**БРЯНСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

**ЗАДАНИЕ**

**на дипломную работу**

**Студенту** **\_\_\_\_\_\_\_\_\_\_\_\_\_\_Егоренкову*\_\_*\_\_\_\_\_\_\_\_** **группы** ***\_\_08-ПО1\_\_\_\_\_\_\_\_\_***

1. **Тема работы:** *Программный учебно-методический комплекс. Модуль адаптивного тестирования*

**Утверждена приказом по БГТУ №** **\_\_861-3\_\_** **от \_\_24 апреля\_\_ 2013г.**

1. **Срок сдачи дипломной работы**  *июнь* 2013г.
2. Рекомендуемые инструментальные средства Android SDK, MySQL \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Содержание папки текстовых документов (перечень разделов)\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Аналитическая часть. 1.1. Общие сведения об электронном образовании, 1.2. Тестирование как средство контроля знаний, 1.3. Методы тестирования, 1.4. Адаптивное тестирование, 1.5. Обзор программных средств, 1.6. Моделирование процессов адаптивного тестирования, 1.7. Постановка задачи дипломной работы. 2. Техническое задание; 2.1. Введение, 2.2. Полное описание, 2.3. Специфические требования

3. Исследовательская часть, 3.1. Исследование алгоритмов адаптивного тестирования, 3.2. Жесткий алгоритм тестирования, 3.3. Мягкий алгоритм тестирования, 3.4. Комбинированный алгоритм, 3.5. Динамическая модель адаптивного тестирования, 3.6. Алгоритм Аванесова, 3.7. Разработка собственного алгоритма тестирования.

4. Конструкторская часть; 4.1. Выбор средств разработки, 4.2. Архитектура программного средства, 4.3. Модель хранения данных, 4.4. Модель потоков данных, 4.5. Проектирование пользовательского интерфейса, 4.6. Алгоритм работы проектируемой системы.

5. Экспериментальная часть; 5.1. Назначение тестирования, 5.2. Тестирование в нормальных условиях, 5.3. Проверка программной системы в экстремальных условиях, 5.4.Проверка программной системы в исключительных ситуациях, 5.5. Основные выводы\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

6. Техническая документация; 6.1. Руководство администратора, 6.2. Руководство пользователя

7. Экономическая часть\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

8. Безопасность и экологичность проекта\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

5. Перечень плакатов с указанием их названия\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Постановка задачи\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Итого \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ листов

6. Особые замечания \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Дата выдачи задания \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2013г.

Заведующий кафедрой \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Подвесовский А.Г.

Руководитель работы \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Азарченков А.А.

Задание принял к исполнению \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2013г.

Студент \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ Евграфова Е.В.

# Аннотация

На основании задания дипломной работы разработано программное средство для распространения фотографий с помощью мобильных устройств на базе операционной системы Android. Основным назначением программного средства является обеспечение возможности любого человека поделиться с другими людьми своими фотографиями, а также возможность увидеть фотографии, сделанные другими людьми.

В аналитической части распространение фотографий рассмотрено с точки зрения распространения информации в целом, изучены существующие методы распространения информации. Проведён обзор и анализ существующих программ-аналогов, выделены их достоинства и недостатки, а также сделан вывод об актуальности создания программного средства. Сформулированы требования к разрабатываемой системе.

В техническом задании определены основание для разработки и назначение разработки. Описаны основные требования к программной системе и этапы ее разработки.

В исследовательской части проведено исследование существующих способов подготовки изображения к передаче его по сети.

В конструкторской части рассмотрена архитектура программного средства, выполнено обоснование выбора языка и средств разработки, проведено моделирование системы, базы данных и интерфейса.

В части технической документации разработаны необходимые документы, сопровождающие программный продукт.

В экспериментальной части описано проведенное тестирование разработанной системы в нормальных условиях, нагрузочное тестирование и тестирование в исключительных ситуациях.

В экономической части дипломной работы приведены расчеты затрат на создание программной системы и проведена оценка эффективности ее внедрения.

В организационной части приводятся рекомендации по охране труда и технике безопасности.

**СОДЕРЖАНИЕ**

[Аннотация 2](#_Toc354407512)

[Введение 6](#_Toc354407513)

[1 Аналитическая часть 8](#_Toc354407514)

[1.1 Общие сведения об электронном образовании 8](#_Toc354407515)

[1.2 Тестирование как средство контроля знаний 9](#_Toc354407516)

[1.3 Методы тестирования 11](#_Toc354407517)

[1.4 Адаптивное тестирование 12](#_Toc354407518)

[1.5 Обзор программных средств 14](#_Toc354407519)

[1.6 Моделирование процессов адаптивного тестирования 19](#_Toc354407520)

[1.7 Постановка задачи дипломной работы 21](#_Toc354407521)

[2 ТЕХНИЧЕСКОЕ ЗАДАНИЕ 23](#_Toc354407522)

[2.1 Введение 23](#_Toc354407523)

[2.1.1 Назначение 23](#_Toc354407524)

[2.1.2 Определения, сокращения, аббревиатуры 23](#_Toc354407525)

[2.2 Полное описание 24](#_Toc354407526)

[2.2.1 Перспектива 24](#_Toc354407527)

[2.2.2 Функция изделия 24](#_Toc354407528)

[2.2.3 Характеристика пользователей 24](#_Toc354407529)

[2.2.4 Ограничения 24](#_Toc354407530)

[2.3 Специфические требования 25](#_Toc354407531)

[2.3.1 Требования к внешним интерфейсам. 25](#_Toc354407532)

[2.3.2 Функциональные требования 25](#_Toc354407533)

[2.3.3 Требования к рабочим характеристикам 26](#_Toc354407534)

[3 Исследовательская часть 27](#_Toc354407535)

[3.1 Исследование алгоритмов адаптивного тестирования 27](#_Toc354407536)

[3.2 Жесткий алгоритм тестирования 27](#_Toc354407537)

[3.3 Мягкий алгоритм тестирования 29](#_Toc354407538)

[3.4 Комбинированный алгоритм 29](#_Toc354407539)

[3.5 Динамическая модель адаптивного тестирования 30](#_Toc354407540)

[3.6 Алгоритм Аванесова 32](#_Toc354407541)

[3.7 Разработка собственного алгоритма тестирования 32](#_Toc354407542)

[4 Конструкторская часть 33](#_Toc354407543)

[4.1 Выбор средств разработки 33](#_Toc354407544)

[4.2 Архитектура программного средства 34](#_Toc354407545)

[4.3 Модель хранения данных 36](#_Toc354407546)

[4.4 Модель потоков данных 38](#_Toc354407547)

[4.5 Проектирование пользовательского интерфейса 39](#_Toc354407548)

[4.6 Алгоритм работы проектируемой системы 43](#_Toc354407549)

[5 Экспериментальная часть 45](#_Toc354407550)

[5.1 Назначение тестирования 45](#_Toc354407551)

[5.2 Тестирование в нормальных условиях 45](#_Toc354407552)

[5.3 Проверка программной системы в экстремальных условиях 46](#_Toc354407553)

[5.4 Проверка программной системы в исключительных ситуациях 46](#_Toc354407554)

[5.5 Основные выводы 46](#_Toc354407555)

[6 Техническая документация 47](#_Toc354407556)

[6.1 Руководство пользователя 47](#_Toc354407557)

[6.2 Руководство администратора 47](#_Toc354407558)

[ЗаключЕние 49](#_Toc354407559)

[Список литературы 50](#_Toc354407560)

# Введение

Когда говорят о распространении информации через Интернет, обычно подразумевают, что либо идёт обмен информацией (новостями, личная переписка и т.д.), либо идёт распространение электронной продукции (программ, музыки, видео, любых других файлов).

Взглянем на все эти задачи более подробно.

Сервисы файлового распространения бывают следующих типов:

* Централизованные
  + FTP-сервера
  + HTTP-сервера
* Децентрализованные
  + cеть eDonkey
  + cеть kad
  + торренты

Стоит отметить, что файлы, зачастую, можно отправить и вместе с другой информацией. Т.е. использовать для этого электронную почту, сервисы мгновенной доставки сообщений и т.д.

В большинстве случаев, для пользователей предприятия, хватает возможностей централизованных средств распространения информации. Однако, бывают и исключения: чтобы снизить нагрузку на свои сервера, владельцы Open Source-проектов (например, Ubuntu) применяют децентрализованные сервисы распространения файлов.

Эти способы используются очень давно и никаких особых проблем с их использованием не возникает. Поэтому трудно услышать что-либо новое в области развития этих технологий.

Возможность обмена пользователей информацией между собой обеспечивается следующими сервисами:

* электронная почта
* сервисы обмена мгновенными сообщениями
* личные сообщения (на порталах, социальных сетях и т.д.)

Сервисы обмена мгновенными сообщениями пришли на смену электронной почте, которая, однако, до сих пор активно используется. Наиболее удобным способом общения являются личные сообщения в социальных сетях. Социальная сеть «ВКонтакте» предоставляет пользователям возможность вложить в сообщение изображения, звук, позволяет передать сообщения быстро, а также предоставляет удобный доступ к списку контактов. Тем не менее, сеть почти не накладывает ограничение на её использование и, в последнее время, сотрудников от неё стараются ограничить, а не приучить к использованию. Это происходит потому, что сотрудники слишком увлекаются общением в социальных сетях.

