**Our table**

**Návrh architektury a databázový model**

Dokument vytvořen pro potřeby předmětu BI-SI1

Autoři: Ondřej Hývnar, Jakub Renc, Jiří Szkandera, Daniel Bukač

**Obsah**

1. Model architektury 3

1.1 pl 3

1.1.1 controller 3

1.1.1.1 Class HomeController 4

1.1.1.2 Class MenuController 4

1.1.1.3 Class ReservationController 4

1.1.1.4 Class ReservationForm 5

1.1.1.5 Class TableSelectForm 5

1.1.2 templates 5

1.2 bl 5

1.2.1 services 5

1.2.1.1 Class ReservationService 6

1.2.1.2 Class TableService 6

1.3 dl 6

1.3.1 entity 6

1.3.1.1 Class Address 7

1.3.1.2 Class Alergen 7

1.3.1.3 Class Customer 7

1.3.1.4 Class DBEngine 7

1.3.1.5 Class Meal 8

1.3.1.6 Class MealOrder 8

1.3.1.7 Class Menu 8

1.3.1.8 Class Reservation 8

1.3.1.9 Class Schedule 8

1.3.1.10 Class Tbl 9

1.3.2 mapper 9

1.3.2.1 Class AddressMapper 9

1.3.2.2 Class AlergenMapper 10

1.3.2.3 Class CustomerMapper 10

1.3.2.4 Class MealMapper 10

1.3.2.5 Class MealOrderMapper 11

1.3.2.6 Class MenuMapper 11

1.3.2.7 Class Reservation 11

1.3.2.8 Class ScheduleMapper 12

1.3.2.9 Class TableMapper 12

2. DDL 14

2.1 Adresa *«table»* 14

2.2 Alergen *«table»* 14

2.3 Jidlo *«table»* 15

2.4 Menu *«table»* 15

2.5 Objednavka\_jidla *«table»* 15

2.6 Rezervace *«table»* 15

2.7 Rozvrh *«table»* 15

2.8 Stul *«table»* 15

2.9 Uzivatel *«table»* 16

# Model architektury



Obrázek 1 - Model architektury

## pl

Balíček obsahuje logiku pro prezentaci dat a pro zpracování vstupu klienta.



Obrázek 2 - pl

### controller

Balíček obsahuje všechny kontrolery - třídy, jejichž účelem je zpracovávat vstupy uživatele.



Obrázek 3 - controller

#### Class HomeController

Třída pro kontrolu hlavní stránky.

| **Název atributu** | **Datový typ** | **Popis** |
| --- | --- | --- |
| home\_bp | Blueprint |  |
| **Název metody** | **Návratový typ** | **Popis** |
| home | string |  |

#### Class MenuController

Třída pro kontrolu stránky s menu.

| **Název atributu** | **Datový typ** | **Popis** |
| --- | --- | --- |
| menu\_bp | Blueprint |  |
| **Název metody** | **Návratový typ** | **Popis** |
| menu | string | Parametry:  **mapper: IMealMapper** - |

#### Class ReservationController

Třída pro kontrolu rezervační stránky.

| **Název atributu** | **Datový typ** | **Popis** |
| --- | --- | --- |
| reservation\_bp | Blueprint |  |
| **Název metody** | **Návratový typ** | **Popis** |
| finish | string |  |
| reserve | Union[string, Responce] |  |
| select\_table | Union[string, Responce] | Parametry:  **table\_mapper: ITableMapper** -  Parametry:  **schedule\_mapper: IScheduleMapper** - |

#### Class ReservationForm

Třída pro získávání vstupu z rezervačního formuláře.

| **Název atributu** | **Datový typ** | **Popis** |
| --- | --- | --- |
| date | date |  |
| guests\_cnt | int |  |
| time | time |  |

#### Class TableSelectForm

Třída pro získání vstupu z formuláře pro výběr stolu.

| **Název atributu** | **Datový typ** | **Popis** |
| --- | --- | --- |
| table\_id | int |  |

### templates

Balíček obstarávající vykreslování informací (webových stránek) uživateli.

Pozn.: Pro paradigma MVC tento balíček reprezentuje "views". Z technických důvodů se tento balíček ale jmenuje "templates".



Obrázek 4 - view

## bl

Balíček obsahuje logiku pro práci s daty, slouží jako podpora pro složitější dotazy z prezentační vrstvy. Neslouží ovšem pro jejich fyzické obstarávání.



Obrázek 5 - bl

### services

Balíček poskytující služby pro vyšší vrstvy. Obstarává složitější dotazy, slouží jako nadstavba nad data layer.



