

<https://doi.org/10.31891/2307-5732-2023-317-1-280-288>

УДК 004.891.3:004.3

ЗАСОРНОВА Ірина

Хмельницький національний університет

<https://orcid.org/0000-0001-6655-5023>

e-mail: zasornovair@khmnu.edu.ua

ЗАСОРНОВ Олександр

Хмельницький національний університет

<https://orcid.org/0000-0002-9387-6997>

e-mail: zasornov.os@khnu.km.ua

МАЗНЕВ Євген

Східноукраїнський національний університет імені Володимира Даля

<https://orcid.org/0000-0001-7952-8980>

e-mail: maznev@snu.edu.ua

САРАНА Олександр

Східноукраїнський національний університет імені Володимира Даля

<https://orcid.org/0000-0003-1898-8283>

e-mail: sarana@snu.edu.ua

АНАЛІТИЧНИЙ ОГЛЯД ІНФОРМАЦІЇ ДЛЯ ТЕСТУВАННЯ ВІДКРИТОЇ ПАРАМЕТРИЧНОЇ СИСТЕМИ АВТОМАТИЗОВАНОГО ПРОЄКТУВАННЯ «ВАЛЕНТИНА»

У статті проаналізовано пропріетарні Lectra systems (Франція), Gerber Garment Technology (США), Investronika (Іспанія), Assyst GmbH (Німеччина), Wild Ginger Software (США) та відкриті PatternViewer (США), Валентина (Україна) системи автоматизованого проєктування для швейного виробництва.

Виконано аналітичний огляд інформації для тестування відкритої параметричної системи автоматизованого проєктування «Валентина». Наведено мінімальний перелік даних для якісного тестування, який містить 10-ть пунктів, основними з яких є: лістинги вихідних текстів програмних продуктів, зовнішня специфікація, план та методики тестування, варіанти тестів та отримані результати.

Отримана інформація для тестування відкритої параметричної системи автоматизованого проєктування «Валентина» в подальшому може бути використана для проведення аналізу способів тестування програмних продуктів та їх покращення, а також для розробки автоматизованих тестів, які можна впровадити у процеси створення та експлуатації програмних продуктів.

Ключові слова: «Валентина», системи автоматизованого проєктування, відкриті програмні продукти, інформація для тестування, GitLab.

ZASORNOVA Iryna, ZASORNOV Alexander

Khmelnytskyi National University

MAZNIEV Ievgen, SARANA Oleksandr

Volodymyr Dahl East Ukrainian National University

ANALYTICAL REVIEW OF INFORMATION FOR TESTING THE OPEN PARAMETRIC SYSTEM OF AUTOMATED PROJECTION «VALENTINA»

The article analyzes proprietary systems, such as Lectra systems (France), Gerber Garment Technology (USA), Investronika (Spain), Assyst GmbH (Germany), Wild Ginger Software (USA) and open PatternViewer (USA), Valentina (Ukraine) automated design systems for sewing production. An analytical review of the information for testing the open parametric system of automated projection and design «Valentina» was performed. It was determined that the theoretical foundations of automated testing of automated design systems are currently an incompletely resolved problem. The minimum list of data for qualitative testing is presented, which contains 10 items, the main of which are: listings of source texts of software products, external specification, test plan and methods, test options and obtained results. The article provides a list of Internet sources of the «Valentina» software product distribution, a list of communication methods, as well as a list of external specification items required for testing the «Valentina» software product. Various versions of the «Valentina» were considered, as well as the changes made in the test version v0.7.52 compared to the previous version v0.7.51. Various types of errors that occur during the development of new functions of software products are analyzed, including the inability to perform one of the functions; inability to complete a function; the appearance on the screen of symbols or program primitives that do not correspond to the action; involuntary closing of the program window; closing the program window without warnings about saving data; other errors. The information obtained for testing the open parametric system of automated projection «Valentina» can be used in the future for the analysis of methods of testing software products and their improvement, as well as for the development of automated tests that can be implemented in the processes of creating and operating software products.

Keywords: «Valentina», automated projection systems; open software products; information for testing, GitLab.

Постановка проблеми у загальному вигляді

та її зв'язок із важливими науковими чи практичними завданнями

Звіт, який надано Консорціумом щодо якості інформації та програмного забезпечення (CISQ), містить інформацію, що проблеми якості програмного забезпечення у США на 2022 рік коштували біля 2,41 трлн. доларів [1]. Ці витрати походять від:

- кібернетичних атак;
- проблем з ланцюгами постачання сторонніх програм, які входять до складу програмного продукту (ПП);

– проблем накопичення технічної заборгованості, яка стосується витрат на усунення недоліків розробленого програмного забезпечення, що робить дані та системи потенційно вразливими.

Саме тому, тестування є важливим етапом у процесі розробки ПП. Серед великої кількості ПП окремо можна виділити системи автоматичного проектування (САПР). ПП САПР доволі складний об'єкт для тестування. Автоматизоване тестування САПР і досі зводиться до окремих тестів, які розробляють тестувальники з урахуванням коду ПП. Якщо САПР є пропрієтарною (proprietary software), то код такого ПП відповідно є власністю компанії та не може бути отриманим.

Це призводить до недостатності тестової інформації САПР і викликає певні труднощі щодо дослідження та розробки теоретичних основ тестування САПР.

Аналіз останніх досліджень і публікацій

Останні дослідження стосовно тестування ПП тісно пов'язані з публікаціями, у яких дослідники здійснюють аналіз даних для розроблення нових способів тестування на різних етапах розробки та використання ПП. Такі дослідники як: Говорущенко Т.О. [2–5], Троян А.М., Моденов Ю.Б. [6], P.D.D. Dominic and Handaru Jati [7], Ісаков, О.С. [8], Мактрейгор Д. [9], Канер С. [10] та інші чітко визначили необхідність отримання даних процесу тестування ПП для аналізу та дослідження.