Для широковещательного распространения информации от пользователя к остальным пользователям используются следующие сервисы:

* социальные сети
* порталы

Однако, по уже указанной причине, сотрудников стараются ограждать, как от порталов, так и от социальных сетей, с целью планомерного ведения работы.

Становится ясно, что современные средства общения, хоть и позволяют оптимизировать процесс обмена информацией, но так имеют и существенные недостатки, в т.ч. сложность контролирования подчинённых. Жёсткий контроль подрывает психологическую атмосферу в коллективе, поэтому появляется необходимость в создании приложения, наследующего необходимый функционал и удобство использования из предыдущих систем, но не реализующая функционал, излишне потребляющий время сотрудников.

***Цель*** – реализация экспериментального мобильного ПО для сотрудников компании Polonium Arts, обеспечивающее внутрикорпоративное общение с возможностью удобного обмена фотографиями и их оценкой.

***Задачи:***

* Исследование предметной области;
* Разработка технического задания;
* организация работы приложения на основе базы данных и разграничения прав доступа к имеющейся информации. Доступ к ресурсам приложения возможен при регистрации или авторизации пользователя.

***Объект исследования*** – средства электронного общения, применимые в корпоративной среде.

***Предмет исследования*** – реализация средства электронного корпоративного общения, применимого в компании Polonium Arts.

# Аналитическая часть

## Общие сведения об электронном общении в организациях

Новые, базирующиеся на электронных средствах коммуникации организации являются принадлежностью постиндустриального общества. Если образ традиционных бюрократических организаций является деревом с расходящейся от высшего управляющего органа системой контроля и информационных потоков, то современные организации можно рассматривать скорее как разветвленные системы, комплексные сущности. Управление и деятельность в них осуществляются по-разному в разных центрах.

Применение высокоскоростных каналов связи, а также увеличение количества пользователей сети Internet позволило перевести процесс общения людей через сеть Internet на новый уровень взаимодействия - общение с собеседниками в реальном времени.

Образование электронных коммуникаций не только стирает грани расстояний, экономит время, но так, же и способствует увеличению большего обмена информацией между сотрудниками компаний, филиалами, представительствами и партнерами за рубежом.

В основном, электронные коммуникации можно представить четырьмя разновидностями, в зависимости от их величины и сложности:

* Внутрикорпоративная рассылка по электронной почте;
* Электронная страничка для сотрудников компании на внешнем корпоративном сайте;
* Внутрикорпоративный веб-сайт;
* Внутренний портал компании.

Внутрикорпоративная рассылка по электронной почте, наверное, самый распространенный способ оповещения сотрудников о каких либо новостях или грядущих событиях в большинстве компаний. Такой практике способствует наличие практически на каждом компьютере программ Outlook или The bat.

Основным достоинством такого рода общения есть то, что многопользовательское сообщение позволяет доставить необходимую информацию каждому из сотрудников компании, не отвлекая их от рабочего процесса. При этом сохраняется интерактивное общение между отправителем и получателем информации.

Несмотря на удобство в пользовании и скорость передачи информации, такой метод общения не исключает ряда недостатков. Например, если очень часто присылать сотрудникам сообщения одинакового типа, то они будут удаляться непрочитанными как самый обыкновенный спам. Кроме того, существует риск излишнего увлечения работников компании перепиской.

Более продвинутым видом внутренних электронных коммуникаций является создание специализированного раздела на внешнем сайте компании. Туда можно поместить фотографии сотрудников и их контактную информацию с возможностью интерактивного общения. Также там можно оставлять информацию и оповещения для сотрудников компании".

Применение такого метода в сочетании с электронной рассылкой, безусловно, позволит повысить уровень коммуникаций внутри компании. Но для крупной компании в любом случае необходимо более соответствующее средство, которым является внутрикорпоративный веб-сайт.

Внутренний сайт - это интерактивный ресурс, но в отличие от внешнего сайта, здесь пользовательской аудиторией являются только сотрудники компании. Внешние пользователи не имеют к нему доступа. В основном это сделано в целях безопасности, так как на сайте находится информация, которая зачастую является конфиденциальной.

Главное преимущество внутреннего сайта - возможность размещения больших массивов информации. Сюда можно поместить доску объявлений, форум, создать опрос сотрудников. С помощью такого ресурса при правильном его использовании можно эффективно управлять настроениями и мотивацией всего коллектива.

В больших компаниях иногда бывает недостаточно одного внутреннего сайта, поэтому для внутренних коммуникаций используется внутренний портал компании. В его состав входят сайты отделов и департаментов, документация системы менеджмента качества, объявления, новости, форумы и прочее. Большим преимуществом внутреннего портала является то, что сотрудники непосредственно вовлечены в процесс общения. На форумах ведется оживленное общение по различным тематикам, каждый из сотрудников может вносить свои предложения и пожелания, влиять на работу компании. С помощью опросов мы получаем обратную связь и собираем идеи для развития компании. Таким образом, мы получаем великолепный инструмент для администрирования документации и управления процессами системы менеджмента качества, для управления компанией и настроениями внутри ее.

Преимущество такого портала в том, что с его помощью можно предоставлять интересную информацию для всего персонала компании. При правильной подаче информации данный ресурс позволяет сплотить коллектив, сделать его дружной командой. Но в маленьких компаниях внутренний портал абсолютно не нужен - здесь лучше использовать другие методы коммуникации, например, такие как собрания и непосредственное общение.

Электронные коммуникации пригодятся при осуществлении антикризисного PR внутри компании и для борьбы со слухами. Учуяв нездоровую атмосферу в коллективе, можно принять превентивные меры. Например, проводить периодический опрос сотрудников и на основании его результатов принять ряд решений по урегулированию ситуации. Опять же, решения по этим результатам можно опубликовать на внутреннем сайте, дабы сотрудники смогли увидеть, что компания оперативно реагирует на их замечания.

Электронное общение все глубже проникает в офисные пространства по всему миру. С одной стороны, это дань моде, но с другой - психологический выверт. Для людей, склонных к интроверсии, электронная коммуникация дает возможность высказать то, что никогда не будет сказано при личном контакте. Интернет сделал коммуникационное пространство практически бесконечным. В то же время многие приверженцы электронной переписки как с клиентами и партнерами, так и с коллегами-соседями уповают на то, что электронная переписка позволяет сохранять конфиденциальность.

Более продвинутым видом внутренних электронных коммуникаций является создание специализированного раздела на внешнем сайте компании. Туда можно поместить фотографии сотрудников и их контактную информацию с возможностью интерактивного общения. Также там можно оставлять информацию и оповещения для сотрудников компании".

Применение такого метода в сочетании с электронной рассылкой, безусловно, позволит повысить уровень коммуникаций внутри компании. Но для крупной компании в любом случае необходимо более соответствующее средство, которым является внутрикорпоративный веб-сайт.

Внутренний сайт - это интерактивный ресурс, но в отличие от внешнего сайта, здесь пользовательской аудиторией являются только сотрудники компании. Внешние пользователи не имеют к нему доступа. В основном это сделано в целях безопасности, так как на сайте находится информация, которая зачастую является конфиденциальной.

Главное преимущество внутреннего сайта - возможность размещения больших массивов информации. Сюда можно поместить доску объявлений, форум, создать опрос сотрудников. С помощью такого ресурса при правильном его использовании можно эффективно управлять настроениями и мотивацией всего коллектива.

В больших компаниях иногда бывает недостаточно одного внутреннего сайта, поэтому для внутренних коммуникаций используется внутренний портал компании. В его состав входят сайты отделов и департаментов, документация системы менеджмента качества, объявления, новости, форумы и прочее. Большим преимуществом внутреннего портала является то, что сотрудники непосредственно вовлечены в процесс общения. На форумах ведется оживленное общение по различным тематикам, каждый из сотрудников может вносить свои предложения и пожелания, влиять на работу компании. С помощью опросов мы получаем обратную связь и собираем идеи для развития компании. Таким образом, мы получаем великолепный инструмент для администрирования документации и управления процессами системы менеджмента качества, для управления компанией и настроениями внутри ее.

Преимущество такого портала в том, что с его помощью можно предоставлять интересную информацию для всего персонала компании. При правильной подаче информации данный ресурс позволяет сплотить коллектив, сделать его дружной командой. Но в маленьких компаниях внутренний портал абсолютно не нужен - здесь лучше использовать другие методы коммуникации, например, такие как собрания и непосредственное общение.

Электронные коммуникации пригодятся при осуществлении антикризисного PR внутри компании и для борьбы со слухами. Учуяв нездоровую атмосферу в коллективе, можно принять превентивные меры. Например, проводить периодический опрос сотрудников и на основании его результатов принять ряд решений по урегулированию ситуации. Опять же, решения по этим результатам можно опубликовать на внутреннем сайте, дабы сотрудники смогли увидеть, что компания оперативно реагирует на их замечания.