Obrázek 6 - services

#### Class ReservationService

Služba pro práci s rezervacemi.

| **Název metody** | **Návratový typ** | **Popis** |
| --- | --- | --- |
| add\_reservation | boolean | Parametry:  **time: datetime** -  Parametry:  **tableID: int** -  Parametry:  **tableMapper: ITableMapper** - |

#### Class TableService

Služba pro práci se stoly.

| **Název metody** | **Návratový typ** | **Popis** |
| --- | --- | --- |
| get\_free\_tables | List<Table> | Parametry:  **since: datetime** -  Parametry:  **count: int** -  Parametry:  **table\_mapper: ITableMapper** -  Parametry:  **schedule\_mapper: IScheduleMapper** - |

## dl

Balíček obsahuje logiku související s uložením dat, a to jak jejich načítání, tak jejich ukládání.



Obrázek 7 - dl

### entity

Balíček obsahuje definici objektů, které se mapují na databázové tabulky.



Obrázek 8 - entity

#### Class Address

Informace o bydlistich jednotlvych uzivatelu

| **Název atributu** | **Datový typ** | **Popis** |
| --- | --- | --- |
| mesto | string |  |
| PSC | int |  |
| stat | string |  |
| ulice | string |  |

#### Class Alergen

Kazdy alergen ma svuj identifikator - cislo a nazev

| **Název atributu** | **Datový typ** | **Popis** |
| --- | --- | --- |
| cislo | int |  |
| nazev | string |  |

#### Class Customer

Obsahuje kontaktni udaje o existujicich uzivatelich.

| **Název atributu** | **Datový typ** | **Popis** |
| --- | --- | --- |
| email | string |  |
| jmeno | string |  |
| prijmeni | string |  |
| telefon | string |  |

#### Class DBEngine

Třída obalující DB engine.

| **Název atributu** | **Datový typ** | **Popis** |
| --- | --- | --- |
| base | AutomapBase |  |
| db\_string | string |  |
| engine | Engine |  |
| session | Session |  |
| valid | boolean |  |
| **Název metody** | **Návratový typ** | **Popis** |
| \_\_init\_base | void |  |
| \_\_init\_engine | void |  |
| \_\_init\_session | void |  |
| get\_base | AutomapBase |  |
| get\_engine | Engine |  |
| get\_session | Session |  |

#### Class Meal

Samotne jidlo, atributy cena a jeho nazev

| **Název atributu** | **Datový typ** | **Popis** |
| --- | --- | --- |
| cena | int |  |
| nazev | string |  |

#### Class MealOrder

Vaze se ke konkretnimu jidlu a urcuje jeho pocet v objednavce

| **Název atributu** | **Datový typ** | **Popis** |
| --- | --- | --- |
| pocet | int |  |

#### Class Menu

Urcuje dostupne menu v danem casovem rozmezi

| **Název atributu** | **Datový typ** | **Popis** |
| --- | --- | --- |
| platnostDo | date |  |
| platnostOd | date |  |

#### Class Reservation

Obsahuje casove udaje kazde rezervaci

| **Název atributu** | **Datový typ** | **Popis** |
| --- | --- | --- |
| datumDo | date |  |
| datumOd | date |  |

#### Class Schedule

Urcuje dostupnost stolu v danem casovem okne

| **Název atributu** | **Datový typ** | **Popis** |
| --- | --- | --- |
| datumDo | date |  |
| datumOd | date |  |
| jeDostupny | boolean |  |

#### Class Tbl

Kazdy stul si udrzuje pocet mist k sezeni

| **Název atributu** | **Datový typ** | **Popis** |
| --- | --- | --- |
| pocetMist | int |  |

### mapper

Balíček obsahuje implementace operací nad entitami.



Obrázek 9 - mapper

#### Class AddressMapper

Trida implementujici logiku pro persistenci dat o adresach v databazi postgreSQL.

| **Název metody** | **Návratový typ** | **Popis** |
| --- | --- | --- |
| add | boolean | Metoda vlozi adresu do databaze  Parametry:  **obj: Address** - Objekt k vlozeni do databaze |
| delete | boolean | Metoda odebere adresu z databaze  Parametry:  **obj: Address** - Objekt ke smazani z databaze |
| get | Address | Metoda vrati adresu z databaze podle id  Parametry:  **obj\_id: int** - ID objektu k ziskani |
| get\_all | List<Address> | Metoda vrati vsechny adresy z databaze jako list |

#### Class AlergenMapper

Trida implementujici logiku pro persistenci dat o alergenech v databazi postgreSQL.

| **Název metody** | **Návratový typ** | **Popis** |
| --- | --- | --- |
| add | boolean | Metoda vlozi alergen do databaze  Parametry:  **obj: Alergen** - Objekt k vlozeni do databaze |
| delete | boolean | Metoda smaze alergen z databaze  Parametry:  **obj: Alergen** - Objekt ke smazani z databaze |
| get | Alergen | Metoda vrati alergen s danym ID  Parametry:  **obj\_id: int** - ID objektu k ziskani |
| get\_all | List<Alergen> | Metoda vrati vsechny alergeny z databaze v podobe listu |

