

нии. Результаты апробации представлены в виде «предложений по структуре и механизму функционирования национальной системы квалификации». Имея продолжительный опыт, можно сформулировать предварительные замечания, опираясь на базовые принципы в создании национальных систем квалификации. На текущий момент Министерством труда и социальной защиты утверждено 12 профессиональных стандартов.

В различных странах существуют собственные модели гармонизации профессиональных и образовательных стандартов. При всех различиях, неизменными остаются два принципиально важных момента:

1. Приоритет профессиональных стандартов.
2. Обязательность социального партнерства как наиболее эффективного способа согласования и реализации интересов заинтересованных участников процесса гармонизации профессиональных и образовательных стандартов.

Заключение

Разработка профессиональных стандартов положительно скажется на качестве человеческого капитала, индексе экономики знаний и уровне экономического развития страны, так как основным условием координации рынка труда и рынка образования является гармонизация профессиональных и образовательных стандартов. Первые раскрывают компетенции с точки зрения их востребованности на рынке труда, вторые – результаты образования.

Список использованной литературы

1. Интеграция содержания дисциплин естественнонаучного и общепрофессионального циклов учебного плана технического вуза / Н.Г. Серебрякова, Л.С. Шабека, Е.В. Галушко, // Профессиональное образование. - 2017. – № 2, С. 19–23.
2. Попов, А.И. Проектирование системы обучения будущих инженеров сельскохозяйственного производства инновационной деятельности / А.И. Попов, В.М. Синельников, Н.Г. Серебрякова// Исследования и результаты. – 2017. – № 3. – С. 413-420.
3. Серебрякова, Н.Г. Современные концепции инженерного образования: анализ в рамках компетентностного подхода/ Н.Г. Серебрякова // Высшая школа. – 2017. – № 6, С. 23–27.

УДК 004.415.53

ФОРМИРОВАНИЕ ТЕСТОВОЙ СТРАТЕГИИ ДЛЯ ТЕСТИРОВАНИЯ BSS СИСТЕМ

**Н.Г. Серебрякова, канд. пед. наук, доцент,
И.Ю. Русецкий, студент, С.М. Люлькин, студент**
БГАТУ, г. Минск, Республика Беларусь

Аннотация. Объектом исследования является тестовая стратегия для тестирования BSS-систем при условии отсутствия оборудования. Целью проекта является планирование и выработка тестовой стратегии для тестиро-

вания смоделированной BSS-системы. Выполнено логическое и физическое моделирование данных.

Abstract. The object of the study is a test strategy for testing BSS systems in the absence of hardware. The goal of the project is to plan and develop a test strategy for testing a simulated BSS system. Logical and physical data modeling is performed.

Ключевые слова: тестовая стратегия, OSS/BSS, ресурсы, test-CASES, система, требования, багтрекинг-системы, тестирование.

Keywords: test strategy, OSS/BSS, resources, test-CASES, system, requirements, bug tracking systems, testing.

Введение

Целью данной работы является поиск и выбор стратегии тестирования BSS-системы при условии отсутствия необходимого оборудования.

Основная часть

Составляющие тестовой стратегии

Составляющей частью планирования тестирования (как отдельного документа или же процесса планирования в целом) является стратегия тестирования.

Стратегия тестирования – это план проведения работ по тестированию системы или её модуля, учитывающий специфику функциональности и зависимости с другими компонентами системы и платформы. Стратегия определяет типы тестов, которые нужно выполнять для данного функционала системы, включает описание необходимых подходов с точки зрения целей тестирования и может задавать описания или требования к необходимым для проведения тестирования инструментам и инфраструктуре [4].

Тестовая стратегия отвечает на вопросы:

- как, каким образом тестирование даст ответ, что данная функциональность работает;
- что нужно сделать и чем пользоваться из инструментальных средств для достижения целей тестирования;
- когда определённая функциональность будет тестироваться и когда ожидать получения результатов.

