

**МИНОБРНАУКИ РОССИИ**

**федеральное государственное бюджетное образовательное учреждение**

**высшего образования**

**«Московский государственный технологический университет «СТАНКИН»**

**(ФГБОУ ВО МГТУ «СТАНКИН»)**

|  |  |
| --- | --- |
| ***ИНСТИТУТ***  информационных систем и технологий | **Кафедра**  информационных систем |

**КУРСОВОЙ ПРОЕКТ**

по дисциплине «**Проектирование информационных систем**»

на тему: «Информационная система аудита безопасности беспроводной локальной вычислительной сети»

|  |  |  |
| --- | --- | --- |
| **Студент** группа ИДБ–15–12 |  | **Королецкий П. В.** |
|  | подпись |  |
| **Руководитель**  старший преподаватель |  | **Овчинников П.Е.** |
|  | подпись |  |

Москва 2018 г.

**ОГЛАВЛЕНИЕ**

[ВВЕДЕНИЕ 3](#_Toc530946714)

[ГЛАВА 1. ФУНКЦИОНАЛЬНАЯ МОДЕЛЬ (IDEF0) 4](#_Toc530946715)

[ГЛАВА 2. МОДЕЛЬ ПОТОКОВ ДАННЫХ (DFD) 10](#_Toc530946716)

[ГЛАВА 3. МОДЕЛЬ «СУЩНОСТЬ-СВЯЗЬ» (ER MODEL) 16](#_Toc530946717)

[ГЛАВА 4. ОЦЕНКА СЛОЖНОСТИ И ТРУДОЗАТРАТ ПО РАЗРАБОТКЕ ИС 18](#_Toc530946718)

[ЗАКЛЮЧЕНИЕ 20](#_Toc530946719)

[СПИСОК ЛИТЕРАТУРЫ 21](#_Toc530946720)

**ВВЕДЕНИЕ**

Глобально и быстро растет потребность в сетях беспроводного доступа стандарта Wi-Fi.

Преимущества беспроводного доступа очевидны, а технология Wi-Fi изначально стала стандартом, которого придерживаются производители мобильных устройств. Огромное количество университетов по всему миру предоставляют доступ в Интернет для своих студентов и преподавателей при помощи технологии Wi-Fi.

Одним из важнейших критериев будущей беспроводной локальной вычислительной сети (БЛВС) является ее безопасность. В настоящее время существует множество потенциальных угроз, уберечься от которых необходимо на этапе проектирования, выбрав актуальные методы и средства защиты, стандарты и технологии.

Целью данной курсовой работы является визуализация процесса создания проекта по обеспечению безопасности БЛВС. Функциональные модели строятся с точки зрения проектировщика системы безопасности БЛВС.

Исходя из того, что темой курсового проекта выбрана «ИС аудита безопасности БЛВС», в процессе создания диаграмм будет сделан акцент именно на конкретных шагах проведения аудита безопасности БЛВС, в том числе будет изучена информация по проведению испытания на проникновение (пентеста).

Процесс пентеста включает в себя активный анализ системы на наличие потенциальных уязвимостей, которые могут спровоцировать некорректную работу целевой системы, либо полный отказ в обслуживании.

**ГЛАВА 1. ФУНКЦИОНАЛЬНАЯ МОДЕЛЬ (IDEF0)**

Стандарт IDEF0 был разработан в 1981 году в США департаментом Военно-воздушных сил для автоматизации промышленных предприятий. В процессе разработки программного обеспечения разработчики столкнулись с необходимостью разработки новых методов анализа бизнес-процессов. В результате появилась методология функционального моделирования IDEF0, в которой для анализа применяются специальные нотации IDEF0.

Нотацией называется формат описания бизнес-процесса, представляющий собой совокупность графических объектов, используемых при моделировании, а также правил моделирования. По сути, нотации – это особый графический язык, который позволяет описывать работу компании, наглядно демонстрировать взаимодействие между различными подразделениями, т.е. описывать бизнес-процессы. Нотации могут применяться для процессного или функционального моделирования (рис. 1.1).

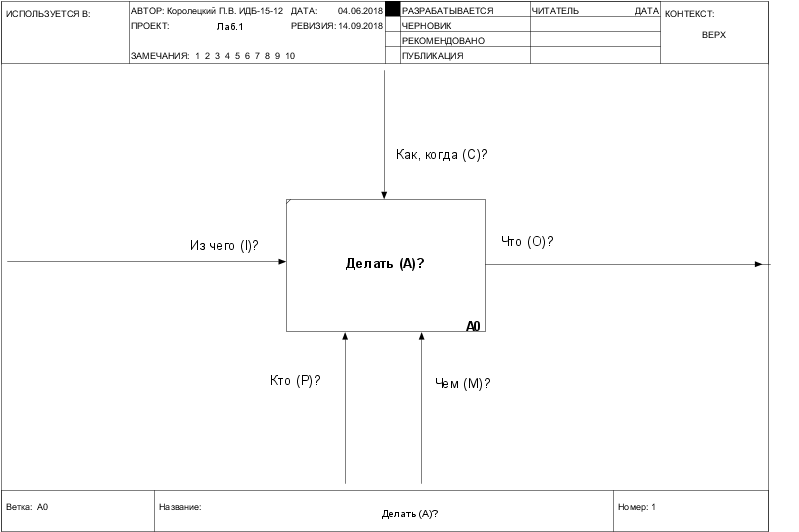


Рис. 1.1. IDEF0 ICOM (Input, Controls, Outputs and Mechanisms) нотация

Функциональная модель IDEF0 представляет собой набор блоков, каждый из которых представляет собой «черный ящик» со входами и выходами, управлением и механизмами, которые детализируются (декомпозируются) до необходимого уровня. Наиболее важная функция расположена в верхнем левом углу. А соединяются функции между собой при помощи стрелок и описаний функциональных блоков. При этом каждый вид стрелки или активности имеет собственное значение. Данная модель позволяет описать все основные виды процессов, как административные, так и организационные [1].

В функциональной модели данные делятся на 4 типа: внешние входные информационные потоки, внешние выходные информационные потоки, внешние управляющие потоки и механизмы.

В данном курсовом проекте внешними входными информационными потоками для модели являются:

1. Задание на ВКР.
2. Прототип БЛВС.

Выходными информационными потоками являются:

1. Проект по обеспечению безопасности БЛВС.
2. Усовершенствованный прототип БЛВС.

Внешними управляющими потоками являются:

1. Семейство стандартов IEEE 802.11.
2. Требования СОРМ.
3. ГОСТы.
4. ФЗ №149.
5. ПП №758, ПП №801.
6. Требования к оформлению.
7. Kali Linux Official Documentation.

Основными механизмами являются:

1. Проектировщик системы безопасности БЛВС.
2. Аналитик.
3. Дизайнер.
4. Kali Linux Notebook.
5. Администратор Linux.
6. Эксперт в области безопасности сетей.
7. Wi-Fi адаптер.

Далее представлены отдельные диаграммы функциональной модели (рис. 1.2-1.8).

