

УДК 004.42

В.І. КОЦУН

Львівська філія ПВНЗ «Європейський університет»

ПРОГРАМНИЙ КОМПЛЕКС ДЛЯ КОНТРОЛЮ ТА ПІДВИЩЕННЯ ЕФЕКТИВНОСТІ РОБОЧОГО ЧАСУ В ІТ КОМПАНІЇ

В даній роботі розроблений програмний комплекс для контролю робочого часу та підвищення ефективності його використання в ІТ компанії. Програмний комплекс являє собою вдосконалену методологію планування та управління проектом разом із програмним забезпеченням, що полегшує впровадження та використання даного підходу. Розглянуті існуючі методології управління та планування проектів в ІТ галузі, визначені їхні недоліки та здійснений пошук покращення цих недоліків. Запропонований та розроблений алгоритм планування проекту, який вирішує проблему встановлення великої кількості неефективних запасів. Суть алгоритму полягає у визначенні критичного шляху проекту, налаштуванню паралельних додаткових шляхів, та встановлення буферу у «вузькому місці», в точці де є найбільш необхідне його застосування. Були виявлені недоліки класичної діаграми Ганта та запропонована модифікація, яка включає в себе введення нових понять, таких як: множинні проекти, реалістичний показник, запланований показник, контекста лінія. Розроблене програмне забезпечення для впровадження та використання методології, описано його програмна архітектура та опис реалізації. Програмне забезпечення розроблене як веб-орієнтована система на основі архітектури клієнт-сервер. В процесі роботи проведено дослідження та впровадження методології, виявлені його основні переваги та обмеження. Результати досліджень можуть бути цікавим представникам ІТ галузі, стартап-інкубаторам, та акселераторам, оскільки результати та описані підходи дозволяють зменшити витрати на розробку програмного забезпечення та проектів.

Ключові слова: програмний комплекс, ІТ, контроль робочого часу, клієнт-сервер, веб-орієнтована система, розробка програмного забезпечення.

В.И. КОЦУН

Львовский филиал ЧВУЗ «Европейский университет»

ПРОГРАММНЫЙ КОМПЛЕКС ДЛЯ КОНТРОЛЯ И ПОВЫШЕНИЯ ЭФФЕКТИВНОСТИ РАБОЧЕГО ВРЕМЕНИ В ИТ КОМПАНИИ

В данной работе разработан программный комплекс для контроля рабочего времени и повышения эффективности его использования в ИТ компании. Программный комплекс представляет собой усовершенствованную методологию планирования и управления проектом вместе с программным обеспечением, облегчая внедрение и использование данного подхода. Рассмотрены существующие методологии управления и планирования проектов в ИТ отрасли, определены их недостатки и осуществлен поиск улучшения этих недостатков. Предложен и разработан алгоритм планирования проекта, который решает проблему установления большого количества неэффективных запасов. Суть алгоритма заключается в определенных критического пути проекта, подстройке параллельных дополнительных путей, и установка буфера в «узком месте», в точке где есть самое необходимое его применение. Были выявлены недостатки классической диаграммы Ганта и предложена модификация, которая включает в себя введение новых понятий, таких как: множественные проекты, реалістичний показател, запланований показател, контекста лінія. Разработанное программное обеспечение для внедрения и использования методологии, описаны его программная архитектура и описание реализации. Программное обеспечение разработано как веб-ориентированная система на основе архитектуры клиент-сервер. В процессе работы проведено исследование и внедрение методологии, обнаружены его основные преимущества и ограничения. Результаты исследований могут быть интересным представителям ИТ отрасли, стартап-инкубаторам и акселераторам, поскольку результаты и описаны подходы позволяют уменьшить затраты на разработку программного обеспечения и проектов.

Ключевые слова: программный комплекс, ИТ, контроль рабочего времени, клиент-сервер, веб-ориентированная система, разработка программного обеспечения.