Электронное общение все глубже проникает в офисные пространства по всему миру. С одной стороны, это дань моде, но с другой - психологический выверт. Для людей, склонных к интроверсии, электронная коммуникация дает возможность высказать то, что никогда не будет сказано при личном контакте. Интернет сделал коммуникационное пространство практически бесконечным. В то же время многие приверженцы электронной переписки как с клиентами и партнерами, так и с коллегами-соседями уповают на то, что электронная переписка позволяет сохранять конфиденциальность.

Почти каждая компания на сегодняшний день имеет как минимум часть электронных информационных коммуникаций сочетающих в себе:

* Интранет для эффективной совместной работы служащих компании;
* Интернет, обеспечивающий выход, к общедоступным серверам предоставляющий доступ к Интернет - услугам;
* Контакт-центры, позволяющие интегрировать услуги для тех, кто не имеет постоянного доступа в Интернет;
* Почтовую службу для рассылки материалов в письменной форме (документы, страховые полисы и т.п.).
* А так же всевозможные электронные программные средства офисного общения между сотрудниками (корпоративный чат или форум)

Действительно, с развитием Интернета задача общения с сотрудниками и получения обратной связи стала куда легче.

## Средства распространения графической информации

Под электронным обменом графической информации подразумевается обмен графической информацией

Адаптивное тестирование определяется в «Стандартах образовательного и психологического тестирования» как «последовательная форма тестирования, при которой последующие задания теста выбираются в зависимости от ответов на предыдущие задания». Отсюда компьютеризированное адаптивное тестирование — это адаптивное тестирование, проведение которого облегчается благодаря использованию компьютера.

Адаптивное тестирование основано на принципах последовательного анализа, впервые изложенных А. Вальдом []. Основной целью этой процедуры является создание теста либо в когнитивной, либо в аффективной области, состоящего из заданий, которые не выходят за пределы способностей индивидуума. Это достигается путем привлечения группы экспертов в данной области или опытных создателей тестов для разработки банка заданий, соответствующих измеряемой области. Каждое задание из этого банка затем ранжируется от низкого к высокому уровню трудности на основе информации о пропорции тестируемых, дающих на него правильный ответ, или пропорции ответов на какой-либо пункт определенным способом.

С началом внедрения компьютерных технологий и применения теории «задание-ответ» (IRT) в качестве инструмента анализа заданий была создана основа для использования компьютеров в адаптивном тестировании. Процедура выбора заданий и порядка их предъявления тестируемому существенно облегчалась машинными операциями.

Стратегия заключается в предъявлении тестируемому следующего по порядку задания умеренной трудности, которое выбирается компьютером из общего банка заданий в зависимости от результата предыдущего ответа. Задания в этом банке градуируются с использованием IRT-методов, реализованных на уровне программного обеспечения. В этом смысле можно сказать, что тест «адаптирует сам себя» к уровню способностей тестируемого. Конечной желаемой целью процедур компьютеризированного адаптивного тестирования является разработка теста, который обнаруживал бы практически одинаковую валидность и надежность в отношении всего диапазона тестируемого содержания или измеряемых конструктов.

К числу важных преимуществ компьютеризованного адаптивного тестирования можно отнести.

1. Тестирование является более качественным и объективным способом оценивания, его объективность достигается путем стандартизации процедуры проведения, проверки показателей качества заданий и тестов целиком.
2. Тестирование – более справедливый метод, оно ставит всех учащихся в равные условия, как в процессе контроля, так и в процессе оценки, практически, исключая субъективизм преподавателя. По данным английской ассоциации NEAB, занимающейся итоговой аттестацией учащихся Великобритании, тестирование позволяет снизить количество апелляций более чем в три раза, сделать процедуру оценивания одинаковой для всех учащихся вне зависимости от места проживания, типа и вида образовательного учреждения, в котором занимаются учащиеся.
3. Тесты – это более объёмный инструмент, поскольку тестирование может включать в себя задания по всем темам курса, в то время как на устный экзамен обычно выносится 2-4 темы, а на письменный – 3-5. Это позволяет выявить знания учащегося по всему курсу, исключив элемент случайности при вытаскивании билета. При помощи тестирования можно установить уровень знаний учащегося по предмету в целом и по отдельным его разделам.
4. Тест – это более точный инструмент, так, например, шкала оценивания теста из 20 вопросов, состоит из 20 делений, в то время, как обычная шкала оценки знаний — только из четырёх.
5. Тестирование более эффективно с экономической точки зрения. Основные затраты при тестировании приходятся на разработку качественного инструментария, то есть имеют разовый характер. Затраты же на проведение теста значительно ниже, чем при письменном или устном контроле. Проведение тестирования и контроль результатов в группе из 30 человек занимает полтора два часа, устный или письменный экзамен — не менее четырёх часов.
6. Тестирование — это более мягкий инструмент, они ставят всех учащихся в равные условия, используя единую процедуру и единые критерии оценки, что приводит к снижению предэкзаменационных нервных напряжений.

## Обзор программных средств

Немаловажным вопросом является оценка качества программных средств тестирования. Качество обучающих средств является одним из условий успешной интеграции компьютерных технологий в учебный процесс.

GMAT (The Graduate Management Admission Test) – тест, результаты которого засчитывают более 1500 бизнес-школ по всему миру. Этот стандартизированный компьютерный тест позволяет бизнес-школам максимально оценить уровень подготовки кандидатов к дальнейшему обучению в области бизнеса и менеджмента при поступлении на программы MBA или другие пост бакалаврские программы по менеджменту. Ежегодно тестирование проходят более 200 тысяч человек.

Особенностью GMAT (рис. 1.1) является то, что тест включает в себя проверку навыков речи, письма и математического счета. Сложность заключается в том, что тест принимается на английском, и, следовательно, требует высокого уровня знания английского языка.

GMAT разработан с целью проверки навыков, необходимых в сфере менеджмента. Однако он не требует специальных знаний и не касается узких областей специализации.

Тест GMAT уникален тем, что по его результатам можно определить потенциальные возможности человека, прогнозировать его успехи в ходе обучения в бизнес-школе.

Тест разработан в США, но используется и в других странах. Существует ряд аналогов GMAT, но популярны они, как правило, в неанглоязычных странах. [3]

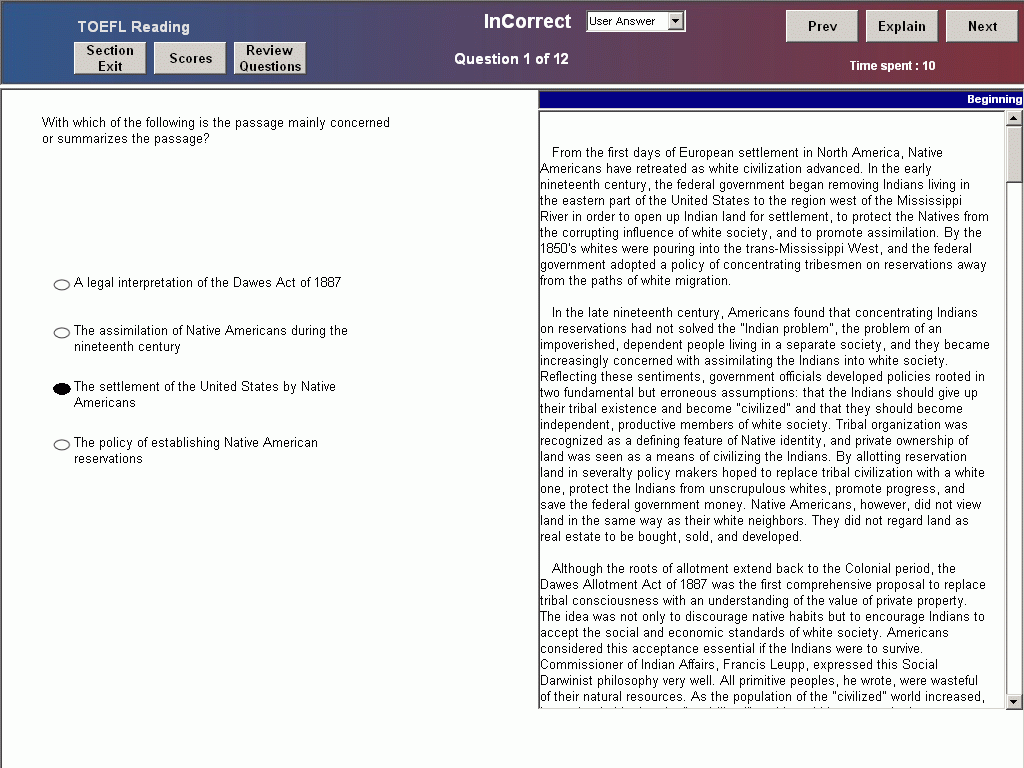


Рис. 1.1.Окно программы GMAT

Первый вопрос каждого раздела относится к средней группе сложности. Если вы отвечаете правильно, уровень сложности следующего вопроса повышается. Вернуться к предыдущему заданию невозможно, а переход к следующему вопросу происходит только после ответа на текущий [3].

