

териале. Проведенные экспериментальные исследования для определения скорости распространения акустической волны в древесине. На основе полученных результатов предложена методика выявления дефектов в древесине акустическим методом.

Ключевые слова: акустические измерения, древесина, неразрушающий контроль, звуковая карта, программное обеспечение.

Storozhuk O.L., Turash R.I., Kens I.R., Sokolowsky Ya.I. Using of information technologies for wood defects detection by the acoustic method

The software is developed here for improvement and practical implementation of an acoustic method for detection of imperfections in timber. This will make it possible to define transit time of a test sonic signal in a stuff by direct measurement. The experimental research to define the velocity of sonic wave propagation in wood was conducted. Obtained results made it possible to propose the technique of wood defects detection by the acoustic method.

Keywords: acoustic measurement, wood, nondestructive control, a sound card, the software.

УДК 004.4 **Викл. В.В. Яцишин – Тернопільський НТУ ім. Івана Пулюя; доц. О.Г. Харченко, канд. техн. наук – Національний АУ, м. Київ**

CASE-ТЕХНОЛОГІЯ РОЗРОБЛЕННЯ ВИМОГ ДО ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ТА ОЦІНЮВАННЯ ЙОГО ЯКОСТІ

Наведено результати дослідження сучасних CASE-технологій, які використовують для підтримання процесів проектування на різних етапах життєвого циклу програмного забезпечення (ПЗ). Внаслідок аналізу виявлено низку недоліків цих технологій та запропоновано власний CASE-засіб, який є web-орієнтованим і спрямований на підтримку технології, базованої на моделі якості стандарту ISO 9126. Показано, що використання цієї моделі, відповідної технології та засобу дає змогу адекватно відобразити потреби замовника та користувачів ПЗ на специфікації вимог, а також визначити міру їх задоволення. Пропонований CASE-засіб дає змогу автоматизувати процес розроблення вимог і керування ними, що значно підвищує якість процесу проектування і скорочує часові межі виконання проекту.

Вступ. Сучасний стан розвитку інформаційних систем (ІС) характеризується постійним зростанням їхньої складності. Це потребує залучення значних ресурсів для забезпечення ефективності їх функціонування. Хоч на цей час розроблено цілу низку методів та методологій проектування ІС, як корпоративних, так і загального використання, але за їх застосування часто виникають проблеми, пов'язані з недостатньою формалізацією процесів проектування. А це своєю чергою перешкоджає впровадженню засобів автоматизації цих процесів.

Тому розроблення формалізованих процедур реалізації процесів проектування ПЗ є актуальною задачею інженерії програмного забезпечення.

Постановка задачі. Технологія створення ПЗ базується на процесах ЖЦ [1], які за своєю природою є досить складними, трудомісткими і творчими. У зв'язку з цим виникає потреба створення CASE-технологій, які б давали змогу автоматизувати операції та стадії як основних, так і допоміжних процесів ЖЦ. Розроблення таких технологій є актуальною задачею, оскільки

їх застосування підвищує ефективність процесу розроблення програмних продуктів приблизно у 6 разів [2].

Аналіз процесів ЖЦ розроблення ПЗ показав, що найменш формалізованими і тому найбільш трудомісткими є процеси на етапі аналізу вимог та оцінювання якості [3, 4]. Це пояснюється складністю використання формальних методів для відображення потреб замовника на специфікації системних вимог, що породжується відсутністю єдиного уніфікованого їх подання [5]. Чинний стандарт IEEE830 дає достатньо довільні трактування структури та форми представлення вимог, що відображається у неоднозначності та неузгодженості у сформульованих вимогах. Звідси випливає, що для створення ефективних CASE-засобів автоматизації процесів розроблення специфікацій вимог, їх контролю та управління необхідно розробити стандартизовані уніфіковані моделі їх представлення і процедури їх побудови. Оскільки моделі вимог якості, які базуються на уніфікованих компонентах і містять критерії оцінювання, є основою для процедур оцінювання якості, то вони також можуть бути базою для створення CASE-засобів оцінювання якості ПЗ.