Тестування є важливим етапом життєвого циклу виробництва ПП [8]. Тестування починають із збору інформації та формуванню вимог щодо ПП. Характеристики специфікації вимог до ПП значною мірою визначаються характеристиками вхідної інформації. У роботі [2] автори виконують аналіз вимог до сучасних ПП. Якщо вхідна інформація є недостатньою, або ж вона є неточною, неоднозначною чи суперечливою, то існує висока імовірність, що усі ці недоліки проявляться і у специфікації вимог до ПП. Така (не якісна) специфікація вимог може негативно вплинути на успішність тестування.

Найбільш поширеним тестування є не автоматизоване тестування, де перевірку роботи програм здійснюють за результатами їх виконання з використанням спеціально підібраних вихідних даних, які називають тестами. Не автоматизоване тестування програм роблять шляхом опрацювання тестових даних і порівняння отриманих результатів з очікуваними. За дослідженнями авторів [6] автоматизоване тестування для забезпечення якості програмних продуктів неможливо здійснити без попереднього аналізу ручного тестування, що дозволяє визначити доцільність автоматизації та можливість її використання у тому чи іншому випадку. Часто для одержання певних даних, необхідно звернутися до зовнішніх сервісів із запитом на виконання операцій та відобразити їхні результати у протоколі тестування інформаційної бази. На думку авторів [8] вибір відповідних елементів програми, для автоматизації тестування, буде визначним для тестування. Проте, це можливо здійснити за умов наявності повної інформації стосовно процесу тестування ПП. До повної інформації необхідно віднести відомості про внутрішню структуру і код ПП. Оскільки без цього неможливо використати тестування методом White Box [8–10].

Авторами К. Fatima, N.Z. Wawany, M. Bukhari [11] проведено оцінку зручності використання, доступності та вразливості пакистанських банківських веб-сайтів у порівнянні з міжнародними. Однак, автори зазначають, що жоден із сайтів не пройшов тести із 100% результатом.

В останні роки було розроблено певну кількість веб-сайтів реєстрації вакцин проти COVID-19. Дослідники Sarah Alismai та Wallace Chipidza провели автоматизований аналіз веб-доступності 54-х офіційних сайтів (США) [12]. Результати показали неоптимальну відповідність інструкціям WCAG 2.0 і 2.1, що визначено за допомогою інструментів AChecker, WAVE і SortSite.

У праці [13] автори Agrawal G., Kumar D. і Singh M. провели аналіз 164-х державних веб-сайтів, які призначені для надання електронних послуг громадянам країни (Індія). Для проведення дослідження використовувались онлайн-інструменти TAW, MobileOK. Результати показали, що більшість розглянутих веб-сайтів мали низьку зручність використання. Це пов'язано із неможливістю повного доступу командою тестувальників до коду сайту. Така ж ситуація спостерігається і при тестуванні програм САПР.

Більшість сучасних пропрієтарних САПР розробляють з урахуванням інтересів розробників та замовників в умовах суворої конфіденційності. Тому, отримати інформацію стосовно тестування такого ПП практично неможливо. Проте, відсутність даних стосовно тестової інформації пропрієтарних САПР ускладнює дослідження та розробку практичних рекомендацій для тестування. Однак, існує доволі велика група відкритих (вільних) САПР.

Доступ до інформації стосовно розробки та тестування ПП можна здійснити, використовуючи інтернет-джерела. Проте, розробники таких САПР використовують різні способи зберігання інформації, контролю версій, тощо. Це значно ускладнює пошук і аналіз необхідних даних.

Формулювання невирішеної раніше проблеми

Теоретичні основи автоматизованого тестування САПР і досі є не до кінця вирішеною проблемою. Тому, отримання і аналітичний огляд тестової інформації є важливими для дослідження та розробки теоретичних основ тестування САПР. Отримані відомості дозволять провести аналіз способів тестування ПП, запропонувати шляхи їх удосконалення та розробити автоматизовані тести, які можна впровадити у процес створення та експлуатації ПП.

Виклад основного матеріалу

Класичним рішенням для швейного виробництва (ШВ) є система автоматизованого проєктування (САПР). Проте, не всі САПР у повній мірі охоплюють потреби створення викрійок, адже у своїй більшості вони не дозволяють проєктувати параметричні креслення. За останні десять років ситуація мало змінилася. Рівень комп'ютеризації середніх та малих ШВ і досі є незадовільним. Навіть ті підприємства, які намагаються впровадити у виробничий процес САПР та використовують пробний (ознайомчий) період, в результаті відмовляються від використання програмного забезпечення і виконують креслення в ручному режимі. В першу чергу, це пов'язано із недостатньою базою креслень (оскільки заповнення кресленнями здійснює саме підприємство, орієнтуючись на власний асортимент). Тому, лише підприємства, у яких відсоток сумісності промислових рішень та асортименту, що існує, перевищує 55% використовують ПП ШВ. Тільки доволі потужне підприємство може дозволити собі мати периферійне обладнання, яке використовують в швейній промисловості (плотери, графопобудовники, комплекси для настилення та крою, тощо). Відсутність такого обладнання робить використання ПП ШВ недоцільним. Користувач в такому випадку не має змоги відчутти доцільність використання САПР. Також стримує розповсюдження ПП ШВ відсутність висококваліфікованих фахівців.