GMAT не подходит для кафедры «Информатика и программное обеспечение», поскольку предполагает:

* знание английского языка на высоком уровне;
* направленность тестируемого на дальнейшее обучение в сфере экономики;
* прохождения тестирования занимает достаточно длительное время, порядка трех часов;
* необходимость в привлечении третьих лиц при определении окончательного результата.

Программа АСТ-Тест Player (рис. 1.2) предназначена для прохождения тестов. Функционал программы определяется параметрами конкретного теста и настройками системы тестирования.

АСТ-Тест Player – комплекс программ для компьютерного тестирования. Клиентский компонент данной системы представлена модулем для прохождения тестов, т.е. основной функцией «АСТ-теста» является контроль, а не формирование навыков и умений.

Функции приложения:

* обеспечение тестовых испытаний;
* создание библиотеки пользователей и тестов;
* создание расписания тестирования;
* контроль за ходом тестирования;
* получение результатов тестирования;
* диагностика работы системы;[4]
* возможность динамического формирования на основе банка тестовых заданий неограниченного количества вариантов тестов, различающихся по составу заданий, тематической структуре, способу оценивания результатов и алгоритмов тестирования.[5]

Возможности:

* авторизация по логину/паролю;
* выбор теста из списка;
* отображение инструкции по тесту;
* задания с аудио-содержанием;
* 4 формы заданий;
* виды оценки: процент, класс, балл, на основе меры трудности;
* ограничение по времени: задание, тест;
* пропуск заданий / изменение ответа на задание;
* отображение признака верно/неверно после ответа на задание;
* восстановление прерванного сеанса тестирования;
* отображение результата тестирования;
* просмотр неверно выполненных заданий;
* просмотр успеваемости по разделам теста;
* отображение истории сеансов тестирования;
* тестирование в локальной сети или через Internet;
* работа в терминальном режиме.[6]

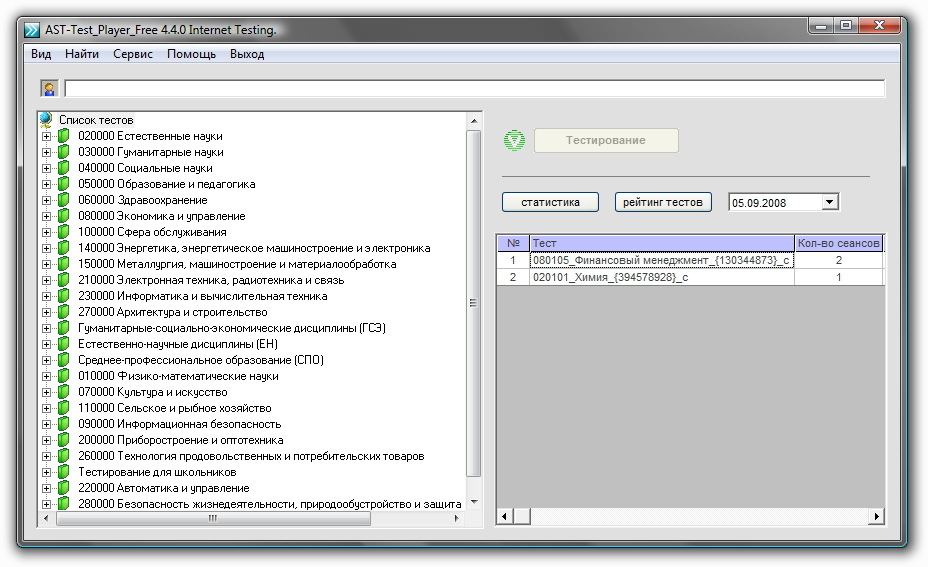


Рис. 1.2. Окно программы, предоставляющее выбор теста через список, сгруппированный по специальностям.

ACT-Тест Player содержит ряд преимуществ, которые не плохо было бы иметь в разрабатываемом продукте для кафедры «Информатика и программное обеспечение». Однако система предполагает интерактивный урок при изучении какой-либо области и дальнейший контроль полученных знаний.

Еще какую-нибудь программу

Анализируя представленные системы тестирования, хотелось бы подчеркнуть, что важным аспектом разработки тестируемых сред является:

* в клиентском компоненте наглядный пользовательский интерфейс в различных форматах;
* в административном компоненте возможность добавления пользователей на основе авторизации, создание расписания тестирования для каждого пользователя;
* в конструктивном компоненте возможность добавлять и редактировать БД вопросов;
* в серверном компоненте надежность хранения учебных материалов и базы данных пользователей и доступность сервера через сеть Интернет.

Каждая из рассмотренных нами систем располагает полезными и удобными возможностями, но не отвечает в полной мере потребностям кафедры «Информатика и программное обеспечение» в плане возможности использования адаптивного тестирования в необходимом объеме.

В таблице 1.1 представлены достоинства и недостатки выше перечисленных программ-аналогов.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Критерии  Программы | Область применения | Назначение | Особенности | Недостатки |
| GMAT | Область бизнеса и менеджмента при поступлении на программы MBA или другие пост бакалаврские программы по менеджменту | Проверка навыков, необходимых в сфере менеджмента. | включает в себя проверку навыков речи, письма и математического счета | Знание английского языка на высоком уровне.  Направленность тестируемого на дальнейшее обучение в сфере экономики.  Тестирование занимает достаточно длительное время.  Необходимость в привлечении третьих лиц при определении окончательного результата. |
| АСТ- Тест Player | Учебные заведения | Контроль знаний | Использование мультимедийных заданий. | Проведение интерактивного урока перед прохождением тестирования. |

В результате анализа предметной области было выяснено, что создание сайта кафедры с возможностью прохождения предметного адаптированного тестирования (дистанционно, например, с целью тренировки) или стационарно (с последующим знанием обучаемого) является актуальным.

## Моделирование процессов адаптивного тестирования

### Модель вариантов использования

UML (Unified Modeling Language) – стандартный язык для написания моделей анализа, проектирования и реализации объектно-ориентированных программных систем. Может использовать для визуализации, спецификации, конструирования и документирования результатов программных проектов.

Диаграмма Use Case определяет поведение системы с точки зрения пользователя. Рассматривается как главное средство для первичного моделирования динамики системы, используется для выяснения требований к разрабатываемой системе, фиксации этих требований в форме, которая позволит проводить дальнейшую разработку.

Достоинствами модели вариантов использования являются такие аспекты, как:

* определяет пользователей и границы системы;
* определяет системный интерфейс;
* является основой для написания пользовательской документации;
* хорошо вписывается в любые методы проектирования (как объектно-ориентированные, так и структурные).

В Русской литературе Use Case часто называют диаграммами прецедентов или диаграммами вариантов использования.[7]

Приведем диаграмму вариантов использования (рис.1.3). Студент и Администратор являются актерами, причем Администратор является потомком пользователя, поэтому может использовать все возможности предусмотренные пользователю. Помимо этого Администратор может формировать новое тестирование. Студент имеет такие возможности, как прохождение тестирования и просмотр результатов пройденных тестов.

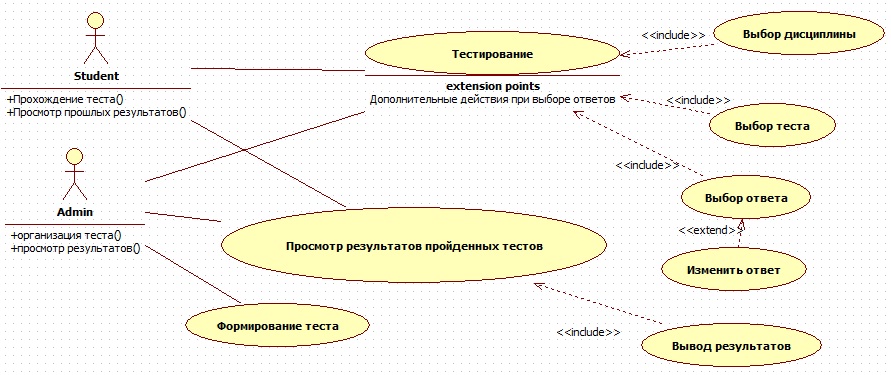


Рис. 1.3. Диаграмма вариантов использования

### Функциональная модель

## Постановка задачи дипломной работы

В настоящее время наличие доступа к интернету не является необычным, скорее наоборот – это норма. Поэтому важно использовать открывающиеся возможности применения интернет-ресурсов для реализации на практике возможностей дистанционного (при необходимости) или стационарного обучения.

Перед нами стоит ***цель*** разработать сайт, который способствовал бы большей автоматизации учебного процесса.

***Задачи:***

* Исследование предметной области;
* Разработка технического задания;
* организация работы сайта на основе базы данных и разграничения прав доступа к имеющейся информации. Доступ к ресурсам сайта возможен при регистрации или авторизации пользователя.

***Объект исследования*** – процесс разработки возможности прохождения отчетного тестирования посредством интернет-сайта.

***Предмет исследования*** – представляет собой автоматизацию работы кафедры с точки зрения преподавания и обучения.

# ТЕХНИЧЕСКОЕ ЗАДАНИЕ

Разрабатываемый модуль сайта относится к классу распределённых информационных систем. Разработка ведётся в соответствии с концепцией MVC.