Огляд існуючих відомостей. Виконаємо аналіз технологій розроблення ПЗ та CASE-засобів, які їх підтримують на етапах ЖЦ. При цьому основну увагу зосередимо на етапі розроблення вимог та оцінювання якості. На практиці використовують низку технологій розроблення ПЗ. Найширше використовуваними є такі:

- Microsoft Solutions Framework (MSF).
- Custom Development Method Oracle (CDMO).
- Rational Unified Process (RUP).

Ці технології базуються на двох основних підходах: структурному та об'єктно-орієнтованому. Технологія MSF є платформно-незалежною, яка орієнтована на розроблення ПЗ і розвиток інформаційної інфраструктури [6]. Засоби цієї технології підтримують розподілені обчислення та застосування технології "клієнт-сервер".

Під час проектування вимог до ПС MSF використовує метод шаблонів та UML діаграм. Засобами графічного моделювання та документування вимог ПЗ є Microsoft Visio та пакет Microsoft Office. Microsoft Visio підтримує генерацію UML-діаграм, зокрема Use case діаграм. Для управління проектом та контролю загального процесу створення ПЗ використовують засіб Microsoft Project.

Технологія RUP базується на принципах об'єктно-орієнтованого підходу створення ПС та мови графічного моделювання UML. Підтримку цієї технології забезпечують засоби фірми IBM групи Rational. Для розроблення вимог за об'єктно-орієнтованим підходом використовують діаграми класів та Use case діаграми, що відображають функціональність майбутньої ПС.

Засобом, що підтримує моделювання діаграм цього типу є Rational Rose. Інший засіб керування вимогами та їх документування – середовище Rational Requisite Pro. Основне призначення цього засобу полягає у наданні можливості відслідковування та зручного внесення змін у вимоги. При цьому вимоги представляють у текстовому вигляді з відображенням діаграм, змодельованих в Rational Rose.

Об'єктно-орієнтовний підхід розроблення вимог втілений також в автоматизованому засобі DOORS компанією Telelogic. Принцип проектування вимог відображає структуру шаблону, що рекомендований у стандарті [3]. В основі технології CDMO, згідно з [6], лежить метод ORACLE CASE* METHOD, який базується на визначенні об'єктів (сутностей) та зв'язків між ними, що фактично є структурним підходом. Технологія CDM підтримується інструментальними засобами компанії Oracle і використовується під час створення автоматизованих інформаційних систем на основі реляційних баз даних. Для розроблення та керування вимогами використовується засіб автоматизації Oracle Designer. Цей засіб дає змогу моделювати діаграми "сутність-зв'язок". З його допомогою можна лише визначати основні сутності всередині системи та зв'язки між ними, але неможливо описати систему загалом та процеси, які в ній відбуваються. В Oracle Designer відсутня можливість представлення вимог якості та обмежень, що істотно звужує область його застосування. Засобами автоматизації, які підтримують принципи структурного підходу щодо проектування ПС, є також програмні продукти ARIS Toolset, ERwin Datamodeler, Process Modeler (BPwin).

Виходячи з результатів проведеного аналізу можна стверджувати, що тільки технології MSF та DOORS мають засоби формалізованого представлення вимог якості, які використовують метод шаблонів. Однак структура і класифікація рубрик шаблонів не уніфіковані, тому у разі їх використання можуть виникати неоднозначності трактувань, що істотно звужує область застосування цих CASE-технологій.

Для перевірки відповідності готового програмного продукту заявленим у специфікації вимогам фірми-розробники використовують автоматизовані засоби тестування. Засоби, які б здійснювали автоматизацію технологічних операцій в процесі оцінювання якості ПЗ відсутні.

У роботах [5,7] містяться основні положення та обґрунтування технології побудови специфікацій вимог та оцінювання якості ПЗ, яка базується на використанні концепції моделі якості ISO 9126 [4]. Ця технологія забезпечує формалізацію процесів відображення потреб замовника на специфікації вимог в уніфікованому, стандартизованому вигляді і використання побудованих специфікацій при оцінюванні якості. У цій роботі розроблено CASE-засіб, який автоматизує процеси цієї технології.

Основна частина. Стандартом [8] визначено три типи вимог щодо якості ПЗ, а саме вимоги у використанні, внутрішньої та зовнішньої. Процес починається з побудови вимог щодо якості у використанні. Для цього фіксуються потреби користувача бізнес-системи, для якої замовляється WEB-продукт.

$$P = \{P_i, C_{ik}\}, \quad i = \overline{1, N}, \quad K = \overline{1, M_i}, \quad (1)$$

де: P_i – i -та потреба користувача; C_{ik} – обмеження на i -ту потребу; N – кількість потреб замовника; K – кількість обмежень на потреби.

