

Тема 3.

# Задачи и методы моделирования и анализа программных продуктов

# Анализ программных продуктов

## Цели, корректность и направления анализа

- Большое количество программных продуктов на современном рынке зачастую вызывает вопросы пользователя: какое ПО выбрать, какое будет качественным, какое максимально отвечает потребностям? Вопросы о ПО возникают не только у пользователей, но и у программистов: как разработать качественное и востребованное ПО, что для того нужно сделать?
- Для ответов на эти вопросы используют два вида мыслительных операций, используемых для исследования объектов, — анализ и синтез. Анализ позволяет выделить и изучить отдельные части объекта исследования, а синтез — объединить результаты исследования по каким-либо критериям и сделать выводы.

## Цели, корректность и направления анализа

3

- Что необходимо для анализа чего-либо, в том числе и ПО? Прежде всего должен быть **предмет исследования** — то, что необходимо изучить. Например, планируется проанализировать ПО, в этом случае предметом исследования будет ПО.
- Затем следует определиться с **объектом исследования**. (**Объект исследования** – та область, то пространство, в рамках которого находится или содержится то, что будет изучаться. **Предмет исследования** – часть **объекта**). Это очень важно. Есть целые группы ПО, которые можно выделить по типу использования ПО и его функциональному назначению.

# Цели, корректность и направления анализа

4

Существуют три большие **группы ПО**:

- системное — комплекс программ, предназначенных для управления работой компьютера;
- прикладное — комплекс программ, предназначенных для решения тех или иных задач пользователей;
- инструментальное — совокупность программ, предназначенных для проектирования, разработки и сопровождения ПО. Среди этих групп в зависимости от функционального назначения можно выделить классы различных программ. При исследовании разных программ могут быть интересны их различные характеристики, т.е. анализ проводится по разным критериям. Таким образом, объект исследования определяет, по каким критериям можно вести анализ данного объекта.

## Цели, корректность и направления анализа

5

Для каждого объекта можно выделить массу критериев анализа, характеризующих те или иные свойства объекта. Выбор критериев для анализа зависит от **цели исследования**. Именно она определяет направление исследования и необходимые результаты, и от нее будет зависеть набор критериев для анализа того или иного объекта, в нашем случае — программного продукта.

После постановки цели анализа выделяют ряд задач, которые нужно решить для достижения этой цели. И самая важная задача — подбор критериев для анализа выбранного объекта, так как от этого будут зависеть корректность и полнота исследования.

## Критерии анализа и оценки программного обеспечения

- Критерии анализа любого объекта, в том числе и программного обеспечения, могут быть объективными и субъективными. Например, корректность работы ПО, позволяющая оценить его правильную работу при вводе корректных данных, будет объективным критерием, а удобство использования интерфейса — субъективным. Субъективные критерии важны не меньше, чем объективные. Но необходимо понимать, что оценка субъективных критериев может различаться у разных пользователей.
- **Поэтому для** корректной оценки субъективного критерия **необходимо:**
  - разработать шкалу оценки того или иного критерия;
  - опросить группу пользователей в соответствии с разработанной шкалой;
  - найти среднее значение оценки.

# Критерии анализа и оценки программного обеспечения

Для корректного анализа критерии должны быть:

- валидными, т. е. оценивать именно то, что необходимо оценить в ходе анализа;
- предметными, т. е. относиться к оценке объекта или предмета исследования в контексте объекта;
- непротиворечивыми;
- интерпретируемыми, т. е. позволять интерпретацию полученной оценки;
- проверяемыми;
- достаточно полными, т. е. максимально охватить все существенные согласно цели исследования характеристики исследуемого объекта.

Отметим еще раз: выбор критериев для анализа ПО зависит от следующих составляющих:

- цель;
- анализ;
- объект исследования.

# Критерии анализа и оценки программного обеспечения

- Рассмотрим существующие критерии оценки для анализа качества ПО.
- **Можно встретить различные** определения понятия «качество ПО»:
  - величина, отражающая, в каком объеме в программный продукт включен набор желаемых функций для повышения эффективности программного продукта в течение его жизненного цикла;
  - качество — это полнота свойств и характеристик продукта, процесса или услуги, которые обеспечивают способность удовлетворять заявленным или подразумеваемым потребностям (определение ISO);
  - степень, в которой оно обладает требуемой комбинацией свойств (определение IEEE - это профессиональная некоммерческая ассоциация).



# Критерии анализа и оценки программного обеспечения

**Существуют** группы стандартов для оценки качества ПО:

- ISO/IEC 25010:2011 (ГОСТ Р ИСО/МЭК 25010—2015 «Информационные технологии (ИТ). Системная и программная инженерия. Требования и оценка качества систем и программного обеспечения (SQaRE). Модели качества систем и программных продуктов») — группа стандартов, описывающих модели качества программных продуктов;
- ISO 9001:2015, ISO/IEC 9126 (ГОСТ Р ИСО/МЭК 9126—93 «Информационная технология (ИТ). Оценка программной продукции. Характеристики качества и руководства по их применению») — стандарты качества ПО;
- ISO/IEC 14598 — набор стандартов, регламентирующих способы оценки характеристик качества.

# Модели качества программного обеспечения

10

**Модель Мак-Кола.** Первая модель качества была предложена Мак-Колом . Она имела **три направления анализа:**

- использование (корректность, надежность, эффективность, целостность, практичность);
- модификация (тестируемость, гибкость, сопровождаемость — факторы качества, важные для разработки новой версии ПО);
- переносимость (мобильность, возможность многократного использования, функциональная совместимость — факторы качества, важные для переносимости программного продукта на другие аппаратные и программные платформы).

