



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)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ «Информатика и системы
управления»
КАФЕДРА «Программное обеспечение ЭВМ и информационные
технологии»

Лабораторная работа №7
По курсу «Экономика программной инженерии»
Тема: Предварительная оценка параметров
программного проекта

Студенты:	Кондрашова О.П. Коротков А.В. Оберган Т.М.
Группа:	ИУ7-85Б
Преподаватели:	Барышникова М.Ю. Силантьева А.В.

Цель работы: ознакомление с существующими методиками предварительной оценки параметров программного проекта и практическая оценка затрат на примере методики COSOMO II.

Содержание проекта: рассчитать количество функциональных точек для разрабатываемого программного приложения. С этой целью разработать программный инструмент. Произвести оценку трудозатрат и длительности разработки по методике COSOMO II с использованием моделей композиции приложения и ранней разработки архитектуры. Определить среднюю численность команды разработчиков. На основе экспертной оценки стоимости человеко-месяца произвести предварительную оценку бюджета проекта. Дать заключение о применимости метода функциональных точек и модели COSOMO II, а также их сравнения с базовой моделью COSOMO для решения поставленной задачи с учетом своего варианта.

Содержание индивидуального задания:

Компания получила заказ на разработку клиентского мобильного приложения брокерской системы. Программа позволяет просматривать актуальную биржевую информацию, производить сделки и отслеживать их выполнение. Приложение имеет 4 страницы: авторизация, биржевые сводки, заявки, новая заявка. Страница авторизации содержит два поля ввода и одну командную кнопку, а также флажок для запоминания параметров авторизации. Страница биржевых сводок содержит таблицу, содержащую колонки: Ценная бумага, Цена, Изменение, кнопку «Добавить» и диалоговое окно с одним полем для ввода и двумя командными кнопками. Заявки содержат таблицу, содержащую колонки: Тип, Имя бумаги, Цена, Количество. При нажатии на любую строку таблицы появляется контекстное меню с возможностью удалить или изменить заявку. Страница новой заявки позволяет оформить заявку на покупку или продажу ценной бумаги и состоит из 4 полей: Бумага, Цена, Покупка (булева переменная) и кнопка «Оформить».

Характеристики продукта:

1. Обмен данными - 5.
2. Распределенная обработка -5
3. Производительность -3
4. Эксплуатационные ограничения по аппаратным ресурсам – 2
5. Транзакционная нагрузка – 3
6. Интенсивность взаимодействия с пользователем (оперативный ввод данных) – 4
7. Эргономические характеристики, влияющие на эффективность работы конечных пользователей - 1

8. Оперативное обновление – 4
9. Сложность обработки – 4
10. Повторное использование – 0
11. Легкость инсталляции – 1
12. Легкость эксплуатации/администрирования – 2
13. Портруемость – 2
14. Гибкость- 2

Для реализации проекта была сформирована новая команда разработчиков, у отдельных членов которой имеется некоторый опыт создания систем подобного типа. В целях сплочения команды были проведены определенные мероприятия, что обеспечило на старте проекта приемлемую коммуникацию внутри коллектива. Заказчик не настаивает на жесткой регламентации процесса, однако график реализации проекта довольно жесткий. Несмотря на то, что предметная область является для разработчиков относительно новой, анализу архитектурных рисков было уделено лишь некоторое внимание. Организация только начинает внедрять методы управления проектами и формальные методы оценки качества процесса разработки. Надежность и уровень сложности (RCPX) разрабатываемой системы оцениваются как очень высокие, повторного использования компонентов не предусматривается (RUSE). Возможности персонала (PERS) – средние, его опыт работы в разработке систем подобного типа (PREX) низкий. Сложность платформы (PDIF) высокая. Разработка предусматривает очень интенсивное использование инструментальных средств поддержки (FCIL). Заказчик настаивает на жестком графике (SCED).