Виходячи з бізнес-вимог та вимог предметної області, для кожної потреби P_i задається множина атрибутів $\{A_{iK}\}$, $K = \overline{1, S_i}$, які відображають ступінь задоволення i -тої потреби. В результаті отримуємо сукупність

$$\{P_i, A_{iK}, C_{iK}\}, i = \overline{1, N}, K = \overline{1, S_i}, \quad (2)$$

де: P_i – i -та потреба користувача; A_{iK} – i -ий атрибут, що відображає i -ту потребу замовника та K -те обмеження на потребу; C_{iK} – обмеження на атрибут;

Сукупність $\{P_i, A_{iK}, C_{iK}\}, i = \overline{1, N}, K = \overline{1, S_i}$ представляє вимоги до WEB-продукту користувача бізнес-системи.

Для запису цих вимог в стандартизованому вигляді відобразимо (2) на елементи структури моделі якості у використанні [9].

$$\{H_i^u, M_{ij}^u\}, i = \overline{1, 4}, j = \overline{1, R_i} \quad (3)$$

де: H_i^u – i -та характеристика моделі якості; M_{ij}^u – метрика i -ої характеристики; R_i – кількість метрик.

У підсумку отримуємо модель вимог щодо якості користувача бізнес-системи, сформульовані у стандартизованих термінах.

$$V_{use} = \{H_i^u, A_{ik}^u, C_{ik}^u, M_{ik}^u\}, i \in N_u^k, K = \overline{1, S_i}, \quad (4)$$

де: V_{use} – вимоги у вигляді моделі якості у використанні; H_i^u – i -та характеристика моделі якості, A_{ik}^u – i -й атрибут, що відображає i -ту потребу замовника та K -те обмеження на потребу; C_{ik}^u – обмеження на атрибут; M_{ik}^u – метрика моделі якості;

Наступним етапом під час проектування є розроблення специфікацій системних вимог до програмного комплексу. Для цього відобразимо елементи сукупності (4) на елементи структури зовнішньої якості.

$$\{H_i^x, P_{ij}^x, M_{ij}^x\}, i = \overline{1, 6}, j = \overline{1, K_i}, \quad (5)$$

де: H_i^x – характеристики; P_{ij}^x – підхарактеристики зовнішньої якості; M_{ij}^x – відповідні метрики; K_i – кількість підхарактеристик i -ої характеристики.

У підсумку отримаємо специфікації вимог якості в стандартизованих термінах.

$$V_{out} = \{H_i^x, P_{iK}^x, A_{iK}^x, C_{iK}^x, M_{iK}^x\}, i \in N_x, K = \overline{1, R_i^x} \quad (6)$$

де: V_{out} – вимоги щодо зовнішньої якості; H_i^x – характеристики зовнішньої моделі якості; P_{iK}^x – підхарактеристики характеристик зовнішньої моделі якості; A_{iK}^x – атрибути підхарактеристик зовнішньої якості; C_{iK}^x – обмеження на атрибути; M_{iK}^x – метрики атрибутів зовнішньої моделі якості; N_x – кількість характеристик зовнішньої моделі якості; K – кількість атрибутів; R_i^x – кількість метрик зовнішньої моделі якості.

Для формування специфікацій вимог V_{in} до модулів програмного комплексу WEB-застосування виконується відображення вимог (6) на елементи структури моделі внутрішньої якості. Це відображення виконується для кожного модуля з врахуванням його функціональності.