**SOFTWARE COMPLEX FOR CONTROL AND WORKING
EFFICIENCY IMPROVEMENT IN IT COMPANY**

In paper, for working time controlling and increasing the efficiency in IT company using the software complex is developed. The developed software package is an advanced methodology for planning and project management. Based on developed software it is give the possibilities to facilitate the implementation and using this methodological approach. The existing methodologies of management and projects planning in the IT industry are considered. Their deficiencies are identified and the search for improvement of these shortcomings is made. project Planning algorithm solved the problem of installing a large number of inefficient stocks is proposed and developed. The essence of the algorithm is to determine the critical path of the project, to configure the parallel additional paths, and to set the buffer in a "bottleneck", at the point where it is most needed for its application. The disadvantages of the classic Gantt chart were identified and the proposed modification, which includes the introduction of new concepts. such as: multiple projects, realistic indicator, planned indicator, contextual line. The software for implementing and using the methodology is developed, its software architecture and implementation description are described. The software is developed as a web-based system based on client-server architecture. In the course of work, the research and implementation of the methodology have been carried out, its main advantages and limitations have been identified. Research results can be of interest to IT industry, startup incubators and accelerators, as the results and described approaches reduce the cost of developing software and projects.

Keywords: software complex, IT, time control, client-server, web-oriented system, software development.

Постановка проблеми

Продуктивність робочого часу та контроль його ефективного використання – це проблема, яка появилася у програмній інженерії одночасно із переходом даної галузі у комерційне русло. Для вирішення даного питання було винайдено багато методологій та програмних засобів.

Аналіз останніх досліджень і публікацій

Однією із найперших методологій була методологія WaterFall [1], яка передбачала чітку послідовність усіх процесів та строго виконання наступного завдання лише після завершення попереднього. Ще одна із методологій того часу RUP [2], особливістю якої була наявність супроводжування відповідними програмними комплексами для розробки ПЗ.

Після цього на зміну «громістких» методологій приходять гнучкі, так звані Agile-методології [3] до таких належить Scrum [4], Kanban [5]; розробка ПЗ стає більш гнучкою та швидшою порівняно із попередніми роками. Нові методології дозволяють підвищувати швидкість розробки та ефективність використання людського ресурсу на проекті. Проте з'являються нові проблеми такі як недостатня сфокусованість менеджера, чи неефективне використання людського ресурсу в контексті проекту.

До програмного забезпечення, яке частково вирішує обрану проблему належить:

- Atlassian JIRA - система відслідковування помилок, призначена для організації спілкування з користувачами та управління проектами [6].
- Redmine – система баг-трекінгу для управління проекту та фіксації часу [7].
- Microsoft management studio – програма для планування проектів від компанії Microsoft [8].
- Програмний комплекс In Time від компанії AxaSoft, який дозволяє планувати та відслідковувати проект тісно прив'язуючись до методології Scrum [9].

Одна із задач, що не вирішується наявними методами це перспектива планування декількох проектів із розподілим людським ресурсом. У плані людського ресурсу вважаємо час спеціаліста. Наприклад, жодна з цих методологій не враховує задіяння одного і того ж працівника на кількох проектах із врахуванням його фізичної зайнятості, переключення контекстів та навантаженні на кожному з проектів.

Жодна із методологій та існуючих засобів ПЗ не враховує бюджетної складової проекту та планування проекту у фінансовому плані, як правило фінансове планування повністю відділено від загального планування проекту.

Ще одна проблема, яку має вирішити даний програмний комплекс – це порівняння запланованого часу із фактично витраченим та його динамічне корегування.

Формулювання мети дослідження

Створення програмного комплексу для підвищення ефективності використання робочого часу. Показником ефективності буде збільшення кількості успішних проектів у ІТ компаніях, збір статистичних та аналітичних даних.

Викладення основного матеріалу дослідження

В описі будь-якої системи передбачається конкретизація її вектора виходу. На рівні підприємства це вектор номенклатури продукції, що випускається (наприклад, Y). Тоді величина різноманітності на виході системи складає $H = 2^Y$ станів, кожний з яких є варіантом структури виробничої програми. Система управління у відповідності з законом необхідної різноманітності повинна мати не меншу різноманітність, тобто мати оцінки для H станів. Ця множина станів утворює системну міру управління. Проте на практиці вибирається й обґрунтовується один варіант із усієї множини, що можна охарактеризувати як системну норму управління. У результаті ($H-1$) станів утворюють структурний допуск на відхилення від норми управління.