#### Class CustomerMapper

Trida implementujici logiku pro persistenci dat o zakaznicich v databazi postgreSQL.

| **Název metody** | **Návratový typ** | **Popis** |
| --- | --- | --- |
| add | boolean | Metoda vlozi daneho uzivatele do databaze  Parametry:  **obj: Customer** - |
| delete | boolean | Metoda smaze daneho uzivatele z databaze  Parametry:  **obj: Customer** - Objekt ke smazani z databaze |
| get | Customer | Metoda vrati daneho uzivatele podle ID z databaze  Parametry:  **obj\_id: int** - ID objektu k ziskani |
| get\_all | List<Customer> | Metoda vrati vsechny uzivatele z databaze v podobe listu |

#### Class MealMapper

Trida implementujici logiku pro persistenci dat o jidlech v databazi postgreSQL.

| **Název metody** | **Návratový typ** | **Popis** |
| --- | --- | --- |
| add | boolean | metoda vlozi do databaze dane jidlo  Parametry:  **obj: Meal** - Objekt k vlozeni do databaze |
| delete | boolean | metoda smaze z databaze dane jidlo  Parametry:  **obj: Meal** - Objekt ke smazani z databaze |
| get | Meal | metoda vrati jidlo s danym id  Parametry:  **obj\_id: int** - ID objektu k ziskani |
| get\_all | List<Meal> | Metoda vrati vsechny jidla z databaze v podobe listu |

#### Class MealOrderMapper

Trida implementujici logiku pro persistenci dat o objednavkach jidel v databazi postgreSQL.

| **Název metody** | **Návratový typ** | **Popis** |
| --- | --- | --- |
| add | boolean | Metoda vlozi danou objednavku do databaze  Parametry:  **obj: MealOrder** - Objekt k vlozeni do databaze |
| delete | boolean | Metoda smaze danou objednavku jidla z databaze  Parametry:  **obj: MealOrder** - Objekt ke smazani z databaze |
| get | MealOrder | Metoda vrati objednavku jidla s danym ID  Parametry:  **obj\_id: int** - ID objektu k ziskani |
| get\_all | List<MealOrder> | Metoda vrati z databaze vsechny objednavky jidel v podobe listu |

#### Class MenuMapper

Trida implementujici logiku pro persistenci dat o menu v databazi postgreSQL.

| **Název metody** | **Návratový typ** | **Popis** |
| --- | --- | --- |
| add | boolean | Metoda vlozi dane menu do databaze  Parametry:  **obj: Menu** - Objekt k vlozeni do databaze |
| delete | boolean | Metoda smaze dane menu z databaze  Parametry:  **obj: Menu** - Objekt ke smazani z databaze |
| get | Menu | Metoda vrati menu s danym ID  Parametry:  **obj\_id: int** - ID objektu k ziskani |
| get\_all | List<Menu> | Metoda vrati vsechny menu z databaze v podobe listu |

#### Class Reservation

Trida implementujici logiku pro persistenci dat o rezervacich v databazi postgreSQL.

| **Název metody** | **Návratový typ** | **Popis** |
| --- | --- | --- |
| add | boolean | Metoda prida rezervaci do databaze  Parametry:  **obj: Reservation** - Objekt k vlozeni do databaze |
| delete | boolean | Metoda smaze danou rezervaci z databaze  Parametry:  **obj: Reservation** - Objekt ke smazani z databaze |
| get | Reservation | Metoda vrati rezervaci s danym id z databaze  Parametry:  **obj\_id: int** - ID objektu k ziskani |
| get\_all | List<Reservation> | Metoda vrati vsechny rezervace z databaza v podobe listu |

#### Class ScheduleMapper

Trida implementujici logiku pro persistenci dat o rozvrzich v databazi postgreSQL.

| **Název metody** | **Návratový typ** | **Popis** |
| --- | --- | --- |
| add | boolean | Metoda prida rozvrh do databaze  Parametry:  **obj: Schedule** - Objekt k vlozeni do databaze |
| delete | boolean | Metoda smaze rozvrh z databaze  Parametry:  **obj: Schedule** - Objekt ke smazani z databaze |
| get | Schedule | Metoda vrati rozvrh s danym id z databaze  Parametry:  **obj\_id: int** - ID objektu k ziskani |
| get\_all | List<Schedule> | Metoda vrati vsechny rozvrhy z databaze v podobe listu |

#### Class TableMapper

Trida implementujici logiku pro persistenci dat o stolech v databazi postgreSQL.