# Модель Мак-Кола

11



# Модель Боэма

12



# Модель FURPS/FURPS+

13

Одна из моделей анализа качества ПО была предложена Грейди и компанией Hewlett Packard. Она построена аналогично предыдущим, но в отличие от них состоит из двух слоев:

- первый слой определяет характеристики;
- второй — связанные с ними атрибуты.

В модели выделяли две категории требований: функциональные (F) и нефункциональные (URPS). Эти категории могут использоваться в качестве требований к программному продукту и в оценке качества ПО. Модель широко используется современными программистами. Аббревиатура FURPS описывает требования к качеству ПО. Символ «+» используется для расширения модели, добавляя к оценке ограничения проекта, ограничения, накладываемые на взаимодействие с внешними системами, требования к выполнению, физические требования и требования к лицензированию.

# Модель FURPS/FURPS+

14

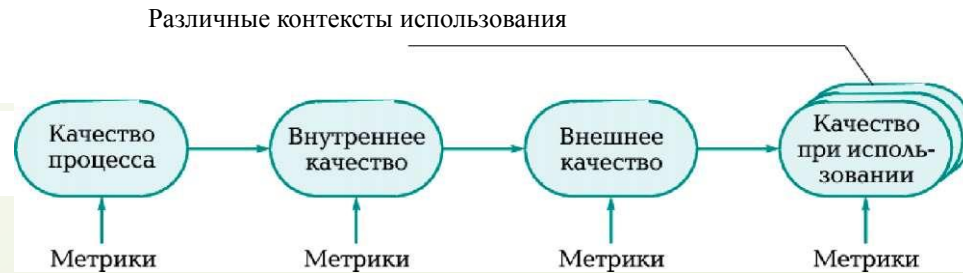
Критерий оценки	Описание критерия
F — Functionality (функциональность)	Функциональные требования: особенности, возможности, безопасность
U — Usability (практичность)	Требования к удобству использования: человеческий фактор, эстетика, последовательность, документация
R — Reliability (надежность)	Требования к надежности: частота возможных сбоев, отказоустойчивость, восстанавливаемость
P — Performance (производительность)	Требования к производительности: время отклика, использование ресурсов, эффективность, мощность, масштабируемость
S — Supportability (эксплуатационная пригодность)	Требования к поддержке: возможность поддержки, ремонтпригодность, гибкость, модифицируемость, модульность, расширяемость, возможность локализации

# Модель качества ПО по стандартам ISO 9126

15

**ISO 9126** — международный стандарт, определяющий оценочные характеристики качества ПО.

Модель качества по стандартам ISO 9126 рассматривает несколько взаимосвязанных метрик качества:



качества	внутреннего	внешнего	качества
процесса	качества	качества	при исполь-
ISO/IEC 9126-1	ISO/IEC 9126-2	ISO/IEC 9126-3	зовании
		ISO/IEC 9126-4	

**Основные аспекты качества программного обеспечения**

Основные аспекты качества программного обеспечения:

- внешнее качество, которое задается требованиями заказчика в спецификациях и отражается в характеристиках конечного продукта;
- внутреннее качество, связанное с характеристиками **ПО** и обусловленное процессом разработки и промежуточными этапами жизненного цикла **ПО**;
- качество при использовании **ПО** в процессе его эксплуатации.



**В первой части стандарта ISO 9126-1** качество **ПО** рассматривается в разрезе шести структурных наборов характеристик, каждая из которых имеет свои субхарактеристики (подхарактеристики).

**1. Функциональность** — набор атрибутов, характеризующий, соответствие функциональных возможностей **ПО** набору требуемой пользователем функциональности. Детализируется следующими субхарактеристиками:

- пригодность для применения;
- корректность (правильность, точность);
- способность к взаимодействию (в частности, сетевому);
- защищенность.

**В первой части стандарта ISO 9126-1** качество ПО рассматривается в разрезе шести структурных наборов характеристик, каждая из которых имеет свои субхарактеристики (подхарактеристики).

**2. Надежность** — набор атрибутов, относящихся к способности ПО сохранять свой уровень качества функционирования в установленных условиях за определенный период времени. Детализируется следующими субхарактеристиками:

- уровень завершенности (отсутствие ошибок);
- устойчивость к дефектам;
- восстанавливаемость;
- доступность;
- готовность.

**В первой части стандарта ISO 9126-1** качество ПО рассматривается в разрезе шести структурных наборов характеристик, каждая из которых имеет свои субхарактеристики (подхарактеристики).

**3. Практичность (применимость)** — набор атрибутов, относящихся к объему работ, которые требуют для исполнения и индивидуальной оценки такого исполнения определенным или предполагаемым кругом пользователей. Детализируется следующими субхарактеристиками:

- понятность;
- обучаемость;
- простота использования;
- привлекательность.

**В первой части стандарта ISO 9126-1** качество ПО рассматривается в разрезе шести структурных наборов характеристик, каждая из которых имеет свои субхарактеристики (подхарактеристики).

**4. Эффективность** — набор атрибутов, относящихся к соотношению между объемом используемых ресурсов при установленных условиях и уровнем качества функционирования ПО. Детализируется следующими субхарактеристиками:

- эффективность с течением времени;
- используемость ресурсов.

**В первой части стандарта ISO 9126-1** качество ПО рассматривается в разрезе шести структурных наборов характеристик, каждая из которых имеет свои субхарактеристики (подхарактеристики).

**5. Сопровождаемость** — набор атрибутов, относящихся к объему работ, которые необходимы для проведения конкретных изменений (модификаций) ПО. Детализируется следующими субхарактеристиками:

удобство для анализа;

- изменяемость;
- стабильность;
- тестируемость.

**В первой части стандарта ISO 9126-1** качество ПО рассматривается в разрезе шести структурных наборов характеристик, каждая из которых имеет свои субхарактеристики (подхарактеристики).