Описание методики функциональных точек

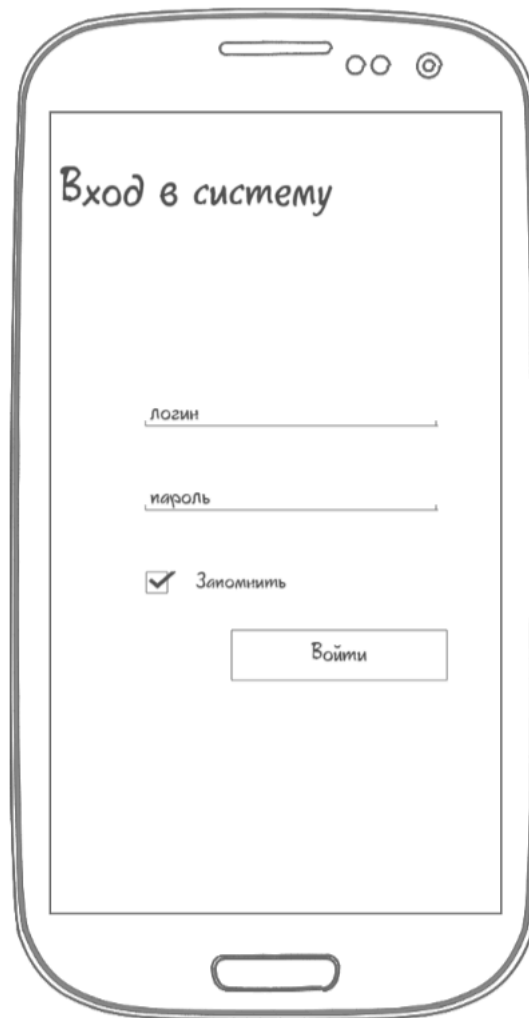
Данный метод используется для измерения производительности взамен устаревшего линейного подхода, где производительность измерялась количеством строк программного кода.

Преимуществом данного метода является то, что поскольку применение функциональных точек основано на изучении требований, то оценка необходимых трудозатрат может быть выполнена на самых ранних стадиях работы над проектом.

Определение числа функциональных точек является методом количественной оценки ПО, применяемым для измерения функциональных характеристик процессов его разработки и сопровождения независимо от технологии, использованной для его реализации.

Трудоемкость вычисляется на основе функциональности разрабатываемой системы, которая, в свою очередь, определяется путем выявления функциональных типов — логических групп взаимосвязанных данных, используемых и поддерживаемых приложением, а также элементарных процессов, связанных с вводом и выводом информации.

Авторизация



На данной странице осуществляется ввод логина и пароля пользователя для входа в систему. Страница содержит два поля ввода и одну командную кнопку, а также флажок-переключатель, который активируется при необходимости запоминания параметров авторизации.

ILF (1 локальный файл для запоминания логина и пароля пользователя и 1 таблица в БД для возможности входа в приложение):

- 1) локальный файл - RET (2 * string – логин и пароль) – 1, DET – 2 (2 столбца - логин и пароль)
- 2) таблица в БД - RET (2 * string – логин и пароль) – 1, DET – 2 (2 столбца - логин и пароль)

EI (ввод логина и пароля для сверки с логином и паролем из БД и запоминания логина и пароля локально в системе)

FTR=2 (2 локальных файла), DET=4 (поля «логин», «пароль», «запомнить», «Войти»)

EQ – уведомление об ошибке (неправильный логин или пароль)

FTR=1 (обращение к таблице из БД), DET=1 (сообщение «неправильный логин или пароль»)

Биржевые сводки

Биржевые сводки отражают текущую ситуацию на бирже. Страница содержит таблицу, кнопку «Добавить» и диалоговое окно с одним полем для ввода и двумя командными кнопками.