Відомо, що більшість ПП ШВ є пропрієтарні (закриті). Такі ПП ШВ є платними і доволі дорогавартісними, а саме тому, не доступними широкому загалу. Наприклад, Lectra systems (Франція), Gerber Garment Technology (США), Investronika (Іспанія), Assyst GmbH (Німеччина), Wild Ginger Software (США), тощо. Не кожне середнє, а тим паче мале підприємство може собі дозволити їх використання [14]. Обмежене використання таких ПП ШВ (лише на великих підприємствах) призводить до недоступності потрібної технічної інформації. Оскільки орієнтація на використання широкодоступних ресурсів (реклама ПП ШВ, технічний опис роботи з САПР, зворотний зв'язок, тощо) не мають сенсу для представників середнього та малого бізнесу. Окрім того, більшість з відомих рішень підтримують тільки операційну систему Windows (тобто, у користувача відсутня можливість вибору іншої операційної системи).

Поряд з пропрієтарними, існують відкриті (вільні) ПП ШВ, які були створені як альтернатива (на протипагу платним ПП). Наприклад, PatternViewer (США), Валентина (Україна), тощо. Саме тому, у більшості випадків такі підприємства використовують безкоштовні графічні редактори або САПР, що є не адаптованими для потреб ШВ. Підприємству потрібно докласти чимало зусиль для адаптації ПП під власний асортимент. Окрім того, доволі важко використовувати такі ПП в індивідуальному виробництві (будинки побуту, весільні салони, майстерні для індивідуального виготовлення одягу, тощо).

Після ознайомлення і аналізу вільного ПП ШВ, для отримання тестової інформації САПР було обрано ПП «Валентина» (рис 1).

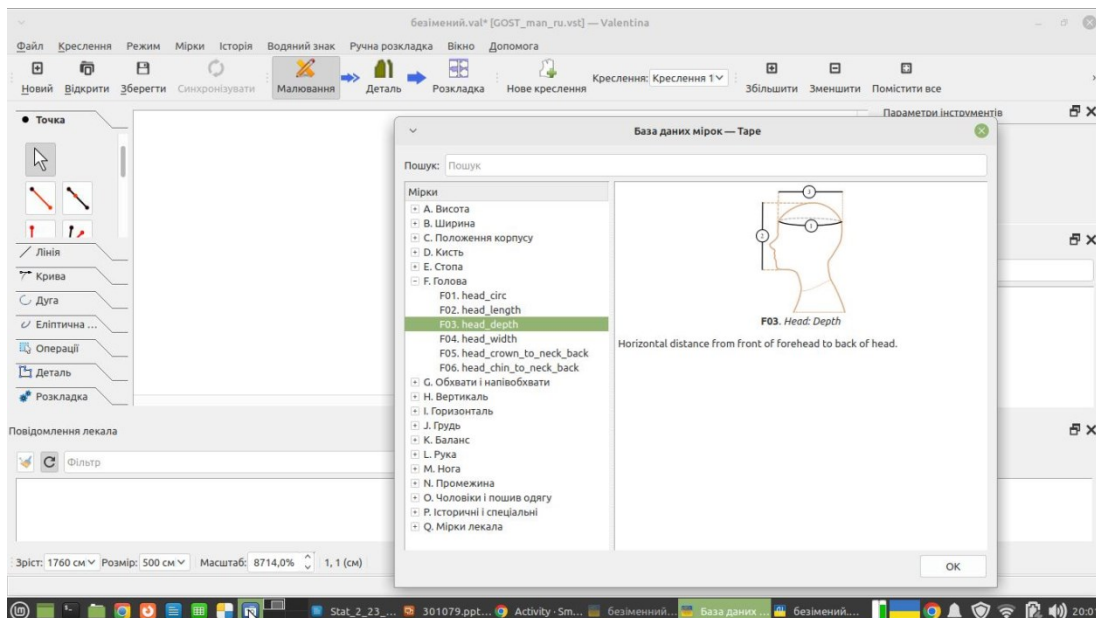


Рис. 1. Екранний образ ПП «Валентина»

Аналітичний огляд тестової інформації неможливо зробити без всебічного ознайомлення з обраним ПП.

Для створення і поширення ПП «Валентина» розробниками була обрана модель відкритого програмного забезпечення, що дозволило широко адаптувати програму для багатьох сучасних платформ (він є доступним не тільки для Windows, а також і для Mac OS X і Linux) [15].

ПП «Валентина» не є традиційним програмним рішенням для ШВ, так як в ній реалізована ідея роботи з параметричними викрійками. ПП «Валентина» — це кросплатформна програма для виготовлення викрійок та лекал деталей моделей одягу. Цей ПП дозволяє створювати шаблони викрійок, використовуючи стандартні таблиці розмірів, або індивідуальний набір мірок (рис 1). ПП «Валентина» поєднує нові

технології з традиційними методами для створення викрійок та лекал деталей моделей одягу [15].

Оскільки ПП «Валентина» є відкритим, то прямий доступ до нього можна отримати на GitLab [16]. Розробники ПП «Валентина» використовують GitLab, як хмарну платформу для хостингу проєкту та спільної розробки. У зв'язку з оновленням ПП «Валентина» необхідний контроль версій здійснюють, використовуючи популярну систему Git [17]. Крім того, вони використовують GitLab, як повноцінну соціальну мережу для розробників. На GitLab ПП «Валентина» розташовані open-source-проєкти. Кожен користувач GitLab ПП «Валентина» може, за згодою розробників, взяти участь у проєкті. Для цього можна ознайомитися з програмними рішеннями проєкту, розмістити своє портфоліо з прикладами коду, щоб додати посилання до резюме. Окрім того, на GitLab ПП «Валентина» можна отримати корисні посилання та пояснення щодо використання ПП. Також дистрибутив можна отримати на сайті програми [15] та з інших інтернет-джерел (табл. 1).

