



IDS – protokol k projektu

Neonila Mashlai
xmashl00

March 10, 2023

Contents

1	První část	2
1.1	Zadání	2
1.2	Diagramy	2
1.3	Popis diagramů	3
1.3.1	Diagram případů užití	3
1.3.2	ER-diagram	3

První část

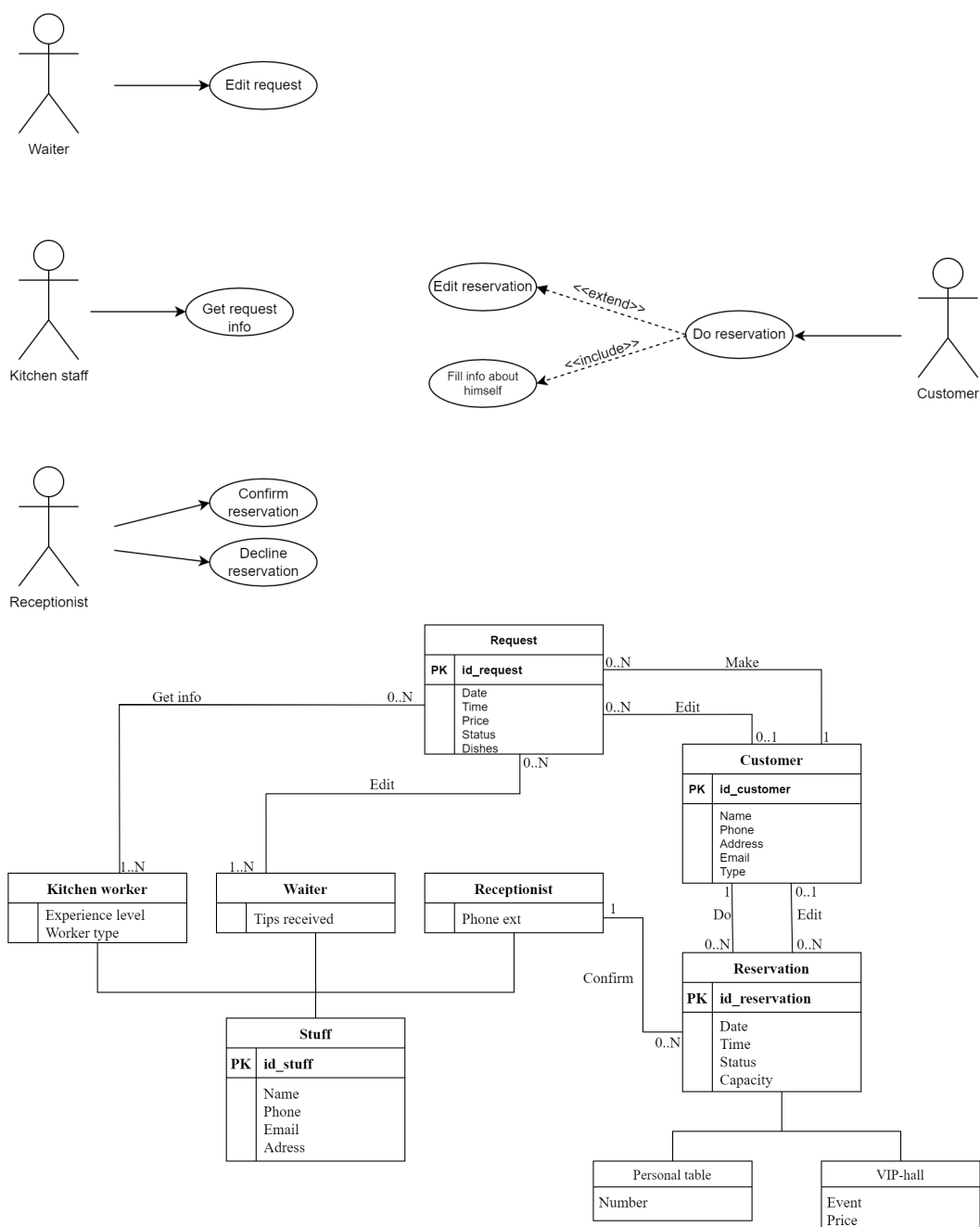
Zadání

Projekt č.: 13

Název projektu: **Restaurace**

Zadání: Restaurace poskytuje běžné stravovací služby veřejnosti. Mimo to umožňuje pořádání akcí v saloncích na základě rezervace. V restauraci je možné rezervovat i jednotlivá místa. Navrhněte informační systém, který podpoří běžné činnosti podniku.