**6. Мобильность** — набор атрибутов, относящихся к способности ПО быть перенесенным из одного окружения в другое. Детализируется следующими субхарактеристиками:

- адаптируемость;
- простота установки (инсталляции);
- совместимость;
- взаимозаменяемость;
- согласованность мобильности.

## **Вторая и третья части стандарта ISO 9126**

посвящены формализации внешних и внутренних метрик для характеристик качества сложных программных средств. Они содержат унифицированную рубрикацию, где отражены имя и назначение метрики, метод ее применения, способ измерения, тип шкалы метрики, тип измеряемой величины, исходные данные для измерения и сравнения, а также этапы жизненного цикла программного средства, к которым применима метрика.

**Четвертая часть стандарта ISO 9126-4** предназначена для покупателей, поставщиков, разработчиков, сопровождающих, пользователей и менеджеров качества программных средств. В ней рассматриваются рекомендуемые виды характеристик программных средств.

# Заключение

24

Анализ ПО — это часть процесса разработки программного обеспечения, включающая в себя сбор требований к программному обеспечению, их систематизацию, выявление взаимосвязей, а также документирование.



# Заключение

25

Анализ модели оценки качества ПО свидетельствует, что существуют разные подходы к его оценке и большое количество критериев анализа.

Говоря об анализе ПО, мы подразумевали, что есть некоторое ПО, которое можно проанализировать по определенным критериям и по результатам анализа сделать выводы. Очень часто перед пользователями и разработчиками встает задача не просто анализа программного продукта, а сравнительного анализа определенных программных продуктов.

# Сравнительный анализ программных

26 **продуктов.**

## Критерии анализа и представление

### результатов

При сравнительном анализе программных продуктов целесообразно выбирать продукты с аналогичным функционалом. Мы можем сравнить велосипед с автомобилем, так как и велосипед, и автомобиль — это средства передвижения, но гораздо интереснее сравнить разные марки велосипедов или разные марки автомобилей между собой. Если говорить о ПО, то операционная система скорее всего будет сравниваться с операционной системой, а информационная система (ИС) для ведения бухгалтерской деятельности — с аналогичной ИС.

Любой из указанных ранее критериев оценки, а также полный набор критериев из какой-либо методики оценки качества ПО, мы можем использовать для сравнительного анализа ПО. Вопрос в том, нужны ли будут все перечисленные критерии для оценки? Ведь оценка по всем критериям какой-либо модели будет трудоемкой и займет большое количество времени.

# Сравнительный анализ программных

27 **продуктов.**

## Критерии анализа и представление

### результатов

Чтобы ответить на этот вопрос, вспомним, что у анализа должна быть цель. Выбор критериев анализа обусловлен целью и объектом исследования.

Потребность в сравнительном анализе ПО может возникнуть у разработчиков ПО, чтобы создать конкурентноспособный продукт, востребованный у потребителя, а также у пользователя, который задумывается, какое ПО купить или установить на свой компьютер.

Компьютерные технологии широко применяются в различных сферах нашей жизни. Большинство из нас имеют персональные компьютеры, а значит, являются пользователями ПО.

# Сравнительный анализ программных

## 28 продуктов.

### Критерии анализа и представление

Чаще всего при выборе ПО мы рассматриваем:

#### результатов

- его функциональные возможности;
- интерфейс (удобство использования, простоту освоения, дизайн);
- наличие поддержки (руководство пользователя и система помощи) ;
- требования к аппаратному обеспечению для корректной работы;
- цену ПО;
- надежность — бесперебойное выполнение всех заявленных функций, корректное завершение работы при возникновении различных возможных ошибок (по вине либо не по вине пользователя). Этот сокращенный список критериев оценки ПО может быть детализирован и расширен.

# Сравнительный анализ программных 29 продуктов.

## Критерии анализа и представление

### результатов

Результаты анализа могут быть представлены в любой удобной для их изучения форме, например, в виде:

- текстового описания, в котором присутствует информация об объекте, целях и задачах исследования, выбранных критериях оценки, полученных результатах с их интерпретацией и выводами;
- табличного представления результатов сравнения;
- графического представления — схемы, рисунки, диаграммы.
- В конце исследования (сравнительного анализа) обязательно необходимо сделать выводы.

# Примеры сравнительного анализа программных продуктов.

## Сравнительный анализ текстовых редакторов

**Условия.** Заказчики — директор ООО «Интеграл» и его секретарь, занимающаяся документацией компании. На компьютере секретаря установили операционную систему Windows 7 с текстовыми редакторами «Блокнот» и WordPad.

- Необходимо произвести сравнительный анализ программных продуктов Microsoft Word, WordPad и «Блокнот» для выбора оптимального продукта в соответствии с рабочими задачами секретаря.
- Для решения данной задачи необходимо узнать у секретаря, какой функционал текстового редактора ей нужен для работы с документацией. Мы выяснили, что необходимо:
  - работать с файлами — создавать их, редактировать и сохранять в различных форматах;
  - создавать и редактировать документы с отформатированным шрифтом и абзацами, таблицами, списками, распределением текста по колонкам;
  - работать с многостраничными документами — вставлять нумерацию страниц и создавать оглавление;
  - создавать схемы, вставлять изображения и формулы;
  - вставлять диаграммы из файлов Excel;
  - использовать предварительный просмотр документа и выводить его на печать.
- Побеседовав с директором, мы выяснили, что он хотел бы получить сделанную документацию и экономно расходовать денежные средства: «Никакого лишнего программного обеспечения!».

# Примеры сравнительного анализа программных продуктов.

## Сравнительный анализ текстовых редакторов

**Критерии и задачи исследования.** Предмет — ПО. Объекты: текстовые редакторы Microsoft Word, WordPad и «Блокнот». Цель — произвести сравнительный анализ текстовых редакторов для выбора редактора с оптимальным функционалом для работы секретаря.

