



## Obsah

1. Návrhová REST API .....	5
1.1 bussiness .....	5
1.1.1 dto .....	6
1.1.1.1 Class ActivityCreateDTO .....	7
1.1.1.2 Class ActivityReadDTO .....	7
1.1.1.3 Class ActivityUpdateDTO .....	8
1.1.1.4 Class FoodCreateDTO .....	8
1.1.1.5 Class FoodMicronutrientCreateDTO .....	8
1.1.1.6 Class FoodMicronutrientReadDTO .....	8
1.1.1.7 Class FoodMicronutrientUpdateDTO .....	8
1.1.1.8 Class FoodReadDTO .....	8
1.1.1.9 Class FoodUpdateDTO .....	9
1.1.1.10 Class GoalCreateDTO .....	9
1.1.1.11 Class GoalDTO .....	9
1.1.1.12 Class GoalReadDTO .....	9
1.1.1.13 Class GoalUpdateDTO .....	9
1.1.1.14 Class MealCreateDTO .....	9
1.1.1.15 Class MealReadDTO .....	10
1.1.1.16 Class MealUpdateDTO .....	10
1.1.1.17 Class MicronutrientCreateDTO .....	10
1.1.1.18 Class MicronutrientUpdateDTO .....	10
1.1.1.19 Class UserActivityCreateDTO .....	10
1.1.1.20 Class UserActivityReadDTO .....	10
1.1.1.21 Class UserActivityUpdateDTO .....	10
1.1.1.22 Class UserCreateDTO .....	11
1.1.1.23 Class UserMealFoodCreateDTO .....	11
1.1.1.24 Class UserMealFoodReadDTO .....	11
1.1.1.25 Class UserMealFoodUpdateDTO .....	11
1.1.1.26 Class UserReadDTO .....	11
1.1.1.27 Class UserUpdateDTO .....	12
1.1.1.28 Interface ICreateDTO .....	12
1.1.1.29 Interface IReadDTO .....	12
1.1.1.30 Interface IUpdateDTO .....	12
1.1.2 service .....	12
1.1.2.1 Class ActivityService .....	13
1.1.2.2 Class FoodMicronutrientService .....	14
1.1.2.3 Class FoodService .....	15
1.1.2.4 Class GoalService .....	17
1.1.2.5 Class MealService .....	18
1.1.2.6 Class MicronutrientService .....	19
1.1.2.7 Class UserActivityService .....	20
1.1.2.8 Class UserMealFoodService .....	21
1.1.2.9 Class UserService .....	22
1.1.2.10 Interface IService .....	23



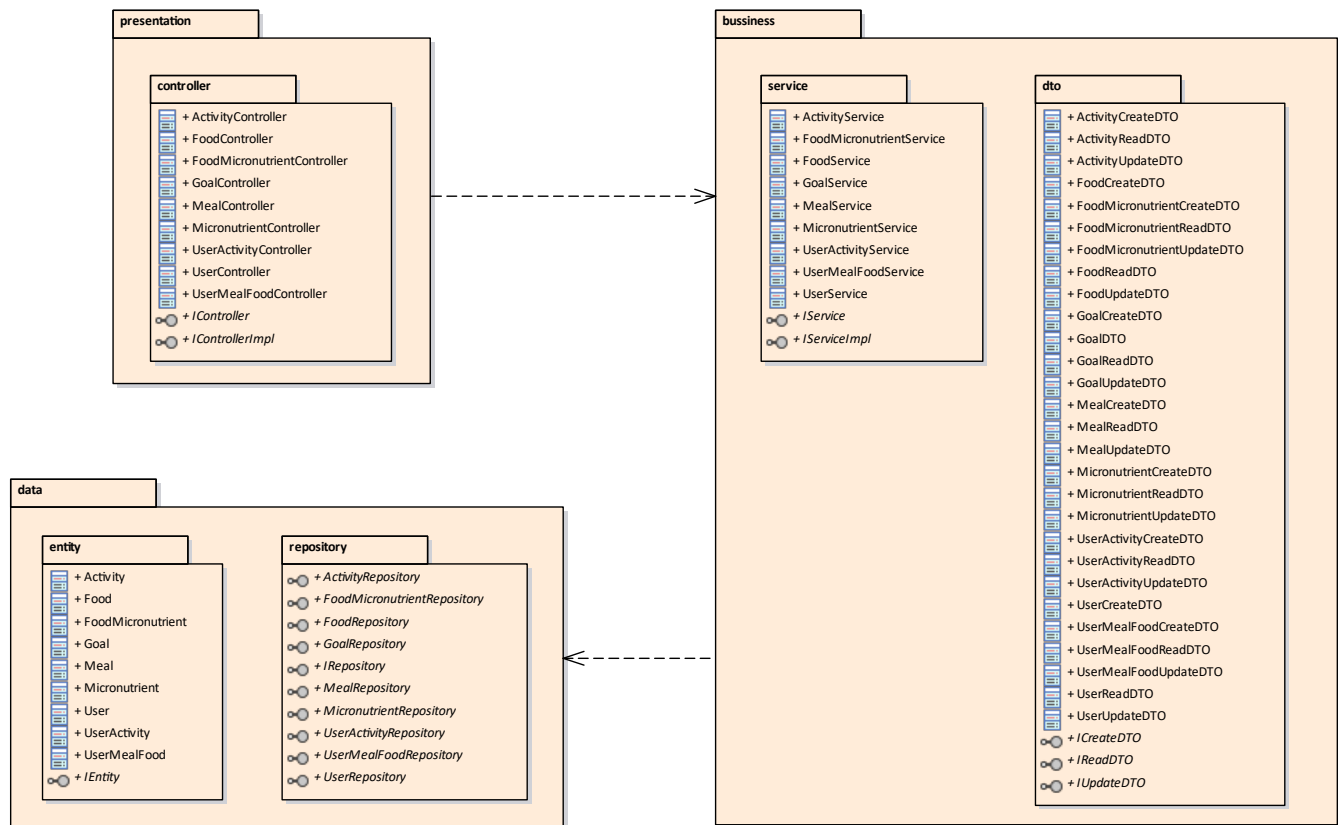
1.1.2.11	Interface IServiceImpl.....	24
1.2	data .....	26
1.2.1	entity .....	26
1.2.1.1	Class Activity.....	26
1.2.1.2	Class Food.....	27
1.2.1.3	Class FoodMicronutrient.....	27
1.2.1.4	Class Goal.....	27
1.2.1.5	Class Meal.....	27
1.2.1.6	Class Micronutrient.....	27
1.2.1.7	Class User.....	27
1.2.1.8	Class UserActivity.....	27
1.2.1.9	Class UserMealFood.....	28
1.2.1.10	Interface IEntity.....	28
1.2.2	repository .....	28
1.2.2.1	Interface ActivityRepository.....	29
1.2.2.2	Interface FoodMicronutrientRepository.....	29
1.2.2.3	Interface FoodRepository.....	29
1.2.2.4	Interface GoalRepository.....	30
1.2.2.5	Interface IRepository.....	30
1.2.2.6	Interface MicronutrientRepository.....	30
1.2.2.7	Interface UserActivityRepository.....	30
1.2.2.8	Interface UserMealFoodRepository.....	30
1.2.2.9	Interface UserRepository.....	31
1.3	presentation .....	31
1.3.1	controller .....	31
1.3.1.1	Class ActivityController.....	32
1.3.1.2	Class FoodController.....	33
1.3.1.3	Class FoodMicronutrientController.....	34
1.3.1.4	Class GoalController.....	35
1.3.1.5	Class MealController.....	36
1.3.1.6	Class MicronutrientController.....	37
1.3.1.7	Class UserActivityController.....	37
1.3.1.8	Class UserController.....	39
1.3.1.9	Class UserMealFoodController.....	40
1.3.1.10	Interface IController.....	41
1.3.1.11	Interface IControllerImpl.....	42
2.	Návrh logické architektury (Applikace) .....	43
2.1	Model .....	44
2.2	Service .....	45
2.2.1	Interface AuthServicing.....	45
2.2.2	Interface CaloryServicing.....	46
2.2.3	Interface NetworkServicing.....	46
2.3	View .....	47
2.3.1	Activity .....	48
2.3.1.1	Class ActivityRecordView.....	48
2.3.1.2	Class ActivitySheet.....	48
2.3.1.3	Class ActivityView.....	48
2.3.1.4	Class AddActivityView.....	49
2.3.2	Dashboard .....	49
2.3.2.1	Class ActivityInfoRow.....	50
2.3.2.2	Class ActivityInfoView.....	50
2.3.2.3	Class ActivityTabView.....	50



2.3.2.4	Class CircleView.....	50
2.3.2.5	Class DashboardView.....	50
2.3.2.6	Class FoodTabView.....	50
2.3.2.7	Class MealInfoView.....	51
2.3.2.8	Class MealNameRow.....	51
2.3.2.9	Class MinusButton.....	51
2.3.2.10	Class PlusButton.....	51
2.3.3	Food .....	51
2.3.3.1	Class AddFoodView.....	52
2.3.3.2	Class FoodRecordView.....	52
2.3.3.3	Class FoodSheet.....	53
2.3.3.4	Class FoodView.....	53
2.4	ViewModel .....	53
2.4.1	Activity .....	53
2.4.1.1	Class ActivityViewModel.....	54
2.4.1.2	Class AddActivityViewModel.....	54
2.4.2	Dashboard .....	55
2.4.2.1	Class DashboardViewModel .....	55
2.4.3	Food .....	57
2.4.3.1	Class AddFoodViewModel.....	57
2.4.3.2	Class FoodViewModel.....	58
3.	Realizace případů užití (Model komunikace) .....	59
3.1	Zaznamenání aktivity .....	59
3.2	Zaznamenání konzumace jídla .....	60
3.3	Zobrazení a hledání aktivit .....	60
3.4	Zobrazení a hledání jídel .....	61



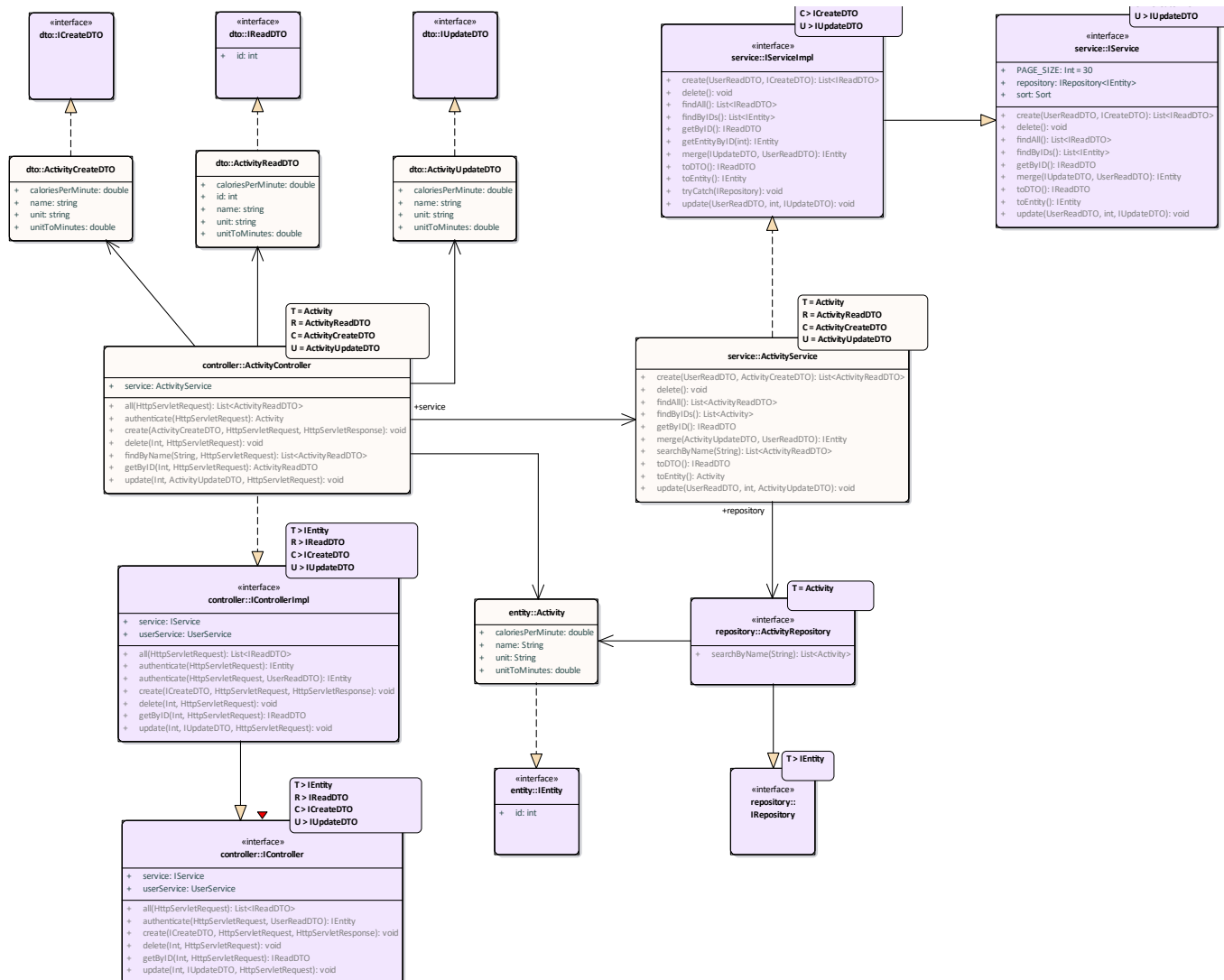
## 1. Návrhová REST API



Obrázek 1 - Návrhová REST API

### 1.1 bussiness

Bussiness vrstva zajišťuje zpracování dat mezi Datovou a Prezentační vrstvou.

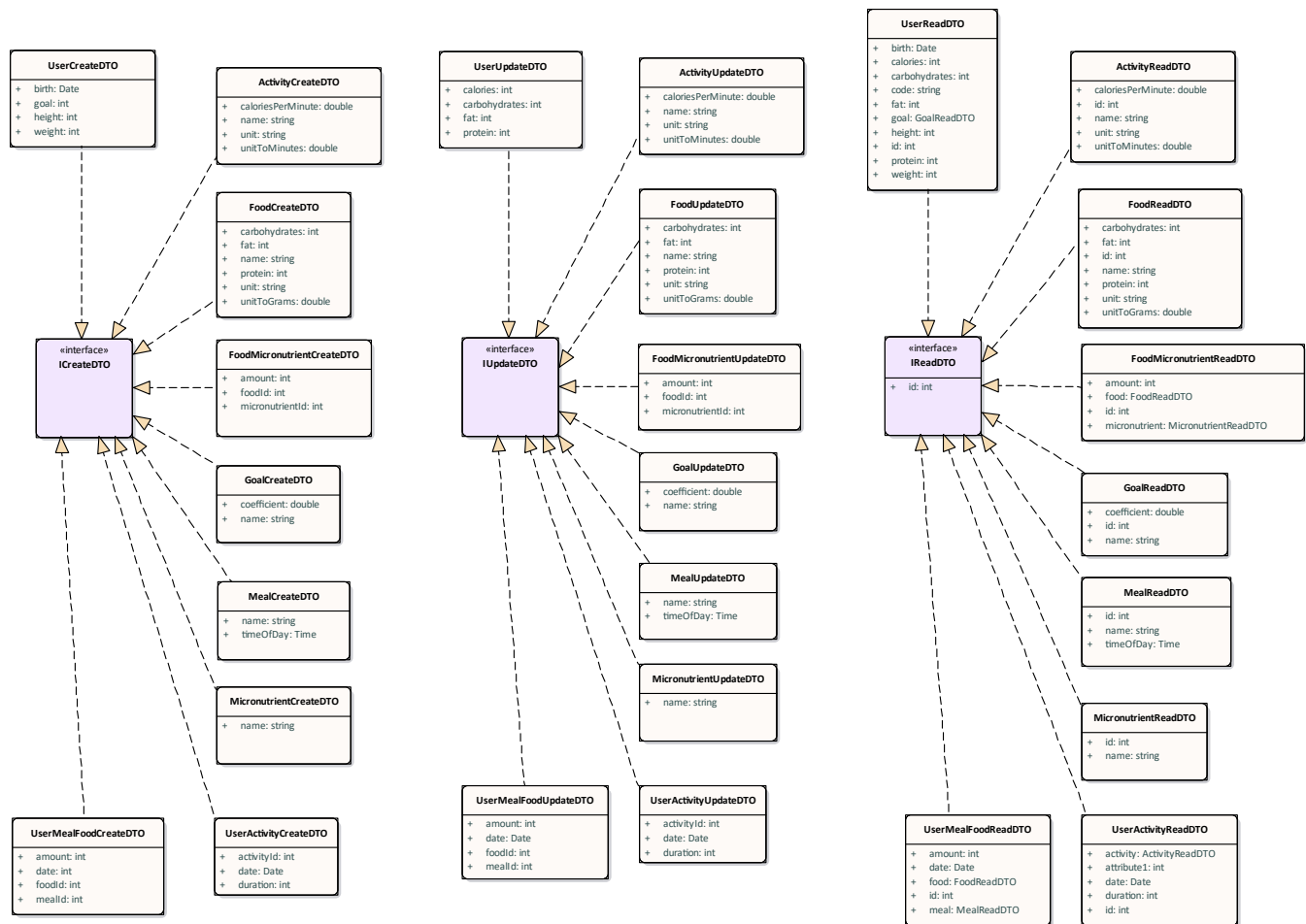


Obrázek 2 - Napojení na ostatní vrstvy

Ukázková závislost jednotlivých vrstev balíčků, stejný model platí pro každou z entit.

