

**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ  
ФЕДЕРАЦИИ**

**Федеральное государственное бюджетное образовательное учреждение  
высшего образования**

**«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»  
(ФГБОУ ВО «КубГУ»)**

**Факультет компьютерных технологий и прикладной математики**

**Кафедра информационных технологий**

## **ОТЧЕТ**

**о выполнении лабораторной работы № 4**

**по дисциплине «Программная инженерия»**

Выполнила: ст. гр. МО 32/2

Переузник В. С.

Проверил: доцент каф. ИТ

Полетайкин А. Н.

Краснодар

2025

Тема: разработка функциональной структуры программного продукта:  
функционально-ориентированный подход.

Индивидуальная тема: «Программа для аппроксимации функции».

Цель: изучение методики функционально-ориентированного подхода программной инженерии для разработки и описания функциональности разрабатываемого программного обеспечения.

Задание: 1) Построить функциональную модель разрабатываемого ПП в виде контекстной диаграммы в нотации IDEF0 при помощи пакета BPWin; 2) На основе контекстной диаграммы и диаграммы требований UML, построенной при выполнении лабораторной работы 3, построить диаграмму декомпозиции A0 на дочерние подпроцессы (автоматизированные функции) ПП; 3) Опираясь на диаграмму A0 выполнить перечисление автоматизированных функций ПП. Для не простых функций построить диаграммы декомпозиции A2.

Ход выполнения работы:

### ***1. Разработка контекстной диаграммы в нотации IDEF0***

На основании модели «Черный ящик» из лабораторной работы 1 и технического задания из лабораторной работы 3 была разработана контекстная диаграмма ПП «Программа для аппроксимации функции» в нотации IDEF0. Диаграмма представляет систему как единый функциональный блок (черный ящик) с определенными входами, выходами, управлениями и механизмами. Контекстная диаграмма представлена на рисунке 1.

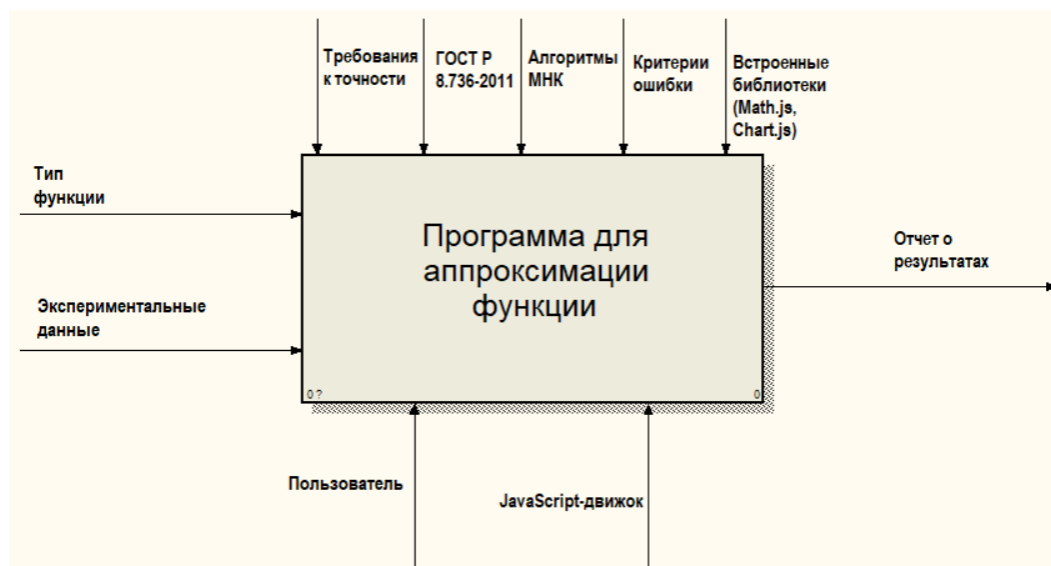


Рис. 1 - Контекстная диаграмма ПП для аппроксимации функции

При построении диаграммы было выполнено переосмысление исходного бизнес-процесса (из лабораторной работы 1) с учетом его автоматизации в рамках ПП. В отличие от исходной модели, где пользователь (исследователь) являлся ключевым механизмом, выполняющим расчеты вручную, в модели ПП механизмами являются программные компоненты, реализующие функциональность: веб-браузер, среда выполнения JS и встроенные библиотеки.

## 2. Диаграмма декомпозиции A0

На основе контекстной диаграммы и диаграммы требований UML из лабораторной работы 3 построена диаграмма декомпозиции A0. Дочерние процессы соответствуют функциональным задачам, определенным в техническом задании. Диаграмма показана на рисунке 2.

