exceptions\InvalidArgumentException
- **addOrder** – Vloží jeden nákup do historie zákazníka.
 - *@param int \ECGM\Model\Order \$order
 - @throws \ECGM\Exceptions\InvalidArgumentException
- **removeOrder**
 - *@param int \$orderId

- **getParameters**
 - @return \ECGM\Model\BaseArray
- **getParameters**
 - *@param \ECGM\Model\BaseArray \$parameters
 - @throws \ECGM\Exceptions\InvalidArgumentException
- **getGroup**
 - @return \ECGM\Model\CustomerGroup
- **setGroup**
 - *@param \ECGM\Model\CustomerGroup \$group
- **getId**
 - @return mixed
- **getParametersAsSimpleArray** – Vrátí hodnoty parametrů zákazníka jako jednoduché indexované pole.
 - @return array

3.8 \ECGM\Model\CustomerGroup

- **__construct**
 - *@param mixed \$id
 - @param \ECGM\Model\BaseArray \$parameters = null
 - @throws \ECGM\Exceptions\InvalidArgumentException
- **mergeCustomers**
 - *@param \ECGM\Model\BaseArray \$customers
 - @throws \ECGM\Exceptions\InvalidArgumentException
- **mergeCustomers**
 - *@param \ECGM\Model\Customer \$customer
 - @throws \ECGM\Exceptions\InvalidArgumentException
- **removeCustomer**
 - *@param \ECGM\Model\Customer \$customer
- **removeCustomers**
 - *@param \ECGM\Model\BaseArray \$customers
- **addParameter**
 - *@param \ECGM\Model\Parameter \$parameter

- @throws \ECGM\Exceptions\InvalidArgumentException
- **removeParameter**
 - *@param int \$parameterId
- **getId**
 - @return \ECGM\Model\BaseArray
- **getParameters**
 - @return \ECGM\Model\BaseArray
- **setParameters**
 - *@param \ECGM\Model\BaseArray \$parameters
 - @throws \ECGM\Exceptions\InvalidArgumentException
- **getParametersAsSimpleArray**
 - @return array
- **setCustomers**
 - *@param \ECGM\Model\BaseArray \$customers
 - @throws \ECGM\Exceptions\InvalidArgumentException
- **getCustomers**
 - @return \ECGM\Model\BaseArray

3.9 \ECGM\Model\Order

- **___construct**
 - *@param mixed \$id
 - *@param BaseArray \$parameters – Parametry zákazníka v době objednávky.
 - *@param \DateTime \$orderDate
 - @throws \ECGM\Exceptions\InvalidArgumentException
- **getId**
 - @return mixed
- **addProduct**
 - *@param \ECGM\Model\OrderProduct \$product
 - @throws \ECGM\Exceptions\InvalidArgumentException
- **removeProduct**
 - *@param int \$productId
- **getCustomerParameters**

- @return \ECGM\Model\BaseArray
- **addProduct**
 - *@param \ECGM\Model\BaseArray \$customerParameters
 - @throws \ECGM\Exceptions\InvalidArgumentException
- **getProducts**
 - @return \ECGM\Model\BaseArray

3.10 \ECGM\Model\StrategyProduct

Zjednodušená reprezentace produktu pro použití při výpočtu strategie zákazníka.

- **___construct**
 - *@param mixed \$id
 - *@param mixed \$orderId
 - *@param float \$price
 - *@param int \$amount
 - *@param float \$discount
- **getId**
 - @return mixed
- **getOrderId**
 - @return mixed
- **getAmount**
 - @return int
- **setAmount**
 - *@param int \$amount
- **getPrice**
 - @return float
- **setPrice**
 - *@param float \$price
- **getDiscountedPrice**
 - @return float|int
- **getDiscount**
 - @return float
- **setDiscount**

- *@param float \$discount

3.11 \ECGM\Model\OrderProduct

Rozšiřuje \ECGM\Model\StrategyProduct. Pro využití v \ECGM\Model\Order.

- **__construct**
 - *@param \ECGM\Model\Product \$product
 - *@param \ECGM\Model\Order \$order
 - *@param int \$amount
- **getExpirationDateType**
 - @return \ECGM\Enum\DateType
- **getComplements**
 - @return \ECGM\Model\BaseArray
- **getExpiration**
 - @return int
- **getOrder**
 - @return \ECGM\Model\Order
- **setOrder**
 - *@param \ECGM\Model\Order \$order

4 CONTROLLER

4.1 \ECGM\Controller\

CustomerParametersMergeController

Implementuje \ECGM\Int\CustomerParametersMergeInterface

Pomocí této třídy dochází k vytvoření váženého průměru parametrů napříč zákaznickovou historií. U extrémních hodnot dochází ke snížení vah, ale ne k jejich odříznutí.