Таблиця 1

Перелік інтернет-джерел дистрибутива ПП «Валентина»

№	Ресурс	Адреса
1	Офіційний сайт	https://smart-pattern.com.ua/uk/valentina/download/
2	Попередній репозиторій	https://bitbucket.org/dismine/valentina/src/master/
3	Основний репозиторій	https://gitlab.com/smart-pattern/valentina
4	Репозиторій GitHub	https://github.com/dismine/valentina
5	Репозиторій codeship	https://app.codeship.com/projects/383590
6	Репозиторій Cirrus CI	https://cirrus-ci.com/github/dismine/valentina
7	Репозиторій LaunchPad	https://code.launchpad.net/~dismine
8	Репозиторій AppVeyor	https://ci.appveyor.com/project/dismine /valentina

ПП «Валентина» дозволено використовувати без спеціалізованого дорогого периферійного обладнання, оскільки будь-яке креслення можливо роздрукувати на звичайному принтері.

Для інсталяції ПП «Валентина» доступними є стабільна та тестова версії:

- стабільна має парний номер;
- тестова має непарний номер.

Стабільною версією ПП «Валентина» вважається після тестового періоду, під час якого здійснюють перевірку роботи програми. На офіційному сайті програми доступними для завантаження є стабільна версія v0.6.1 та остання тестова версія v0.7.52 [18]. Для використання на виробництві рекомендовано використовувати стабільну версію v0.6.1.

Слід зазначити, що ПП «Валентина» також додана в репозиторій деяких дистрибутивів Linux. Наприклад, репозиторій Ubuntu вміщує наступні версії [19]:

- focal (20.04LTS) v0.6.1~dfsg-9: amd64 arm64 ppc64el s390x;
- jammy (22.04LTS) v0.7.49~dfsg-2: amd64 arm64 armhf ppc64el s390x;
- kinetic (22.10) v0.7.51~dfsg-1: amd64 arm64 armhf ppc64el s390x;
- lunar (23.04) v0.7.52~dfsg-1: amd64 arm64 armhf ppc64el s390x.

З наведеного списку бачимо, що залежно від версії операційної системи, змінюється версія ПП «Валентина». Проте, завжди існує можливість інсталяції останньої версії з сайту розробника.

Різниця між запропонованими версіями доволі суттєва, оскільки розробники постійно удосконалюють і розширяють можливості ПП «Валентина». Саме тому, ПП «Валентина» v0.7.52 (у порівнянні з v0.6.1) зазнала значних змін: були додані нові корисні функції та інструменти. З повним переліком змін можна ознайомитись у файлі ChangeLog.txt [20]. Цей файл дозволяє проаналізувати, які зміни були внесені в ПП «Валентина», починаючи з версій – v0.2.0, яка є першим публічним випуском і була представлена користувачу у 2013 році.

Для прикладу розглянемо, які зміни здійснені у тестовій версії v0.7.52 ПП «Валентина» (у порівнянні із попередньою версією v0.7.51):

- локалізація ги оголошена не сумісною. Якщо у налаштуваннях вказана ги локаль, то програма буде аварійно завершуватися;
- удосконалено попередження при розрахунках сегментів кривих. На додачу до імені кривої, додано ім'я точки, яку шукають;
- додаток Таре. Пункт «Додати Роздільник» додано до меню;
- додаток Таре. Пункт меню «Експортувати з наявних ...» тепер підтримує також експорт із багаторозмірних таблиць;
- при додаванні функції в Майстрі формул разом з іменем будуть додаватися дужки та знак розділювача аргументів, якщо такий потрібен;
- додано можливість вказувати тип одиниць виміру для значення прибавки;
- при експорті даних розкладки в додатку Puzzle тепер будуть враховуватися значення налаштувань «Примусове перевертання» та «Заборонити відзеркалення». Відповідні налаштування будуть автоматично застосовані до деталі без можливості відключення;
- запис про останні обрані значення вимірів багаторозмірної таблиці тепер зберігаються у файлі викрійки, що дозволяє обирати їх при наступному відкритті викрійки;
- візуалізацію створення декількох інструментів покращено, що дозволить в більш інтерактивному

режимі створювати об'єкти;

- об'єкт «Еліптична дуга» тепер підтримує параметр апроксимації кривої;
- розмір діалогу інструменту «Складна крива» тепер можна змінювати;
- удосконалено функцію «Показати у файловому менеджері» для системи Linux. Тепер відкривається не тільки директорія із файлом, але й сам файл виділяється.

Зрозуміло, що при розробці нових функцій, у програмі можуть виникнути різного роду помилки, до яких можна віднести:

- неможливість виконати одну з функцій;
- неможливість завершити функцію;
- поява на екрані символів або примітивів програми, що не відповідають дії;
- мимовільне закриття вікна програми;
- закриття вікна програми без попереджень про збереження даних;
- інші помилки.

Тобто, при використанні тестової версії можливо отримати помилки разом з новими можливостями останньої тестової версії. Проте, для тестування потрібно використовувати саме останню версію, оскільки вона ймовірно містить найбільшу кількість помилок, які необхідно виявити. При ознайомленні з командою розробника виявлено, що кількість офіційно закріплених тестувальників рівна п'яти. Така їх кількість може забезпечити тестування змін, які були впроваджені. Проте, більш повну перевірку роботи ПП «Валентина» можуть зробити лише користувачі [21]. Досвідчені фахівці, використовуючи нестабільні тестові версії і знаходячи помилки, допомагають розвитку ПП «Валентина». Зв'язок може бути здійснено з використанням GitLab та інших способів комунікації, запропонованих розробником, таблиця 2.

