Міністерство освіти і науки України  
Тернопільський національний економічний університет  
Факультет комп’ютерних інформаційних технологій

Кафедра економічної кібернетики

Комплексне практичне індивідуальне завдання

з дисципліни «Конструювання ПЗ»

на тему:

«Засоби моніторингу ефективності роботи розробника»

Виконав :

студент групи ПЗС-31

Дериш І.Р.

Перевірив:

Сирник О. Й.

Тернопіль 2015

Зміст

Вступ

1. Системи керування версій, як засіб моніторингу ефективності роботи працівників
2. Існуючі засоби моніторингу ефективності роботи працівників
3. Власний засіб моніторингу ефективності роботи працівників

Висновки

Список використаних джерел

Вступ

Оскільки дедалі більше компаній надають своїм працівникам в якості робочого інструменту комп’ютер або взагалі практикують віддалену роботу, моніторингу ефективності роботи стає надзвичайно складним процесом, навіть проблемою. Це зумовлено тим, що за комп’ютером людина може виконувати різні дії: від виконання своїх робочих обов’язків до перегляду відео з кошенятами. На щастя, люди швидко збагнули, що за проблему слід братися одразу, адже своєчасне реагування шанс вирішення будь-якої проблеми значно зростає. В ході даної роботи я розгляну декілька існуючих засобів моніторингу ефективності робітників та представлю власний засіб, розроблений таким чином, щоб бути максимально простим у використанні кінцевим користувачем та підходити до будь-якої предметної області.

1. Системи керування версій, як засіб моніторингу ефективності роботи працівників

Для початку наведу короткі відомості про системи керування версій.

Система керування версіями (англ. source code management, SCM) — програмний інструмент для керування версіями одиниці інформації: вихідного коду програми, скрипту, веб-сторінки, веб-сайту, 3D моделі, текстового документу тощо. Система керування версіями — це потужний інструмент, який дозволяє одночасно, не заважаючи один одному, здійснювати роботу над груповими проектами. Системи керування версіями зазвичай використовуються при розробці програмного забезпечення для відстеження, документування та керування над поступовими змінами в електронних документах: у вихідному коді, кресленнях, електронних моделях та інших документах, над змінами яких одночасно працюють декілька людей. Кожна версія позначається унікальною цифрою чи літерою, зміни документу занотовуються. Зазвичай також зберігається автор зробленої зміни та її час. Інструменти для керування версій входять до складу багатьох інтегрованих середовищ розробки. Система керування версіями існують двох основних типів: з централізованим та розподіленим сховищем.

На перший погляд може здатися, що системи керування версіями не мають нічого спільного з моніторингом ефективності роботи працівника, однак це не так. Кожна зміна, що реєструється системою, окрім дати та власне змін містить також дані про автора, що допомагає оцінити об’єм робіт, який виконав кожен працівник.

Оскільки використання системи керування версіями для моніторингу ефективності роботи працівника можливе лише в ІТ-індустрії, нижче будуть розглянуті більше універсальні засоби.

2. Існуючі засоби моніторингу ефективності роботи працівників

Розгляну три існуючих засоби:

1. Yaware.TimeTracker
2. ZKTeco
3. Wrike

Системи Wrike та Yaware.TimeTracker пропонують клієнтам однакові можливості: моніторинг ефективності роботи за робочим місцем, відстеження місця розташування працівника за допомогою мобільного телефону і т.д. ZKTeco на додачу до цього пропонує також біометричні засоби фіксації часу прибуття та залишання робочого місця. На жаль, з принципами роботи даних систем познайомитися не вдалося, адже це комерційна таємниця. Однак, можна зазначити, що у них однотипні системи перегляду зібраних даних – веб-додаток, який дозволяє переглядати таблиці з даними.

3. Власний засіб моніторингу ефективності роботи працівників

Я розробив власний засіб моніторингу ефективності роботи працівників для ОС Windows. Дана система збирає дані про всі дії, які виконував користувач протягом робочого дня, а також точний час початку та кінця робочого дня. Звіт надсилається на вказану клієнтом електронну адресу у вигляді структурованого текстового файлу – це полегшує процес перегляду даних та дозволяє переглядати звіт практично з усіх пристроїв. У випадку відсутності з’єднання з мережею, система автоматично зберігатиме файли з даними і за першого ж успішного підключення до мережі надішле звіти. Після тестування встановлено, що дана система сумісна з ОС Windows 7, Windows 8, Windows 10. Принцип роботи системи базується на фундаментальних основах роботи ОС Windows. Без перебільшення можна сказати, що середньостатистичний користувач не зможе «обманути» дану систему і всі його дії будуть зафіксовані. Знизу будуть наведені результати роботи та вихідний код даної системи.

