

Пропозиція по тестуванню сервісу для створення онлайн діаграм

<https://www.onlinecharts.ru/>



3MICT

1. Вступ	2
1.1. Purpose	2
1.2. Background.....	3
1.3. Scope.....	3
2. Requirements for Test	3
3. Test Strategy	5
3.1. Testing Types	5
3.1.1. System and Usability Testing	5
3.1.2. Functional and UI Testing	6
3.1.3. Cross-platform Testing	8
3.1.4. Integration Testing	9
3.1.5. Localization Testing	9
3.1.6. Load Testing	10
3.1.7. Security and Access Control Testing	11
4. Resources	12
4.1. Tools	12
4.2. Roles	12
5. Project Milestones	13
6. Deliverables.....	13
6.1. Review	13

1. Вступ

1.1. Purpose

Безперервність і якість роботи будь-якого програмного забезпечення безпосередньо залежить від якості тестування програмного забезпечення на етапі розробки. Тестування і контроль якості є важливим етапом процесу розробки програмного забезпечення.

1.2. Background

Сервіс для створення онлайн діаграм - onlinecharts.ru.

1.3. Scope

Метою тестування сервісу onlinecharts.ru є перевірка коректної роботи його функціональності, зручності сервісу для користувача, а також виявлення побажань користувача з розширення або виправлення існуючих функціоналів. В процесі тестування також повинна бути перевірена правильність взаємодії onlinecharts.ru з іншими підсистемами.

Підсумком процесу тестування повинен стати розгорнутий огляд, що дає розробникам, менеджерам і користувачам даного продукту картину щодо зручності користування сервісом кінцевим користувачам.

2. Requirements for Test

Сервіс повинен задовольняти потреби користувача у створенні онлайн діаграм. Користувач повинен мати можливість створювати власні діаграми і графіки за допомогою 5 простих кроків.

Вимоги до сервісу:

- Користувач повинен мати можливість обирати тип діаграми, а також персоналізувати за допомогою додаткових параметрів, які впливають на зовнішній вигляд діаграми;

- Користувач повинен мати можливість додавати необхідні дані для побудови діаграми;
- Користувач повинен мати можливість обирати підписи і шрифти для діаграми;
- Користувач повинен мати можливість попередньо переглянути побудовану діаграму, для перевірки коректного відображення;
- Користувач повинен мати можливість надіслати створену діаграму за допомогою email, зберегти діаграму в одному із запропонованих форматів(svg, png, jpg, pdf, csv) на локальний диск. Зареєстровані користувачі мають можливість зберегти створену діаграму у особистому кабінеті;
- Користувач повинен мати можливість опублікувати діаграму у соціальних мережах(facebook, twitter, google plus);
- Користувач повинен мати можливість зареєструватися та авторизуватись у системі;
- Авторизований користувач повинен мати можливість переглянути раніше створенні ним діаграми у особистому кабінеті;
- Авторизований користувач повинен мати можливість змінити персональну інформацію, пароль, а також видалити свій обліковий запис у особистому кабінеті;
- Користувач повинен мати можливість обрати одну із 6-ти запропонованих мов для сервісу(українська, російська, англійська, іспанська, німецька, нідерландська);
- Користувач повинен мати можливість скористатися формою зворотнього зв'язку у разі виникнення питань або труднощів під час користування сервісом.

3. Test Strategy

Планується три етапи проведення процесу тестування. Таким чином, досягається максимальна деталізація глибини тестування, що, в свою чергу, дозволяє більш точно визначити використання ресурсів, а так само дозволяє розробникам проекту почати виправляти дефекти на ранніх етапах.

На першому етапі буде проведено системне тестування, засноване на коректному сценарії роботи користувача з додатком, що описують повний цикл роботи з додатком. В ході системного тестування планується виявити usability-недоробки, а також запропонувати розширити функціонал з метою підвищення зручності роботи з додатком.

На другому етапі, в ході проведення функціонального тестування, планується виявити помилки, засновані на взаємодії функціоналів шляхом реалізації нетривіальних сценаріїв.

На третьому етапі буде проведено ряд тестів, що визначають здатність додатки працювати з певним навантаженням. В силу ряду причин (для цієї програми розроблено для роботи одиночними користувачами; ризик навантаження в умовах реальної експлуатації мінімальний) навантажувальний тестування проводитиметься без задіяння інструментів для моделювання навантаження.