Задачи:

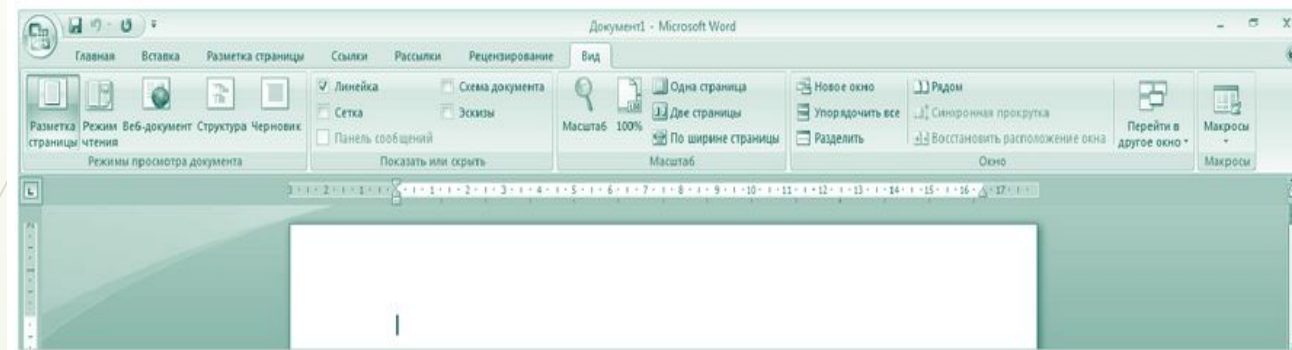
- сформировать критерии оценки программных продуктов в зависимости от потребностей пользователей;
- изучить ПО;
- произвести сравнительный анализ программных продуктов в соответствии с выбранными критериями;
- сделать выводы в зависимости от полученных результатов.

**Последовательность проведения исследования.** Рассмотрим интерфейс ПО и сформируем таблицу с остальными критериями оценки функциональных возможностей ПО. Выявим, какими функциональными возможностями обладает каждый программный продукт.

# Примеры сравнительного анализа программных продуктов.

32

## Сравнительный анализ текстовых редакторов



### Интерфейс Microsoft Word

- строка заголовка;
- ленточные панели с пиктограммами команд (9 шт.), закладка со списком команд;
- рабочая область;
- строка состояния с возможностью просмотреть статистику документа, изменить вид и масштаб документа.



# Примеры сравнительного анализа программных продуктов.

## Сравнительный анализ текстовых редакторов



Интерфейс WordPad

- строка заголовка;
- ленточные панели с пиктограммами команд (2 шт.), закладка со списком команд;
- рабочая область;
- строка состояния с возможностью изменить масштаб документа.

# Примеры сравнительного анализа программных продуктов.

## Сравнительный анализ текстовых редакторов

Функциональные возможности программного обеспечения для создания и редактирования документов секретаря			
Критерий оценки	Программный продукт		
	Microsoft Word	WordPad	«Блокнот»
Работа с файлами (создание, редактирование, сохранение)	Да	Да	Да
Критерий оценки	Программный продукт		
	Microsoft Word	WordPad	«Блокнот»
Форматы файлов при сохранении	rtf, doc, docx, html, txt, wps	rtf, doc, txt	txt
Форматирование шрифта	Тип шрифта, начертание, размер, стиль, заливка, цвет, видоизменение, положение на странице	Тип шрифта, начертание, размер, стиль, заливка, цвет	Тип шрифта, начертание, размер

# Примеры сравнительного анализа программных продуктов.

## Сравнительный анализ текстовых редакторов

Форматирование абзацев: установка абзацных отступов и выравнивание	Да	Да	Нет
Создание таблиц	Да	Нет	Нет
Создание списков	Да	Да	Нет
Распределение текста по колонкам	Да	Нет	Нет
Вставка нумерации страниц	Да	Нет	Нет
Создание оглавления	Да	Нет	Нет
Создание схем	Да	Нет	Нет
Вставка изображений	Да	Да	Нет
Вставка формул	Да	Нет	Нет
Предварительный просмотр	Да	Да	Да
Вывод на печать	Да	Да	Да
Поддержка встраивания и связывания объектов	Да	Да	Нет
Финансовые затраты	Необходимы дополнительные затраты на покупку лицензионного ПО	Устанавливается с операционной системой, не требует дополнительных финансовых затрат	

# Примеры сравнительного анализа программных продуктов.

## Сравнительный анализ текстовых редакторов

**Выводы.** По результатам исследования можно сделать выводы.

- Анализ интерфейсов ПО свидетельствует, что редактор программы Microsoft Word имеет больше возможностей для работы с текстовыми документами. Пока этот вывод — один из предварительных, поскольку мы еще не оценили функционал ПО в соответствии с потребностями секретаря компании.
- На основе анализа функциональных возможностей ПО для создания и редактирования документов можно сделать вывод, что для работы секретаря подойдет только текстовый редактор Microsoft Word. Другие редакторы не смогут удовлетворить потребности компании для разработки документов.

Заметим, что при анализе текстовых редакторов мы не использовали субъективные критерии.

# Сравнительный анализ скорости работы браузеров

37

**Условия.** У пользователя на компьютере установлены три браузера: Mozilla Firefox, версия 61.0.2; Google Chrome, версия 68.0.3440.106; Opera, версия 54.0.2952.64.

Необходимо выяснить, какой браузер работает быстрее.

**Критерии и задачи исследования.** Предмет — ПО. Объекты: браузеры Mozilla Firefox, версия 61.0.2; Google Chrome, версия 68.0.3440.106; Opera, версия 54.0.2952.64. **Цель** — произвести сравнительный анализ скорости работы браузеров, оценив, как быстро браузер открывает различные интернет-ресурсы на компьютере с определенным аппаратным обеспечением и как быстро происходит запуск браузера.