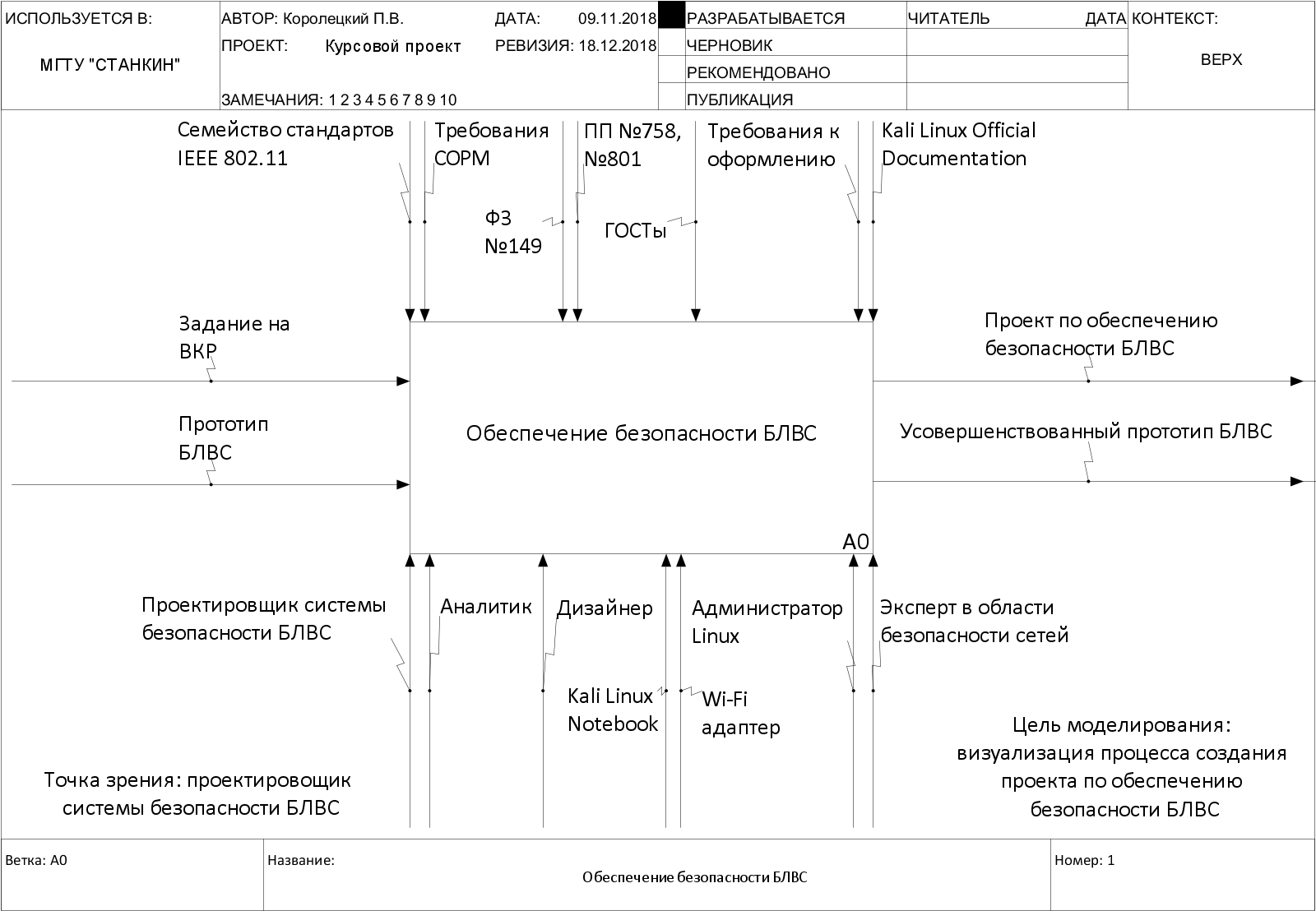


Рис. 1.2. Контекстная диаграмма

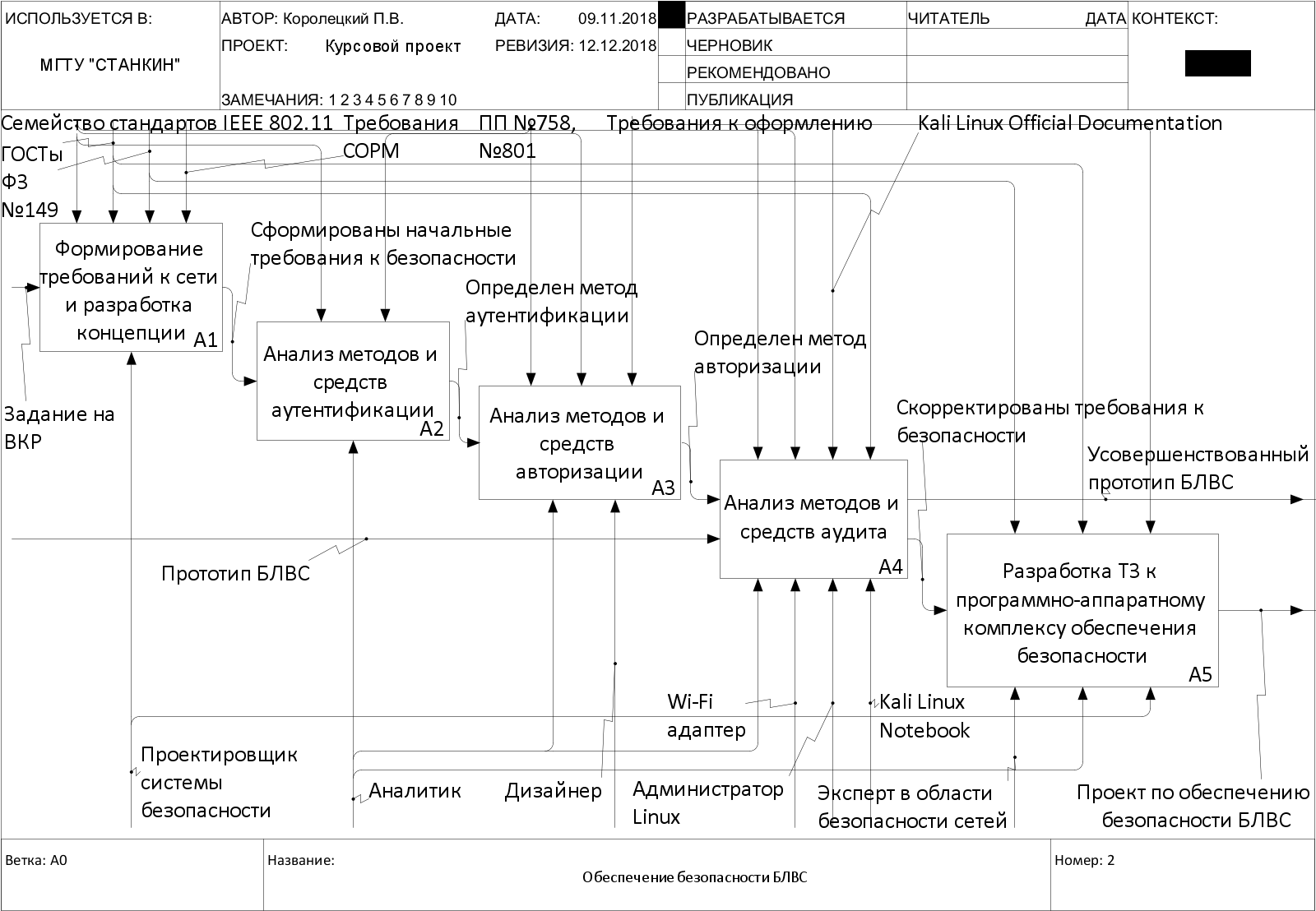


Рис. 1.3. Диаграмма процесса создания проекта