### 1.1.1 dto

Balíček obsahuje všechny třídy, které slouží ke komunikaci Business vrstvy s Prezentační vrstvou. Zavedení DTO nad standardními entitami umožňuje prevenci zacyklení (objekty jsou nahrazeny svými ID) a také provádění dotazů, které nezávisí na konkrétních entitách.



Obrázek 3 - dto

### 1.1.1.1 Class ActivityCreateDTO

Název atributu	Datový typ	Popis
caloriesPerMinute	double	
name	string	
unit	string	
unitToMinutes	double	

### 1.1.1.2 Class ActivityReadDTO

Název atributu	Datový typ	Popis
caloriesPerMinute	double	
id	int	Členská proměnná přepisuje základní identifikátor rozhraní.
name	string	
unit	string	
unitToMinutes	double	

### 1.1.1.3 Class ActivityUpdateDTO

Název atributu	Datový typ	Popis
caloriesPerMinute	double	
name	string	
unit	string	
unitToMinutes	double	

### 1.1.1.4 Class FoodCreateDTO

Název atributu	Datový typ	Popis
carbohydrates	int	
fat	int	
name	string	
protein	int	
unit	string	
unitToGrams	double	

### 1.1.1.5 Class FoodMicronutrientCreateDTO

Název atributu	Datový typ	Popis
amount	int	
foodId	int	
micronutrientId	int	

### 1.1.1.6 Class FoodMicronutrientReadDTO

Název atributu	Datový typ	Popis
amount	int	
food	FoodReadDTO	
id	int	Členská proměnná přepisuje základní identifikátor rozhraní.
micronutrient	MicronutrientReadDT O	

### 1.1.1.7 Class FoodMicronutrientUpdateDTO

Název atributu	Datový typ	Popis
amount	int	
foodId	int	
micronutrientId	int	

### 1.1.1.8 Class FoodReadDTO

Název atributu	Datový typ	Popis
carbohydrates	int	
fat	int	
id	int	Členská proměnná přepisuje základní identifikátor rozhraní.

name	string	
protein	int	
unit	string	
unitToGrams	double	

### 1.1.1.9 Class FoodUpdateDTO

Název atributu	Datový typ	Popis
carbohydrates	int	
fat	int	
name	string	
protein	int	
unit	string	
unitToGrams	double	

### 1.1.1.10 Class GoalCreateDTO

Název atributu	Datový typ	Popis
coefficient	double	
name	string	

### 1.1.1.11 Class GoalDTO

Název atributu	Datový typ	Popis
id	int	Členská proměnná přepisuje základní identifikátor rozhraní.

### 1.1.1.12 Class GoalReadDTO

Název atributu	Datový typ	Popis
coefficient	double	
id	int	Členská proměnná přepisuje základní identifikátor rozhraní.
name	string	

### 1.1.1.13 Class GoalUpdateDTO

Název atributu	Datový typ	Popis
coefficient	double	
name	string	

### 1.1.1.14 Class MealCreateDTO

Název atributu	Datový typ	Popis
name	string	
timeOfDay	Time	





#### 1.1.1.15 Class MealReadDTO

Název atributu	Datový typ	Popis
id	int	Členská proměnná přepisuje základní identifikátor rozhraní.
name	string	
timeOfDay	Time	

#### 1.1.1.16 Class MealUpdateDTO

Název atributu	Datový typ	Popis
name	string	
timeOfDay	Time	

#### 1.1.1.17 Class MicronutrientCreateDTO

Název atributu	Datový typ	Popis
name	string	

#### 1.1.1.18 Class MicronutrientUpdateDTO

Název atributu	Datový typ	Popis
name	string	

#### 1.1.1.19 Class UserActivityCreateDTO

Název atributu	Datový typ	Popis
activityId	int	
date	Date	
duration	int	

#### 1.1.1.20 Class UserActivityReadDTO

Název atributu	Datový typ	Popis
activity	ActivityReadDTO	
attribute1	int	
date	Date	
duration	int	
id	int	Členská proměnná přepisuje základní identifikátor rozhraní.

#### 1.1.1.21 Class UserActivityUpdateDTO

Název atributu	Datový typ	Popis
activityId	int	
date	Date	
duration	int	



### 1.1.1.22 Class UserCreateDTO

Název atributu	Datový typ	Popis
birth	Date	
goal	int	
height	int	
weight	int	

### 1.1.1.23 Class UserMealFoodCreateDTO

Název atributu	Datový typ	Popis
amount	int	
date	int	
foodId	int	
mealId	int	

### 1.1.1.24 Class UserMealFoodReadDTO

Název atributu	Datový typ	Popis
amount	int	
date	Date	
food	FoodReadDTO	
id	int	Členská proměnná přepisuje základní identifikátor rozhraní.
meal	MealReadDTO	

### 1.1.1.25 Class UserMealFoodUpdateDTO

Název atributu	Datový typ	Popis
amount	int	
date	Date	
foodId	int	
mealId	int	

### 1.1.1.26 Class UserReadDTO

Název atributu	Datový typ	Popis
birth	Date	
calories	int	
carbohydrates	int	
code	string	
fat	int	
goal	GoalReadDTO	
height	int	
id	int	Členská proměnná přepisuje základní identifikátor rozhraní.
protein	int	
weight	int	



### 1.1.1.27 Class UserUpdateDTO

Název atributu	Datový typ	Popis
calories	int	
carbohydrates	int	
fat	int	
protein	int	

### 1.1.1.28 Interface ICreateDTO

Rozhraní pro společnou implementaci tříd určujících strukturu Entit při tvorbě dat HTTP metodou POST. Neobsahuje ID protože to se tvoří v databázi samo, pomocí funkce auto-increment. ID je pak v odpovědi na tvorbu navraceno v implementaci IReadDTO.

### 1.1.1.29 Interface IReadDTO

Rozhraní pro společnou implementaci tříd určujících strukturu Entit při získávání dat HTTP metodou GET.

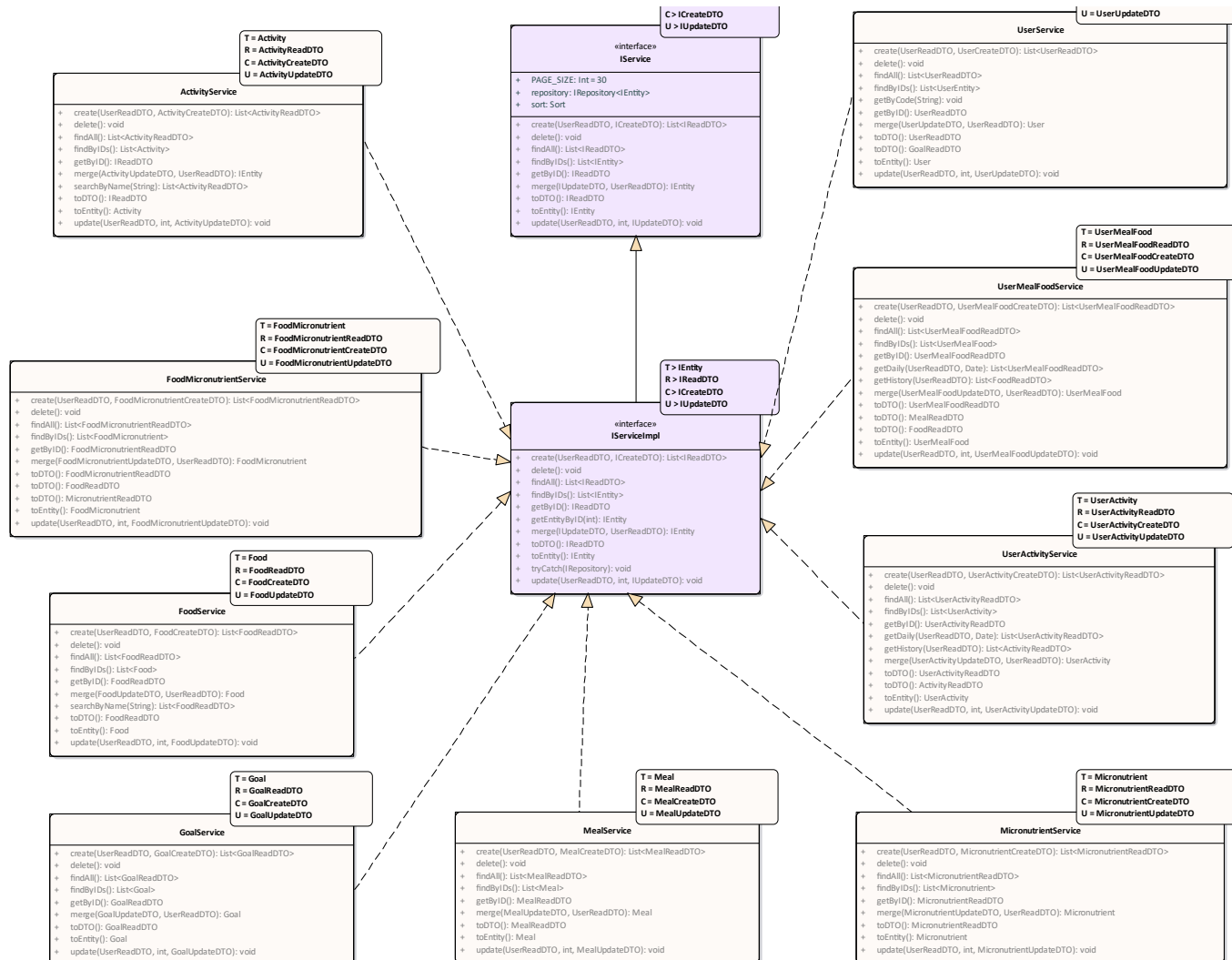
Název atributu	Datový typ	Popis
id	int	

### 1.1.1.30 Interface IUpdateDTO

Rozhraní pro společnou implementaci tříd určujících strukturu Entit při úpravě dat HTTP metodou PUT. Všechny parametry jsou nepovinné a nezmíněná data se neuloží. Neobsahuje ID protože to je specifikováno v URL dotazu, zároveň neumožňujeme identifikátor entity měnit.

## 1.1.2 service

Balíček je určen pro třídy, které přijmou a zpracují z Prezentační vrstvy jednotlivé požadavky. Součástí zpracování požadavků je také čtení, zapisování a úprava dat v databázi a práce s Repositories, která probíhá přes rozhraní.



Obrázek 4 - service

### 1.1.2.1 Class ActivityService

Název metody	Návratový typ	Popis
create	List<ActivityReadDTO> >	Metoda sloužící k vytvoření entit na základě daných vstupních Create DTO. Parametry: <b>user: UserReadDTO</b> - Parametry: <b>dtos: ActivityCreateDTO</b> -
delete	void	Metoda sloužící k vymazání entity na základě daného identifikátoru

findAll	List<ActivityReadDTO>	Metoda sloužící k navrácení seznamu všech entit, které jsou v databázi.
findByIds	List<Activity>	Metoda sloužící k navrácení seznamu příslušných entit na základě daného seznamu identifikátorů entit. Pokud entita s daným identifikátorem není nalezena, nebude přidána do výstupního seznamu.
getById	IReadDTO	Metoda sloužící k navrácení příslušné Read DTO na základě daného identifikátoru entity.
merge	IEntity	Metoda přetěžuje základní implementaci rozhraní a slouží k aktualizaci entit na základě vstupních Update DTO. Pokud je jakékoliv pole prázdné - informace jsou nekompletní, aktualizace entity neproběhne. Parametry: <b>dto: ActivityUpdateDTO</b> - Parametry: <b>user: UserReadDTO</b> -
searchByName	List<ActivityReadDTO>	Metoda sloužící k hledání aktivit na základě částečné shody v názvu aktivit. Metoda vrací seznam nalezených aktivit. Parametry: <b>name: String</b> -
toDTO	IReadDTO	Metoda přetěžuje základní implementaci rozhraní a slouží k mapování entit na příslušné Read DTO.
toEntity	Activity	Metoda přetěžuje základní implementaci rozhraní a slouží k mapování Create DTO na příslušnou entitu.
update	void	Metoda sloužící k aktualizaci entit na základě daných vstupních Update DTO. Parametry: <b>user: UserReadDTO</b> - Parametry: <b>id: int</b> - Parametry: <b>dto: ActivityUpdateDTO</b> -

### 1.1.2.2 Class FoodMicronutrientService

Název metody	Návratový typ	Popis
create	List<FoodMicronutrientReadDTO>	Metoda sloužící k vytvoření entit na základě daných vstupních Create DTO. Parametry: <b>user: UserReadDTO</b> - Parametry: <b>dtos: FoodMicronutrientCreateDTO</b> -

delete	void	Metoda sloužící k vymazání entity na základě daného identifikátoru
findAll	List<FoodMicronutrientReadDTO>	Metoda sloužící k navrácení seznamu všech entit, které jsou v databázi.
findByIds	List<FoodMicronutrient>	Metoda sloužící k navrácení seznamu příslušných entit na základě daného seznamu identifikátorů entit. Pokud entita s daným identifikátorem není nalezena, nebude přidána do výstupního seznamu.
getById	FoodMicronutrientReadDTO	Metoda sloužící k navrácení příslušné Read DTO na základě daného identifikátoru entity.
merge	FoodMicronutrient	Metoda přetěžuje základní implementaci rozhraní a slouží k aktualizaci entit na základě vstupních Update DTO. Pokud je jakékoliv pole prázdné - informace jsou nekompletní, aktualizace entity neproběhne. Parametry: <b>dto: FoodMicronutrientUpdateDTO -</b> Parametry: <b>user: UserReadDTO -</b>
toDTO	FoodMicronutrientReadDTO	Metoda přetěžuje základní implementaci rozhraní a slouží k mapování entit na příslušné Read DTO.
toDTO	FoodReadDTO	Pomocná rozšiřující metoda sloužící ke konverzi Food na FoodReadDTO.
toDTO	MicronutrientReadDTO	Pomocná rozšiřující metoda sloužící ke konverzi Micronutrient na MicronutrientReadDTO.
toEntity	FoodMicronutrient	Metoda přetěžuje základní implementaci rozhraní a slouží k mapování Create DTO na příslušnou entitu.
update	void	Metoda sloužící k aktualizaci entit na základě daných vstupních Update DTO. Parametry: <b>user: UserReadDTO -</b> Parametry: <b>id: int -</b> Parametry: <b>dto: FoodMicronutrientUpdateDTO -</b>

### 1.1.2.3 Class FoodService

Název metody	Návratový typ	Popis
create	List<FoodReadDTO>	Metoda sloužící k vytvoření entit na základě daných vstupních Create DTO.



