

VYSOKÉ UČENÍ TECHNICKÉ
V BRNĚ
FAKULTA INFORMAČNÍCH
TECHNOLOGIÍ

Dokumentace k projektu do
předmětu IDS
Adam Cologna, Tomáš Souček

1 Zadání (Vězeňská pekárna - 7. IUS)

Navrhněte jednoduchý informační systém vězeňské pekárny, která dodává pečivo do okolních věznic. Systém umožňuje spravovat informace o nabízeném pečivu, věznicích, jejich vězních a umožňuje zákazníkům (vězňům) vytvářet objednávky pečiva.

Objednávka musí obsahovat informaci o tom kolik a jakého pečiva je objednáváno a za jakou cenu, termín dodání, způsob dodání atd.

Vězeňská pekárna navíc dle zákaznickova přání do každého pečiva zapeče vybraný předmět (pilník, šroubovák, ...). Dále bude systém evidovat suroviny, které se pro výrobu pečiva používají včetně jejich aktuálního množství skladem a nákupní ceny. Musí také evidovat kolik a jakých surovin je pro výrobu daného pečiva potřeba, zda se jedná o bezlepkové pečivo, jeho druh (celozrnné, běžné, atd.), seznam alergenů a hmotnost (bez zapečeného předmětu).

Do jednotlivých věznic dodávají pečivo pašeráci. Pašeráci jsou domluveni vždy s konkrétním dozorcem, díky kterému mohou do věznice pečivo bezpečně dodat. V systému tak musí být i informace o směnách, při kterých dozorcí dohlíží na pořádek ve věznici. Na každé směně je více dozorců, směna má čas nástupu a čas ukončení. Vězeňská pekárna spolupracuje s několika pašeráky, přičemž jeden z nich může dodávat pečivo do několika věznic.

Systém musí umožnit vedení pekárny plánovat produkci v závislosti na objednávkách a evidovat zákazníky. Pašeráci si mohou vypsát své rozvozy podle oblastí nebo podle zákazníků. Navíc, aby zaměstnanci pekárny věděli, zda budou vůbec schopni pečivo zákazníkovi dodat, musí mít v systému informaci o tom, ve které cele se zákazník nachází a o jaký typ cely se jedná (zda není zákazník na samotce, kam by se pečivo pašovalo stěží).

2 Popis řešení projektu

Ve druhé části projektu při vytváření databáze se u specializace/generalizace spojily dvě rozdílné entity (Pekař a Dozorce) do jedné (Zaměstnanec).

Ve čtvrté části projektu bylo za úkol vytvořit několik pokročilých objektů databáze. Pro práci ve dvojici byla přidělena přístupová práva pomocí `GRANT ALL ON jmeno_tabulky TO login`.

Jako první trigger byl vytvořen takový, který přidělí pašeráka do vězení č.1. Po přidání nové položky nebo provedení změny v tabulce Pašerák se provede vklad do tabulky `Paserak.Vezeni` a id vězení bude nastaveno na 1, jelikož jej chceme přidat do vězení č.1. Druhý trigger funguje podobně, ale v tomto případě přiděluje zaměstnance k určité směně.

První procedura se jménem `vypis_predmetu` slouží pro vypsání všech možných předmětů, které si může nechat klient objednat k zapečení. Při implementaci tohoto objektu byl využit i cursor, který slouží pro uchování předmětů a následně pomocí příkazů pro práci s cursory jako `OPEN`, `LOOP`, `FETCH`, atd. vypíše všechny potřebné informace. Další procedura se využívá pro zjištění stavu objednávky. Tento objekt očekává jako parametr id dané objednávky. Proběhne zde ošetření výjimky (v případě, že objednávka s daným id neexistuje tak místo jejího stavu vypíše informaci o její neexistenci).

Co se týče materializovaného pohledu, tak byl vytvořen seznam pašeráků a vězení, do kterých dodávají objednávky. Získání potřebných dat z vícero tabulek bylo implementováno přes `LEFT JOIN`. Pro práci s tímto pohledem a vypsání seznamu stačí použít dotaz `SELECT * FROM nazev_pohledu`.

Implementace indexu (v našem případě slouží pro rychlejší vyhledávání podle rodného čísla) se spojila s vytvořením `EXPLAIN PLANU`, který ukáže jednotlivé kroky dotazu na databázi. Nejprve dojde k vytvoření první verze explain planu, následně deklarace indexu a poté opět provedení explainu planu s využitím indexu, který právě díky indexu bude mnohem rychlejší. Indexace je jedno z nejlepších řešení pro zrychlení dotazů. Spojení tabulek opět proběhlo pomocí `LEFT JOIN` příkazu.

ER DIAGRAM FINALNÍ

