

СИСТЕМА ОТСЛЕЖИВАНИЯ ДЕФЕКТОВ ПРОГРАММНЫХ ПРОДУКТОВ И ЕЕ ЭРГОНОМИЧЕСКОЕ ОБЕСПЕЧЕНИЕ

Глик А.Г.

*Белорусский государственный университет информатики и радиоэлектроники,
г. Минск, Республика Беларусь*

Научный руководитель: Медведев О.С. – ст. преподаватель кафедры ИПиЭ

Аннотация. Данная статья посвящена анализу предметной области, аналогов, а также особенностям разработки приложения для структурированного хранения, обработки и изменения данных, необходимых для функционирования отдела тестирования предприятия или любого другого отдела, схожего по назначению. Рассматриваются решаемые системой задачи, его структура и средства разработки данной системы.

Ключевые слова: программное обеспечение, приложение, отслеживание дефектов, тестирование

Введение. На данный момент происходит интенсивная автоматизация работ, трудно поддающихся формализации и общая автоматизация разработки программного обеспечения. Исходя из динамики прошлых лет видно, что рынок требует не только усовершенствования старых систем, но и разработки совершенно новых. Их качество должно соответствовать довольно высоким требованиям, ведь учет и дальнейшая работа с дефектами программных средств является неотъемлемой частью производства; упущенные или необработанные дефекты могут привести к серьезным временным и материальным потерям, в крайнем случае – судебным разбирательствам, что, очевидно, губительно для любого производства или бизнеса.

Цель разработки модели данного программного средства заключается в предоставлении базового решения, что в дальнейшем позволит разработать, и удовлетворить обще-рыночную потребность. Данное программное средство позволит структурированно хранить, обрабатывать и изменять данные необходимые для функционирования отдела тестирования предприятия или другого отдела, схожего по назначению.

Основная часть. Тестирование, исправление дефектов и интеграция нового функционала являются неотъемлемой частью работы любого разработчика и непрерывно сопровождают продукт на каждой стадии цикла разработки программного обеспечения.

За счет интеграции системы отслеживания дефектов в рабочее окружение разработки, организации способны уменьшить количество ошибок, связанных с управлением дефектами, повысить точность предоставляемой информации, улучшить эффективность, повысить общий уровень автоматизации трудового процесса, а так же облегчить участникам цикла разработки некоторые из повседневных задач.

Система включает в себя частичную автоматизацию таких задач, как мониторинг и управление дефектами программных продуктов, анализ их воздействия, управление приоритетами, отслеживание процессов устранения дефектов. Назначение системы отслеживания дефектов программных продуктов – регистрация, мониторинг и управление дефектами в соответствии с функциональностью и характеристиками программного продукта, предоставляемого пользователями.

Данная система решает следующие задачи:

1. Отслеживание дефектов программных продуктов и процессов их устранения: система позволяет эффективно регистрировать и отслеживать дефекты в программном обеспечении, облегчая процесс их обнаружения и фиксации; позволяет отслеживать состояние процесса устранения дефектов, что способствует более прозрачной работе над их исправлением.

2. Анализ воздействия дефектов и управление их приоритетами: система предоставляет инструменты для анализа воздействия дефектов на работу программного продукта, что помогает определить их важность и приоритетность.

3. Оценка эффективности: путем анализа данных системы можно оценивать эффективность процесса управления дефектами и вносить необходимые улучшения.

В системе, являющейся базовым решением, необходимо реализовать лишь необходимый для правильной работы системы функционал, что в дальнейшем позволит легко масштабировать и интегрировать само программное решение. В данный список следует включить: регистрация пользователей, идентификация пользователей, аутентификация пользователей, авторизация пользователей, добавления новых программных продуктов, система выводит список заранее введенных отслеживаемых систем, добавление новых дефектов, просмотр списка отслеживаемых дефектов, просмотр информации об отслеживаемом дефекте, установка приоритетов отслеживаемых дефектов.

Для реализации вышеперечисленного функционала достаточно одной общей базы данных вне зависимости от ее класса [1], она может быть как централизованная, так и распределенная. Современным и наилучшим решением будет применение локальной базы данных на ПК каждого пользователя, которая может быть в дальнейшем связана между собой – это может быть как посредник в виде общего сервера, так и прямая связь между клиентами пользователей.

Приложение разрабатывается с выбранным стеком (набором) технологий, или максимально похожим на него, в силу принятой индустриальной практики, при которой команды разработчиков стремятся к максимальной эффективности и оптимизации процесса создания программного обеспечения [2].