## **Задачи:**

- произвести сравнительный анализ скорости запуска браузеров (первый и повторный запуски);
- осуществить сравнительный анализ общей производительности браузеров, используя сервис [browserbench.org](http://browserbench.org).

# Сравнительный анализ скорости работы браузеров

38

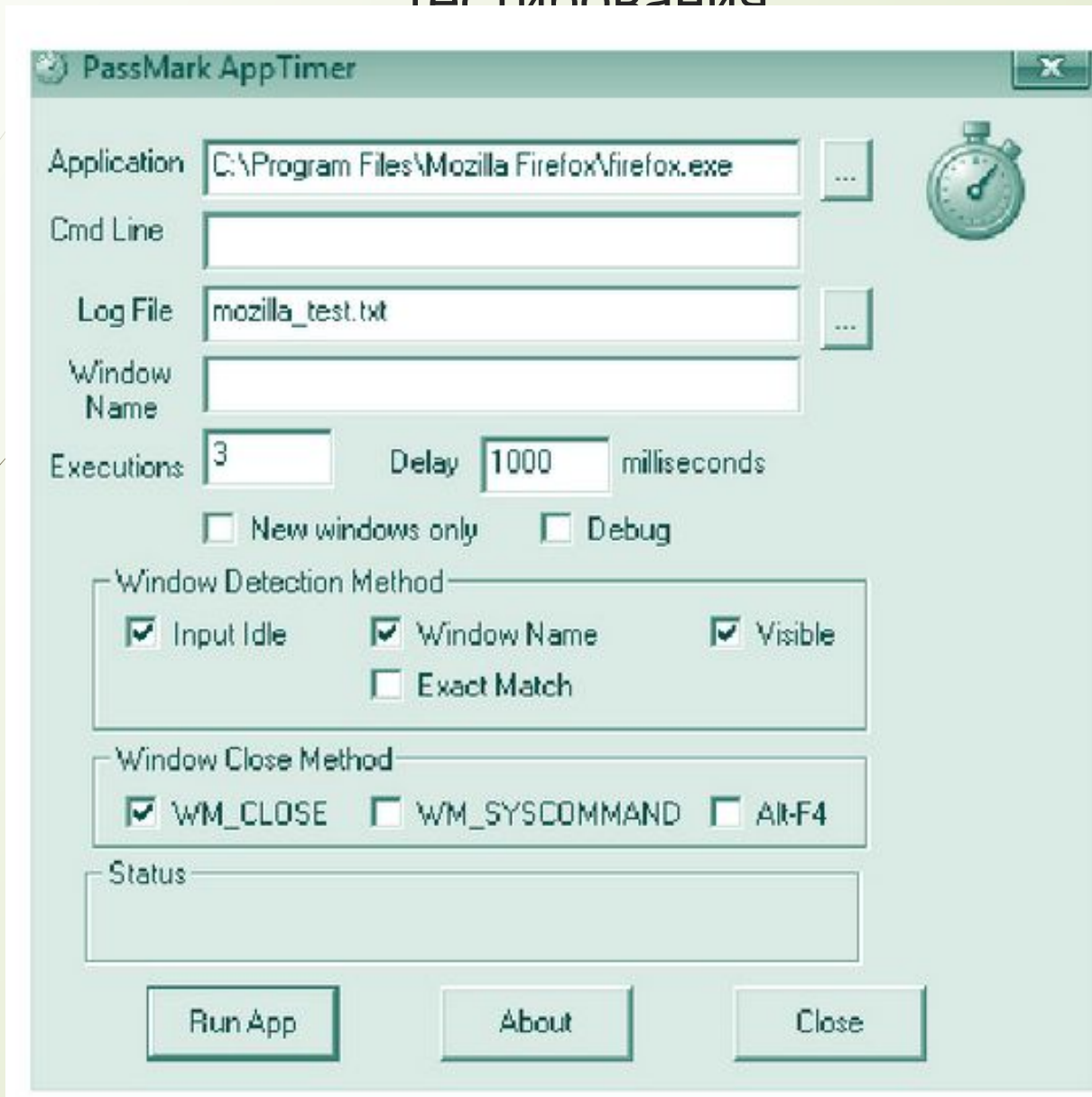
## Последовательность проведения исследования.

Рассмотрим основные этапы исследования.

- Проанализируем ПО. Существует множество утилит для исследования работы браузеров и иного ПО. В нашем исследовании оценим скорость загрузки интернет-страниц и время запуска браузера. Для получения информации о скорости запуска браузеров используем программу AppTimer, позволяющую получить информацию о скорости запуска ПО.
- Для тестирования ПО с помощью данной программы необходимо указать адрес приложения (Application), файл для записи результатов (Log File), количество тестирований через определенный промежуток времени (Execution), задержку между тестированиями в миллисекундах (Delay).

