

**Кравцова Екатерина Юрьевна**, аспирант, 1 курс, СТАНКИН – Московский государственный технологический университет (МГТУ «СТАНКИН»), 127055, Россия, г. Москва, Вадковский переулок, д. 3-А, Институт информационных технологий, Россия, г. Москва

**ПОДХОД К ФОРМИРОВАНИЮ АДАПТИВНОГО  
ПОЛЬЗОВАТЕЛЬСКОГО ИНТЕРФЕЙСА НА ОСНОВЕ АНАЛИЗА  
ПОЛЬЗОВАТЕЛЬСКИХ СЦЕНАРИЕВ И ИНТЕЛЛЕКТУАЛЬНОЙ  
ОБРАТНОЙ СВЯЗИ**

**Аннотация**

В статье рассматривается подход к построению адаптивного интерфейса веб-приложения, основанный на сочетании анализа пользовательских сценариев и интеллектуальной обработки обратной связи. Актуальность темы связана с ростом сложности цифровых сервисов, в которых большое число функций, ролей и контекстов взаимодействия затрудняет выполнение пользовательских задач. В качестве теоретической основы использованы два направления: снижение сложности интерфейса через анализ сценариев использования и разработка адаптивных интерфейсов с применением контекстно-зависимых рекомендательных механизмов. Предложена модель, включающая четыре взаимосвязанных блока: выявление пользовательских сценариев, классификацию элементов интерфейса, контекстную интеллектуальную адаптацию и оценку результата на основе обратной связи. В рамках модели пользовательское поведение рассматривается не только как источник требований на этапе проектирования, но и как постоянный поток данных для уточнения интерфейсной структуры. Такой подход позволяет

связать статическую настройку интерфейса с динамической адаптацией веб-приложения в процессе эксплуатации.

### **Annotation**

The article presents an approach to building an adaptive web application interface based on a combination of user scenario analysis and intelligent feedback processing. The relevance of the topic is related to the growing complexity of digital services, in which a large number of functions, roles, and interaction contexts complicate user task completion. Two theoretical directions are employed as the foundation: reducing interface complexity through usage scenario analysis and developing adaptive interfaces using context-dependent recommendation mechanisms. A model is proposed that includes four interrelated components: identification of user scenarios, classification of interface elements, context-aware intelligent adaptation, and result evaluation based on feedback. Within the framework of this model, user behavior is considered not only as a source of requirements at the design stage, but also as a continuous data stream for refining interface structure. This approach makes it possible to link static interface configuration with dynamic adaptation of the web application during operation.

Ключевые слова: адаптивный пользовательский интерфейс, сценарии использования, пользовательское взаимодействие, сложность интерфейса, рекомендательный механизм.

Keywords: adaptive user interface, use cases, user interaction, interface complexity, recommendation engine.

### **Введение**

Современные веб-приложения становятся более функциональными, но вместе с расширением возможностей возрастает сложность пользовательского взаимодействия. На одном экране могут размещаться элементы поиска, фильтрации, уведомлений, рекомендаций, настроек и аналитики. При отсутствии адаптации интерфейс начинает требовать от пользователя лишних действий, повышает нагрузку на внимание и увеличивает вероятность ошибок.

Особенно заметна данная проблема в системах, где разные группы пользователей работают с одним приложением, но решают неодинаковые задачи. Универсальная компоновка интерфейса не всегда подходит для таких условий. Одним пользователям требуется быстрый доступ к поиску и справочной информации, другим необходимы функции анализа, настройки или подтверждения операций. При одинаковом наборе экранных объектов часть элементов становится избыточной для конкретного сценария.

Адаптивный интерфейс позволяет изменять представление функций с учетом роли пользователя, выполняемой им задачи, поведения в системе и условий взаимодействия. Такая логика предполагает не статичное размещение элементов, а гибкую настройку структуры UI под конкретный пользовательский сценарий. Снижение сложности интерфейса может достигаться за счет анализа сценариев использования, выделения значимых объектов и удаления лишних элементов. При этом адаптация может опираться не только на предварительно заданные требования, но и на данные о пользовательских действиях, контекстные параметры и работу рекомендательного механизма.

Сочетание указанных подходов формирует основу для модели адаптивного интерфейса веб-приложения. Анализ сценариев отвечает за первоначальное понимание задач пользователя, а интеллектуальная обратная связь обеспечивает последующее уточнение интерфейса в ходе эксплуатации. Такая связка важна для веб-приложений, в которых пользовательский контекст меняется быстрее, чем проектные решения, принятые на этапе разработки.

## **1. Теоретическая основа адаптивного интерфейса**

Адаптивный пользовательский интерфейс отличается от статического интерфейса тем, что его структура изменчива в зависимости от условий взаимодействия. Изменению могут подлежать состав элементов, порядок их отображения, размер, группировка, приоритетность функций и способы подачи подсказок. Главная задача адаптации состоит не в простом изменении

внешнего вида, а в повышении соответствия интерфейса пользовательской задаче.