Для побудови відображень (4), (6) використовується процедура класифікації, в якій використано алгоритм пошуку асоціацій на ієрархічній струк-

турі, якою видається модель якості [10]. При цьому, для кожного структурного елемента множини вхідних вимог здійснюється пошук найбільш залежних структурних елементів вихідних вимог. Процедура (7) ітераційна, реалізується скануванням за структурою "згори-вниз", а потім "знизу-вгору". Залежності визначаються експертом в режимі діалогу.

$$\begin{array}{ccc}
 H = \{H_1, H_2, \dots, H_n\} & & \\
 \downarrow & & \downarrow \\
 P_1 = \{P_{11}, P_{12}, \dots, P_{1k}\} & P_n = \{P_{n1}, P_{n2}, \dots, P_{nk}\} & \\
 \downarrow & \downarrow & \downarrow \quad \downarrow \\
 A_1 = \{A_{11}, A_{12}, \dots, A_{1k}\} & A_n = \{A_{n1}, A_{n2}, \dots, A_{ns}\} & \\
 \downarrow & \downarrow & \downarrow \quad \downarrow \quad \downarrow \\
 \{M_{11}, M_{12}, \dots, M_{1k}\} & \dots & \{M_{n1}, M_{n2}, \dots, M_{ns}\}
 \end{array} \quad (7)$$

Вершиною ієрархії є $H_1, \dots, H_n$ – стандартизовані характеристики, на нижчому рівні показано $\{P_{ij}\}$ – множини підхарактеристик, $\{A_{im}\}$ – множина атрибутів підхарактеристик, які вибрані з урахуванням специфіки предметної області, $\{M_{im}\}$ – відповідні метрики, які можуть вибиратися із стандартизованого переліку. Для реалізації цієї процедури створено репозиторії метрик, атрибутів, характеристик і підхарактеристик.

Проаналізувавши принципи побудови моделей якості і дані, які будуть зберігатись в репозиторії, розроблено функціональну схему CASE-засобу, яку наведено на рис. 1.

Вхідними даними для розробки вимог до ПЗ, згідно з рис. 1, є потреби замовника, які представляють у деякому формалізованому текстовому вигляді. Їх можна визначати на основі Use case діаграм, згенерованих і формально специфікованих за допомогою інших автоматизованих систем. Записавши потреби замовника, виконуємо їх аналіз та класифікацію за стандартизованими характеристиками $H_1, \dots, H_n$, підхарактеристиками $\{P_{ij}\}$ моделі якості [9]. Крім того, у CASE-засобі забезпечено вибір метрик M_{jm} з відповідного репозиторію (БД стандартизованих метрик). Після цього формуються відповідні записи множини вимог V_{use}, V_{out}, V_{in} у репозиторій проекту (БД вимог у вигляді моделі якості ISO 9126). Керування та зміна потреб і вимог відслідковується через модуль керування вимогами. Для зручності заповнення репозиторіїв спроектовано відповідні системи керування, зокрема для заповнення бази даних стандартизованих метрик. Після того як вимоги до ПЗ узгодженні зі всіма зацікавленими в проекті сторонами, формуються специфікації вимог, які використовують на наступних стадіях життєвого циклу.

Оцінювання якості готового програмного продукту відбувається шляхом використання тих же репозиторіїв, але з урахуванням особливостей цього процесу. Зокрема, для кількісного відображення міри якості програмного продукту потрібно обчислити інтегральний показник якості. У роботі [11] наведено методи обчислень локальних, частинних та глобального показників якості ПЗ. Враховуючи аспекти технології оцінювання якості, у CASE-засобі

розроблено відповідну підсистему для визначення міри задоволення вимог, функціональну схему якої наведено на рис. 2.

