

Класс «Program» — стандартный класс при создании приложения в котором определена точка входа для приложения, а именно метод Main().

Класс «sign\_up» — класс для авторизации пользователей.

Класс «login\_in» — класс для регистрации пользователей в базе данных.

Класс «checkUser» — класс для проверки ролей пользователей.

Класс «about\_author» — класс с информацией об авторе проекта.

Класс «Resources» — класс, содержащий ресурсы, вложенные в сборку.

Класс «Settings» — класс для определения конфигурации приложения.

Класс «search\_form» — класс для осуществления поиска.

Класс «PanelAdmin» — класс для осуществления администрирования в приложении.

Класс «DB» — класс для реализации связи с базой данных.

Класс «md5» — класс для реализации алгоритма хеширования.

Класс «lessonsRedactor» — класс, отвечающий за форму редактирования дисциплин.

Класс «teachersRedactor» — класс, отвечающий за форму редактирования преподавателей

**Заключение.** В процессе разработки были использованы современные технологии программирования и базы данных, что позволило создать высокоэффективную и функциональную систему, способную учитывать множество факторов при составлении расписания занятий. Она позволяет автоматически распределять занятия по времени, аудиториям и преподавателям, учитывая их занятость и предпочтения. Также система имеет возможность генерировать отчеты о составленном расписании и анализировать его эффективность.

В итоге, можно заключить, что проектирование автоматизированной системы для составления расписания занятий является актуальной и перспективной темой для исследований. Результаты могут быть использованы для создания новых систем и улучшения уже существующих.

#### Список цитируемых источников

1. Родионов, А. А. Современные информационные технологии в педагогическом образовании : учеб. пособие / А. А. Родионов. — Пенза : ПГУ, 2019. — 164 с.

2. Автоматизация процессов [Электронный ресурс]. — Режим доступа: [http://www.kpms.ru/Automatization/Process\\_automation.htm/](http://www.kpms.ru/Automatization/Process_automation.htm/). — Дата доступа: 05.04.2023.

УДК 004.4+651

Я. А. Поддевалина

Белорусский государственный университет, Минск, Республика Беларусь

## КРИТЕРИИ ФОРМИРОВАНИЯ ТРЕБОВАНИЙ К КАЧЕСТВУ СИСТЕМ ЭЛЕКТРОННОГО ДОКУМЕНТООБОРОТА

**Введение.** От качества сформированных требований к системам электронного документооборота зависит и само качество указанного вида систем. В свою очередь, качество выявленных требований к системам электронного документооборота (далее — СЭД), а, следовательно, и успех функционирования системы в организации, зависит от того, насколько правильно голос пользователя будет услышан разработчиками. Поэтому становится актуальным вопрос, какие инструменты для формирования требований к качеству СЭД необходимо применять и каковы критерии качественных требований.

**Основная часть.** Для того, чтобы грамотно сформировать требования к качеству СЭД, можно воспользоваться известными и широко применяемыми инструментами работы с требованиями к программному обеспечению (далее — ПО):

1. Методология SMART.

2. INVEST — принцип создания пользовательских историй.

3. Критерии качества требований к ПО, предложенные К Вигерсом.

4. Характеристики качества требований к ПО в соответствии с BABOK (Business Analysis Body of Knowledge — Свод знаний по бизнес-анализу).

Несмотря на то, что методология SMART первоначально использовалась для постановки эффективных целей в менеджменте [1], ее основы применимы и для формирования требований к качеству СЭД. SMART — это акроним, который включает следующие характеристики:

S (specific) — требование должно быть конкретным, учитывающим специфику и имеющим отношение к деятельности организации, внедряющей СЭД в свою ИТ-инфраструктуру.

M (measurable) — требование должно быть измеримым (числовой показатель, отражающий количество в абсолютном или процентном виде).

A (attainable) — требование должно быть достижимым (его можно реализовать с учетом внутренних и внешних ресурсов).

R (relevant) — требование должно быть актуальным (желаемый итог его реализации согласован с ресурсами организации и не нарушает баланса в ее деятельности).

