



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»

ФАКУЛЬТЕТ Робототехника и комплексная автоматизация

КАФЕДРА Системы автоматизированного проектирования (РК-6)

РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА К НАУЧНО-ИССЛЕДОВАТЕЛЬСКОЙ РАБОТЕ

на тему:
«Обзор известных подходов к разработке GUI»

Студент РК6-73Б
(Группа)


(Подпись, дата)

А.Р. Василян
(И.О.Фамилия)

Руководитель курсовой работы (проекта)

(Подпись, дата)

А.П. Соколов
(И.О.Фамилия)

Москва, 2022 г.

Оглавление

Программный инструментарий для создания подсистем ввода данных при разработке систем инженерного анализа	3
Разработка программного обеспечения генерации кода на основе шаблонов при создании систем инженерного анализа	3
Методы построения GUI.....	4
Метод построения оконного интерфейса пользователя на основе моделирования пользовательских целей	4
Построение пользовательского интерфейса с использованием интерактивного машинного обучения	5
Методический подход к созданию универсального пользовательского интерфейса	5
Список литературы	6

Программный инструментарий для создания подсистем ввода данных при разработке систем инженерного анализа

Была изучена статья “Программный инструментарий для создания подсистем ввода данных при разработке систем инженерного анализа”.

Автоматическое построение GUI — построение графических форм ввода исходных данных для прикладной программы без участия человека с использованием программы-генератора GUI и файла исходных данных в специальном формате; Метаданные — список входных параметров и их текущие значения в рамках выполнения прикладной программой вычислительной задачи.

На рисунке 1 представлена диаграмма потока данных при построении и эксплуатации GUI на основе созданного программного инструментария и формата aINI.



Рис. 1. Диаграмма потока данных при построении и эксплуатации GUI на основе созданного программного инструментария и формата aINI

Разработка программного обеспечения генерации кода на основе шаблонов при создании систем инженерного анализа

В результате анализа статьи “Разработка программного обеспечения генерации кода на основе шаблонов при создании систем инженерного анализа” были изучены особенности использования параметров шаблонов векторного типа, архитектура программного инструментария и выводы:

- представляется целесообразным при разработке программного обеспечения инженерного анализа применение “гибридных” подходов, а именно: для разработки общесистемных стандартных функциональных возможностей следует использовать генераторы кода на основе шаблонов, тогда как при разработке программных реализаций алгоритмически сложных вычислительных процедур следует использовать генерацию кода на основе графовых представлений алгоритмов.

- применение средств поддержки процессов разработки и средств генерации кода позволяет систематизировать процесс разработки multifunctional программных комплексов.
- актуальными и востребованными на практике являются программные механизмы предоставления удалённого доступа к разработанному программному инструментарию и библиотеке шаблонов посредством web-приложения. Решение такой задачи позволит сформировать основу средств автоматизации процессов разработки программного обеспечения, документирования, безбумажного документооборота, а также позволит предоставить доступ к созданному инструментарию широкому кругу пользователей.

Методы построения GUI

Были изучены статьи на тему разработку графического пользовательского интерфейса и были выделены представленные ниже подходы.

Метод построения оконного интерфейса пользователя на основе моделирования пользовательских целей

При изучении методов построения на основе пользовательских целей были выделены два метода: ограничительный и направляющий.

Для средств взаимодействия пользователя и ЭВМ с оконным интерфейсом типичным является ограничительный метод. Метод заключается в том, что пользователю предоставляется некий набор операций, которые имеют название, исходные данные и результаты. Пользователь выбирает нужную операцию и задает для нее исходные данные, после чего ЭВМ выполняет указанную операцию, активируя соответствующие функции приложения, и выдаёт результаты операции пользователю. По итогам выполнения очередной операции пользователь решает, какую следующую операцию ему нужно выбрать, передаёт ее ЭВМ на выполнение и этот процесс продолжается до тех пор, пока в итоге выполнения операций не будет достигнут желаемый результат

Основой направляющего метода является ДТ-модель (ДТ – диалоговая транзакция). Цели участников диалога делятся на практические цели (ПЦ) и коммуникативные цели (КЦ). ПЦ — это описание определенной ситуации в общей предметной области участников, которую один из участников (инициатор ПЦ) стремится достичь с помощью другого участника. КЦ — это намерение участника-инициатора ПЦ довести свою ПЦ до сведения второго участника.

Главное отличие направляющего метода взаимодействия "пользователь-ЭВМ" по сравнению с ограничительным методом состоит в том, что инициатива взаимодействия по достижению цели пользователя принадлежит не пользователю, а ЭВМ. Это означает, что ЭВМ играет активную роль в целенаправленном взаимодействии по выполнению задания пользователя.

