HistoryStats Lite

Shrnutí

- Cílem je implementovat program počítající statistiky z jednoduchého textového logu instant messaging komunikace (ICQ, Facebook chat, ...)
- Výstupem programu je HTML soubor s vizualizacemi statistik pro každý kontakt
- Formát HTML je připraven, stačí jen při výpisu dodržet jeho strukturu
- Hlavní program přijímá jako parametr jméno souboru s logem, po zpracování všech statistik a jejich vypsání do HTML úspěšně skončí
- Vstupní soubor neobsahuje chyby
- Výstupní soubor se jmenuje stats.html

Statistiky

- Pro každý kontakt (např. Pepa, Honza, ...) evidujeme několik statistik, které vypisujeme do výsledného HTML souboru
- Tyto statistiky jsou
    - Počet přijatých a odeslaných zpráv
    - Poměr přijatých a odeslaných zpráv
    - Celkový počet slov v komunikaci s kontaktem
    - Nejčastější slova v komunikaci s kontaktem (nehledě na velikost písmen)
    - Histogram počtu zpráv s kontaktem vzhledem k hodinám (tedy pro každou hodinu celkový počet zpráv v komunikaci)
    - Vývoj komunikace od počátku sledovaného období do konce sledovaného období, po týdnech
- Za slovo považujeme jakoukoliv sekvenci znaků oddělených od ostatních mezerami a interpunkcí (uvažujme .,!?;)
- Za sledované období považujeme čas mezi první a poslední evidovanou zprávou (celkově, ne s konkrétním člověkem)

Agregované statistiky

- Kromě statistik pro jednotlivé uživatele zobrazujeme i celkovou statistiku kompletní historie
- Jednotlivé statistiky jsou stejné jako pro konkrétní uživatele
- Celková statistika je vypisována po statistikách všech uživatelů
- Do políčka pořadí je vložena hodnota "0."
- Do políčka se jménem je vložena hodnota "TOTAL"

Formát statistik jednoho uživatele

- Statistiky každého uživatele jsou do výsledného HTML souboru vypisovány coby řádek tabulky (<tr></tr>)
- Jednotlivá data jsou vypisovány do buněk tabulky (<td></td>)
- Buňky jsou v následujícím pořadí:
    - Hodnocení kontaktu (jeho pořadí vzhledem k celkovému počtu vyměněných zpráv) - např. 2.
    - Jméno kontaktu (Nickname) - např. Pepa
    - Celkový počet vyměněných zpráv s kontaktem
    - Vizualizace poměru příchozích a odchozích zpráv
        - Vizualizace používá zaokrouhlená procenta, která se počítají jako <Počet příchozích zpráv> / <Počet zpráv celkem>
        - Z hlediska HTML odpovídá následujícímu fragmentu, kde **X** odpovídá procentuální hodnotě:

```
<div class="outDiv"><div class="inDiv" style="width:X%;">X%</div></div>
```

    - Celkový počet slov ve zprávách s kontaktem
    - Pět nejčastějších slov oddělených čárkou a mezerou - např. "hoj, ahoj, jak, jasně, je"
        - Slova jsou v lowercase a slova stejné frekvence jsou v lexikografickém pořadí
    - Histogram zpráv pro konkrétní hodiny
        - Z hlediska HTML odpovídá následujícímu fragmentu, kde **X** a **Y** jsou hodnoty z intervalu <0, 60> představující výšku sloupce histogramu konkrétní hodiny (v pořadí 0 až 23):

```
<div class="barContainer">
        <div class="hoursContainer"><div class="hoursBar" style="height:Xpx;"></div></div>
        <div class="hoursContainer"><div class="hoursBar" style="height:Ypx;"></div></div>
        ...
</div>
```

        - Výška intervalu je dána poměrem počtu zpráv v konkrétní hodině ku počtu zpráv v nejvytíženější hodině
        - Zprávy se řadí dle hodinové části svého času, tj. zpráva s časem 22:13 je zařazena pod hodinu 22
        - Například tedy pokud nejvíce zpráv (např. 512) bylo vyměněno mezi 22 a 23 hodinou, tak hodina 12 s 128 zprávami má výšku 15 oproti výšce 60 hodiny 22
    - Průběh komunikace za sledované období
        - Zobrazuje počet zpráv v týdenních úsecích sledovaného období (poslední úsek může být kratší než týden)
        - Výška sloupců se počítá stejně jako u hodinového histogramu, tedy nejfrekventovanější týden má výšku 60, ostatní poměrně vůči maximu
        - Odpovídá následujícímu HTML fragmentu:

```
<div class="barContainer">
        <div class="timelineContainer"><div class="graphBar" style="height:Xpx;"</div></div>
        <div class="timelineContainer" ><div class="graphBar" style="height:Ypx;" ></div></div>
        ...
</div>
```

Formát vstupního souboru

- Vstupní soubor obsahuje záznam konverzací, konverzace je vždy s jedním kontaktem a obsahuje jednotlivé zprávy
- První řádek v konverzaci vždy obsahuje jméno kontaktu
- Předpokládá se, že zprávy v konverzaci, které neposílá kontakt, jsou od vlastníka logu (uživatele)
- Jednotlivé konverzace jsou odděleny separátorem ---
- Každá zpráva je dána hlavičkou ve formátu <Jméno odesílatele>, <Datum>
- Text zprávy následuje na další řádce za hlavičkou
- Předpokládáme, že text se nachází na jedné řádce a neobsahuje žádné zlé znaky
- Formát data odpovídá konvencím .NET typu DateTime
- Kódování souboru se předpokládá UTF-8

Příklad

Pepa
Pepa, 12.6.2012 17:22:37
Ahoj, jak je?
Pepa, 12.6.2012 17:35:12
Jsi tu? :)
Me, 12.6.2012 17:42:13
Ne.

Honza
Me, 2.12.2013 14:02:38
Kde jsi? Prednaska uz zacala :)

Další garance

- V textech zpráv se nikde neobjevuje sekvence ---
- Zprávy jsou vždy uvedeny na jednom řádku a neobsahují žádné extrémně zákeřné znaky (např. tabulátory)
- Vstupní soubor neobsahuje chyby. Způsob reakce programu na chybové vstupy je na autorovi programu, například tedy lze při libovolné chybě vypsat nějakou generickou chybovou hlášku nebo spadnout
- Uživatel není zlý a zadává to co má (tedy správné jméno souboru)
- Data se kompletně vejdou do paměti
- Mezi kontakty a konverzacemi nejsou duplicity

Hinty

- Pro reprezentaci času a data je nejlepší použít třídu DateTime
- DateTime.Parse umožňuje jednoduché načítání data a času ze stringové reprezentace
- DateTime.Parse si poradí s mezerami okolo data
- Třída TimeSpan představuje časový úsek konkrétní délky, jeho instance se dají přičítat k instancím DateTime, čímž vznikne nová instance DateTime
- Instance DateTime se dají porovnávat
- Všimněte si, že agregované statistiky (TOTAL) se jinak chovají jako běžný kontakt, ačkoliv mají data složená ze všech ostatních kontaktů
- Uvozovky lze do řetězce vložit pomocí escape značení - například "\"" je tedy řetězec obsahující jeden znak pro uvozovky
- Pro ignorování speciálních znaků v řetězci lze použít string literals - například @"\n" je tedy řetězec obsahující zpětné lomítko a n, nikoliv konec řádku