3.1. Testing Types

3.1.1. System and Usability Testing

Goal:

Переконалися, що користувач може використовувати основні функціонали програми в процесі реального використання.

Перевірити ступінь зручності сервісу для користувача на підставі загальних критеріїв і суб'єктивного сприйняття.

Process description:

- Емуляція повного циклу використання програми: створення облікового запису, авторизація у системі, зміна персональних даних та паролю, створення діаграм різних типів, створення різноманітних оформлень діаграм(тип, форма, колір, фон, шрифт). Збереження створених діаграм у особистому кабінеті, а також на локальному диску;
- Аналіз зручності роботи користувача з додатком на основі виникаючих потреб, а також на основі порівняння з існуючими сервісами зі схожими функціоналами.

Testing approach:

В процесі тестування буде застосовано ad-hoc тестування.

Completion Criteria:

Сервіс задовольняє потреби користувача по створенню онлайн діаграм.

3.1.2. Functional and UI Testing

Goal:

Виявити функціональні помилки, засновані на повномасштабному використанні функціоналів системи. Тестування різноманітних комбінацій у створенні діаграми.

Process description:

Виробляється умовний поділ додатка на функціонали, на підставі цього поділу проводиться цілеспрямоване тестування:

- Створення діаграми
 - Стовпчаста
 - Лінійна
 - З областями
 - Кругова
 - ХУ графік
 - Радіальна
 - Точкова
 - Бульбашкова
 - Полярні бульбашки
 - Спідометр
 - Стовпчаста/Лінійна
 - Піраміда
- Імпорт даних для діаграми
- Збереження діаграми у різних форматах на локальному диску
- Відправлення діаграми в email

Testing approach:

В процесі тестування під час створення тест кейсів буде застосовано наступні техніки тест дизайну:

- Класи еквівалентності;
- Аналіз граничних значень;
- Pairwise тестування.

Остання техніка, pairwise, допомагає зменшити кількість тест кейсів і відповідно зменшити використання ресурсів. Таким чином, кількість тест кейсів для однієї діаграми було значно зменшено (наприклад для стовпчастої діаграми, кількість тест кейсів було зменшено з 14 млрд. до 153, що можна побачити на рис. 3.1). Для підрахунку кількості тест кейсів був використаний додаток TConfig.

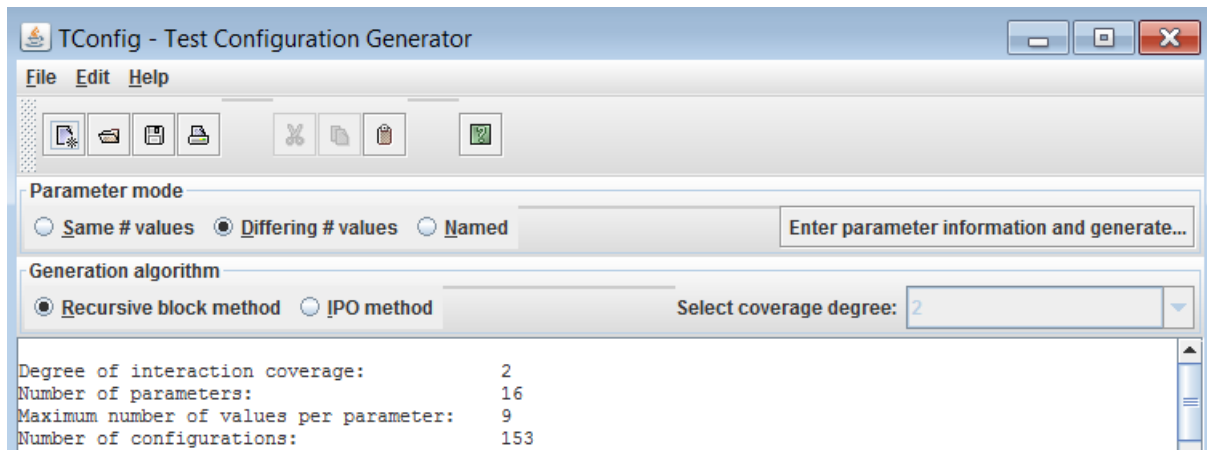


Рис. 3.1 Додаток TConfig

Completion Criteria:

- всі заплановані області протестовані
- всі знайдені помилки і зауваження зафіксовані

3.1.3. Cross-platform Testing

Goal:

Перевірити коректну роботу програми на різних операційних системах, а також у різних браузерях.