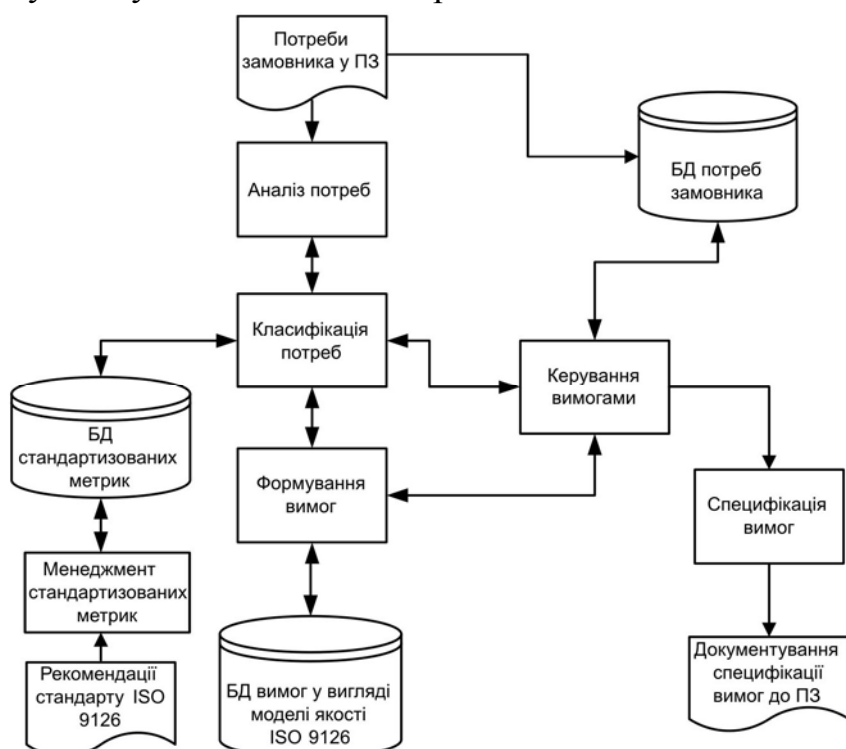


Рис. 1. Функціональна схема CASE-засобу розроблення вимог

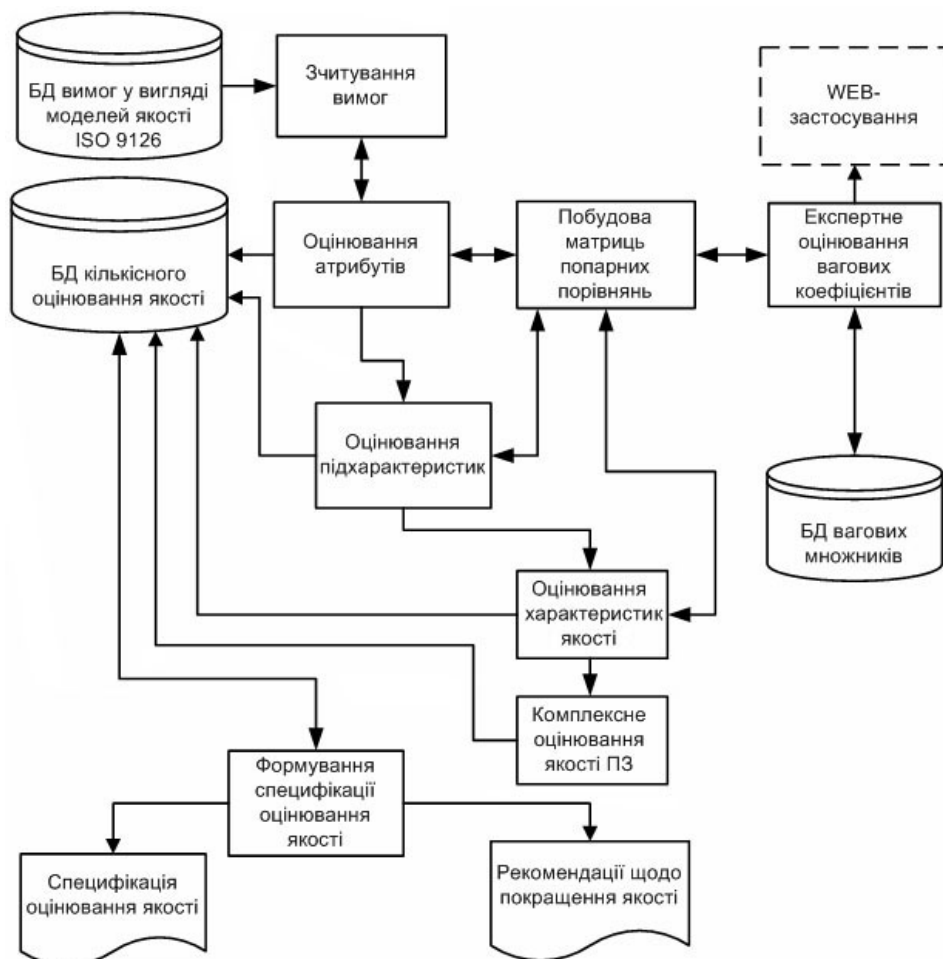


Рис. 2. Функціональна схема підсистеми оцінювання CASE-засобу

Із схеми, наведеної на рис. 2, видно, що процес оцінювання якості ПЗ починається із звернення до репозиторію вимог, які сформульовано у вигляді характеристик, підхарактеристик, атрибутів та метрик моделей якості. Для кожного атрибута якості ставимо у відповідність пряму або непряму метрику і задаємо елементарну функцію для його оцінювання. Для визначення ваги і впливу кожного атрибута запропоновано використати матриці попарних порівнянь, які дає змогу генерувати CASE-засіб. Побудовані шаблони матриць попарних порівнянь оцінюють експерти. Відношення впливу між атрибутами опрацьовують методами статистичного оброблення. В результаті отримаємо вагові коефіцієнти кожного атрибута. Аналогічно до атрибутів будують матриці попарних порівнянь для підхарактеристик та характеристик, що дає змогу визначити кількісні часткові показники якості за відповідними характеристиками та підхарактеристиками моделей якості. Усі вагові множники, визначені експертами, записують у репозиторій вагових множників, а результати оцінювання всіх рівнів моделі якості – у репозиторій кількісного оцінювання якості. На основі даних, які відображають кількісну міру якості, формують специфікацію процесу оцінювання та розробляють рекомендації щодо покращення якості програмного продукту.

Розроблений CASE-засіб є web-орієнтованим і його можна використовувати як web-сервіс для підтримки процесів розробки вимог і оцінювання якості ПЗ. У ньому забезпечено можливість віддаленого формування потреб замовника, специфікації вимог до ПЗ і визначення ступеня задоволення та відповідності вимог і потреб.

Структурно CASE-засіб складається з двох частин: клієнтської і серверної. Клієнтська частина надає замовнику можливість задання потреб у вигляді шаблону моделей якості. Шаблон становить набір стандартизованих характеристик, підхарактеристик, атрибутів та метрик. Замовник має змогу вибирати властивості, які необхідно реалізувати, базуючись на цьому шаблоні (рис. 3).

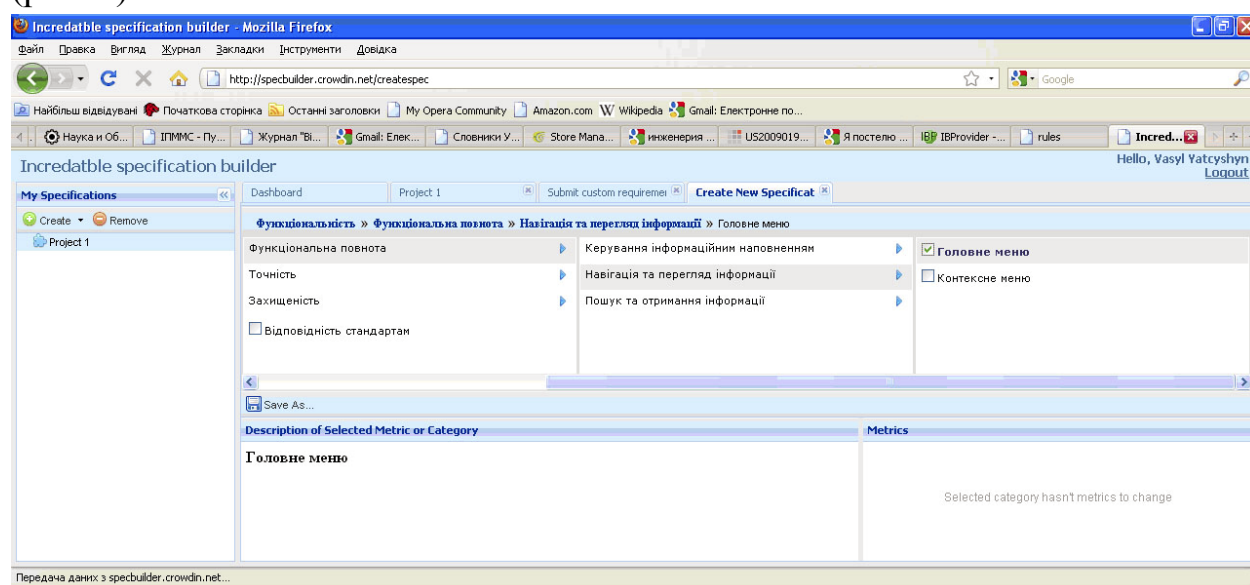


Рис. 3. Шаблон формування потреб замовника

У CASE-засобі реалізовано також можливість формування потреб замовника у формальному текстовому вигляді. Процедура адаптації потреб, за-

даних таким чином, виконується на сервері шляхом їх класифікації за відповідними атрибутами, підхарактеристиками і характеристиками.