В общем случае разработка стратегии тестирования способствует разбору задачи тестирования на составляющие, выделение тестовых областей и более полное понимание задачи тестирования в конкретном проекте.

Для разработанных логической и физической моделей необходимо выбрать необходимые типы тестов, покрывающие разработанную функциональность. Инструментальными средствами для достижения целей тестирования являются система отслеживания ошибок и система управления тестированием. Возможность запуска тестирования определяется готовностью тестового окружения и тестовой документации, по которой будет выполняться тестирование.

Система отслеживания ошибок

Для лучшего контроля качества разработки используются различные системы отслеживания ошибок (bug tracking system).

Система отслеживания ошибок представляет собой прикладную программу, разработанную с целью помочь команде разработчиков учитывать и

контролировать ошибки и неполадки, найденные в программах, пожелания пользователей, а также следить за процессом устранения этих ошибок и выполнения или невыполнения пожеланий. Ошибки могут добавляться в систему отслеживания ошибок заказчиками, командой тестирования и самими разработчиками. Чаще всего для всех участников процесса настроены определенные права доступа для организации безопасности. Заведенную по неосторожности ошибку удалить сможет только администратор системы отслеживания ошибок. Это сделано для того, чтобы собирать исторические метрики по количеству, критичности, типу ошибок, на основе которых осуществляется поиск узких мест в работе, влияющих на качество проекта.

Главный компонент такой системы – база данных, содержащая сведения об обнаруженных дефектах. Эти сведения могут включать в себя:

- 1) номер (идентификатор) дефекта;
- 2) кто сообщил о дефекте;
- 3) дата и время, когда был обнаружен дефект;
- 4) версия продукта, в которой обнаружен дефект;
- 5) серьёзность (критичность) дефекта и приоритет решения;
- 6) описание шагов для выявления дефекта (воспроизведения неправильного поведения программы);
- 7) кто ответственен за устранение дефекта;
- 8) обсуждение возможных решений и их последствий;
- 9) текущее состояние (статус) дефекта;
- 10) версия продукта, в которой дефект исправлен.

Кроме того, развитые системы предоставляют возможность прикреплять файлы, помогающие описать проблему (например, дампы памяти или скриншоты).

Наиболее известная система отслеживания ошибок Atlassian JIRA[2]. Atlassian JIRA – коммерческая система отслеживания ошибок, предназначена для организации общения с пользователями, хотя в некоторых случаях систему можно использовать для управления проектами. Разработана компанией Atlassian Software Systems. Платная. Имеет веб-интерфейс. Система позволяет работать с несколькими проектами. Для каждого из проектов создаёт и ведёт схемы безопасности и схемы оповещения.

Среди бесплатных багтрекинг-систем стоит выделить Redmine[4]. Redmine – бесплатная система для управления проектами и задачами (в том числе и отслеживания ошибок). Можно одновременно вести несколько проектов. Имеет гибкую систему доступа, основанную на ролях, ведение новостей проекта, документов и управление файлами, оповещение об изменениях с помощью RSS-потоков и электронной почты, возможность самостоятельной регистрации новых пользователей.

В учебном проекте в качестве системы отслеживания ошибок была выбрана система Jira. Несмотря на коммерческую основу выбранной системы отслеживания ошибок, Jira имеет возможность интеграции с системами управления тестированием, средствами гибких методологий разработки, системами хранения проектной технической документации.

Отслеживание ошибок достигается во многом за счет глубокой интеграции с исходным кодом и средой разработки.

Отслеживание проекта: JIRA позволяет централизованно управлять всеми проектами, не теряя из виду общую картину. Для любой команды важно, чтобы внутри нее люди могли легко обмениваться информацией и обратиться за помощью, когда им это необходимо.

Ключевые особенности JIRA:

- контроль и планирование проекта;
- служба техподдержки;
- доступность с мобильных девайсов;
- возможность кастомизации рабочего процесса;
- составление отчетности;
- система учета ошибок (Bug Tracking);
- интеграция с Git;
- поддержка дополнений, расширений, плагинов.