		Parametry: <b>user: UserReadDTO -</b> Parametry: <b>dtos: FoodCreateDTO -</b>
delete	void	Metoda sloužící k vymazání entity na základě daného identifikátoru
findAll	List<FoodReadDTO>	Metoda sloužící k navrácení seznamu všech entit, které jsou v databázi.
findByIDs	List<Food>	Metoda sloužící k navrácení seznamu příslušných entit na základě daného seznamu identifikátorů entit. Pokud entita s daným identifikátorem není nalezena, nebude přidána do výstupního seznamu.
getByID	FoodReadDTO	Metoda sloužící k navrácení příslušné Read DTO na základě daného identifikátoru entity.
merge	Food	Metoda přetěžuje základní implementaci rozhraní a slouží k aktualizaci entit na základě vstupních Update DTO. Pokud je jakékoliv pole prázdné - informace jsou nekompletní, aktualizace entity neproběhne. Parametry: <b>dto: FoodUpdateDTO -</b> Parametry: <b>user: UserReadDTO -</b>
searchByName	List<FoodReadDTO>	Metoda sloužící k hledání jídel na základě částečné shody v názvu jídel. Metoda vrací seznam nalezených jídel. Parametry: <b>name: String -</b>
toDTO	FoodReadDTO	Metoda přetěžuje základní implementaci rozhraní a slouží k mapování entit na příslušné Read DTO.
toEntity	Food	Metoda přetěžuje základní implementaci rozhraní a slouží k mapování Create DTO na příslušnou entitu.
update	void	Metoda sloužící k aktualizaci entit na základě daných vstupních Update DTO. Parametry: <b>user: UserReadDTO -</b> Parametry: <b>id: int -</b> Parametry: <b>dto: FoodUpdateDTO -</b>

### 1.1.2.4 Class GoalService

Název metody	Návratový typ	Popis
create	List<GoalReadDTO>	Metoda sloužící k vytvoření entit na základě daných vstupních Create DTO. Parametry: <b>user: UserReadDTO -</b> Parametry: <b>dtos: GoalCreateDTO -</b>
delete	void	Metoda sloužící k vymazání entity na základě daného identifikátoru
findAll	List<GoalReadDTO>	Metoda sloužící k navrácení seznamu všech entit, které jsou v databázi.
findByIds	List<Goal>	Metoda sloužící k navrácení seznamu příslušných entit na základě daného seznamu identifikátorů entit. Pokud entita s daným identifikátorem není nalezena, nebude přidána do výstupního seznamu.
getById	GoalReadDTO	Metoda sloužící k navrácení příslušné Read DTO na základě daného identifikátoru entity.
merge	Goal	Metoda přetěžuje základní implementaci rozhraní a slouží k aktualizaci entit na základě vstupních Update DTO. Pokud je jakékoliv pole prázdné - informace jsou nekompletní, aktualizace entity neproběhne. Parametry: <b>dto: GoalUpdateDTO -</b> Parametry: <b>user: UserReadDTO -</b>
toDTO	GoalReadDTO	Metoda přetěžuje základní implementaci rozhraní a slouží k mapování entit na příslušné Read DTO.
toEntity	Goal	Metoda přetěžuje základní implementaci rozhraní a slouží k mapování Create DTO na příslušnou entitu.
update	void	Metoda sloužící k aktualizaci entit na základě daných vstupních Update DTO. Parametry: <b>user: UserReadDTO -</b> Parametry: <b>id: int -</b> Parametry: <b>dto: GoalUpdateDTO -</b>



### 1.1.2.5 Class MealService

Název metody	Návratový typ	Popis
create	List<MealReadDTO>	Metoda sloužící k vytvoření entit na základě daných vstupních Create DTO. Parametry: <b>user: UserReadDTO</b> - Parametry: <b>dtos: MealCreateDTO</b> -
delete	void	Metoda sloužící k vymazání entity na základě daného identifikátoru
findAll	List<MealReadDTO>	Metoda sloužící k navrácení seznamu všech entit, které jsou v databázi.
findByIds	List<Meal>	Metoda sloužící k navrácení seznamu příslušných entit na základě daného seznamu identifikátorů entit. Pokud entita s daným identifikátorem není nalezena, nebude přidána do výstupního seznamu.
getById	MealReadDTO	Metoda sloužící k navrácení příslušné Read DTO na základě daného identifikátoru entity.
merge	Meal	Metoda přetěžuje základní implementaci rozhraní a slouží k aktualizaci entit na základě vstupních Update DTO. Pokud je jakékoliv pole prázdné - informace jsou nekompletní, aktualizace entity neproběhne. Parametry: <b>dto: MealUpdateDTO</b> - Parametry: <b>user: UserReadDTO</b> -
toDTO	MealReadDTO	Metoda přetěžuje základní implementaci rozhraní a slouží k mapování entit na příslušné Read DTO.
toEntity	Meal	Metoda přetěžuje základní implementaci rozhraní a slouží k mapování Create DTO na příslušnou entitu.
update	void	Metoda sloužící k aktualizaci entit na základě daných vstupních Update DTO. Parametry: <b>user: UserReadDTO</b> - Parametry: <b>id: int</b> - Parametry: <b>dto: MealUpdateDTO</b> -



### 1.1.2.6 Class MicronutrientService

Název metody	Návratový typ	Popis
create	List<MicronutrientReadDTO>	Metoda sloužící k vytvoření entit na základě daných vstupních Create DTO. Parametry: <b>user: UserReadDTO -</b> Parametry: <b>dtos: MicronutrientCreateDTO -</b>
delete	void	Metoda sloužící k vymazání entity na základě daného identifikátoru
findAll	List<MicronutrientReadDTO>	Metoda sloužící k navrácení seznamu všech entit, které jsou v databázi.
findByIds	List<Micronutrient>	Metoda sloužící k navrácení seznamu příslušných entit na základě daného seznamu identifikátorů entit. Pokud entita s daným identifikátorem není nalezena, nebude přidána do výstupního seznamu.
getById	MicronutrientReadDTO	Metoda sloužící k navrácení příslušné Read DTO na základě daného identifikátoru entity.
merge	Micronutrient	Metoda přetěžuje základní implementaci rozhraní a slouží k aktualizaci entit na základě vstupních Update DTO. Pokud je jakékoliv pole prázdné - informace jsou nekompletní, aktualizace entity neproběhne. Parametry: <b>dto: MicronutrientUpdateDTO -</b> Parametry: <b>user: UserReadDTO -</b>
toDTO	MicronutrientReadDTO	Metoda přetěžuje základní implementaci rozhraní a slouží k mapování entit na příslušné Read DTO.
toEntity	Micronutrient	Metoda přetěžuje základní implementaci rozhraní a slouží k mapování Create DTO na příslušnou entitu.
update	void	Metoda sloužící k aktualizaci entit na základě daných vstupních Update DTO. Parametry: <b>user: UserReadDTO -</b> Parametry: <b>id: int -</b> Parametry: <b>dto: MicronutrientUpdateDTO -</b>



### 1.1.2.7 Class **UserActivityService**

Název metody	Návratový typ	Popis
create	List<UserActivityReadDTO>	Metoda sloužící k vytvoření entit na základě daných vstupních Create DTO. Parametry: <b>user: UserReadDTO</b> - Parametry: <b>dtos: UserActivityCreatedDTO</b> -
delete	void	Metoda sloužící k vymazání entity na základě daného identifikátoru
findAll	List<UserActivityReadDTO>	Metoda sloužící k navrácení seznamu všech entit, které jsou v databázi.
findByIDs	List<UserActivity>	Metoda sloužící k navrácení seznamu příslušných entit na základě daného seznamu identifikátorů entit. Pokud entita s daným identifikátorem není nalezena, nebude přidána do výstupního seznamu.
getByID	UserActivityReadDTO	Metoda sloužící k navrácení příslušné Read DTO na základě daného identifikátoru entity.
getDaily	List<UserActivityReadDTO>	Metoda sloužící k nalezení všech aktivit, které daný uživatel v daný den vykonal. Parametry: <b>user: UserReadDTO</b> - Parametry: <b>day: Date</b> -
getHistory	List<ActivityReadDTO>	Metoda sloužící k nalezení veškerých aktivit, které kdy daný uživatel vykonal, seřazených dle četnosti. Parametry: <b>user: UserReadDTO</b> -
merge	UserActivity	Metoda přetěžuje základní implementaci rozhraní a slouží k aktualizaci entit na základě vstupních Update DTO. Pokud je jakékoliv pole prázdné - informace jsou nekompletní, aktualizace entity neproběhne. Parametry: <b>dto: UserActivityUpdateDTO</b> - Parametry: <b>user: UserReadDTO</b> -
toDTO	UserActivityReadDTO	Metoda přetěžuje základní implementaci rozhraní a slouží k mapování entit na příslušné Read DTO.
toDTO	ActivityReadDTO	Pomocná rozšiřující metoda sloužící ke konverzi Activity na ActivityReadDTO.



toEntity	UserActivity	Metoda přetěžuje základní implementaci rozhraní a slouží k mapování Create DTO na příslušnou entitu.
update	void	Metoda sloužící k aktualizaci entit na základě daných vstupních Update DTO. Parametry: <b>user: UserReadDTO</b> - Parametry: <b>id: int</b> - Parametry: <b>dto: UserActivityUpdateDTO</b> -

### 1.1.2.8 Class UserMealFoodService

Název metody	Návratový typ	Popis
create	List<UserMealFoodReadDTO>	Metoda sloužící k vytvoření entit na základě daných vstupních Create DTO. Parametry: <b>user: UserReadDTO</b> - Parametry: <b>dtos: UserMealFoodCreateDTO</b> -
delete	void	Metoda sloužící k vymazání entity na základě daného identifikátoru
findAll	List<UserMealFoodReadDTO>	Metoda sloužící k navrácení seznamu všech entit, které jsou v databázi.
findByIds	List<UserMealFood>	Metoda sloužící k navrácení seznamu příslušných entit na základě daného seznamu identifikátorů entit. Pokud entita s daným identifikátorem není nalezena, nebude přidána do výstupního seznamu.
getByID	UserMealFoodReadDTO	Metoda sloužící k navrácení příslušné Read DTO na základě daného identifikátoru entity.
getDaily	List<UserMealFoodReadDTO>	Metoda sloužící k nalezení všech jídel, které daný uživatel v daný den zkonsumoval. Parametry: <b>user: UserReadDTO</b> - Parametry: <b>day: Date</b> -
getHistory	List<FoodReadDTO>	Metoda sloužící k nalezení veškerých jídel, které kdy daný uživatel zkonsumoval, seřazených dle četnosti. Parametry: <b>user: UserReadDTO</b> -

merge	UserMealFood	Metoda přetěžuje základní implementaci rozhraní a slouží k aktualizaci entit na základě vstupních Update DTO. Pokud je jakékoliv pole prázdné - informace jsou nekompletní, aktualizace entity neproběhne. Parametry: <b>dto: UserMealFoodUpdateDTO -</b> Parametry: <b>user: UserReadDTO -</b>
toDTO	UserMealFoodReadDTO	Metoda přetěžuje základní implementaci rozhraní a slouží k mapování entit na příslušné Read DTO.
toDTO	MealReadDTO	Pomocná rozšiřující metoda sloužící ke konverzi Meal na MealReadDTO.
toDTO	FoodReadDTO	Pomocná rozšiřující metoda sloužící ke konverzi Food na FoodReadDTO.
toEntity	UserMealFood	Metoda přetěžuje základní implementaci rozhraní a slouží k mapování Create DTO na příslušnou entitu.
update	void	Metoda sloužící k aktualizaci entit na základě daných vstupních Update DTO. Parametry: <b>user: UserReadDTO -</b> Parametry: <b>id: int -</b> Parametry: <b>dto: UserMealFoodUpdateDTO -</b>

### 1.1.2.9 Class UserService

Název metody	Návratový typ	Popis
create	List<UserReadDTO>	Metoda sloužící k vytvoření entit na základě daných vstupních Create DTO. Parametry: <b>user: UserReadDTO -</b> Parametry: <b>dtos: UserCreateDTO -</b>
delete	void	Metoda sloužící k vymazání entity na základě daného identifikátoru
findAll	List<UserReadDTO>	Metoda sloužící k navrácení seznamu všech entit, které jsou v databázi.
findByIds	List<UserEntity>	Metoda sloužící k navrácení seznamu příslušných entit na základě daného seznamu identifikátorů entit. Pokud entita s daným identifikátorem není nalezena, nebude

		přidána do výstupního seznamu.
getByCode	void	Metoda, která slouží k získání informací ohledně aktuálně přihlášeného uživatele. Parametry: <b>code: String</b> -
getByID	UserReadDTO	Metoda sloužící k navrácení příslušné Read DTO na základě daného identifikátoru entity.
merge	User	Metoda přetěžuje základní implementaci rozhraní a slouží k aktualizaci entit na základě vstupních Update DTO. Pokud je jakékoliv pole prázdné - informace jsou nekompletní, aktualizace entity neproběhne. Parametry: <b>dto: UserUpdateDTO</b> - Parametry: <b>user: UserReadDTO</b> -
toDTO	UserReadDTO	Metoda přetěžuje základní implementaci rozhraní a slouží k mapování entit na příslušné Read DTO.
toDTO	GoalReadDTO	Pomocná rozšiřující metoda sloužící ke konverzi Goal na GoalReadDTO.
toEntity	User	Metoda přetěžuje základní implementaci rozhraní a slouží k mapování Create DTO na příslušnou entitu.
update	void	Metoda sloužící k aktualizaci entit na základě daných vstupních Update DTO. Parametry: <b>user: UserReadDTO</b> - Parametry: <b>id: int</b> - Parametry: <b>dto: UserUpdateDTO</b> -

### 1.1.2.10 Interface IService

Generické rozhraní pro implementaci CRUD úprav jednotlivých Entit.

Název atributu	Datový typ	Popis
PAGE_SIZE	Int	Statická veřejná proměnná obsahující počet vypisovaných položek pro požadavky, které mají mít jako výstup seznam položek.
repository	IRepository<IEntity>	
sort	Sort	
Název metody	Návratový typ	Popis
create	List<IReadDTO>	Metoda sloužící k vytvoření entit na základě daných vstupních Create DTO. Parametry: <b>user: UserReadDTO</b> - Parametry:



		<b>dtos: ICreateDTO -</b>
delete	void	Metoda sloužící k vymazání entity na základě daného identifikátoru
findAll	List<IReadDTO>	Metoda sloužící k navrácení seznamu všech entit, které jsou v databázi.
findByIDs	List<IEntity>	Metoda sloužící k navrácení seznamu příslušných entit na základě daného seznamu identifikátorů entit. Pokud entita s daným identifikátorem není nalezena, nebude přidána do výstupního seznamu.
getByID	IReadDTO	Metoda sloužící k navrácení příslušné Read DTO na základě daného identifikátoru entity.
merge	IEntity	Rozšiřující metoda sloužící k aktualizaci entit na základě vstupních Update DTO. Pokud je jakékoliv pole prázdné - informace jsou nekompletní, aktualizace entity neproběhne. Parametry: <b>dto: IUpdateDTO -</b> Parametry: <b>user: UserReadDTO -</b>
toDTO	IReadDTO	Rozšiřující metoda sloužící k mapování entit na příslušné Read DTO.
toEntity	IEntity	Rozšiřující metoda sloužící k mapování Create DTO na příslušnou entitu.
update	void	Metoda sloužící k aktualizaci entit na základě daných vstupních Update DTO. Parametry: <b>user: UserReadDTO -</b> Parametry: <b>id: int -</b> Parametry: <b>dto: IUpdateDTO -</b>

### 1.1.2.11 Interface IServiceImpl

Rozhraní implementuje metody IRepository a mapuje je na generické metody Repozitářů.

Název metody	Návratový typ	Popis
create	List<IReadDTO>	Metoda přetěžuje základní implementaci rozhraní a slouží k vytvoření entit na základě daných vstupních Create DTO. Parametry: <b>user: UserReadDTO -</b> Parametry: <b>dtos: ICreateDTO -</b>

delete	void	Metoda přetěžuje základní implementaci rozhraní a slouží k vymazání entity na základě daného identifikátoru
findAll	List<IReadDTO>	Metoda přetěžuje základní implementaci rozhraní a slouží k navrácení seznamu všech entit, které jsou v databázi.
findByIDs	List<IEntity>	Metoda přetěžuje základní implementaci rozhraní a slouží k navrácení seznamu příslušných entit na základě daného seznamu identifikátorů entit. Pokud entita s daným identifikátorem není nalezena, nebude přidána do výstupního seznamu.
getByID	IReadDTO	Metoda přetěžuje základní implementaci rozhraní a slouží k navrácení příslušné Read DTO na základě daného identifikátoru entity.
getEntityByID	IEntity	Pomocná metoda sloužící k získání entity na základě daného identifikátoru. Parametry: <b>id: int</b> -
merge	IEntity	Rozšiřující metoda sloužící k aktualizaci entit na základě vstupních Update DTO. Pokud je jakékoliv pole prázdné - informace jsou nekompletní, aktualizace entity neproběhne. Parametry: <b>dto: IUpdateDTO</b> - Parametry: <b>user: UserReadDTO</b> -
toDTO	IReadDTO	Rozšiřující metoda sloužící k mapování entit na příslušné Read DTO.
toEntity	IEntity	Rozšiřující metoda sloužící k mapování Create DTO na příslušnou entitu.
tryCatch	void	Pomocná šablonová metoda sloužící k provedení akce v repozitáři. Metoda zachytí jakoukoliv výjimku a změní ji na ResponseStatusException. Parametry: <b>block: IRepository</b> -
update	void	Metoda přetěžuje základní implementaci rozhraní a slouží k aktualizaci entit na základě daných vstupních Update DTO. Parametry: <b>user: UserReadDTO</b> - Parametry: <b>id: int</b> - Parametry: <b>dto: IUpdateDTO</b> -

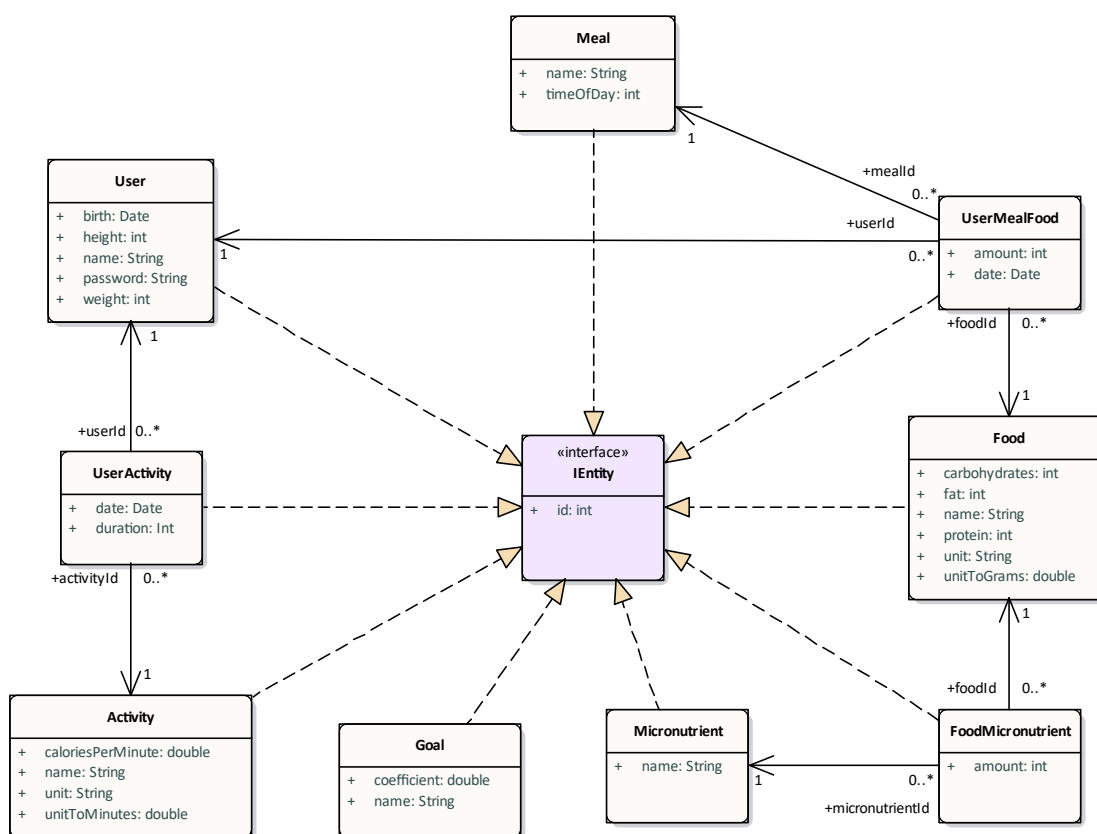


## 1.2 data

Tato vrstva obsahuje třídy, které umožňují načítání, úpravu a ukládání dat z databáze.