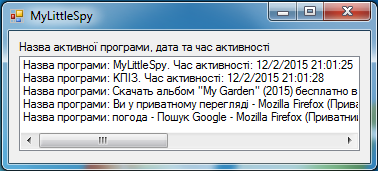


Рис.1 – вікно активної клієнтської частини системи

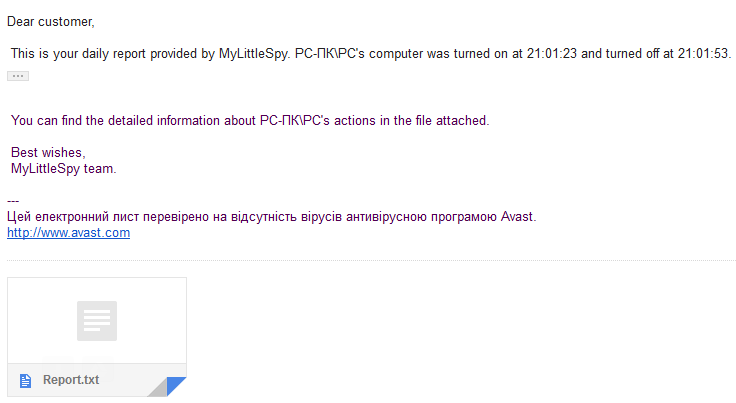


Рис. 2 – повідомлення-звіт, надіслане системою ( перегляд через браузер ПК )

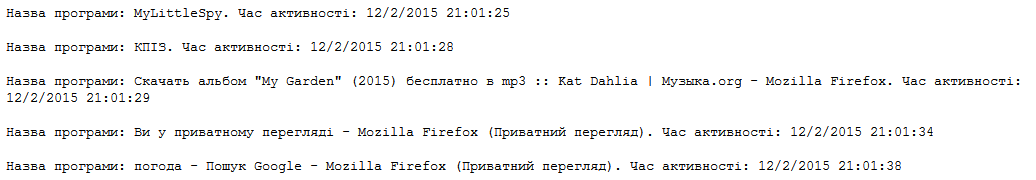


Рис. 3 – фрагмент файлу звіту (перегляд через браузер ПК )