по обеспечению безопасности БЛВС

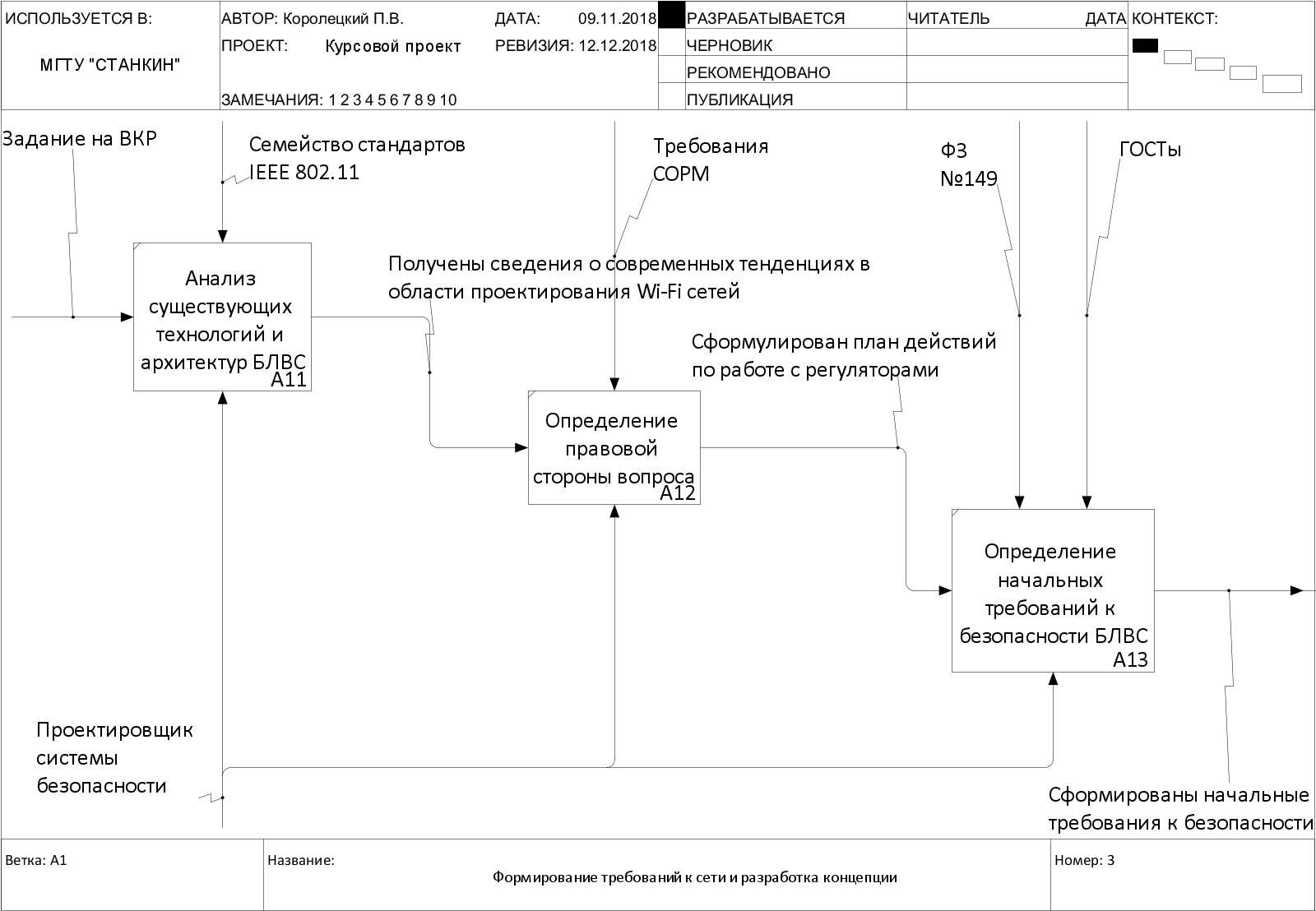


Рис. 1.4. Диаграмма процесса формирования требований к сети и разработки концепции