Diagramy



Popis diagramů

Diagram případů užití

V **diagramu případů užití** máme 4 aktéry, to jsou lidé, kteří mají právo provádět nějaké změny v informačním systému.

Customer má právo vyplnit o sobě údaje, provést rezervaci a upravit ji. Vyplňování údajů o sobě máme jako *«include»* rezervace, protože součástí rezervace je vyplnění údajů o zákazníkovi, ale také číslo stolu či sálu, datum a čas. Editaci rezervace máme jako *«extend»* pro vytvoření rezervace. Nemůžete upravit rezervaci, pokud ještě neexistuje, úprava rezervace je volitelná. Úprava rezervace může znamenat změnu stolu, místnosti, data, času nebo zrušení rezervace.

Waiter má právo požadavek upravit. Editace požadavku znamená pro waiter jeho vytvoření při přijímání objednávky od customer, možnost změny stávajících jídel a přidání dalších nebo zrušení objednávky.

Kitchen staff v informačním systému přijímá a kontroluje informace o objednávce vytvořené klientem, aby věděl, jaké pokrmy připravit.

Receptionist může rezervaci potvrdit a zrušit.

ER-diagram

V **ER-Diagramu** máme 9 entit.

Staff entita je spojena generalizací-specializačním vztahem s entitami **kitchen staff**, **waiter**, **receptionist**, protože každý z nich je personál, ale zároveň mají pro entitu jedinečné argumenty a různé vazby na jiné subjekty. Primární klíč pro všechny entity je **id_stuff**.

Entita **kitchen staff** má vztah „get info“ s entitou **request**. Každý zaměstnanec kuchyně může zkontrolovat od 0 do N objednávek, každou objednávku může zkontrolovat 1 až N zaměstnanců kuchyně, aby restaurace fungovala, musí objednávku zkontrolovat alespoň 1 kuchař, protože bez toho ji nemůže uvařit, ale při zároveň si tuto objednávku může prohlédnout neomezený počet kuchařů.

Entita **waiter** má vztah „edit“ s entitou **request**. Každý číšník může upravovat od 0 do N objednávek, editace může znamenat vytvoření dotazu po přijetí objednávky od zákazníka a může upravovat pokrmy, stav, počet a evidovat argumenty ceny. Každou objednávku může změnit 1 až N číšníků. Měl by být alespoň 1, ale možná i více.

Entita **request** má s entitou **customer** 2 vztahy: „make“ a „edit“. To znamená vytvořit objednávku a upravit ji. Každý zákazník může vytvořit 0 až N objednávek, ale každou objednávku může vytvořit pouze 1 zákazník. Každý zákazník může editovat 0 až N objednávek, ale každá objednávka může editovat 0 až 1 zákazníka, protože maximálně 1 objednávku může editovat 1 zákazník, ale nemusí.

Entitu **reservation** spojuje vztah generalizace-specializace s entitami **personal table** a **Vip-hall**, protože každá z nich je rezervací, ale zároveň mají i argumenty jedinečné pro entitu. Primární klíč pro všechny entity je **id_reservation**.

Entit **customer** má s entitou **reservation** 2 vztahy: „do“ a „edit“. To znamená vytvořit rezervaci a upravit ji. Každý zákazník může vytvořit 0 až N rezervací, ale každou rezervaci může vytvořit pouze 1 zákazník. Každý zákazník může editovat 0 až N rezervací, ale každou rezervaci může editovat 0 až 1 zákazník, protože 1 rezervaci může editovat maximálně 1 zákazník, ale nemusí.

Entita **receptionist** má s entitou **reservation** „confirm“ vztah. Každá recepční může potvrdit od 0 do rezervace. Každou rezervaci může potvrdit pouze 1 recepční. Pokud recepční rezervaci nepotvrdí, stav rezervace se změní na „unsuccessful“.