Результат в виде проектирования объектно-ориентированного пользовательского интерфейса и анализ сценариев использования, связывается с адаптацией пользовательского интерфейса. Object-Oriented User Interface (OOUI) представляет собой подход к проектированию интерфейса, при котором основой взаимодействия выступают объекты предметной области, а пользовательские действия организуются вокруг этих объектов. В отличие от функционально-ориентированного интерфейса, где взаимодействие начинается с выбора операции, OOUI предполагает первичное представление сущностей, с которыми пользователь работает в системе, поскольку позволяет связывать поведение интерфейса с характеристиками конкретных объектов, пользовательскими сценариями и контекстом взаимодействия. Такой подход позволяет последовательно выделять цели пользователей, их роли и основные действия в системе, а затем переводить полученные сведения в структуру адаптивного UI (AUI). Для этого могут применяться интервью, анализ пользовательских задач и классификация элементов интерфейса по степени их значимости для выполнения конкретного сценария. Благодаря AUI идет снижение сложности пользовательского интерфейса, связанного с визуальной перегруженностью, раздробленным расположением элементов и избыточным количеством объектов в рабочей области.

Дополнительным направлением развития AUI выступает контекстная адаптация. В рамках такого подхода пользовательские взаимодействия рассматриваются как основной источник данных для изменения интерфейса. Система может учитывать историю действий, текущую задачу, параметры среды, устройство и пользовательские предпочтения. На основе этих данных формируется интерфейсная конфигурация, которая лучше соответствует конкретной ситуации использования. Для реализации контекстной адаптации могут применяться формализованные модели предметной области. Данные о поведении пользователя и рекомендательные механизмы позволяют

определить, какие функции следует вывести на первый план, какие элементы можно скрыть, какие действия стоит предложить пользователю и как изменить структуру интерфейса без нарушения логики взаимодействия.

## **2. Проблема адаптации веб-интерфейса**

Для веб-приложений характерна проблема перегруженности интерфейса. Система должна поддерживать разные пользовательские задачи, поэтому в ней появляется большое количество функций, разделов и элементов управления. При этом избыток объектов на экране усложняет восприятие и увеличивает время выполнения действий. Поэтому при проектировании адаптивного UI важно сохранить нужную функциональность, но не перегружать пользователя лишними элементами.

Традиционный подход к проектированию часто фиксирует структуру интерфейса на этапе разработки. После запуска веб-приложения пользовательская активность меняется: часть функций используется регулярно, часть теряет значимость, а новые сценарии появляются по мере накопления опыта. При отсутствии механизма обратной связи интерфейс не отражает реальные изменения в поведении пользователей. Проблема усиливается при наличии интеллектуальных функций. Рекомендации, персонализация, подсказки могут повысить удобство, но при непрозрачной реализации вызывают недоверие. Пользователь должен понимать, почему интерфейс предлагает определенное действие, какие данные учитываются и как можно изменить результат адаптации. В следствии чего адаптивный интерфейс веб-приложения должен решать две задачи одновременно. Первая задача связана с уменьшением визуальной и функциональной сложности. Вторая задача связана с поддержкой динамической настройки на основе пользовательского поведения и контекстных данных. Для решения данных задач требуется модель, объединяющая сценарный анализ и интеллектуальную обратную связь.

## **3. Предлагаемая модель адаптивного интерфейса**

Предлагаемая модель рассматривает адаптивный интерфейс веб-приложения как систему, состоящую из четырех взаимосвязанных блоков: анализа пользовательских сценариев, классификации элементов интерфейса, интеллектуальной контекстной адаптации и оценки результата по обратной связи. Каждый блок выполняет отдельную функцию, но итоговая адаптация возникает только при их совместной работе.

Таблица 1 – Структура модели адаптивного пользовательского интерфейса веб-приложения

Блок модели	Назначение	Основной результат
Анализ пользовательских сценариев	Выявление ролей, задач, условий и последовательности действий	Описание типовых сценариев и пользовательских целей
Классификация элементов UI	Разделение объектов интерфейса по степени значимости для задачи	Набор обязательных, контекстных и избыточных элементов
Интеллектуальная контекстная адаптация	Обработка поведения, предпочтений и условий работы пользователя	Изменение состава, порядка и приоритета элементов
Оценка по обратной связи	Проверка влияния адаптации на сложность и удобство взаимодействия	Уточнение правил адаптации и параметров интерфейса

Первый блок ориентирован на выявление пользовательских сценариев. На данном этапе фиксируются роли пользователей, основные цели, последовательность действий, частые ошибки и условия выполнения задач. В отличие от общего описания требований, сценарный анализ позволяет связать каждую функцию интерфейса с конкретной пользовательской ситуацией. Данный принцип согласуется с UCA (Use Case Analysis), где задача пользователя рассматривается через взаимодействие пользователя и системы.