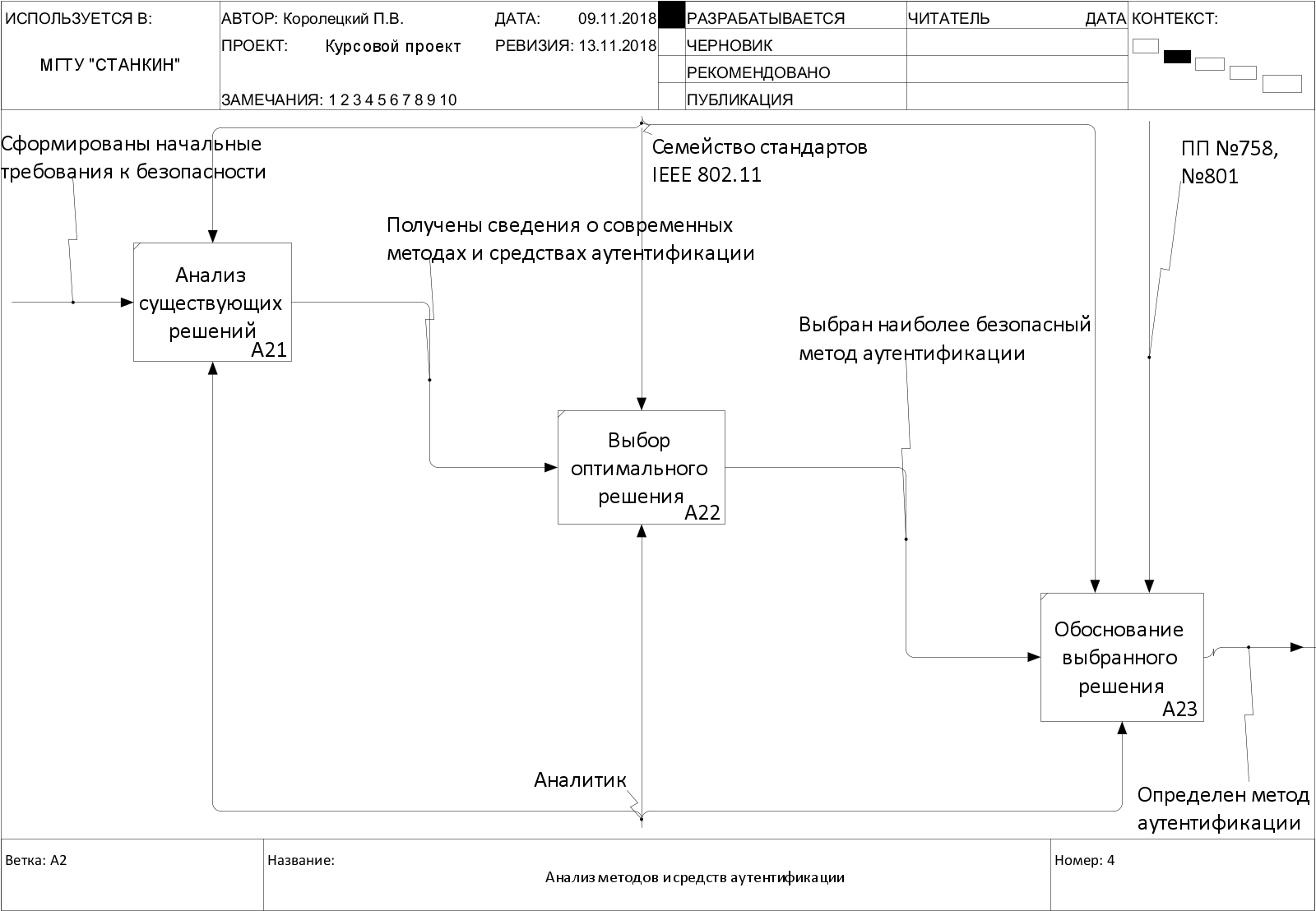


Рис. 1.5. Диаграмма процесса анализа методов и средств аутентификации

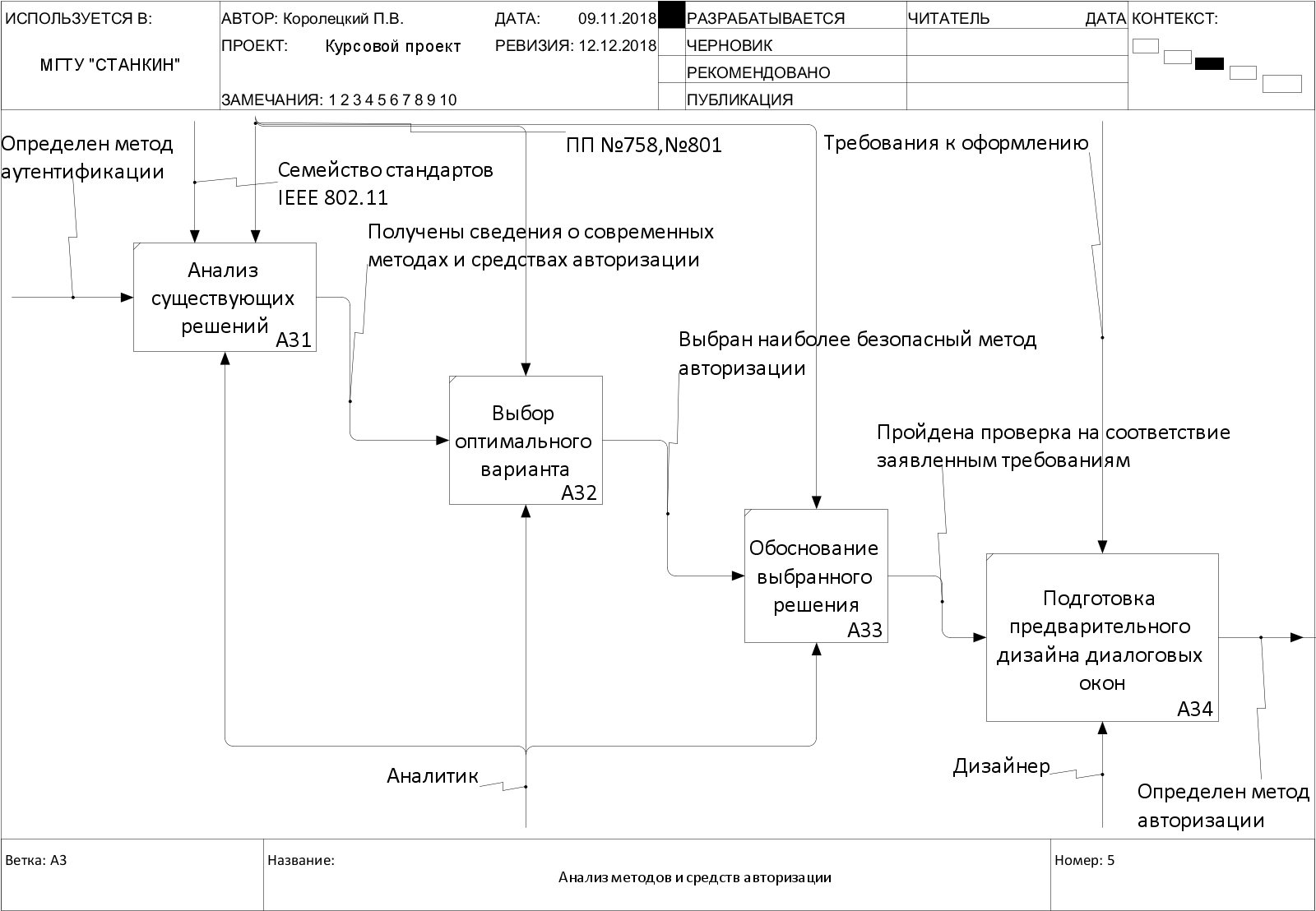


Рис. 1.6. Диаграмма процесса анализа методов и средств авторизации

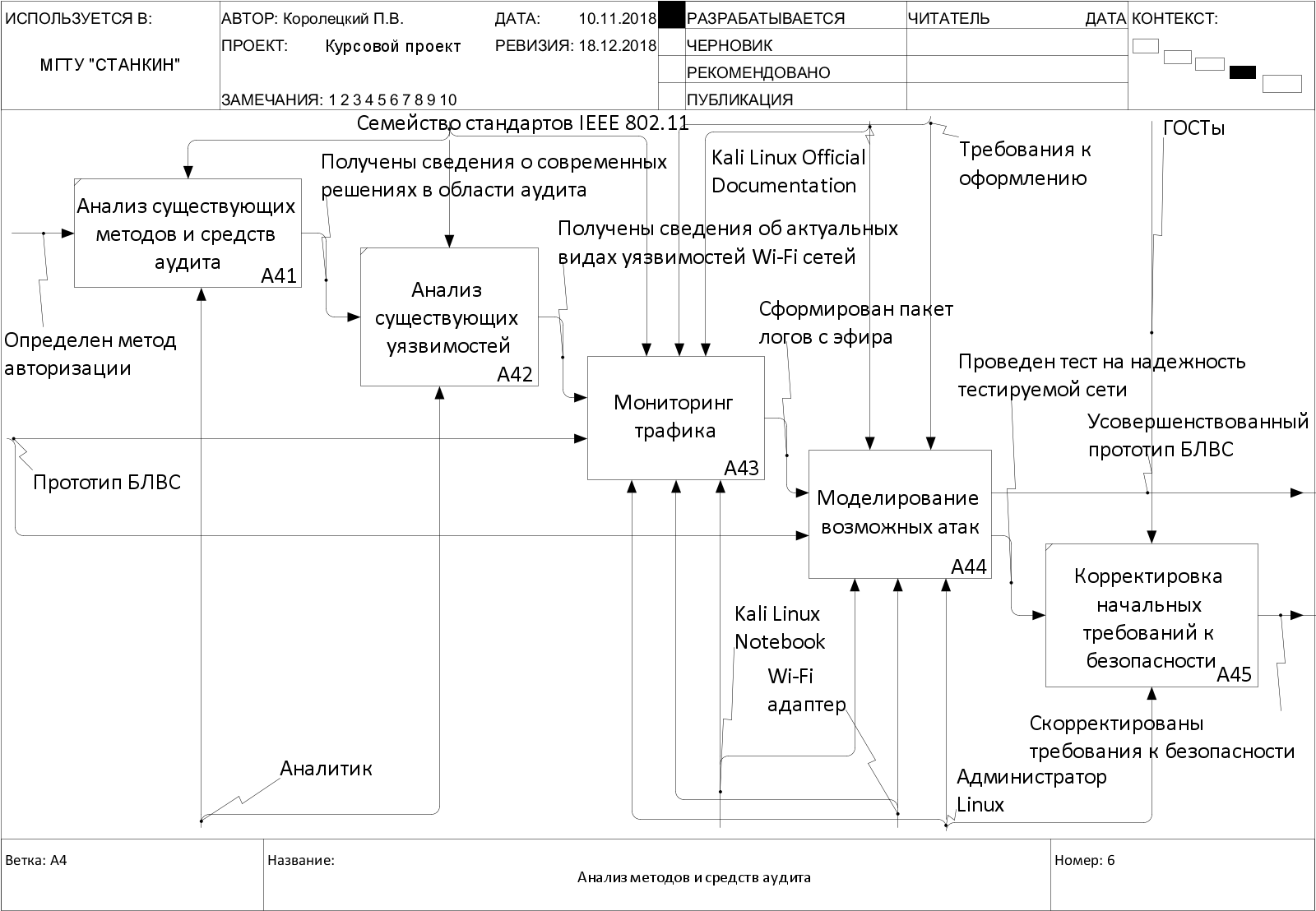


Рис. 1.7. Диаграмма процесса анализа методов и средств аудита

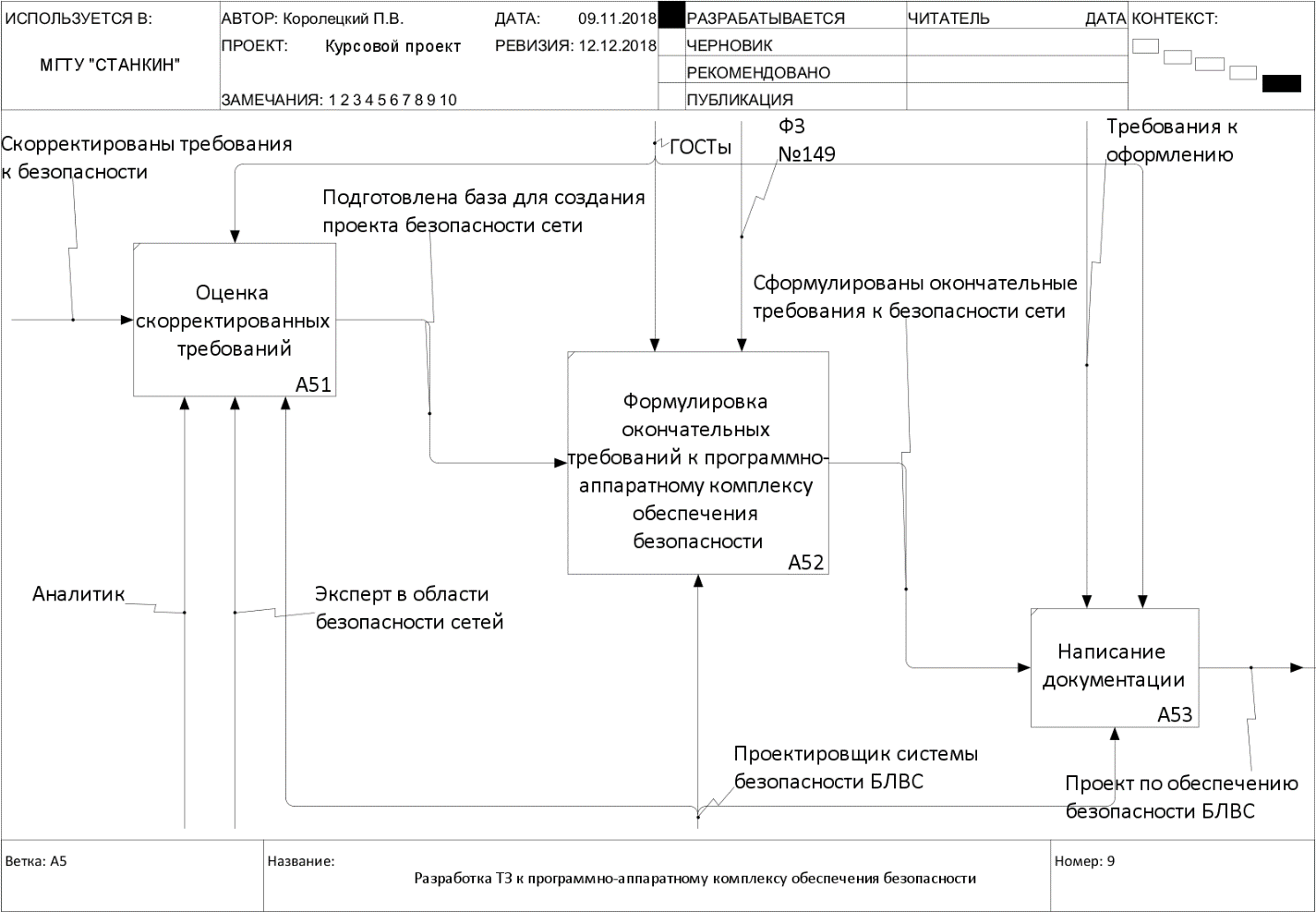


Рис. 1.8. Диаграмма процесса разработки ТЗ к программно-аппаратному комплексу обеспечения безопасности

**Определение числовых показателей для поставленной цели моделирования**

Текущая модель позволяет определить процессы, которые будут оптимизированы. Автоматизированная система позволит ускорить процесс идентификации вектора атаки на сеть и предпринять соответствующие меры.

Подсистема аудита безопасности сети является ключевым механизмом автоматизации и подлежит исследованию, т.к. она сокращает затраты на выполнение процесса отслеживания статуса сети и снижает время оповещения ответственных сотрудников о различных злонамеренных вторжениях в сеть.

**ГЛАВА 2. МОДЕЛЬ ПОТОКОВ ДАННЫХ (DFD)**

DFD – это нотация, предназначенная для моделирования информационных систем с точки зрения хранения, обработки и передачи данных. Нотации DFD – удобный инструмент для создания нерегламентированных диаграмм, которые можно сделать быстро и с максимумом свободы.

Применяется этот вид нотации в случае, когда требуется описание системы как хранилища данных. Т.е. нотация должна наглядно отвечать на вопросы:

1. Из чего состоит информационная система?