Системна норма управління – це загальна мета компанії, на яку орієнтовані всі показники виробничої діяльності. У загальному вигляді її можна уявити сукупністю елементів:

- вектора номенклатури й обсягу продукції з допусками на відхилення:

$$N = \begin{pmatrix} N_1 \pm \delta_{N1} \\ N_2 \pm \delta_{N2} \\ \dots \\ N_m \pm \delta_{Nm} \end{pmatrix} \quad (1)$$

- вектора можливих станів підприємства:

$$S = \begin{pmatrix} S_1 \\ S_2 \\ \dots \\ S_k \end{pmatrix} \quad (2)$$

- матриці ймовірностей переходу зі стану в стан:

$$M = \begin{matrix} S_1 \\ S_2 \\ \vdots \\ S_k \end{matrix} \begin{pmatrix} P_{11} & P_{12} & \dots & P_{1k} \\ P_{21} & P_{22} & \dots & P_{2k} \\ \dots & \dots & \dots & \dots \\ P_{k1} & P_{k2} & \dots & P_{kk} \end{pmatrix} \quad (3)$$

- матриці основного економічного показника, наприклад, прибутку (або декількох показників) із допусками на відхилення:

$$P = \begin{matrix} S_1 \\ S_2 \\ \dots \\ S_k \end{matrix} \begin{pmatrix} P_{11} \pm \delta_{11} & P_{12} \pm \delta_{12} & \dots & P_{1k} \pm \delta_{1k} \\ P_{21} \pm \delta_{21} & P_{22} \pm \delta_{22} & \dots & P_{2k} \pm \delta_{2k} \\ \dots & \dots & \dots & \dots \\ P_{k1} \pm \delta_{k1} & P_{k2} \pm \delta_{k2} & \dots & P_{kk} \pm \delta_{kk} \end{pmatrix} \quad (4)$$

- матриці часу перебування системи в тому або іншому стані з допусками на відхилення:

$$T = \begin{matrix} S_1 \\ S_2 \\ \vdots \\ S_k \end{matrix} \begin{pmatrix} T_{11} \pm \delta_{t11} & T_{12} \pm \delta_{t12} & \dots & T_{1k} \pm \delta_{t1k} \\ T_{21} \pm \delta_{t21} & T_{22} \pm \delta_{t22} & \dots & T_{2k} \pm \delta_{t2k} \\ \dots & \dots & \dots & \dots \\ T_{k1} \pm \delta_{tk1} & T_{k2} \pm \delta_{tk2} & \dots & T_{kk} \pm \delta_{tkk} \end{pmatrix} \quad (5)$$

Необхідність введення статистичних допусків на відхилення для елементів норми управління пояснюється умовами невизначеності як зовнішнього, так і внутрішнього середовища підприємства. [10]

Планування будь-якого проекту ґрунтується на попередній його оцінці та зважуванні ресурсів компанії. Найголовнішим чинником в даному процесі є визначення модулів та встановлення критичного шляху проекту та відокремлення другорядних.

Після визначення критичного та додаткових шляхів проекту кожен із етапів оцінюється окремо, залежно від методології розробки програмного забезпечення у годинах чи по іншому. Очевидно, що до оцінки проектів варто залучати експертів даної інженерної галузі в контексті якої відбувається проект.

Наступна фаза це закладання буферу запасу на проект. Після визначення буферу проект можна запускати проте необхідно робити дані ітераційні дії кожного разу після завершення певного етапу проекту на критичному шляхові.

Запропонована система має вирішити ряд задач:

- планування проектів;
- контроль виконання проектів;
- обрахунок використання коштів.

До додаткових показників можуть належати:

- мотивація команди;
- рівень загального навчання;
- отримання досвіду компанії та інші.

Тобто менші терміни проекту, та менші фінансові витрати при сталій якості ПЗ буде вважатися досягненням результату.

Отже вихідні дані програмного комплексу:

- запланований час проекту;
- фактичний час проекту;
- заплановані витрати проекту;
- фактичний час проекту.

Вхідні дані представляють собою текстову інформацію, що вводиться вручну в певні спеціально для цього призначені місця системи, зберігаються в базі даних та правильним чином відтворюються.

Програмний комплекс надає інтерфейс, що дозволяє в ручному режимі вводити дані.

Вихідні дані на основі яких можна буде робити порівняльний аналіз можна буде представити у вигляді графіків та таблиць, що відобразатимуться у різних перспективах.

Розроблена система повинна бути крос-платформною та легкодоступною, тому було рішення, зробити її на основі архітектури клієнт-сервер. Для реалізації були використані наступні технології та засоби:

- Мова програмування PHP 5;
- Мова розмітки HTML 5 та стилізація CSS3;
- Мова програмування JavaScript;
- Framework jQuery;
- Реляційна база даних MySQL;
- Мова запитів до БД SQL;
- Технологія AJAX.