- **mergeCustomerHistory**
 - `*@param \ECGM\Model\BaseArray $customerHistory`
 - `@return \ECGM\Model\BaseArray`
 - `@throws \ECGM\Exceptions\InvalidArgumentException`
 - `@throws \ECGM\Exceptions\UndefinedException`

4.2 \ECGM\Controller\

CustomerParametersCleaningController

Implementuje \ECGM\Int\CustomerParametersCleaningInterface

Pomocí této třídy dochází k očištění parametrů zákazníka a k vytvoření jejich průměrů. Primárně při očišťování dochází k transformaci kruhových hodnot na body ve 2D prostoru.

- **getCustomerParametersMergeController**
 - `@return \ECGM\Int\CustomerParametersMergeInterface`
- **setCustomerParametersMergeController**
 - `*@param \ECGM\Int\CustomerParametersMergeInterface $customerParametersMergeController`
- **cleanCustomerGroups**
 - `*@param \ECGM\Model\BaseArray $customerGroups`
 - `@return \ECGM\Model\BaseArray`
 - `@throws \ECGM\Exceptions\InvalidArgumentException`
- **cleanCustomerGroup**
 - `*@param \ECGM\Model\CustomerGroup $customerGroup`

- @return \ECGM\Model\CustomerGroup
- @throws \ECGM\Exceptions\InvalidArgumentException
- **cleanCustomers**
 - *@param \ECGM\Model\BaseArray \$customers
 - @return \ECGM\Model\BaseArray
 - @throws \ECGM\Exceptions\InvalidArgumentException
- **cleanCustomer**
 - *@param \ECGM\Model\Customer \$customer
 - @return \ECGM\Model\Customer
 - @throws \ECGM\Exceptions\InvalidArgumentException

4.3 \ECGM\Controller\CustomerGroupingController

Implementuje \ECGM\Int\CustomerGroupingInterface

Zapouzdřuje shlukovací algoritmy a pomocí nich shlukuje zákazníky. Volání by mělo probíhat pouze na zákazníky s již očištěnými parametry. V opačném případě může dojít k nesprávnému zařazení.

- **___construct**
 - *@param \ECGM\MainInterface \$mainInterface
 - *@param int \$dimension
 - *@param int \$initK – Počáteční počet shluků. Pokud je \$autoKAdjustment nastaveno na false, nebude se toto číslo měnit.
 - @param bool \$autoKAdjustment = true – Pokud je nastaveno na true, bude docházet k automatickému upravování počtu shluků. Jedná se o výpočetně náročný proces, který není doporučen pokud je počet zákazníků větší než 5000.
 - @param bool \$verbose = false – Pokud bude nastaveno na true, bude docházet k zalogování aktivity shlukovacího algoritmu.
 - @throws \ECGM\Exceptions\InvalidArgumentException
 - @throws \ECGM\Exceptions\UndefinedException
- **getValidationClass**
 - @return \ECGM\Int\GroupingValidationInterface
- **setValidationClass**

- `*@param \ECGM\Int\GroupingValidationInterface $validationClass`
- **getGroupingClass**
 - `@return \ECGM\Int\GroupingImplementationInterface`
- **setGroupingClass**
 - `*@param \ECGM\Int\GroupingImplementationInterface $groupingClass`
- **getDistanceFunctions**
 - `@return \ECGM\Int\DistanceFuncInterface`
- **setDistanceFunctions**
 - `*@param \ECGM\Int\DistanceFuncInterface $distanceFunctions`
- **groupCustomers**
 - `*@param \ECGM\Model\BaseArray $customers`
 - `@param \ECGM\Model\BaseArray $initialGroups = null` – Pokud dojde k naplnění této proměnné, nebudou se vytvářeny počáteční shluky, ale dojde k přidání zákazníků do těchto shluků.
 - `@return \ECGM\Model\BaseArray|mixed|null`
 - `@throws \ECGM\Exceptions\InvalidArgumentException`
 - `@throws \ECGM\Exceptions\LogicalException`
- **assignToGroup**
 - `*@param \ECGM\Model\Customer $customer`
 - `*@param \ECGM\Model\BaseArray $groups`
 - `@return \ECGM\Model\Customer`
 - `@throws \ECGM\Exceptions\InvalidArgumentException`
- **getDimension**
 - `@return int`
- **getK**
 - `@return int`

4.4 \ECGM\Controller\DealerStrategyController

Implementuje `\ECGM\Int\DealerStrategyInterface`.