2. Что нужно, чтобы обработать информацию?

Непосредственно DFD нотация состоит из следующих элементов:

* Процесс (англ. Process), т.е. функция или последовательность действий, которые нужно предпринять, чтобы данные были обработаны. Это может быть создание заказа, регистрация клиента и т.д. Здесь нет строгой системы требований, как, например, в IDEF0, где нотации имеют жестко определенный синтаксис, так как они могут быть исполняемыми. Но все же определенных правил стоит придерживаться, чтобы не вносить путаницу при чтении DFD другими людьми.
* Внешние сущности (англ. External Entity). Это любые объекты, которые не входят в саму систему, но являются для нее источником информации либо получателями какой-либо информации из системы после обработки данных. Это может быть человек, внешняя система, какие-либо носители информации и хранилища данных.
* Хранилище данных (англ. Data store). Внутреннее хранилище данных для процессов в системе. Поступившие данные перед обработкой и результат после обработки, а также промежуточные значения должны где-то храниться. Это и есть базы данных, таблицы или любой другой вариант организации и хранения данных. Здесь будут храниться данные о клиентах, заявки клиентов, расходные накладные и любые другие данные, которые поступили в систему или являются результатом обработки процессов.
* Поток данных (англ. Data flow). В нотации отображается в виде стрелок, которые показывают, какая информация входит, а какая исходит из того или иного блока на диаграмме [2].