Список саме цих технологій обґрунтовується тим, що вони знаходяться у відкритому доступі тобто OpenSource, зв'язка цих технологій разом давно відома і добре працює, підтримка даних технологій на серверах не несе за собою додаткових фінансових витрат.

Для написання бізнес логіки проекту використовується мова програмування PHP 5. Для роботи із базою даних використовується мова SQL. Для візуального оформлення і привабливого вигляду програми були використані каскадні стилі CSS.

Як зазначалося вище для реалізації програмного комплексу була обрана архітектура клієнт-сервер. Один із важливих етапів проектування системи є створення та проектування структури бази даних зображеної на рис. 1.

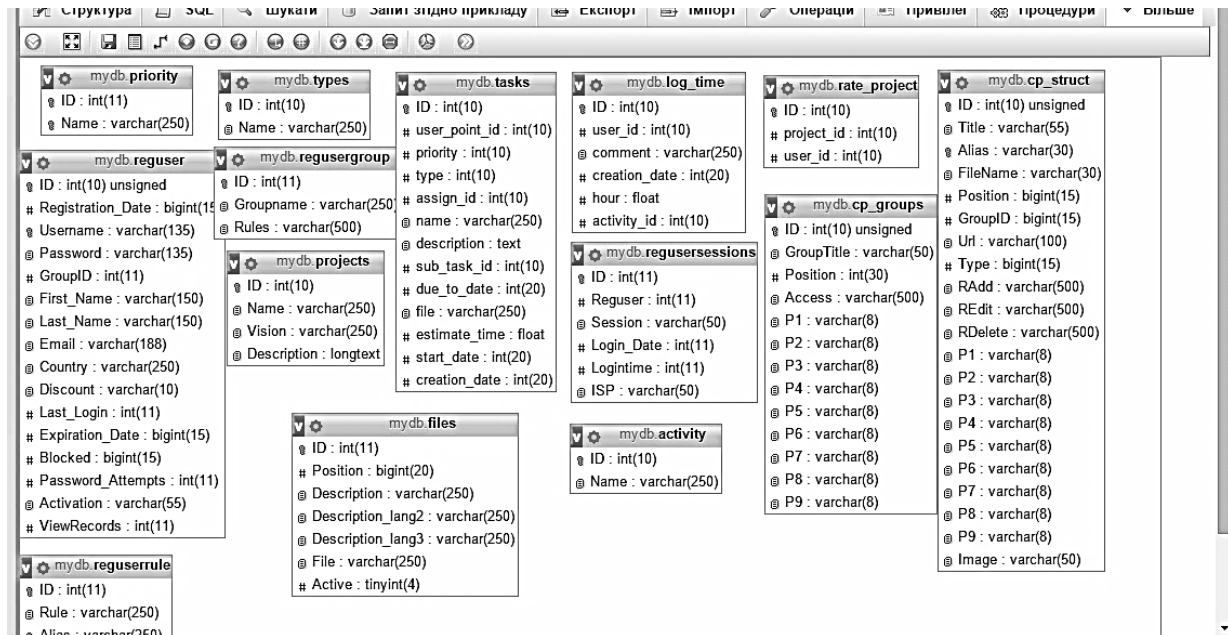


Рис. 1. Загальна схема БД

База даних спроектована таким чином, щоб уникнути надлишковості та бути простою та зручною у користуванні із програмного інтерфейсу.

Для користування програмним засобом його потрібно встановити на веб-сервер та передбачається, що кожна окрема компанія, буде встановлювати свій веб-сервер із окремою копією системи.

Система може працювати у режимах декількох основних акторів, менеджера проекту та працівників. Також присутній додатковий актор, що може змінювати конфігурацію системи, наприклад добавляти типи активностей чи діяльності. Кожна компанія може мати свої власні налаштування тому такий додатковий функціонал є доцільним.

Розглянемо спершу систему із точки зору менеджера. При першому вході у систему менеджер вводить пароль та логін наданий йому адміністратором системи, після того менеджер може поміняти свій власний пароль.

Згідно специфікації вимог менеджера доступні наступні функції:

- керування користувачами;
- керування проектами;
- перегляд звітності.

Для керування користувачами менеджер повинен зайти у розділ користувачі. Далі у нього є можливість вибору дії CRUD-операцій над користувачами. Про працівника вводяться усі дані, що вимагає база даних.