## Введение

### Назначение

Данное программное средство предназначено для автоматизации распространения графической информации в рамках одного предприятия.

Данный программный продукт рассчитан на любой класс пользователей, так как для него рассчитан простой, интуитивно понятный интерфейс, для работы с данным программным продуктом не требуется никаких определенных навыков или дополнительных знаний.

### Определения, сокращения, аббревиатуры

ПО – программное обеспечение.

ПС – программное средство.

ОС – операционная система.

Мобильная ОС – ОС, предназначенная для эксплуатации, в первую очередь, на мобильных устройствах (телефоны, планшеты и т.д.)

Мобильное приложение – ПО или ПС, разработанное для мобильной ОС.

Android – конкретная реализация ОС из группы мобильных ОС.

Android-приложение – ПО или ПС, клиентская сторона которого является мобильным приложением для ОС Android.

Java – объектно-ориентированный язык программирования, разработанный компанией Sun Microsystems (в последующем приобретённой компанией Oracle). Приложения Java обычно компилируются в специальный байт-код, поэтому они могут работать на любой виртуальной Java-машине (JVM) вне зависимости от компьютерной архитектуры[1].

Лайк – операция над чем-либо выполняющаяся пользователем, когда он хочет показать, что ему это что-то нравится.

Аватар – изображение, привязанное к конкретному пользователю. Является «лицом» пользователя, используется для дифференцирования пользователей между собой.

## Полное описание

### Перспектива

Программное средство должно иметь клиент-серверную архитектуру, построенную на базе web и мобильных технологий. Должна использовать бесплатную СУБД для возможности расширения функциональности не отраженной в данных ТЗ.

### Функция изделия

Функция ввода данных – позволяет ввести новые данные и отправить их на сервер, где они сохранятся в базу данных.

Функция просмотра – предназначена для возможности просмотра, введённых всеми пользователями данных в виде «новостной ленты».

### Характеристика пользователей

Пользователь должен уметь пользоваться мобильным устройством на базе операционной системы Android, уметь ориентироваться в стандартных элементах управления.

### Ограничения

Работа с ОС Android версии 2.3 и новее.

## Специфические требования

### Требования к внешним интерфейсам.

Интерфейс пользователя. Реализовать графический интерфейс пользователя, представляющий собой страницу, содержащую поля для входа в систему или регистрации в системе. После авторизации пользователь имеет возможность перехода на экран, на котором располагаются все фотографии (начиная с новейших), подписи к ним, а также флажок, позволяющий поставить или снять лайк с фотографии.

На этом экране располагается кнопка для создания новой фотографии, а также кнопка для перехода в профиль пользователя. В своём профиле пользователь имеет возможность посмотреть выложенные фотографии и посмотреть, кто и к каким фотографиям поставил лайк.

В профиле есть находится кнопка, позволяющая перейти на экран смены персональных данных (имя, адрес электронной почты, дата рождения, пол, пароль) и аватара.

На экране создания новой фотографии можно выбрать фото из галлереи телефона. После фотографию предлагается обрезать, после чего пользователь должен попасть на экран, где он может либо выбрать какую-то заранее подготовленную подпись к фотографии, либо ввести свою собственную. На этом же экране пользователь может поделиться с фотографией в социальные сети (поддерживаются «Вконтакте», Facebook, Twitter).

### Функциональные требования

Программа предоставляет пользователю следующие возможности:

* авторизация;
* регистрация;
* доступ к данным своего профиля (для чтения);
* доступ к данным своего профиля (для записи);
* просмотр информации о фотографиях в виде «новостной ленты»;
* возможность отправить фотографию и подпись в социальные сети;
* возможность поставить или снять лайк какой-либо фотографии
* возможность посмотреть, лайки, поставленные другими пользователями
* возможность посмотреть все свои фотографии, отдельно от остальных

Администратор приложения, через администраторскую панель (на сервере) может удалять пользователей, выложенные фотографии, а также править любую другую информацию. Он имеет доступ ко всем возможностям приложения, кроме просмотра паролей других пользователей.

Неавторизованный пользователь не имеет права доступа ни к каким функциям приложения, кроме авторизации и регистрации.

Авторизованный пользователь имеет права доступа ко всем возможностям приложения, предназначенных для него (т.е. может редактировать свои данные, а не чужие и т.п.).

.

### Требования к рабочим характеристикам

Число одновременно поддерживаемых пользователей не ограничено, однако вход с одного мобильного устройства разным пользователям не возможен. На уровне приложения вообще не должно быть возможности выйти из учётной записи пользователя. Однако, это становится возможно, если приложение переустановить.

## Этапы разработки

Программное обеспечение программного средства должно разрабатваться в следующем порядке:

* анализ предметной области;
* разработка технического задания;
* проектирование системы;
* реализация системы;
* тестирование законченной программы.

## Порядок контроля и приемки

Для всестороннего контроля работы системы необходимо разработать специальные тестовые сценарии, в результате следования которым, в полной мере отразится работоспособность системы. Для проверки правильности работы программы должно быть проведено тестирование всех режимов работы.

# Исследовательская часть

## Исследование алгоритмов адаптивного тестирования

Анализ алгоритмов адаптивного тестирования позволяет выделить следующие варианты:

1. Алгоритм тестирования «жесткий»;
2. Алгоритм тестирования «мягкий»;
3. Комбинированный алгоритм.

## Жесткий алгоритм тестирования

«Жесткий» алгоритм тестирования основывается на формировании области незнания, т.е. вопросы будут задаваться по темам, которые пользователь не знает, или знает плохо. Таким образом, пользователь наказывается за незнание какой-либо темы. Вначале в области незнания находятся все темы теста. По мере прохождения опроса, область может динамически меняться:

* если пользователь отвечает на вопрос правильно, то тема, к которой принадлежит вопрос, удаляется из области;
* если пользователь отвечает на вопрос неправильно или частично правильно, то тема остается в области;
* если область пуста, то она заполняется снова всеми темами теста.

Если вопросы из области незнания закончились, то система задает вопросы из других тем.

Тестирование прекращается в одном из следующих случаев:

* пройден максимальный уровень сложности;
* пользователь не сумел преодолеть минимальный уровень сложности;
* истекло время на прохождение теста или преодоление уровня сложности;
* исчерпан лимит вопросов.

После прекращения процесса тестирования определяется рейтинг пользователя, который вычисляется по формуле:

где

Бреал. – количество баллов, которое набрал пользователь;

Бмакс. — количество баллов, которое мог бы набрать пользователь, если бы отвечал правильно на все вопросы;

Утекущ. — текущий уровень сложности;

Умакс. — максимальный уровень сложности.

Умножение максимального количества баллов на коэффициент необходимо, чтобы учесть возможность того, что за прошедшее время студент не достиг максимального уровня сложности и, соответственно, претендовать на хорошую оценку не может.

Достоинства такого подхода следующие: при данном подходе полностью исключается удача при выборе очередного вопроса. Таким образом, пользователю недостаточно знать несколько тем, чтобы хорошо пройти тест, так как в случае верного ответа система начнет задавать вопросы из других тем, которые пользователь может не знать. А если он не знает какой-либо темы, то он может застрять на ней надолго и испортить себе оценку неправильными ответами. Недостатком такого подхода является то, что если пользователь знает предмет хорошо, но несколько тем знает плохо, то система, выяснив это, будет «валить» пользователя, и он не сможет получить хорошую оценку. Ситуация усугубляется тем, что если пользователю сразу попался вопрос из темы, которую он плохо знает, то он может сразу получить плохую оценку, даже если знает в совершенстве все остальные темы.

## Мягкий алгоритм тестирования

«Мягкий» алгоритм тестирования основывается на формировании области знания, т.е. вопросы будут задаваться по темам, которые пользователь знает хорошо. Таким образом, пользователь не наказывается за незнание какой-либо темы.

Вначале в области знания находятся все темы теста. По мере прохождения опроса, область может динамически меняться:

* если пользователь отвечает на вопрос неправильно, то тема, к которой принадлежит вопрос, удаляется из области;
* если пользователь отвечает на вопрос правильно или частично правильно, то тема остается в области;
* если область пуста, то она заполняется снова всеми темами теста.

Если вопросы из области знания закончились, то система задает вопросы из других тем.

Прекращение тестирования и выставление оценки осуществляется аналогично предыдущему алгоритму.

Достоинства такого подхода следующие: при данном подходе полностью исключается возможность застрять на какой-нибудь теме и, следовательно, испортить себе оценку неправильными ответами.

Недостатком такого подхода является то, что если пользователь знает предмет плохо или знает всего несколько тем, то система будет «вытягивать» пользователя и у него всегда есть шанс получить хорошую оценку, даже если он ее не заслуживает. Ситуация усугубляется тем, что если пользователю сразу попался вопрос из темы, которую знает, то он может сразу получить хорошую оценку, даже если вообще не знает все остальные темы.