# Использование программы AppTimer для ТЕСТИРОВАНИЯ

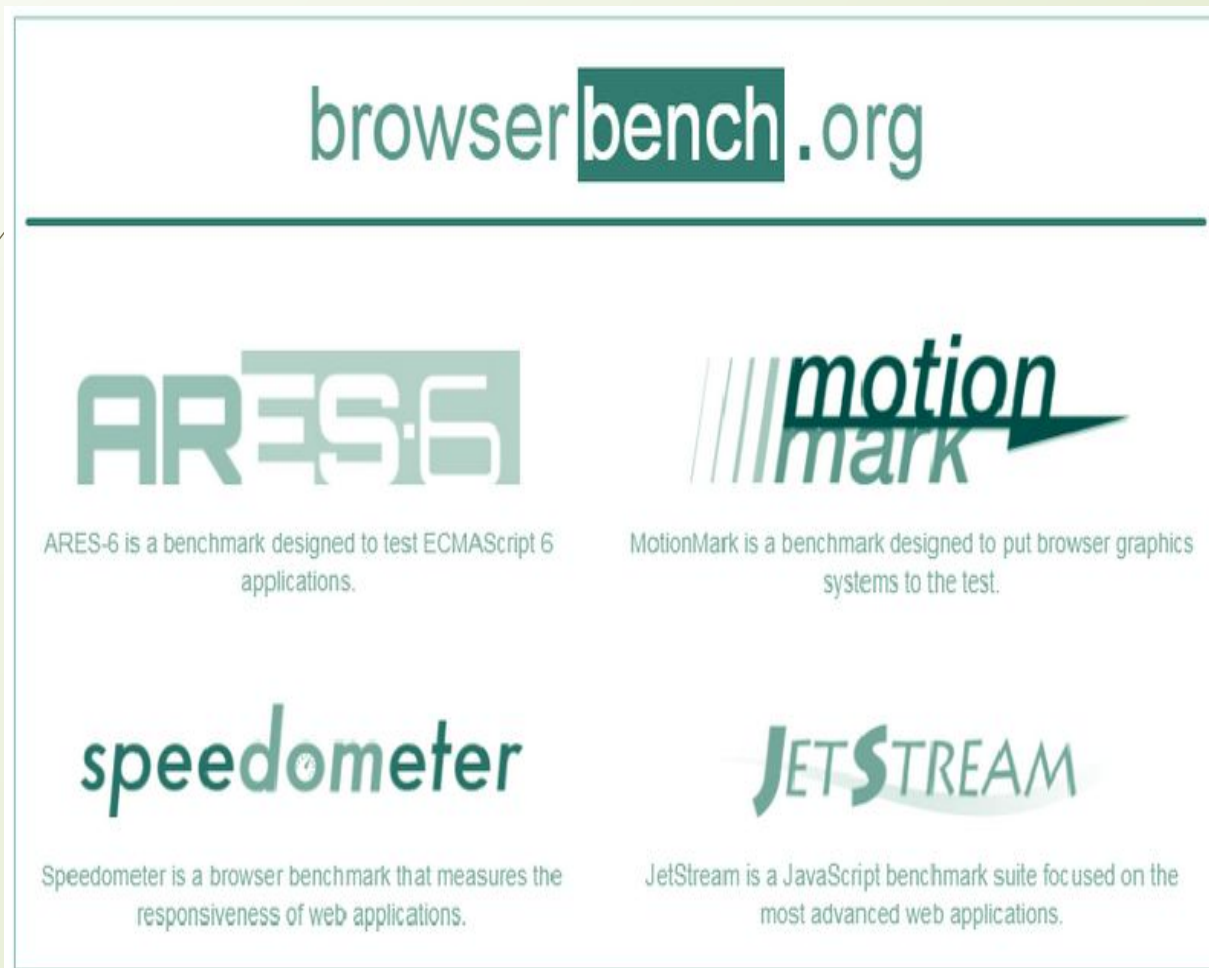
39



# Сервис <https://browserbench.org/> – тестирование общей производительности браузеров

40

Для исследования общей производительности браузеров используем сервис <https://browserbench.org/>. Сервис имеет несколько утилит:



The screenshot displays the browserbench.org website with a dark green header and a white background. Below the header, four benchmarking tools are listed in a 2x2 grid, each with its logo and a brief description.

Tool Name	Description
<b>ARES-6</b>	ARES-6 is a benchmark designed to test ECMAScript 6 applications.
<b>MotionMark</b>	MotionMark is a benchmark designed to put browser graphics systems to the test.
<b>speedometer</b>	Speedometer is a browser benchmark that measures the responsiveness of web applications.
<b>JETSTREAM</b>	JetStream is a JavaScript benchmark suite focused on the most advanced web applications.



## Утилиты <https://browserbench.org/>.

41

ARES -6 — измеряет время выполнения различных функций JavaScript. Состоит из четырех подтестов: Air, Basic, Babylon и ML;

Speedometer 2,0 — измеряет среднюю скорость открытия браузером большого количества интернет-страниц с различным контентом за минуту. Чем выше эта скорость, тем быстрее работает браузер;

JetStream — содержит большое количество тестов, которые позволяют оценить работу браузеров с контентом JavaScript при различных нагрузках и определить средний результат производительности браузеров по результатам выполнения различных тестов;

MotionMark — позволяет оценить производительность браузеров при обработке графики и анимации.

# Результаты скорости загрузки браузеров

42

Тестирование скорости загрузки браузеров				
Браузер	Скорость загрузки, %			
	Тест 1	Тест 2	Тест 3	Среднее значение
Первый запуск				
Opera	53,74	54,08	53,78	53,86
Google Chrome	51,21	51,47	51,21	51,30
Mozilla Firefox	54,42	54,08	52,10	53,53
Повторный запуск				
Opera	58,93	59,43	58,95	59,10
Google Chrome	56,54	56,41	55,23	56,06
Mozilla Firefox	58,78	56,64	57,86	57,76

# Результаты теста Speedometer 2.0.

43

Таблица 1.4. Тестирование производительности браузеров с помощью теста Speedometer 2.0

Браузер	Производительность, %			
	Тест 1	Тест 2	Тест 3	Среднее значение
Opera	19,9	19,3	19,3	19,5
Google Chrome	21,8	21,5	21,7	21,7
Mozilla Firefox	16,7	16,2	16,9	16,6

