

УЗБЕКСКОЕ АГЕНТСТВО СВЯЗИ И ИНФОРМАТИЗАЦИИ
ТАШКЕНТСКИЙ УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ
ТЕХНОЛОГИЙ

Зав. Кафедрой

«___»_____2012г.

ВЫПУСКНАЯ
КВАЛИФИКАЦИОННАЯ РАБОТА БАКАЛАВРА

На тему:

РАЗРАБОТКА СИСТЕМЫ КОМПЬЮТЕРИЗИРОВАННОГО
АДАПТИВНОГО ТЕСТИРОВАНИЯ

Выпускник: Бендик В.А.

Руководитель: Каримова В.А.

Консультант по БЖД: Абдуллаева С.М.

Рецензент:

ТАШКЕНТ - 2012

УЗБЕКСКОЕ АГЕНТСТВО СВЯЗИ И ИНФОРМАТИЗАЦИИ
ТАШКЕНТСКИЙ УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

Факультет: Информационные технологии

Кафедра: Информационные технологии

Направления (специальность): 5521900 –« Информатика и информационные технологии»

У Т В Е Р Ж Д А Ю

Зав кафедрой _____

«_____» _____ 2012г.

ЗАДАНИЕ

на выпускную квалификационную работу

студента Бендика Виталия Анатольевича

1. Тема работы: *Разработка системы компьютеризированного адаптивного тестирования*
2. Утверждена приказом по университету № 212-07 от «2 марта 2012 года»
3. Срок сдачи законченной работы: *7 июня 2012 года*
4. Исходные данные к работе:
5. Содержание расчётно–пояснительной записи (перечень подлежащих разработке вопросов): .
6. Перечень графического материала: таблицы, пользовательские интерфейсы.
7. Дата выдачи задания:

Руководитель _____
(подпись)

Задание принял _____
(подпись)

8. Консультанты по отдельным разделам выпускной работы

Раздел	Ф.И.О руководителя	Подпись дата	
		Задание выдал	Задание получил
Основная часть	Каримова В.А.		
БЖД	Абдуллаева С.М.		

9. График выполнения работы

№	Наименование раздела работы	Срок выполнения	Отметка руководителя о выполнении
1			
2			
3			
4			
5			
6			

Выпускник _____
(подпись)

« _____ » _____ 2012 г.

Руководитель _____
(подпись)

« _____ » _____ 2012 г.

Аннотация

В выпускной квалификационной работе была исследована проблема разработки системы компьютеризированного адаптивного тестирования, были рассмотрены и изучены принципы построения и организации подобных систем, проведен обзор сторонних разработок и исследований в этой области. Результатом данной работы является программный продукт, позволяющий организовать компьютеризированное адаптивное тестирование.

Summary

In the graduation thesis the problem of development of computerized adaptive testing system was studied. The principles of building and organization computerized adaptive testing systems were considered. The result of the graduation thesis is software which allows organizing the computerized adaptive testing.

Аннотация

Битирув малакавий ишида компьютерлашган адаптив тестлаш тизимини яратиш муаммоси ўрганилган. Шунингдек, компьютерлашган адаптив тестлаш тизимини яратиш ва ташкил этиш принциплари қараб чиқилган. Битирув малакавий ишининг натижаси сифатида компьютерлашган адаптив тестлаш тизимининг дастурий таъминоти яратилган.

Содержание

	ВВЕДЕНИЕ.....	6
I.	ПРИМЕНЕНИЕ КОМПЬЮТЕРИЗИРОВАННОГО АДАПТИВНОГО ТЕСТИРОВАНИЯ В ОБРАЗОВАТЕЛЬНОМ ПРОЦЕССЕ.....	10
1.1	Контроль знаний с использованием информационно-коммуникационных технологий.....	10
1.2	Понятие адаптивного тестирования, компьютеризированного адаптивного тестирования, требования, предъявляемые к ним..	12
1.3	Обзор и анализ исследований и разработок в области компьютеризированного адаптивного тестирования.....	18
1.4	Постановка задачи.....	20
1.5	Выводы.....	21
II.	АЛГОРИТМЫ РАЗРАБОТКИ СИСТЕМЫ КОМПЬЮТЕРИЗИРОВАННОГО АДАПТИВНОГО ТЕСТИРОВАНИЯ.....	22
2.1	Модели, применяемые в системах компьютеризированного адаптивного тестирования.....	22
2.2	Определение назначения и функциональных возможностей модулей системы компьютеризированного адаптивного тестирования.....	28
2.2.1	Калиброванный пул вопросов и механизм их калибровки.....	28
2.2.2	Механизм отбора вопросов из пула.....	29
2.2.3	Механизм вычисления уровня подготовленности испытуемого	30
2.2.4	Механизм остановки процесса тестирования (STOP-правило)...	30
2.3	Выводы.....	31
III.	ПРОЕКТИРОВАНИЕ И РЕАЛИЗАЦИЯ СИСТЕМЫ КОМПЬЮТЕРИЗИРОВАННОГО АДАПТИВНОГО ТЕСТИРОВАНИЯ.....	32
3.1	Проектирование системы компьютеризированного адаптивного тестирования.....	32
3.1.1	Инструменты и технологии.....	32
3.1.2	Архитектура системы.....	32
3.1.2.1	Серверная часть.....	33
3.1.2.2	Клиентская часть.....	38
3.2	Разработка системы компьютеризированного адаптивного тестирования.....	48
3.2.1	Серверная часть.....	48
3.2.2	Клиентская часть.....	50
3.3	Выводы.....	53
IV.	БЕЗОПАСНОСТЬ ЖИЗНЕДЕЯТЕЛЬНОСТИ.....	54
	ЗАКЛЮЧЕНИЕ.....	68
	СПИСОК ЛИТЕРАТУРЫ.....	69

Введение

Исследования зарубежных учёных свидетельствуют о том, что традиционные методы обучения и контроля знаний имеют общую направленность на некоторого «среднего обучаемого», общую модель обучаемого, имеющую мало общего с реальными обучаемыми. Такая тенденция приводит к потере мотивации к обучению, так как сильные обучаемые не получают стимула для получения новых знаний и саморазвития, а слабые просто оказываются не в состоянии освоить представленный материал [5].

Однако, в последние годы, традиционные методы обучения и контроля начинают постепенно уступать более современным и гибким – адаптивное обучение, адаптивный контроль знаний. Адаптивное обучение, в отличие от традиционного, основывается на иных теоретико-методологических основах и технологиях конструирования учебных материалов и тестов и их предъявления [3, 5].

В нашей стране исследования в данной области пока ещё представлены явно недостаточно. Речь идёт не только о самой методологии адаптивного обучения и тестирования, но и о практической программной её реализации.

Актуальность исследования определяется необходимостью повышения эффективности процесса обучения и контроля, что, в свою очередь, связано с уменьшением числа заданий, времени, стоимости тестирования, повышением точности оценок, полученных испытуемыми по результатам выполнения теста. Кроме того, об актуальности данного исследования говорит недостаток программных средств, позволяющих в полной мере осуществлять адаптивное обучение и контроль знаний.

Целью исследования является обоснование, проектирование и разработка системы компьютеризированного адаптивного тестирования и внедрение в профессиональную деятельность преподавателей ВУЗов.

В результате выявился ряд вопросов, требующих конкретных решений и явившихся обоснованием для проведения работ по данной теме и

доказывающих ее научную ценность. Исходя из этого, определен круг решаемых задач данной работой.

Задачи исследования:

- Сделать обзор научно-методической, научно исследовательской литературы по исследуемой проблеме;
- Изучить теоретические основы построения систем компьютеризированного адаптивного тестирования;
- Определить наиболее целесообразные методы и программные инструментальные средства разработки системы компьютеризированного адаптивного тестирования;
- Сделать сравнительный анализ систем компьютеризированного адаптивного тестирования, применяемых в системе образования;
- Определить функциональные возможности и структуру системы компьютеризированного адаптивного тестирования;
- Разработать систему компьютеризированного адаптивного тестирования.

Объектом исследования является контроль знаний учащихся ВУЗов с применением системы компьютеризированного адаптивного тестирования.

Предметом исследования является проектирование и реализация системы компьютеризированного адаптивного тестирования.

Методологической основой исследования являются:

- Указы Президента «О дальнейшем развитии компьютеризации и информационно-коммуникационных технологий», «Концепции информатизации сферы образования Республики Узбекистан», «Концепции создания и развития системы дистанционного образования Республики Узбекистан» [1, 2];
- Работы ведущих ученых Узбекистана В.Кабулова, А.Абдукадырова, М.Арипова, У.Бегимкулова, Н.Тайлакова,

А.Хайитова, У.Юлдашева по вопросам компьютеризации образования и внедрения ИТ в учебный процесс;

- Теория адаптивного обучения, основанная на концепции об уровнях умственного развития (В.В. Богорев, Н.П. Капустин, А.В. Петровский, А.В. Хуторской, Е.А. Ямбург и др.);
- Концепция педагогического тестирования как совокупности методов измерения, обработки и интерпретации результатов обучения (В.С. Аванесов, В.П. Беспалько, Н.Ф. Ефремова, М.Б. Чельшкова и др.);

В ходе исследования была выдвинута и сформулирована **гипотеза исследования**: если разработать и внедрить в образовательный процесс ВУЗов систему компьютеризированного адаптивного тестирования, то в лучшую сторону изменится эффективность контроля знаний, что в конечном итоге положительно скажется на успеваемости и мотивации к обучению, как сильных, так и слабых учащихся.

Практическая значимость проведенного исследования заключается в том, что:

- разработана система компьютеризированного адаптивного тестирования;

Структура и объем работы: работа состоит из введения, трех глав, двенадцати параграфов, заключения, библиографии и приложений. Во введении обоснована актуальность работы, поставлена цель и задачи работы. В первой главе рассмотрен контроль знаний с использованием информационно-коммуникационных технологий, понятия адаптивного тестирования, компьютеризированного адаптивного тестирования, а также требования предъявляемые к ним; обзор сторонних разработок в данной области. Во второй главе рассмотрены модели применяемые в системах компьютеризированного адаптивного тестирования, определено назначение и функциональные возможности компонентов системы. В третьей главе рассмотрено проектирование системы компьютеризированного адаптивного

тестирования, разработка системы компьютеризированного адаптивного тестирования, а также применение системы компьютеризированного адаптивного тестирования для практического проведения контроля знаний. В четвертой главе рассмотрены основы безопасности жизнедеятельности.

I. ПРИМЕНЕНИЕ КОМПЬЮТЕРИЗИРОВАННОГО АДАПТИВНОГО ТЕСТИРОВАНИЯ В ОБРАЗОВАТЕЛЬНОМ ПРОЦЕССЕ

1.1 Контроль знаний с использованием информационно-коммуникационных технологий.

С начала XXI в, в образовании при проведении тестирования стали широко применяться компьютеры. В руках педагогов появился новый инструмент компьютерное тестирование, при котором предъявление тестов, оценивание результатов учащихся и выдача им результатов осуществляется с помощью ПК.

Компьютерное тестирование может проводиться в различных формах, различающихся по технологии объединения заданий в тест.

Первая форма — самая простая. Готовый тест, стандартизованный или предназначенный для текущего контроля, вводится в специальную оболочку, функции которой могут различаться по степени полноты. Обычно при итоговом тестировании оболочка позволяет предъявлять задания на экране, оценивать результаты их выполнения.

Вторая форма компьютерного тестирования предполагает автоматизированную генерацию вариантов теста, осуществляемую с помощью инструментальных средств. Варианты создаются перед экзаменом или непосредственно во время его проведения из банка калиброванных тестовых заданий с устойчивыми статистическими характеристиками. Калибровка достигается благодаря длительной предварительной работе по формированию банка, параметры заданий которого получают на репрезентативной выборке учащихся. Содержательная валидность и параллельность вариантов обеспечиваются за счет строго регламентированного отбора заданий каждого варианта в соответствии со спецификацией теста.

Третья форма — компьютерное адаптивное тестирование — базируется на специальных адаптивных тестах. В основе идей адаптивности лежат

соображения о том, что учащемуся бесполезно давать задания теста, которые он выполнит наверняка правильно без малейших затруднений или гарантированно не справится с ними в силу высокой трудности. Поэтому предлагается оптимизировать трудность заданий, адаптируя ее к уровню подготовленности каждого испытуемого, и сократить за счет исключения части заданий длину теста.

Компьютерное тестирование имеет определенные преимущества по сравнению с традиционным бланковым тестированием, которые проявляются особенно заметно при массовых проверках. Предъявление вариантов теста на компьютере позволяет сэкономить средства, расходуемые обычно на печать и транспортировку бланковых тестов.

Благодаря компьютерному тестированию можно повысить информационную безопасность и предотвратить рассекречивание теста. Упрощается также процедура подсчета результирующих баллов в тех случаях, когда тест содержит только задания с выбором ответов.

Другие преимущества компьютерного тестирования проявляются в текущем контроле, при самоконтроле и самоподготовке учащихся; благодаря компьютеру можно незамедлительно выдать тестовый балл и принять неотложные меры по коррекции усвоения нового материала на основе анализа протоколов по результатам выполнения корректирующих и диагностических тестов. Возможности педагогического контроля при компьютерном тестировании значительно увеличиваются за счет расширения спектра измеряемых умений и навыков в инновационных типах тестовых заданий, использующих многообразные возможности компьютера при включении аудио- и видеофайлов, интерактивности, динамической постановки проблем с помощью мультимедийных средств и др.

Благодаря компьютерному тестированию повышаются информационные возможности процесса контроля, появляется возможность сбора дополнительных данных о динамике прохождения теста отдельными

учащимися и для осуществления дифференциации пропущенных и не достигнутых заданий теста.