| **Název metody** | **Návratový typ** | **Popis** |
| --- | --- | --- |
| add | boolean | Metoda prida dany stul do databaze  Parametry:  **obj: Tbl** - Objekt k vlozeni do databaze |
| delete | boolean | Metoda smaze dany stul z databaze  Parametry:  **obj: Tbl** - Objekt ke smazani z databaze |
| get | Tbl | Metoda vrati stul s danym ID z databaze  Parametry:  **obj\_id: int** - ID objektu k ziskani |
| get\_all | List<Tbl> | Metoda vrati vsechny stoly z databaze v podobe listu |
| get\_free\_tables | List<Schedule> | Parametry:  **seats: int** -  Parametry:  **since: datime** -  Parametry:  **untile: datetime** - |

# DDL



Obrázek 10 - DDL

## Adresa *«table»*

Informace o bydlistich jednotlvych uzivatelu. Adresa neni povinna tedy zalezi pouze na uzivateli, ktere udaje o sobe zpristupni

| **Název atributu** | **Datový typ** | **Not null** | **Popis** |
| --- | --- | --- | --- |
| Mesto | varchar(50) | False |  |
| Psc | integer | False |  |
| Stat | varchar(50) | False |  |
| Ulice | varchar(50) | False |  |
| AdresaID | integer | True |  |

## Alergen *«table»*

Kazdy alergen ma svuj identifikator - cislo a nazev

| **Název atributu** | **Datový typ** | **Not null** | **Popis** |
| --- | --- | --- | --- |
| Cislo | integer | True | Zobrazovane cislo alergenu. |
| Nazev | varchar(50) | True |  |
| AlergenID | integer | True | ID alergenu. Je pouzito, aby bylo mozne pripadne menit cislo alergenu (Cislo) a zaroven nebylo nutno menit jeho ID. |

## Jidlo *«table»*

Samotne jidlo, atributy cena a jeho nazev

| **Název atributu** | **Datový typ** | **Not null** | **Popis** |
| --- | --- | --- | --- |
| Cena | integer | True |  |
| Nazev | varchar(50) | True |  |
| JidloID | integer | True |  |

## Menu *«table»*

Urcuje dostupne menu v danem casovem rozmezi

| **Název atributu** | **Datový typ** | **Not null** | **Popis** |
| --- | --- | --- | --- |
| Platnostdo | date | True |  |
| Platnostod | date | True |  |
| MenuID | integer | True |  |

## Objednavka\_jidla *«table»*

Vaze se ke konkretnimu jidlu a urcuje jeho pocet v objednavce

| **Název atributu** | **Datový typ** | **Not null** | **Popis** |
| --- | --- | --- | --- |
| Pocet | integer | True |  |
| Objednavka\_jidlaID | integer | True |  |
| JidloID | integer | True |  |

## Rezervace *«table»*

Obsahuje casove udaje kazde rezervaci

| **Název atributu** | **Datový typ** | **Not null** | **Popis** |
| --- | --- | --- | --- |
| Datumdo | date | True | Nutne vyplnena, system vyplnuje jako Datumod + 2 hodiny, ale je moznost pripadne zmeny. |
| Datumod | date | True |  |
| RezervaceID | integer | True |  |
| Objednavka\_jidlaID | integer | False |  |
| UzivatelID | integer | False |  |

## Rozvrh *«table»*

Urcuje dostupnost stolu v danem casovem okne

| **Název atributu** | **Datový typ** | **Not null** | **Popis** |
| --- | --- | --- | --- |
| Datumdo | date | True |  |
| Datumod | date | True |  |
| Jedostupny | boolean | True |  |
| RozvrhID | integer | True |  |
| StulID | integer | True |  |

## Stul *«table»*

Kazdy stul si udrzuje pocet mist k sezeni

| **Název atributu** | **Datový typ** | **Not null** | **Popis** |
| --- | --- | --- | --- |
| Pocetmist | integer | True |  |
| StulID | integer | True |  |

## Uzivatel *«table»*

Obsahuje kontaktni udaje o existujicich uzivatelich.

| **Název atributu** | **Datový typ** | **Not null** | **Popis** |
| --- | --- | --- | --- |
| Email | varchar(50) | True | Email slouzi k zakladni komunikaci mezi portalem a uzivatelem. Proto je nutny i unikatni. |
| Jmeno | varchar(50) | False |  |
| Prijmeni | varchar(50) | True | Nutny identifikator pro email/telefon. (Osloveni, jmeno, podle ktere se pripadne bude hledat rezervace,...) |
| Telefon | varchar(50) | False | Jedna se o unikatni cislo ke kazdemu uzivateli. Pro komunikaci je email urceny jako primarni kanal a telefonni spojeni funguje jako dobrovolna alternativa. |
| UzivatelID | integer | True |  |