Model::Backend

User

-id: int {unique}
-email: string {unique}
-password: string
-blacklist: Ingredient[0..*]

+create(email: string, password: string): User
+get_by_id(id: int): User
+get_by_email(email: string): User
+__construct(id: int, email: string, password: string)
+get_id(): int
+get_email(): string
+check_password(password: string): boolean
+set_email(email: string): boolean
+set_password(password: string): boolean
+get_blacklist(): Ingredient
+add_to_blacklist(item_id: int): boolean
+rm_from_blacklist(item_id: int): boolean

Session

-token: string
-user: User
-admin: bool

+generate_barcode(): int
+__construct(token: string)
+__construct(email: string, password: string)
+__construct(barcode: int)
+destroy(): boolean
+get_user(): User
+get_token(): string
+is_admin(): bool
+is_token(token: string): boolean

API

-path: string
-declarations: string[0..*]

+get(path: string, exec: function)
+post(path: string, exec: function)
+put(path: string, exec: function)
+delete(path: string, exec: function)
+define(regexp: string, name: string)

Exception

#message: string
#code: int

+getMessage(): string
+getCode(): int

Product

-ean: int {unique}
-name: string
-ingredients: Ingredient[1..*]

+create(ean: int, name: string): Product
+get_by_id(id: int): Product
+get_by_name(): Product[0..*]
+__construct(id: int, name: string)
+get_id(): int
+get_name(): string
+set_name(name: string): boolean
+get_ingredients(): Ingredient[1..*]
+add_ingredient(ingredient: Ingredient): boolean
+rm_ingredient(ingredient: Ingredient): boolean

Ingredient

-id: int {unique}
-name: string

+get_by_id(id: int): Ingredient
+get_by_name(name: string): Ingredient[0..*]
+create(name: string): Ingredient
+__construct(id: int, name: string)
+get_id(): int
+get_name(): string
+set_name(name: string): boolean

UserError

InternalError