- **getDealerStrategy**
 - `*@param \ECGM\Model\BaseArray $products`
 - `@return array`

- @throws \ECGM\Exceptions\InvalidArgumentException

4.5 \ECGM\Controller\CustomerStrategyController

Implementuje \ECGM\Int\CustomerStrategyInterface.

- **__construct**
 - *@param \ECGM\MainInterface \$mainInterface
 - *@param float \$coefficient – Hodnota této proměnné je využita pro zvýšení váhy nákupů zákazníka oproti nákupům jeho skupiny a ke zvýšení váhy doplňků. Čím vyšší, tím více budou tyto hodnoty upřednostňovány.
 - @throws \ECGM\Exceptions\InvalidArgumentException
- **getCustomerStrategy**
 - *@param \ECGM\Model\Customer \$customer
 - *@param \ECGM\Model\BaseArray \$currentProducts
 - @param \ECGM\Model\Order|null \$currentOrder = null – Hodnota reprezentující aktuální nákup zákazníka, pokud již má nějaké produkty v košíku.
 - @return array
 - @throws \ECGM\Exceptions\InvalidArgumentException

4.6 \ECGM\Controller\StrategyController

Implementuje \ECGM\Int\StrategyInterface.

- **__construct**
 - *@param float \$coefficient – Hodnota této proměnné je využita pro zvýšení váhy nákupů zákazníka oproti nákupům jeho skupiny a ke zvýšení váhy doplňků. Čím vyšší, tím více budou tyto hodnoty upřednostňovány.
 - *@param \ECGM\MainInterface \$mainInterface
 - @param int \$maxProductsInStrategy = 40 – Počet produktů, které budou zahrnuty do výpočtu strategie. Pokud bude tato proměnná nastavena na 40, bude do strategie zahrnuto prvních 40 produktů, které jsou ve strategii zákazníka nejvýše.
 - @throws \ECGM\Exceptions\InvalidArgumentException
- **getPassiveStrategyController**

- @return \ECGM\Int\StrategyTypeInterface
- **setPassiveStrategyController**
 - *@param \ECGM\Int\StrategyTypeInterface \$passiveStrategyController
- **getConservativeStrategyController**
 - @return \ECGM\Int\StrategyTypeInterface
- **setConservativeStrategyController**
 - *@param \ECGM\Int\StrategyTypeInterface \$conservativeStrategyController
- **getAggressiveStrategyController**
 - @return \ECGM\Int\StrategyTypeInterface
- **setAggressiveStrategyController**
 - *@param \ECGM\Int\StrategyTypeInterface \$aggressiveStrategyController
- **getStrategy**
 - *@param \ECGM\Model\Customer \$customer
 - *@param \ECGM\Model\AssociativeBaseArray \$currentProducts – Produkty, které jsou aktuálně v prodeji.
 - @param \ECGM\Model\Order|null \$currentOrder = null
 - *@param \ECGM\Enum\StrategyType \$strategyType = \ECGM\Enum\StrategyType::CONSERVATIVE
 - @return \ECGM\Model\AssociativeBaseArray
 - @throws \ECGM\Exceptions\InvalidArgumentException
 - @throws \ECGM\Exceptions\LogicalException
 - @throws \ReflectionException

4.7 \ECGM\Controller\PassiveStrategyTypeController

Implementuje \ECGM\Int\StrategyTypeInterface.

Tento typ strategie určí pořadí produktů v jakém je nejvýhodnější zákazníkovi prodat, tak aby byla zároveň co nejlépe zachována předpovězená strategie daného zákazníka. Nedochází ke změně slev produktů.

- **___construct**
 - *@param float \$coefficient – Hodnota této proměnné je využita pro zvýšení váhy nákupů zákazníka oproti nákupům jeho skupiny a ke zvýšení váhy doplňků. Čím vyšší, tím více budou tyto hodnoty upřednostňovány.