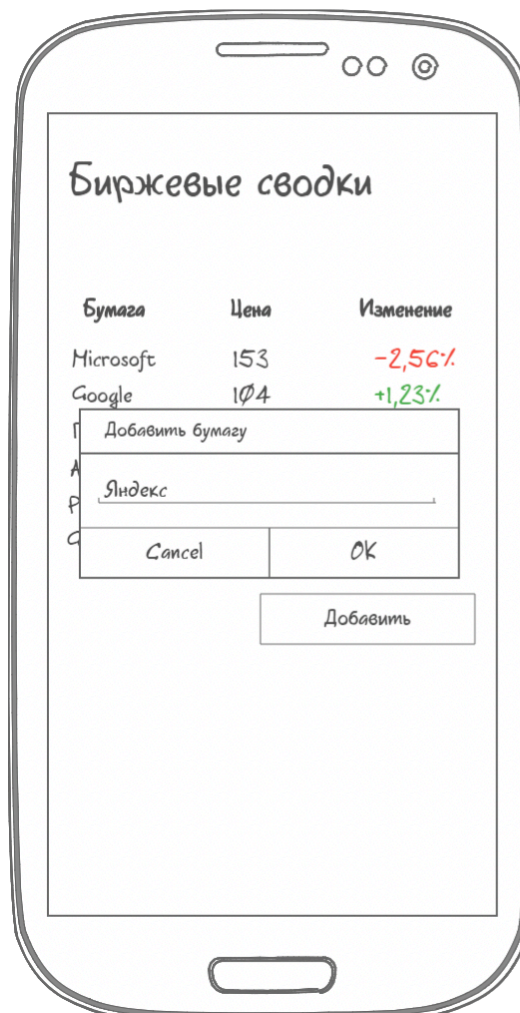


Таблица содержит три колонки: цены Ценная бумага (имя бумаги), Цена (цена за одну ценную бумагу), Изменение (изменение бумаги со времени последнего закрытия биржи). Кнопка «Добавить» вызывает диалоговое окно для добавления новой бумаги (окно состоит из поля ввода и кнопок ОК, Cancel).

ILF (таблица в БД, хранящая информацию о биржевых сводках, столбцы «бумага», «цена», «изменение»)

RET – 2 (string - бумага, 2 * float – цена/изменение), DET – 3 (3 столбца в таблице)

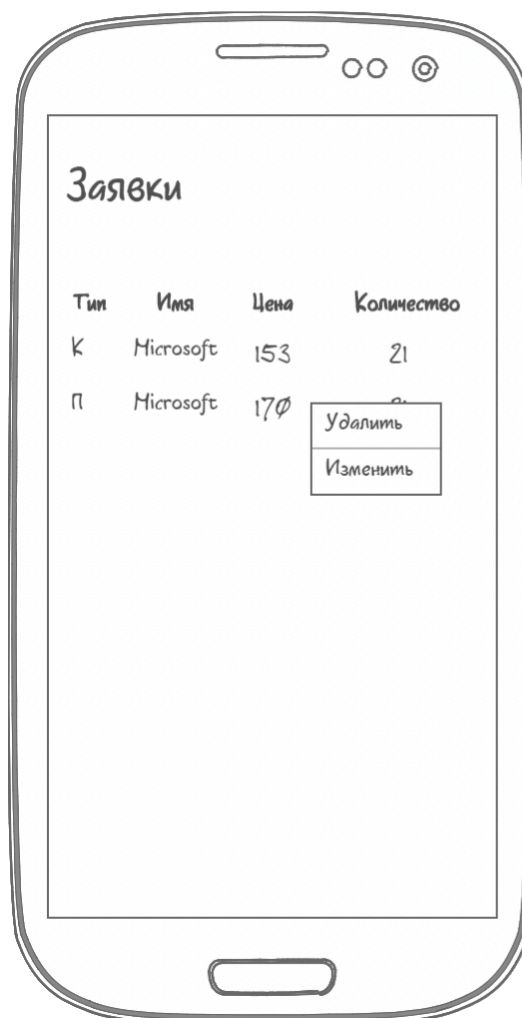
EO – запрос по сводкам

FTR – 1 (запрос к 1-ой таблице), DET – 3 (необходимы 3 столбца «бумага», «цена», «изменение»)

EI – добавление бумаги

FTR – 1 (запрос к одной таблице из БД), DET – 4 (пользователь нажимает кнопку «Добавить», вводит название бумаги, может нажать кнопку «ОК» или «Cancel»)

Заявки



Заявки содержат таблицу, отображающую текущие (еще не выполненные) заявки на покупку или продажу ценных бумаг. Таблица содержит четыре поля: Тип (покупка/продажа), Имя бумаги, Цена по которой готовы покупать/продаваться бумаги, Количество бумаг для покупки/продажи. При нажатии на любую строку таблицы появляется контекстное меню с возможностью удалить или изменить заявку.