# Результаты теста MotionMark

44

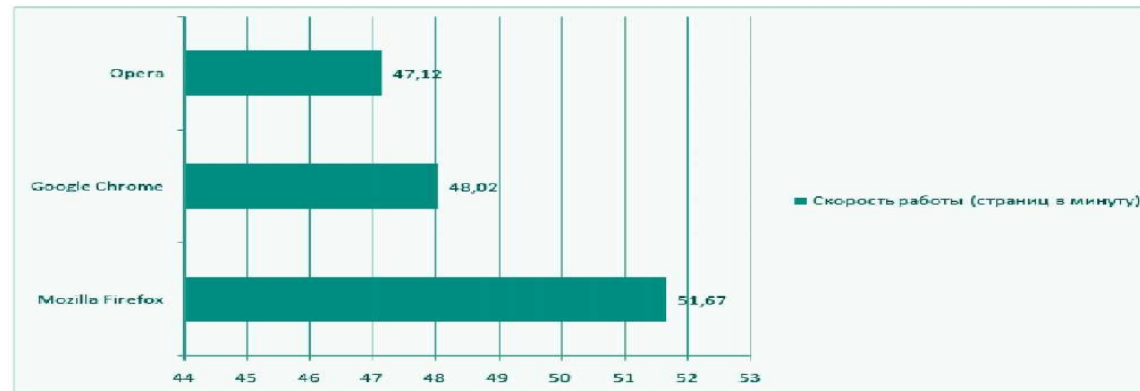


Рис. 1.17. Сравнительный анализ скорости работы браузеров с помощью теста JetStream

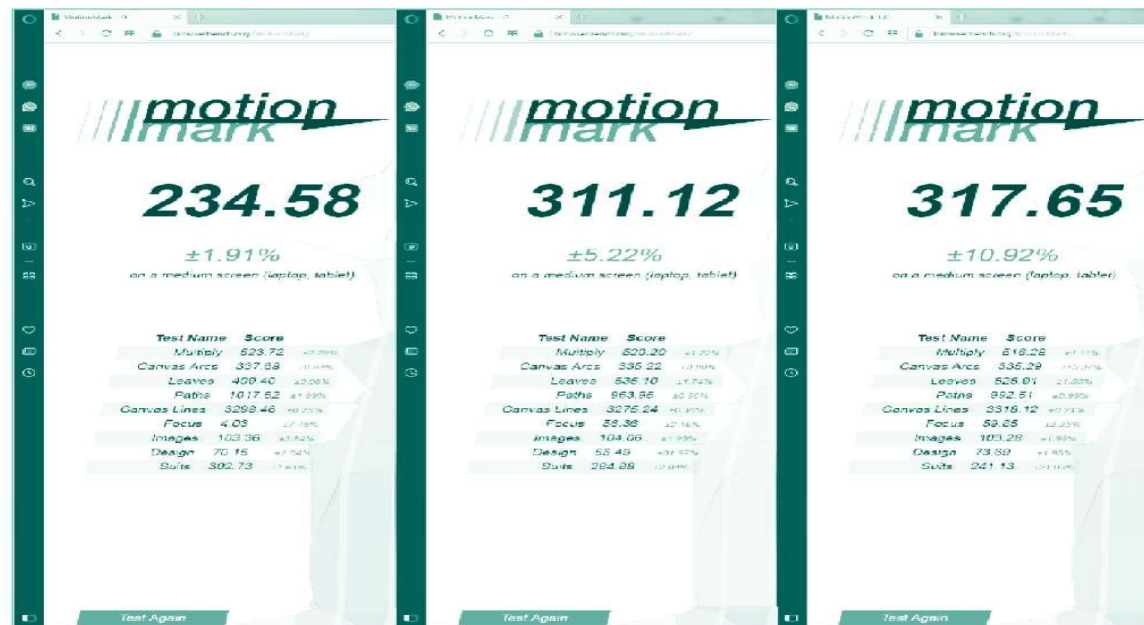


Рис. 1.18. Результаты трех вариантов тестирования скорости работы браузера Опера с помощью MotionMark

# Выводы

45

1. Проанализировав данные табл. 1.3, можно сказать, что самый быстрый запуск у браузера Google Chrome, второе место занимает Mozilla Firefox, а Opera запускается немного медленней, чем два предыдущих браузера.

2. Проанализировав результаты тестирования скорости работы с помощью теста Speedometer 2.0, можно сказать, что самую высокую скорость работы продемонстрировал браузер Google Chrome, скорость браузера Opera на 10,1%, а Mozilla Firefox – на 23,5% ниже.