### 1.2.1 entity

Balíček obsahuje definici datových objektů, se kterými Services pracují. Tento balíček bude využíván v Business vrstvě pro ukládání do databáze.



Obrázek 5 - entity

#### 1.2.1.1 Class Activity

Název atributu	Datový typ	Popis
caloriesPerMinute	double	Kolik kalorií se spaluje při minutě vykování dané aktivity
name	String	Název aktivity
unit	String	Jednotka kterou se měří výkon aktivity
unitToMinutes	double	Převod dané jednotky na minuty výkonu aktivity

#### 1.2.1.2 Class Food

Název atributu	Datový typ	Popis
carbohydrates	int	Počet gramů sacharidů v jídle
fat	int	Počet gramů tuku v jídle
name	String	Název jídla
protein	int	Počet gramů bílkovin v jídle
unit	String	Jednotka na kterou se dělí jídlo, např. ks, gramy atd.
unitToGrams	double	Přepočet jednotky na gramy jídla

### 1.2.1.3 Class FoodMicronutrient

Název atributu	Datový typ	Popis
amount	int	Počet gramů mikroživin v jídle

### 1.2.1.4 Class Goal

Název atributu	Datový typ	Popis
coefficient	double	Násobitel požadované denní aktivity
name	String	Označení cíle

### 1.2.1.5 Class Meal

Název atributu	Datový typ	Popis
name	String	Název chodu, např. snídaně, oběd, svačina atd.
timeOfDay	int	V kolik hodin bylo jídlo konzumováno

### 1.2.1.6 Class Micronutrient

Název atributu	Datový typ	Popis
name	String	Název mikroživiny

### 1.2.1.7 Class User

Název atributu	Datový typ	Popis
birth	Date	
height	int	Výška uživatele v centimetrech
name	String	Uživatelské jméno
password	String	Šifrované přístupové heslo
weight	int	Váha uživatele v kilogramech

### 1.2.1.8 Class UserActivity

Název atributu	Datový typ	Popis
date	Date	Datum vykonávání aktivity
duration	Int	

### 1.2.1.9 Class UserMealFood

Název atributu	Datový typ	Popis
amount	int	Kolik jídla na 100g bylo zkonsumováno
date	Date	Který den bylo jídlo konzumováno

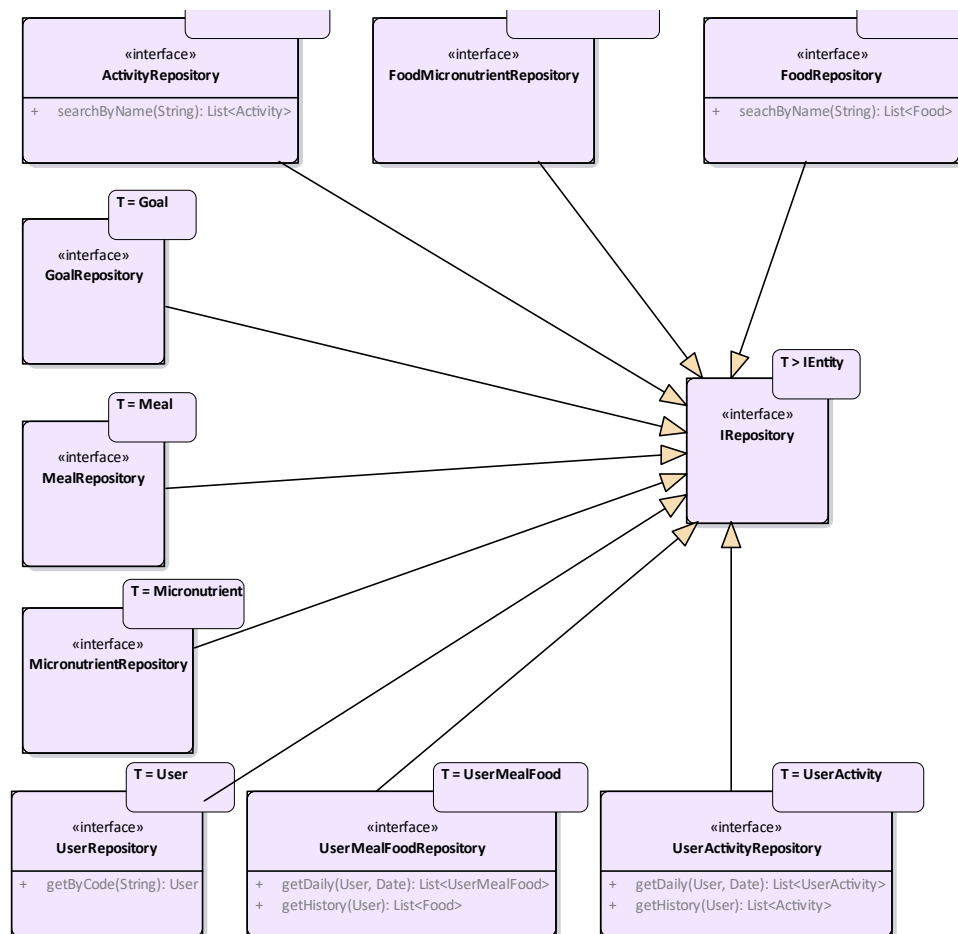
### 1.2.1.10 Interface IEntity

Základní model pro všechny Entity v balíčku, předává jim povinný atribut ID kterým odlišuje jednotlivé záznamy Entit a slouží jako identifikátor v databázových relacích.

Název atributu	Datový typ	Popis
id	int	

## 1.2.2 repository

Balíček obsahuje definici rozhraní, přes které bude Business vrstva ukládat, načítat a upravovat data. Implementace rozhraní v tomto balíčku bude vytvořena automaticky pomocí Hibernate a JPA.



Obrázek 6 - repository

### 1.2.2.1 Interface ActivityRepository

Název metody	Návratový typ	Popis
searchByName	List<Activity>	Umožňuje vyhledávat v seznamu aktivit pomocí částečné shody názvu. Používá SQL porovnávání LIKE ku vstupnímu parametru name. Parametry: <b>name: String</b> -

### 1.2.2.2 Interface FoodMicronutrientRepository

### 1.2.2.3 Interface FoodRepository

Název metody	Návratový typ	Popis
seachByName	List<Food>	Umožňuje vyhledávat v seznamu aktivit pomocí částečné shody názvu. Používá SQL porovnávání LIKE ku vstupnímu parametru name. Parametry: <b>name: String</b> -

### 1.2.2.4 Interface GoalRepository

### 1.2.2.5 Interface IRepository

Centrální repozitář sloužící jako rozhraní nad JPArepository Springframework balíčku Hibernate. Poskytuje jednotný přístup k řadě perzistentních databázových modelů.

### 1.2.2.6 Interface MicronutrientRepository

### 1.2.2.7 Interface UserActivityRepository

Název metody	Návratový typ	Popis
getDaily	List<UserActivity>	Vrací všechny aktivity které uživatel provozoval v zadaný den Parametry: <b>user: User</b> - Parametry: <b>day: Date</b> -
getHistory	List<Activity>	Vrací všechny aktivity které uživatel za používání aplikace vykonával, seřazené sestupně podle četnosti Parametry: <b>user: User</b> -



### 1.2.2.8 Interface UserMealFoodRepository

Název metody	Návratový typ	Popis
getDaily	List<UserMealFood>	Vrací všechna jídla které uživatel konzumoval v zadaný den Parametry: <b>user: User</b> - Parametry: <b>day: Date</b> -
getHistory	List<Food>	Vrací všechna jídla které uživatel za používání aplikace zkonzumoval, seřazené sestupně podle četnosti Parametry: <b>user: User</b> -

### 1.2.2.9 Interface UserRepository

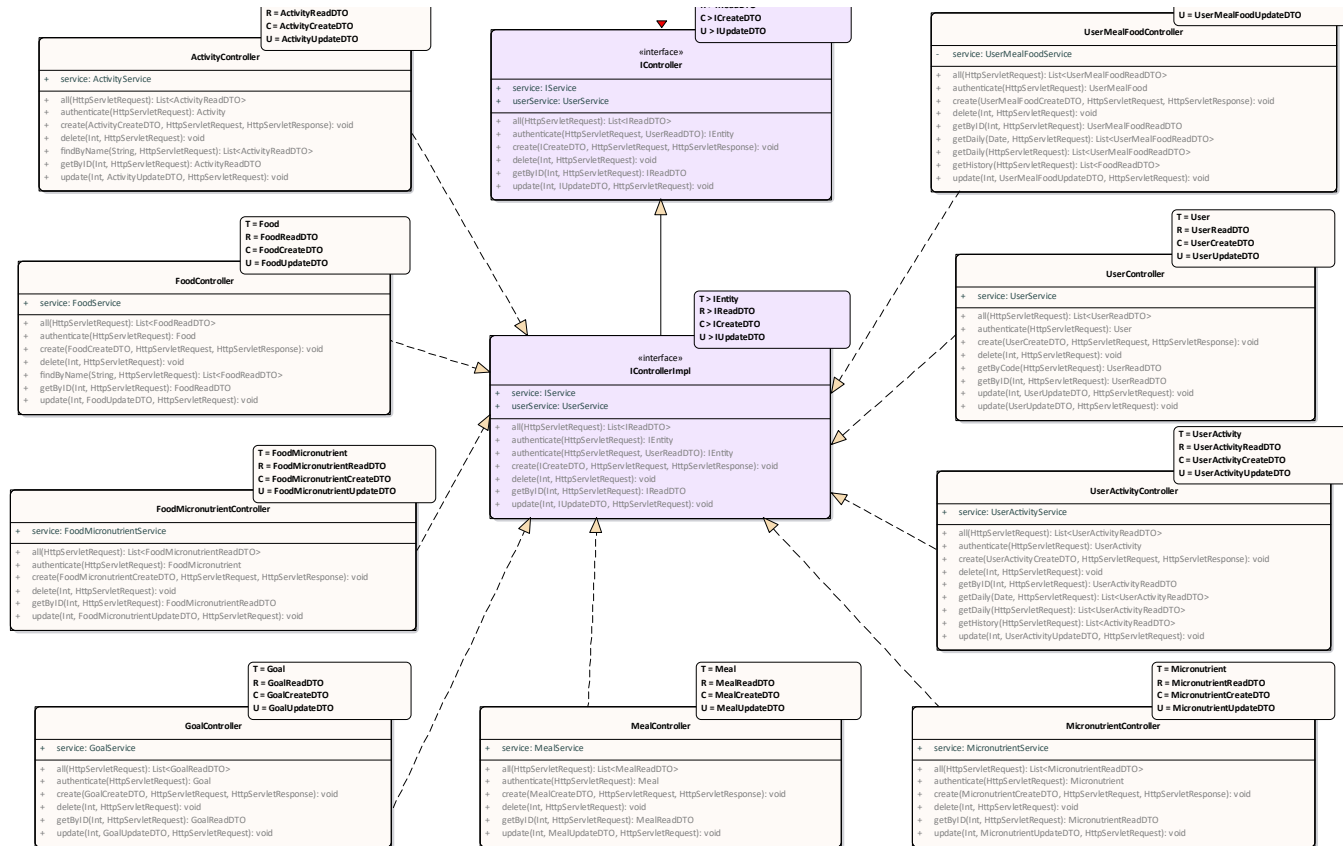
Název metody	Návratový typ	Popis
getByCode	User	Umožňuje vyhledávat v seznamu uživatelů pomocí jeho unikátního kódu. Využívá se při autentizaci klienta při zpracování požadavků na REST API. Parametry: <b>code: String</b> -

## 1.3 presentation

Prezentační vrstva Rest API, která reaguje na HTTP požadavky a odesílá HTTP odpovědi. Tyto požadavky jsou pomocí Spring anotací převedeny do metod jednotlivých Controllerů, které následně komunikují se Services v Business vrstvě.

### 1.3.1 controller

Balíček je určen pro třídy, které zpracují HTTP požadavky a transformují je na volání metod Business vrstvy. HTTP dělí podle standardizované REST hlavičky na jednotlivé CRUD metody pro práci s entitami v databázi, mimo to i specializované funkce (například vyhledávání).



Obrázek 7 - controller

### 1.3.1.1 Class ActivityController

Název atributu	Datový typ	Popis
service	ActivityService	Přetěhuje interface servisu konkrétní implementací ActivityService.
Název metody	Návratový typ	Popis
all	List<ActivityReadDTO>	Metoda sloužící k nalezení veškerých entit v databázi. Parametry: <b>request: HttpServletRequest</b>
authenticate	Activity	Metoda sloužící k autorizaci uživatele na základě HTTP požadavku. Parametry: <b>request: HttpServletRequest</b>
create	void	Metoda sloužící k vytvoření entity na základě zadaných vstupních DTO. Parametry: <b>dto: ActivityCreateDTO</b> Parametry: <b>request: HttpServletRequest</b>

		<b>response: HttpServletResponse -</b>
delete	void	Metoda sloužící k vymazání entity z databáze na základě zadaných vstupních DTO. Parametry: <b>id: Int -</b> Parametry: <b>request: HttpServletRequest -</b>
findByName	List<ActivityReadDTO>	Metoda, která zobrazí seznam aktivit na základě částečné shody v názvu aktivit. Parametry: <b>name: String -</b> Parametry: <b>request: HttpServletRequest -</b>
getByID	ActivityReadDTO	Metoda sloužící k navrácení příslušné entity na základě daného identifikátoru. Parametry: <b>id: Int -</b> Parametry: <b>request: HttpServletRequest -</b>
update	void	Metoda sloužící k aktualizaci entity v databázi na základě zadaných vstupních DTO. Parametry: <b>id: Int -</b> Parametry: <b>dto: ActivityUpdateDTO -</b> Parametry: <b>request: HttpServletRequest -</b>

### 1.3.1.2 Class FoodController

Název atributu	Datový typ	Popis
service	FoodService	
Název metody	Návratový typ	Popis
all	List<FoodReadDTO>	Metoda sloužící k nalezení veškerých entit v databázi. Parametry: <b>request: HttpServletRequest -</b>
authenticate	Food	Metoda sloužící k autorizaci uživatele na základě HTTP požadavku. Parametry: <b>request: HttpServletRequest -</b>
create	void	Metoda sloužící k vytvoření entity na základě zadaných vstupních DTO. Parametry: <b>dto: FoodCreateDTO -</b>



		Parametry: <b>request: HttpServletRequest -</b> Parametry: <b>response: HttpServletResponse -</b>
delete	void	Metoda sloužící k vymazání entity z databáze na základě zadaných vstupních DTO. Parametry: <b>id: Int -</b> Parametry: <b>request: HttpServletRequest -</b>
findByName	List<FoodReadDTO>	Metoda, která zobrazí jídel na základě částečné shody v názvu jídel. Parametry: <b>name: String -</b> Parametry: <b>request: HttpServletRequest -</b>
getByID	FoodReadDTO	Metoda sloužící k navrácení příslušné entity na základě daného identifikátoru. Parametry: <b>id: Int -</b> Parametry: <b>request: HttpServletRequest -</b>
update	void	Metoda sloužící k aktualizaci entity v databázi na základě zadaných vstupních DTO. Parametry: <b>id: Int -</b> Parametry: <b>dto: FoodUpdateDTO -</b> Parametry: <b>request: HttpServletRequest -</b>

