

ПРОБЛЕМЫ ПОСТРОЕНИЯ ФУНКЦИОНАЛЬНЫХ ИНТЕРФЕЙСОВ ИНФОРМАЦИОННЫХ СИСТЕМ

Антоненко И.В., Мальцев Д.О.

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

Научный руководитель: Алексеев В.Ф. – канд.техн.наук, доцент, доцент кафедры ПИКС

Аннотация. Показано, что разработка интерфейсов программных средств, удовлетворяющих требованиям пользователей, является одной из важнейших задач при создании программного обеспечения. Отмечено, что общей тенденцией является усложнение пользовательских интерфейсов, связанное как с увеличением функциональности программ, так и с различными и часто изменяющимися условиями их эксплуатации. По этой причине важно учесть проблемы, которые связаны с разработкой интерфейсов.

Ключевые слова: программное средство, информационная система, интерфейс, идентификация интерфейсов, проблемы разработки интерфейсов

Введение. Разработка интерфейсов программных средств, удовлетворяющих требованиям пользователей, является одной из важнейших задач при создании программного обеспечения. Общей тенденцией является усложнение пользовательских интерфейсов, связанное как с увеличением функциональности программ, так и с различными и часто изменяющимися условиями их эксплуатации.

Одна из проблем при проектировании информационной системы (ИС) связана с разработкой интерфейса. По этой причине важным требованием для каждой ИС, которую мы проектируем, является требование(я) к интерфейсу. Вместе с тем, как писать требования к интерфейсу, почти не освещается в литературе, а то, что есть в литературе, не соответствует друг другу. По этой причине целесообразным является рассмотрение методов проектирования интерфейсов: понимание того, что представляет собой интерфейс, как идентифицировать интерфейсы, как определять и документировать определения интерфейсов, что составляет требования к хорошему интерфейсу, где и как документировать требования к интерфейсу [1–6].

Основная часть. Информационные системы (ИС) являются частью других систем, состоят из подсистем и взаимодействуют с другими системами независимо от того на каком уровне находится интересующая нас система. Это применяется как к простым, так и к сложным системам, аппаратным и программным системам, гибридным аппаратным(программным) комплексы. Эти взаимодействия между нашей системой и другими ИС являются интерфейсами.

Идентификация интерфейсов помогает определить границы системы. Также она помогает понять зависимость проектируемой системы от других систем. Неспособность идентифицировать интерфейс может иметь неприятные последствия.

Идентификация интерфейсов помогает обеспечить совместимость между проектируемой системой и другими системами, с которыми нам необходимо взаимодействовать. Многие проекты пренебрегают идентификацией и контролем своих интерфейсов до тестирования.

Первое знакомство с результатами такой оплошности часто происходит, когда тестировщики узнают, что не могут подключить тестовое оборудование к своей системе для выполнения тестов. Более критическая ситуация происходит, когда мы приводим свою систему в эксплуатацию и обнаруживаем отсутствующий интерфейс, из-за которого наша система не может работать или другая система, зависящая от разработанной нами системы, не может работать. Заблаговременно определяя внешние интерфейсы и управляя ими, мы определяем ключевые драйверы для своего продукта, которые необходимо учитывать в системных требованиях.

Определение интерфейсов также помогает выявить потенциальные проблемные области и риски для нашего проекта. На практике, как правило, могут быть системы, с которыми нам нужно взаимодействовать, которые в настоящее время не существуют, но которые разрабатываются параллельно с нашей системой.

Возникает проблема как можно разработать требования для своей ИС, если мы не знаем, какие у нас интерфейсы или характеристики этих интерфейсов? По этой причине очень важно заранее знать о любых проблемах, связанных с нашими интерфейсами, чтобы можно было обеспечить совместимость с существующими системами или работать с другой разрабатываемой системой, чтобы совместно определить интерфейсы, которые были бы совместимы.

Серьезные проблемы могут возникать и возникают на интерфейсах из-за неотъемлемых рисков, связанных с интерфейсами системы. Поскольку интерфейсы представляют системы, находящиеся вне нашего контроля, то разрабатываемая система уязвима на наших интерфейсах. Если интерфейс не совсем понятен, не определен или подлежит изменению, это повлияет на проектируемую ИС. Существует также проблема от того, что кто-то за пределами нашей системы влияет на производительность разрабатываемой ИС – намеренно или непреднамеренно. Есть старая поговорка: «Если вы хотите саботировать чью-то систему, делайте это через интерфейс».

Из-за важности выявления, определения, разработки требований к интерфейсу и управления этими действиями интерфейсы должны быть главной задачей системного инженера проекта, ведущего инженера-программиста, бизнес-аналитика или любого другого лица, участвующего в разработке требований.

Учитывая важность интерфейсов, можно предположить, что существует стандартный процесс идентификации и определения интерфейсов, разработки требований к интерфейсу и управления этими действиями. К сожалению, такого на практике нет. Учитывая разные культуры внутри отраслей и внутри организаций, каждая из них управляет этими видами деятельности по-разному. Это приводит к путанице в отношении того, где документировать эту информацию и даже как называть эти документы.

Разработка пользовательского интерфейса включает те же основные этапы, что и разработка программного обеспечения:

1. Определение типа интерфейса и общих требований к нему.
2. Определение сценариев использования.
3. Определение пользовательской модели интерфейса.
4. Программирование и тестирование программных интерфейсов.