1.2 Понятие адаптивного тестирования, компьютеризированного адаптивного тестирования, требования, предъявляемые к ним.

Адаптивный тест – тест, в котором сложность заданий меняется в зависимости от правильности ответов испытуемого. Если обучаемый правильно отвечает на тестовые задания, сложность последующих заданий повышается, если неправильно – понижается. Также есть возможность задавать дополнительные вопросы по темам, которые обучаемый знает не очень хорошо для более точного выяснения уровня знаний в данных областях.

Адаптивное тестирование знаний – это способ экзаменационного контроля уровня подготовки обучаемого, при котором процедура выбора и предъявления ему очередного тестового задания на $(t+1)$ -м шаге тестирования определяется ответами обучаемого на предыдущих t шагах теста. Математическую основу такого учета составляет модель объединения тестовых заданий в тематические последовательности со взвешенным ранжированием как отдельных заданий, так и целых последовательностей и выведением итоговой оценки за тест с учетом нормированной суммы баллов, накапливаемой за выбранные обучаемым варианты ответов.

Появление адаптивного тестирования было вызвано стремлением к повышению эффективности педагогических измерений, которая, как правило, связывалась с уменьшением числа заданий, времени, стоимости тестирования, а также с повышением точности оценок учащихся. В основе адаптивного подхода лежит индивидуализация процедуры отбора заданий теста, которая за счет оптимизации трудности заданий применительно к уровню подготовленности обучаемых обеспечивает генерацию эффективных тестов [3].

Оптимизация трудности заданий обычно проводится пошагово. Если учащийся выполняет задание верно, то затем ему дается более трудное задание. При неправильном выполнении задания совершается отход назад к более легким заданиям банка. При невыполнении трех заданий подряд процесс останавливается и специальными методами (чаще всего с помощью теории IRT) определяется балл учащегося за выполненные задания по сформированному специально для него адаптивному тесту. Таким образом, в компьютерном адаптивном предъявлении число тестовых заданий и их трудность индивидуально подбираются для каждого экзаменуемого на основании его ответов, а индивидуальная совокупность заданий образует адаптивный тест. Адаптивные тесты в группе испытуемых состоят в основном из разных заданий и различаются по количеству и трудности заданий тем сильнее, чем больше разброс среди испытуемых тестируемой группы по подготовленности.

Получить одновременный прирост эффективности измерений по всем критериям невозможно, поэтому обычно при организации адаптивного тестирования на первый план выходит один, в лучшем случае, два критерия. Например, в одних случаях при экспресс-диагностике в адаптивном режиме наибольшее внимание уделяется минимизации времени испытания и количеству предъявляемых заданий, а вопросы точности оценок отходят на второй план. В других случаях приоритетной может быть точность измерения и тестирование каждого испытуемого продолжается до тех пор, пока не достигается запланированная минимальная ошибка измерения.

На длине адаптивного теста существенно сказывается качество структуры знаний учащихся. Обычно испытуемые с четкой структурой знаний выполняют задания нарастающей трудности, уточняя с каждым очередным верно выполненным заданием оценку подготовленности. Они выполняют небольшое число заданий адаптивного теста и быстро доходят до порога своей компетентности. Учащиеся с нечеткой структурой знаний, у которых чередуются верные и неверные ответы, получают колеблющиеся по

трудности задания. Процесс тестирования затягивается, поскольку при скачкообразном изменении трудности заданий не происходит пошагового нарастания точности измерения и число заданий, адаптированных по трудности, нередко оказывается даже большим, чем в обычном, традиционном тесте.

Преимущества адаптивного тестирования. К числу важных преимуществ компьютеризованного адаптивного тестирования можно отнести:

- высокую эффективность;
- высокий уровень секретности;
- индивидуализацию темпа выполнения теста;
- высокий уровень мотивации к тестированию у наиболее слабых обучающихся за счет исключения из процесса предъявления излишне трудных заданий;
- сообщение результата в интервальной шкале тестовых баллов каждому испытуемому незамедлительно, сразу после окончания его работы над индивидуально подобранным набором заданий в адаптивном тесте.

Стратегии адаптивного тестирования. Стратегии предъявления тестовых заданий в адаптивном тестировании можно разделить на двухшаговые и многошаговые, сообразно которым используется различная технология формирования адаптивных тестов. Двухшаговая стратегия предполагает наличие двух этапов. На первом этапе всем испытуемым выдается одинаковый входной тест, цель которого — осуществление предварительной дифференциации учащихся вдоль оси переменной измерения. По результатам дифференциации на втором этапе организуется адаптивный режим и строятся адаптивные тесты.

В результате развития теории IRT, обеспечивающей единую интервальную шкалу для оценок параметров испытуемых и трудности заданий теста, появилась возможность по-новому осуществить оптимизацию процедуры отбора заданий для моделирования эффективных адаптивных

тестов. Стали развиваться многошаговые стратегии адаптивного тестирования, в рамках которых в процессе выполнения наборов заданий каждый испытуемый движется по своей индивидуальной траектории.

Многошаговые стратегии адаптивного тестирования подразделяются на фиксировано-ветвящиеся и варьирующе-ветвящиеся в зависимости от того, как конструируются многошаговые адаптивные тесты. Если один и тот же набор заданий с их фиксированным расположением на оси трудности используется для всех испытуемых, но каждый учащийся движется по набору заданий индивидуальным путем в зависимости от результатов выполнения очередного задания, то стратегия адаптивного тестирования является фиксировано-ветвящейся.

Задания по трудности в наборе заданий обычно располагают на равном расстоянии друг от друга или выбирают убывающий шаг сообразно нарастанию трудности, что позволяет подстроить темп тестирования под испытуемого, поскольку по мере выполнения заданий у него нарастает утомление и снижается мотивация к выполнению заданий теста.

Варьирующе-ветвящаяся стратегия адаптивного тестирования предполагает отбор заданий непосредственно из банка по определенным алгоритмам, которые прогнозируют оптимальную трудность последующего задания по результатам выполнения испытуемым предыдущего задания адаптивного теста. Таким образом, шаг за шагом из отдельных заданий получается адаптивный тест. В нем варьирует не только трудность, но и шаг, определяемый разностью трудностей двух соседних заданий адаптивного теста. Отличительной особенностью варьирующей-ветвящейся стратегии адаптивного тестирования является пошаговая переоценка уровня подготовленности испытуемого, предпринимаемая после каждого выполнения очередного задания теста.

Вход и выход из адаптивного тестирования. Выбор начальных оценок для входа в адаптивное тестирование осуществляется по-разному, в зависимости от вида стратегии и имеющихся технологических возможностей

при генерации адаптивных тестов. Один из методов определения начальных оценок основан на выдаче испытуемым перед началом адаптивного тестирования входного претеста. В претест обычно включают 5— 10 заданий из различных разделов содержания, охватывающих по трудности весь диапазон предполагаемого расположения тестируемой выборки учащихся на оси переменной измерения.

Для выхода из режима тестирования либо вводят ограничения по времени или по числу заданий, либо задаются планируемой точностью измерений.

Надежность, валидность и длина теста при адаптивном тестировании. Так же как и при традиционном тестировании, отбор заданий в адаптивные тесты осуществляется в соответствии со спецификой теста. Оптимизируя трудность, можно лишь уменьшить число предъявляемых заданий по каждому разделу и сохранить при этом для каждого испытуемого содержательный план теста. Таким образом, адаптивное тестирование вне зависимости от стратегии предъявления заданий и их числа должно обеспечивать высокую содержательную валидность каждого генерируемого адаптивного теста.

Надежность в адаптивном тестировании зависит от совокупности факторов. К ним относятся: число заданий, наличие систематического контроля за частотой выбора заданий банка при генерации адаптивного теста. На надежность также влияют характеристики банка тестовых заданий, связанные с качеством измерений и качеством входного контроля.

Итак, адаптивное тестирование должно удовлетворять следующим требованиям:

1. Регулируемость пропорций предъявляемых легких, средних и трудных заданий в зависимости от числа правильных ответов тестируемого.
2. Регулируемость пропорций предъявляемых различных тематических разделов учебной программы в тесте.

3. Регулируемость уровня сложности предъявляемых тестов с учетом семантической компетенции тестируемого.
4. Включение адаптивного механизма перевода на более высокий уровень заданий на одном и том же уровне предъявляемых заданий
5. Каждое задание более высокого уровня оценивается более высокими баллами.

Адаптивные методы максимально используют данные из модели студента (например, уровень подготовленности студента, уровень беспокойства-тревоги, правильность ответа и др.) и/или модели учебного материала (например, взаимосвязи между проверяемыми понятиями) и позволяют организовать контроль индивидуально для каждого студента, поддерживая, например, оптимальный для студента уровень трудности выдаваемых контрольных заданий или формируя индивидуальные стратегии контроля по отдельной теме, разделу или курсу в целом [5].

Проверка знаний студентов может быть осуществлена с учетом различных критериев формирования оценки. В зависимости от этого методы оценки знаний можно разделить на три основных класса:

- на основе количественных критериев:
 - о простейшая модель;
 - о модели, учитывающие типы заданий;
 - о модели, учитывающие характеристики заданий;
 - о модели, учитывающие характеристики заданий и параметры КЗ.
- на основе вероятностных критериев:
 - о модели, учитывающие вероятность правильного ответа;
 - о модели, учитывающие неопределенность ответа.
- на основе классификационных критериев:
 - о модели на основе алгоритма вычисления оценок (АВО);
 - о модели на основе нечетких множеств.

В моделях на основе **количественных** критериев в качестве измерения обычно выступает числовое множество, то есть количественная шкала,

предназначенная для представления оценки числом. В данном случае предусматривается вычисление некоторой величины, которая затем, как правило, сравнивается с предварительно заданными граничными значениями, определяющими интервалы оценивания в принятой шкале измерения. Рассчитываемая величина может представлять собой, например, сумму баллов, полученных студентом за правильные ответы на контрольные задания. При этом начисление баллов может происходить как с учетом типа предлагаемых студенту контрольных заданий, их дидактических характеристик, так и с учетом параметров КЗ, используемых при формировании данной оценки [8, 12].

В моделях на основе **вероятностных** критериев главным являются утверждения о зависимости вероятности правильного ответа студента от уровня его подготовленности и от параметров задания. Модели данного типа также позволяют решать задачи диагностики (контроля), которые характеризуются необходимостью учета неопределенности ответов обучаемых [4].

Модели на основе **классификационных** критериев, то есть на основе определения класса принадлежности предусматривают отнесение студента к одному из устойчивых классов с учетом совокупности признаков, определяющих данного студента. При этом используется специальная процедура вычисления степени похожести (оценки) распознаваемой строки (совокупности признаков обучаемого) на строки, принадлежность которых к классам заранее известна [11].

1.3 Обзор и анализ исследований и разработок в области компьютеризированного адаптивного тестирования.

Проект «irt-cat» – IRT-Computerized Adaptive Testing

Адрес проекта: <http://irt-cat.sourceforge.net>

Проект направлен на обеспечение открытого доступа к Интернет версии программы компьютеризированного адаптивного тестирования (CAT),

основанной на 3-параметрической логистической модели или, иначе, модели Раша (Rasch).

Проект «Assessment Center»

Адрес проекта: <http://www.assessmentcenter.net>

Assessment Center – бесплатный онлайн-инструмент, который предоставляет функциональные возможности для создания и распространения адаптивных тестов.

Проект «CATSim»

Адрес проекта: <http://www.assess.com/xcart/product.php?productid=428>

CATSim – Симулятор компьютеризированного адаптивного тестирования

CATSim реализует три типа моделирования для компьютеризированного адаптивного тестирования (КАТ): постфактумное (на реальных данных), моделирование, гибридное моделирование и моделирование Монте-Карло. При реализации программы КАТ, все три типа моделирования могут быть использованы на различных этапах процесса разработки КАТ. Опции CATSim позволяют осуществлять все три типа моделирования. CATSim осуществляет моделирование для банков вопросов до 999 элементов, без ограничений на количество испытуемых для постфактумного и гибридного моделирования, и предел в 10000 испытуемых при моделировании методом Монте-Карло. Тем не менее, моделирование КАТ можно произвести, используя не более чем 200 испытуемых или даже меньше, если они адекватно представляют выборку, к которой КАТ будет применяться. CATSim реализует моделирование для всех трех двухпараметрических моделей IRT (Item Response Theory) и для пяти многопараметрических моделей IRT.

Проект «IRTtool.com»

Адрес проекта: <http://www.stat.ucla.edu/~jeroen/irttool.html>

IRTtool.com является веб-интерфейсом к пакету Димитриса Ризополоса. Это попытка сделать IRT моделирование более доступных для прикладных

исследователей. Приложение поддерживает моделирование 1 параметрической модель (Раша), 2-х и 3-х параметрических моделей IRT. Другие возможности приложения включают импорт и экспорт данных, построение графиков и экспорт в формате PDF.

Проект «OSCATS»

Адрес проекта: <http://code.google.com/p/oscats/>

Библиотека OSCATS реализует модели IRT и когнитивной диагностики (латентной классификации) и алгоритмы выбора элемента, используемые в компьютеризированном адаптивном тестировании (КАТ). OSCATS способствует разработке КАТ и моделированию КАТ, предоставляя готовый к использованию код, который можно использовать для выборки КАТ-элементов и классификационной оценки. Это расширяемый модульный фреймворк. Библиотека написана на объектно-ориентированном C с применением GObject, и имеет связи с Python, Perl, PHP и Java.