Простой, интуитивно понятный интерфейс JIRA позволяет сотрудничать с коллегами по команде и выполнять работу более эффективно. Интерфейс создания дефекта приведен на рисунке 1.

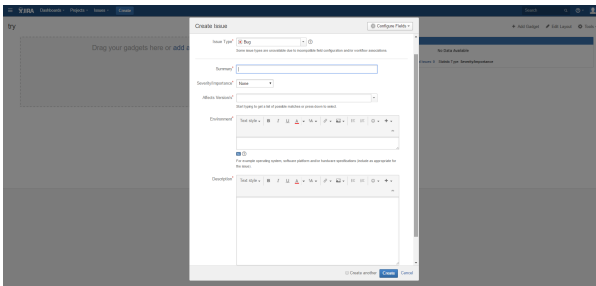


Рисунок 1 – Интерфейс Jira для создания дефекта

Заключение

Разработана спецификация BSS-решения для сценариев совершения голосового вызова; составлены заголовки для системного тестирования; детально описан один тестовый сценарий.

Проверяемые функции программного обеспечения, которые принимают участие в сценариях совершения голосовых вызовов:

- возможность совершать голосовые вызовы;
- возможность принятия вызовов;
- достаточность баланса клиента;
- изменение статусов клиента, абонента после совершения вызова.

Выбранная стратегия тестирования позволяет эффективно тестировать телекоммуникационное решение предприятия связи и обеспечивать

качество продукта с учетом непрерывного развития системы. В рамках улучшения проекта предусматривается участие команды автоматизаторов, что позволит снизить затраты на ручное тестирование за счет автотестов.

Список использованной литературы

1. Серебрякова, Н.Г. Современные концепции инженерного образования: анализ в рамках компетентностного подхода/ Н.Г. Серебрякова // Высшая школа. – 2017. – № 6, С. 23–27.

2. Попов, А.И. Проектирование системы обучения инновационной деятельности будущих инженеров сельскохозяйственного производства / А.И. Попов, В.М. Синельников, Н.Г. Серебрякова // Исследования, результаты. – 2017. – N 3. – С. 413–420.

3. Серебрякова, Н.Г. Интеграция содержания дисциплин естественнонаучного и общепрофессионального циклов учебного плана технического вуза / Н.Г. Серебрякова, Л.С. Шабека, Е.В. Галушко, // Профессиональное образование. – 2017. – № 2, С. 19–23.

4. Быков, В.Л. Информатика: учебно-методическое пособие для студентов вузов группы специальностей 74 06 «Агроинженерия» / В.Л. Быков, Н.Г. Серебрякова; Минсельхозпрод РБ, УО БГАТУ, Кафедра прикладной информатики. – Минск : БГАТУ, 2013. – 656 с.

УДК 519.2

ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ДЛЯ РЕАЗАЦИИ НАГРУЗОЧНОГО И СТРЕССОВОГО ТЕСТИРОВАНИЯ WEB-ПРИЛОЖЕНИЙ

Н.Г. Серебрякова¹, канд. пед. наук, доцент,

Н.И. Болтянская², канд. техн. наук, доцент,

С.М. Люлькин¹, студент,

А.А. Подсеваткина², магистрант

¹БГАТУ, г. Минск, Республика Беларусь

²Таврический государственный агротехнологический университет
имени Дмитрия Моторного, г. Мелитополь, Украина

³Белорусский национальный технический университет,
г. Минск, Республика Беларусь

Аннотация. В статье рассматривается первый этап разработки программного обеспечения для проведения нагрузочного и стрессового тестирования *web*-приложений.

Abstract. This article discusses the first stage of software development for load and stress testing of *web* applications.

Ключевые слова: нагрузочное тестирование, стрессовое тестирование, *web*-приложения.

Keywords: load testing, stress testing, web applications.