### 1.3.1.3 Class FoodMicronutrientController

Název atributu	Datový typ	Popis
service	FoodMicronutrientService	
Název metody	Návratový typ	Popis
all	List<FoodMicronutrientReadDTO>	Metoda sloužící k nalezení veškerých entit v databázi. Parametry: <b>request: HttpServletRequest -</b>
authenticate	FoodMicronutrient	Metoda sloužící k autorizaci uživatele na základě HTTP požadavku. Parametry: <b>request: HttpServletRequest -</b>



create	void	Metoda sloužící k vytvoření entity na základě zadaných vstupních DTO. Parametry: <b>dto: FoodMicronutrientCreateDTO -</b> Parametry: <b>request: HttpServletRequest -</b> Parametry: <b>response: HttpServletResponse -</b>
delete	void	Metoda sloužící k vymazání entity z databáze na základě zadaných vstupních DTO. Parametry: <b>id: Int -</b> Parametry: <b>request: HttpServletRequest -</b>
getByID	FoodMicronutrientReadDTO	Metoda sloužící k navrácení příslušné entity na základě daného identifikátoru. Parametry: <b>id: Int -</b> Parametry: <b>request: HttpServletRequest -</b>
update	void	Metoda sloužící k aktualizaci entity v databázi na základě zadaných vstupních DTO. Parametry: <b>id: Int -</b> Parametry: <b>dto: FoodMicronutrientUpdateDTO -</b> Parametry: <b>request: HttpServletRequest -</b>

### 1.3.1.4 Class GoalController

Název atributu	Datový typ	Popis
service	GoalService	
Název metody	Návratový typ	Popis
all	List<GoalReadDTO>	Metoda přetěžuje základní implementaci rozhraní a slouží k nalezení veškerých entit v databázi. Parametry: <b>request: HttpServletRequest -</b>
authenticate	Goal	Metoda sloužící k autorizaci uživatele na základě HTTP požadavku. Parametry: <b>request: HttpServletRequest -</b>
create	void	Metoda sloužící k vytvoření entity na základě zadaných vstupních DTO. Parametry:

		<b>dto: GoalCreateDTO -</b> Parametry: <b>request: HttpServletRequest -</b> Parametry: <b>response: HttpServletResponse -</b>
delete	void	Metoda sloužící k vymazání entity z databáze na základě zadaných vstupních DTO. Parametry: <b>id: Int -</b> Parametry: <b>request: HttpServletRequest -</b>
getByID	GoalReadDTO	Metoda sloužící k navrácení příslušné entity na základě daného identifikátoru. Parametry: <b>id: Int -</b> Parametry: <b>request: HttpServletRequest -</b>
update	void	Metoda sloužící k aktualizaci entity v databázi na základě zadaných vstupních DTO. Parametry: <b>id: Int -</b> Parametry: <b>dto: GoalUpdateDTO -</b> Parametry: <b>request: HttpServletRequest -</b>

### 1.3.1.5 Class MealController

Název atributu	Datový typ	Popis
service	MealService	
Název metody	Návratový typ	Popis
all	List<MealReadDTO>	Metoda sloužící k nalezení veškerých entit v databázi. Parametry: <b>request: HttpServletRequest -</b>
authenticate	Meal	Metoda sloužící k autorizaci uživatele na základě HTTP požadavku. Parametry: <b>request: HttpServletRequest -</b>
create	void	Metoda sloužící k vytvoření entity na základě zadaných vstupních DTO. Parametry: <b>dto: MealCreateDTO -</b> Parametry: <b>request: HttpServletRequest -</b> Parametry:



		<b>response: HttpServletResponse -</b>
delete	void	Metoda sloužící k vymazání entity z databáze na základě zadaných vstupních DTO. Parametry: <b>id: Int -</b> Parametry: <b>request: HttpServletRequest -</b>
getByID	MealReadDTO	Metoda sloužící k navrácení příslušné entity na základě daného identifikátoru. Parametry: <b>id: Int -</b> Parametry: <b>request: HttpServletRequest -</b>
update	void	Metoda sloužící k aktualizaci entity v databázi na základě zadaných vstupních DTO. Parametry: <b>id: Int -</b> Parametry: <b>dto: MealUpdateDTO -</b> Parametry: <b>request: HttpServletRequest -</b>

### 1.3.1.6 Class MicronutrientController

Název atributu	Datový typ	Popis
service	MicronutrientService	
Název metody	Návratový typ	Popis
all	List<MicronutrientReadDTO>	Metoda sloužící k nalezení veškerých entit v databázi. Parametry: <b>request: HttpServletRequest -</b>
authenticate	Micronutrient	Metoda sloužící k autorizaci uživatele na základě HTTP požadavku. Parametry: <b>request: HttpServletRequest -</b>
create	void	Metoda sloužící k vytvoření entity na základě zadaných vstupních DTO. Parametry: <b>dto: MicronutrientCreateDTO -</b> Parametry: <b>request: HttpServletRequest -</b> Parametry: <b>response: HttpServletResponse -</b>
delete	void	Metoda sloužící k vymazání entity z databáze na základě zadaných vstupních DTO.



		Parametry: <b>id: Int</b> - Parametry: <b>request: HttpServletRequest</b> -
getByID	MicronutrientReadDT O	Metoda sloužící k navrácení příslušné entity na základě daného identifikátoru. Parametry: <b>id: Int</b> - Parametry: <b>request: HttpServletRequest</b> -
update	void	Metoda sloužící k aktualizaci entity v databázi na základě zadaných vstupních DTO. Parametry: <b>id: Int</b> - Parametry: <b>dto: MicronutrientUpdateDTO</b> - Parametry: <b>request: HttpServletRequest</b> -

### 1.3.1.7 Class **UserActivityController**

Název atributu	Datový typ	Popis
service	UserActivityService	
Název metody	Návratový typ	Popis
all	List<UserActivityRead DTO>	Metoda sloužící k nalezení veškerých entit v databázi. Parametry: <b>request: HttpServletRequest</b> -
authenticate	UserActivity	Metoda sloužící k autorizaci uživatele na základě HTTP požadavku. Parametry: <b>request: HttpServletRequest</b> -
create	void	Metoda sloužící k vytvoření entity na základě zadaných vstupních DTO. Parametry: <b>dto: UserActivityCreateDTO</b> - Parametry: <b>request: HttpServletRequest</b> - Parametry: <b>response: HttpServletResponse</b> -
delete	void	Metoda sloužící k vymazání entity z databáze na základě zadaných vstupních DTO. Parametry: <b>id: Int</b> - Parametry: <b>request: HttpServletRequest</b> -



getByID	UserActivityReadDTO	Metoda sloužící k navrácení příslušné entity na základě daného identifikátoru. Parametry: <b>id: Int</b> - Parametry: <b>request: HttpServletRequest</b> -
getDaily	List<UserActivityReadDTO>	Metoda sloužící k nalezení všech aktivit, které daný uživatel v daný den vykonal. Parametry: <b>day: Date</b> - Parametry: <b>request: HttpServletRequest</b> -
getDaily	List<UserActivityReadDTO>	Metoda sloužící k nalezení všech aktivit, které daný uživatel dnes vykonal. Parametry: <b>request: HttpServletRequest</b> -
getHistory	List<ActivityReadDTO>	Metoda sloužící k nalezení veškerých aktivit, které kdy daný uživatel vykonal, seřazených dle četnosti. Parametry: <b>request: HttpServletRequest</b> -
update	void	Metoda sloužící k aktualizaci entity v databázi na základě zadaných vstupních DTO. Parametry: <b>id: Int</b> - Parametry: <b>dto: UserActivityUpdateDTO</b> - Parametry: <b>request: HttpServletRequest</b> -

### 1.3.1.8 Class UserController

Název atributu	Datový typ	Popis
service	UserService	
Název metody	Návratový typ	Popis
all	List<UserReadDTO>	Metoda sloužící k nalezení veškerých entit v databázi. Parametry: <b>request: HttpServletRequest</b> -
authenticate	User	Metoda sloužící k autorizaci uživatele na základě HTTP požadavku. Parametry: <b>request: HttpServletRequest</b> -
create	void	Metoda sloužící k vytvoření entity na základě zadaných vstupních DTO.



		Parametry: <b>dto: UserCreateDTO -</b> Parametry: <b>request: HttpServletRequest -</b> Parametry: <b>response: HttpServletResponse -</b>
delete	void	Metoda přetěžuje základní implementaci rozhraní. Metoda zakazuje mazání entit z databáze na základě jejich identifikátoru. Při pokusu o takové vymazání entity metoda vyhodí výjimku. Parametry: <b>id: Int -</b> Parametry: <b>request: HttpServletRequest -</b>
getByCode	UserReadDTO	Metoda, která slouží k získání informací ohledně aktuálně přihlášeného uživatele. Parametry: <b>request: HttpServletRequest -</b>
getByID	UserReadDTO	Metoda sloužící k navrácení příslušné entity na základě daného identifikátoru. Parametry: <b>id: Int -</b> Parametry: <b>request: HttpServletRequest -</b>
update	void	Metoda přetěžuje základní implementaci rozhraní. Metoda zakazuje aktualizaci informací o uživateli na základě identifikátoru. Při pokusu o takovou aktualizaci informací metoda vyhodí výjimku. Parametry: <b>id: Int -</b> Parametry: <b>dto: UserUpdateDTO -</b> Parametry: <b>request: HttpServletRequest -</b>
update	void	Metoda sloužící k aktualizaci informací o aktuálně přihlášeném uživateli na základě daných vstupních DTO. Parametry: <b>dto: UserUpdateDTO -</b> Parametry: <b>request: HttpServletRequest -</b>

### 1.3.1.9 Class UserMealFoodController

Název atributu	Datový typ	Popis
service	UserMealFoodService	



Název metody	Návratový typ	Popis
all	List<UserMealFoodReadDTO>	Metoda sloužící k nalezení veškerých entit v databázi. Parametry: <b>request: HttpServletRequest -</b>
authenticate	UserMealFood	Metoda sloužící k autorizaci uživatele na základě HTTP požadavku. Parametry: <b>request: HttpServletRequest -</b>
create	void	Metoda sloužící k vytvoření entity na základě zadaných vstupních DTO. Parametry: <b>dto: UserMealFoodCreateDTO -</b> Parametry: <b>request: HttpServletRequest -</b> Parametry: <b>response: HttpServletResponse -</b>
delete	void	Metoda sloužící k vymazání entity z databáze na základě zadaných vstupních DTO. Parametry: <b>id: Int -</b> Parametry: <b>request: HttpServletRequest -</b>
getByID	UserMealFoodReadDTO	Metoda sloužící k navrácení příslušné entity na základě daného identifikátoru. Parametry: <b>id: Int -</b> Parametry: <b>request: HttpServletRequest -</b>
getDaily	List<UserMealFoodReadDTO>	Metoda sloužící k nalezení všech jídel, které daný uživatel v daný den zkonsumoval. Parametry: <b>day: Date -</b> Parametry: <b>request: HttpServletRequest -</b>
getDaily	List<UserMealFoodReadDTO>	Metoda sloužící k nalezení všech jídel, které daný uživatel dnes zkonsumoval. Parametry: <b>request: HttpServletRequest -</b>
getHistory	List<FoodReadDTO>	Metoda sloužící k nalezení veškerých jídel, které kdy daný uživatel zkonsumoval, seřazených dle četnosti. Parametry: <b>request: HttpServletRequest -</b>
update	void	Metoda sloužící k aktualizaci entity v databázi na základě zadaných vstupních DTO.



		Parametry: <b>id: Int</b> - Parametry: <b>dto: UserMealFoodUpdateDTO</b> - Parametry: <b>request: HttpServletRequest</b> -
--	--	---

### 1.3.1.10 Interface IController

Rozhraní definuje interface pro CRUD úpravy entit a mapuje je na implementaci Servisů.

Název atributu	Datový typ	Popis
service	IService	
userService	UserService	
Název metody	Návratový typ	Popis
all	List<IReadDTO>	Metoda sloužící k nalezení veškerých entit v databázi. Parametry: <b>request: HttpServletRequest</b> -
authenticate	IEntity	Metoda sloužící k autorizaci uživatele na základě HTTP požadavku. Parametry: <b>request: HttpServletRequest</b> - Parametry: <b>block: UserReadDTO</b> -
create	void	Metoda sloužící k vytvoření entity na základě zadaných vstupních DTO. Parametry: <b>dto: ICreateDTO</b> - Parametry: <b>request: HttpServletRequest</b> - Parametry: <b>response: HttpServletResponse</b> -
delete	void	Metoda sloužící k vymazání entity z databáze na základě zadaných vstupních DTO. Parametry: <b>id: Int</b> - Parametry: <b>request: HttpServletRequest</b> -
getByID	IReadDTO	Metoda sloužící k navrácení příslušné entity na základě daného identifikátoru. Parametry: <b>id: Int</b> - Parametry: <b>request: HttpServletRequest</b> -
update	void	Metoda sloužící k aktualizaci entity v databázi na základě zadaných vstupních DTO. Parametry:





		<b>id: Int</b> - Parametry: <b>dto: IUpdateDTO</b> - Parametry: <b>request: HttpServletRequest</b> -
--	--	--

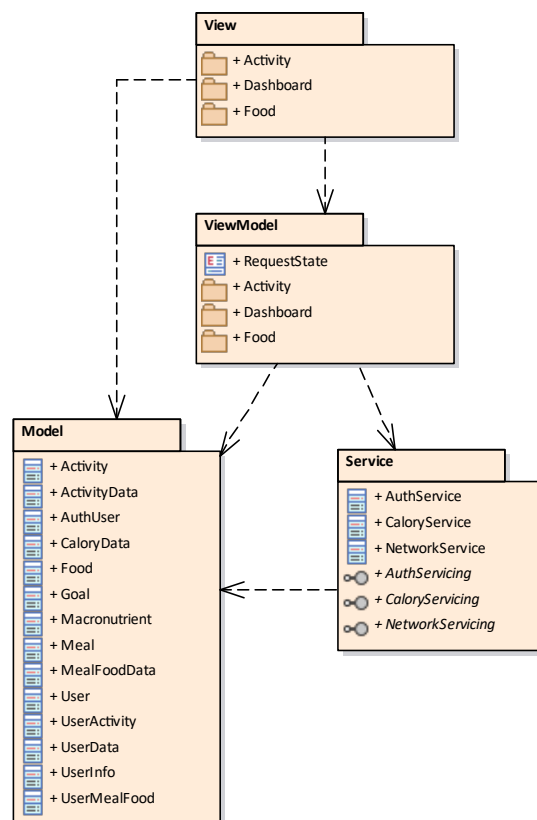
### 1.3.1.11 Interface IControllerImpl

Rozhraní implementuje metody IControlleru a mapuje je na HTTP požadavky. Každému zpracování předchází autentizace uživatele: ověření kodu z HTTP hlavičky požadavku.

Název atributu	Datový typ	Popis
service	IService	
userService	UserService	
Název metody	Návratový typ	Popis
all	List<IReadDTO>	Metoda přetěžuje základní implementaci rozhraní a slouží k nalezení veškerých entit v databázi. Parametry: <b>request: HttpServletRequest</b> -
authenticate	IEntity	Metoda přetěžuje základní implementaci rozhraní a slouží k autorizaci uživatele na základě HTTP požadavku. Parametry: <b>request: HttpServletRequest</b> -
authenticate	IEntity	Metoda sloužící k autorizaci uživatele na základě HTTP požadavku. Parametry: <b>request: HttpServletRequest</b> - Parametry: <b>block: UserReadDTO</b> -
create	void	Metoda přetěžuje základní implementaci rozhraní a slouží k vytvoření entity na základě zadaných vstupních DTO. Parametry: <b>dto: ICreateDTO</b> - Parametry: <b>request: HttpServletRequest</b> - Parametry: <b>response: HttpServletResponse</b> -
delete	void	Metoda přetěžuje základní implementaci rozhraní a slouží k vymazání entity z databáze na základě zadaných vstupních DTO. Parametry: <b>id: Int</b> - Parametry: <b>request: HttpServletRequest</b> -
getByID	IReadDTO	Metoda přetěžuje základní implementaci rozhraní a slouží k navrácení příslušné entity na základě daného identifikátoru. Parametry:

		<b>id: Int -</b> Parametry: <b>request: HttpServletRequest -</b>
update	void	Metoda přetěžuje základní implementaci rozhraní a slouží k aktualizaci entity v databázi na základě zadaných vstupních DTO. Parametry: <b>id: Int -</b> Parametry: <b>dto: IUpdateDTO -</b> Parametry: <b>request: HttpServletRequest -</b>