Если стек технологий достаточно гибок для реализации требуемого функционала и способен беспрепятственно интегрироваться в требуемое окружение, то предпочтение отдается знакомым инструментам и технологиям, что позволяет избежать излишних затрат времени и ресурсов на изучение новых стеков для каждого нового проекта. Такой подход содействует ускорению процесса разработки, минимизации рисков и обеспечивает более высокое качество конечного продукта благодаря глубокому знанию выбранного стека технологий со стороны команды разработчиков.

Стек, задействованный в разработке данного программного средства, включает в себя фреймворк React, языки программирования JavaScript и TypeScript, и такие технологии как SASS, NodeJs и ElectronJs. Последний выступает в качестве некоторой прослойки, обеспечивающей связь между движком Chromium и операционной системой. Данный стек полностью соответствует требованиям, перечисленным выше, и используется для разработки данного программного средства.

На рисунке 1 представлена диаграмма вариантов использования системы, показывающая разнообразие и взаимосвязь основных функций системы.

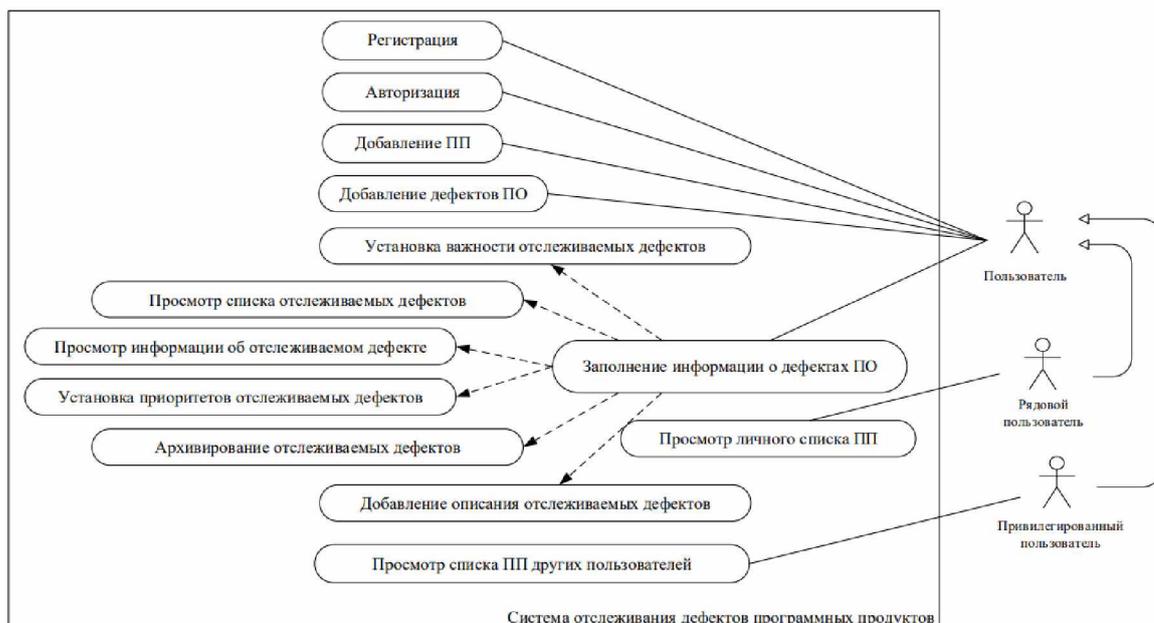


Рисунок 1 – Диаграмма вариантов использования

Заключение. В данной статье были рассмотрены анализ предметной области разрабатываемой системы и ее аналогов, средства разработки приложения, намечена его архитектура. Было выявлено, что разрабатываемое приложение включает в себя основные алгоритмы, структуры, интерфейсы и другие ключевые компоненты, которые определяют функциональность и возможности программы для последующих модификаций и расширений. Приложение решает вопрос структурированного хранения, обрабатывания и изменения данных, внутри отдела тестирования предприятия.

Список литературы

1. Базы данных [Электронный ресурс]. – Режим доступа: <https://en.wikipedia.org/wiki/database>: 13.01.2024
2. Орлов, С. Технологии разработки программного обеспечения / С. Орлов. – СПб.: Питер, 2002. — 464 с.

UDC 004.414.2:331.101.1

TRACKING SOFTWARE DEFECTS SYSTEM

Glick A.G.

Belarusian State University of Informatics and Radioelectronics, Minsk, Republic of Belarus

Medvedev O.S. – senior lecturer of the Department of EPE

Annotation. This article is dedicated to the analysis of the subject area, analogues, as well as peculiarities of developing an application for structured storage, processing, and modification of data required for the functioning of the testing department of an enterprise or any other department with a similar purpose. The tasks solved by the system, its structure, and the means of developing this system are considered.

Keywords: software, application, defect tracking, testing