Построение пользовательского интерфейса с использованием интерактивного машинного обучения

На первом этапе производится сбор входных данных. В качестве таких данных будут выступать частота, последовательность, достигаемый результат и время между применениями рассматриваемых функций.

На основании собранных данных проводится обучение целью которого является сократить путь для достижения конкретного результата, сокращение количества шагов и затрачиваемого времени для выполнения идентичных задач.

Алгоритм обучения выбирает лучший порог для каждого. Использование матрицы Гессияна и весов позволяет вычислять прирост информации, вызванный применением каждой функции и правила принятия решения для узла.

По результатам обучения строится последовательность действий для достижения необходимого результата. При её построении учитывается время поиска элемента интерфейса, его доступность, соответствие описания и ожидаемого результат. На основании полученных результатов вносятся корректировки в существующий интерфейс, после чего обучение продолжается.

Методический подход к созданию универсального пользовательского интерфейса

На рисунке 2 представлены составные элементы подхода к созданию универсального средства построения пользовательского интерфейса.

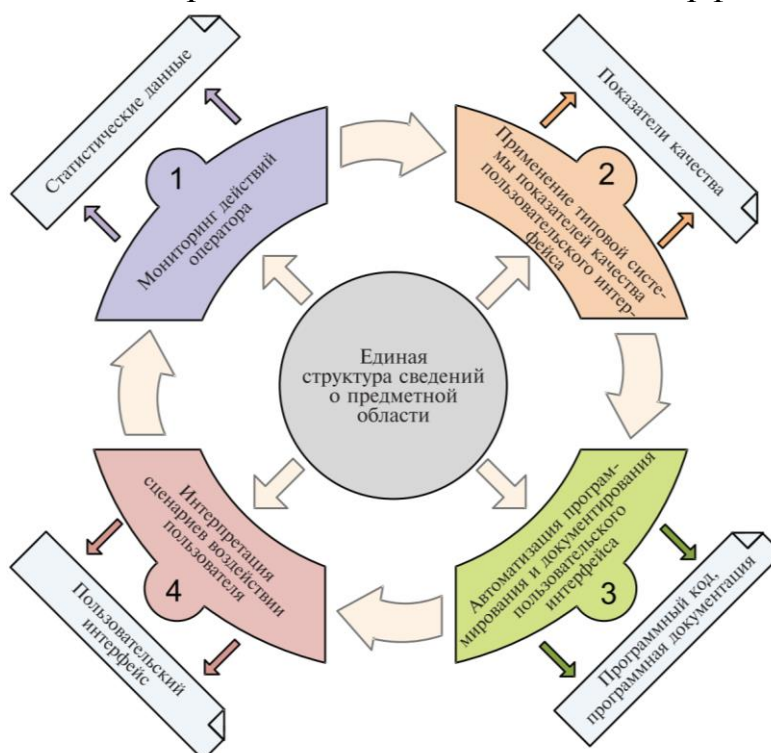


Рис. 2. Составные элементы подхода к созданию универсального средства построения пользовательского интерфейса программных средств

Мониторинг действий оператора (блок 1 на рис. 2) позволяет осуществлять сбор и накопление статистики деятельности оператора во время эксплуатации программных средств.

Оценка качества пользовательского интерфейса обеспечивается за счет:

- применения типовой системы показателей качества (блок 2 на рис. 2);
- разработки методики оценки качества;
- выполнения мероприятий по оценке показателей качества.

“Автоматизация программирования и документирования пользовательского интерфейса” (блок 3 на рис. 2) подразумевает возможность автоматизированного документирования интерфейса программы.

Отображение некоторого абстрактного сценария осуществляет механизм его интерпретации (блок 4 на рис. 2) в стандартные программные процедуры, характерные для выбранной программноаппаратной платформы. Наличие интерпретатора предписывает применение некоторой формальной логики (языка), с помощью лексем которой выражаются любые сценарии. Одна лексема описывает типовую атомарную операцию над элементом управления пользовательского интерфейса, идентифицирующие сведения о которой содержатся в параметрах лексемы.

Список литературы

[1] Крехтунова Д., Ершов В., Муха В., Тришин И., Василян А. Р. Разработка систем инженерного анализа и ресурсоемкого ПО (rndhpc): Научно-исследовательские заметки. / Под редакцией Соколова А.П. [Электронный ресурс] - Москва: 2021. - 85 с. URL: <https://arch.rk6.bmstu.ru> (облачный сервис кафедры РК6)