В рамках данной курсовой работы была произведена декомпозиция блоков A43 и A44 в DFD (рис. 2.1-2.2); пропущены блоки воздействия типа mechanism. При этом полагается, что все формы и модули, изложенные в DFD, управляются Администратором Linux при помощи Kali. Далее перечислены используемые типы хранилищ:

* Среда: Эфир — это беспроводная среда передачи данных.
* БД: Уязвимости безопасности — это несколько таблиц, содержащих информацию о современных уязвимостях в области безопасности БЛВС, а также ссылки на конкретные утилиты, позволяющие бороться с уязвимостями.
* Репозиторий: Aircrack-ng — это набор утилит, используемых при тестировании беспроводных сетей.
* Лог-файл точки доступа — это файл, содержащий структурированную по времени информацию о результатах подключений клиентов к ТД и много другой полезной информации.
* Лог-файл программы — это файл, содержащий структурированную по времени информацию о результатах анализа эфира и соответствующего определенной ТД трафика.
* Документ: Результаты — это документ, который будет сформирован после аудита; данный документ будет подспорьем при корректировке первоначальных требований к безопасности.
* Файл: Настройки точки доступа — это конфигурационный файл ТД, который исходя от прошивки устройства определяет потенциал в области защиты сети.

Также были определены основные средства автоматизации, а именно:

* Конфигурация технических средств: активные точки доступа (хранящие лог-файлы), ноутбук (рабочая станция с ОС Kali Linux на борту, ассоциирующаяся с точкой доступа для проведения пентеста), ПК (используется при написании документации), широкодиапазонный беспроводной USB-адаптер Wi-Fi от ALFA Network (используется для проведения пентеста).
* Конфигурация программных средств: ОС Kali Linux – GNU/Linux-дистрибутив, возникший как результат слияния WHAX и Auditor Security Collection. Проект создали Мати Ахарони (Mati Aharoni) и Макс Мозер (Max Moser). Предназначен прежде всего для проведения тестов на безопасность. Наследник развивавшегося до 2013 года на базе Knoppix дистрибутива BackTrack.
* Допустимые виды хранилищ и их размещения: с точки зрения архитектуры, использовать необходимо классическое нормализованное хранилище данных, данные в котором находятся в предметно ориентированных таблицах третьей нормальной формы. Хранилище будет локальным и может располагаться на ПК, используемом при написании документации.