## Комбинированный алгоритм

Комбинированный алгоритм сочетает в себе достоинства предыдущих алгоритмов. Пользователь не наказывается за незнание какой-либо темы, но и не поощряется чрезмерно. В данном алгоритме нет области знания и незнания – есть просто область тем, из которых выбираются вопросы.

Вначале в области находятся все темы теста. По мере прохождения опроса, область может динамически меняться:

* если пользователь отвечает на вопрос неправильно и у него мало баллов, то тема, к которой принадлежит вопрос, удаляется из области;
* если пользователь отвечает на вопрос правильно и у него много баллов, то тема, к которой принадлежит вопрос, также удаляется из области;
* если область пуста, то она заполняется снова всеми темами теста.

Если вопросы из области тем закончились, то система задает вопросы из других тем.

Прекращение тестирования и выставление оценки осуществляется аналогично предыдущему алгоритму.

Таким образом, пользователь не штрафуется, если у него мало баллов, но и не поощряется, если баллов много.

## Динамическая модель адаптивного тестирования

Разработана модель адаптивного тестирования знаний, ориентированная не только на определение уровня подготовленности пользователей, но и позволяющая динамически изменять число вопросов в зависимости от числа их уровней трудности, при сохранении надежности получаемых результатов не ниже чем при линейном тестировании.

Эффективное управление тестированием знаний предполагает, что для пользователей с различным уровнем подготовленности будут предложены вопросы различного уровня трудности, т.е. будет проведено адаптивное тестирование. При адаптивном тестировании каждый из пользователей получает свое количество вопросов и выходит на определенный уровень трудности, после чего тестирование заканчивается, а результаты переносятся на метрическую шкалу.

Адаптивное тестирование позволяет решить две задачи: уменьшение времени тестирования и уменьшение загрузки каналов связи и компьютерного оборудования.

Суть предлагаемой модели адаптивного тестирования заключается в следующем: пусть имеется -уровней трудности вопросов (число вопросов на каждом уровне имеет значение только для уточнения уровня знаний). Разобьем -уровней на отрезков по уровней трудности вопросов на каждом (с округлением до целого).

Проведем верхнюю оценку минимального числа вопросов, которое потребуется для определения уровня знаний условного пользователя. Самое большое количество вопросов (при старте с самого нижнего уровня) должно быть задано пользователю, подготовленность которого находится вблизи самого большого уровня трудности вопросов (на последнем отрезке). Для того чтобы дойти до последнего отрезка, потребуется задать *x*-вопросов (по числу отрезков) и не более чем  – вопросов, чтобы уточнить его уровень на последнем отрезке. Таким образом, функция *f(x) -* числа вопросов будет иметь вид: . Данная функция имеет минимум при , а максимальное число вопросов будет не более чем .

На рис.3.1. представлена графическая схема алгоритма уточнения уровня подготовленности пользователя после первого неверного ответа (точка А), – отрезок номер *k*, – отрезок номер *k+1*. Стрелками над линией указаны переходы после правильных ответов, под линией – при ошибочных.



Рис.3.1. Графическая схема алгоритма уточнения

уровня подготовленности пользователя

Для проверки эффективности использования предлагаемого алгоритма адаптивного тестирования были проведены эксперименты, 75 студентам был предложен тест состоящий из 36 заданий. При проведении эксперимента студентам было предложено пройти данный тест, как в линейной, так и в адаптивной форме.

Расчет коэффициента корреляции Пирсона дает величину *r*=0,88, это говорит о том, что между баллами, полученными при линейном тестировании и адаптивном существует сильная корреляционная связь, а значит, результаты, полученные при использовании адаптивного тестирования являются не менее валидными, чем обычного линейного.

Отметим, что при адаптивном тестировании каждому испытуемому задается в среднем 9,2 заданий против 36 при линейном, что не превышает верхней оценки .

## Алгоритм Аванесова

## Разработка собственного алгоритма тестирования

# Конструкторская часть

## Выбор средств разработки

Для разработки автоматизированной системы кафедры использованы следующие инструменты:

* Android SDK
* Java в качестве языка программирования для клиентской части программного средства.
* Язык XML для описания интерфейсов, а также основных ресурсов приложения (анимаций, стилей и т.п.)
* Eclipse IDE совместно с ADT (Android Development Tools)
* MySQL сервер (mysqld)
* Улучшенный 9-patch редактор [2]
* Ruby, Ruby on Rails, ActiveAdmin

Разрабатываемое приложение не имеет строгих критериев выбора операционной системы для серверной стороны, а также других программных продуктов, используемых совместно с проектируемым. Поэтому в качестве языка разработки был выбран язык Ruby. В качестве СУБД была выбрана MySQL.

Основными критериями выбора вышеописанных средств являются: удобство их совместного использования, хорошая степень владения каждым из компонентов в отдельности, а также всеми в связке, наличие различных библиотек, существенно позволяющих облегчить разработку.

Ruby — динамический, рефлективный, интерпретируемый высокоуровневый язык программирования для быстрого и удобного объектно-ориентированного программирования. Язык обладает независимой от операционной системы реализацией многопоточности, строгой динамической типизацией, сборщиком мусора и многими другими возможностями.

Кроссплатформенная реализация интерпретатора языка является полностью свободной[3].

В качестве средства, облегчающего взаимодействие между программами, написанными на языке Ruby и средой Internet используется программная библиотека Rails on Rails.

Ruby on Rails предоставляет архитектурный образец Model-View-Controller (модель-представление-контроллер) для веб-приложений, а также обеспечивает их интеграцию с веб-сервером и сервером базы данных.

Для создания администраторской панели используется дополнение для Ruby On Rails под названием ActiveAdmin.

Для создания клиентской части приложения было выбрано средство официальное средство разработки Android SDK. Оно позволяет писать приложения для операционной системы Android на языке Java, удобно составлять интерфейсы пользователя при помощи языка XML, а также предоставляет много других удобных возможностей, в том числе дополнение Android Development Tools для среды разработки Eclipse.

Когда стоит задание написать приложение для ОС Android, автоматически ставится задача выбора языка программирования для реализации этого приложения. На выбор есть: Java, C++, Ruby (совместно с ruboto [4]), связка HTML + CSS + JavaScript.

Использование Ruboto предполагает неиспользование средств из ADT, что усложняет разработку приложения (в частности, создание пользовательских интерфейсов и переработку кода).

Связка HTML + CSS + JavaScript позволяет написать приложение не только для Android, но и для других платформ, что даёт ей очень большие преимущества перед родным SDK. Но также у неё есть и существенный недостаток: на самом Android’е такие приложения работают неприемлимо медленно, особенно в части интерфейса – каждое нажатие на кнопку идёт с задержкой, что раздражает пользователей.

С++ даёт возможность писать кроссплатформенные приложения, которые будут выполняться с максимально возможной скоростью. Так же, для C++ существует огромное количество библиотек и инструментов, что избавляет его (в контексте написания приложений для Android) от недостатков Ruby и HTML, но Google заявляет, что использование C++ для приложений невыгодно большинству приложений и неоправданно сложно, рекомендуя использовать оффициальный SDK [5].

Разметка интерфейса производится с помощью языка XML. Этот подход оффициально удтверждён Google и используется в приложениях, используемых в Android SDK повсеместно. Альтернатива этому подходу – создавать интерфейс вручную, используя для этого соответствующие вызовы функций в коде. Использование XML предпочтительно, т.к. для разных устройств можно задать разные XML-файлы, разные значения констант (размер шрифта и т.п.), которые будут выбраны операционной системой автоматически при установке приложения на телефон.

В качестве среды разработки выбран Eclipse + ADT. Это оффициальные средства разработки от Google и наиболее распространённые. По причине распространённости они и были выбраны: все библиотеки, предназначенные для работы с социальными сетями используют в своих репозиториях файлы именно для этой среды. Стоит подчеркнуть, что среда популярна не зря: тут есть и удобные средства для программирования (средства переработки кода; множество «мастеров», облегчающих повседневные задачи, шаблоны кода и т.п.), так и другие полезные утилиты: запаковка приложения в APK-файл, визуальный редактор интерфейсов и т.д.

СУБД MySQL выбрана по причине повсеместной распространённости, бесплатности и опыта разработки под неё приложений. Т.к. приложение не использует интенсивно возможности БД, на выборе СУБД внимание заострено не было, но в ней был высмотрено преимущество: есть совместимый с MySQL продукт, называемый MySQL Cluster NDB. Это версия MySQL совместимая с оригинальной и позволяющая организовывать распределённый вычислительные сети. Способна хорошо масштабироваться, в т.ч. в связке с уже существующими продуктами и является абсолютно бесплатной.

Изображения 9-patch используются для создания масштабируемых интерфейсов и фонов. «Улучшенный 9-patch редактор» был выбран из-за того, что позволяет удобно просматривать изображения, произвольно их растягивать и помещать сверху произвольные надписи. Кроме того, он позволяет делать 9-png разметку, не рисуя пиксели вручную, что очень удобно и ускоряет разработку интерфейсов.

## Архитектура программного средства

Разрабатываемая программная система для упрощения разработки разбита на отдельные модули, которые можно модифицировать, не сильно влияя на остальные части приложения.