1.4. Постановка задачи

Разработать систему компьютеризированного адаптивного тестирования, удовлетворяющую следующим требованиям:

1. Простота и удобство использования
2. Гибкость и расширяемость
3. Защищенность
4. Регулируемость пропорций предъявляемых легких, средних и трудных заданий в зависимости от числа правильных ответов тестируемого.
5. Регулируемость пропорций предъявляемых различных тематических разделов учебной программы в тесте.
6. Включение адаптивного механизма перевода на более высокий уровень заданий на одном и том же уровне предъявляемых заданий

7. Каждое задание более высокого уровня должно оцениваться более высокими баллами.

1.5. Выводы

Система компьютеризированного адаптивного тестирования должна быть простой и удобной в использовании, чтобы у пользователей не возникало затруднений в её эксплуатации. Должна быть достаточно гибкой и расширяемой. Система такого рода должна быть защищенной от сбоев и злонамеренных пользователей, так как разрабатывается для применения в учебном процессе.

Система должна использовать максимально возможное количество информации об испытуемом, адаптироваться под уровень его подготовленности, регулировать пропорции предъявляемых легких, средних и сложных заданий, реализовывать адаптивный механизм повышения или понижения уровня сложности предъявляемых заданий, справедливо оценивать уровень знаний испытуемого.

II. АЛГОРИТМЫ РАЗРАБОТКИ СИСТЕМЫ КОМПЬЮТЕРИЗИРОВАННОГО АДАПТИВНОГО ТЕСТИРОВАНИЯ

2.1 Модели, применяемые в системах компьютеризированного адаптивного тестирования

Тестирование по методу цепочек вопросов

Автоматизированная система контроля знаний предлагает преподавателю использовать цепочную систему вопросов, когда несколько вопросов объединяются в фиксированную последовательность (цепочку) по некоторому смысловому признаку, определяемому преподавателем, а каждому вопросу в цепочке присваивается некоторый коэффициент важности данного вопроса в данной цепочке. Этот коэффициент изменяется от 0 и сумма коэффициентов вопросов в цепочке принимается равной 1. Смысл коэффициента раскрывается при обработке результатов тестирования: оценка за ответы на вопросы, объединенные в цепочку, выставляется в зависимости от важности вопросов, на которые были даны правильные ответы. Цепочка может содержать неограниченное число вопросов, объединенных по семантическому признаку внутри выбранной темы тестирования. Вырожденным случаем цепочной структуры является наличие в цепочке всего лишь одного вопроса. В этом случае коэффициент его важности, очевидно, устанавливается равным единице.

Автоматизированный контроль знаний по методике уточняющих вопросов

Концепция базируется на автоматизации методики уточняющих вопросов, широко используемой в педагогической практике для выявления глубины знаний обучаемого. Относительная важность задаваемых вопросов определяется их весовыми коэффициентами, учитываемыми при подведении результатов тестирования.

Каждая цепочка представляет собой последовательность близких по тематике вопросов, формулируемых для уточнения знаний экзаменуемого. Очередной вопрос в цепочке задается только после получения ответа на

предыдущий вопрос. В зависимости от стратегии тестирования, избираемой организатором контроля знаний, очередной вопрос в цепочке может предъявляться до первой ошибки (“строгий” преподаватель), либо экзаменуемому предоставляется возможность продемонстрировать максимум знаний, отвечая на все вопросы данной тематической последовательности.

В цепочку может объединяться неограниченное количество тематически близких вопросов. В вырожденном случае цепочка состоит из единственного вопроса, коэффициент важности которого устанавливается равным 1.

Процедура количественного оценивания знаний, выявленных в ходе тестирования, состоит из трех этапов. На первом рассчитываются баллы, набранные за правильные ответы в рамках каждой отдельной тематической последовательности. На втором этапе рассчитывается суммарный балл $S\Sigma$ за ответы на все вопросы теста с учетом количества цепочек вопросов, на которые экзаменуемый успел ответить за отведенное время. На третьем этапе определяется итоговая оценка знаний экзаменуемого. Для этого набранный им суммарный балл $S\Sigma$ проецируется на оценочную шкалу, имеющую вид $[0; I_1; I_2; I_3; 1]$, где $0 < I_1 < I_2 < I_3 < 1$ – границы интервальных диапазонов оценок, задаваемые преподавателем при организации тестирования.

Модель адаптивного тестирования с использованием элементов теории графов

Пусть на каждое тестовое задание имеется пять вариантов ответа, степень истинности которых задается по 5-значной семантической шкале вида $W_c = ["неправильно", "неопределенно", "неточно", "неполно", "правильно"]$. Всем ответам, кроме правильного, ставится в соответствие последующее задание со своим подмножеством ответов. На 1-м шаге тестирования предъявляется основной вопрос. Если на него дан неточный ответ, следующим задается уточняющий вопрос, причем подмножество ответов содержит как "более правильные" ("правильно", "неполно"), так и "менее правильные" ("неопределенно", "неправильно") ответы. Если и на

этот вопрос дан ответ, отличный от правильного, предъявляется несложный вопрос, затрагивающий основные понятия предметной области, охваченной тестом. Назовем его СТОП-вопросом, поскольку в случае получения на него ответа, отличного от правильного, тестирование может быть прекращено. В случае правильного ответа на СТОП-вопрос или дополнительные вопросы тестирование переходит к следующему основному вопросу. Таким образом, процесс тестирования представляет собой движение по ориентированному графу $\langle G \rangle$, вершины которого есть тестовые задания (вопросы), а дуги — переходы между ними. В графе тестирования можно выделить три слоя: I слой содержит основные вопросы, II слой — дополнительные, III слой — СТОП-вопросы. В случае правильного ответа на основной вопрос происходит переход к следующему основному вопросу.

Если обучаемый не дает правильного ответа на СТОП-вопрос, то преподаватель может прервать тестирование (для этого в АСПКЗ предусмотрена соответствующая настройка). Такую тактику целесообразно использовать, например, при рубежном контроле знаний. В случае же текущего контроля необходимо выявить все пробелы в знании предметной области обучаемым, поэтому тестирование целесообразно продолжать до окончания теста.

Итак, процедура тестирования конкретного обучаемого отображается на графе $\langle G \rangle$ в виде некоторого пути. Анализируя его, можно сформировать ряд критериев, позволяющих говорить об уровне знаний обучаемого в данной предметной области. Так, чем путь длиннее, чем больше дополнительных и СТОП-вопросов было предъявлено, тем ниже должна быть оценка. Далее: предъявление хотя бы одного СТОП-вопроса лишает обучаемого надежды на оценки "хорошо" и "отлично"; если дополнительных более 10% от количества основных, то оценка не может быть выше, чем "хорошо" [16].

Модель Раша

Теория IRT позволяет установить связь между уровнем знаний испытуемых и результатами выполнения тестов, что позволяет определить уровень знаний независимо от сложности заданий. В качестве меры трудности заданий и меры уровня знаний применяется логит. Г.Раш ввел две меры: «логит уровня знаний» и «логит уровня трудности задания». Первую он определил как натуральный логарифм отношения доли правильных ответов испытуемого, на все задания теста, к доле неправильных ответов, а вторую – как натуральный логарифм другого отношения – доли неправильных ответов на задание теста к доле правильных ответов на то же задание, по множеству испытуемых. Единая логарифмическая шкала позволяет установить требуемое соответствие между уровнем обученности и трудностью задания и, более того, произвести коррекцию результатов тестирования при тестах разной сложности [3, 17].

В IRT вводится основное предположение о существовании некоторой взаимосвязи между наблюдаемыми результатами тестирования и латентными качествами испытуемых, выполняющих тест. Предполагается, что каждому испытуемому ставится в соответствие только одно значение латентного параметра. Элементы первого множества — это уровни знаний N испытуемых Θ_i , где $i = 1, \dots, N$. Второе множество образуют значения латентного параметра δ_j , $j = 1, \dots, N$, равные трудностям n заданий теста. На практике решается задача: по ответам испытуемых на задания теста оценить значения латентных параметров Θ и δ .

Основной математической моделью IRT является однопараметрическая логистическая функция Раша, называемая характеристической кривой, имеющая вид:

$$P_j = \frac{1}{1 + \exp(\delta_j - \theta)}; \quad (2.1)$$

где P_j - вероятность правильного ответа испытуемых любого уровня подготовленности на задание определённого уровня трудности под номером j ;

δ - уровень трудности конкретного, j -го задания проектируемого теста;

θ - уровень знаний, латентная переменная;

Чем выше крутизна функции P , тем уже интервал, на котором это задание работает. Таким образом, возникла мысль об улучшении модели Раша за счет введения в выражение второго параметра a_j . Параметр a_j даёт информацию о задании с точки зрения оценки его дифференцирующей способности, на заданном интервале. Геометрически значение параметра выражается крутизной характеристической кривой, аналитически – значением производной функции в точке перегиба. После введения в выражение параметра a_j получается двухпараметрическая модель:

$$P_j = \frac{1}{1 + e^{-a_j(\delta_j - \theta)}}; \quad (2.2.)$$

Эмпирические пределы значений для параметра a_j - от минус 2,80 до плюс 2,80. Возможно также добавление параметра C , определяющего вероятность угадывания правильного ответа.

Возникает проблема адекватной оценки трудности тестовых заданий и начального уровня знаний студентов. Начальная оценка уровня знаний испытуемого определяется по формуле

$$\theta_i = \ln\left(\frac{p_i}{q_i}\right) \quad (2.3)$$

, где p_i - доля правильных ответов i -го испытуемого, q_i - доля неправильных ответов. Аналогично определяется начальная оценка уровня трудности задания теста

$$\delta_j = \ln\left(\frac{q_j}{p_j}\right) \quad (2.4)$$

где p_j - доля правильных ответов на j -е задание теста, q_j - доля неправильных ответов.

В силу действия различных случайных факторов оценки параметров θ и δ , полученные на нескольких выборках, будут, конечно, различаться. Если объем выборки достаточно велик, то можно ставить вопрос о вычислении устойчивых значений параметров θ и δ , которые будут наиболее эффективными оценками и могут быть приняты в качестве объективных оценок параметров θ и δ .

При любом проведении процесса тестирования результаты вычисления $\tilde{\theta}_i$ - статистических оценок θ_i , и $\tilde{\delta}_j$ - статистических оценок δ_j будут отличаться от существующих точных значений. По своему смыслу оценки являются определенными функциями исходных случайных значений элементов матрицы ответов A_{nk} , состоящей из N - строк и K –столбцов и поэтому сами являются случайными величинами. Таким образом, возникает вопрос о нахождении математических ожиданий и дисперсий этих случайных величин. Необходимо чтобы математическое ожидание соответствующих оценок совпадало с соответствующими точными значениями, а дисперсия оценки была бы минимальной. Статистическая оценка уровня подготовленности и уровня трудности будут являться несмещенными оценками, если их математическое ожидание при любом объеме выборки испытуемых будет равно самому оцениваемому параметру. На практике обычно используют асимптотически несмещенную оценку, математическое ожидание которой стремится к истинному значению оцениваемого параметра, при неограниченном увеличении объема выборки. Статистическая оценка эффективна если при заданной выборке, она имеет возможную наименьшую дисперсию D^* при неполной информации,

возможно, получить лишь оценку с $D > D^*$. Если отношение $D/D^* \rightarrow 1$, при увеличении выборки, то оценка называется асимптотически эффективной. Статистическая оценка состоятельна, если несмещенная оценка не является эффективной, но при увеличении объема выборки ее дисперсия уменьшается. Несмещенность, эффективность и состоятельность являются независимыми свойствами, характеризующими оценки с разных сторон. Задача отыскания эффективных несмещенных оценок имеет особо важное значение при обработке результатов малых выборок испытуемых. Для получения оценки параметров δ и θ применяются метод моментов или метод наибольшего правдоподобия.

2.2 Определение назначения и функциональных возможностей модулей системы компьютеризированного адаптивного тестирования

Любая система компьютеризированного адаптивного тестирования должна иметь в своём составе следующие компоненты:

1. Калиброванный пул вопросов и механизм их калибровки
2. Механизм отбора вопросов из пула
3. Механизм вычисления уровня подготовленности испытуемого
4. Механизм остановки процесса тестирования (STOP-правило)

2.2.1 Калиброванный пул вопросов и механизм их калибровки

Наиболее важной частью системы компьютеризированного адаптивного тестирования является калиброванный пул вопросов, включающий в себя полный спектр вопросов различного уровня сложности. Успех любой системы компьютеризированного адаптивного тестирования во многом зависит от качества пула вопросов, которое можно определить двумя основными критериями:

- 1) Общее количество вопросов в пуле должно быть достаточным для предоставления вопросов на протяжении всей сессии проведения теста;

- 2) Вопросы в пуле должны иметь характеристики, предоставляющие адекватную информацию об уровне знаний испытуемого. Эти характеристики предоставляют наибольший интерес для разработчиков тестов.

Параметры вопросов, входящих в пул, зависят от выбранной IRT-модели выбора следующего вопроса из пула и модели вычисления уровня подготовленности испытуемого.

2.2.2 Механизм отбора вопросов из пула

В теории IRT процедура отбора вопросов – это процесс отбора вопроса из пула вопросов, для последующего предоставления их испытуемому. Разумно предположить, что каждый испытуемый, отвечая на тестовый вопрос обладает некоторыми знаниями (базой). Таким образом, можно сделать вывод, что каждый испытуемый имеет численное значение своей подготовленности или иначе, оценку, которая располагает его на определённом месте шкалы подготовленности. Обозначим эту оценку через θ . На каждом уровне шкалы подготовленности, существует определенная вероятность того, что испытуемый даст правильный ответ на вопрос. Обозначим эту вероятность через P . В случае обычного тестового вопроса эта вероятность будет мала для испытуемых с низким уровнем подготовленности и высока для испытуемых с высоким уровнем подготовленности.