Керування проектами – це один із основного функціоналу системи, він включає в себе:

- можливість створення проекту;
- можливість виділення людей на проект;
- можливість створення задач до проекту;
- можливість оцінки кожного окремого проекту.

Для цього менеджерові потрібно зайти в розділ керування проектами вибрати необхідний проект та заповнити дані про нього. Також в цьому ж розділі менеджер має можливість створити задачі до проекту призначити її до конкретної людини та оцінити час її виконання. Система автоматично на основі даних оцінок дасть можливість перегляду планування проекту.

Скріншот роботи із задачами менеджера зображений на рис. 2.

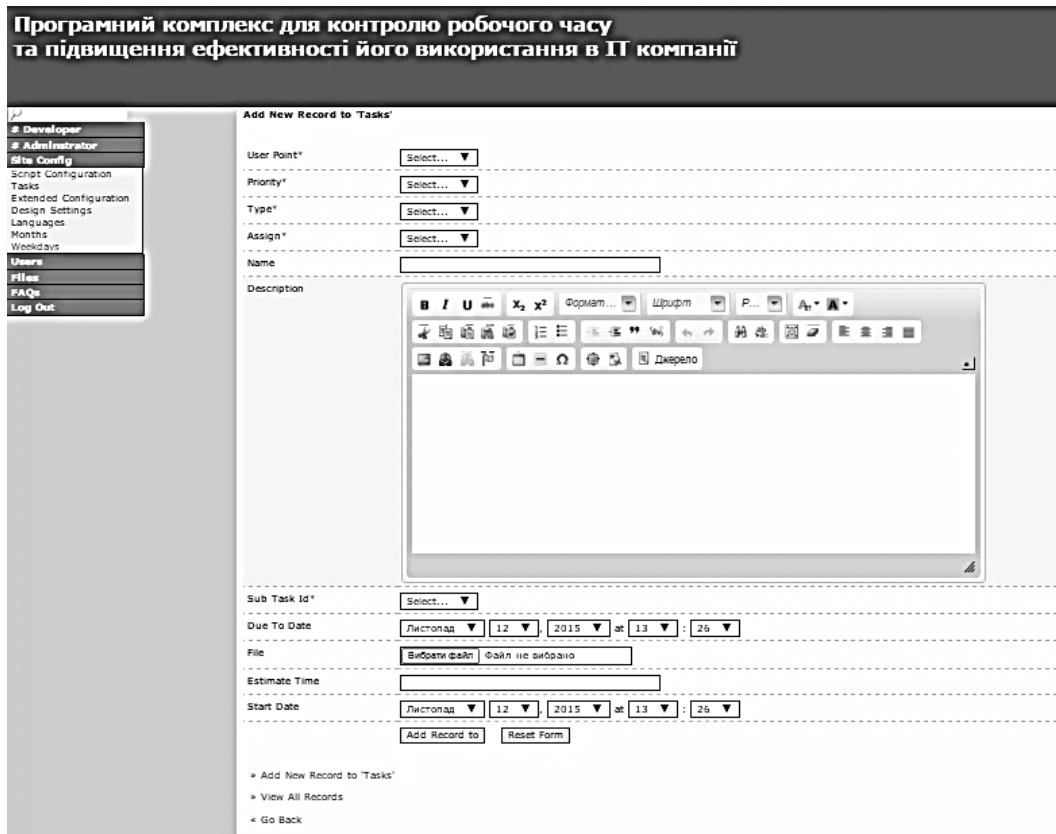


Рис. 2. Створення задачі менеджером

У менеджера також є функція перегляду звітності, яка дозволяє переглядати звітність користувачів у табличному вигляді, а також у вигляді вдосконаленої діаграми Ганта.

Іншим типом користувачів системи є працівники компанії у них доступні наступні функції:

- перегляд задач доступних для їхнього виконання із сортуванням по пріоритетності
- можливість звітності згідно виконаних задач
- перегляд власної статистики