Серверна частина реалізує функції менеджера роботи з репозиторіями вимог, потреб, вагових множників. На рис. 4 зображено інтерфейс серверної частини.

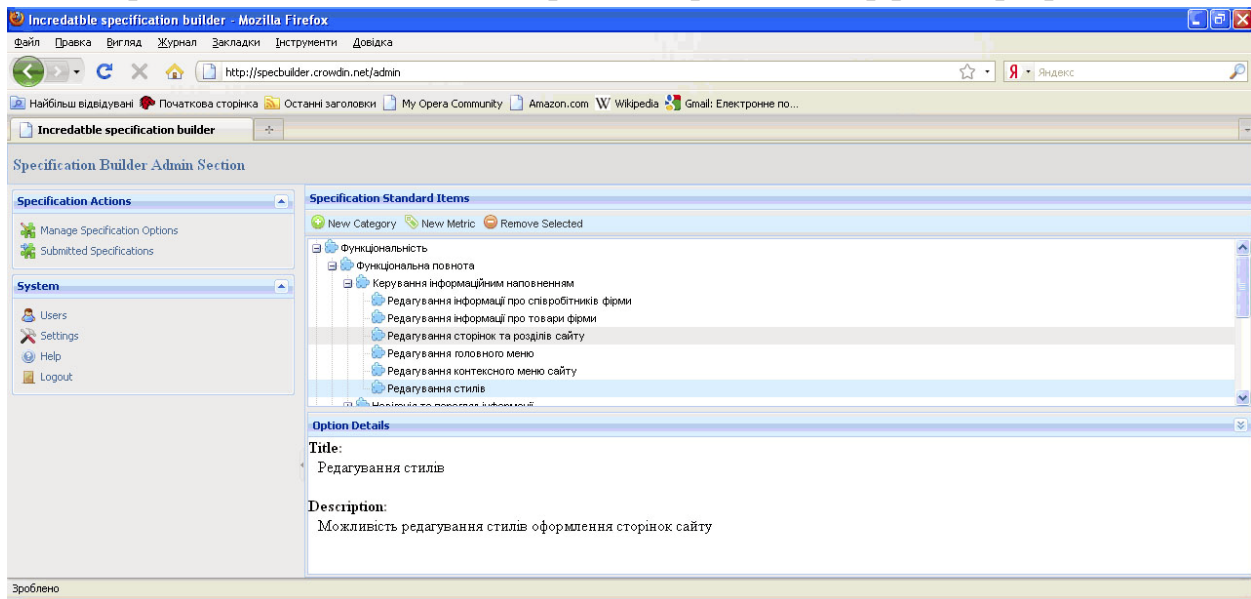


Рис. 4. Серверна частина CASE-засобу

На рис. 4 показано наповнення репозиторію стандартизованих метрик для оцінювання якості ПЗ. Крім того, адміністратор CASE-засобу має можливість контролю, керування та наповнення усіх репозиторіїв відповідними даними. У серверній частині забезпечено можливість розподілу прав доступу, налаштування зовнішнього інтерфейсу, спроектовано систему допомоги з користування CASE-засобом.

Під час розроблення засобу автоматизації, орієнтованого на підтримку етапу аналізу вимог та оцінювання якості використано відкриті компоненти та засоби розробки. Зокрема, клієнтську частину реалізовано за допомогою компонентів JavaScript (Ext JS). Серверну частину розроблено за допомогою PHP 5.1 (Zend Framework) та MySQL 4.4 (InnoDB). Web-сервіс можна використовувати за адресою в мережі Internet <http://specbuilder.crowdin.net/createspec>