Выбор одного из основных классов моделей IRT определяется тем, как оцениваются ответы на вопросы. Вопросы, на которые возможен только правильный или неправильный ответ моделируются двухпараметрическими моделями IRT. Вопросы, на которые возможны и другие ответы (например, почти правильно) моделируются многопараметрическими моделями IRT.

Функция информативности тестового вопроса (Item Information Function, IIF) также имеет важное значение в процессе отбора вопроса из пула. Она даёт информацию о вопросе, который будет представлен испытуемому во время адаптивного тестирования. Для выбора вопроса соответствующего

уровню подготовленности испытуемого. Функция информативности тестового вопроса должна быть вычислена для всех вопросов теста и вопрос с наибольшим значением данной функции должен быть представлен испытуемому. Это даёт дополнительную информацию об уровне подготовленности испытуемого и определяется по формуле:

$$I_i(\theta) = P_i(\theta)(1-P_i(\theta))$$

где $P_i(\theta)$ вероятность правильного ответа на вопрос i , для ответа на который необходим уровень подготовленности θ

2.2.3 Механизм вычисления уровня подготовленности испытуемого

После представления испытуемому очередного вопроса и его оценке, вычисляется уровень подготовленности испытуемого, который, далее, используется процедурой отбора вопроса из пула, для выбора следующего вопроса. Чаще всего для вычисления уровня подготовленности испытуемого применяется метод максимального правдоподобия. Подобно оценке параметров каждого вопроса из пула (собственно, калибровка вопроса), эта процедура имеет итеративный характер. За начальное значение уровня подготовленности испытуемого берётся некоторое заранее определённое значение.

2.2.4 Механизм остановки процесса тестирования (Стоп-правило)

Одной из важных характеристик системы компьютеризированного адаптивного тестирования это критерий остановки процесса тестирования (Стоп-правило). Критерий остановки процесса тестирования, как правило основан на точности, с которой был определён уровень подготовленности испытуемого. Во многих системах компьютеризированного адаптивного тестирования окончание процесса тестирования может быть основано на количестве представленных испытуемому вопросов, точности (количество знаков после запятой) вычисления уровня подготовленности или их комбинации. Точность вычисления уровня подготовленности испытуемого

обычно основана на ошибке (погрешности) связанной с ней. Стандартная ошибка, при данном уровне подготовленности испытуемого вычисляется путём суммирования значений функции информативности тестовых вопросов при данном уровне подготовленности испытуемого. Таким образом мы получаем уровень информативности теста $TI(\theta)$.

$$TI(\theta) = \sum_{i=1}^N I_i(\theta) \quad (2.2.4.1)$$

Далее, вычисляется стандартная ошибка по следующей формуле:

$$SE(\theta) = \frac{1}{\sqrt{TI(\theta)}} \quad (2.2.4.2)$$

Стандартная ошибка при данном уровне подготовленности испытуемого, вычисляется после каждого представления вопроса испытуемому для того, чтобы определить выбирать новый вопрос из пула вопросов или же прекратить тестирование.

2.3 Выводы

Системы адаптивного тестирования строятся на различных математических моделях. Каждая модель по-своему влияет на структуру и компоненты системы. Поэтому выбор модели адаптивного тестирования является самым важным на раннем этапе проектирования системы. В данной главе были рассмотрены различные модели адаптивного тестирования.

При реализации системы компьютеризированного адаптивного тестирования описываемой в данной работе, в качестве наиболее подходящей модели была выбрана модель Раша, основанная на теории IRT.

Использование системы в образовательных целях подразумевает использование её при тестировании многих испытуемых. Пул вопросов должен, при необходимости, легко обновляться и дополняться. В связи с

этим было принято решение о применении клиент-серверной архитектуры при разработке данной системы.

Компоненты системы, отвечающие за хранение вопросов и их отбор, было решено вынести в серверную часть системы, а механизм определения уровня подготовленности испытуемого и механизм остановки процесса тестирования в клиентскую часть.

III. ПРОЕКТИРОВАНИЕ И РЕАЛИЗАЦИЯ СИСТЕМЫ КОМПЬЮТЕРИЗИРОВАННОГО АДАПТИВНОГО ТЕСТИРОВАНИЯ

3.1 Проектирование системы компьютеризированного адаптивного тестирования

3.1.1 Инструменты и технологии

В ходе разработки системы компьютеризированного адаптивного тестирования были применены следующие инструменты и технологии.

1. Языки:

PHP – сервер

Java (Android) – клиент

2. Платформы:

Windows или Linux (для серверной части)

Android (для клиентской части)

3. Технологии:

http-протокол, html, xml

3.1.2 Архитектура системы

В качестве архитектуры системы компьютеризированного адаптивного тестирования была выбрана клиент-серверная, как наиболее подходящая в плане разделения компонентов системы там, где они будут функционировать наиболее эффективно.

В качестве клиент-серверной конфигурации была выбрана двухуровневая модель, в которой клиент обращается к серверу за необходимой информацией.

Диалоговые компоненты (PS – Presentation Services, PL – Presentation Logic), а также бизнес-логика (BL – Business Logic) были реализованы в клиентской части системы, а компоненты управления данными (DS – Data Services) и логика управления данными (DL – Data Logic) в серверной части системы (рис. 3.1).

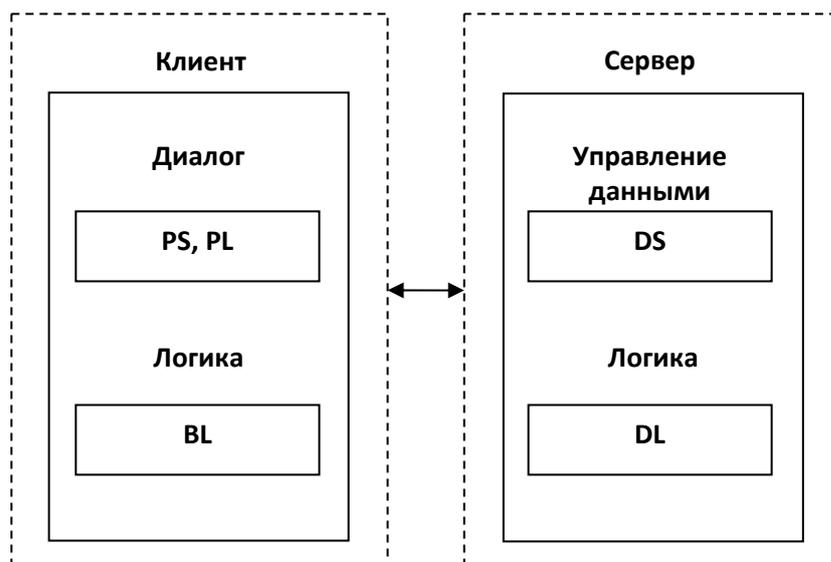


Рисунок 3.1. Архитектура системы

В качестве серверной части системы был выбран сервер на базе Apache с установленным на него php в связке с СУДБ MySQL.

В качестве клиентской части была выбрана мобильная платформа Android, так как разработка под эту платформу является актуальной, в связи с всё большим распространением мобильных устройств на базе этой платформы.

3.1.2.1 Серверная часть

Серверная часть разработанной системы была реализована на языке программирования php. В качестве Фреймворка был выбран Code Igniter. В качестве используемой СУБД была выбрана MySQL. Благодаря использованию вышеуказанного Фреймворка серверная часть разрабатывалась с использованием модели Model-View-Controller (рис. 3.2).

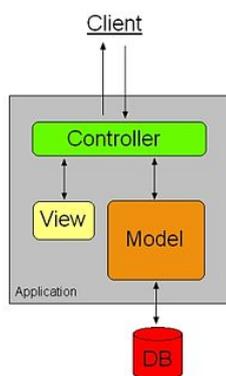


Рисунок 3.2. Концепция MVC-модели

Концепция MVC была описана в 1979 году Трюгве Реенскаугом, тогда работающем над языком программирования Smalltalk в Xerox PARC. Оригинальная реализация описана в статье «Applications Programming in Smalltalk-80: How to use Model-View-Controller» [23].

В оригинальной концепции была описана сама идея и роль каждого из элементов модели, представления и контроллера. Но связи между ними были описаны без конкретизации. Кроме того, различали две основные модификации:

Пассивная модель — модель не имеет никаких способов воздействовать на представление или контроллер, и используется ими в качестве источника данных для отображения. Все изменения модели отслеживаются контроллером и он же отвечает за перерисовку представления, если это необходимо. Такая модель чаще используется в структурном программировании, так как в этом случае модель представляет просто структуру данных, без методов их обрабатывающих.

Активная модель — модель оповещает представление о том, что в ней произошли изменения, а представления, которые заинтересованы в оповещении, подписываются на эти сообщения. Это позволяет сохранить независимость модели как от контроллера, так и от представления.

Классической реализацией паттерна MVC принято считать версию именно с активной моделью.

С развитием объектно-ориентированного программирования и понятия о шаблонах проектирования было создано ряд модификаций концепции MVC, которые при реализации у разных авторов могут отличаться от оригинальной.

Концепция MVC позволяет разделить данные, представление и обработку действий пользователя на три отдельных компонента:

Модель (англ. Model). Модель предоставляет знания: данные и методы работы с этими данными, реагирует на запросы, изменяя своё состояние. Не содержит информации, как эти знания можно визуализировать.

Представление, вид (англ. View). Отвечает за отображение информации (визуализацию). Часто в качестве представления выступает форма (окно) с графическими элементами.

Контроллер (англ. Controller). Обеспечивает связь между пользователем и системой: контролирует ввод данных пользователем и использует модель и представление для реализации необходимой реакции.

Важно отметить, что как представление, так и контроллер зависят от модели. Однако модель не зависит ни от представления, ни от контроллера. Тем самым достигается назначение такого разделения: оно позволяет строить модель независимо от визуального представления, а также создавать несколько различных представлений для одной модели.

Для реализации схемы Model-View-Controller используется достаточно большое число шаблонов проектирования (в зависимости от сложности архитектурного решения), основные из которых «наблюдатель», «стратегия», «компоновщик».

Наиболее типичная реализация отделяет вид от модели путем установления между ними протокола взаимодействия, используя аппарат событий (подписка/оповещение). При каждом изменении внутренних данных в модели она оповещает все зависящие от неё представления, и представление обновляется. Для этого используется шаблон «наблюдатель». При обработке реакции пользователя вид выбирает, в зависимости от нужной реакции, нужный контроллер, который обеспечит ту или иную связь с моделью. Для этого используется шаблон «стратегия», или вместо этого может быть модификация с использованием шаблона «команда». А для возможности однотипного обращения с подобъектами сложно-составного

иерархического вида может использоваться шаблон «компоновщик». Кроме того, могут использоваться и другие шаблоны проектирования, например, «фабричный метод», который позволит задать по умолчанию тип контроллера для соответствующего вида.

Итак, серверная часть состоит из 4-х контроллеров

1. **Admin** – администрирование системы
2. **Auth** – авторизация пользователей
3. **User** – пользовательская информация
4. **Test** – тестирование испытуемых

Контроллер Admin. Реализует функционал администрирования системы компьютеризированного адаптивного тестирования. Позволяет администратору создавать, удалять, изменять адаптивные тесты, группы вопросов, а также управлять пользователями.

Контроллер Auth. Реализует авторизацию пользователя в системе компьютеризированного адаптивного тестирования.

Контроллер User. Реализует функционал для предоставления пользователям информации о своих результатах.

Контроллер Test. Реализует функционал адаптивного тестирования системы.

Описание базы данных

При проектировании базы данных необходимо решить вопрос о наиболее эффективной структуре данных. Основные цели, которые при этом преследуются: обеспечить быстрый доступ к данным в таблицах; исключить ненужное повторение данных, которое может явиться причиной ошибок при вводе и нерационального использования дискового пространства; обеспечить целостность данных таким образом, чтобы при изменении одних

объектов автоматически происходило соответствующее изменение связанных с ними других объектов.

Таблица 3.1. Таблица пользователей

Поле	Тип	Описание
id	int(11) <i>Auto Increment</i>	Индекс
login	varchar(27)	Логин
name	varchar(100)	Имя
lastName	varchar(100)	Фамилия
pass	varchar(32)	Пароль
email	varchar(255)	Электронный ящик
ip	varchar(16)	IP-адрес

Таблица 3.2. Таблица тестов

Поле	Тип	Описание
id	int(11) <i>Auto Increment</i>	Индекс
test_title	text	Название теста
content	text	Описание
minScore	double	Минимальный балл
maxScore	double	Максимальный балл
reExam	double	Возможность передачи

Таблица 3.3. Таблица вопросов

Поле	Тип	Описание
id	int(11) <i>Auto Increment</i>	Индекс
testId	int(11)	Индекс теста
question	text	Текст вопроса
complexity	int(1)	Сложность

Таблица 3.4. Таблица ответов

Поле	Тип	Описание
id	int(11) <i>Auto Increment</i>	Индекс
questionId	int(11)	Индекс вопроса
answer	text	Текст ответа
isRight	int(1)	Правильность ответа

3.1.2.2 Клиентская часть

Клиентская часть разработанной системы была реализована на языке программирования Java. В качестве платформы была выбрана мобильная платформа Android. В качестве используемой локальной СУБД была выбрана SQLite, так как является единственной СУБД реализованной для данной платформы.

Выбор мобильной платформы Android для реализации клиентской части системы компьютеризированного адаптивного тестирования была определенная следующими факторами.

Android — портативная (сетевая) операционная система для коммуникаторов, планшетных компьютеров, цифровых проигрывателей, наручных часов, нетбуков и смартфонов, основанная на ядре Linux. Изначально разрабатывалась компанией Android Inc., которую затем купила Google. Впоследствии Google инициировала создание альянса Open Handset Alliance (ОНА), который сейчас и занимается поддержкой и дальнейшим развитием платформы. Android позволяет создавать Java-приложения, управляющие устройством через разработанные Google библиотеки. Android Native Development Kit создаёт приложения, написанные на Си и других языках [22].