Первым этапом в разработке пользовательского интерфейса является прототипирование, которое выполняется во время сбора требований к системе. При необходимости для каждого отдельного процесса системы создается частичный прототип, включающий: экранную форму, диалог или отчет. Затем определяются требования разграничения доступа к данным.

После детального рассмотрения процессов определяется количество функциональных элементов разрабатываемой системы. Это позволяет разделить информационную систему на ряд подсистем, каждая из которых реализуется одной командой разработчиков.

Пользователи часто судят о качестве системы в целом, исходя из качества ее интерфейса. Более того, от качества интерфейса зависит эффективность использования системы.

В [3] представлено проектирование опыта взаимодействия в виде пяти уровней:

- стратегии (или идеи);
- набора возможностей;
- структуры;
- компоновки;
- поверхности (или внешнего вида интерфейса).

Эти пять уровней составляют концептуальную основу для обсуждения связанных с опытом взаимодействия проблем и средств их решения. Пять наших слоев делятся на две части (рисунок 1) [5].



Рисунок 1 – Элементы проектирования опыта взаимодействия

Слева – все, что касается использования *Web* как интерфейса программы. Справа – все, что связано с информационной структурой. Каждый уровень зависит от уровней, расположенных ниже, и диапазон выбора решений ограничен решениями, принятыми на нижних уровнях.

На каждом из уровней размещаются элементы проектирования.

Автором в [6] сказано, что пользователи могут быть не в состоянии помочь в технических вопросах планирования продукта. В качестве членов проектной бригады пользователи могут оказать поддержку в рассмотрении технических аспектов задач разработки продукта. Истинный сторонник интересов пользователей может работать с целью привлечь к участию в проекте подходящих людей в подходящее время.

Пользователи могут и не знать, что такое ориентированная на пользователя разработка продукта, или иметь слабое представление о различиях в методах и поставляемых компонентах. Однако другие участники бригады обязаны ответить на вопросы пользователей, объяснить цели их участия в проекте, привести некоторые примеры.

Основываясь на этих разъяснениях, пользователи, участвующие в проекте, могут подать идеи о способах своего участия —включая методы, которые до этого не рассматривались техническими участниками бригады. Для достижения большей эффективности участие

пользователей должно смещаться от пассивных видов деятельности к активному партнерству на всех этапах разработки продукта.

Многие пользователи —эксперты в своей области (например, в управлении ведением бизнеса, помощи клиентам и зарабатывании денег в поддержку деятельности по развитию). Их глубокое знание деловых задач и способов использования компьютеров и ПО играют крайне важную роль.

Опытные разработчики продукта могут быть удивлены и смущены рассуждениями пользователей и деловых людей по поводу требований и продуктов, а также по поводу использования этих продуктов.

Заключение. Независимо от названий, которые мы даем различным документам, содержащим информацию об интерфейсах, существуют некоторые руководящие принципы и рекомендации, которым можно следовать. Эти передовые методы основаны на уроках, извлеченных в результате знакомства с различными подходами к управлению требованиями к интерфейсу, а также изучения того, какие подходы работают лучше всего, а какие, как правило, приводят к проблемам.

Целесообразно выполнять документирование требований к интерфейсу в *SRD* каждой системы вместе со всеми другими системными требованиями. С точки зрения инструмента управления требованиями каждое из требований будет иметь ссылку друг на друга, показывая зависимость, а также каждый имеет ссылку на общий определение. Они также оба будут иметь ссылку на общего родителя.

Список литературы

1. Вигерс, К. Разработка требований к программному обеспечению: Пер. с англ. / К. Вигерс. – Москва : Издательско-торговый дом "Русская Редакция", 2004. – 576 с.
2. Купер, А. Об интерфейсе. Основы проектирования взаимодействия / А. Купер, Р. Рейман, Д. Кронин. – 2. Санкт-Петербург : ООО "Издательство "Символ-плюс", 2016. – 688 с.
3. Гарретт, Дж. Веб-дизайн: Элементы опыта взаимодействия. – Пер. с англ. / Дж. Гарретт. – СПб. : Символ-Плюс, 2008. – 192 с.
4. Скотт, Б. Проектирование веб-интерфейсов. - Пер. с англ. А. Минаевой / Б. Скотт, Т. Нейл. – Санкт-Петербург : Символ-Плюс, 2008. – 352 с.
5. Брусенцова, Т. П. Проектирование интерфейсов пользователя : пособие для студентов специальности 1-47 01 02 «Дизайн электронных и веб-изданий» / Т. П. Брусенцова, Т. В. Кишкурно. – Минск : БГТУ, 2019. – 172 с.
6. Торрес, Роберт, Дж. Практическое руководство по проектированию и разработке пользовательского интерфейса.: Пер. с англ. — М.: Издательский дом "Вильямс", 2002. — 400 с.:

UDC 004.5

PROBLEMS OF CONSTRUCTION OF FUNCTIONAL INTERFACES OF INFORMATION SYSTEMS

Antonenko I.V., Maltsev D.O.

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

Alexeev V.F. – PhD, assistant professor, associate professor of the department of ICSD

Annotation. It is shown that the development of software interfaces that meet the requirements of users is one of the most important tasks in software development. It is noted that the general trend is the complication of user interfaces, associated both with an increase in the functionality of programs, and with various and often changing conditions for their operation. For this reason, it is important to consider the challenges that come with designing interfaces.

Keywords: software tool, information system, interface, interface identification, interface development problems.