ILF – таблица в БД, хранящая столбцы «Тип», «Имя», «Цена», «Количество».

RET (bool - тип, str - имя, float - цена, int - количество) – 4, DET – 4 (4 столбца)

ЕО – получение списка заявок из таблицы БД

FTR – 1 (запрос к 1-ой таблице), DET – 4 (получение 4 столбцов из БД)

ЕІ – пользователь может удалить или изменить запись.

Удаление: FTR – 1 (запрос к 1-ой таблице из БД), DET – 6 (поля «Тип», «Имя», «Цена», «Количество», кнопка «Удалить», сообщение об успешном удалении)

Изменение: FTR – 1 (запрос к 1-ой таблице из БД), DET – 6 (поля «Тип», «Имя», «Цена», «Количество», кнопка «Изменить», сообщение об успешном изменении)

Новая заявка

Новая заявка

Бумага

Цена

Количество

Покупка ☒ ON

Страница позволяет оформить заявку на покупку или продажу ценной бумаги. Страница состоит из 4 полей: Бумага (имя бумаги), Цена (цена по которой необходимо купить/продать бумагу), Покупка (булева переменная в значение true обозначает покупку, false – продажа) и кнопки «Оформить» - для подтверждения оформления заявки.

ILF - используется та же таблица из БД, что и для экрана «Заявки».

EI – создание новой заявки

FTR – 1 (обращение к 1-ой таблице), DET – 5 (поля «Бумага», «Цена», «Количество», «Покупка», кнопка «Оформить»)

Описание методики COSOMO II

COSOMO II является развитием стандартного COSOMO. В методику входят три различные модели оценки стоимости:

- Модель композиции приложения – это модель, которая подходит для проектов, созданных с помощью современных инструментальных средств. Единицей измерения служит объектная точка
- Модель ранней разработки архитектуры. Эта модель применяется для получения приблизительных оценок проектных затрат периода выполнения проекта перед тем как будет определена архитектура в целом. В этом случае используется небольшой набор новых драйверов затрат и новых уравнений оценки. В качестве единиц измерения используются функциональные точки либо KSLOC
- Постархитектурная модель –наиболее детализированная модель COSOMOII, которая используется после разработки архитектуры проекта. В состав этой модели включены новые драйверы затрат, новые правила подсчета строк кода, а также новые уравнения

Модель композиции приложения	Модель ранней разработки архитектуры	Постархитектурная модель
Грубые входные данные	Ясно понимаемые особенности проекта	Детальное описание проекта
Оценки низкой точности	Оценки умеренной точности	Высокоточные оценки
Приблизительные требования	Ясно понимаемые требования	Стабилизировавшиеся основные требования
Концепция архитектуры	Ясно понимаемая архитектура	Стабильная базовая архитектура

Рисунок 1. Характеристики моделей оценки

В данной работе рассматриваются модель композиции приложения и модель ранней разработки архитектуры.

Показатели проекта:

- Новизна проекта (PREC) – полное отсутствие прецедентов, полностью непредсказуемый проект (т.к. была сформирована новая команда разработчиков, только отдельные члены имели некоторый опыт создания систем подобного типа)

- Гибкость процесса разработки (FLEX) – большей частью согласованный процесс (график жесткий, точной регламентации нет)
- Разрешение рисков в архитектуре системы (RESL) – некоторое (40%)
- Сплоченность команды (TEAM) – некоторая согласованность (команд новая, но были проведены определенные мероприятия по сплочению)
- Уровень развития процесса разработки (PMAT) – уровень 1+ (только начинают внедрять)

Модель композиции приложения

Для применения модели композиции приложения необходимо оценить количество объектных точек в проектируемом проекте.

По условию задачи, проект состоит из четырех форм, двух отчетов и двух модулей.