* **Модуль оповещения пользователя** – модуль, отвечающий за вывод информации о текущем событии и уже пройденном событии, если на момент события телефон пользователя был выключен.
* **Модуль регистрации и аутентификации –** модуль, отвечающий за вход пользователей в приложение и их регистрацию.
* **Модуль просмотра фотографий –** модуль, отвечающий, за просмотр фотографий, вылаженных текущим и другими пользователями.
* **Модуль создания фотографии –** модуль отвечает за взаимодействие с камерой (создание фотографии), обрезку фотографии до нужного размера, приложения к ним подписи и отправки в новостную ленту.
* **Модуль профиля пользователя –** содержит важную (с точки зрения приложения) информацию для пользователя – выложенные им фотографии и лайки других пользователей.
* **Модуль настроек пользователя –** отвечает за возможность сменить пользователю аватар, основную информацию и пароль.

Пользователь заходит на сайт через модуль интерфейса, где пользователю предоставляется возможность пройти авторизацию для дальнейшей работы.

После того как пользователь авторизовался на сайте, ему присваивается определенный статус, в соответствии с которым определяются его права на доступ к тому или иному разделу.

Администратор имеет полный доступ ко всему содержимому сайта.

На рис. 4.1 изображена общая архитектура сайта. Модуль работы с БД позволит администратору делать резервные копии базы данных и восстанавливать из них, осуществлять изменения базы данных вручную.

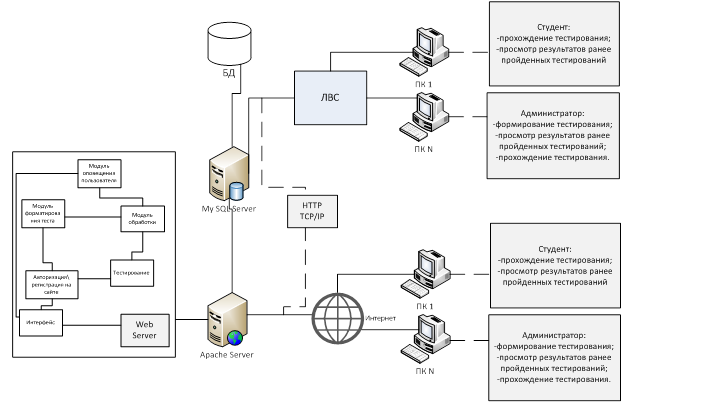


Рис 4.1 общая архитектура сайта.

## Моделирование работы автоматизированной системы

### Модель размещения

### Модель хранения данных

В качестве способа моделирования данных были выбраны диаграммы «сущность-связь».

Логический уровень модели данных представлен на рис.3.2.

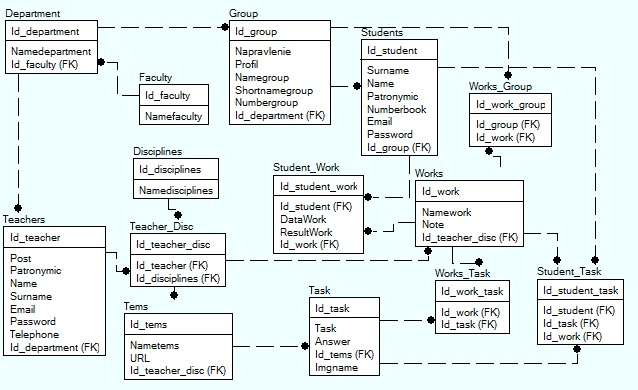


Рис.3.2. Модель данных. Логический уровень.

Модель данных разработана на основе технического задания, она включает в себя таблицы, необходимые для хранения информации и заемщике. Основными являются таблицы:

* Department. В данной таблице содержится информация о кафедрах, которые входят в факультет.
* Group. Содержит информацию о группе, ее полном и коротком названии. К какой кафедре принадлежит группа.
* Teacher. Включает полную информацию о преподавателе кафедры, его должность. Его адрес почты и пароль для аутентификации и телефон.
* Work. В данной таблице представлена информация о работах, которые внесены в план дисциплины.
* Student. Таблица содержит полную информацию о студенте, номер в группе, его адрес почты и пароль для аутентификации.
* Faculty. Таблица включает информацию наименовании факультета и кафедрах, входящих в нее.
* Task. Это таблица, содержащая информацию заданиях, планируемых для определенной дисциплины.

Остальные представленные таблицы являются более подробно рассмотренными таблицами Work, Task и Teacher. Алгоритмическое конструирование модуля социальной сети.

### Диаграмма классов

### Модель потоков данных

При проектировании системы был применен структурный подход [8]. Сущность структурного подхода состоит в декомпозиции программного средства на автоматизируемые функции: система разбивается на функциональные подсистемы, которые в свою очередь делятся на подфункции, подразделяемые на задачи и так далее. Процесс разбиения продолжается вплоть до конкретных процедур. При этом автоматизируемая система сохраняет целостное представление, в котором все составляющие компоненты взаимоувязаны.

В рамках структурного подхода к проектированию системы в качестве модели, отражающей функциональность системы, было выбрано построение диаграмм потоков данных (DFD) [9]. Диаграммы DFD позволяют описать процесс обмена информацией между элементами изучаемой системы. DFD отображает источники и адресаты данных, идентифицирует процессы и группы данных, связывающие в потоки одну функцию с другой, а также, что важно, определяет накопители (хранилища) данных, которые используются в исследуемом процессе.

Модель системы определяется как иерархия диаграмм потоков данных, описывающих асинхронный процесс преобразования информации от ее ввода в систему до выдачи пользователю.

Диаграммы верхних уровней иерархии определяют основные процессы или подсистемы с внешними входами и выходами. Они детализируются с помощью диаграмм нижнего уровня. Такая детализация продолжается, создавая многоуровневую иерархию диаграмм до тех пор, пока не будет достигнут уровень декомпозиции, на котором процессы становятся элементами и детализировать их не имеет смысла.

Источники информации (внешние сущности) порождают информационные потоки (потоки данных), переносящие информацию к подсистемам или процессам. Те в свою очередь преобразуют информацию и порождают новые потоки, которые переносят информацию к другим процессам или подсистемам, накопителям данных или внешним сущностям - потребителям информации.

Внешняя сущность представляет собой материальный предмет или физическое лицо, представляющее собой источник или приемник информации, в нашем случае - это преподаватель, оператор, которые инициируют процесс создания тестовых заданий.

Процесс представляет собой преобразование входных потоков данных в выходные в соответствии с определенным алгоритмом. В нашем случае процессами являются: подготовка материалов, разбиение материалов на вопросы, ответы и уровни сложности, а так же работа с программой.

Накопитель данных представляет собой базу данных, где хранятся созданные вопросы, ответы к ним и уровень сложности, выставляемый к ответам.

…….

## Проектирование пользовательского интерфейса

Вопрос пользовательских интерфейсов стал играть важную роль при разработке программного обеспечения только тогда, когда прямой доступ к компьютерам получил кто-то еще, помимо программистов и операторов. Современная концепция пользовательских интерфейсов появилась только тогда, когда появились терминалы, напрямую или косвенно соединенные с компьютерами. По мере роста числа людей, которые могли напрямую взаимодействовать с компьютерами, человеко-машинный интерфейс становился все более значимой задачей для разработчиков и проектировщиков программных средств. Если вначале решались в основном проблемы технологий (пользовательские интерфейсы), то потом стали решаться проблемы людей (пользователей).

Чтобы интерфейс программы считался правильным, должны выполняться следующие условия:

* Доступность. Интерфейс программы подготовки адаптивного тестирования можно назвать доступным и интуитивно понятным, так как все поля для заполнения подписаны и расположены в порядке, в котором мы чаще всего задаем вопрос (вопрос, ответ к нему…)
* Эффективность. Наряду с доступностью программа не перегружена дополнительными формами, которые не будут мешать опытному пользователю постоянно использовать программу
* Контекст. Система должна удовлетворяет существующим реалиям и текущей среде эксплуатационного контекста, внутри которого она будет разворачиваться и применяться.

Классы для представления модуля адаптивного тестирования. На рис.3.3. представлена диаграмма классов для представления модуля адаптивного тестирования. Ниже представлена их спецификация.

|  |
| --- |
| Рис. 3.2. Диаграмма классов модуля адаптивного тестирования |

Класс Группа – необходим для хранения информации о группах, имеющих возможность пройти тестирование. В качестве член-данных выступают: номер группы. Методами являются: выбор группы.

Класс Студент – необходим для хранения информации о студентах конкретной группы, имеющих возможность пройти тестирование. В качестве член-данных выступают: ФИО, группа, логин, пароль, результат. Методами являются: регистрация, авторизация, восстановление пароля, выбор ФИО, выбор группы, показать результаты.

Класс Дисциплина – необходим для выбора тестирования по конкретной дисциплине, изучаемой в рамках указанной ранее группы. В качестве член-данных выступают: наименование. Методами являются: выбор дисциплины.

Класс Оценка – необходим для хранения информации о результатах пройденного тестирования. В качестве член-данных выступают: результат. Методами являются: показать результат.