## 2. Návrh logické architektury (Aplikace)

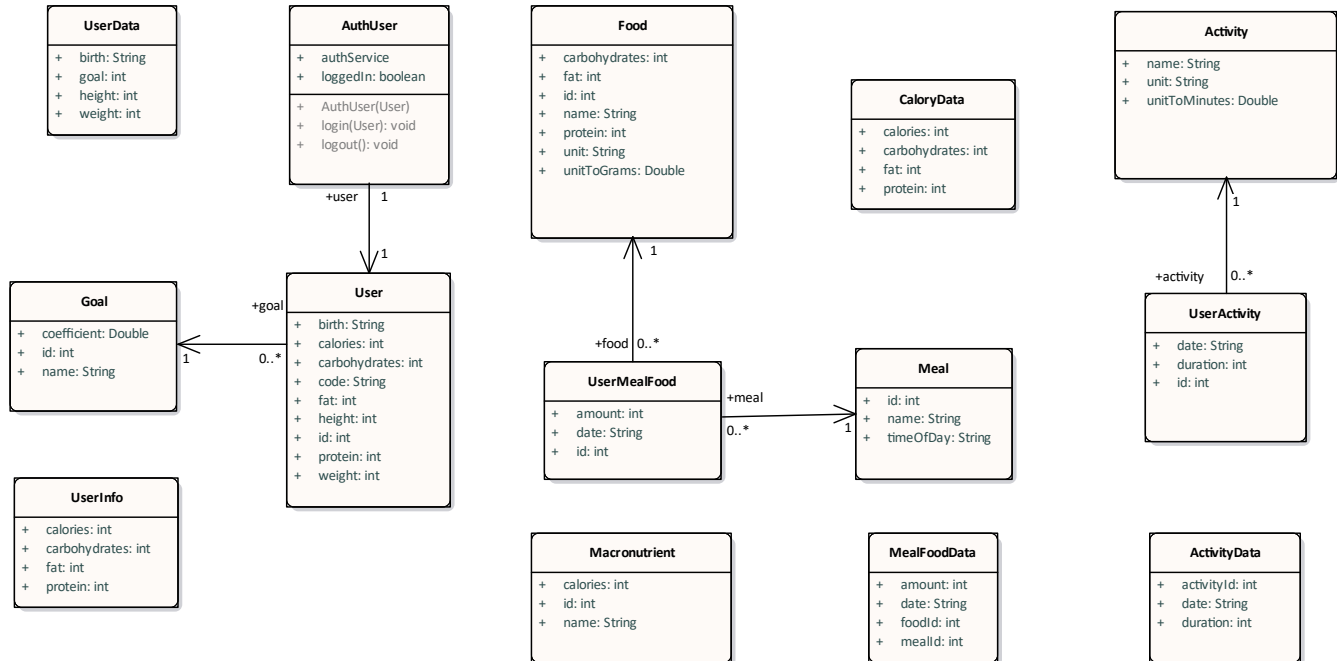


Obrázek 8 - Návrh logické architektury (Aplikace)

### 2.1 Model

Balíček obsahuje datové třídy, které slouží jako rozhraní pro prezentační vrstvu, a které jsou načítány z Rest API pomocí

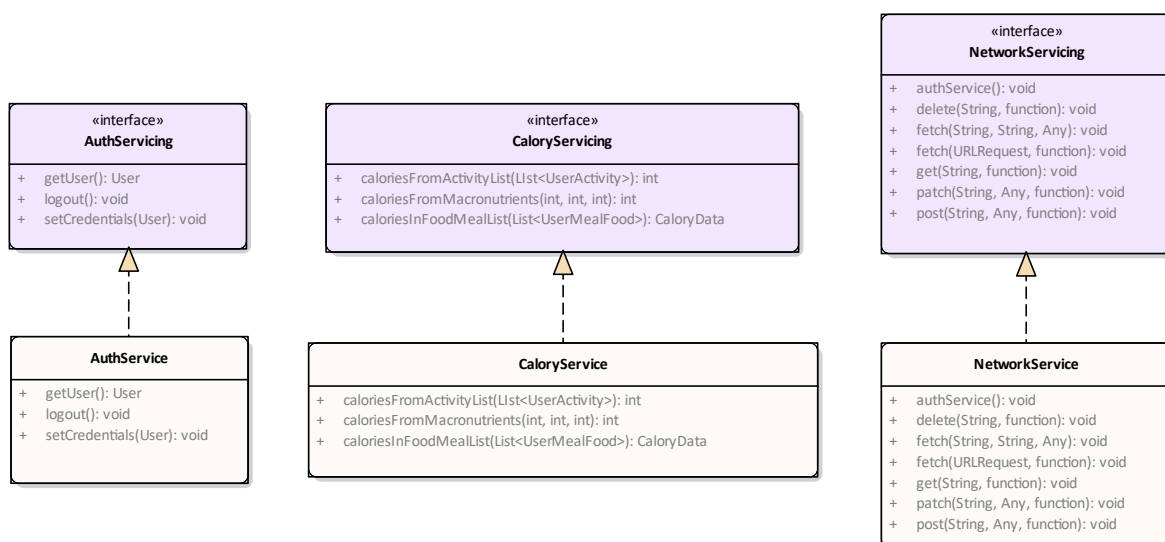
Services. Jedná se o první část **datové-business** vrstvy.



Obrázek 9 - Model

## 2.2 Service

Balíček je určen pro rozhraní, která slouží ke komunikaci s Rest API a správě přihlášeného uživatele. Jednotlivé implementace těchto rozhraní budou také v tomto balíčku. Jedná se o druhou část **datové-business** vrstvy.



Obrázek 10 - Service

### 2.2.1 Interface AuthServiceing

Interface identifikuje servis, ktorý je schopný ukládať a načítavať momentálne prihláseného užívateľa.

Název metody	Návratový typ	Popis
getUser	User	
logout	void	Odstraňuje informácie o užívateľovi z databázy.
setCredentials	void	Uloží informácie o užívateľovi, ktorý je momentálne prihlásený v databáze. Parametry: <b>user: User</b> -

### 2.2.2 Interface CaloryServicing

Identifikuje servis, ktorý kalkuluje väzby medzi mikroživinami, kalóriami, jedlom a aktivitami.

Název metody	Návratový typ	Popis
caloriesFromActivityList	int	Funkcia, ktorá vykonáva výpočty spálených kalórií. Parametry: <b>activityList: List&lt;UserActivity&gt;</b> -
caloriesFromMacronutrients	int	Funkcia, ktorá vykonáva výpočty danými kalórií. Parametry: <b>protein: int</b> - Parametry: <b>carbohydrates: int</b> - Parametry: <b>fat: int</b> -
caloriesInFoodMealList	CaloryData	Funkcia, ktorá vykonáva výpočty prijatých kalórií. Parametry: <b>foodMealList: List&lt;UserMealFood&gt;</b> -

### 2.2.3 Interface NetworkServicing

Interface spracúvajúci sieťové pripojenie.

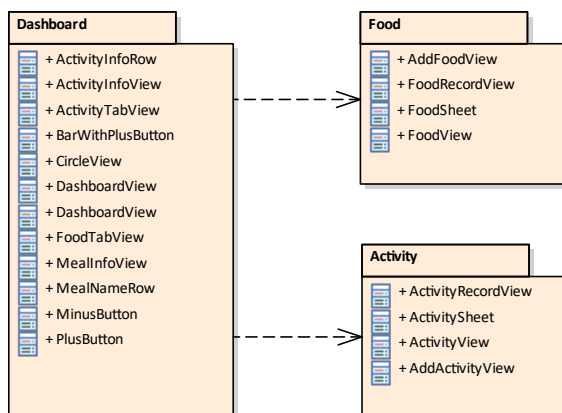
Název metody	Návratový typ	Popis
authService	void	
delete	void	Parametry: <b>url: String</b> - Parametry: <b>completion: function</b> -
fetch	void	Parametry: <b>url: String</b> - Parametry: <b>method: String</b> - Parametry:



		<b>body: Any -</b>
fetch	void	Parametry: <b>urlRequest: URLRequest -</b> Parametry: <b>completion: function -</b>
get	void	Parametry: <b>url: String -</b> Parametry: <b>completion: function -</b>
patch	void	Parametry: <b>url: String -</b> Parametry: <b>body: Any -</b> Parametry: <b>completion: function -</b>
post	void	Parametry: <b>url: String -</b> Parametry: <b>body: Any -</b> Parametry: <b>completion: function -</b>

## 2.3 View

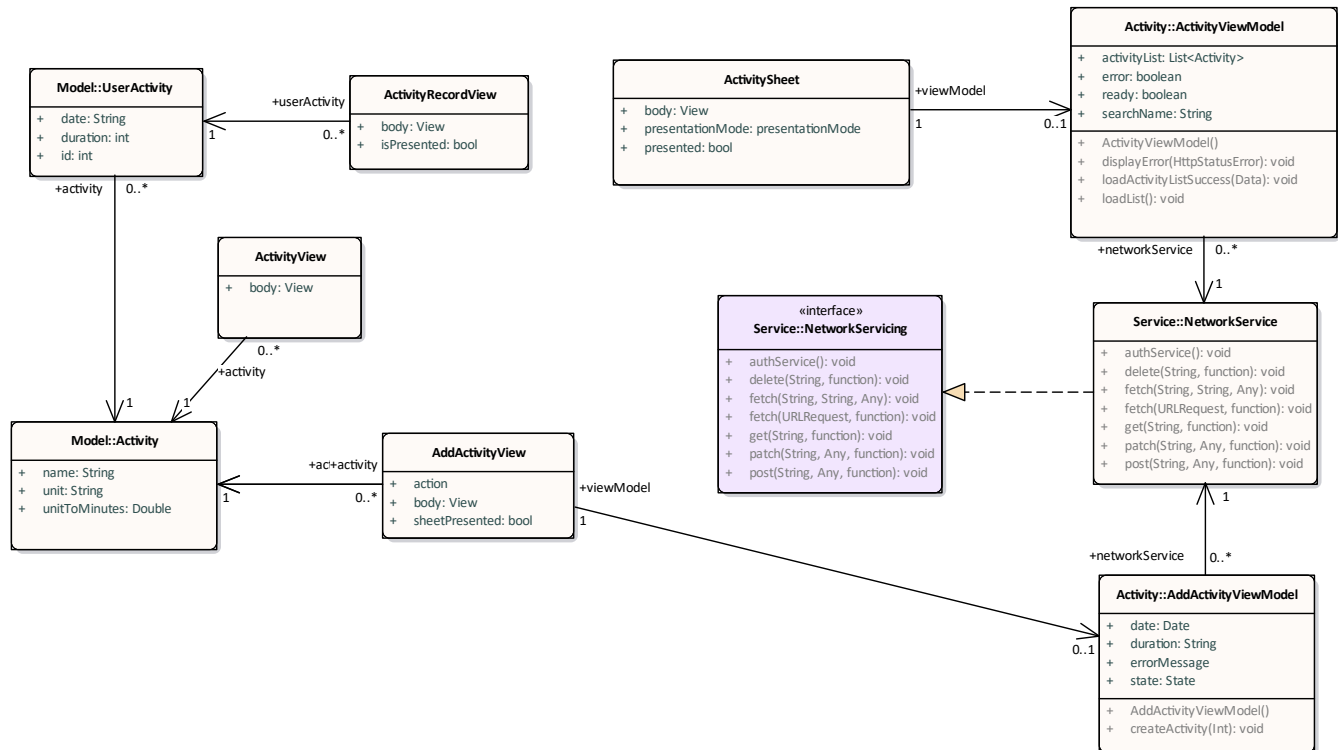
Balíček je určen pro třídy, které zobrazují jednotlivá data z ViewModelu uživateli pomocí grafického rozhraní. Jedná se o jednu ze dvou částí **prezentační** vrstvy.



Obrázek 11 - View

### 2.3.1 Activity

Balíček obsahuje views týkající se přidávání, úpravy a hledání aktivit.



Obrázek 12 - Activity

#### 2.3.1.1 Class ActivityRecordView

Trieda zobrazuje jednotlivu aktivitu uživatele a jej dĺžku.

Název atributu	Datový typ	Popis
body	View	Telo view.
isPresented	bool	

#### 2.3.1.2 Class ActivitySheet

Trieda zobrazí dialogové okno pre hľadanie aktivít.

Název atributu	Datový typ	Popis
body	View	Telo view
presentationMode	presentationMode	
presented	bool	

#### 2.3.1.3 Class ActivityView

Trieda zobrazí jednotlivú aktivitu z množstva aktivít.

Název atributu	Datový typ	Popis
body	View	



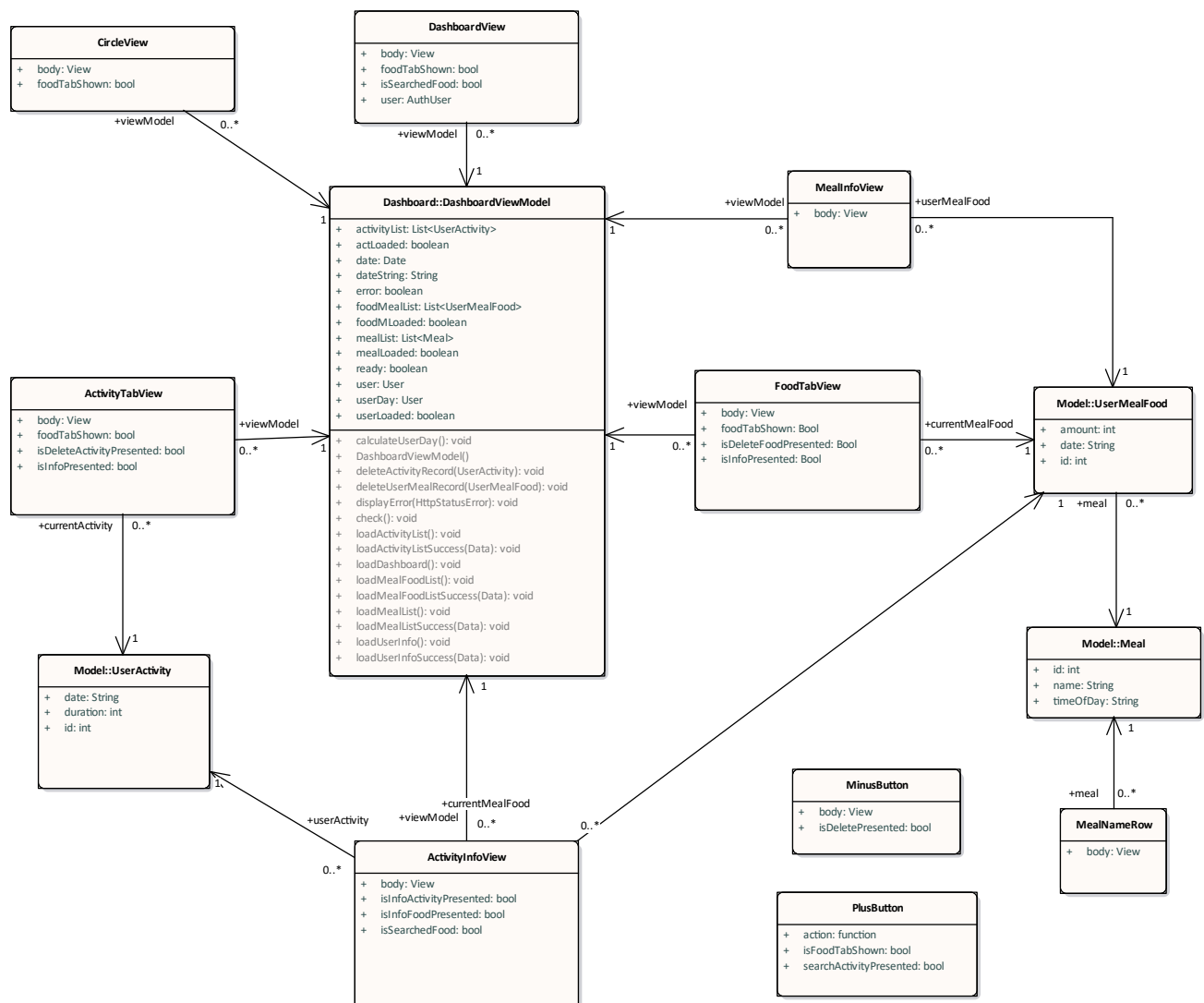
### 2.3.1.4 Class AddActivityView

Trieda zobrazí formulár pre pridanie aktivity

Název atributu	Datový typ	Popis
action		
body	View	
sheetPresented	bool	

### 2.3.2 Dashboard

Balíček obsahuje views týkajúci sa denného prehľadu.



Obrázek 13 - Dashboard



### 2.3.2.1 Class ActivityInfoRow

Trieda sprostredkúva zobrazenie jednotlivých podrobných informácií o jedle/aktivite.

Název atributu	Datový typ	Popis
body	View	
description	String	
name	String	

### 2.3.2.2 Class ActivityInfoView

Trieda zobrazuje podrobné informácie o jednotlivých vykonaných aktivitách.

Název atributu	Datový typ	Popis
body	View	
isInfoActivityPresented	bool	
isInfoFoodPresented	bool	
isSearchedFood	bool	

### 2.3.2.3 Class ActivityTabView

Trieda zobrazuje tlačítka pre pridanie a odobranie aktivít. Zároveň zobrazuje denný prehľad aktivít.

Název atributu	Datový typ	Popis
body	View	
foodTabShown	bool	
isDeleteActivityPresented	bool	
isInfoPresented	bool	

### 2.3.2.4 Class CircleView

Trieda zobrazí prehľadové informácie o splnení denného kalorického cieľa a dialogové tlačítka **Jedla** a **Aktivity**.

Název atributu	Datový typ	Popis
body	View	
foodTabShown	bool	

### 2.3.2.5 Class DashboardView

Trieda zobrazí úvodné a hlavné dialógové okno.

Název atributu	Datový typ	Popis
body	View	
foodTabShown	bool	
isSearchedFood	bool	
user	AuthUser	

### 2.3.2.6 Class FoodTabView

Trieda zobrazuje tlačítka pre pridanie a odobranie jedál. Zároveň zobrazuje denný prehľad jedál.

Název atributu	Datový typ	Popis
body	View	
foodTabShown	Bool	
isDeleteFoodPresented	Bool	
isInfoPresented	Bool	





### 2.3.2.7 Class MealInfoView

Trieda zobrazí podrobne informácie o konkrétnej konzumácii jedla.

Název atributu	Datový typ	Popis
body	View	

### 2.3.2.8 Class MealNameRow

Trieda, ktorá prezentuje jednotlivé chody (ranajky, obed, večera apod.).