T (time-bound) — требование должно быть ограниченным во времени, отслеживаемым (для этого необходимо определить четкие временные рамки его реализации).

Однако, данная методология больше подходит для формулирования бизнес-требований (высокоуровневые бизнес-цели организации), которые также могут отражать в своем содержании аспекты качества СЭД. Например: «Увеличить производительность системы электронного документооборота на 10 % за 3 месяца».

С возникновением гибких методологий разработки (agile), основанных на итеративном подходе, начали активно использоваться пользовательские истории — краткие формулировки функции системы, описывающие то, что система должна сделать для пользователя [2].

Принцип INVEST применяется для создания пользовательских историй [3], которые можно использовать на этапе сбора и анализа требований к СЭД (в том числе и требований к качеству СЭД). Это также акроним, каждая буква которого раскрывает отдельный критерий качественной пользовательской истории:

I (independent) — пользовательская история может быть реализована и протестирована независимо от других пользовательских историй, а реализуемая часть функционала системы удовлетворяет некоторую потребность.

N (negotiable) — содержание пользовательской истории может быть предметом обсуждения, так как пользовательская история — это общее краткое описание желаемых функций системы, конкретные детали выясняются на этапе коммуникации.

V (valuable) — пользовательская история должна быть ценной. Это наиболее важное свойство, так как главная цель проектирования и разработки СЭД — удовлетворение потребностей (или решение некоторых проблем) организации, для которой разрабатывается система.

E (estimate) — возможность оценки предварительных трудозатрат на реализацию пользовательской истории (в рамках итерации — одного цикла работ). Если пользовательскую историю оценить невозможно, то такая пользовательская история является слишком большой или неопределенной.

S (small) — пользовательская история должна быть компактной и реализуемой в рамках одной итерации. В противном случае она не будет представлять ценности по окончании итерации.

T (testable) — пользовательская история может быть протестирована, чтобы подтвердить ее завершенность.

Примером пользовательской истории, отражающей требование к защищенности СЭД, является следующая формулировка: «Как пользователь я могу подписывать документы электронной цифровой подписью, чтобы придать им юридическую силу».

Карл Вигерс предложил следующие параметры качественных требований [4, с. 238—240], которые также можно применить к требованиям к качеству СЭД:

1. Полнота — требование должно содержать всю информацию, которая необходима для того, чтобы понять его.

2. Корректность — требование должно точно описывать возможность, которая будет удовлетворять конкретную потребность заинтересованного лица и четко определять функциональность, которую необходимо реализовать.

3. Осуществимость — необходимая возможность реализовать требование при известных ограничениях как самой системы, так и рабочей среды (организации), а также в рамках ограниченности ресурсов (временных, ресурсных, бюджетных и т.д.).

4. Необходимость — требование должно отражать возможность, которая действительно предоставит ожидаемую ценность всем заинтересованным лицам.

5. Назначение приоритетов — определение приоритетов требований на основании важности для получения требуемой пользы с учетом сложности реализации, затрат, рисков и т.д.

6. Недвусмысленность — четкая формулировка, исключающая возможность различной интерпретации одного и того же требования.

7. Проверимость — возможность тестирования требования.

В Своде знаний по бизнес-анализу (Business Analysis Body of Knowledge, BABOK), разработанном Международным институтом бизнес-анализа, также перечислены характеристики качества требований [5, с. 143]. По BABOK качественные требования должны быть:

1. Атомарными (atomic) — автономными и способными быть понятными независимо от других требований.

2. Полными (complete) — достаточными для реализации и детализированными на соответствующем уровне для того, чтобы продолжить работу.

3. Последовательными (consistent) — согласованными с выявленными потребностями заинтересованных лиц и непротиворечащими другим требованиям.

4. Лаконичными (concise) — не содержащими в себе посторонней и ненужной информации.

5. Выполнимыми (feasible) — реализуемыми в рамках допустимого риска, графика и бюджета.