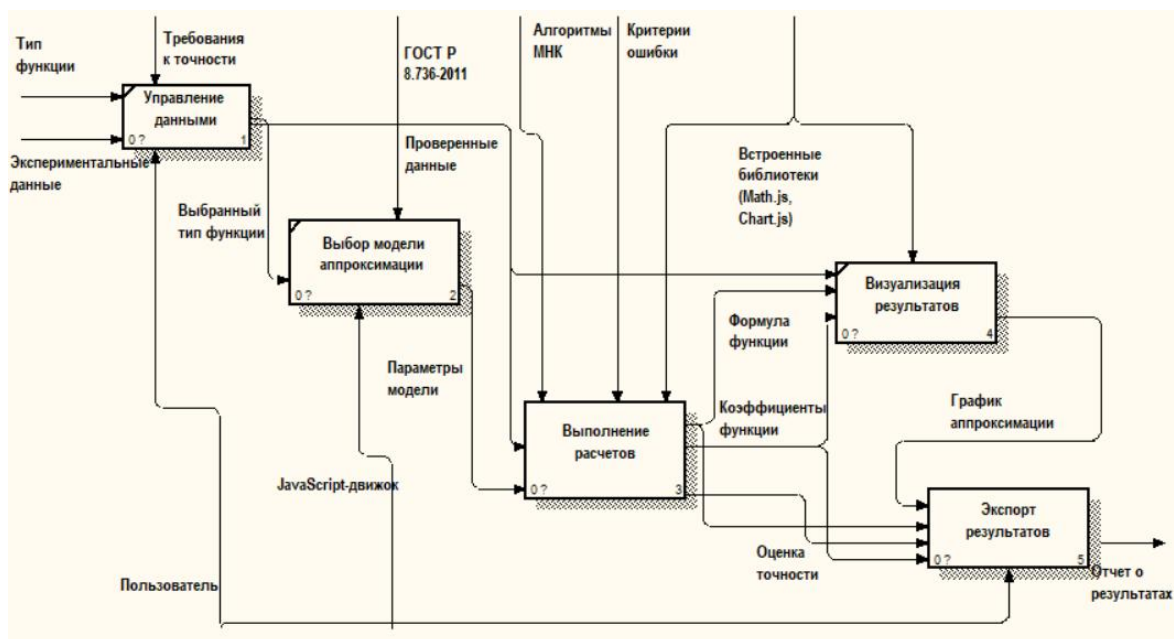


Рис. 2 - Диаграмма декомпозиции A0

### 3. Перечисление автоматизированных функций ПП

#### 1) Управление данными

Функция будет обеспечивать ввод и подготовку исходных экспериментальных данных для последующей обработки в программном продукте. Пользователь сможет вводить массив числовых пар (X,Y) через табличный интерфейс или импортировать данные из CSV-файла. Система будет выполнять валидацию данных, проверяя числовой формат и отсутствие пустых значений. Также пользователь будет выбирать тип аппроксимирующей функции из выпадающего списка. В результате функция будет формировать проверенный массив данных и выбранный тип модели для передачи в вычислительные модули.

#### 2) Выбор модели аппроксимации

Функция будет определять математическую модель для аппроксимации экспериментальных данных. На основе выбранного пользователем типа функции система будет устанавливать соответствующие параметры модели: для линейной функции будут определяться два коэффициента, для квадратичной - три коэффициента, для кубической - четыре коэффициента, для экспоненциальной - два параметра. Выбор модели

будет осуществляться согласно требованиям ГОСТ Р 8.736-2011 к методам обработки результатов измерений.

### 3) Выполнение расчетов

Функция будет реализовывать математическое ядро системы, выполняя аппроксимацию данных методом наименьших квадратов (МНК). На основе проверенных данных и выбранного типа функции система будет формировать и решать систему линейных уравнений для определения коэффициентов аппроксимирующей функции. После расчета коэффициентов система будет вычислять значения метрик точности, включая коэффициент детерминации  $R^2$ . Все вычисления будут выполняться с использованием библиотеки Math.js.

### 4) Визуализация результатов

Функция будет обеспечивать графическое представление результатов аппроксимации. С использованием библиотеки Chart.js будет строиться график, на котором будут отображаться исходные экспериментальные точки и кривая подобранной функции. График будет включать подписи осей координат, легенду для идентификации данных и сетку для удобства чтения значений. Визуализация позволит наглядно оценить качество аппроксимации и соответствие подобранной функции исходным данным.

### 5) Экспорт результатов

Функция будет обеспечивать формирование и представление итогов работы системы в структурированном виде. Будет создаваться отчет, содержащий аналитическое представление аппроксимирующей функции в виде математической формулы, численные значения коэффициентов, рассчитанные метрики точности и графическое представление результатов. Пользователь сможет просматривать отчет непосредственно в интерфейсе программы и копировать отдельные элементы результатов для использования в других приложениях.

