

интерфейс пользователя реализуется при помощи языка программирования JavaScript.

Применение подобного подхода существенно ускоряет процесс непрерывной интеграции посредством использования распределенных вычислений при выполнении интеграционного тестирования, что позволяет наиболее эффективно использовать преимущества практики непрерывной интеграции. Предлагаемый подход может быть внедрен в процесс разработки программного средства. Наибольшая эффективность достигается при внедрении в процесс разработки комплексного программного средства со строгими требованиями к качеству и большим набором интеграционных тестов.

Список использованных источников:

1. Дюваль, П. М. Непрерывная интеграция. Улучшение качества программного обеспечения и снижение риска / П. М. Дюваль. – М. : Вильямс, 2008. – 240 с.
2. Хамбл, Д. Непрерывное развертывание программного обеспечения / Д. Хамбл. – М. : Вильямс, 2011. – 432 с.
3. Кристин, Л. Scrum: гибкая разработка программного обеспечения / Л. Кристин. – М. : Вильямс, 2011. – 432 с.

## ПРИНЦИПЫ ПОВЫШЕНИЯ КАЧЕСТВА СПЕЦИФИКАЦИЙ ТРЕБОВАНИЙ WEB-ПРИЛОЖЕНИЙ

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

*Чиркова А. Ю.*

*Бахтизин В. В. – к. т. н., доцент*

В настоящее время создается огромное количество web-приложений, однако, большая их часть являются неудовлетворительными для конечного пользователя или заказчика или даже заканчиваются провалом. Во многих случаях причиной этому служит плохо составленная спецификация требований.

Разработке спецификации требований к web-приложениям зачастую уделяется недостаточно внимания в связи с тем, что web-приложения стереотипно не рассматриваются как сложные программные средства. Однако, вне зависимости от сложности и шаблонности разрабатываемого web-приложения, спецификация должна состоять из качественных требований. Качественные требования – это требования, которые выражают то, что основные заинтересованные лица действительно хотят получить как результат работы разрабатываемого программного продукта [1].

В спецификациях требований к web-приложениям нужно описать рамки проекта, это позволит избежать использования лишней информации. Рамки проекта пишутся в виде сценариев работы пользователей с web-приложением и описывают общую функциональность и интеракции с интерфейсом.

Ключевой частью спецификации является описание информационной архитектуры и интерфейсов. Информационная архитектура определяет то, как будет выглядеть и работать web-приложение с пользователями.

Информационная архитектура состоит из структуры web-приложения (так называемые высокоуровневые прототипы), шаблонов страниц (низкоуровневые прототипы, описывающие непосредственно интерфейс), описания контента (описание содержания каждой страницы web-приложения).

Кроме того, с целью сокращения временных затрат на приемочные квалификационные испытания рекомендуется включать в спецификацию требований к web-приложениям описание верификации готового приложения.

При создании спецификации требований для web-приложения также следует руководствоваться ключевыми принципами, перечисленными в [2]:

Принцип выявления наиболее значимых целей и задач продукта. В спецификации требований важно описать, какую выгоду, ценность или результат предоставляет проект.

Принцип применения различных техник при сборе требований. В настоящее время существует несколько техник сбора требований, но все они имеют свои достоинства и недостатки. Совокупность применения нескольких техник позволяет оптимизировать процесс сбора требований.

Принцип квантификации требований, или определения количественных мер в требованиях. Использование чисел в требованиях является базовым и мощным методом определения качества требования.

Принцип разделения задач и решений. Требование должно быть составлено и абстрагировано таким образом, чтобы оно четко и однозначно определяло задачу, а не конкретное решение, которое может не описывать все ожидаемые результаты, быть недостаточно эффективным и т.д.

Принцип валидации требований. Все требования должны быть понятны всем участникам проекта. Все заинтересованные лица должны подтвердить, что они понимают все описанные требования, и что сформулированные требования соответствуют поставленным задачам.

Принцип использования методов управления качеством требований. Для снижения влияния субъективного мнения на оценку качества требований к программным средствам предлагается внедрять в процесс разработки спецификации метод управления качеством требований, базирующийся на наборе методологий совершенствования процессов CMMI[3].