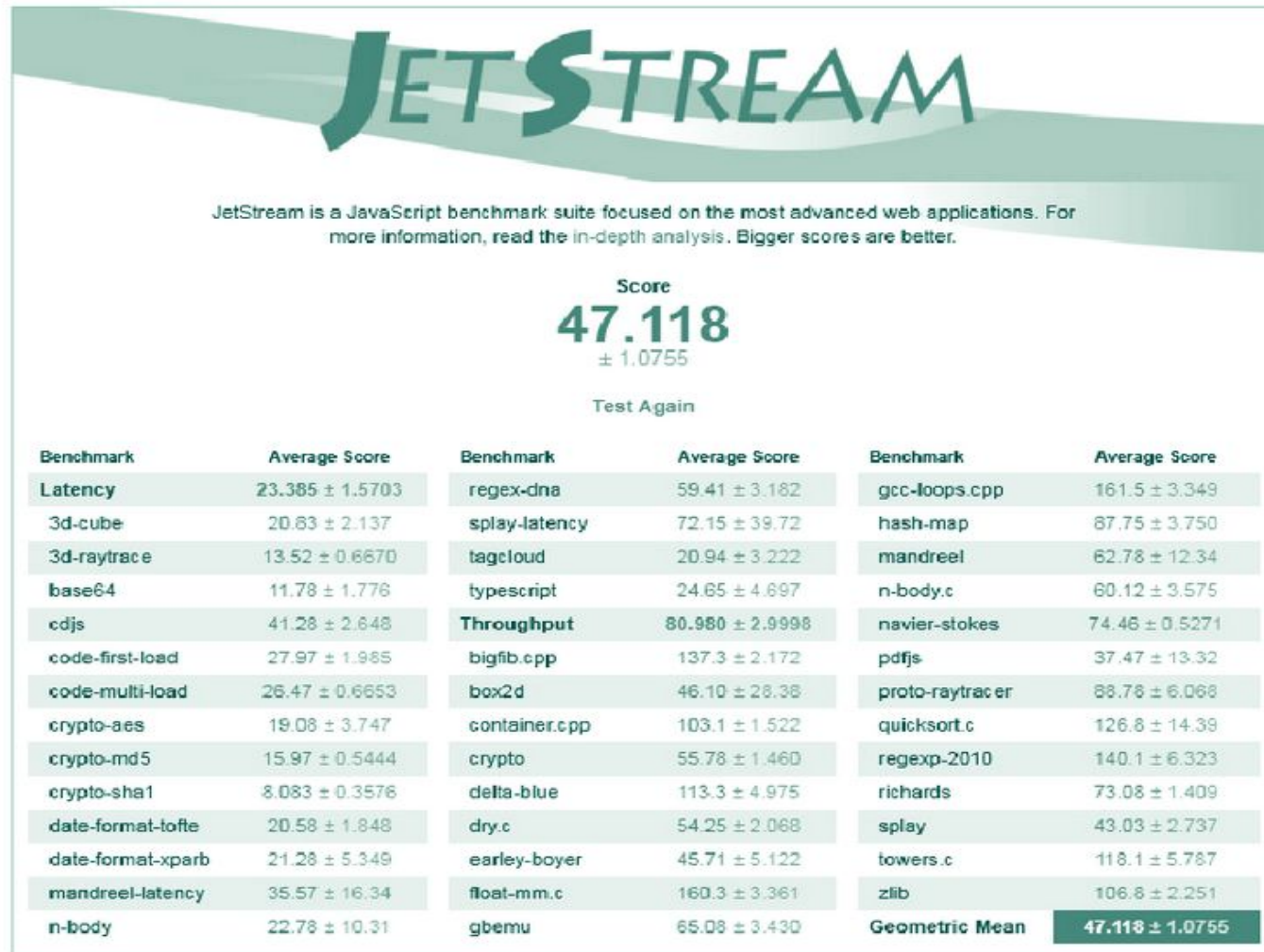
3. Оценка скорости работы браузеров с Java Script с помощью теста JetStream, показала, что самую высокую производительность при обработке Java Script продемонстрировал браузер Mozilla Firefox, средние показатели тестирования у браузеров Google Chrome и Opera – соответственно на 6,7% и 8,8% ниже.

4. Объективно оценить производительность браузеров при обработке анимации и графики на компьютере пользователя с помощью теста MotionMark не удалось.

5. В ходе сравнительного анализа скорости работы браузеров нельзя сделать однозначный вывод о том, что какой-то из браузеров является самым быстрым, так как существуют браузеры, которые быстрее загружаются, есть те, которые имеют более высокую скорость загрузки интернет-страниц, но скорость обработки Java Script по результатам тестирования у них ниже.

# Результаты теста JetStream

46



*a*

Рис. 1.16. Результаты тестирования с помощью теста JetStream браузеров Опера (а), Mozilla Firefox (б) и Google Chrome (в)

# Использование критериев оценки для сравнения

47

**Шаг 1.** Комиссия экспертов формирует таблицу критериев оценки, являющихся самыми важными для потребителя.

**Шаг 2.** Та же комиссия для каждого критерия определяет методику оценки выполнения критерия таким образом, чтобы "обезразмерить" исходные показатели (шкалирование)

**Шаг 3.** Для каждого критерия эксперты выставляют коэффициенты значимости критерия для оценки продукта. Коэффициенты распределяются на отрезке от 0 до 1.

**Шаг 4.** Производится расчет аддитивной суммы интегральной оценки для каждого сравниваемого продукта по следующей формуле:

$$O_{3n} = \sum_{i=1}^n (Z_i \cdot K_i) ,$$

**Шаг 5.** Значения интегральных оценок для каждого сравниваемого продукта ранжируются по убыванию

# Пример сравнительного анализа языков программирования

Показатель	Вес	PHP		HTML		CSS		JavaScript	
		Оценка	Оценка с учетом веса	Оценка	Оценка с учетом веса	Оценка	Оценка с учетом веса	Оценка	Оценка с учетом веса
Скорость обработки	0,1	6	0,6	7	0,7	8	0,8	6	0,6
Читабельность	0,1	7	0,7	6	0,6	6	0,6	7	0,7
Простота	0,1	6	0,6	8	0,8	3	0,3	5	0,6
GUI (графический интерфейс пользователя)	0,05	7	0,35	6	0,3	4	0,2	8	0,4
Кросс-платформенность	0,1	9	0,9	9	0,9	6	0,6	9	0,9
Обучаемость	0,05	6	0,3	7	0,35	5	0,25	6	0,3
Библиотеки	0,1	8	0,8	7	0,7	8	0,8	7	0,7
Надёжность	0,1	7	0,7	6	0,6	4	0,4	7	0,7
Интеграция	0,2	6	1,2	8	1,6	6	1,2	8	1,6
Модульное тестирование	0,1	8	0,8	7	0,7	6	0,6	7	0,7
<b>Итог</b>	<b>1</b>	<b>70</b>	<b>6,95</b>	<b>71</b>	<b>7,25</b>	<b>56</b>	<b>5,75</b>	<b>70</b>	<b>7,2</b>