6. Недвусмысленными (unambiguous) — четко сформулированными таким образом, чтобы было ясно, соответствует ли решение связанной с ним потребности или нет.

7. Тестируемыми (testable) — проверяемыми, что отдельные требования (или разработка полностью) были выполнены.

8. Приоритетными (prioritized) — ранжированными, сгруппированными с точки зрения важности и ценности по сравнению с другими требованиями.

9. Понятными (understandable) — представленными с использованием общепринятой терминологии.

**Заключение.** Таким образом, все критерии качества требований можно представить в виде сводной таблицы (таблица 1). Критерии по SMART выделены цветом, так как данная методология формирования требований больше подходит для высокоуровневых требований, а функциональные и нефункциональные требования сформировать с помощью SMART будет проблематично, потому что эта методология больше подходит для формулирования бизнес-требований.

Т а б л и ц а 1 — Сводная таблица критериев качества требований

BAВOK	INVEST	К. Вигерс	SMART
Атомарное	Независимое	—	—
Полное	—	Полное	—
Лаконичное	Компактное	—	—
Выполнимое	—	Выполнимое	Выполнимое
Недвусмысленное	—	Недвусмысленное	—
Тестируемое	Тестируемое	Тестируемое	—
Приоритетное	—	Приоритетное	Актуальное
—	Поддающееся оценке	—	Измеримое
—	—	Корректное	Конкретное
—	Ценное	Необходимое	—
Последовательное	—	—	—
Понятное	—	—	—
—	Обсуждаемое	—	—
—	—	—	Ограниченное во времени

Данные критерии применимы и к требованиям к качеству систем электронного документооборота (так как они являются важной составляющей требований к СЭД в целом) и позволяют управлять требованиями в изменяющихся условиях.

#### Список цитируемых источников

1. *Doran George, T.* There's a S.M.A.R.T. way to write management's goals and objectives [Electronic resource]. — Режим доступа: <https://community.mis.temple.edu/mis0855002fall2015/files/2015/10/S.M.A.R.T-Way-Management-Review.pdf>. — Дата доступа: 09.04.2023.
2. *Паттон, Дж.* Пользовательские истории. Искусство гибкой разработки ПО / Дж. Паттон. — СПб. : Питер, 2017.
3. *Wake, B.* INVEST in Good Stories, and SMART Tasks / [Electronic resource]. — Mode of access: <https://xp123.com/articles/invest-in-good-stories-and-smart-tasks/>. — Date of access: 09.04.2023.
4. *Вигерс, К.* Разработка требований к программному обеспечению / К. Вигерс, Дж. Битти. — СПб. : БХВ, 2021.
5. A Guide to the Business Analysis Body of Knowledge [Electronic resource]. — Mode of access: <https://suitecloud.vn/wp-content/uploads/2020/10/BAВOK-Guide-v3-Member.pdf/>. — Date of access: 09.04.2023.

УДК 004.378+651

Е. Э. Попова, О. Л. Липницкая

Белорусский государственный университет, Минск, Республика Беларусь

## ФОРМИРОВАНИЕ КОМПЕТЕНЦИЙ СПЕЦИАЛИСТА ПО ИНФОРМАЦИОННОМУ ОБЕСПЕЧЕНИЮ УПРАВЛЕНИЯ: РАЗРАБОТКА ЦЕЛЕЙ ДИСЦИПЛИНЫ НА ОСНОВЕ ТАКСОНОМИИ БЛУМА

**Введение.** Развитие производства и технологий, внедрение автоматизированных информационных систем невозможно без специалистов, обладающих необходимыми и соответствующими современным условиям деятельности компетенциями.

Условия, в которых приходится работать специалисту — условия цифровой трансформации, характеризующиеся высоким уровнем применения цифровых технологий в производстве, сфере услуг, науке, образовании.

Цифровизация создает предпосылки для кардинального обновления как содержательно-целевых, так и технологических сторон обучения будущих специалистов. Очевидно, что это приводит к необходимости (и уже