

ГОСУДАРСТВЕННЫЙ КОМИТЕТ СВЯЗИ, ИНФОРМАТИЗАЦИИ И
ТЕЛЕКОММУНИКАЦИОННЫХ ТЕХНОЛОГИЙ РЕСПУБЛИКИ УЗБЕКИСТАН

ТАШКЕНТСКИЙ УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ
ТЕХНОЛОГИЙ

К защите

зав. Кафедрой

« ____ » ____ 2014г.

ВЫПУСКНАЯ

КВАЛИФИКАЦИОННАЯ РАБОТА БАКАЛАВРА

На тему:

Разработка информационных ресурсов для социальной студенческой сети

ТУИТ

Выпускник

Норматов М.

Руководитель

Рафиков Т.

Рецензент

Каримова Д. К.

Консультант по БЖД

Абдуллаева С.М.

ТАШКЕНТ – 2014 г.

ТАШКЕНТСКИЙ УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ

Факультет: КИ Кафедра: Информационные технологии

Направление (специальность): Информатика и информационные технологии

УТВЕРЖДАЮ

Зав _____ кафедрой

« _____ » _____ 2014

г.

ЗАДАНИЕ

на выпускную квалификационную работу

Норматов Муроджон Иброхимжон ўғли

(фамилия, имя, отчество)

1. Тема работы: Разработка информационных ресурсов для социальной студенческой сети ТУИТ.
2. Утверждена приказом по университету от « 19 » 04 2014 г. № 254-15
3. Срок сдачи законченной работы ____.
4. Исходные данные к работе Литературные данные, _____ входящие данные
5. Содержание расчётно – пояснительной записки (перечень подлежащих разработке вопросов) Аннотация. Введение. Глава 1. Описание предметной области Глава 2. Проектирование БД. Глава 3 . Описание БД. Глава 4 . Безопасность жизнедеятельности, Заключение, Использованная литература, Приложение
6. Перечень графического материала Презентационные материалы
7. Дата выдачи задания _____

Руководитель _____ Задание принял _____
 (подпись) (подпись)

8. Консультанты по отдельным разделам выпускной работы

Раздел	Ф.И.О руководителя	Подпись дата	
		Задание выдал	Задание получил
Основная часть	Рафиков Т.		
БЖД	Абдуллаева С.М.		

9. График выполнения работы

№	Наименование раздела работы	Срок выполнения	Отметка руководителя о выполнении
1	Введение	3.03.2014-7.03.2014	
2	Проектирование БД	8.03.2014-21.03.2014	
3	Структура БД	22.03.2014-7.04.2014	
4	Создание таблицы	8.04.2014-11.04.2014	
5	Создание тестовых данных	12.04.2014-25.04.2014	
6	Тестирование БД	26.04.2014-15.05.2014	
7	Безопасность жизнедеятельности	16.05.2014-19.05.2014	
8	Заключение	20.05.2014-21.05.2014	

Выпускник _____ « _____ » _____ 200__ г.
 (подпись)

Руководитель _____ « _____ » _____ 200__ г.
 (подпись)

В выпускной квалификационной работе был разработан информационный ресурс для социальной сети, который включает базу данных студентов, новостей, предметов, статей и комментариев. В качестве инструментария использовались такие программные средства: как ОС семейства Linux Debian 7, СУБД MySQL 5, Web сервер Apache 2, язык интерпретатор PHP 5.3

Ушбу битирувмалакавийишидаталабалармаълумотларибазаси, янгиликлар, фанлар, маколаларвашархларни уз ичига олганижтимоийтармоқучунахборот ресурси шлабчикилди.

БМинитайёрлашдакуйидагидастурийтаъминотларданфойдаланилди: ОТ Linux Debian 7 ОТ оиласи, СУБД MySQL 5, Apache 2 Webсерверива PHP 5.3 интерпретатор тили.

In this qualification work was developed information resource for the social network, which involves student's database, news, subjects, articles and comments. As a tool used different types of software such as OS Linux Debian 7, MySQL 5 SCDB, Web server Apache 2, PHP 5.3.

СОДЕРЖАНИЕ

Введение	4
Глава 1. Описание предметной области	8
1.1. Социальные сети	8
1.2. Требования к аппаратному обеспечению	14
1.3. Постановка задачи	15
1.4. Выводы по первой главе	16
Глава 2. Проектирование БД	17
2.1. Этапы разработки БД	17
2.2. Подготовка БД	22
2.3. Выводы к второй главе	27
Глава 3. Описание БД	28
3.1. Структура БД	28
3.2. Создание таблицы	41
3.3. Создание тестовых данных	45
3.4. Тестирование БД	51
3.5. Увеличение производительности сервера базы данных	51
3.6. Выводы к третьей главе	57
Глава 4. Безопасность жизнедеятельности	58
4.1. Гиподинамия (монотония) и ее влияние на организм человека	58
4.2. Эргономика	65
4.3. Выводы к четвертой главе	76
Заключение	78
Список литературы	79
Приложение	80

ВВЕДЕНИЕ

Кардинальные преобразования образовательно-воспитательной системы, широкомасштабные реформы проводимые в обществе, позволяют говорить о том, что Узбекистан идет к совершенно новой модели государства, где на первый план выходят формирующие начала нестандартной и глубоко мыслящей личности, способной обновить и преобразовать общество. С первых шагов по пути независимости Узбекистан придает большое значение возрождению и дальнейшему развитию нашей духовности, совершенствованию системы национального образования, укреплению его национальной основы, повышению их до уровня мировых стандартов в гармонии с требованиями времени.

Внедрение новейших информационных технологий в образовательный процесс является важным условием интеллектуального развития студентов и учащихся и социально-экономического развития общества. Наша страна в годы независимости достигла в данном направлении больших успехов. Политика нашего государства в области развития информатизации, основополагающие принципы которой определены Президентом Исламом Каримовым, непосредственно направлена на интеграцию нашей страны в мировое информационное пространство. Мощный импульс этому процессу придало постановление главы нашего государства «О мерах по дальнейшему внедрению и развитию современных информационно-коммуникационных технологий» от 21 марта 2012 года.

Internet развивается довольно стремительно. Быстро растет количество изданий, посвященных Сети, что предвещает широкое ее распространение даже в далеких от техники областях. Internet превращается из большой игрушки для интеллектуалов в полноценный источник разнообразной полезной информации для любой категории пользователей.

Сегодня, когда практически во всех ВУЗах добавляются новые специальности, проводятся курсы переподготовки, часы самостоятельной работы возникает потребность оптимизации учебного процесса, его улучшении, а точнее даже о том, что подход к образованию устарел. Таким образом возникает вопрос «как с помощью WWW нести *желание учиться* в массы?»

Социальная сеть — это виртуальное сообщество людей, объединенных общими интересами или родом деятельности, которые общаются между собой при помощи различных систем обмена сообщениями (мгновенных или электронной почты). Кроме обмена сообщениями в социальных сетях, как правило, имеется возможность совместного просмотра различных объектов, обмена фотографиями, видео и текстовыми файлами. Социальные сети — одна из отличительных черт web 2.0. Свое распространение они получили не так давно, но уже плотно вошли в нашу жизнь.