Второй блок связан с классификацией элементов интерфейса. Объекты UI могут быть разделены на обязательные, контекстные и избыточные. Обязательные элементы необходимы для выполнения основной задачи.

Контекстные элементы нужны только при определенных условиях, например при смене роли, устройства, стадии работы или уровня опыта пользователя. Избыточные элементы не поддерживают текущий сценарий и могут быть скрыты, перенесены во вторичный уровень либо удалены из основной области экрана.

Третий блок обеспечивает интеллектуальную адаптацию. Для данного блока могут использоваться данные о частоте действий, времени выполнения задач, ошибках, возвратах на предыдущие шаги, выбранных фильтрах, отказах от предложений, уровне завершенности сценария и параметрах устройства. На основе таких данных формируется контекстная модель пользователя. Рекомендательный механизм определяет, какие элементы следует показать, скрыть, сгруппировать или повысить в визуальном приоритете.

Четвертый блок отвечает за оценку результата. Обратная связь может быть явной и неявной. Явная обратная связь включает оценки пользователя, ответы на короткие вопросы, выбор причины отказа от рекомендации. Неявная обратная связь выражается через поведенческие показатели: повторные клики, длительные паузы, частое использование поиска, отмену действия, возврат к старой компоновке. Система должна учитывать оба типа данных, поскольку пользователь не всегда готов давать прямую оценку интерфейсу. Интеллектуальная обратная связь в предлагаемой модели понимается как постоянный контур сбора, анализа и применения пользовательских данных для уточнения интерфейса. Отличается от обычной формы отзыва тем, что используется не только для последующего анализа разработчиками, но и для оперативного изменения поведения UI. Для предотвращения избыточной автоматизации необходимо сохранять управляемость адаптации. Пользователь должен иметь возможность отменить изменение, вернуть прежний вариант интерфейса или ограничить персонализацию, что снижает риск недоверия к системе и поддерживает ощущение контроля.

## **Заключение**

Предложенная модель объединяет сильные стороны двух подходов. Анализ сценариев использования помогает построить первоначальную структуру интерфейса на основе пользовательских задач. Контекстно-зависимая интеллектуальная адаптация позволяет корректировать интерфейс после запуска веб-приложения, когда появляются реальные данные о поведении пользователей.

Связь между двумя подходами особенно важна для приложений, где один и тот же интерфейс используется пользователями с разным опытом. Начинающему пользователю могут быть нужны подсказки, пошаговая навигация и видимые элементы помощи. Опытному пользователю важнее быстрый доступ к часто используемым функциям, сокращения и минимальное число промежуточных экранов. Статический интерфейс вынужден искать компромисс, тогда как адаптивная модель позволяет менять представление функций в зависимости от накопленных данных.

Практическая ценность модели состоит в возможности использовать ее как основу для проектирования веб-приложений с персонализированным интерфейсом. Сценарный блок снижает риск хаотичной адаптации, поскольку каждое изменение связывается с пользовательской задачей. Интеллектуальный блок расширяет модель за счет учета контекста и обратной связи. Оценочный блок позволяет проверять, приводит ли адаптация к снижению сложности, а не к появлению новых трудностей.

### **Литература**

1. Муравейко, Е. С. Искусственный интеллект как инструмент в разработке адаптивных и интерактивных веб-приложений / Е. С. Муравейко, Т. В. Федюкович // Актуальные вопросы экономики и информационных технологий: сборник материалов докладов 61-й научной конференции аспирантов, магистрантов и студентов БГУИР, Минск, 20–25 апреля 2025 г. / Белорусский государственный университет информатики и радиоэлектроники. – 2025. – С. 223–226.

2. Каюков, А. И. Проектирование адаптивных пользовательских интерфейсов / А. И. Каюков, И. А. Галушка, Т. А. Коваленко // Парадигма. – 2025. – № 5. – С. 145–151.
3. Якунин, А. В. Технологии искусственного интеллекта как фактор эволюции веб-интерфейса: потенциал применения / А. В. Якунин // Динамика медиасистем. – 2026. – Т. 6, № 1. – С. 185–189.
4. Costa, A. Towards an AI-Driven User Interface Design for Web Applications / A. Costa, F. Silva, J. J. Moreira // Procedia Computer Science. – 2024. – Vol. 237.– P. 179–186.
5. Benyon, D. Applying User Modeling to Human-Computer Interaction Design / D. Benyon, D. Murray // Artificial Intelligence Review. – 1993. — Vol. 7, No. 3– 4. – P. 199–225.