## ***3.1. Декомпозиция не простых функций***

На основе диаграммы декомпозиции A0 было выявлено, что не все функции являются простыми. Для функций, включающих в себя несколько последовательных или параллельных операций, была выполнена дальнейшая

детализация на диаграммах уровня А2. В качестве не простой функции была определена «Выполнение расчетов», показанная на Рис. 3.

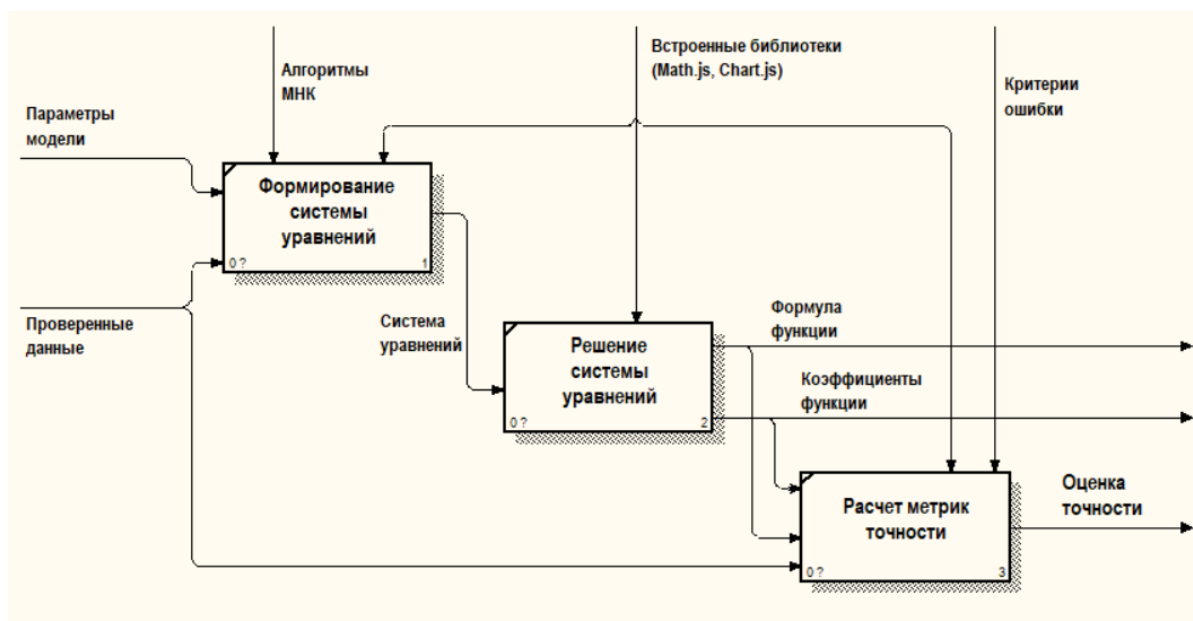


Рис. 3 - Диаграмма декомпозиции А2 для функции "Выполнение расчетов"

Декомпозиция функции «Выполнение расчетов»:

Функция выполняется в три последовательных этапа. На первом этапе система формирует систему нормальных уравнений методом наименьших квадратов на основе проверенных экспериментальных данных и параметров выбранной модели. Используются алгоритмы МНК и библиотека Math.js для математических операций.

На втором этапе система решает полученную систему уравнений численными методами. В результате определяются точные значения коэффициентов аппроксимирующей функции, и формируется ее итоговая математическая формула. Для решения уравнений применяются вычислительные возможности библиотеки Math.js.

На третьем этапе система рассчитывает метрики точности аппроксимации, в частности коэффициент детерминации  $R^2$ . Для этого используются рассчитанные коэффициенты функции и исходные экспериментальные данные. Расчеты выполняются согласно заданному критерию ошибки с использованием математических функций библиотеки Math.js.

Вывод: в ходе выполнения лабораторной работы была успешно разработана функциональная структура программного продукта для аппроксимации функций с использованием функционально-ориентированного подхода. Построена контекстная диаграмма IDEF0, представляющая систему как единый функциональный блок с четко определенными границами. На ее основе создана диаграмма декомпозиции A0, детализирующая пять основных автоматизированных функций. Проведен анализ сложности функций, в результате которого выполнена декомпозиция наиболее сложной функции «Выполнение расчетов» на три математические операции. Разработанная функциональная модель полностью соответствует требованиям технического задания и служит надежной основой для последующей реализации программного продукта, демонстрируя переход от описания ручного бизнес-процесса к проектированию автоматизированной системы. Все элементы модели взаимосвязаны и соответствуют правилам нотации IDEF0.