Социальные сети с каждым днем набирают популярность во всем мире. Они возникли около 10 лет назад, но уже сегодня трудно найти человека, который бы пользуясь интернетом, не пользовался бы социальными сетями. Именно в этом, мы считаем, и заключается актуальность выбранной темы. Они являются неотъемлемой частью Web 2.0 (методика проектирования систем, которые путём учёта сетевых взаимодействий становятся тем лучше, чем больше людей ими пользуются), и их дальнейшее развитие в значительной степени определяет и перспективы развития самого Интернета. На конец 2009 года в социальных сетях было зарегистрировано 917 млн. пользователей или 13,3 % населения мира. Ожидается, что к 2015 году количество зарегистрированных в социальных сетях пользователей увеличится до 2 млрд. (27 % населения мира).

Ежедневно миллионы людей общаются, договариваются о встречах, ищут информацию именно в социальных сетях и самыми активными пользователями являются молодежь. В современном мире молодежь стала

удовлетворять свою потребность в общении путем использования этих социальных сетей.

Целью данной работы является разработка информационных ресурсов для студенческой социальной сети, включающую следующие подсистемы:

1. система регистрации/авторизации пользователей;
2. система восстановления паролей;
3. система присутствия на сайте – пользователи находящиеся на сайте, отмечаются статусом «online»;
4. система пользовательских профилей;
5. система друзей – возможность установки связи между зарегистрированными пользователями;
6. система личных сообщений – внутренняя почта для зарегистрированных пользователей с уведомлением на email;
7. ленты активности — позволяет пользователям, находящимся в связи отслеживать события друг друга;
8. форум – позволяет пользователям создавать обсуждения по интересам;
9. система фотоальбомов – кроме своих фотографий, пользователи обладают возможностью создания альбома для отслеживания своих результатов и успехов;
10. система личных статусов - пользователи проекта могут устанавливать свой личный статус (послание) для своего профиля (система микроблогов). При использовании в проекте ленты активности, личные статусы пользователей попадают в неё;
11. система комментирования;
12. будет хранить в себе лекционные материалы, практические работы, лабораторные работы и другие материалы;
13. система обмена – пользователи могут делиться фотографиями, документами и т.п.;
14. возможность объединят студентов по интересам

В выпускной квалификационной работе использовались следующие инструментальные средства:

СУБД -MySQL;

Операционная система -Debian 7;

Web-сервере -Apache2;

Интерпретатор языка PHP;

Для работы с СУБД –HeidiSQL

Выпускная квалификационная работа состоит из введения, четырех глав, заключения и списка использованной литературы и приложение.

В введении - обоснована актуальность работы, поставлена цель и задачи выпускной квалификационной работы.

В главе 1 «Описание предметной области» – проведен полный обзор популярных социальных сетей а также исходя из этого обзора составления требования к аппаратному обеспечению, поставлена задача к нашей системе

В главе 2 «Проектирование БД»– определены этапы разработки БД а также подготовлена БД для дальнейшей работы.

В главе 3 «Описание БД »– в этом разделе создана структура БД с полным её функционалом, созданы таблицы, была рассмотрена сама таблица с введенными тестовыми данными, после введения данных необходимых для работы системы мы протестировали базу данных на работоспособность. Последующим нашим шагом было увечить производительность сервера для максимального функционирования нашей базы данных.

В глава 4. «Техника безопасности»– проводится ознакомление с мерами безопасности при работе с персональным компьютерам.

В заключении приведены основные практические и теоретические выводы к работе. Система была протестирована на просторах социальной сети, разработана максимально удобная и доступная для обращения база данных.

В приложении приведены – запросы БД, программный код и интерфейс основных компонентов.

ГЛАВА 1. ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ

1.1. Обзор социальных сетей

В конце 1970-х годов начали появляться первые электронные доски объявлений, на которых пользователи могли общаться между собой. Конечно, в то время это было не так удобно, быстро и комфортно как можно делать сегодня, но это были первые попытки объединить людей во всемирной паутине.

Следующим этапом было появление первых on-line сервисов общения. Первой компанией, которая создала чат для общения, была CompuServe. Недостатком данной компании было достаточно дорогое использование, ведь за час общения на сервисе приходилось платить до 30 долларов. Далее было появление систем мгновенных сообщений. Первым сервисом по передачи файлов и быстрого общения в чате был IRC (Internet Relay Chat), созданный в 1988 года в качестве замены программы MultiUser Talk. Популярным IRC стал с 1991 года после военной операции «Буря в пустыне», когда в режиме «on-line» транслировались сообщения со всего мира. Недостатком в IRC было ограниченное пользование, так как сервис был UNIX системой.

В середине 1990-х годов был создан ICQ и был первой системой мгновенных сообщений для персональных компьютеров. ICQ была создана компанией Mirabilis. На сегодняшний момент ICQ входит в состав Mail.Ru Group и насчитывает 23.4 млн. пользователей по всему миру на март 2012 года.

Главной причиной, по которой социальные сети имели возможность появиться, это же конечно желание людей общаться между собой. Это, наверное, единственная потребность, которая была необходима каждому

человеку до появления самого интернета. Только раньше все было совсем иначе.

Чтобы увидеться и пообщаться люди встречались и общались. Когда парень хотел познакомиться с девушкой, он знакомился на улице, приглашал гулять, водил ее в кино, театр или просто в парк. А как знакомится в большей степени молодежь сегодня? Это добавляют незнакомых людей в социальных сетях и начинают общаться.

Раньше не было ни телефонов, ни социальных сетей. Тяжело сказать, когда было лучше, когда люди общались без интернета или с ним. Как говорит поколение наших родителей: «У нас не было телефонов, но мы всегда знали, где наши друзья».

С появлением и распространением сети интернет все начало меняться. Общение с друзьями потихоньку начало переходить в виртуальность. Хорошо это или плохо, каждый сам решает для себя, но то, что это уже свершилось отрицать глупо. В последнее время все больше людей начинают общаться именно в социальных сетях, иногда даже отказываясь от личного общения. Что же дало толчок к появлению и развитию социальных сетей? Конечно же, быстрорастущая сеть интернет, которая и стала причиной появления социальных сетей.

А сейчас я вкратце опишу историю того, как и когда начали появляться первые социальные сети.

Первым аналогом социальной сети в интернете был Classmates запущенная в 1995 году Рэнди Конрадсом. Переводиться сайт как «Одноклассники». Цель сайта была предоставить возможность найти своих одноклассников и школьных друзей. Сайт не стал изначально сильно популярным по той причине, что изначально не было функций создания профилей и добавления в друзья. А пользователи сайта могли только видеть список учебных заведений, и кто учился в них. Именно Classmates дал толчок к появлению социальных сетей. Проект оказался успешных и начал приносить прибыль своим владельцам, что в свою очередь спровоцировало

других людей создавать подобные сервисы и сайты. На данный момент сайт до сих пор существует и насчитывает уже более 50 млн. пользователей по всему миру.