Název atributu	Datový typ	Popis
body	View	

### 2.3.2.9 Class MinusButton

Trieda zodpovedá za tlačítko, ktoré sprostredkúva odoberanie jedál/aktivít.

Název atributu	Datový typ	Popis
body	View	
isDeletePresented	bool	

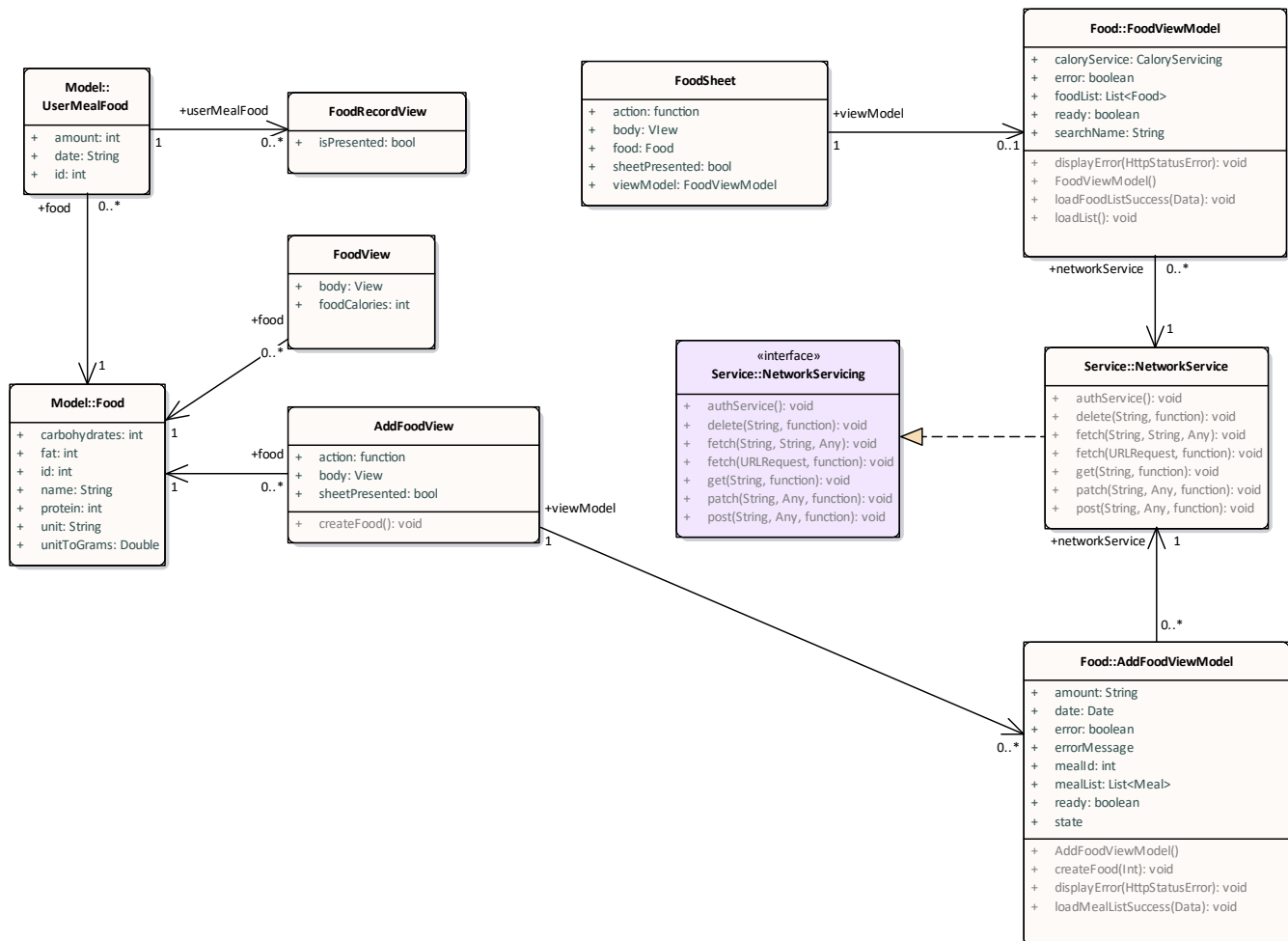
### 2.3.2.10 Class PlusButton

Trieda zodpovedá za tlačítko, ktoré sprostredkúva pridanie jedál/aktivít.

Název atributu	Datový typ	Popis
action	function	
isFoodTabShown	bool	
searchActivityPresented	bool	

## 2.3.3 Food

Balíček obsahuje obrazovky týkajúce sa pridávania, úpravy a hľadania jedel.



Obrázek 14 - Food

### 2.3.3.1 Class AddFoodView

Trieda zobrazuje formulár pre pridávanie jedál.

Název atributu	Datový typ	Popis
action	function	
body	View	
sheetPresented	bool	
Název metody	Návratový typ	Popis
createFood	void	

### 2.3.3.2 Class FoodRecordView

Trieda zobrazuje podrobné informácie o jedle.

Název atributu	Datový typ	Popis
isPresented	bool	

### 2.3.3.3 Class FoodSheet

Trieda zobrazuje dialógové okno pre vyhľadávanie jedál.

Název atributu	Datový typ	Popis
action	function	
body	View	
food	Food	
sheetPresented	bool	
viewModel	FoodViewModel	

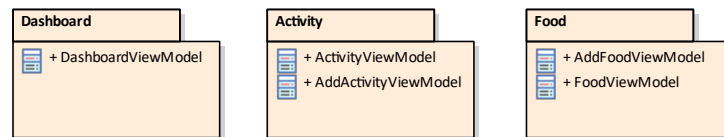
### 2.3.3.4 Class FoodView

Trieda zobrazuje jednotlivé jedla alebo sprostredkúva zobrazenie jednotlivých jedál pri ich prehľadávaní.

Název atributu	Datový typ	Popis
body	View	
foodCalories	int	

## 2.4 ViewModel

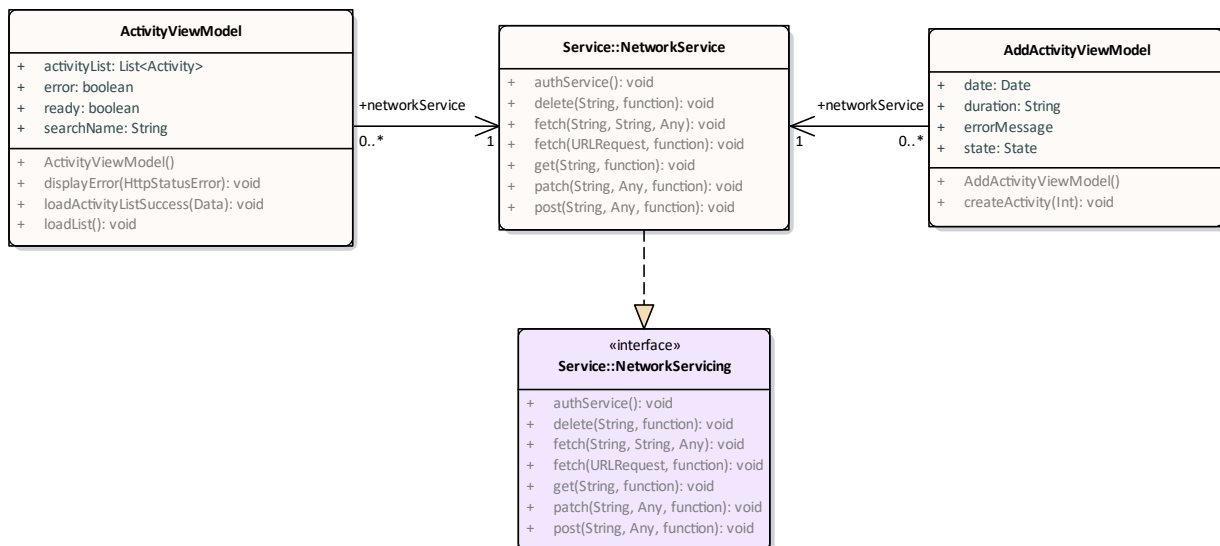
Balíček je určen pro třídy, které načítají data z datovo-business vrstvy pomocí rozhraní a poskytují je jako pozorovatelné kolekce třídám z balíčku View. Jedná se o druhou část **prezentační** vrstvy.



Obrázek 15 - ViewModel

### 2.4.1 Activity

Balíček obsahuje view modely týkající se přidávání, úpravy a hledání aktivit.





Obrázek 16 - Activity

### 2.4.1.1 Class ActivityViewModel

Trieda, ktorá sa používa na vyhľadanie a zobrazenie aktivít.

Název atributu	Datový typ	Popis
activityList	List<Activity>	
error	boolean	
ready	boolean	
searchName	String	
Název metody	Návratový typ	Popis
ActivityViewModel		
displayError	void	Funkcia načítava zoznam aktivít v prípade neúspechu. Parametry: <b>error: HttpStatusCode -</b>
loadActivityListSuccess	void	Funkcia načítava zoznam aktivít v prípade úspechu. Parametry: <b>data: Data -</b>
loadList	void	Funkcia na načítanie zoznamu aktivít. Ak je vyhľadávaný text prázdny, načíta sa aktivita užívateľa. Inak sa vykoná vyhľadaný text.

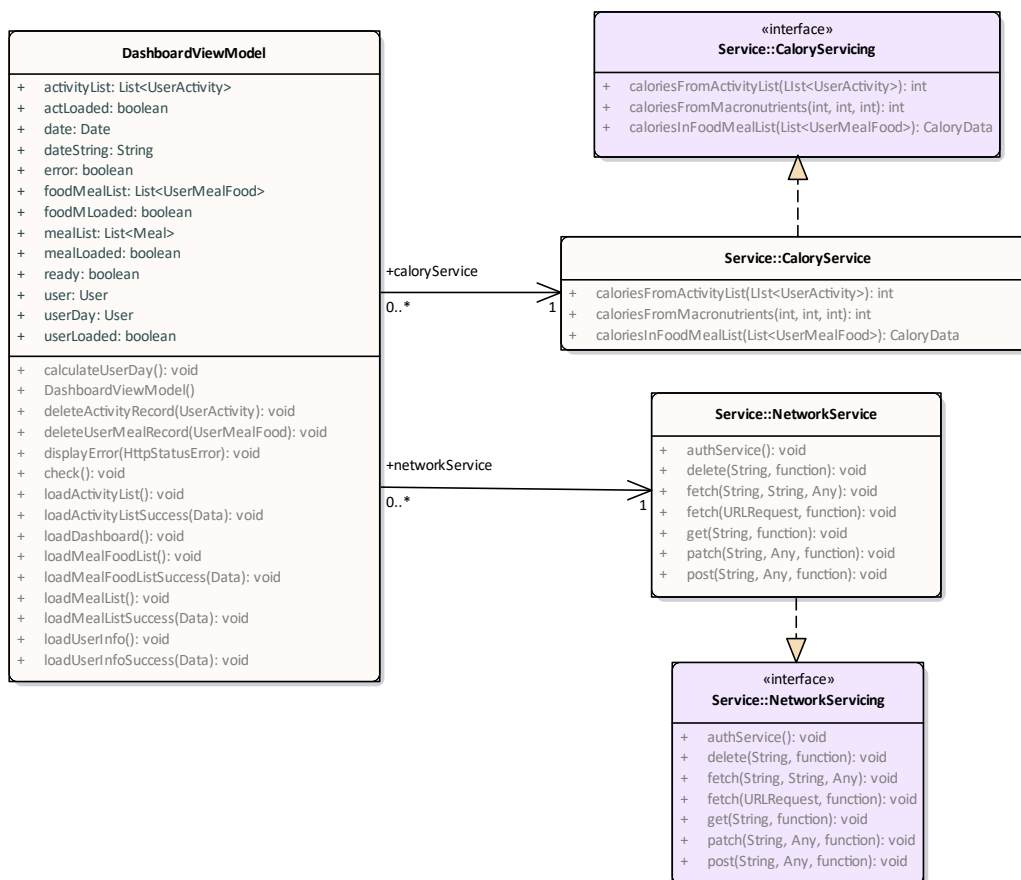
### 2.4.1.2 Class AddActivityViewModel

Trieda, ktorá sa používa na pridanie aktivity užívateľom.

Název atributu	Datový typ	Popis
date	Date	
duration	String	
errorMessage		
state	State	
Název metody	Návratový typ	Popis
AddActivityViewModel		
createActivity	void	Funkcia ktorá zavolá "Log activity" na serveri. Parametry: <b>activityId: Int -</b>

## 2.4.2 Dashboard

Balíček obsahuje view modely týkajúci sa denného prehľadu.



Obrázek 17 - Dashboard

### 2.4.2.1 Class DashboardViewModel

Trieda, ktorá drží data zobrazené na obrazovky.

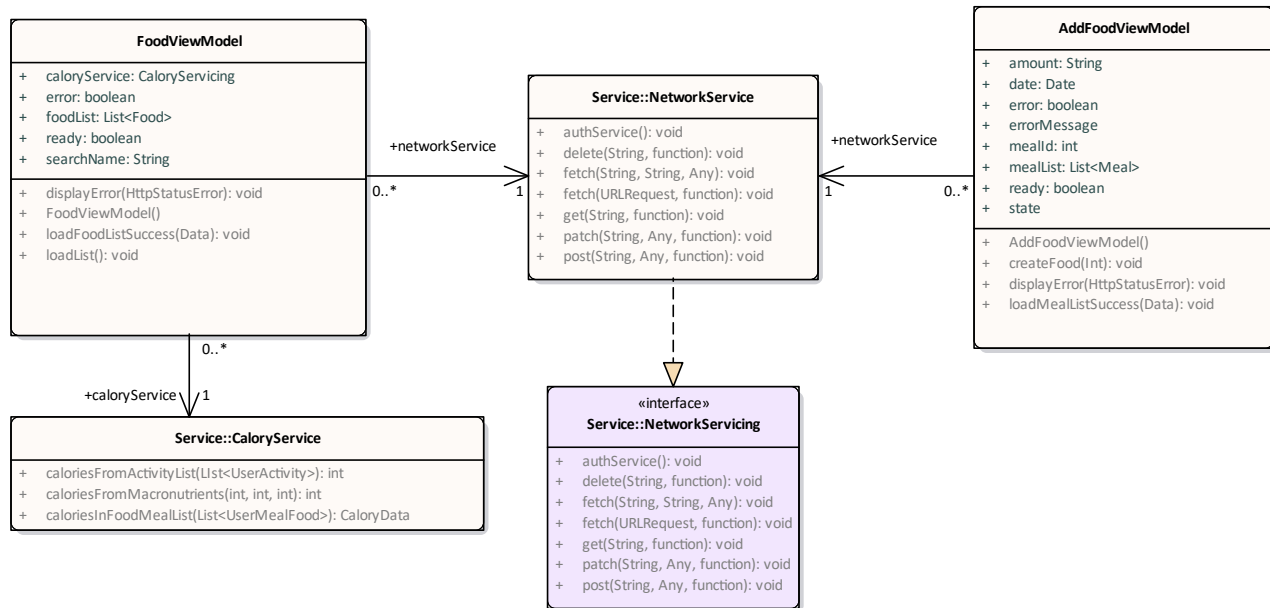
Název atributu	Datový typ	Popis
activityList	List<UserActivity>	
actLoaded	boolean	
date	Date	
dateString	String	
error	boolean	
foodMealList	List<UserMealFood>	
foodMLoaded	boolean	
mealList	List<Meal>	
mealLoaded	boolean	
ready	boolean	
user	User	
userDay	User	
userLoaded	boolean	
Název metody	Návratový typ	Popis
calculateUserDay	void	Funkcia, ktorá nájde informácie o kalóriách, ktoré užívateľ spálil



		a kalóriách, ktoré dnes užívateľ zjedol.
DashboardViewModel		
deleteActivityRecord	void	Funkcia na odstránenie aktivity. Parametry: <b>activity: UserActivity -</b>
deleteUserMealRecord	void	Funkcia na mazanie zjedeného jedla. Parametry: <b>userMeal: UserMealFood -</b>
displayError	void	Parametry: <b>error: HttpStatusError -</b>
check	void	Funkcia, ktorá je volaná stále po tom, ako sa časť dôležitých dát načíta.
loadActivityList	void	Funkcia, ktorá asynchrónne načíta aktivity ktoré užívateľ vykonal.
loadActivityListSuccess	void	Načíta data obsahujúce zoznam aktivít dnes vykonaných. Parametry: <b>data: Data -</b>
loadDashboard	void	Funkcia na načítanie informácie obrazovky.
loadMealFoodList	void	Funkcia, ktorá asynchrónne načíta jedlo, ktoré užívateľ skonzumoval počas dneška.
loadMealFoodListSuccess	void	Načíta data obsahujúce zoznam dnes zjedeného jedla. Parametry: <b>data: Data -</b>
loadMealList	void	Funkcia, ktorá asynchrónne načíta všetko prístupné jedlo.
loadMealListSuccess	void	Načíta data obsahujúce zoznam všetkých jedál. Parametry: <b>data: Data -</b>
loadUserInfo	void	Funkcia, ktorá asynchrónne načíta informácie o užívateľovom dlhodobom ciele.
loadUserInfoSuccess	void	Načíta data obsahujúce zoznam užívateľov. Parametry: <b>data: Data -</b>

### 2.4.3 Food

Balíček obsahuje view modely týkajúce sa pridávania, úpravy a hľadania jediel.