Таблиця 2

Перелік способів комунікації ПП «Валентина»

Ресурс	Адреса
Telegram	https://t.me/valentina_project
Email	contact@smart-pattern.com.ua
Skype	skype:dismi?chat
Viber	viber://chat?number=0380983467136
WhatsApp	https://wa.me/0380983467136
Threema	https://threema.id/WVDPXTJC?text=
GitLab	https://gitlab.com/smart-pattern/valentina

Окрім того, можна повідомити про помилку, використавши пункт головного меню – «Допомога», а далі підпункт – «Повідомити про помилку».

Як було вказано раніше, нововведення передбачають тестування програмного продукту для виявлення помилок його роботи. Після оновлення ПП з версії v0.7.51 до v0.7.52 тестувальники виявили ряд помилок, а саме:

- некоректне оновлення деталі в додатку «Пазл» при повторному імпорті необроблених даних;
- візуалізації інструменту «Точка перетину лінії та осі»;
- падіння ПП «Валентина» після скасування інструменту;
- зчитування імені деталі в додатку «Пазл»;
- попередження при розрахунку сегмента віддзеркаленої дуги;
- некоректну валідацію основного контуру. Помилково позитивне розпізнавання подвійної кривої;
- налаштування початкових параметрів вимірів багаторозмірної таблиці;
- оновлення міток вимірів багаторозмірної таблиці в додатку ПП «Валентина»;
- налаштування пунктів меню для Mac OS;

– проблему із сегментуванням кривої. При деяких конфігураціях кривої розрахунок відбувався не коректно;

– експорт PDF плиткою для випадків, коли принтер не підтримує роздільну здатність 96 пікселів на дюйм;

- розрахунок еліптичної дуги;
- не коректну генерацію прибавки на шов;

Після виявлення і виправлення перерахованих помилок ПП «Валентина» була представлена користувачам (тестова версія v0.7.52).

Раніше розробники використовували Bitbucket для зберігання файлів проєкту. Проте, останнє звернення було 15.10.2022 року [22]. Разом із тим, Bitbucket і досі містить цікаву інформацію стосовно тестування стабільної версії v0.6.1, (рис. 2).

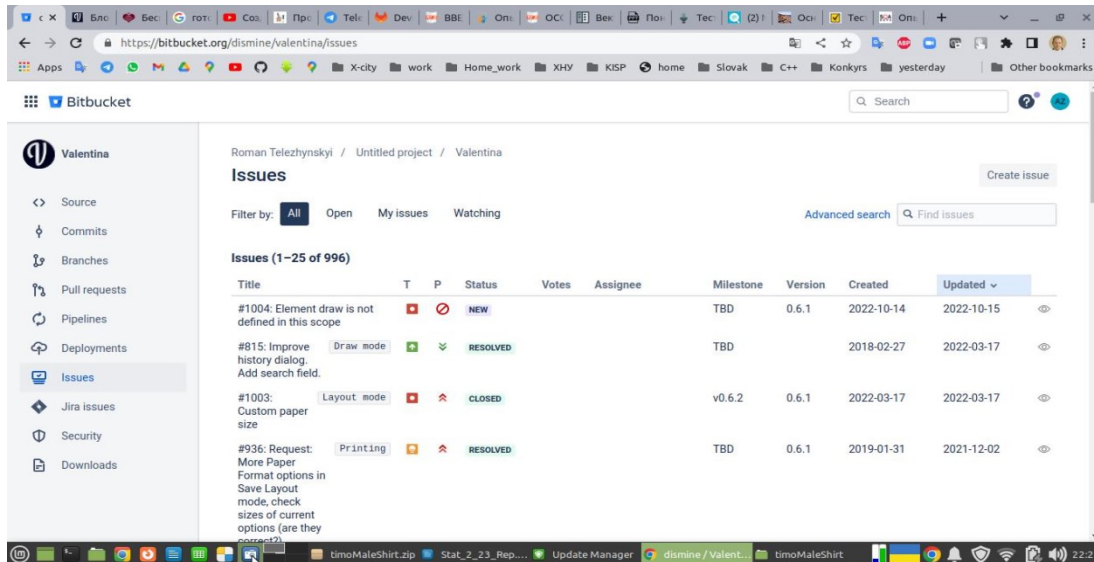


Рис. 2. Екранний образ Bitbucket ПП «Валентина» з інформацією стосовно тестування

Потім проєкт було розташовано на GitLab ПП «Валентина». Саме тут на Boards подано останню інформацію стосовно тестування (рис 3).