Приложения для Android являются программами в нестандартном байт-коде для виртуальной машины Dalvik.

Google предлагает для свободного скачивания инструментарий для разработки (Software Development Kit), который предназначен для x86-машин под операционными системами Linux, Mac OS X (10.4.8 или выше), Windows XP, Windows Vista и Windows 7. Для разработки требуется JDK 5 или более новый.

Разработку приложений для Android можно вести на языке Java (не ниже Java 1.5). Существует плагин для Eclipse — «Android Development Tools» (ADT), предназначенный для Eclipse версий 3.3-3.7. Для IntelliJ IDEA также существует плагин, облегчающий разработку Android-приложений.

Для среды разработки NetBeans IDE разработан плагин, который начиная с версии Netbeans 7.0 перестал быть экспериментальным, тем не менее пока не является официальным. Кроме того существует Motodev Studio for Android, которая представляет собой комплексную среду разработки, основанную на базе Eclipse и позволяет работать непосредственно с Google SDK.

Кроме того в 2009 году в дополнение к ADT был опубликован Android Native Development Kit (NDK), пакет инструментариев и библиотек позволяющий вести разработку приложений на языке C/C++. NDK рекомендуется использовать для разработки участков кода критичных к скорости.

Доступные библиотеки:

- Bionic (библиотека стандартных функций, несовместимая с libc);
- libc (стандартная системная библиотека языка Си);
- мультимедийные библиотеки (на базе PacketVideo OpenCORE; поддерживают такие форматы, как MPEG4, H.264, MP3, AAC, AMR, JPG и PNG);
- SGL (движок двухмерной графики);
- OpenGL ES 1.0 ES 2.0 (движок трёхмерной графики);
- Surface Manager (обеспечивает для приложений доступ к 2D/3D);
- WebKit (готовый движок для Web-браузера; обрабатывает HTML, JavaScript);
- FreeType (движок обработки шрифтов);
- SQLite (легковесная СУБД, доступная для всех приложений);
- SSL (протокол, обеспечивающий безопасную передачу данных по сети).

По сравнению с обычными приложениями Linux, приложения Android подчиняются дополнительным правилам:

- Content Providers — обмен данными между приложениями;

- Resource Manager — доступ к таким ресурсам, как файлы XML, PNG, JPEG;
- Notification Manager — доступ к строке состояния;
- Activity Manager — управление активными приложениями.

Для Android был разработан формат установочных пакетов .apk.

Достоинства ОС Android

1) Некоторые обозреватели отмечают, что Android проявляет себя лучше одной из своих конкуренток, Apple iOS, в ряде особенностей, таких как: веб-сёрфинг, интеграция с сервисами Google Inc. и прочих. Android, в отличие от iOS является открытой платформой, что позволяет реализовать на ней больше функций.

2) В отличие от iOS и Windows Phone 7, в Android полноценная реализация Bluetooth стека, позволяющая в том числе передачу и приём файлов. Присутствует реализация FTP-сервера, режима точки доступа к сети (службы PAN) и групповой одноранговой сети через Bluetooth (службы GN).

3) В Android аппаратах как правило присутствует MicroSD-кардридер, позволяющий быстрый перенос файлов с компьютера на телефон, минуя скоростные ограничения USB и других способов передачи без извлечения карты памяти; кроме того, в iOS и Windows Phone 7 невозможна прямая передача каких-либо файлов в/из телефона, кроме как через программы синхронизации (iTunes и Zune), в то время как телефоны на Android экспортируют файловую систему карты памяти как USB mass storage device (как «флешку»).

4) Несмотря на изначальный запрет на установку программ из «непроверенных источников» (например, с карты памяти), это ограничение отключается штатными средствами в настройках аппарата, что позволяет легко устанавливать программы на телефоны и планшеты без интернет-подключения (например пользователям, не имеющим Wi-Fi-точки доступа и не желающим тратить деньги на мобильный интернет, который обычно стоит слишком дорого), а также позволяет всем желающим бесплатно писать

приложения для Android и тестировать на своём аппарате, в то время как для iOS и Windows Phone 7 пришлось бы покупать учётную запись разработчика, даже если вы не собираетесь в дальнейшем распространять свои собственные программы. С другой стороны, возможность установки программ минуя маркет позволяет легко использовать пиратские (взломанные либо просто скопированные, в зависимости от наличия в программе проверки лицензии) копии программ, что снижает привлекательность платформы для разработчиков платных приложений (в основном игр). Впрочем, покупатель будет более склонен к покупке программы после того, как он её проверит на своём аппарате в работе, чем если ему предлагают купить программу «вслепую».

5) Android доступен для различных аппаратных платформ, таких как ARM, MIPS, x86.

Клиентская часть разбита на 4 основных пакета

1. **auth** – авторизация на сервере
2. **dashboard** – главное окно программы
3. **results** – результаты тестирования
4. **test** – процесс тестирования

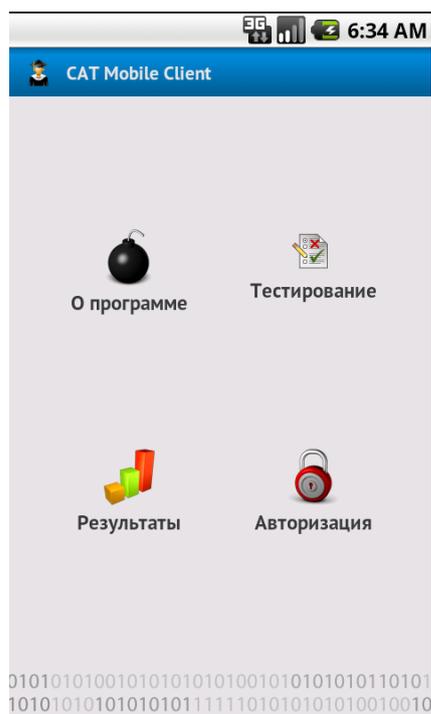


Рисунок 3.3. Клиентское приложение

Пакет Auth. Содержит лишь один элемент – активити (окно), отвечающую за получение и отправку на сервер логина и пароля.

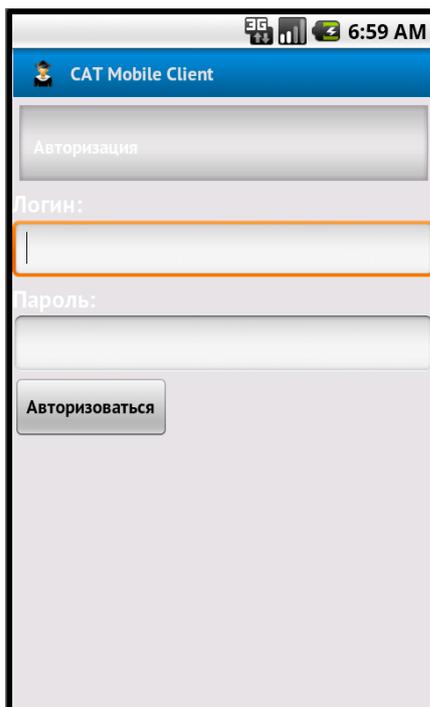


Рисунок 3.4. Окно авторизации

Пакет Dashboard. Содержит в себе 2 элемента, активити, отображающее меню клиентской части и лэйаут (схема размещения компонентов интерфейса) для адаптирующегося расположения пунктов меню в зависимости от ширины экрана и ориентации мобильного устройства.



Рисунок 3.5. Главное окно приложения

Пакет Results. Содержит в себе 2 элемента, активити, отображающее результаты тестирования испытуемого и адаптер (механизм отображения определённых данных в определённом интерфейсе) для отображения каждого результата.

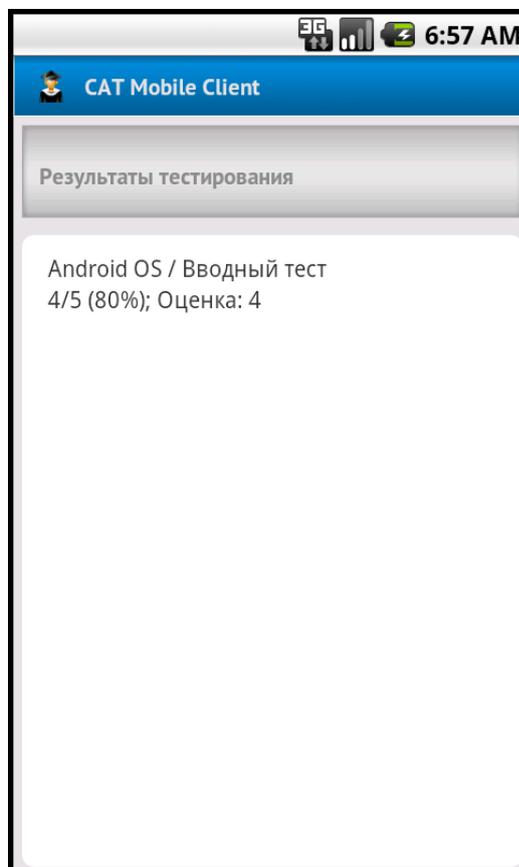
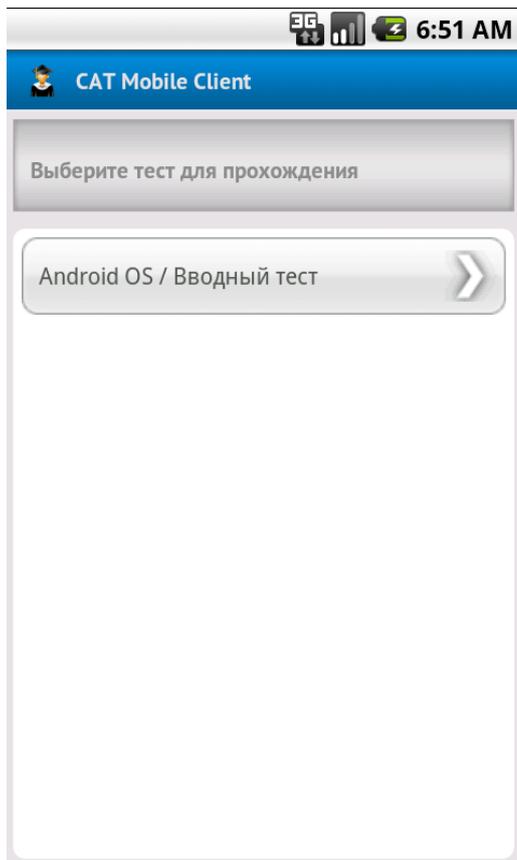
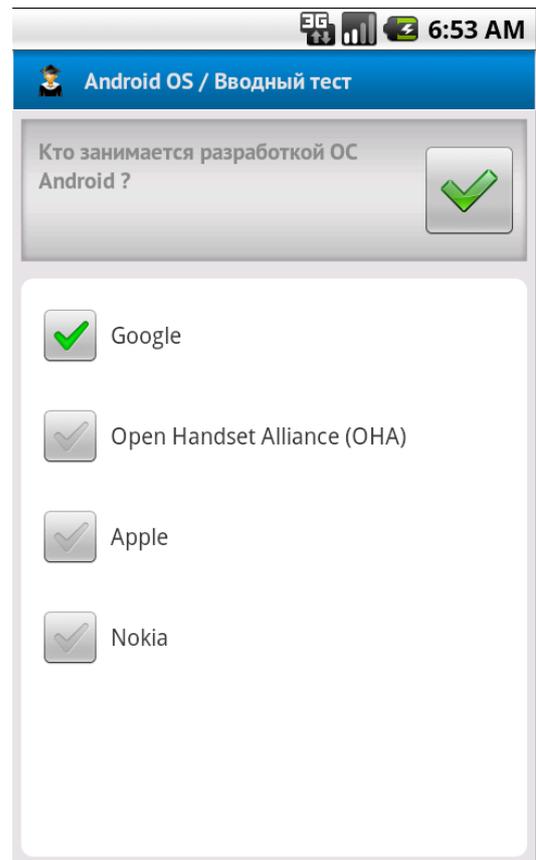


Рисунок 3.6. Окно результатов тестирования

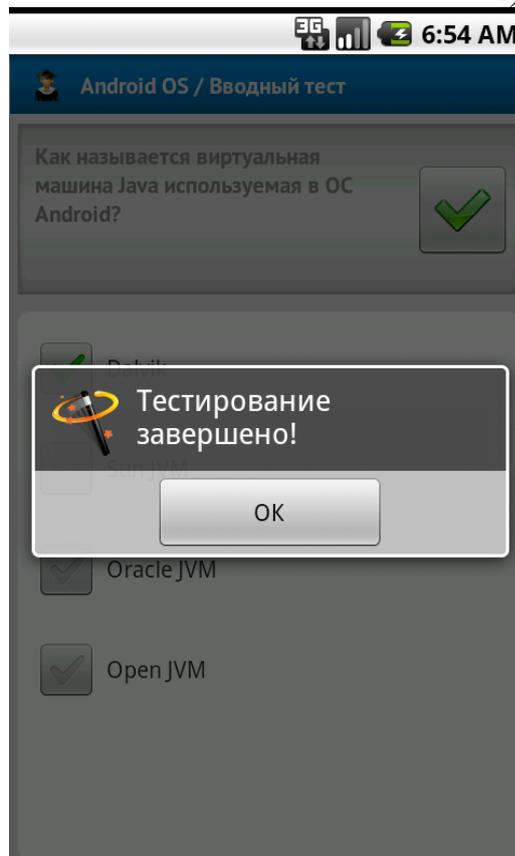
Пакет Test. Самый большой пакет клиенткой части. Содержит в себе 10 элементов, среди которых: вспомогательный класс для работы с локальной СУБД, класс реализующий интерфейс взаимодействия с сервером, активити, для проведения самого тестирования, активити для отображения списка тестов, которые можно пройти, классы содержащие необходимую информацию и методы для хранения и передачи между сервисами клиентской части тестов, вопросов, ответов, результатов тестирования. Адаптеры, для отображения списка тестов и вопроса с ответами (во время тестирования),



а)



б)



в)

Рисунок 3.7. Окно тестирования. а) выбор теста для прохождения; б) выбор ответа на тест; в) окончание тестирования

Описание локальной базы данных

Клиентская часть использует локальную СУБД SQLite, для промежуточного хранения данных.

