

Этап 1

«Определение базовых трудозатрат разрабатываемой ИС»

Определяется количества функций.

Функции обработки информации определяются на основе данных технического задания на разработку информационной системы. Функции обработки информации включают две группы - новые и развиваемые.

1. Базовые трудозатраты определяются по каждой из подсистем разрабатываемой информационной системы по формулам (1) и (2):

$$\Phi T = \sum_{i=1}^8 (\Phi_i \times a_{\Phi_i}), \quad (1)$$

$$BT = \Phi T^B \times K_n, \quad (2),$$

где:

ΦT - общее количество функциональных точек разрабатываемой подсистемы информационной системы;

Φ_i - количество функций обработки информации ИС i -го типа;

a_{Φ_i} - значение коэффициента сложности для функций обработки информации i -го типа;

BT - базовые трудозатраты, чел.-месяцев;

K_n - коэффициент преобразования количества функциональных точек в базовые трудозатраты, чел.-месяцев/(точку в степени коэффициента масштаба B);

B - коэффициент масштаба, определяющий снижение удельных трудозатрат на разработку ИС при росте числа функциональных точек (установлен расчетно $B = 0,94$).

2. Функции обработки информации определяются на основе данных технического задания на разработку информационной системы. Функции обработки информации включают две группы - новые и развиваемые. Для новых функций в расчетах учитываются следующие типы функций обработки информации:

2.1. Внутренние логические файлы (Φ_1) - логически связанные группы данных или блоки управляющей информации, которые поддерживаются внутри ИС. К данной группе может быть отнесено в том числе: хранение информации, объединенной до одной логической группы вне зависимости от способа хранения (например, таблицы ролей; таблицы для логирования и журналирования информации; таблицы связей; справочники; данные статусной модели и другие логические группы), а также исполнимые последовательности шагов бизнес-процесса, автоматизируемого и хранимого в системе. Информация для хранения в системе может поступать в рамках информационного взаимодействия с внешними ИС в результате действий пользователя или алгоритмов системы.

2.2. Внешние интерфейсные файлы (Φ_2) - функции получения логически связанных групп данных или блоков управляющей информации из внешних ИС (входящие интеграции), а также функции передачи логически связанных групп данных или блоков управляющей информации во внешние ИС (исходящие интеграции).

2.3. Внешние входные элементы (Φ_3) - процессы передачи логически связанных групп данных, файлов от пользователя. К данной группе могут быть отнесены функции, связанные с внесением изменений пользователем во внутренние файлы ИС (добавление, редактирование) при помощи пользовательского интерфейса с последующим сохранением групп данных в системе. Внесение изменений во внутренние файлы ИС может осуществляться как при помощи пользовательского ввода, так и при помощи импорта файлов.

2.4. Внешние выходные элементы и обработка информации (Φ_4) - процессы выдачи логически связанных групп данных из внутренних файлов ИС пользователю, во внешние

или внутренние файлы с обработкой (изменением, валидацией) этих данных. Данные процессы могут запускаться пользователем или системой. К данной группе могут быть отнесены, например: функции вывода информации с обработкой исходных данных; алгоритмы расчета информации с последующим выводом или сохранением во внутренних файлах ИС; ETL-процессы; функции выгрузки в файл с обработкой исходных данных.

2.5. Внешние запросы (Φ_5) - процессы отображения данных одной логической группы пользователю, в том числе в комбинации ввода-вывода - когда запрос приводит к немедленному программному ответу в форме элементов пользовательского интерфейса и/или системного сообщения (данный диалоговый ввод в программу не сохраняется, не модифицирует внутренние логические файлы). К данной группе могут быть отнесены, например, функции поиска; фильтрации; группировки и сортировки; функции отображения информации, авторизации; функции вывода регламентных отчетов; функции выгрузки в файл, печати; функции уведомлений; функции проверки (контроля), которые не приводят к изменению данных (например, условного форматирования или индикации пороговых значений); а также функции предоставления возможности использования ранее реализованных функций системы (использование реализованных функций в рамках микросервисной архитектуры).

Для развиваемых функций обработки информации в расчетах учитываются следующие типы:

2.6. Развитие функций хранения информации и интеграций (Φ_6) - изменения ранее созданных внутренних логических файлов и внешних интерфейсных файлов (Φ_1 и Φ_2), приводящие к приращению функциональности, но не меняющие ее полностью.

2.7. Развитие функций ввода-вывода информации (Φ_7) - изменение ранее созданных, внешних входных элементов, внешних выходных элементов и внешних запросов (Φ_3 , Φ_4 и Φ_5), приводящие к приращению функциональности, но не меняющие ее полностью.

2.8. Функции повышения удобства использования (Φ_3) - процессы изменения функций обработки информации, направленные на улучшение визуального отображения данных и иные необходимые изменения системы, не приводящие к приращению функциональности, но повышающие удобство использования информационной системы (изменение дизайна экранных форм; изменение названий полей, форм, адаптивная верстка страниц; скрипты проверки работоспособности).

3. При выполнении работ по развитию (модернизации) информационной системы в расчетах учитываются новые и развиваемые функции обработки информации в части создаваемых (модернизируемых) компонентов ИС (подсистем, модулей и т.д.).

4. Значения коэффициентов сложности по функциям обработки информации определяются следующим образом (таблица 1):

Таблица 1

Тип функций обработки информации	Значение коэффициента a_{Φ_i}
Внутренние логические файлы (Φ_1)	11
Внешние интерфейсные файлы (Φ_3)	6
Внешние входные элементы (Φ_3)	5
Внешние выходные элементы и обработка информации (Φ_4)	5
Внешние запросы (Φ_5)	4
Развитие функций хранения информации и интеграций (Φ_6)	4

Развитие функций ввода-вывода информации (Φ_7)	2
Функции повышения удобства использования (Φ_8)	1

5. Коэффициенты преобразования количества функциональных точек в базовые трудозатраты, в зависимости от способа реализации (уровня языка разработки), представлены в таблице 2:

Таблица 2

Способ реализации (уровень языка разработки)	K_n
Сверхвысокий уровень, RPA, Low code (No-code) разработка	
Язык сверхвысокого уровня (в среднем), RPA, Low code (No-code) разработка	0,117
Высокий уровень	
Язык высокого уровня (в среднем) <*>	0,255
Низкий уровень	
Язык низкого уровня (в среднем)	0,468

<*> Также используется в случае, если способ реализации (уровень языка программирования) не определен в техническом задании.

6. Расчеты по формулам (1) и (2) проводятся отдельно по каждой из подсистем разрабатываемой ИС. В случае если архитектурой системы вместо разделения на подсистемы предусмотрено разделение на модули, сервисы либо иные прикладные компоненты, то расчеты по формулам (1) и (2) проводятся, соответственно, по модулям, сервисам, либо иным прикладным компонентам, представляющим верхний уровень в структуре системы.