- `*@param \ECGM\MainInterface $mainInterface`
- `@param int $maxProductsInStrategy = 40` – Počet produktů, které budou zahrnuty do výpočtu strategie. Pokud bude tato proměnná nastavena na 40, bude do strategie zahrnuto prvních 40 produktů, které jsou ve strategii zákazníka nejvýše.
- `@throws \ECGM\Exceptions\InvalidArgumentException`
- **getCustomerStrategyController**
 - `@return \ECGM\Int\CustomerStrategyInterface`
- **setCustomerStrategyController**
 - `*@param \ECGM\Int\CustomerStrategyInterface $customerStrategyController`
- **getDealerStrategyController**
 - `@return \ECGM\Int\DealerStrategyInterface`
- **setDealerStrategyController**
 - `*@param \ECGM\Int\DealerStrategyInterface $dealerStrategyController`
- **getIdealStrategy**
 - `*@param \ECGM\Model\Customer $customer`
 - `*@param \ECGM\Model\AssociativeBaseArray $currentProducts` – Produkty, které jsou aktuálně v prodeji.
 - `@param \ECGM\Model\Order|null $currentOrder = null`
 - `@return \ECGM\Model\AssociativeBaseArray` – Produkty jsou vráceny seřazeně od nejlepšího po nejhorší.
 - `@throws \ECGM\Exceptions\InvalidArgumentException`

4.8 \ECGM\Controller\ConservativeStrategyTypeController

Implementuje \ECGM\Int\StrategyTypeInterface.

Tento typ strategie navrhne minimální možnou slevu takovou, aby došlo k upřednostnění produktu ve strategii zákazníka, ale zároveň nedošlo k jeho poklesu ve strategii prodejce. Slevy jsou přiřazovány pouze produktům, u kterých mají smysl, to znamená, že pokud by nebyla sleva pro prodejce výhodná, není aplikována.

- **__construct**

- `*@param float $coefficient` – Hodnota této proměnné je využita pro zvýšení váhy nákupů zákazníka oproti nákupům jeho skupiny a ke zvýšení váhy doplňků. Čím vyšší, tím více budou tyto hodnoty upřednostňovány.
- `*@param \ECGM\MainInterface $mainInterface`
- `@param int $maxProductsInStrategy = 40` – Počet produktů, které budou zahrnuty do výpočtu strategie. Pokud bude tato proměnná nastavena na 40, bude do strategie zahrnuto prvních 40 produktů, které jsou ve strategii zákazníka nejvýše.
- `@throws \ECGM\Exceptions\InvalidArgumentException`
- **getCustomerStrategyController**
 - `@return \ECGM\Int\CustomerStrategyInterface`
- **setCustomerStrategyController**
 - `*@param \ECGM\Int\CustomerStrategyInterface $customerStrategyController`
- **getDealerStrategyController**
 - `@return \ECGM\Int\DealerStrategyInterface`
- **setDealerStrategyController**
 - `*@param \ECGM\Int\DealerStrategyInterface $dealerStrategyController`
- **getIdealStrategy**
 - `*@param \ECGM\Model\Customer $customer`
 - `*@param \ECGM\Model\AssociativeBaseArray $currentProducts` – Produkty, které jsou aktuálně v prodeji.
 - `@param \ECGM\Model\Order|null $currentOrder = null`
 - `@return \ECGM\Model\AssociativeBaseArray` – Produkty jsou vráceny seřazeně od nejlepšího po nejhorší, společně s navrženými slevami.
 - `@throws \ECGM\Exceptions\InvalidArgumentException`

4.9 \ECGM\Controller\AggressiveStrategyTypeController

Implementuje `\ECGM\Int\StrategyTypeInterface`.

Tento typ strategie navrhne maximální možnou slevu takovou, aby došlo k upřednostnění produktu ve strategii zákazníka, ale zároveň nedošlo k jeho poklesu ve strategii prodejce. Slevy jsou přiřazovány pouze produktům, u kterých mají smysl, to znamená, že pokud by nebyla sleva pro prodejce výhodná, není aplikována.