Локальная встраиваемая система управления базами данных — архитектура систем управления базами данных, когда СУБД тесно связана с прикладной программой и работает на том же компьютере, не требуя профессионального администрирования.

Встраиваемые СУБД применяются во многих программах, которые хранят большие массивы данных, но при этом не требуется доступ с многих компьютеров. На «рабочем столе» неопытного пользователя тоже есть программы, в которых может найтись встраиваемая СУБД: почтовые клиенты и мессенджеры (базы переписки), медиапроигрыватели (плейлисты и обложки), различные локальные БД наподобие телефонных справочников и геоинформационных систем (предоставляемые данные).

Исторически локальные и файл-серверные СУБД предоставляли скриптовый язык, на котором пользователь мог писать прикладную программу. Так устроены Microsoft Access, FoxPro, Clipper, первые версии 1С: Бухгалтерии. Недостатком этого подхода была крайняя бедность результирующих программ, ограниченные средства отладки. И зачастую не существовало компактной среды исполнения, которую можно распространять вместе с программой; нужна программа — устанавливай весь пакет. С распространением динамической линковки и opensource-сообщества маятник качнулся в другую сторону: пусть программист пишет свою программу на том языке высокого уровня, на котором удобно. СУБД же будет подсоединена к программе и станет единым целым с ней.

Физически встраиваемая СУБД является библиотекой, статически или динамически подсоединённой к основной программе. Программа и СУБД общаются не через сетевые сокеты, а через специализированное API. В этом, правда, есть и недостаток: зачастую программист сам должен разрешать многопоточность. Имеют высокую скорость и малый расход памяти,

особенно на длинных строках и BLOBах. Благодаря специализированному API количество операций чтения-записи минимально. Как правило, имеют специализированный язык запросов или неполную совместимость с SQL-92

В угоду производительности часто разработчики реализуют или неполный SQL (SQLite), или специализированный язык запросов (BerkeleyDB). К тому же встраиваемые СУБД могут действовать совсем по другому принципу, чем клиент-серверные: пинг к БД нулевой, и запрос можно выполнять по частям, несколькими обращениями к СУБД. Как правило в таких СУБД, нет пользовательских прав; простейшая изоляция транзакций Большого однопользовательской БД и не нужно. Как правило, транзакции изолируются по принципу «записываем по одному» с помощью стандартных механизмов ОС наподобие блокировки файлов. Доступ к одному файлу из нескольких программ наладить можно. Как правило, в таких СУБД нет архивации и репликации БД. Встраиваемая БД надёжна настолько, насколько надёжна библиотека СУБД и файловая система, на которой база данных располагается. Как правило, нет языковых средств, упрощающих доступ к БД

Специализированные СУБД-языки наподобие xBase обычно представляют собой симбиоз языка запросов и языка прикладного программирования. В языках программирования общего назначения правильность запроса будет проверена не при компиляции, а лишь тогда, когда запрос будет послан в СУБД. И, конечно же, на языке программирования общего назначения конструкции доступа к БД (как встраиваемой, так и клиент-серверной) будут тяжеловесными.

Таблица 3.5. Таблица тестов

Поле	Тип	Описание
_id	<i>int(11) Auto Increment</i>	Индекс
CODE	<i>text</i>	Код теста
TITLE	<i>text</i>	Название теста
SUBJECT	<i>double</i>	Предмет теста

Таблица 3.6. Таблица вопросов

Поле	Тип	Описание
_id	int(11) <i>Auto</i> <i>Increment</i>	Индекс
TEST_CODE	text	Код теста
CODE	text	Код вопроса
QUESTION	text	Текст вопроса
TYPE	text	Тип вопроса

Таблица 3.7. Таблица ответов

Поле	Тип	Описание
_id	int(11) <i>Auto</i> <i>Increment</i>	Индекс
QUESTION_CODE	text	Код вопроса
CODE	text	Код ответа
TEXT	text	Текст ответа

Таблица 3.8. Таблица результатов

Поле	Тип	Описание
_id	int(11) <i>Auto</i> <i>Increment</i>	Индекс
TITLE	text	Название теста
RIGHT	INT(11)	Правильных ответов
TOTAL	INT(11)	Всего вопросов
PERCENT	INT(11)	Процент правильных ответов
GRADE	INT(11)	Оценка

3.2 Разработка системы компьютеризированного адаптивного тестирования

3.2.1 Серверная часть

В серверной части системы компьютеризированного адаптивного тестирования были реализованы 4 контроллера, 6 моделей и свыше 20 отображений.

Контроллер Admin. Реализует функционал администрирования системы компьютеризированного адаптивного тестирования.

```
Class Admin extends Controller
{
  function Admin() {...}
  function index() {...}
  function users($action="", $id=0) {...}
  function tests($action="", $id=0) {...}
  function questions($action="", $testId=0, $id=0) {...}
  function answers($action="", $testId=0, $questionId=0, $id=0)
  {...}
  function qgroups($action="", $id=0) {...}
}
```

Контроллер имеет конструктор и 6 экшенов:

1. `index` – показывает главную страницу панели администрирования
2. `users` – позволяет создавать, редактировать, удалять пользователей, а также сбрасывать пароль пользователя, в случае необходимости
3. `tests` – позволяет создавать, редактировать и удалять тесты
4. `questions` – позволяет создавать, редактировать и удалять вопросы
5. `answers` – позволяет создавать, редактировать и удалять ответы
6. позволяет создавать, редактировать и удалять группы вопросов, необходимые для проведения адаптивного тестирования

Контроллер Auth. Реализует авторизацию пользователя в системе компьютеризированного адаптивного тестирования.

```
Class Auth extends Controller
{
```

```

function Auth() {...}
function login() {...}
function logout() {...}
}

```

Контроллер имеет конструктор и 2 экшена:

1. login – позволяет авторизоваться в серверной части
2. logout – позволяет разлогиниться из системы

Контроллер User. Реализует функционал для предоставления пользователям информации о своих результатах.

```

Class User extends Controller
{
function User() {...}
function register() {...}
function lost() {...}
function profile($id=0) {...}
function results($action="", $id=0) {...}
}

```

Контроллер имеет конструктор и 4 экшена:

1. register – позволяет зарегистрироваться новому пользователю
2. lost – позволяет восстановить утерянный пароль
3. profile – реализует профиль пользователя
4. results – отображает результаты тестирования

Контроллер Test. Реализует функционал адаптивного тестирования системы.

```

Class Test extends Controller
{
function Test() {...}
function next() {...}
function auth() {...}
function update() {...}
function results() {...}
}

```

Контроллер имеет конструктор и 4 экшена:

1. next – переходит к следующему вопросу
2. auth – авторизует клиента в системе (часть API для взаимодействия между клиентом и сервером)
3. update – посылает клиенту новые тесты

4. results – получает от клиента ответы и посылает ему результат тестирования

3.2.2 Клиентская часть

В клиентской части системы компьютеризированного адаптивного тестирования были реализованы 5 активити, 1 сервис и свыше 10 вспомогательных классов.

Пакет Auth. Содержит лишь один элемент – активити (окно), отвечающую за получение и отправку на сервер логина и пароля.

```
public class AuthActivity extends Activity {
    public void onCreate(Bundle savedInstanceState) {...}
    public void authHandler(View v) {...}
    public void logoutHandler(View v) {...}
}
```

- Метод onCreate реализует отображение полей для ввода логина и пароля
- Метод authHandler реализует отправку запроса на авторизацию к серверу
- Метод logoutHandler реализует разлогинивание на сервере

Пакет Dashboard. Содержит в себе 2 элемента, активити, отображающее меню клиентской части и лэйаут (схема размещения компонентов интерфейса) для адаптирующегося расположения пунктов меню в зависимости от ширины экрана и ориентации мобильного устройства.

```
public class DashboardActivity extends Activity {
    public static class TestsUpdateBroadcastReceiver extends
        BroadcastReceiver {
        public void onReceive(Context context, Intent intent){
            ...
        }
    }
    public void onCreate(Bundle savedInstanceState) {...}
    public void startTestActivity(View v) {...}
}
```

```

    public void startResultsActivity(View v) {...}
    public void startAuthActivity(View v) {...}
}

```

- Подкласс `TestsUpdateBroadcastReceiver` получает сообщение о приходе результатов тестирования, успешной или неуспешной авторизации
- Метод `onCreate` реализует отображение меню клиентской части
- Метод `startTestActivity` запускает активности со списком тестов
- Метод `startResultsActivity` запускает активности для просмотра результатов
- Метод `startAuthActivity` запускает активности для авторизации

Пакет Results. Содержит в себе 2 элемента, активности, отображающее результаты тестирования испытуемого и адаптер (механизм отображения определённых данных в определённом интерфейсе) для отображения каждого результата.

```

public class ResultsActivity extends Activity {
    public void onCreate(Bundle savedInstanceState) {...}
}

```

- Метод `onCreate` реализует отображение результатов тестирования

```

public class ResultAdapter extends ArrayAdapter<Result>{
    public ResultAdapter(Context context,
        int layoutResourceId,
        List<Result> data) {...}
    public View getView(int position, View convertView,
        ViewGroup parent) {...}
}

```

- Конструктор `ResultAdapter` инициализирует поля адаптера
- Метод `getView` формирует вьюшку для отображения в списке результатов

Пакет Test. Самый большой пакет клиенткой части. Содержит в себе 10 элементов, среди которых: вспомогательный класс для работы с локальной СУБД, класс реализующий интерфейс взаимодействия с сервером, активности, для проведения самого тестирования, активности для отображения списка тестов, которые можно пройти, классы содержащие необходимую информацию и методы для хранения и передачи между сервисами клиентской части тестов, вопросов, ответов, результатов тестирования. Адаптеры, для отображения списка тестов и вопроса с ответами (во время тестирования),

```
public class TestActivity extends Activity {
    public void onCreate(Bundle savedInstanceState) {...}
    public void nextQuestion(View v) {...}
    protected void sendResults(ArrayList<TestQuestion> qsts,
                               String testCode) {...}
}
```

- Метод `onCreate` реализует отображение тест
- Метод `nextQuestion` реализует переход к следующему вопросу
- Метод `sendResults` реализует отправку ответов на сервер

```
public class TestsInterfaceService extends IntentService {
    protected void onHandleIntent(Intent intent) {...}
    private StringBuilder postDataToServer(String url,
                                           String name,
                                           String data) {...}
    private SimpleXML prepareAuthXML(String login,
                                     String password) {...}
    private SimpleXML prepareResultsXML(
        ArrayList <TestQuestion> questions,
        String testCode,
        String login,
        String authKey){...}
    private void updateDatabase(String data) {...}
    private boolean checkInternetConnection() {...}
}
```

- Метод `onHandleIntent` реализует взаимодействие активности с сервисом

- Метод `prepareAuthXML` подготавливает xml-запрос на авторизацию
- Метод `prepareResultsXML` подготавливает xml-запрос на получение результатов тестирования
- Метод `updateDatabase` обновляет локальную БД, добавляет туда тесты и результаты тестирования
- Метод `checkInternetConnection` проверяет подключение к сети Интернет

3.2.3 Выводы

В данной главе были обоснованы и описаны используемые во время разработки инструментальные средства и технологии. Была обоснована и представлена архитектура системы, способы её реализации.

Так же были рассмотрены основные классы и методы как серверной, так и клиентской части системы компьютеризированного адаптивного тестирования.

IV. Безопасность жизнедеятельности

4.1. Гиподинамия (монотония) и влияние ее на здоровье человека

В соответствии с принятой физиологической классификацией трудовой деятельности в настоящее время различают следующие формы труда.

Физический труд. Физическим трудом (работой) называют выполнение человеком энергетических функций в системе "человек - орудие труда".

Физическая работа требует значительной мышечной активности. Она подразделяется на два вида: динамическую и статическую. Динамическая связана с перемещением тела человека, его рук, ног пальцев в пространстве; статическая - с воздействием нагрузки на верхние конечности, мышцы корпуса и ног при удерживании груза при выполнении работы стоя или сидя.

Физическая тяжесть работы определяется энергетическими затратами в процессе трудовой деятельности и подразделяется на следующие категории; легкие, средней тяжести и тяжелые физические работы.

Легкие физические работы (категория I) подразделяются на 2 категории: Ia, Ib. К категории Ia относятся работы, проводимые сидя и сопровождающиеся незначительным физическим усилием. К категории Ib относятся работы, проводимые сидя, стоя или связанные с ходьбой и сопровождающиеся некоторым физическим усилием.

Физические работы средней тяжести (категория II) подразделяются на две категории: IIa, IIб. К первой относятся работы, связанные с постоянной ходьбой, перемещением мелких (до 1 кг) изделий или предметов в положении стоя или сидя и требующие определенных физических усилий. К категории IIб относятся работы, связанные с ходьбой, перемещением и перенесением тяжестей массой до 10кг и сопровождающиеся умеренным физическим усилием.

Тяжелые физические работы характеризуются большим расходом энергии. К этой категории относятся работы, связанные с постоянными

перемещениями, перемещением значительных (более 10 кг) тяжестей и требующие больших физических усилий.