Полноценной в нашем понимании первой социальной сетью можно назвать SixDegrees, запущенный в 1997 году Эндрю Вейнрейхом. Целью социальной сети была попытка доказать «теорию шести рукопожатий» («Six degrees separation» в английской терминологии). Данная теория является одной из основных в изучении социальных связей, ее суть заключается в том, что любых 2 человека на Земле друг с другом связаны цепочкой знакомств максимум из 6 человек. В SixDegrees предлагались уже такие функции создание своей собственной странички, список друзей и поиск друзей по социальной сети. Проект начал быстро набирать популярность. В 2000-м году был продан за 125 млн. долларов. Однако в 2001-м году данная социальная сеть перестала существовать.

Следующими социальными сетями были AsianAvenue, MiGente, BlackPlanet они появились в период между 1997 и 1999 годами. По функциональности они сильно не отличались от своих предшественников.

18 марта 1999 году американским студентом-программистом Брэдом Фицпатриком был создан «живой журнал» Livejournal. Данный сервис впоследствии стал массовым хостингом блогов и имеет очень большую популярность в странах СНГ. Livejournal предоставил впервые возможность создавать группы и общаться в них. Сегодня мы видим это во всех основных социальных сетях.

Первым web-ресурсом ориентированным на поиск и поддержание деловых контактов стал Ryze . Он был создан в 2001 году. Именно этот проект стал толчком к созданию более известных проектов, таких как Friendster и LinkedIn.

В 2002 году Джонатаном Абрамсомом был разработан сайт знакомств Friendster. Уникальность данной социальной сети была в том, что она не пошла по пути предшественников не пытаясь подружить двух незнакомых

людей. Основная идея сайта была в том, что бы помочь людям находить новых друзей и знакомых в списках своих друзей. Данное нововведение позволило сайту собрать 3 млн. пользователей за первые 3 месяца существования. На данный момент на ресурсе зарегистрировано более 115 млн. человек по всему миру.

В том же году, в декабре, была создана социальная сеть LinkedIn. Запущена социальная сеть была в мае 2003 года Ридом Хоффманом. Целью данной социальной сети был поиск и установление деловых контактов. На данный момент в LinkedIn зарегистрировано более 200 млн. пользователей, представляющих более 150 отраслей бизнеса более чем в 200 стран.

В 2003 году была создана новая социальная сеть MySpace. В ней предоставлена возможность создания персональных профилей, самим настраивать внешний вид своих профилей, создавать сообщества по интересам, размещать фотографии, видео и ауди известных исполнителей. При регистрации каждый получал собственный блог. Эти преимущества и позволили MySpace в 2006 году стать самой популярной социальной сетью во всем мире. Основными пользователями MySpace стали рок-коллективы. Для многих музыкантов данный сайт стал настоящей площадкой для само презентации, а так же возможность для общения со своими кумирами, иногда даже быть в числе их друзей. Но на данный момент рейтинг MySpace с каждым днем падает, уступая позиции более крупным и популярным социальным сетям. По рейтингу популярности сайтов в 2009 году MySpace был на 5 месте, а на начало 2013 года уже опустился до 229 места.

В 2004 года компанией Ludicorp был запущен проект Flickr. Создатели этой идеи Стюарт Баттерфилд и Катерина Фейк. Целью проекта была возможность игрокам Game Neverending сохранять в ходе игры свои изображения, что бы в дальнейшем выкладывать их в интернете. Сразу сервис начал пользоваться большой популярностью, и уже в 2005 года Flickr был куплен Yahoo став доступным для любого пользователя сети интернет.

В 2004 году Марком Цукербергом была основана социальная сеть TheFacebook. В 2005 году сеть стала называться Facebook. Изначальная цель социальной сети была предоставить возможность студентам Гарвардского университета общаться между собой, размещать фотографии, видеозаписи, изменять статусы в режиме on-line. В 2008 году социальная сеть Facebook стала самой популярной социальной сетью во всем мире, и с этого периода свое лидерство не упустила ни разу. На протяжении всего периода существования Facebook постоянно добавляет новые функции, включая мгновенные сообщения, чат, множества приложений. Так же есть собственная платформа для разработчиков. Facebook сегодня является 2 по популярности сайтом в мире, уступая 1 позицию поисковой системе Google.

В 2006 году Джек Дорси запустил проект Twitter и сразу стал быстро растущим новым социально-сетевым проектом. Уникальность Twitter состоит в возможности пользователей отправлять короткие текстовые сообщения (до 140 символов). Изначально цель проекта была предоставить возможность человеку ответить на простые вопросы «Что ты сейчас делаешь?» «Чем ты сейчас занимаешься?» и т.д. Ответ при этом был доступен сразу всем людям, которые подписаны на данного пользователя. Данное нововведение позволило быстро завоевывать рынок и расширить функциональность. Так с 2010 года на сайте можно смотреть фото, видео, а функционирование сайта стало быстрее и проще. На данный момент Twitter входит в 10-ку самых посещаемых сайтов мира, занимая почетное 8 место на начало 2013 года.

На территории стран СНГ социальные сети начали появляться с 2006 года. Первыми крупными социальными сетями были «Одноклассники» и «ВКонтакте». «Одноклассники» аналог проекта Classmates. «ВКонтакте» можно назвать точной копией FaceBook, хотя ее создатели это отрицают.

Проект «Одноклассники» был запущен 4 марта 2006 года Альбертом Попковым. Главной задачей социальной сети было поиск одноклассников, однокурсников, бывших выпускников, друзей, знакомых, а так же

родственников, с которыми была потеряна связь. В 2012 году на сайте было зарегистрировано уже более 150 млн. пользователей и это число продолжает увеличиваться с каждым днем.

Самая известная и популярная социальная сеть в СНГ «ВКонтакте» была запущена 10 октября 2006 года и позиционировала себя как социальная сеть для выпускников ВУЗов. На сегодняшний день ежедневная посещаемость сайта более 42 млн. пользователей и является самым популярным и самым посещаемым ресурсом на рынке СНГ. В 2012 году «ВКонтакте» перешел на новый домен «VK.com». На сегодняшний день «ВКонтакте» самый большой видео- и аудио-хостинг в Рунете.

Id.uz - Социальная сеть. Id.uz – это современная полноценная социальная сеть, со всеми необходимыми возможностями для комфортного поиска и общения: выбор друзей, обмен сообщениями, ведение и чтение микроблогов, установка статуса и чтения статусов друзей, приглашение пользователей для просмотра своих фотоальбомов, отслеживание своей активности и отслеживание активности пользователей на проектах Ziyonet и Id.uz, создание групп по интересам и участие в уже созданных группах. Id.uz – удобен в пользовании за счет того, что многие социальные сервисы вынесены в отдельные проекты: блоги в fikr.uz, видео в utube.uz, виджеты с информерами и RSS-лентами на desk.uz.