Таким образом, предложенные принципы значительно расширят подходы, которые применяются в настоящее время для создания спецификаций требований к разрабатываемым программным средствам, в том числе и к web-приложениям. Не только в совокупности, но и по отдельности предложенные принципы улучшат качество спецификаций требований, а, следовательно, и качество самих программных средств.

Список использованных источников:

1. T. Gilb, "Competitive Engineering: A Handbook for Systems Engineering, Requirements Engineering, and Software Engineering Using Planguage," Elsevier Butterworth-Heinemann, Boston, 2005.
2. А. Ю. Чиркова, В. В. Бахтизин. Принципы улучшения качества спецификаций требований программных средств // международная научно-техническая конференция, приуроченная к 50-летию МРТИ-БГУИР: материалы конференции. – Минск: БГУИР, 2014.
3. А. Ю. Чиркова, В. В. Бахтизин. Управление качеством требований при разработке программных средств дистанционного обучения // Дистанционное обучение – образовательная среда XXI века : материалы VIII междунар. науч.- метод. конф. (Минск, 5–6 декабря 2013 года). – Минск : БГУИР, 2013. – 407 с.

## **ТЕХНОЛОГИЯ ПРОЦЕССА АВТОМАТИЗАЦИИ РАСПРЕДЕЛЕНИЯ И УЧЕТА УЧЕБНОЙ НАГРУЗКИ ПРЕПОДАВАТЕЛЕЙ В УЧРЕЖДЕНИЯХ ОБРАЗОВАНИЯ С ИНТЕГРИРОВАННОЙ ФОРМОЙ ОБУЧЕНИЯ**

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

*Гилевский П. Г.*

*Скудняков Ю. А. – канд. техн. наук, доцент*

Одной из актуальных проблем работы кафедр учебного заведения является распределение и учет выполнения учебной нагрузки профессорско-преподавательского состава. Выполнение этой работы занимает много времени, неизбежны ошибки и многочисленные корректировки.

Проблемой автоматизации управления своими подразделениями занимаются многие ВУЗы Беларуси. На текущий момент нерешенной остается задача автоматизации процесса распределения и учета выполнения учебной нагрузки профессорско-преподавательского состава. Для выполнения данной работы, предварительно необходимо собрать полную информацию об учебных планах, кафедрах, учебных дисциплинах, штатном расписании и учебных группах. После сбора и систематизации данной информации рассчитывается текущее планирование учебной нагрузки преподавателей. Данный процесс занимает очень много времени, и при этом неизбежны ошибки и многочисленные корректировки, которые вызваны человеческим фактором.

Учреждения образования с интегрированной формой обучения сталкиваются с дополнительной сложностью в вычислении учебной нагрузки преподавателей, которая вызвана тем, что существует высшее и среднее специальное образование. Для каждой ступени образования установлены собственные нормы времени для различных видов контроля и занятий (экзамен, дифференцированный зачет, курсовой проект, курсовая работа, организационно-воспитательные консультации и т.д.), которые используются в планировании педагогической нагрузки преподавателей. Многие из этих норм зависят от количества студентов в группе (подгруппе). Также большую сложность вызывает учет занятий, проводимых в виде точных лекций.

В Минском государственном высшем радиотехническом колледже (МГВРК), который относится к учреждению образования с интегрированной формой обучения, все расчёты по учебной нагрузке выполняются при помощи программного пакета Microsoft Excel и, в настоящее время, в таблице «Текущее планирование» содержится более 1500 записей, которые трудно просматривать и редактировать.

К сожалению, несмотря на наметившиеся положительные тенденции в использовании программного пакета Microsoft Excel, распределение и учет выполнения учебной нагрузки до сих пор выполняется практически вручную и при этом не может быть и речи об автоматизации.

Такая работа имеет множество негативных аспектов, таких как:

- отсутствие единой базы и, как следствие, необходимость хранения всех данных в нескольких файлах;
- ручной ввод всех данных и отсутствие их проверки приводит к тому, что малейшая ошибка дает неверные расчеты и все действия выполняются заново;
- при изменении количества обучающихся в группе (подгруппе) – всю нагрузку для данной группы (подгруппы) необходимо вручную пересчитывать;
- настройка вида таблиц, вставка формул и оформление отнимают много времени;
- невозможность одновременной работы с программой нескольких пользователей;
- отсутствие защиты от повреждения или случайного удаления данных или файлов.