Механизированные формы физического труда. Они изменяют характер мышечных нагрузок и усложняют программы действий. Профессии механизированного труда нередко требуют специальных знаний и навыков.

В условиях механизированного производства наблюдается уменьшение объема мышечной деятельности, в работу вовлекаются мелкие мышцы дистальных отделов конечностей, которые должны обеспечить скорость и точность движений, необходимых при управлении механизмами. Однообразие простых и большей частью локальных действий, однообразие и малый объем воспринимаемой в труде информации приводят к монотонности труда. Монотония наиболее ярко проявляется при конвейерной форме организации трудового процесса. Этот труд требует синхронной работы участников в соответствии с заданным ритмом и темпом. При этом чем меньше времени тратит работник на операцию, тем монотоннее работа и проще ее содержание.

Монотония - выражается в преждевременной усталости и нервном истощении. В основе этого явления лежит преобладание процесса торможения в корковой деятельности, развивающиеся при действии однообразных повторных раздражителей, что снижает возбудимость анализаторов, рассеивает внимание, уменьшает скорость реакции и, как следствие, быстро наступает утомление.

Деятельность человека при механизированной форме физического труда происходит по одному из процессов:

детерминированному - по заранее известным правилам, инструкциям, алгоритмам действий, жесткому технологическому графику;

недетерминированному - когда возможны неожиданные события в выполняемом технологическом процессе, неожиданное появление сигналов, но в то же время известны управляющие действия при появлении

неожиданных событий (расписаны правила, инструкции) в выполняемом процессе.

Умственный труд (интеллектуальная деятельность). Этот труд объединяет работы, связанные с приемом и переработкой информации, требующие преимущественного напряжения внимания, сенсорного аппарата, памяти, а также активации процессов мышления, эмоциональной сферы (управление, творчество, преподавание, наука, учеба и т.д.).

операторский труд - отличается большой ответственностью и высоким нервно-эмоциональным напряжением.

Управленческий труд - определяется чрезмерным ростом объема информации, возрастанием дефицита времени ее переработки, повышения личной ответственности за принятие решений, периодическим возникновением конфликтных ситуаций.

Творческий труд - требует значительного объема памяти, напряжения внимания, нервно-эмоционального напряжения.

Труд преподавателя - постоянный контакт с людьми, повышенная ответственность, дефицит времени и информации для принятия решений, - это обуславливает высокую степень нервно-эмоционального напряжения.

Труд учащегося - требует значительного объема памяти, напряжения внимания, частым возникновением стрессовых ситуаций.

Для умственного труда характерны гипокинезия, т.е. значительное снижение двигательной активности, приводящее к ухудшению реактивности организма и повышению эмоционального напряжения. Гипокинезия является неблагоприятным производственным фактором, одной из причин сердечно-сосудистой патологии у лиц умственного труда.

При интенсивной деятельности потребность мозга в энергии повышается, составляя 15-20% от общего объема в организме. При этом потребление кислорода 100гр коры головного мозга оказывается в 5 раз больше, чем расходует скелетная мышца такого же веса при максимальной нагрузке.

При различных формах труда, происходят и различные изменения в организме работника. Любой вид трудовой деятельности представляет собой сложный комплекс физиологических процессов, в который вовлекаются все органы системы человеческого тела. Огромную роль в этой деятельности играет центральная нервная система, обеспечивающая координацию функциональных умений, развивающихся в организме при выполнении работы.

В процессе физической деятельности происходят изменения в мышечных тканях, увеличивается объем легочной вентиляции (в основном, за счет углубления дыхания), повышается артериальное давление, происходят биохимические изменения в крови, повышается потребление кислорода (расходуется на окислительные процессы в мышцах). Может наблюдаться определенные изменения водно-солевого обмена при работе в горячих цехах или при выполнении тяжелой физической работы. При этом значительное повышение деятельности потовых желез может снизить выделительную функцию почек. При тяжелой физической нагрузке возможно торможение секреции и моторной функции желудка, а также замедление переваривания и всасывания пищи. Нередко ведет к понижению корковой возбудимости, нарушению условно-рефлекторной деятельности, а также к повышению порога чувствительности зрительного, слухового и тактильного анализаторов.

В процессе умственной деятельности повышается кровяное давление, увеличивается объем легочной вентиляции (за счет учащения дыхания), повышается потребление кислорода (для работы коры головного мозга), увеличивается кровенаполнение сосудов конечностей и брюшной полости, замедляется пульс (лишь иногда значительные умственные напряжения учащают пульс за счет уменьшения диастолы). Газообмен или совсем не изменяется, или изменяется незначительно.

Умственная работа тесно связана с работой органов чувств, в первую очередь, органов зрения и слуха. Известно, что умственная работа более плодотворно протекает в тишине.

Мышечная работа при умственной деятельности человека играет большую роль. Установлено, что легкая мышечная работа стимулирует умственную деятельность, а тяжелая, изнурительная, наоборот, понижает ее, снижает качество.

Работоспособность человека и ее динамика

Условия труда влияют на работоспособность работника. Работоспособность - состояние человека, определяемое возможностью физиологических и психических функций организма, которое характеризует его способность выполнять конкретное количество работы заданного качества за требуемый интервал времени. Работоспособность - величина переменная, изменение ее во времени называют динамикой работоспособности.

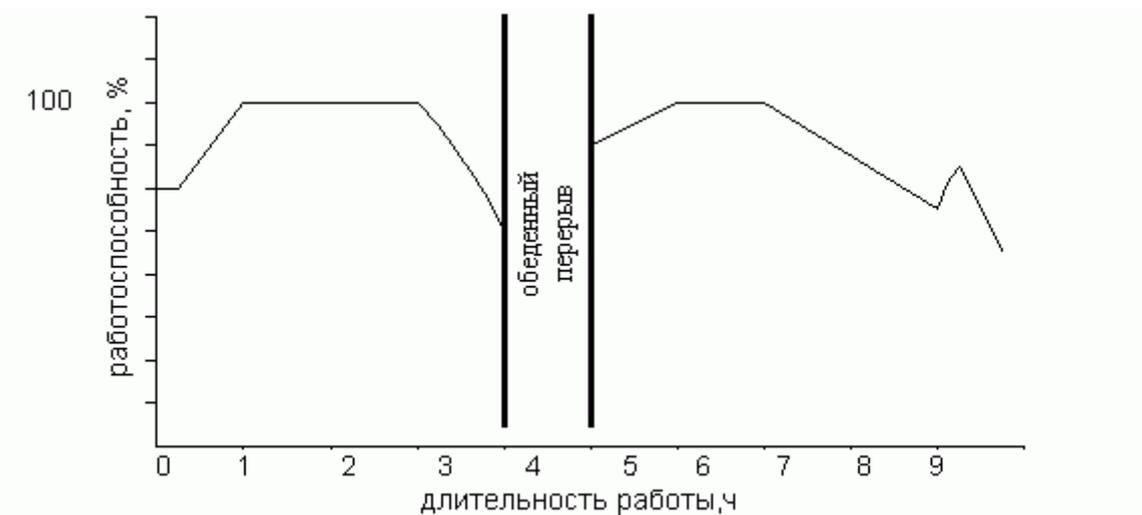


Рисунок 4.1. Фазы работоспособности человека в течение рабочего дня

Вся трудовая деятельность протекает по фазам (рис 4.1).

I .Предрабочее состояние (фаза мобилизации) - субъективно выражается в обдумывании предстоящей работы, вызывает определенные

рабочие сдвиги в нервно-мышечной системе, соответствующее характеру предстоящей нагрузки.

Вырабатываемость или стадия нарастающей работоспособности (фаза гиперкомпенсации) - период, в течении которого совершается переход от состояния покоя к рабочему, т.е. преодоление инертности покоя системы и налаживания координации между участвующими в деятельности системами организма. Длительность периода вырабатываемости может быть значительной. Например, утром после сна все характеристики сенсомоторных реакций значительно ниже, чем в дневные. Производительность труда в эти часы ниже. Период может занять от нескольких минут до двух-трех часов. На длительность сказываются: интенсивность работы, возраст, опыт, тренированность, отношение к работе.

Период устойчивой работоспособности (фаза компенсации) - устанавливается оптимальный режим работы систем организма, вырабатывается стабилизация показателей, а его длительность составляет ко всему времени работы примерно 2/3. Эффективность труда в этот период максимальная. Период устойчивой работоспособности служит важнейшим показателем выносливости человека при данном виде работы и заданном уровне интенсивности.

Выносливость обуславливается следующими факторами:

1. Интенсивностью работы. Чем больше интенсивность, тем короче период устойчивости работоспособности.

2. Спецификой работы. Например, динамическая работа может продолжаться без признаков утомления в десятки раз дольше, чем статическая. Имеет значение то, какой орган включен в действие. Для мышц ног выносливость в 1,5...2 раза больше, чем для мышц рук. Среди мышц рук выносливее сгибатели, среди мышц ног - разгибатели.

3. Возрастом. В юношеском и молодом возрасте выносливость увеличивается, в пожилом - снижается.

Полом. При нагрузке, равной половине максимальных возможностей, выносливость при статической и двигательной деятельности у мужчин и женщин одинакова. При больших нагрузках женщины выносливее.

Концентрацией внимания и волевым напряжением при интенсивной работе снижают показатели выносливости.

Эмоциональным состоянием. Положительное - уверенность, спокойствие, хорошее настроение - активизируют деятельность, удлиняя период устойчивой работоспособности. Отрицательные - страх, неуверенность, плохое настроение - оказывает угнетающее действие, снижая период устойчивой работоспособности.

Наличием умений, навыков, тренированностью - снижают волевое и эмоциональное напряжение, повышая работоспособность.

Типом высшей нервной деятельности (индивидуальные природные возможности нервной системы). Сила нервной системы характеризует работоспособность и надежность работы оператора особенно в экстремальных ситуациях.

Период утомления (фаза декомпенсации). Характеризуется снижением продуктивности, замедляя скорость реакции, появляются ошибочные и несвоевременные действия, физиологическая усталость. Утомление может быть мышечным (физическим), умственным (психическим). Утомление - временное снижение работоспособности из-за истощения энергетических ресурсов организма.

Период возрастания продуктивности за счет эмоционально-волевого напряжения.

Период прогрессивного снижения работоспособности и эмоционально-волевого напряжения.

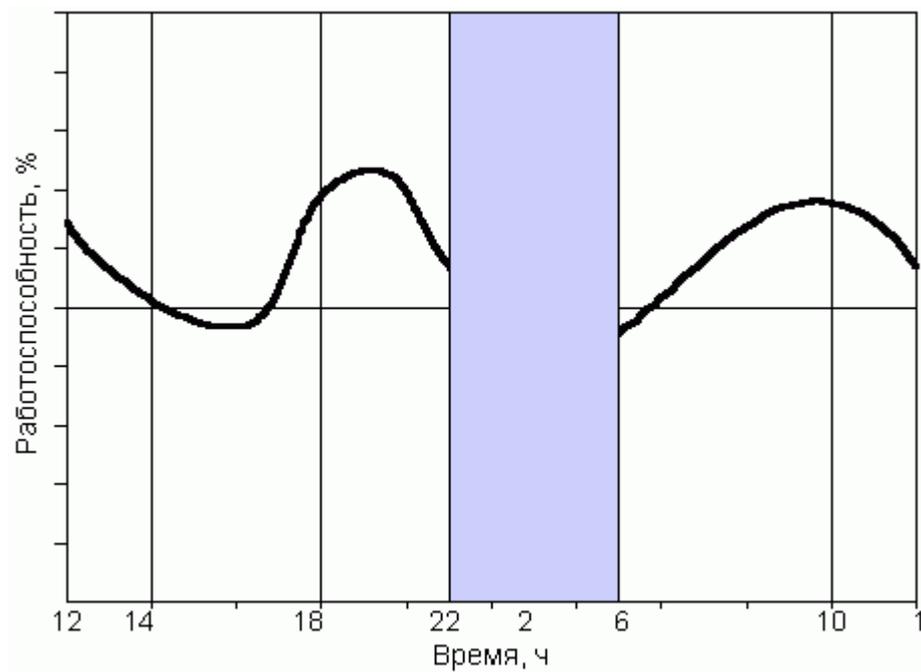


Рисунок 4.2 Колебания работоспособности в течение суток

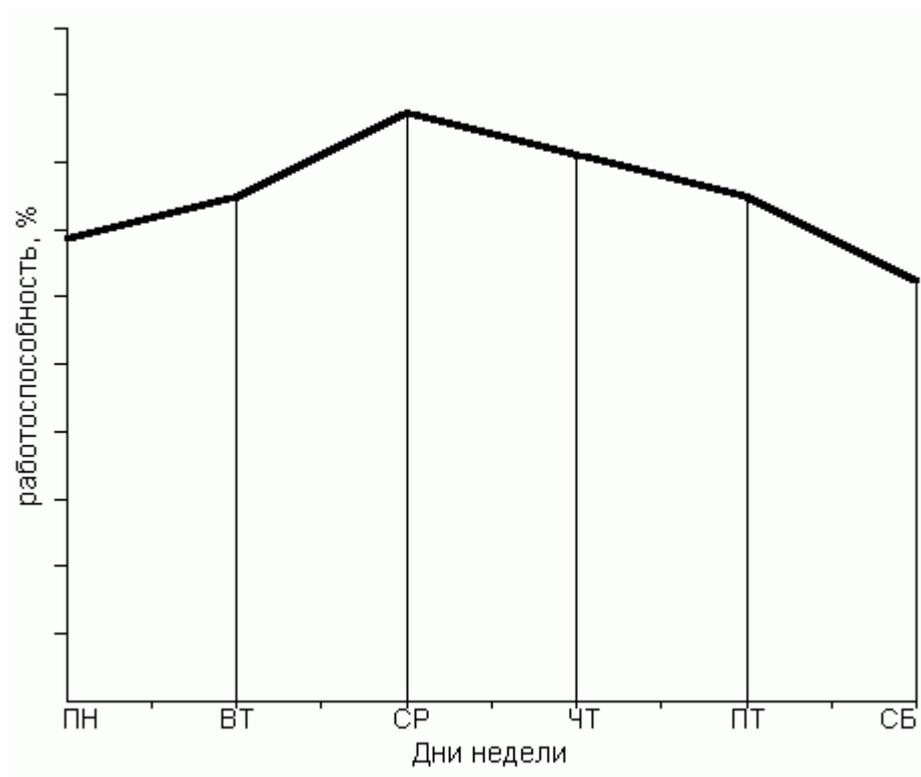


Рисунок 4.3 Колебания работоспособности в течение недели

После рабочего процесса организму необходим период восстановления работоспособности. Продолжительность этого периода определяется тяжестью проделанной работы, величиной кислородного долга, величиной

сдвигов в нервно-мышечной системе. После легкой однократной работы период может длиться 5 минут. После тяжелой однократной работы - 60...90 минут, а после длительной физической нагрузки восстановление может наступить через несколько дней.