- **__construct**
 - `*@param float $coefficient` – Hodnota této proměnné je využita pro zvýšení váhy nákupů zákazníka oproti nákupům jeho skupiny a ke zvýšení váhy doplňků. Čím vyšší, tím více budou tyto hodnoty upřednostňovány.
 - `*@param \ECGM\MainInterface $mainInterface`
 - `@param int $maxProductsInStrategy = 40` – Počet produktů, které budou zahrnuty do výpočtu strategie. Pokud bude tato proměnná nastavena na 40, bude do strategie zahrnuto prvních 40 produktů, které jsou ve strategii zákazníka nejvýše.
 - `@throws \ECGM\Exceptions\InvalidArgumentException`
- **getCustomerStrategyController**
 - `@return \ECGM\Int\CustomerStrategyInterface`
- **setCustomerStrategyController**
 - `*@param \ECGM\Int\CustomerStrategyInterface $customerStrategyController`
- **getDealerStrategyController**
 - `@return \ECGM\Int\DealerStrategyInterface`
- **setDealerStrategyController**
 - `*@param \ECGM\Int\DealerStrategyInterface $dealerStrategyController`
- **getIdealStrategy**
 - `*@param \ECGM\Model\Customer $customer`
 - `*@param \ECGM\Model\AssociativeBaseArray $currentProducts` – Produkty, které jsou aktuálně v prodeji.
 - `@param \ECGM\Model\Order|null $currentOrder = null`
 - `@return \ECGM\Model\AssociativeBaseArray` – Produkty jsou vráceny seřazeně od nejlepšího po nejhorší, společně s navrženými slevami.
 - `@throws \ECGM\Exceptions\InvalidArgumentException`

5 POMOCNÉ TŘÍDY

5.1 \ECGM\Util\DistanceFunctions

Implementuje \ECGM\Int\DistanceFuncInterface.

- **distanceQuick** – Rychlejší, ale méně přesná implementace distančního algoritmu.

V tomto případě se jedná o Manhattanskou vzdálenost.

- `*@param array $v1`
- `*@param array $v2`
- `@return float|int`
- `@throws \ECGM\Exceptions\InvalidArgumentException`

- **distancePrecise** – Pomalejší, ale přesnější implementace distančního algoritmu. V

tomto případě se jedná o Euklidovskou vzdálenost.

- `*@param array $v1`
- `*@param array $v2`
- `@return float|int`
- `@throws \ECGM\Exceptions\InvalidArgumentException`

5.2 \ECGM\Util\MathFunctions

- **arrayMedian**

- `*@param array $array`
- `@return float|int`
- `@throws \ECGM\Exceptions\UndefinedException`

5.3 \ECGM\Util\MiscFunctions

- **mergeAssociativeArrays** – Sloučí asociativní pole.

- `*@param array $arr1`
- `*@param array $arr2`
- `@return array`
- `@throws \ECGM\Exceptions\InvalidArgumentException`

- **isAssoc**

- *@param array \$arr
- @return bool

5.4 \ECGM\Util\KmeansPlusPlus

Implementuje \ECGM\Int\GroupingImplementationInterface.

Implementace shlukovacího algoritmu k-means++.

- **__construct**
 - *@param int \$dimension
 - @throws \ECGM\Exceptions\InvalidArgumentException
 - @throws \ECGM\Exceptions\UndefinedException
- **getDistanceFunctions**
 - @return \ECGM\Int\DistanceFuncInterface
- **setDistanceFunctions**
 - *@param \ECGM\Int\DistanceFuncInterface \$distanceFunctions
- **setInitialGroups**
 - *@param \ECGM\Model\BaseArray \$initialGroups
 - @throws \ECGM\Exceptions\InvalidArgumentException
- **addCustomer**
 - *@param \ECGM\Model\Customer \$customer
 - @throws \ECGM\Exceptions\InvalidArgumentException
- **setCustomers**
 - *@param \ECGM\Model\BaseArray \$customers
 - @throws \ECGM\Exceptions\InvalidArgumentException
- **getDimension**
 - @return int
- **solve**
 - *@param int \$nbGroups
 - @return \ECGM\Model\BaseArray|mixed|null
 - @throws \ECGM\Exceptions\InvalidArgumentException
- **getCustomerParametersAsArray**
 - *@param \ECGM\Model\BaseArray \$parameters
 - @return array

- @throws \ECGM\Exceptions\InvalidArgumentException

5.5 \ECGM\Util\SilhouetteAnalysis

Implementuje \ECGM\Int\GroupingValidationInterface.

Implementace siluetové analýzy, pro měření správnosti zařazení objektu do shluku.

- **__construct**
 - @param bool \$verbose = false – Pokud nastaveno na true, dojde k logování výsledků.
- **getDistanceFunctions**
 - @return \ECGM\Int\DistanceFuncInterface
- **setDistanceFunctions**
 - *@param \ECGM\Int\DistanceFuncInterface \$distanceFunctions
- **getDistanceFunctions**
 - *@param \ECGM\Model\DistanceFuncInterface \$groups
 - @return float|int
 - @throws \ECGM\Exceptions\InvalidArgumentException
 - @throws \ECGM\Exceptions\LogicalException