Vsetut.uz - Социальная сеть Узбекистана. Сайт для общения, обмена информацией, клубы, дневники, флэш игры, тесты, видео и другие развлекательные анонсы. В этом сайте можно зарегистрироваться и спокойно использовать ресурсы и возможности сайта. Галерея со множеством фотографий. Создав страничку сможем контролировать собственный аккаунт, просматривать действия друзей, публиковать услуги, ваш гороскоп и приложения.

Sinfdosh.uz - социальная сеть. Сайт «Sinfdosh.uz» создан для поиска старых знакомых. Здесь можно найти человека, которого не видели десятки лет, завязать новые знакомства, основанные на общих воспоминаниях,

организовать виртуальные и реальные встречи. Благодаря этому сайту сотни людей нашли своих одноклассников, познакомились, восстановили, казалось бы, навсегда утерянную дружбу.

Muloqot.uz .Пользовательский интерфейс Muloqot.uz разработан на двух языках – узбекском и русском – что дает возможность выбрать тот язык, на котором пользователю удобнее общаться. Возможность использования SMS-интерфейса, реализованного в мобильной версии Muloqot.uz. Это очень удобно в условиях, когда доступ к Интернет не всегда возможен с компьютера. А с мобильного телефона всегда можно продолжить общение в нашей сети через SMS-сообщения. Если доступ в Интернет осуществляется по технологии ADSL, то используемый трафик в обращениях по адресу Muloqot.uz будет бесплатным, потому что наш сайт размещен на хостинг-площадке участников Tas-IX.

В последнее время очень много создается новых социальных сетей, при этом не всегда они создаются для общего пользования. Есть узконаправленные социальные сети, цель которых собрать людей с общими интересами. Количество людей, конечно, в таких социальных сетях на порядок ниже, чем те, которые акцент делают на охват всех пользователей, но в таких социальных сетях есть своим преимущества.

1.2. Требования к аппаратному обеспечению.

Система предъявляет следующие требования к аппаратному и программному обеспечению сервера:

Минимальные требования к серверу: 1 компьютер с процессором Pentium IV 1 ГГц и выше, 1 Гб оперативной памяти.

Рекомендуемые требования к серверу: 2 компьютера с процессором Pentium(R) DualCore 2ГГц и выше, 2Гб оперативной памяти. В этом случае предполагается, что один компьютер будет выполнять функции SQL сервера, а другой – веб-сервера Apache 2, PHP интерпретатор.

Существует также возможность инсталляции системы на платформе Windows. При этом требования к предустановленному программному обеспечению остаются прежними.

Для работы с клиентской частью системы требуется компьютер, подключенный по протоколу TCP/IP к сети, в которой находится любой браузер.

1.3. Постановка задачи

Разрабатываемый информационный ресурс для информационной системы, должен реализовать следующие задачи:

1. разработать систему регистрации/авторизации пользователей;
2. разработать систему восстановления паролей;
3. разработать систему присутствия на сайте – пользователи находящиеся на сайте, отмечаются статусом «online»;
4. разработать систему пользовательских профилей;
5. разработать систему друзей – возможность установки связи между зарегистрированными пользователями;
6. создать систему личных сообщений – внутренняя почта для зарегистрированных пользователей с уведомлением на email;
7. включить ленты активности — позволяет пользователям, находящимся в связи отслеживать события друг друга;
8. создать форум, который позволяет пользователям создавать обсуждения по интересам;
9. разработать систему фотоальбомов – кроме своих фотографий, пользователи обладают возможностью создания альбома для отслеживания своих результатов и успехов;
10. разработать систему личных статусов - пользователи проекта могут устанавливать свой личный статус (послание) для своего профиля (система микро-блогов). При использовании в проекте ленты активности, личные статусы пользователей попадают в неё;

11. создать систему комментирования;
12. разработать модуль, который будет хранить в себе лекционные материалы, практические работы, лабораторные работы и другие материалы;
13. построить систему обмена – пользователи могут делиться фотографиями, документами и т.п.;
14. дать возможность объединять студентов по интересам

Создаваемая база данных должна отвечать условиям целостности и непротиворечивости данных. При проектировании необходимо с одной стороны учитывать правила нормализации, а с другой стороны насколько оправдано применение данных ограничений с учетом эффективности работы запросов при этой структуре данных.

Выводы по первой главе:

- Приведен краткий обзор истории популярных социальных сетей.
- Рассмотрены задачи решаемые ИР.

ГЛАВА 2. ПРОЕКТИРОВАНИЕ БД

2.1. Этапы разработки БД

При разработке БД можно выделить следующие этапы работы:

1. Проектирование таблицы для работы с пользователями;
 2. Проектирование таблицы для блога;
 3. Проектирование таблицы для контента;
 4. Проектирование таблицы для функционирования форума
 5. Проектирование системных таблиц;
1. Проектирование таблицы для работы с пользователями. В этом этапе проектируется таблицы для работы с пользователями: учет пользователей, пользовательские группы, фотографии, файлы а также, таблицы для функционирования переписки сообщений между пользователями.

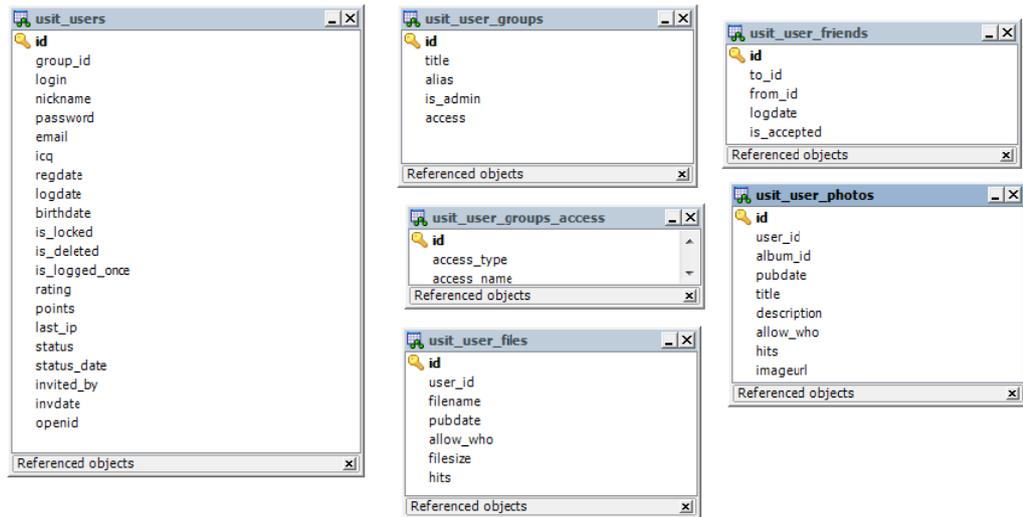
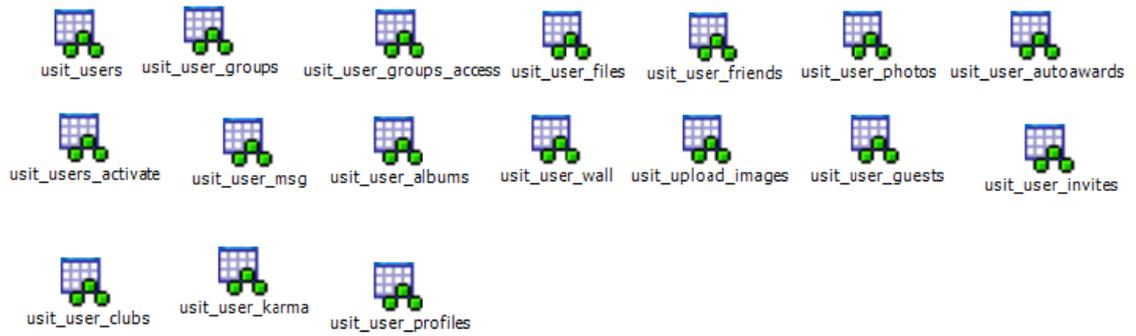