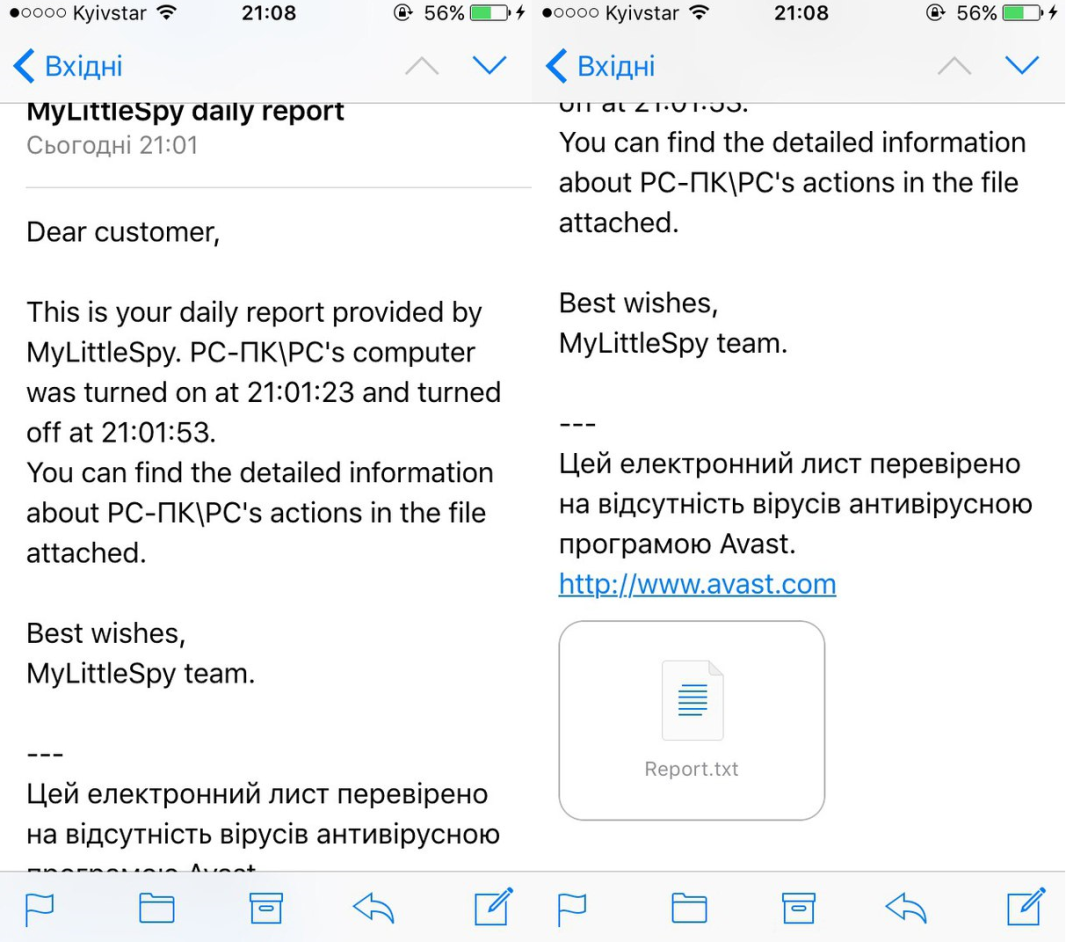


Рис. 4 – повідомлення-звіт, надіслане системою ( перегляд через браузер мобільного телефону )

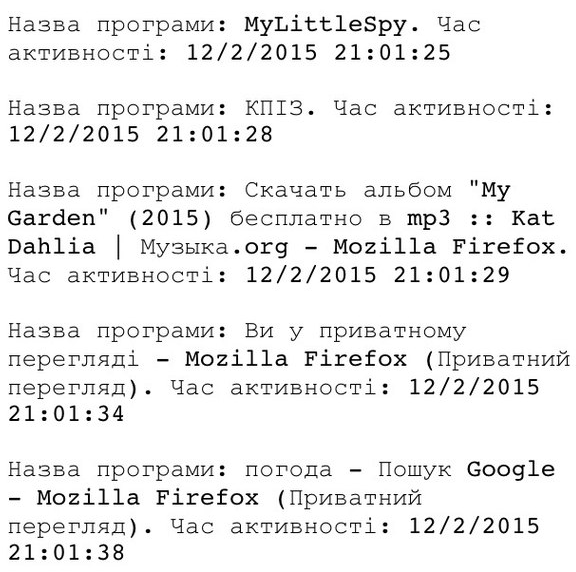


Рис. 5 – фрагмент файлу звіту (перегляд через браузер мобільного телефону )

Переглянувши вищенаведені знімки екрану, можна помітити два ключові моменти:

1. система реєструє навіть дії, які виконуються у «приватному» режимі браузера
2. робота з системою є надзвичайно простою, адже користувачеві потрібно лише один раз встановити її на комп’ютер і потім отримувати звіти, які з легкістю переглядаються на будь-якому пристрої.

Висновки

Проаналізувавши предметну область та наявні засоби моніторингу ефективності роботи працівників, можна впевнено сказати, що даний сегмент ІТ-ринку є надзвичайно перспективним, адже кількість підприємств, що потребують таких систем стрімко зростає, породжуючи попит. Зважаючи на те, що не всі компанії можуть дозволити собі, як правило, дорогі та нераціонально поділені на пакети продукти від «іменитих» розробників, даний сегмент ринку є непоганою стартовою площадкою для компаній-початківців. Також я вирішив внести деякі корективи в свою систему та спробувати знайти можливість протестувати її в «бойових» умовах – на якому-небудь підприємстві.

Список використаних джерел

1. Jeffrey Richter – CLR via C# (4th Edition) (Developer Reference)
2. Herbert Schildt – C# 4.0 The Complete Reference
3. Microsoft Developer Network (MSDN)
4. Веб-сайт Уaware <http://timetracker.yaware.com.ua/>
5. Веб-сайт Wrike <https://www.wrike.com/ru/>
6. Веб-сайт Zkteco <http://zkteco.com.ua/company.htm>