В каждом из рассмотренных периодов работоспособности используются определенные возможности организма. Периоды I - III используют максимальные энергетические возможности организма. В дальнейшем поддержание работоспособности происходит за счет эмоционально-волевого напряжения с последующим прогрессивным снижением продуктивности труда и ослаблением контроля за безопасностью своей деятельности.

В течении суток работоспособность также меняется определенным образом. На кривой работоспособности, записанной в течении суток, выделяются три интервала, отражающие колебания работоспособности (рис.1.2). С 6 до 15 ч - первый интервал, во время которого работоспособность постепенно повышается. Она достигает своего максимума к 10-12 ч, а затем постепенно начинает понижаться. Во втором интервале (15...22 ч) работоспособность повышается, достигая максимума к 18 ч, а затем начинает уменьшаться до 22ч. Третий интервал (22..6 ч) характеризуется тем, что работоспособность существенно снижается и достигает минимума около трех часов утра, затем начинает возрастать, оставаясь при этом, однако, ниже среднего уровня.

По дням недели работоспособность также меняется (рис. 1.3). Вработывание приходится на понедельник, высокая работоспособность на вторник, среду и четверг, а развивающиеся утомление на пятницу и особенно субботу.

4.2. Пожарная безопасность.

Пожар - это горение вне специального очага, которое не контролируется и может привести к массовому поражению и гибели людей, а также к нанесению экологического, материального и другого вреда.

Горение - это химическая реакция окисления, сопровождающаяся выделением теплоты и света. Для возникновения горения требуется наличие трех факторов: горючего вещества, окислителя и источника загорания. Окислителями могут быть кислород, хлор, фтор, бром, йод, окиси азота и другие. Кроме того, необходимо чтобы горючее вещество было нагрето до определенной температуры и находилось в определенном количественном соотношении с окислителем, а источник загорания имел определенную энергию.

Наибольшая скорость горения наблюдается в чистом кислороде. При уменьшении содержания кислорода в воздухе горение прекращается. Горение при достаточной и надмерной концентрации окислителя называется полным, а при его нехватке - неполным.

Выделяют три основных вида самоускорения химической реакции при горении: тепловой, цепной и цепочно-тепловой. Тепловой механизм связан с экзотермичностью процесса окисления и возрастанием скорости химической реакции с повышением температуры. Цепное ускорение реакции связано с катализом превращений, которое осуществляют промежуточные продукты превращений. Реальные процессы горения осуществляются, как правило, по комбинированному (цепочно-тепловой) механизму.

Процесс возникновения горения подразделяется на несколько видов.

Вспышка - быстрое сгорание горючей смеси, не сопровождающееся образованием сжатых газов.

Возгорание - возникновение горения под воздействием источника зажигания.

Воспламенение - возгорание, сопровождающееся появлением пламени.

Самовозгорание - явление резкого увеличения скорости экзотермических реакций, приводящее к возникновению горения вещества при отсутствии источника зажигания.

Самовоспламенение - самовозгорание, сопровождается появлением пламени.

Взрыв - чрезвычайно быстрое (взрывчатое) превращение, сопровождающееся выделением энергии с образованием сжатых газов.

Основными показателями пожарной опасности являются температура самовоспламенения и концентрационные пределы воспламенения.

Температура самовоспламенения характеризует минимальную температуру вещества, при которой происходит резкое увеличение скорости экзотермических реакций, заканчивающееся возникновением пламенного горения.

Температура вспышки - самая низкая (в условиях специальных испытаний) температура горючего вещества, при которой над поверхностью образуются пары и газы, способные вспыхивать в воздухе от источника зажигания, но скорость их образования еще недостаточна для последующего горения.

Горючими называются вещества, способные самостоятельно гореть после изъятия источника загорания.

По степени горючести вещества делятся на: горючие (сгораемые), трудногорючие (трудносгораемые) и негорючие (несгораемые).

К трудногорючим относятся такие вещества, которые не способны распространять пламя и горят лишь в месте воздействия источника зажигания.

Негорючими являются вещества, не воспламеняющиеся даже при воздействии достаточно мощных источников зажигания (импульсов).

Горючие вещества могут быть в трех агрегатных состояниях: жидком, твердом и газообразном. Большинство горючих веществ независимо от

агрегатного состояния при нагревании образует газообразные продукты, которые при смешении с воздухом, содержащим определенное количество кислорода, образуют горючую среду. Горючая среда может образоваться при тонкодисперсном распылении твердых и жидких веществ.

Из горючих газов и пыли образуются горючие смеси при любой температуре, в то время как твердые вещества и жидкости могут образовывать горючие смеси только при определенных температурах.

В производственных условиях может иметь место образование смесей горючих газов или паров в любых количественных соотношениях. Однако взрывоопасными эти смеси могут быть только тогда, когда концентрация горючего газа или пара находится между границами воспламеняемых концентраций.

Минимальная концентрация горючих газов и паров в воздухе, при которой они способны загораться и распространять пламя, называемое *нижним концентрационным пределом воспламенения*.

Максимальная концентрация горючих газов и паров, при которой еще возможно распространение пламени, называется *верхним концентрационным пределом воспламенения*.

Указанные пределы зависят от температуры газов и паров: при увеличении температуры на 100°C величины нижних пределов воспламенения уменьшаются на 8 -10 %, верхних - увеличиваются на 12 - 15 %.

Пожарная опасность вещества тем больше, чем ниже нижний и выше верхний пределы воспламенения и чем ниже температура самовоспламенения.

Пыли горючих и некоторых не горючих веществ (например алюминий, цинк) могут в смеси с воздухом образовывать горючие концентрации.

Наибольшую опасность по взрыву представляет взвешенная в воздухе пыль. Однако и осевшая на конструкциях пыль представляет опасность не

только с точки зрения возникновения пожара, но и вторичного взрыва, вызываемого в результате взвихривания пыли при первичном взрыве.

Минимальная концентрация пыли в воздухе, при которой происходит ее загорание, называется *нижним пределом воспламенения пыли*.

Поскольку достижение очень больших концентраций пыли во взвешенном состоянии практически нереально, термин "верхний предел воспламенения" к пылям не применяется.

Воспламенение жидкости может произойти только в том случае, если над ее поверхностью имеется смесь паров с воздухом в определенном количественном соотношении, соответствующим нижнему температурному пределу воспламенения.

Меры по пожарной профилактике.

Мероприятия по пожарной профилактике разделяются на организационные, технические, режимные и эксплуатационные.

Организационные мероприятия: предусматривают правильную эксплуатацию машин и внутризаводского транспорта, правильное содержание зданий, территории, противопожарный инструктаж и тому подобное.

Технические мероприятия: соблюдение противопожарных правил и норм при проектировании зданий, при устройстве электропроводов и оборудования, отопления, вентиляции, освещения, правильное размещение оборудования.

Режимные мероприятия - запрещение курения в неустановленных местах, запрещение сварочных и других огневых работ в пожароопасных помещениях и тому подобное.

Эксплуатационные мероприятия – своевременная профилактика, осмотры, ремонты и практика тушения пожаров наибольшее распространение получили следующие принципы прекращения горения:

изоляция очага горения от воздуха или снижение концентрации кислорода путем разбавления воздуха негорючими газами (углеводы $CO_i < 12 - 14 \%$).

охлаждение очага горения ниже определенных температур;

интенсивное торможение (ингибирование) скорости химической реакции в пламени;

механический срыв пламени струей газа или воды;

создание условий огнепреграждения (условий, когда пламя распространяется через узкие каналы).

4.3 Выводы

В данной главе была рассмотрена гиподинамия, её влияние на здоровье человека, формы и условия труда.

Так же были рассмотрены вопросы пожарной безопасности, процессы возникновения горения и меры пожарной профилактики.

Заключение

При выполнении выпускной квалификационной работы, была поставлена задача: разработать систему компьютеризированного адаптивного тестирования.

В работе была изучена предметная область, сторонние разработки по данной проблеме, даны определения адаптивному тестированию и компьютеризированному адаптивному тестированию, были рассмотрены требования предъявляемые к системам подобного рода.

Кроме того были рассмотрены и описаны основные модели и алгоритмы применяемые в компьютеризированном адаптивном тестировании. В соответствии с этим были выбраны математическая модель системы и методы решения поставленной задачи.

В ходе выполнения работы были проектированы и реализованы две взаимосвязанные части одной системы, а также спроектирована и разработана база данных системы.

Предложенная программная реализация модели адаптивного тестирования может быть использована при контроле знаний, как в средних, так и высших учебных заведениях.

Список использованных источников и литературы

1. Каримов И.А. Указ Президента «О дальнейшем развитии компьютеризации и информационно-коммуникационных технологий» от 30 мая 2002 г.
2. Каримов И.А. Указ Президента «Концепции информатизации сферы образования Республики Узбекистан» от 11 февраля 2004 г.
3. Аванесов В.С. Адаптивное обучение и адаптивный тестовый контроль. // <http://testolog.narod.ru/Theory41.html>
4. Белоус И.В., Пархоменко С.А. Методы математической статистики для анализа тестовых результатов // Вестник ХГТУ №1 (14), 2002. С. 544 - 545
5. Беспалько В. П. Основы теории педагогических систем. Воронеж: Издательство Воронежского университета, 1977.
6. Дуплик С. В. Модель адаптивного тестирования на нечеткой математике. // «Информатика и образование» №11/2004.
7. Евсеев В.В., Алехина С.В., Евсеева И.В. Выбор релевантного алгоритма оценивания знаний обучаемых в системе дистанционного обучения. // Сб. научных трудов конференции ВИРТ-2003, 2003.- с.311-315.
8. Евсеев В.В., Безуглая А.Е., Алехина С.В. Методы формирования оценки знаний в системах дистанционного обучения // Сб. научных трудов 6-й Международной конференции Украинской ассоциации дистанционного обучения. Харьков - Ялта: 2002. - С. 372-376
9. Зайцева Л.В., Новицкий Л.П., Грибкова В.А. Разработка и применение автоматизированных обучающих систем на базе ЭВМ. — Рига. Зинатне, 1989. -174 с.
10. Информационные технологии в науке и образовании». Шахты, 2001. С. 29-31.
11. Кирий В.Г., Ульянов Д.А. Корреляционный анализ ответов на вопросы при адаптивном тестировании // Вестник ИрГТУ. Иркутск. -2003. - № 3-4 (15-16). - С. 125-128
12. Кривуля Г., Пиженко И., Шкиль А. Проверка знаний при дистанционном обучении // Образование и виртуальность -2001. Харьков -: УАДО, 2001. - С. 212-219.
13. Мельникова М.Б. Теория и практика конструирования педагогических тестов. М.: Логос, 2002. - 410 с.

14. Печенежский Н.А., Маслов А.А., Модель системы компьютеризированного адаптивного тестирования. // Труды конференции ВИРТ-202
15. Пугачев А. А. Адаптивные компьютерные обучающие системы // Материалы международной научно-практической конференции
16. Шойтов Д.В. Проектирование системы динамического трехуровневого тестирования, основанной на теории графов. // [http:// scientific-notes.ru/pdf/010-02.pdf](http://scientific-notes.ru/pdf/010-02.pdf)
17. Baker, F.B. The Basics of Item Response Theory. ERIC Clearinghouse on Assessment and Evaluation, 2001.
18. Birnbaum, A. Some latent trait models and their use in inferring an examinee's ability. Part 5 in F.M. Lord and M.R. Novick. Statistical Theories of Mental Test Scores. Reading, MA: Addison-Wesley, 1968.
19. Hambleton, R.K., and Swaminathan, H. Item Response Theory: Principles and Applications. Hingham, MA: Kluwer, Nijhoff, 1984.
20. Lord, F.M. Application of Item Response Theory to Practical Testing Problems. Hillsdale, NJ: Erlbaum, 1980.
21. Rudner, L.M. Item Response Theory, <http://edres.org/irt/> Извлечено 03.08.04
22. Wikipedia. Android // <http://ru.wikipedia.org/wiki/Android>
23. Wikipedia. Model-View-Controller // <http://ru.wikipedia.org/wiki/MVC>
24. Wright, B.D., and Mead, R.J. BICAL: Calibrating Items with the Rasch Model. Research Memorandum No. 23. Statistical Laboratory, Department of Education, University of Chicago, 1976.
25. Wright, B.D., and Stone, M.A. Best Test Design. Chicago: MESA Press, 1979.