Рис.2.1. Таблицы для систематизации пользователей

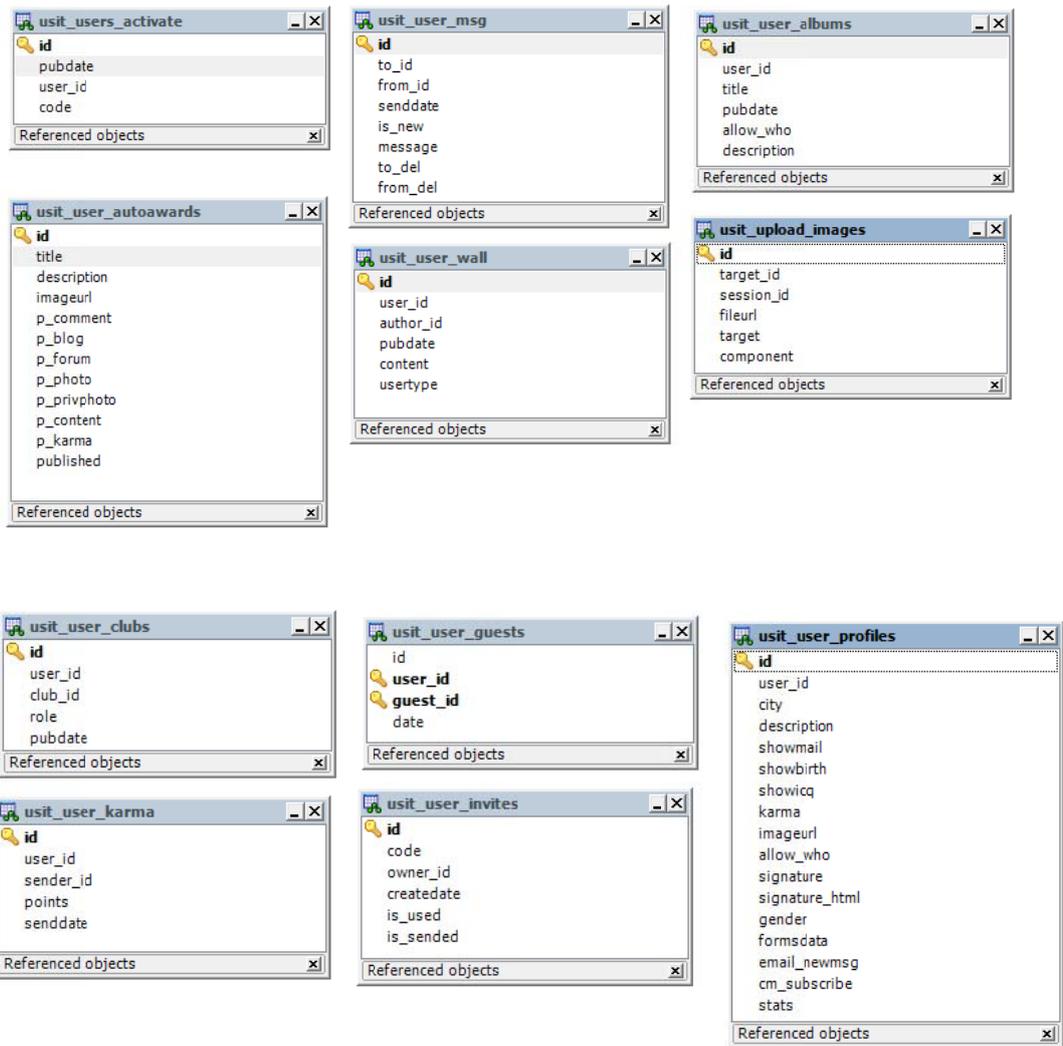


Рис.2.2. Таблицы для систематизации пользователей(продолжение)

2. Проектирование таблицы для блога. В этом этапе проектируются таблицы для функционирования пользовательского блога. А именно таблицы блога, посты, авторы и категории.



Рис.2.3. Таблицы для систематизации блога

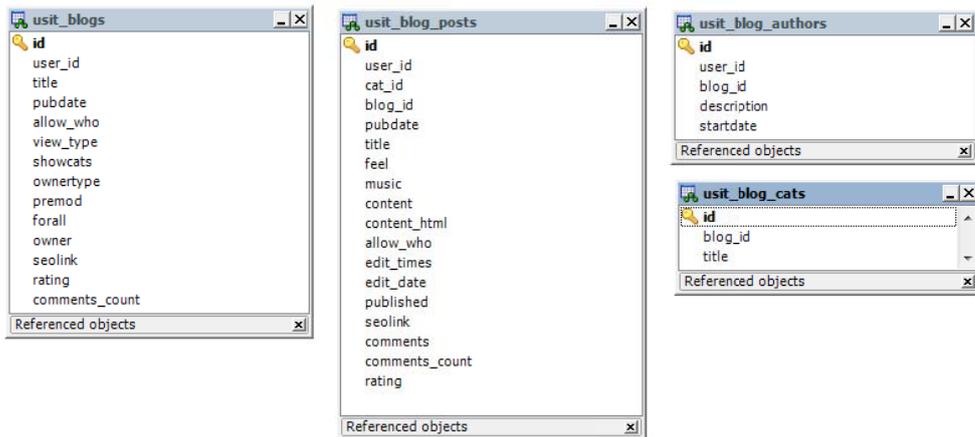


Рис.2.4.Таблицы для систематизации блога(полный вид)