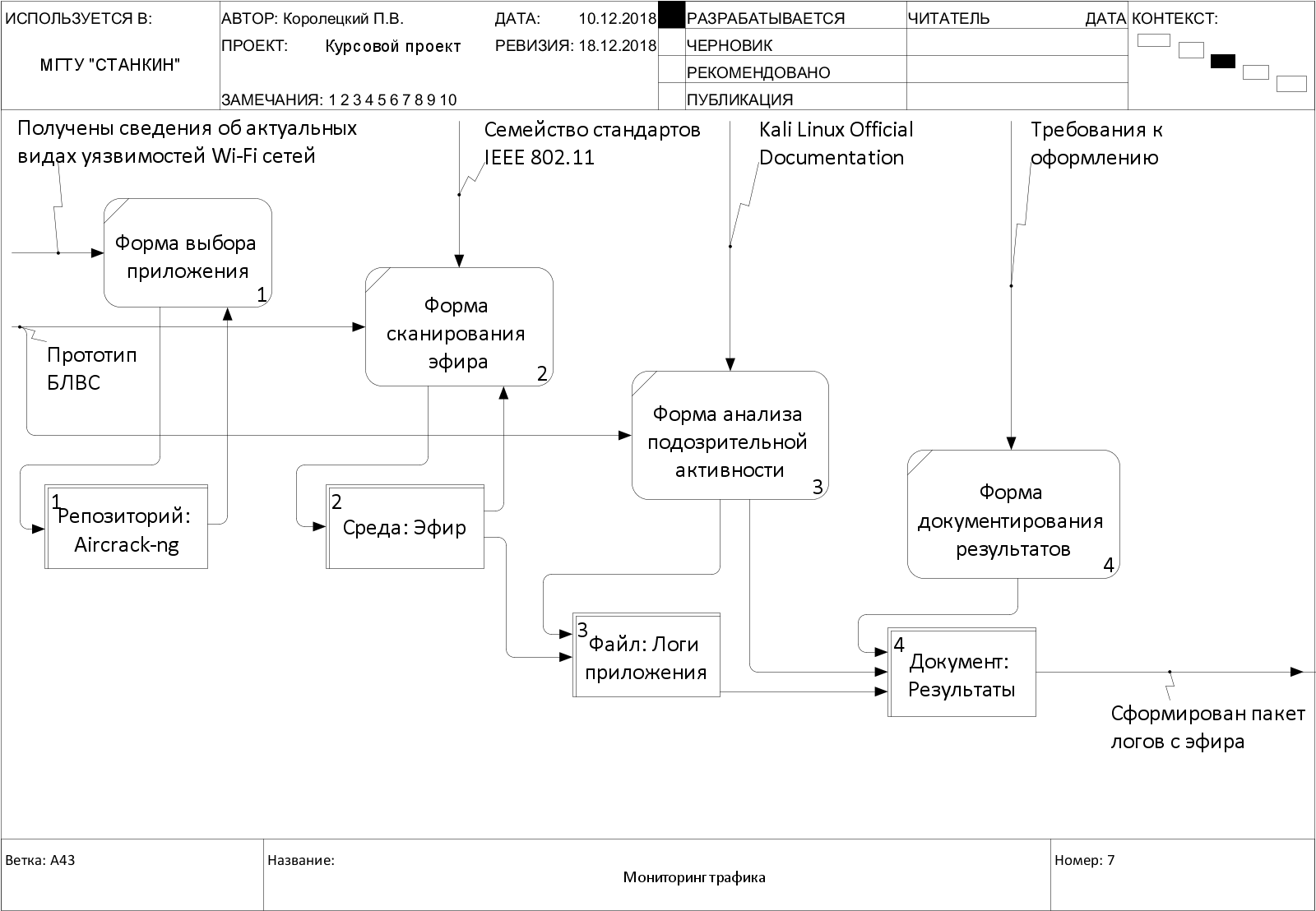


Рис. 2.1. Мониторинг трафика

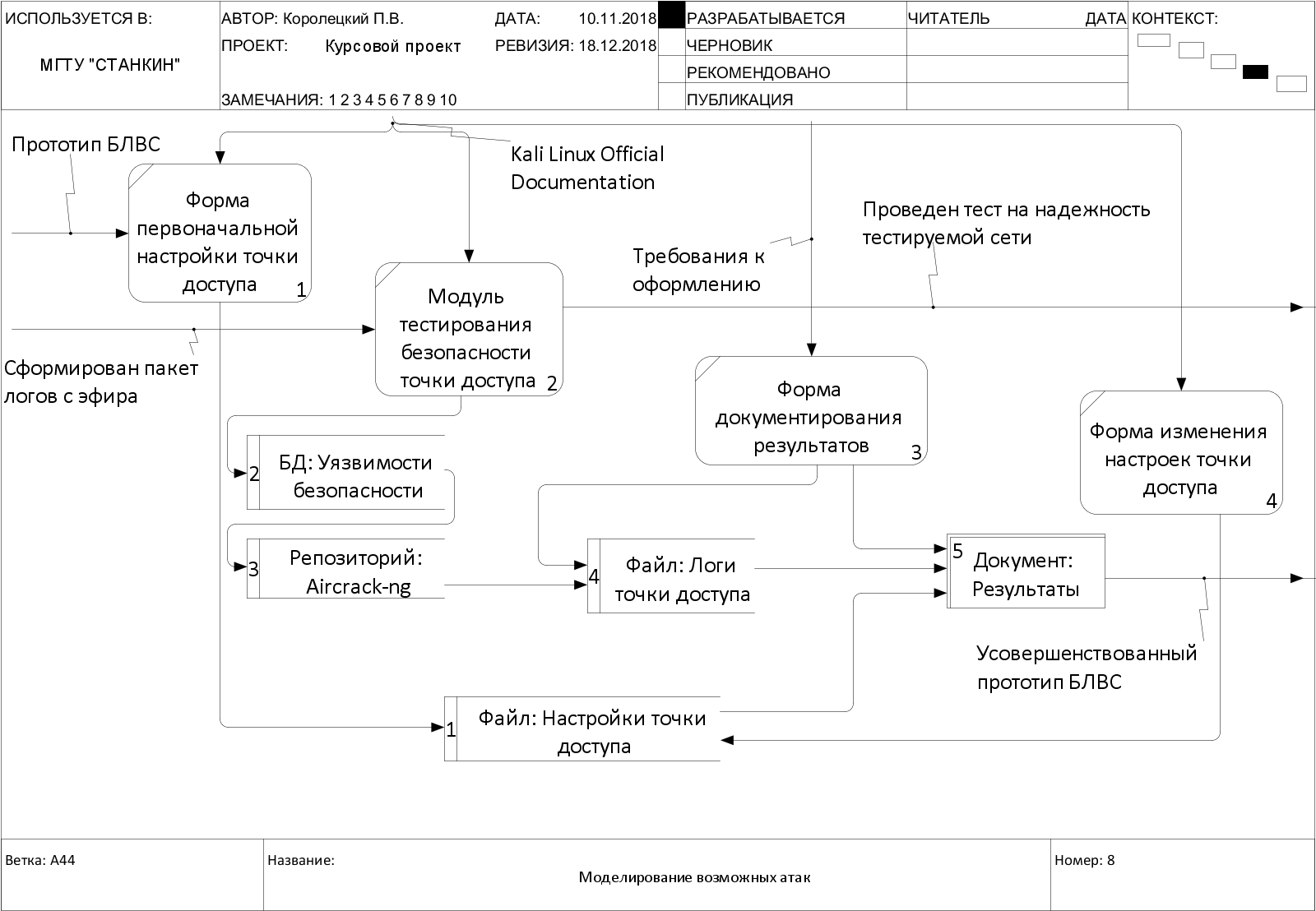


Рис. 2.2. Моделирование возможных атак

**Определение числовых показателей для цели потенциального проекта автоматизации**

Паттерн "автоматизация снижает время обслуживания (ожидания)".

Данный паттерн прямо следует из понятия "мура" (неравномерность) и связан, как правило, с совершенствованием процессов диспетчерского управления, т.е. с качеством распределения потоков поступающих заданий на выполнение определенных операций по исполнителям (рабочим центрам).

В проекте по обеспечению безопасности данный паттерн ложится на модуль "Мониторинг трафика", который оповещает о критических нарушениях работы БЛВС. Автоматизация данного модуля позволит снизить время на оповещения о различных злонамеренных вторжениях в сеть с 1 часа до 30 секунд, что позволит быстрее принимать решения ответственным за безопасность сети сотрудникам.

**Эффект от проекта**

Таблица 1.

|  |  |
| --- | --- |
| Рассматриваемый период – 1 месяц (20 рабочих дней)  Сотрудник - 1 | |
| **С использованием АС** | **Ручной труд** |
| Сотрудник проверяет 50 точек доступа (ТД) в день | Сотрудник проверяет 5 точек доступа (ТД) в день |
| **Расчет экономии времени от реализации проекта для блоков А43 и А44** | |
| Время на проверку 1 ТД у сотрудника:   1. 480/50 = 9,6 мин (9 мин. 36 с.)   Сотрудник тратит 9 мин. 36 с. мин на 1 ТД.   1. 50\*20 = 1000 ТД.   За месяц сотрудник проверяет 1000 ТД. | Время на проверку 1 ТД у сотрудника:   1. 480/5 = 96 мин   Сотрудник тратит 96 мин на 1 ТД.   1. 5\*20 = 100 ТД.   За месяц сотрудник проверяет 100 ТД. |

**ГЛАВА 3. ДИАГРАММЫ КЛАССОВ (ERD)**

ER-модель (от англ. entity-relationship model, модель «сущность-связь») – модель данных, позволяющая описывать концептуальные схемы предметной области. ER-модель представляет собой формальную конструкцию, которая сама по себе не предписывает никаких графических средств её визуализации. В качестве стандартной графической нотации, с помощью которой можно визуализировать ER-модель, была предложена диаграмма «сущность-связь» (англ. entity-relationship diagram, ERD, ER-диаграмма) [3].

В данной главе используется UML – унифицированный язык моделирования (Unified Modeling Language) – это система обозначений, которую можно применять для объектно-ориентированного анализа и проектирования. Его можно использовать для визуализации, спецификации, конструирования и документирования программных систем [4]. Важно уточнить, что будет использоваться диаграмма классов UML, которая позволяет обозначать отношения между классами и их экземплярами. Для взаимосвязи классов будет использоваться связь «обобщение».

Обобщение (Generalization) показывает, что один из двух связанных классов (подтип) является частной формой другого (надтипа), который называется обобщением первого. На практике это означает, что любой экземпляр подтипа является также экземпляром надтипа. Обобщение также известно, как наследование или «is a» взаимосвязь (или отношение «является») [5].

В рамках курсового проекта были построены ERD (диаграммы классов без атрибутов) для всех потоков (рис. 3.1), ролей (рис. 3.2) и модулей (рис. 3.3).