Класс Обработчик – служит для анализа поступающих данных и формировании дальнейшего перечня вопросов. В качестве член-данных выступают: количество ответов, количество правильных ответов, время max. Методами являются: обработка ответов.

Класс Тест – содержит набор предлагаемых тем тестов. В качестве член-данных выступают: тема, вопросы max, время max. Методами являются: выбор темы.

Класс Информация о тесте – содержит информацию, о пройденных ранее тестированиях. В качестве член-данных выступают: ФИО тестируемого, дата, результат. Методами являются: вывод информации.

Класс Задания – служит для доступа к вопросам различных уровней сложности. В качестве член-данных выступают: вопросы нижнего уровня сложности, вопросы среднего уровня сложности, вопросы высокого уровня сложности, сложность вопроса. Методами являются: выбор вопросов нижнего уровня сложности, выбор вопросов среднего уровня сложности, выбор вопросов высокого уровня сложности.

Класс Администратор – служит для обеспечения бесперебойной работы администратора сайта. В качестве член-данных выступают: логин, пароль. Методами являются: регистрация, авторизация, восстановление пароля.

Класс Интерфейс – класс, хранит форму, посредством которой осуществляется визуализация программы. В качестве член-данных выступают: форма. Методами являются: передача данных, вывод сообщения.

## Алгоритм работы проектируемой системы

На рис. 3.4 представлена алгоритмическая модель модуля адаптивного тестирования. В этой модели видно, что сайт состоит из нескольких модулей. Количество модулей может изменяться, что позволяет легко модифицировать модульную структуру без вмешательств в алгоритм работы основного сайта.

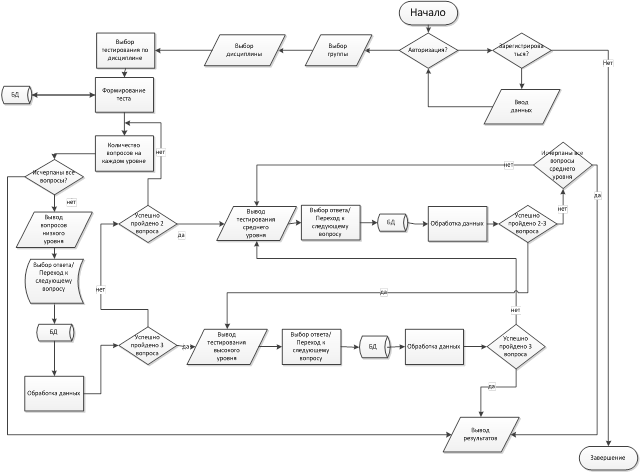


Рис. 3.4. Алгоритмическая модель программной системы

# Экспериментальная часть

## Назначение тестирования

Тестирование – это процесс, который позволяет оценить качество производимого продукта. Качественный программный продукт должен отвечать предъявленным к нему требованиям. Тестирование представляет собой процесс проверки разрабатываемого продукта на соответствие этим требованиям.

Оно направлено на выявление дефектов и на оценку свойств разрабатываемой системы. Целью тестирования в нашем случае является оценка качества разрабатываемой автоматизированной системы.

## Тестирование в нормальных условиях

Тестирование программного комплекса в нормальных условиях проводилось с целью определения его соответствия требованиям технического задания и подтверждения корректности работы программы в характерных ситуациях. Проверка на соответствие техническому заданию проводилась с целью определения функциональной полноты разрабатываемого комплекса и соответствия требованиям качества результатов работы системы. Основными направлениями экспериментальных испытаний были исследования работы системы с входными данными, корректная работа базы данных.

Для проверки надежности и функциональности системы предлагаются следующие типы тестов.

1. Работа функции авторизации, проверка прав доступа разным группам пользователей: студент.
2. Работа функции выбора дисциплины, для прохождения тестирования.
3. Работа функции выбора темы тестирования по выбранной дисциплине.
4. Работа функции тестирования.
5. Работа функции просмотра результатов и даты прохождения предыдущих тестов.

Проверка функции тестирования.

1. Проверка функции начальной точки тестирования;
2. Проверка функции перехода со среднего уровня заданий на высокий;
3. Проверка функции перехода со среднего уровня заданий на низкий;
4. Проверка функции перехода с низкого уровня заданий на средний;
5. Проверка функции перехода с высокого уровня заданий на средний;
6. Проверка функции вывода результатов тестирования.

## Проверка программной системы в экстремальных условиях

## Проверка программной системы в исключительных ситуациях

Проверка в исключительных ситуациях выполняется при вводе данных и отсутствующим или ограниченным подключением к интернету.

Однако при работе модуля преподавателя нарушения работы связанные с техническими проблемами не является критичным.

## Основные выводы

# Техническая документация

## Руководство пользователя

Студенту предоставляется возможность прохождения адаптивного тестирования, просмотра результата ранее пройденного тестирования. Также студент имеет права чтения всех возможностей сайта. Имеет доступ к просмотру такой информации, как состав преподавателей и их расписание, групп и их расписание, темы лекций, лабораторных работ, вопросы к экзамену/зачету, задания РГР, курсовые работы для каждой дисциплины, а также о предполагаемых тестах и о количестве часов выделяемых на дисциплину.

## Руководство администратора

Коротко о всех файлах в проекте их назначение и что в них находится, куда все это записать на сервере и как создать БД в СУБД

Администратор имеет права доступа ко всем возможностям сайта. Он регистрирует пользователей. Имеет права доступа к внесению и редактированию информации, составляющей «каркас» тестирования, состав преподавателей и их расписание, групп и их расписание, темы лекций, лабораторных работ, вопросы к экзамену/зачету, задания РГР, курсовые работы для каждой дисциплины, а также информацию о предполагаемых тестах и о количестве часов выделяемых на дисциплину. Администратор может просматривать результаты ранее пройденных тестирований.

«Каркас» тестирования включает в себя вопросы низкого уровня сложности, вопросы среднего уровня сложности, вопросы высокого уровня сложности, время прохождения тестирования, количество всех вопросов на определенном уровне.

Результаты ранее пройденных тестирований содержат ФИО тестируемого, дату прохождения тестирования и результат.

Состав преподавателей включает в себя список преподавателей, работающий на кафедре, контактную информацию. Наличие возможности просмотра дисциплин преподавателя, а также все, что входит в дисциплину: темы лекций, задания лабораторных работ и РГР, вопросы к экзамену/зачету и темы курсовых работ.

Расписание преподавателей содержит информацию о наименовании дисциплины, времени проведения, группе и аудитории.

Информация о группе включает в себя список групп, и студентов входящих в нее. Расписание группы содержит информацию о наименовании дисциплины, преподавателе, времени проведения и аудитории.

Всю информацию о дисциплине (темы лекций, лабораторных работ, вопросы к экзамену/зачету, задания РГР, курсовые работы ) Администратор-Преподаватель может выложить как отдельным файлом, так и информацией на странице сайта.

# ЗаключЕние

# Список литературы

1. Java [Электронный ресурс].- Режим доступа: http://ru.wikipedia.org/wiki/Java
2. Гарин, М. Улучшенный 9-patch редактор [ Электронный ресурс ].- Режим доступа:http://habrahabr.ru/company/alee/blog/136667/
3. Лицензия Ruby [Электронный ресурс]. – Режим доступа: http://www.ruby-lang.org/en/about/license.txt
4. Степченко П. Пишем приложение для Android на Ruby (Ruboto) [Электронный ресурс]. – Режим доступа: http://habrahabr.ru/post/165071/
5. Оффициальная страница Android NDK [Электронный ресурс]. – Режим доступа: http://developer.android.com/tools/sdk/ndk/index.html
6. ACT-Тест Player [Электронный ресурс]. – Режим доступа: <http://www.call.lunn.ru/pdf/pr_otchoyt_I.pdf>
7. Буч Г., Якобсон А., Рамбо Дж. UML. Классика CS. 2-е изд. / Пер. с англ.; Под общей редакцией проф. С. Орлова — СПб.: Питер, 2006. — 736 с
8. Рудаков, А.В. Технология разработки программных продуктов / А.В. Рудаков. – М.: Издательский центр "Академия", 2006.

ПРИМЕРНОЕ СОДЕРЖАНИЕ ДИПЛОМА

1. Исследовательская часть
   * 1. Исследование алгоритмов адаптивного тестирования
     2. Алгоритм Аванесова!
     3. Разработка алгоритма адаптивного тестирования
2. Конструкторская часть
   * 1. Выбор средств разработки
     2. Архитектура программного средства
     3. Модель хранения данных (ER)
     4. Модель потоков данных
     5. Проектирование пользовательского интерфейса
     6. Алгоритм работы программной системы
3. Тестирование
   * 1. Назначение тестирования
     2. Тестирование в нормальных условиях
     3. Нагрузочное тестирование
     4. Тестирование в исключительных ситуациях
     5. Основные выводы
4. Техническая документация
   * 1. Руководство программиста
     2. Руководство пользователя
5. Экономическая часть
6. Организационная часть
7. Заключение
8. Список литературы