3. Проектирование таблицы для контента

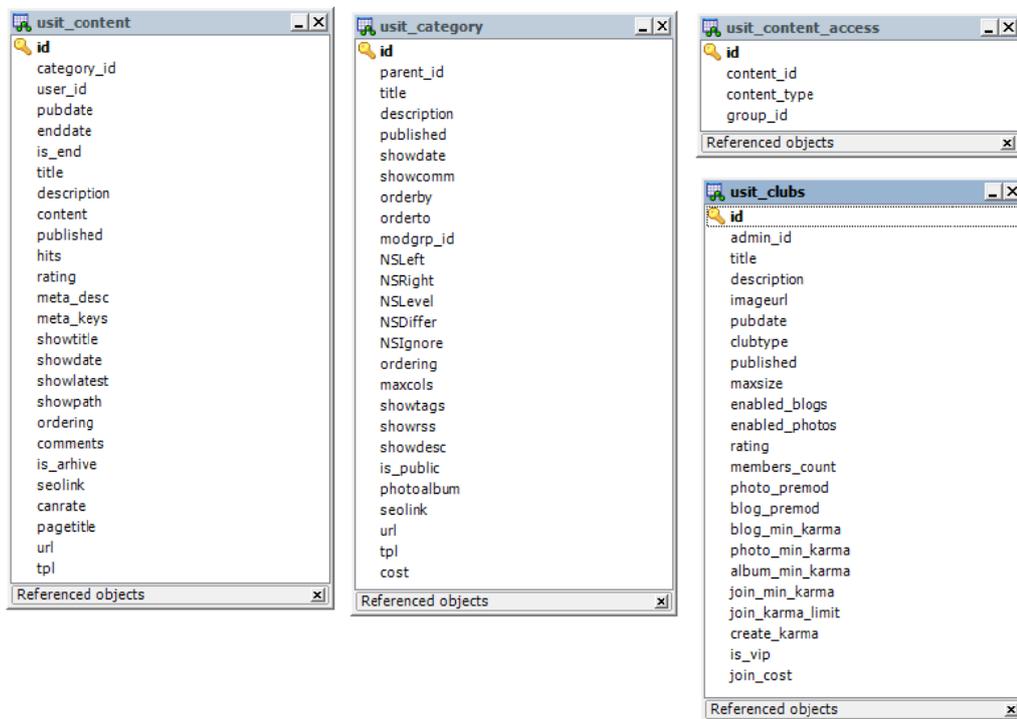


Рис.2.5.Таблицы для систематизации контента

4. Проектирование таблицы для функционирования форума

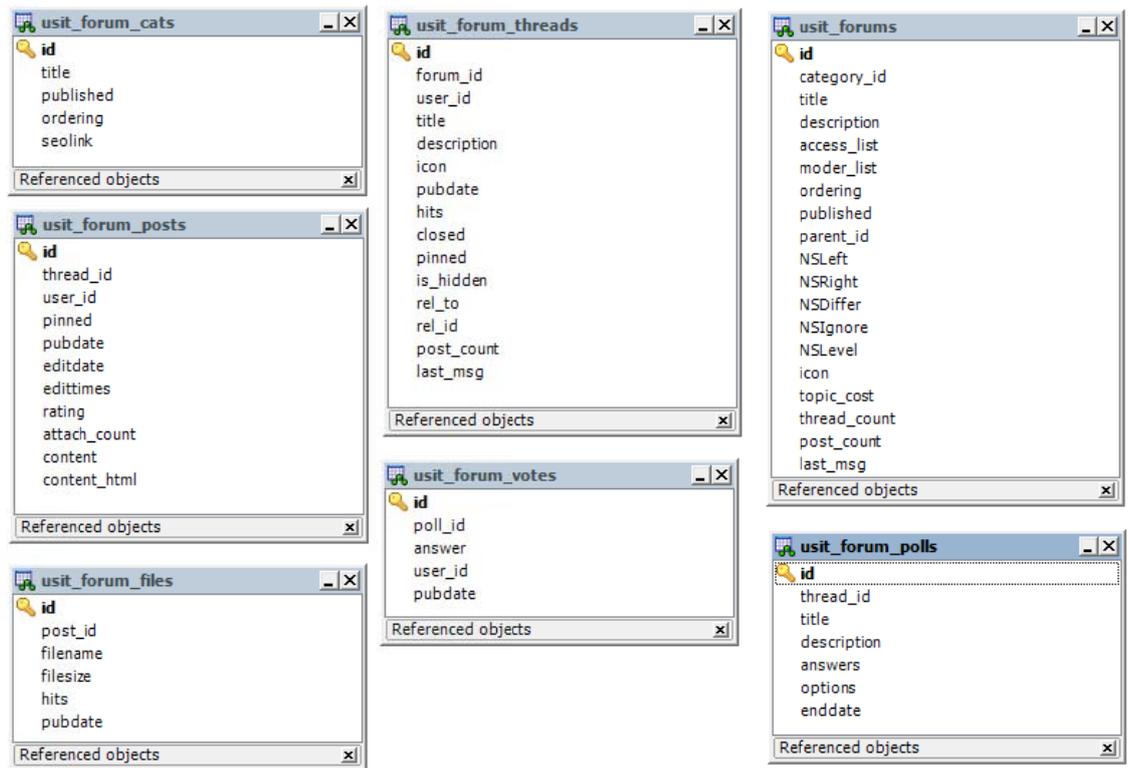
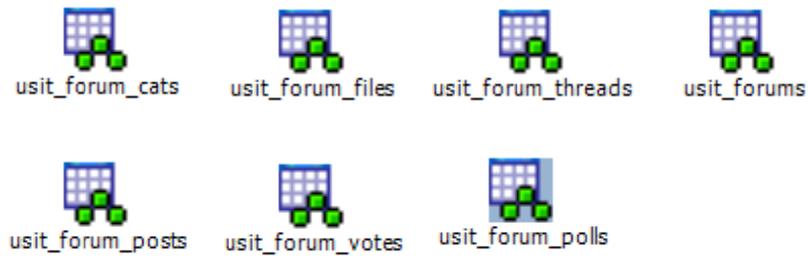