Process description:

- Емуляція повного циклу використання сервісу у різноманітних комбінаціях ОС – браузер.

- Аналіз коректності відображення та функціонування з точки зору користувача.

Testing approach:

Необхідно провести smoke тестування основного функціоналу з метою виявлення помилок у різних комбінаціях ОС – браузер.

Completion Criteria:

- всі заплановані області протестовані
- всі знайдені помилки і зауваження зафіксовані

3.1.4. Integration Testing

Goal:

Перевірити коректну роботу програми з соціальними мережами

Process description:

- Перевірити коректність публікації діаграм у соціальних мережах (Facebook, Twitter, Google Plus).

Completion Criteria:

- всі заплановані області протестовані
- всі знайдені помилки і зауваження зафіксовані

3.1.5. Localization Testing

Goal:

Перевірити коректний переклад та відображення частин системи у 6-ти запропонованих локалізаціях.

Process description:

- Емуляція повного циклу використання сервісу у запропонованих локалізаціях.
- Аналіз коректності відображення та перекладу функціоналу.

Testing approach:

Необхідно провести smoke тестування основного функціоналу з метою виявлення помилок у різних локалізаціях.

Completion Criteria:

- всі заплановані області протестовані
- всі знайдені помилки і зауваження зафіксовані

3.1.6. Load Testing***Goal:***

Перевірити роботу системи під час великого навантаження. Визначити час відгуку системи, та час малювання діаграм.

Process description:

- Емуляція повного циклу використання сервісу одразу декількома користувачами.
- Створення діаграм із великою кількістю даних.
- Аналіз коректності роботи функціоналу та задовільного часу відгуку системи.

Testing approach:

Тестування буде проводитися за допомогою засобу Apache Jmeter. Даний додаток дозволить емулювати велику кількість користувачів, а також створити необхідні звіти по даному типу тестування.

Для створення навантаження, необхідно виділити комп'ютерні ресурси, розмір яких залежить від кількості користувачів та складності сценарія для виконання.

Completion Criteria:

- Час відгуку задовольняє вимоги ($0.8 < APDEX < 1.0$)
- Під час навантаження усі запити коректно оброблюються системою.

3.1.7. Security and Access Control Testing

Goal:

Перевірити роботу системи на три основні принципи:

- Конфіденційність
- Цілісність
- Доступність

Process description:

- Перевірка програмного забезпечення на наявність вразливих місць з точки зору безпеки.
 - XSS
 - CSRF
 - Code injections
 - SSI

Completion Criteria:

- всі заплановані області протестовані
- всі знайдені помилки і зауваження зафіксовані

4. Resources

4.1. Tools

Під час організації процесу тестування будуть використані наступні додатки/програми:

- Jira – для фіксації дефектів та розподілення роботи між працівниками
 - Structure – пагін для Jira, для додавання нового типу запису як Test, а також для створення структури тест кейсів
 - Zephir – для фіксації релізів продукту, а також для додавання артефакту під назвою TestCycle, для фіксації виконання тест кейсів
- SQL Management Studio – для підключення та управління базою даних.
- Fiddler – для відслідковування коректності відправлення запитів та отримання відповідей на них.
- Apache Jmeter – для виконання тестування навантаження.
- TConfig – для створення тест-кейсів, за допомогою техніки тест дизайну pairwise

4.2. Roles

Для повного циклу тестування необхідно найняти наступних спеціалістів:

- QA Tech Lead – для керування процесом тестування

- Performance Test Engineer – для виконання та аналізу тестування навантаження.
- 2 QA Engineer – для виконання великої кількості мануальних тест кейсів.

У випадку впровадження автоматизації тестування у майбутньому, необхідно також найняти QA Automation Engineer.

5. Project Milestones

Milestone Task	Effort	Start Date	End Date
Plan Test	2 man/hours	01.07.17	01.12.17
Design Test	1 man/hours	02.07.17	03.07.17
Execute Test	6 man/hours	03.07.17	08.07.17
Analyse Test	1 man/hours	08.07.17	09.07.17
Prepare Review	3 man/hours	09.07.17	09.07.17

6. Deliverables

6.1. Review

Кінцевим підсумком проведення тестування повинен стати оформлений кінцевий результат процесу тестування з описаними дефектами, а також рекомендаціями щодо поліпшення продукту з точки зору кінцевого користувача. Також до звітної документації повинен бути доданий окремий звіт із тестування навантаження.