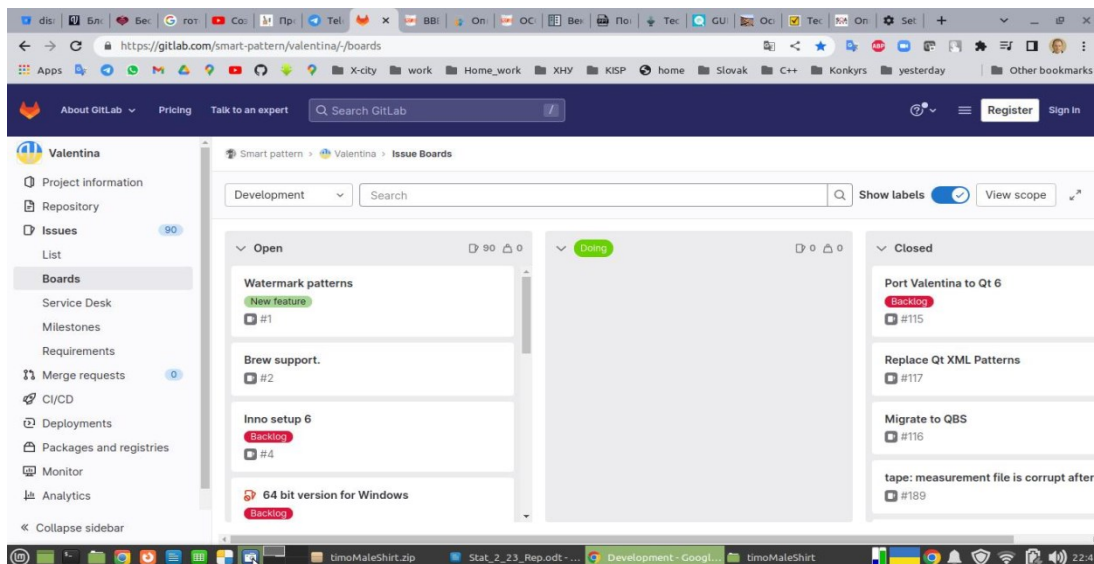


Рис. 3. Екранний образ Github ПП «Валентина» з інформацією стосовно тестування

Використовуючи репозиторій ПП «Валентина», Bitbucket, GitLab та інші інтернет-ресурси, можна отримати необхідну інформацію стосовно тестування та покращення ПП.

Така інформація повинна містити все, що відноситься до зовнішньої специфікації. Тобто, опис зовнішніх функцій та очікуваної поведінки ПП з погляду зовнішнього по відношенню до нього користувача. Перелік пунктів зовнішньої специфікації, необхідної для тестування ПП «Валентина» наведено у таблиці 3.

Таблиця 3

Перелік пунктів зовнішньої специфікації, необхідної для тестування ПП «Валентина»

№	Пункти зовнішньої специфікації ПП «Валентина»	Наявність інформації
1	Назва та короткий опис ПП	присутня
2	Функціональна схема ПП	присутня
3	Організація діалогу ПП з користувачем	присутня
4	Опис меню, підменю, дій функціональних кнопок	присутня
5	Усі екранні форми або протокольні екранні повідомлення	присутня
6	Повідомлення для користувача (під час роботи з ПП) та відповіді на них	присутня
7	Повідомлення про помилки	присутня
8	Підказки користувачу та організація допомоги	присутня
9	Структура та організація баз даних ПП	присутня
10	Опис підготовки вхідних даних ПП	присутня
11	Вихідні друкарські форми ПП	присутня
12	Інші зовнішні сполучення ПП	присутня

Зовнішня специфікація містить опис лише зовнішніх аспектів ПП «Валентина» і не пов'язана з його внутрішньою структурою і кодом ПП. Окрім зовнішньої специфікації, важливим елементом тестових даних є сукупність усіх елементів інформації, які були створені у процесі конструювання ПП «Валентина». Мінімальний перелік даних для тестування повинен включати наступні базові елементи, таблиця 4.

Таблиця 4

Мінімальний перелік даних, необхідних для тестування ПП «Валентина»

№	Пункти переліку даних, необхідних для тестування ПП «Валентина»	Наявність інформації
1	Лістинги вихідних текстів ПП	присутня
2	Дистрибутив ПП	присутня
3	Посібники з роботи та інсталяції	присутня
4	Посібник користувача з налаштування	присутня
5	Документи супроводу, запити супроводу, стандарти та методики конструювання	присутня
6	Зовнішня специфікація	присутня
7	План та методики тестування	присутня
8	Варіанти тестів та отримані результати	присутня
9	Звіти про проблеми ПП	присутня
10	Звіти про конструкторські зміни	присутня

З таблиці 4 видно, що дані для тестування включають у себе велику кількість інформації. Проте, наявність усіх перерахованих відомостей не гарантує, що усі елементи переліку вірно розроблені, враховані і досить деталізовані для підтримки ПП в повному життєвому циклі. Розробник не може передбачити, як замовник реально буде використовувати ПП. Тому, для виявлення помилок (які здатен знайти тільки кінцевий користувач) при розробці ПП «Валентина» використовують процеси, Alpha Testing і Beta Testing.

Висновки з даного дослідження і перспективи подальших розвідок у даному напрямі

Попередній огляд показав, що не усі існуючі САПР в повній мірі охоплюють потреби створення викрійок, адже в більшості своїй вони не дозволяють проектувати параметричні креслення. Поряд з пропріетарними, існують відкриті (вільні) ПП ШВ. Підприємству необхідно докладати багато зусиль для адаптації таких ПП під свій асортимент. Отримати необхідні дані для тестування САПР можна:

- у складі команди розробників (для пропріетарного та відкритого ПП);
- використавши інтернет-джерела (для відкритого ПП).

Після ознайомлення і аналізу вільного ПП ШВ для розробки структурованого підходу для отримання тестової інформації САПР було обрано ПП «Валентина».

Аналітичний огляд інформації для тестування відкритої параметричної САПР «Валентина» показав, що кількість основних інтернет-джерел рівна восьми. В процесі розробки дані постійно змінюються, тому найбільш повну інформацію містить репозиторій, в якому відбувається робота команди розробника. Розробник піклується про якість своєї продукції, оскільки процес тестування не припиняється. Окрім того, розробник чітко організував зворотній зв'язок з користувачами і доволі швидко реагує на помилки і побажання, здійснює якісний супровід ПП та має усю необхідну супроводжуючу документацію.