Рис.2.6. Таблицы для систематизации форума

5. Проектирование системных таблиц

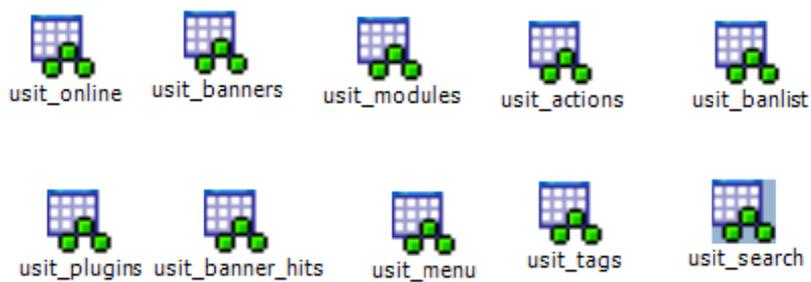


Рис.2.7. Системные таблицы

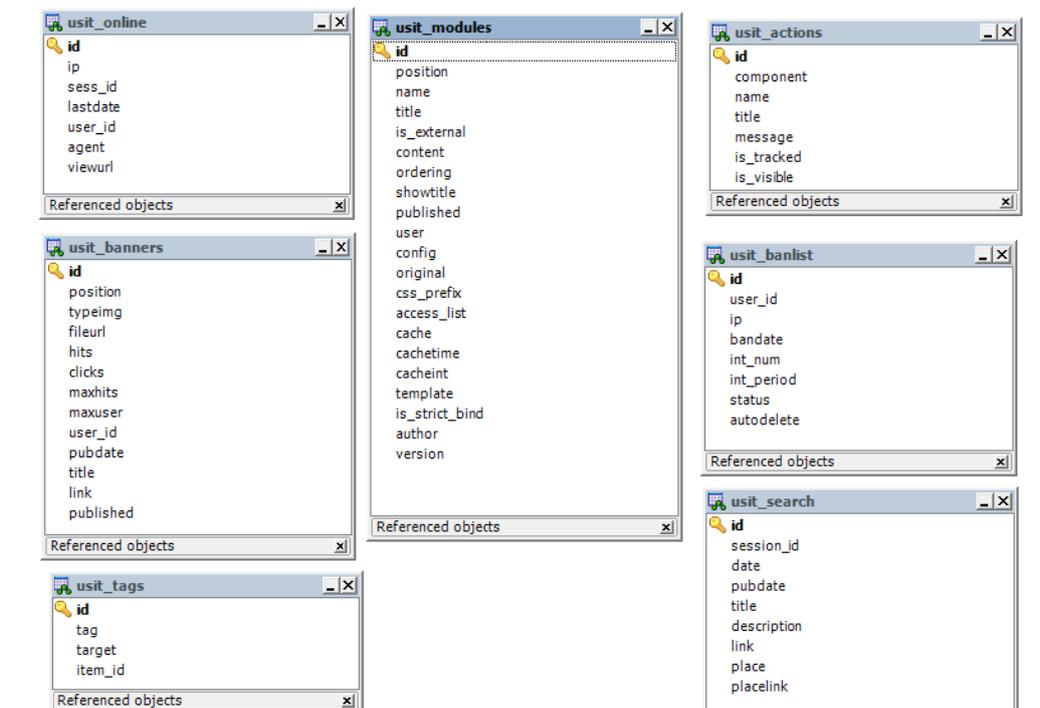


Рис.2.8. Системные таблицы(для модулей, для баннеров и т.д.)

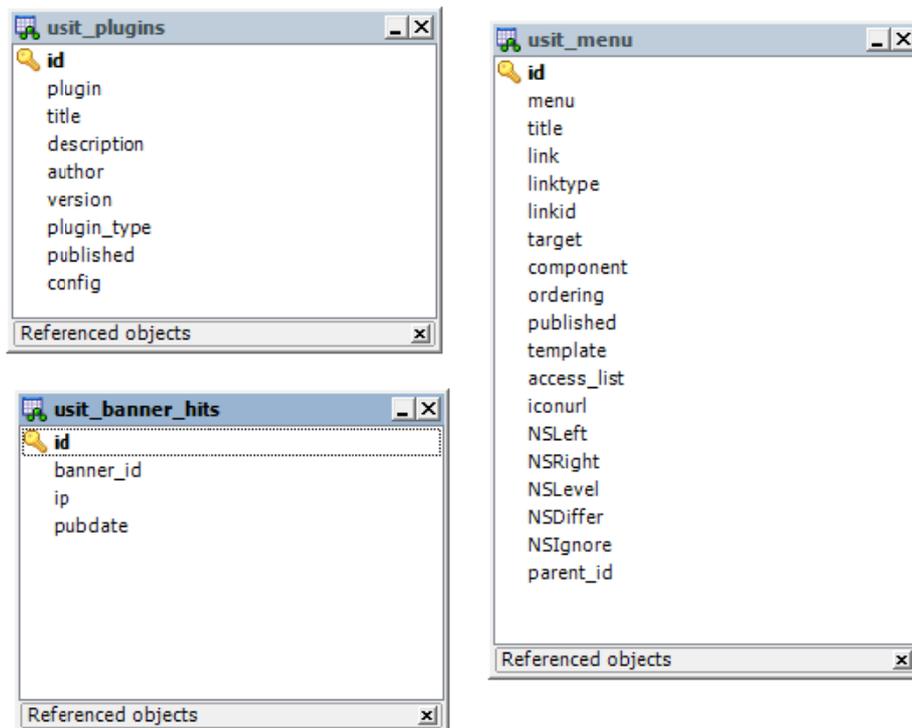


Рис.2.9.Системные таблицы(для плагины, меню)

2.2. Подготовка БД

MySQL является наиболее приспособленной для применения в среде web СУБД (системой управления базами данных). Не секрет, что для исполнения приложений клиента на большинстве хостинг-площадок провайдеры предоставляют небольшое количество ресурсов (как вычислительных, так и дисковых). Поэтому для данного применения необходима высокоэффективная СУБД, обладающая при этом высокой надежностью (большинство web-приложений и сайтов должны работать в режиме 24/7). По всем этим причинам MySQL стала незыблемым стандартом в области СУБД для web, а теперь в ней развиваются возможности для использования ее в любых критичных бизнес-приложениях, то есть конкурирует на равных с такими СУБД таких производителей, как Oracle, IBM, Microsoft и Sybase.

Основные преимущества MySQL:

- многопоточность, поддержка нескольких одновременных запросов;
- оптимизация связей с присоединением многих данных за один проход;
- записи фиксированной и переменной длины;
- ODBC драйвер;
- гибкая система привилегий и паролей;
- гибкая поддержка форматов чисел, строк переменной длины и меток времени;
- интерфейс с языками C и Perl, PHP;
- быстрая работа, масштабируемость;
- совместимость с ANSI SQL;
- бесплатна в большинстве случаев;
- хорошая поддержка со стороны провайдеров услуг хостинга;
- быстрая поддержка транзакций через механизм InnoDB.

Описание операционной системы Debian 7.

Debian — это организация, полностью состоящая из добровольцев, посвятившая себя разработке свободного программного обеспечения и распространению идеалов сообщества Свободного Программного Обеспечения (FreeSoftware). Проект Debian начался в 1993 году, когда Иэн Мёрдок (IanMurdock) выпустил открытое приглашение к разработчикам программ внести свой вклад в полноценный программный дистрибутив, основанный на относительно новом ядре Linux. Небольшая группа энтузиастов, на которых повлияла философия GNU (<http://www.gnu.org/gnu/the-gnu-project.html>), изначально поддерживаемая Фондом Свободного Программного Обеспечения

(<http://www.fsf.org/>), с годами выросла в организацию, с количеством разработчиков Debian.

Разработчики Debian выполняют различные задачи, включая администрирование Web (<http://www.debian.org/>) и FTP (<ftp://ftp.debian.org/>) сайтов, графический дизайн, юридический анализ программных лицензий, написание документации и, конечно же, сопровождение пакетов.

В интересах распространения нашей философии и привлечения разработчиков, которые верят в принципы Debian, проект Debian выпустил множество документов, которые подчёркивают наши ценности и рассказывают, что значит быть разработчиком Debian:

- **Общественный Договор Debian** (http://www.debian.org/social_contract) — это утверждение обязательств Debian перед Сообществом Свободного Программного Обеспечения.

Любой, кто соглашается следовать Общественному Договору Debian может стать сопровождающим (<http://www.debian.org/doc/maint-guide/>). Любой сопровождающий может добавить новые программы в Debian — при условии, что программы соответствуют нашим критериям свободы, а пакеты следуют нашим стандартам качества.