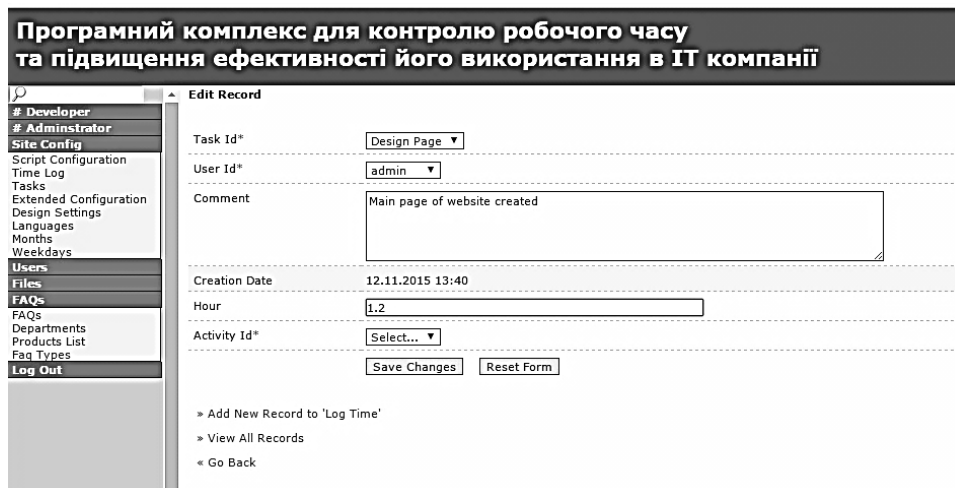


Рис. 3. Логування задачі працівником

Для того, щоб працівник міг увійти в систему, попередньо менеджер повинен згенерувати пароль та логін для нього. Після цього працівник може користуватися системою. Для того, щоб переглянути списки задач, які йому поставлені по проектах працівник повинен зайти у розділ «список задач». Після виконання задачі, працівник може залогувати час, який він потратив на повне або часткове

вирішення задачі, таким чином у системі завжди знаходяться актуальні дані і можуть надати звітність в будь-який потрібний момент часу на рис. 3 зображено скріншот логування задачі працівником.

Також працівник може переглянути час власної статистики на проекті, це така ж функція як у менеджера проте із обмеженням перегляду лише по власному логіну.

У системі присутні також додаткові функції такі, як конфігурація самої системи, конфігурація користувачів, проектів та їх параметрів.

Висновки

В результаті проведеної роботи розроблене програмне забезпечення для контролю робочого часу та підвищення ефективності його використання в ІТ компанії та описана його програмна архітектура і опис реалізації. Програмне забезпечення розроблене як веб-орієнтована система на основі архітектури клієнт-сервер. Проведене дослідження та впровадження методології дозволило виявити його основні переваги, а саме:

- Методологія є добре застосованою до малих та середніх проектів, які відносяться до категорії fix-price, зменшення вартості становить 12%-17%, а строків 7 – 10%.
- Економія бюджет проекту, за рахунок паралельної роботи над декількома проектами.
- Методологія є краще застосовною для веб-орієнтованих проектів.

Результати роботи можуть бути цікавим представникам ІТ галузі, стартап-інкубаторам та акселераторам, оскільки результати та описані підходи дозволяють зменшити витрати на розробку програмного забезпечення та проектів.

Список використаної літератури

1. Модель водоспад. [Електронний ресурс]. – Режим доступу: https://en.wikipedia.org/wiki/Waterfall_model.
2. RUP. [Електронний ресурс]. – Режим доступу: https://uk.wikipedia.org/wiki/Rational_Unified_Process.
3. Agile. [Електронний ресурс]. – Режим доступу: <http://agile.org.ua/>.
4. Scrum. [Електронний ресурс]. – Режим доступу: [https://en.wikipedia.org/wiki/Scrum_\(software_development\)](https://en.wikipedia.org/wiki/Scrum_(software_development)).
5. Kanban. [Електронний ресурс]. – Режим доступу: [https://en.wikipedia.org/wiki/Kanban_\(development\)](https://en.wikipedia.org/wiki/Kanban_(development)).
6. Jira Atlassian. [Електронний ресурс]. – Режим доступу: <https://jira.atlassian.com/secure/Dashboard.jspa>.
7. Redmine. [Електронний ресурс]. – Режим доступу: <http://www.redmine.org/>.
8. Microsoft Project Management Studio. [Електронний ресурс]. – Режим доступу: <https://msdn.microsoft.com/en-us/library/mt238290.aspx>.
9. InTime. [Електронний ресурс]. – Режим доступу: <http://www.axosoft.com/>.
10. Климова Т.Н. Расчет и анализ производственного цикла // Проблемы праці, економіки та моделювання: зб. наук. пр., ч.2. – Хмельницький: ТУП, 1997. – С.171-174.