Рассмотрим формы:

1. Авторизация. Средняя сложность
2. Биржевые сводки. Низкая сложность
3. Заявки. Низкая сложность
4. Новая заявка. Средняя сложность

Отчеты:

1. Список заявок. Средняя сложность
2. Биржевые сводки. Средняя сложность

Модули:

1. Модуль авторизации
2. Модуль работы с биржей

Подсчитаем количество объектных точек:

2 формы низкой сложности, 2 формы средней сложности, 2 отчета средней сложности, 2 модуля. Итого:

$$2 * 1 + 2 * 4 + 2 * 5 + 2 * 10 = 40$$

По условию задачи количество объектных точек было оценено в 40 штук.

Результаты расчетов:

Факторы, влияющие на показатель степени

PREC (Новизна проекта)

Полное отсутствие прецедентов, полностью непредсказуемый проект

FLEX (Гибкость процесса разработки)

Большой частью согласованный процесс

RESL (Разрешение рисков в архитектуре системы)

Некоторое (40 %)

TEAM (Сплоченность команды)

Некоторая согласованность

PMAT (Уровень зрелости процесса разработки)

Уровень 1+ CMM

Модель композиции приложения

Модель ранней разработки архитектуры

RUSE (%)

0

Опытность команды/разработчика

Низкая

Экранные формы

Простые

2

Средние

2

Сложные

0

Отчеты

Простые

0

Средние

2

Сложные

0

Модули на языках 3 поколения

2

Результаты: модель композиции приложения

Трудозатраты

5.143

Длительность

5.561

Средняя зарплата

60000

Численность команды разработчиков

1

Бюджет

308571.429

Рассчитать

Результаты: модель ранней разработки архитектуры

Трудозатраты

0

Длительность

0

Средняя зарплата

60000

Численность команды разработчиков

0

Бюджет

0

Рассчитать

Количество строк кода:

2515

Модель ранней разработки архитектуры

PERS (возможности персонала) - номинальный

RCPX (надежность и уровень сложности разрабатываемой системы) – очень высокий

RUSE (повторное использование компонентов) - низкий

PDIF (сложность платформы разработки) - высокий

PREX (опыт персонала) - низкий

FCIL (средства поддержки) – очень высокий

SCED (требуемые сроки разработки) – очень высокий

Результаты расчетов:

Факторы, влияющие на показатель степени

PREC (Новизна проекта)

Полное отсутствие прецедентов, полностью непредсказуемый проект

FLEX (Гибкость процесса разработки)

Большой частью согласованный процесс

RESL (Разрешение рисков в архитектуре системы)

Некоторое (40 %)

TEAM (Сплоченность команды)

Некоторая согласованность

PMAT (Уровень зрелости процесса разработки)

Уровень 1+ CMM

Модель композиции приложения

Модель ранней разработки архитектуры

PERS (Классификация персонала)

Номинальный

RCPX (Надежность и сложность разрабатываемой системы)

Очень высокий

RUSE (Повторное использование компонентов)

Низкий

PDIF (Сложность платформы разработки)

Высокий

PREX (Опыт персонала)

Низкий

FCIL (Средства поддержки)

Очень высокий

SCED (Требуемые сроки разработки)

Очень высокий

Результаты: модель композиции приложения

Трудозатраты

5.143

Длительность

5.561

Средняя зарплата

60000

Численность команды разработчиков

1

Бюджет

308571.429

Рассчитать

Результаты: модель ранней разработки архитектуры

Трудозатраты

16.088

Длительность

8.546

Средняя зарплата

60000

Численность команды разработчиков

2

Бюджет

965286.03

Рассчитать

Количество строк кода:

2515

Выводы

Методика COSOMO II, как и ее первая версия, позволяет досочно быстро оценить длительность и трудозатраты проекта, основываясь на субъективных данных. В условиях отсутствия объективной информации о предполагаемых трудозатратах особенно важно правильно спрогнозировать характеристики проекта.

Методика функциональных точек позволяет оценить размер программного продукта на этапе его проектирования. С помощью этого подхода можно применять методики COSOMO, которым необходимо знание о размере продукта. Хотя методика функциональных точек является неточной (как и любая методика прогнозирования), этих результатов достаточно для общего понимания объемов работ и получения приблизительных оценок параметров проекта.

В ходе выполнения лабораторной работы было выяснено, что модель композиции приложения дает более оптимистичный прогноз, по сравнению с моделью ранней архитектуры приложения.