Висновки. Внаслідок застосування запропонованого CASE-засобу реалізовано автоматизацію процесу розроблення та аналізу вимог, а також оцінювання якості ПЗ. Цей засіб є спеціалізованим і втілює основні принципи технології, базованої на рекомендаціях стандарту ISO 9126 1-4. Система дає змогу автоматизувати процес аналізу вимог, шляхом побудови розробленого в роботі шаблону для формування потреб. Наявність такого шаблону дає змогу однозначно та повною мірою враховувати побажання кінцевих користувачів, цілі замовника бізнес-системи і відповідність стандартам. На етапі розроблення вимог до ПЗ автоматизовано процеси: розроблення потреб замовника і формування відповідної специфікації, перетворення потреб у вимоги до ПЗ, генерування специфікації вимог. На етапі оцінювання якості ПЗ автоматизовано функції роботи з репозиторіями характеристик, підхарактеристик і метрик, процес визначення вагових множників і обчислення інтегрального показника якості.

Література

1. ISO/IEC 12207:2008 Systems and software engineering – Software life cycle processes.
2. Вендров А.М. CASE-технологии: Современные методы и средства проектирования информационных систем / А.М. Вендров. – М. : Изд-во "Финансы и статистика", 1998. – 176 с.
3. IEEE Std 830-1993, IEEE Recommended Practice for Software Requirements Specifications (ANSI).
4. Брауде Е. Технология разработки программного обеспечения / Е. Брауде. – СПб. : Изд-во "Питер", 2004. – 655 с.
5. Харченко О. Розроблення та керування вимогами до програмного забезпечення на основі моделі якості / О. Харченко, В. Яцишин // Вісник ТДТУ. – 2009. – Том 14, № 1. – С. 201-207.
6. Авраменко Е.А. Метод и средства редокументирования наследуемого программного обеспечения : дис... канд. техн. наук: спец. 01.05.03 / Национальный авиационный ун-т. – К., 2008. – 144 с.
7. Райчев І.Е. Принципи побудови моделі якості у використанні програмних систем / І.Е. Райчев, О.Г. Харченко // Збірник наукових праць інституту проблем моделювання в енергетиці. – 2007. – Вип. 39. – С. 31-38.
8. ISO/IEC 25030 Software engineering-Software product Quality Requirement and Evaluation (SQuaRE), Quality Requirements, 2007.
9. ISO/IEC 9126 (1-4). Software engineering – Product quality – Part 1: Quality model, Part 2: External metrics, Part 3: Internal metrics, Part 4: Quality in use metrics, 2001–2004.
10. Барсегян А.А. Технологии анализа данных / А.А. Барсегян, М.С. Куприянов, В.В. Степаненко, И.И. Холод. – СПб. : Изд-во "БХВ-Петербург", 2008. – 384 с.
11. Яцишин В.В. Аналіз технологій розробки та управління вимогами до програмного забезпечення / В.В. Яцишин // Інженерія програмного забезпечення 2007 : матер. Всеукраїнської конф. аспірт. і студентів. – К. : Изд-во НАУ. – С. 81-85.

Яцишин В.В., Харченко О.Г. CASE-технология разработки требований к программному обеспечению и оцениванию его качества

Приведены результаты исследования современных CASE-технологий, которые используют для поддержки процессов проектирования на разных этапах жизненного цикла программного обеспечения (ПО). В результате анализа обнаружен ряд недостатков этих технологий и предложена собственная CASE-технология, которая является web-ориентированной и направленной в поддержку технологии, базированной на модели качества стандарта ISO 9126. Показано, что использование этой модели, соответствующей технологии и средства дает возможность адекватно отобразить потребности заказчика и пользователей ПО на спецификации требований, а также определить степень их удовлетворения. Предлагаемая CASE-технология дает возможность автоматизировать процесс разработки требований и управления ими, что значительно повышает качество процесса проектирования и сокращает временные рамки выполнения проекта.

Yatsyshyn V.V., Harchenko O.G. CASE technology of the requirements development and quality evaluation of software

The article describes the results of the modern CASE technologies researchments that are used for supporting the process at different stages of the life cycle of the software. Some disadvantages of these technologies were discovered and an own CASE method was proposed, oriented on the technology supportment, based on the usage of the quality model (standart ISO 9126-1). The article states that the usage of this technology gives us opportunity to show adequately the needs of the supplier and of the users of software on the requirements specification and also define the measure of these requirements satisfaction. This CASE method is web-oriented, it lets us to automates the process of the development and the requirements of quality management, that rises its quality and makes the process of design faster.