Проведений аналітичний огляд інформації для тестування показав, що для якісного тестування САПР недостатньо мати лише зовнішню специфікацію, яка включає у себе 12-ть пунктів. Мінімальний перелік даних вміщує 10-ть пунктів, основними з яких є: лістинги вихідних текстів ПП, зовнішня специфікація, план та методики тестування, варіанти тестів та отримані результати. При цьому, виконати тестування функціоналу ПП, маючи лише інформацію згідно зовнішньої специфікації, неможливо. Це пов'язано із тим, що тестувальник може не знати про усі функціональні можливості роботи ПП (якщо він не має доступу до коду ПП). Інформація про тестування, яке вже було проведено, потрібна для того, щоб не повторювати роботу та розробити шляхи удосконалення тестів. Окрім того, потрібно знати, які ПП були використані для тестування та які тести проводилися в ручному режимі, а які в автоматичному.

Отримана інформація для тестування відкритої параметричної САПР «Валентина» в подальшому може бути використана для проведення аналізу способів тестування ПП та їх покращення, а також для розробки автоматизованих тестів, які можна впровадити у процеси створення та експлуатації ПП.

Література

1. Software Quality Issues in the U.S. Cost an Estimated \$2.41 Trillion in 2022. URL: <https://www.prnewswire.com/news-releases/software-quality-issues-in-the-us-cost-an-estimated-2-41-trillion-in-2022--301695684.html>.

2. Говорущенко Т.О., Боднар М.А., Кушнір В.О. Сучасні проблеми формування та аналізу вимог до програмного забезпечення. Міжнародний науково-технічний журнал «Вимірювальна та обчислювальна техніка в технологічних процесах». – Хмельницький, 2019. – С. 45-53. ISSN2219-9365. URL: <http://78.152.183.36/bitstream/123456789/8263/1/10.pdf>.

3. Hovorushchenko, T. & Popov, P.T. (2021). Method of developing the defect-free medical software by establishing the presence of residual defects. Proceedings of the 4th International Conference on Informatics & Data-Driven Medicine, 3038, pp. 11-21. ISSN 1613-0073. URL: <https://openaccess.city.ac.uk/id/eprint/27370/1/>.
4. I. Lopatto, T. Hovorushchenko, P. Popov and O. Pavlova, «Intelligent Multi-Agent System for Improving the Quality of Software by Taking into Account the Information of the Subject Area at All Stages of its Development.» 2021 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), Cracow, Poland, 2021, pp. 548-551, doi: 10.1109/IDAACS53288.2021.9660866.
5. T. Hovorushchenko and O. Pavlova, «Multi-Agent Technology for Assessing the Availability of Information at the Software Development Initial Stages.» 2021 IEEE 16th International Conference on Computer Sciences and Information Technologies (CSIT), LVIV, Ukraine, 2021, pp. 353-356, doi: 10.1109/CSIT52700.2021.9648832.
6. Троян А.М., Моденов Ю.Б. Доцільність автоматизованого тестування для забезпечення якості програмних продуктів. Проблеми інформатизації та управління. Том 1, №57-58. 2017. – С. 86-89. URL: <https://jrnل.nau.edu.ua/index.php/PIU/article/view/12798>.
7. P.D.D. Dominic and Handaru Jati. A comparison of Asian airlines websites quality: using a non-parametric test. International Journal of Business Innovation and Research. Vol. 5, No.5. 2011. – Pp. 599–623. URL: <https://doi.org/10.1504/IJBIR.2011.042451>.
8. Ісаков О.С. Дослідження моделей процесів тестування зручності використання програмних продуктів / О.С. Ісаков, О.Ю. Чердніченко, В.В. Мозгін, О.В. Янголенко // Вісник Національного технічного університету «ХПІ». Серія: Стратегічне управління, управління портфелями, програмами та проектами: зб. наук. пр. – Харків: НТУ «ХПІ», 2018. – №2 (1278). – С. 73-80.
9. Allison R., Hayes C., McNulty CAM, Young V. A Comprehensive Framework to Evaluate Websites: Literature Review and Development of GoodWeb. JMIR Form Res. Vol. 3, Issue 4, (2019), pp: e14372. URL: doi:10.2196/14372.
10. Канер С., Фолк Д., Нгуен Е.К. Тестування програмного забезпечення. – К.: ДіаСофт, 2001. – 544 с.
11. K. Fatima, N. Z. Bawany and M. Bukhari, «Usability and Accessibility Evaluation of Banking Websites.» 2020 International Conference on Advanced Computer Science and Information Systems (ICACSIS), Depok, Indonesia, 2020, pp. 247-256, doi: 10.1109/ICACSIS51025.2020.9263083.
12. Sarah Alismail, Wallace Chipidza. Accessibility evaluation of COVID-19 vaccine registration websites across the United States. Journal of the American Medical Informatics Association, Volume 28, Issue 9, September 2021, Pp. 1990–1995, <https://doi.org/10.1093/jamia/ocab105>.
13. Agrawal, G., Kumar, D. & Singh, M. Assessing the usability, accessibility, and mobile readiness of e-government websites: a case study in India. Univ Access Inf Soc 21, pp. 737–748 (2022). URL: <https://doi.org/10.1007/s10209-021-00800-8>.
14. Огляд програм для моделювання одягу. URL: <https://education.in.ua/overview-of-programs-for-modeling-clothes/>.
15. Smart pattern. Валентина. Про проект. URL: <https://smart-pattern.com.ua/uk/valentina/about/>.
16. Valentina. GitLab. URL: <https://gitlab.com/smart-pattern/valentina>.
17. Version control with GIT. URL: <https://www.atlassian.com/git/tutorials/what-is-git>.
18. Smart pattern. Валентина. Історія версій. URL: <https://smart-pattern.com.ua/uk/valentina/history/>.
19. Ubuntu Repository Packages. URL: <https://packages.ubuntu.com/search?keywords=valentina&searchon=names%20&suite=all§ion=all>.
20. Change Log. URL: <https://gitlab.com/smart-pattern/valentina/-/blob/develop/ChangeLog.txt>.
21. Revolutionizing garment-making in Italy with Valentina. URL: <https://librearts.org/2021/01/showcase-valentina-in-lavoro-tailor-shop/>.
22. Bitbucket. Valentina. URL: <https://bitbucket.org/dismine/valentina/src/master/>.

