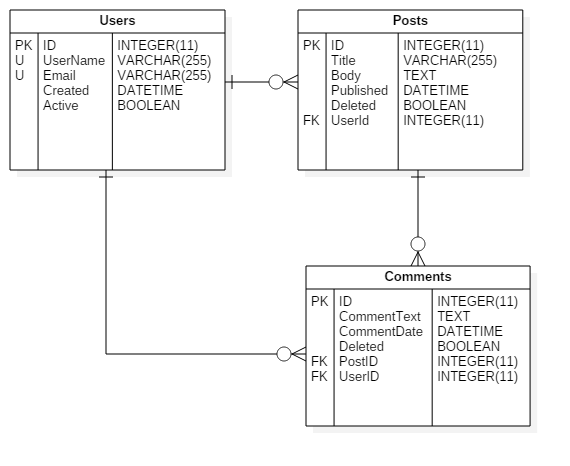
**Testování výkonnosti vytvořeného frameworku**

Cílem testování bylo porovnat rychlost vytvořeného frameworku s Entity Frameworkem, na kterém je námi vytvořený autentizační framework postaven, a dokázat tak, že námi vytvořený framework je prakticky použitelný.

Testování proběhlo na notebooku HP Probook 4530s s těmito parametry:

* **Procesor:** Intel Core i5 2.40GHz
* **RAM:**8.00 GB RAM
* **HDD:** Samsung SSD850 EVO, 250 GB
* **OS:** Windows 10 (64-bit)

Pro účely testování byla vytvořena MySQL databáze obsahující 3 tabulky (viz ERD diagram).



Nad databázi byla provedena sada různých dotazů (5x INSERT, 5x SELECT, 2x UPDATE). Dotazy byly vykonány jak za použití generického účtu v Entity Framework, tak za použití autentizovaného spojení v navrženém frameworku. Časy volání dotazů byly změřeny pro obě varianty a porovnány. Testování proběhlo nad různými velikostmi databáze (viz. Tabulka č. 1). Data byla vygenerována pomocí programu dbForge Data Generator for MySQL.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | users = 847 | users = 8470 | users = 84700 | users = 218500 | users = 847000 |
|  | posts = 100 | posts = 1000 | posts = 10000 | posts = 25000 | posts = 100000 |
|  | comments = 67 | comments = 670 | comments = 6700 | comments = 16750 | comments = 67000 |
| Entity Framework | 4.856676 | 4.6131696 | 8.2606122 | 16.5647873 | 51.68472 |
| Autentizační Framework | 5.023223 | 4.5587687 | 8.1413081 | 15.6254877 | 52.4423764 |

**Tabulka č. 1**: Znázorňuje čas (v sekundách) potřebný pro vykonání sady dotazů pomocí generického účtu v Entity Framework vs. autentizačního frameworku nad různými velikostmi tabulek (číslovky u jednotlivých názvu tabulek určují počet řádků dané tabulky)

**Graf č. 1:** Porovnání doby trvání vykonání dotazů v Entity Frameworku vs. Autentizačním Frameworku

Jak lze vidět, námi vytvořený framework dosahuje na databázích různé velikosti stejné rychlosti, jako původní Entity Framework.

**Možnost cachování frameworkem**

Druhotným cílem experimentu, bylo prozkoumat podporu cachování dotazů daným frameworkem. Samotný Entity Framework cachování dotazů nepodporuje. Existuje však několik knihoven postavených nad ním, které cachování dotazů umožňují.

Pro tento případ byla vybrána knihovna **Z.EntityFramework.Plus** (viz <http://entityframework.net/cache>), poskytující defaultní cachování bez nutnosti vlastní implementace.

Testování rychlosti pomocí cachování proběhlo nad stejnou databází jako testování výkonnosti. Pro účely testování bylo vytvořeno 9 jednoduchých SELECT dotazů.

V následující tabulce jsou uvedeny časy doby trvání vykonání dotazů pomocí autentizačního frameworku s použitím a bez použití cache a také uvedeno zrychlení při použití cache.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | users = 847 | users = 8470 | users = 84700 | users = 218500 | users = 847000 |
|  | posts = 100 | posts = 1000 | posts = 10000 | posts = 25000 | posts = 100000 |
|  | comments = 67 | comments = 670 | comments = 6700 | comments = 16750 | comments = 67000 |
| Bez cache | 4.7278847 | 4.9601139 | 6.3103762 | 8.914261 | 17.6982474 |
| S cachí | 4.2259629 | 4.830742 | 5.4163162 | 6.78546 | 11.368527 |
| Zrychlení o | 11.88% | 2.68% | 16.51% | 31.37% | 55.68% |

**Tabulka č. 1:** Znázorňuje čas (v sekundách) potřebný pro vykonání sady dotazů bez použití a s použitím cache a zrychlení při použití cache.

Z tabulky můžeme odvodit, že cachování je efektivní zejména u databází s větším počtem řádků v tabulkách.