Obrázek 18 - Food

### 2.4.3.1 Class AddFoodViewModel

Trieda, ktorá sa používa na pridávanie jedla chodu užívateľom.

Název atributu	Datový typ	Popis
amount	String	
date	Date	
error	boolean	
errorMessage		
mealId	int	
mealList	List<Meal>	
ready	boolean	
state		
Název metody	Návratový typ	Popis
AddFoodViewModel		
createFood	void	Parametry: <b>foodId: Int</b> -
displayError	void	Parametry: <b>error: HttpStatusError</b> -
loadMealListSuccess	void	Parametry: <b>data: Data</b> -



--	--	--

### 2.4.3.2 Class FoodViewModel

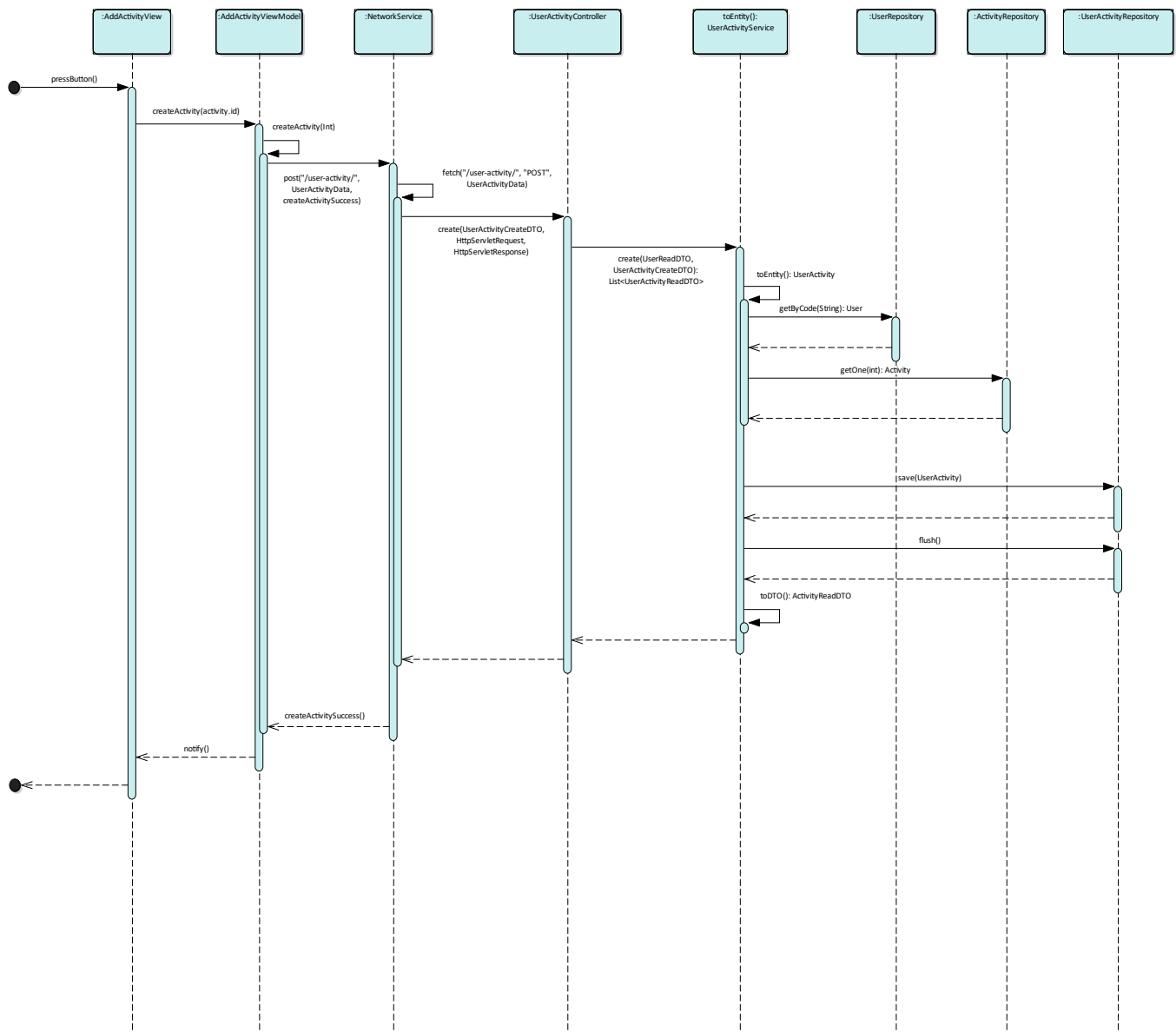
Trieda, ktorá sa používa na zobrazenie zoznamu jedla.

Název atributu	Datový typ	Popis
caloryService	CaloryServicing	
error	boolean	
foodList	List<Food>	Trieda, ktorá sa používa na vyhľadanie, alebo zobrazenie zoznamu jedla.
ready	boolean	
searchName	String	
Název metody	Návratový typ	Popis
displayError	void	Parametry: <b>error: HttpStatusError</b> -
FoodViewModel		
loadFoodListSuccess	void	Funkcia, ktorá načítava zoznam jedla v prípade úspechu. Parametry: <b>data: Data</b> -
loadList	void	Funkcia, ktorá asynchrónne načítava zoznam jedla.

## 3. Realizace případů užití (Model komunikace)

### 3.1 Zaznamenání aktivity

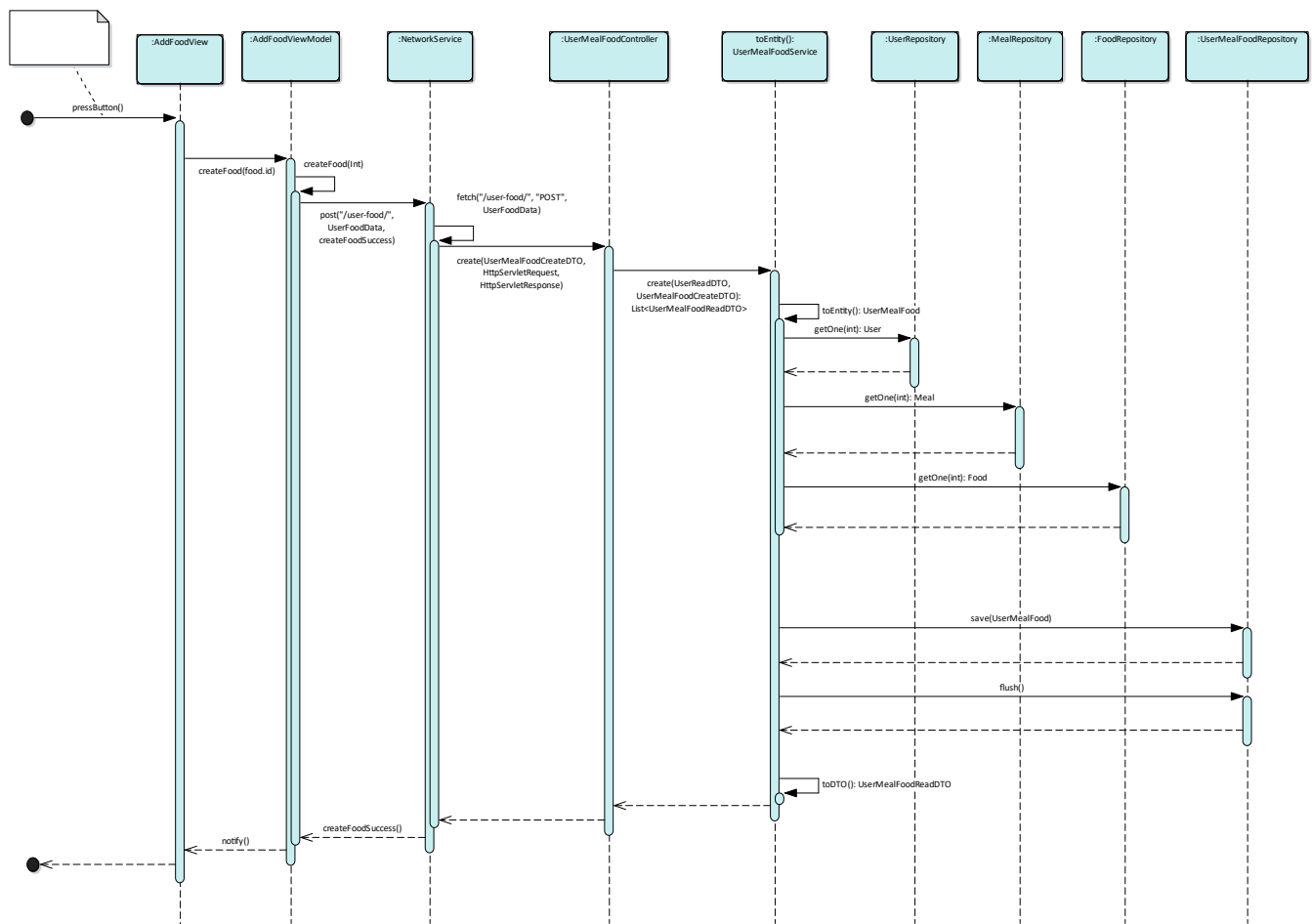




Obrázek 19 - Zaznamenání aktivity

Model popisuje realizaci případu užití "Zaznamenání aktivity" v Aplikaci a RestAPI.

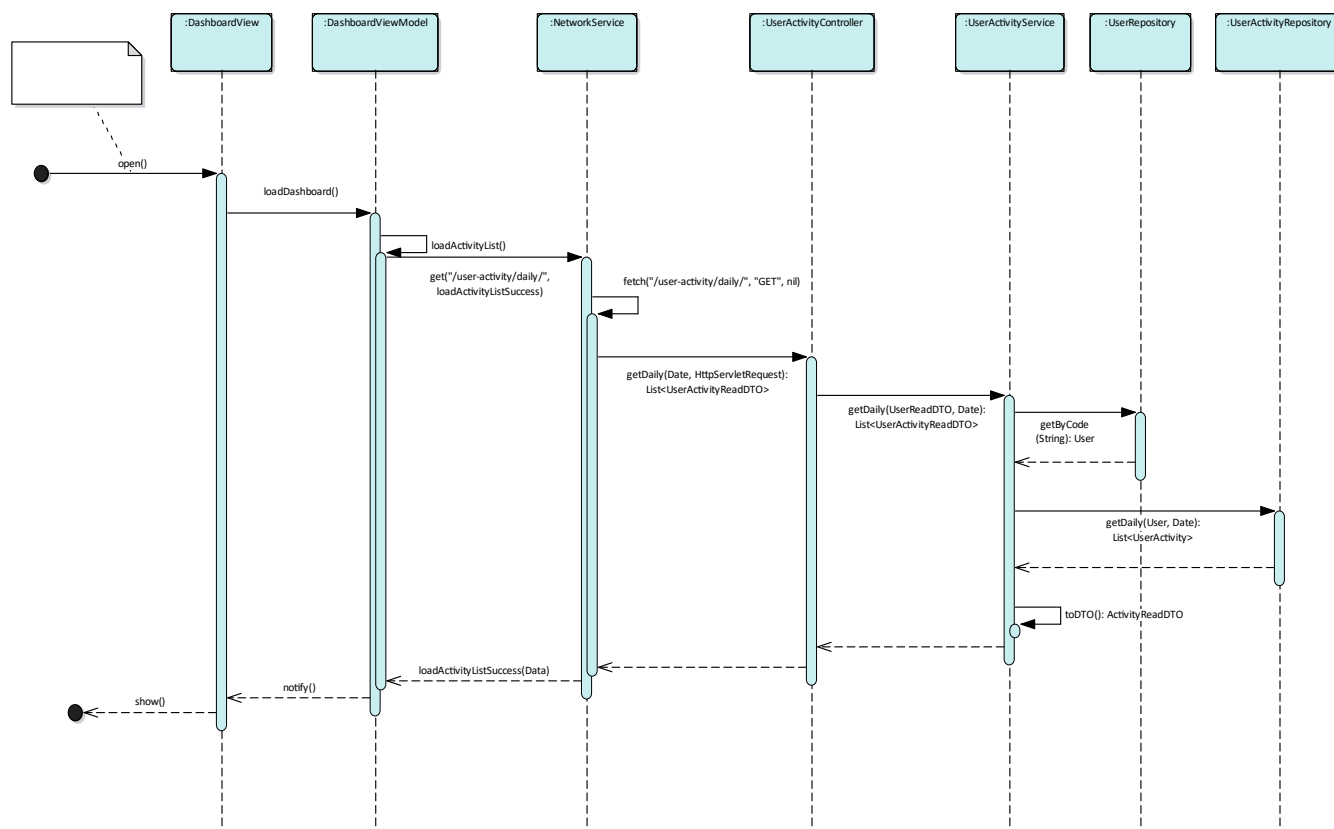
### 3.2 Zaznamenání konzumace jídla



Obrázek 20 - Zaznamenání konzumace jídla

Model popisuje realizaci případu užití "Zaznamenání konzumace jídla" v Aplikaci a RestAPI.

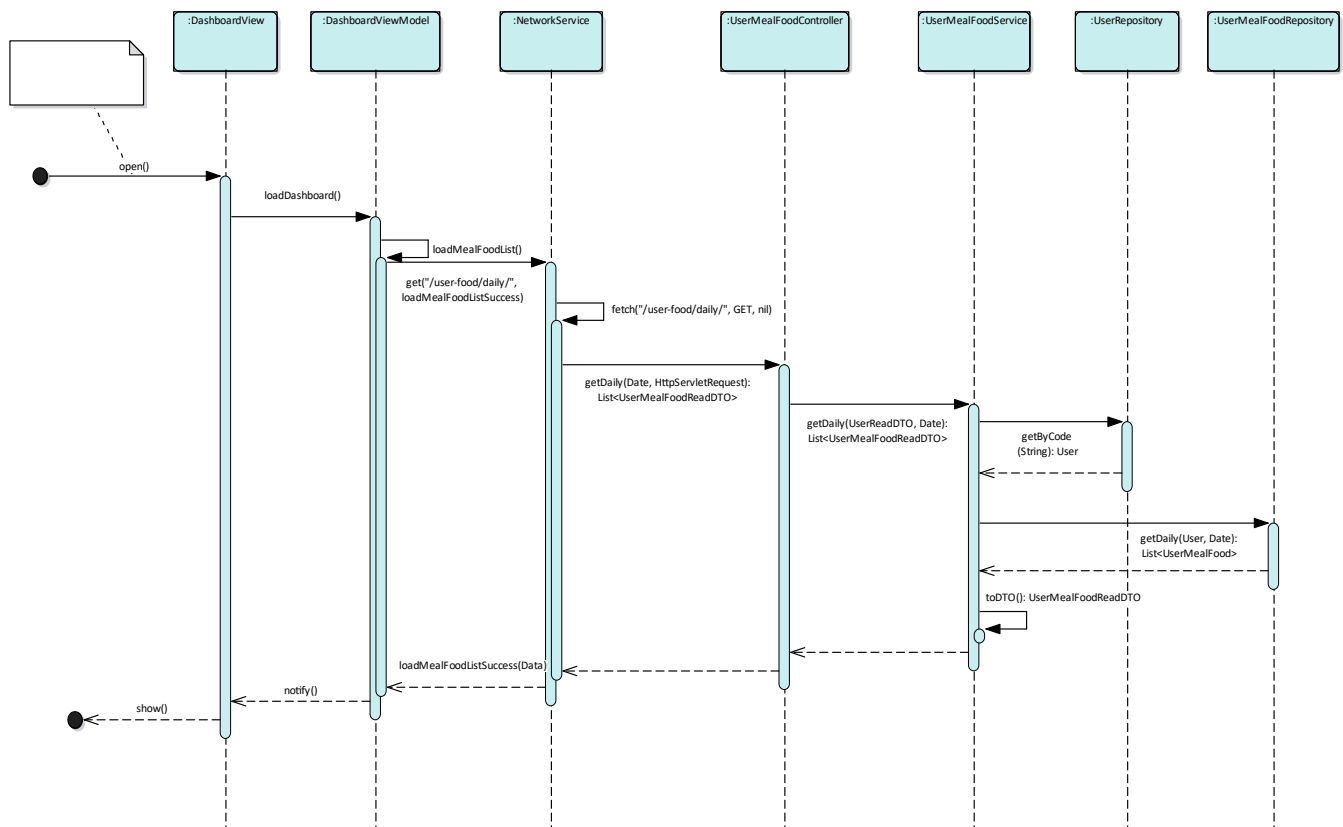
### 3.3 Zobrazení a hledání aktivit



Obrázek 21 - Zobrazení a hledání aktivit

Model popisuje realizaci případu užití "Zobrazení a hledání aktivit" v Aplikaci a RestAPI.

### 3.4 Zobrazení a hledání jídel



Obrázek 22 - Zobrazení a hledání jídel

Model popisuje realizaci případu užití "Zobrazení a hledání jídel" v Aplikaci a RestAPI.



**FAKULTA  
INFORMAČNÍCH  
TECHNOLOGIÍ  
ČVUT V PRAZE**