References

1. Economic-mathematical tools for building up a project team in the system of company's knowledge management / I. Chaikovska, T. Fasolko, L. Vaganova, O. Barabash // Eastern-European journal of enterprise technologies. – 2017. – № 3/3 (87). – P. 29–37.
2. Hovorushchenko T.O., Bodnar M.A., Kushnir V.O. Modern problems of the formation and analysis of software requirements. International Scientific and Technical Journal «Measuring and Computing Techniques in Technological Processes». – Khmelnytskyi, 2019. – P. 45–53. ISSN2219-9365. URL: <http://78.152.183.36/bitstream/123456789/8263/1/10.pdf>.
3. Hovorushchenko, T. & Popov, P.T. (2021). Method of developing the defect-free medical software by establishing the presence of residual defects. Proceedings of the 4th International Conference on Informatics & Data-Driven Medicine, 3038, pp. 11-21. ISSN 1613-0073. URL: <https://openaccess.city.ac.uk/id/eprint/27370/1/>.
4. I. Lopatto, T. Hovorushchenko, P. Popov and O. Pavlova, «Intelligent Multi-Agent System for Improving the Quality of Software by Taking into Account the Information of the Subject Area at All Stages of its Development.» 2021 11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS), Cracow, Poland, 2021, pp. 548–551, doi: 10.1109/IDAACS53288.2021.9660866.
5. T. Hovorushchenko and O. Pavlova, «Multi-Agent Technology for Assessing the Availability of Information at the Software Development Initial Stages.» 2021 IEEE 16th International Conference on Computer Sciences and Information Technologies (CSIT), LVIV, Ukraine, 2021, pp. 353–356, doi: 10.1109/CSIT52700.2021.9648832.

6. Troyan A.M., Modenov Y.B. The feasibility of automated testing to ensure the quality of software products. Problems of informatization and management. Volume 1, No.57-58. 2017. – P. 86–89. URL: <https://jrnlnau.edu.ua/index.php/PIU/article/view/12798>.
7. P.D.D. Dominic and Handaru Jati. A comparison of Asian airlines websites quality: using a non-parametric test. International Journal of Business Innovation and Research. Vol. 5, No.5. 2011. – Pp. 599–623. URL: <https://doi.org/10.1504/IJBIR.2011.042451>.
8. Isakov O.S. Study of models of software usability testing processes / O.S. Isakov, O.Yu. Cherednichenko, V.V. Mozgin, O.V. Yangolenko // Bulletin of the National Technical University «KhPI». Series: Strategic management, management of portfolios, programs and projects: coll. of science pr. – Kharkiv: NTU «KhPI», 2018. – No.2 (1278). – P. 73–80.
9. Allison R., Hayes C., McNulty CAM, Young V. A Comprehensive Framework to Evaluate Websites: Literature Review and Development of GoodWeb. JMIR Form Res. Vol. 3, Issue 4, (2019), pp: e14372. URL: [doi:10.2196/14372](https://doi.org/10.2196/14372).
10. Kaner S., Faulk D., Nguyen E.K. Software testing. – K.: DiaSoft, 2001. – 544 p.
11. K. Fatima, N. Z. Bawany and M. Bukhari, «Usability and Accessibility Evaluation of Banking Websites,» 2020 International Conference on Advanced Computer Science and Information Systems (ICACSIS), Depok, Indonesia, 2020, pp. 247-256, doi: 10.1109/ICACSIS51025.2020.9263083.
12. Sarah Alismail, Wallace Chipidza. Accessibility evaluation of COVID-19 vaccine registration websites across the United States. Journal of the American Medical Informatics Association, Volume 28, Issue 9, September 2021, Pp. 1990–1995, <https://doi.org/10.1093/jamia/ocab105>.
13. Agrawal, G., Kumar, D. & Singh, M. Assessing the usability, accessibility, and mobile readiness of e-government websites: a case study in India. Univ Access Inf Soc 21, pp. 737–748 (2022). URL: <https://doi.org/10.1007/s10209-021-00800-8>.
14. Overview of clothing modeling programs. URL: <https://education.in.ua/overview-of-programs-for-modeling-clothes/>.
15. Smart pattern. Valentina about the project. URL: <https://smart-pattern.com.ua/uk/valentina/about/>.
16. Valentina. GitLab. URL: <https://gitlab.com/smart-pattern/valentina>.
17. Version control with GIT. URL: <https://www.atlassian.com/git/tutorials/what-is-git>.
18. Smart pattern. Валентина. Історія версій. URL: <https://smart-pattern.com.ua/uk/valentina/history/>.
19. Ubuntu Repository Packages. URL: <https://packages.ubuntu.com/search?keywords=valentina&searchon=names%20&suite=all§ion=all>.
20. Change Log. URL: <https://gitlab.com/smart-pattern/valentina/-/blob/develop/ChangeLog.txt>.
21. Revolutionizing garment-making in Italy with Valentina. URL: <https://librearts.org/2021/01/showcase-valentina-in-lavoro-tailor-shop/>.
22. Bitbucket. Valentina. URL: <https://bitbucket.org/dismine/valentina/src/master/>.