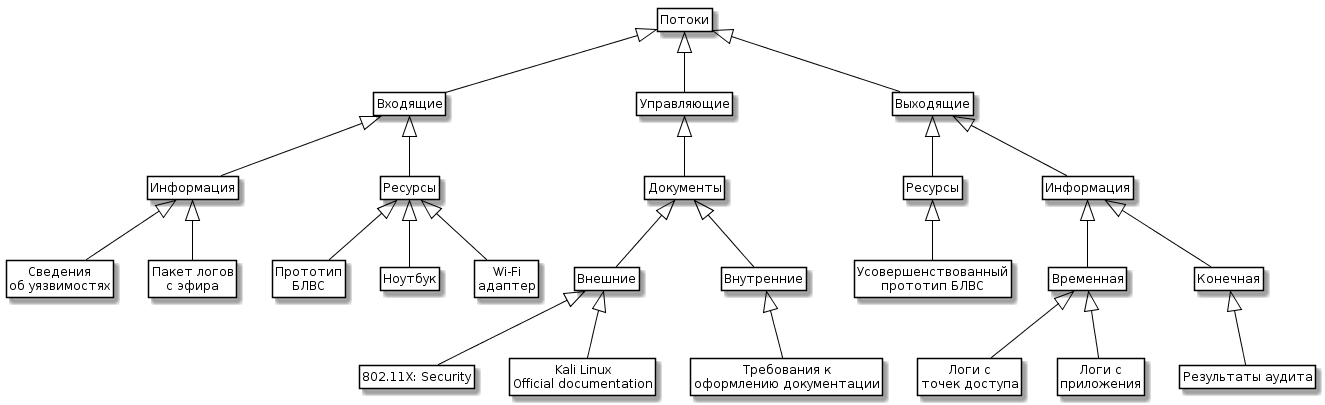


Рис. 3.1. Диаграмма ER для всех потоков

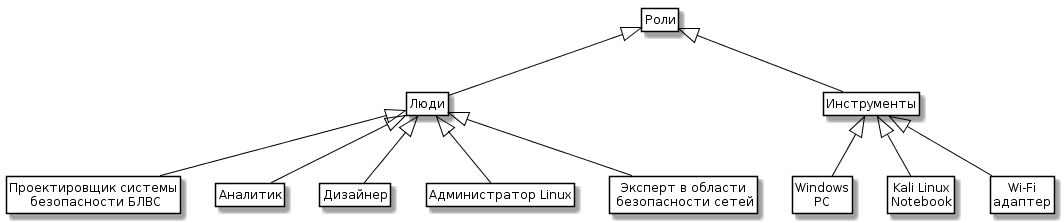


Рис. 3.2. Диаграмма ER для всех ролей

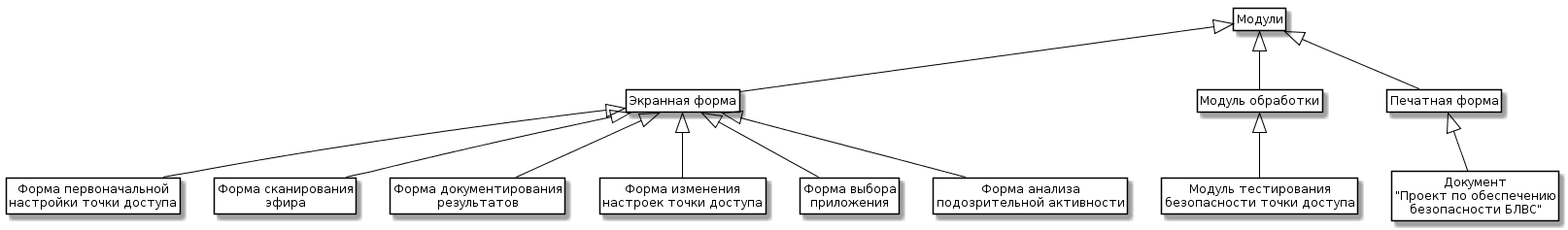


Рис. 3.3. Диаграмма ER для всех модулей

**ГЛАВА 4. ОЦЕНКА СЛОЖНОСТИ И ТРУДОЗАТРАТ ПО РАЗРАБОТКЕ ИС**

Оценка трудоемкости должна быть вероятностным утверждением. Это означает, что для нее существует некоторое распределение вероятности, которое может быть очень широким, если неопределенность высокая, или достаточно узким, если неопределенность низкая.

Использование собственного опыта или опыта коллег, полученного в похожих проектах, это наиболее прагматичный подход, который позволяет получить достаточно реалистичные оценки трудоемкости и срока реализации программного проекта, быстро и без больших затрат.

Если собственный опыт аналогичных проектов отсутствует, а коллеги-эксперты недоступны, то необходимо использовать формальные методики, основанные на обобщенном отраслевом опыте. Среди них наибольшее распространение получили два подхода:

* FPA IFPUG – метод функциональных точек.
* Метод COCOMO II, Constructive Cost Model.

Методика COCOMO позволяет оценить трудоемкость и время разработки программного продукта. Впервые была опубликована Бари Боэмом в 1981 году в виде результат анализа 63 проектов компании «TRW Aerospace». В 1997 методика была усовершенствована и получила название COCOMO II. Калибровка параметров производилась по 161 проекту разработки. В модели используется формула регрессии с параметрами, определяемыми на основе отраслевых данных и характеристик конкретного проекта [6].

Метод FPA разработан на основе опыта реализации множества проектов создания ПО и поддерживается международной организацией IFPUG (International Function Point User Group). Согласно данной методике трудоемкость вычисляется на основе функциональности разрабатываемой системы, которая, в свою очередь, определяется на основе выявления функциональных типов – логических групп взаимосвязанных данных, используемых и поддерживаемых приложением, а также элементарных процессов, связанных с вводом и выводом информации [7].

При расчетах сложности разработки методом FPA/IFPUG и трудозатрат на разработку «с нуля» методом COCOMO II для данного курсового проекта были получены следующие значения: сложность требуемых для создания информационной системы программных средств была оценена в 88 выровненных функциональных точек (DFP), а объем программного кода на языках программирования высокого уровня – в 4418 строк кода (FPA/IFPUG). Расчеты, выполненные методом COCOMO II, позволяют оценить общие трудозатраты проекта разработки программных средств в 15 человеко-месяцев, а ожидаемую продолжительность проекта – в 8 месяцев.

**ЗАКЛЮЧЕНИЕ**

Беспроводные технологии с каждым годом становятся все более незаменимыми в современной жизни человека. Технология Wi-Fi обеспечивает не привязанную к отдельным помещениям сеть и доступ в Интернет, она дает пользователям возможность перемещаться по территории предприятия или организации, оставаясь подключенными к сети. Одним из важнейших критериев будущей Wi-Fi сети является ее безопасность. Именно поэтому необходимо использовать актуальные методы и средства защиты, чтобы уберечься от потенциальных угроз.

В главе 1 были представлены отдельные диаграммы функциональной модели, описанные по IDEF0 ICOM нотации. Важно отметить, что уже на этом этапе благодаря декомпозиции процесс создания проекта по обеспечению безопасности стал более прозрачным, чем был в начале пути.

В главе 2 была произведена декомпозиция блока A43 в DFD. Помимо этого, были перечислены используемые типы хранилищ, а также определены основные средства автоматизации. Именно здесь был сделан упор на аудит безопасности БЛВС, а также виден результат анализа предметной области.

В главе 3 было описано завершение процесса идентификации через представление ER-диаграмм для всех потоков, ролей и модулей.

В главе 4 был показан результат расчетов сложности разработки методом FPA/IFPUG и трудозатрат на разработку «с нуля» методом COCOMO II.

Изученная информация и созданные диаграммы IDEF0 будут использованы в выпускной квалификационной работе.

**СПИСОК ЛИТЕРАТУРЫ**

1. Сайт «Habrahabr. Знакомство с нотацией IDEF0» [Электронный ресурс] – Режим доступа: https://habr.com/company/trinion/blog/322832, свободный. Дата обращения: 25.11.2018 г.
2. Сайт «Habrahabr. Что такое DFD» [Электронный ресурс] – Режим доступа: https://habr.com/company/trinion/blog/340064, свободный. Дата обращения: 25.11.2018 г.
3. Сайт «Википедия. ER-модель» [Электронный ресурс] – Режим доступа: https://ru.wikipedia.org/wiki/ER-модель, свободный. Дата обращения: 25.11.2018 г.
4. Сайт «Программирование на Си и C++. Про UML» [Электронный ресурс] – Режим доступа: https://prog-cpp.ru/uml-classes, свободный. Дата обращения: 25.11.2018 г.
5. Сайт «Википедия. Диаграмма классов» [Электронный ресурс] – Режим доступа: https://ru.wikipedia.org/wiki/Диаграмма\_классов#Обобщение, свободный. Дата обращения: 25.11.2018 г.
6. Сайт «СIT Forum. Лекции по управлению программными проектами» [Электронный ресурс] – Режим доступа: http://citforum.ru/SE/project/arkhipenkov\_lectures/13.shtml, свободный. Дата обращения: 25.11.2018 г.
7. Сайт «Multiurok. Методы оценки размера программных продуктов» [Электронный ресурс] – Режим доступа: https://multiurok.ru/index.php/files/mietody-otsienki-razmiera-proghrammnykh-produktov.html, свободный. Дата обращения: 25.11.2018 г.