Разработчики Debian также вовлечены и в другие проекты, некоторые из которых созданы специально для Debian, другие выполняются для части или для всего сообщества Linux. Вот несколько примеров:

- Проект LinuxStandardBase (<http://www.linuxbase.org/>) (LSB, Основа Стандарта Linux) нацелен на стандартизацию базовой системы GNU/Linux, что позволит сторонним

- Стандарт Иерархии Файловой Системы (<http://www.pathname.com/fhs/>) (FHS) — это попытка стандартизировать организацию файловой системы Linux. Стандарт FHS позволит разработчикам программ сконцентрировать свои усилия на создании программы и не беспокоиться о том, как пакет будет устанавливаться в различные дистрибутивы GNU/Linux.

- Внутренний проект Debian для детей (<http://www.debian.org/devel/debian-jr/>), его цель — уверенность в том, что Debian'у есть что предложить нашим младшим пользователям.

Получение Debian

Информацию о том, как загрузить Debian GNU/Linux из Интернет или у кого можно купить официальные компакт-диски с Debian, смотрите на странице распространения (<http://www.debian.org/distrib/>). Список серверов-зеркал Debian (<http://www.debian.org/distrib/ftplist>) содержит полный перечень официальных зеркал Debian, так что вы можете легко найти ближайший к вам сервер.

После установки Debian может быть легко обновлён. Процедура установки поможет настроить систему так, чтобы вы смогли сделать это обновление после завершения установки, если нужно.

История релизов Debian:

Версия	Кодовое имя	Дата выхода	Архитектуры	Пакеты	Окончание срока поддержки
0.93R6		26 октября	1		

		1995			
1.1	Базз (Buzz)	17 июня 1996	1	474	1996
1.2	Рекс (Rex)	12 декабря 1996	1	848	1996
1.3	Бо (Bo)	5 июня 1997	1	974	1997
2.0	Хэмм (Hamm)	24 июля 1998	2	≈ 1,500	1998
2.1	Слинк (Slink)	9 марта 1999	4	≈ 2,250	декабрь 2000 г.
2.2	Потэйто (Potato)	15 августа 2000	6	≈ 3,900	апрель 2003 г.
3.0	Вуди (Woody)	19 июля 2002	11	≈ 8,500	август 2006 г.
3.1	Сарж (Sarge)	6 июня 2005	11	≈ 15,400	апрель 2008 г.
4.0	Этч (Etch)	8 апреля 2007	11	≈ 18,000	февраль 2010 г.
5.0	Ленни (Lenny)	14 февраля 2009	12[15]	>23,000[15]	февраль 2012 г.[16]
6.0	Сквиз (Squeeze)[17]	6 февраля 2011[18]	9+2[A]	>29,050	будет объявлено позднее
7.0	Уизи (Wheezy)[19]	пока не объявлена	неизвестно	неизвестно	будет объявлено позднее

Установка apache, mysql, php

Для начала стоит обновить все пакеты в системе:

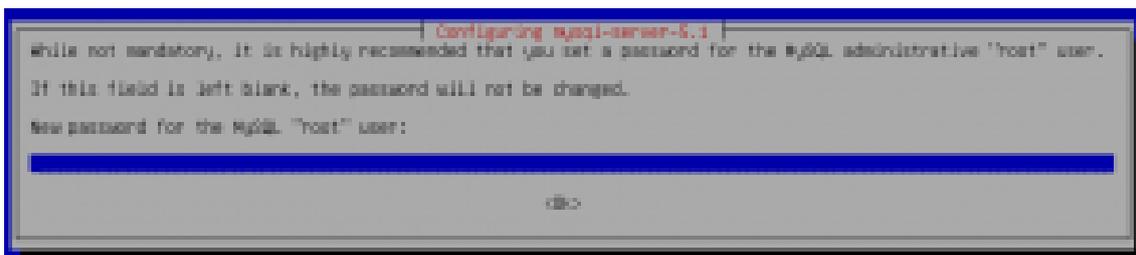
apt-get update

apt-get upgrade

Теперь можно установить нужные пакеты:

```
apt-get install apache2 php5 php5-mysql mysql-server
```

Во время установки *mysql-server* будет предложено ввести пароль для *root*:



Когда установка завершена настроим наш *apache2* так, что бы можно было держать два(или более) разных сайтов на одной машине. Для этого создадим для каждого сайта отдельные директории в папке */var/www/*.

```
mkdir /var/www/usit.uz/
```

```
mkdir /var/www/test.usit.uz/
```

Не забудем так же создать файлы логов для каждого сайта:

```
touch /var/log/apache2/usit-uz-error.log
```

```
touch /var/log/apache2/test-usit-uz-error.log
```

После этого нужно настроить конфиг для каждого сайта. Перейдем в директорию */etc/apache2/sites-available/* командой:

```
cd /etc/apache2/sites-available
```

Создаем в ней два файла с именами сайтов:

```
touch usit-uz
```

```
touch test-usit-uz
```

Теперь нужно заполнить каждый файл настройками. Открываем в любом редакторе и пишем:

```
<VirtualHost *:80>
```

```
    ServerName usit.uz
```

```
    DocumentRoot /var/www/usit.uz
```

```
    ErrorLog /var/log/apache2/usit-uz-error.log
```

```
</VirtualHost>
```

Аналогичные операции нужно проделать и со вторым файлом, лишь заменив *usit.uz* на *test.usit.uz*.

Теперь нужно активировать сайты.

a2ensiteusit.uz

a2ensite test.usit.uz

a2dissite default

Необходимо перезапустить *apache2*:

`/etc/init.d/apache2 restart`

Для удобства работы с базой данных мы можем установить *phpmyadmin*:

`apt-getinstallphpmyadmin`

Выводы ко второй главе:

В этой главе приведена краткая информация о СУБД MySQL. Приведены этапы проектирования базы данных, а также установлено все необходимое программное обеспечение (ПО).

ГЛАВА 3. ОПИСАНИЕ БД

3.1. Структура БД

База данных в выпускной квалификационной работе реализована на языке запросов SQL в качестве СУБД использовался MySQL и содержит 82 таблиц, которые связаны отношением один-ко-многим, один-к-одному а также многие-ко-многим. Нарис.3.1. – 3.14 приведена структура таблиц использующая полную информацию о пользователях, в которых рассмотрены следующие разделы базы данных: учет пользователей, альбомы пользователей, наград пользователей, таблица дружба пользователей, пользовательские группы, гости, карма пользователей, сообщения, пользовательские фотографии и профили, стена пользователей, Бан-лист.

#	Name	Datatype	Length/Set	Unsigned
 1	id	INT	11	<input type="checkbox"/>
 2	group_id	INT	11	<input type="checkbox"/>
 3	login	VARCHAR	100	
4	nickname	VARCHAR	100	
5	password	VARCHAR	100	
 6	email	VARCHAR	100	
7	icq	VARCHAR	15	
8	regdate	DATETIME		
9	logdate	DATETIME		
 10	birthdate	DATE		
11	is_locked	TINYINT	1	<input type="checkbox"/>
12	is_deleted	TINYINT	1	<input type="checkbox"/>
13	is_logged_once	TINYINT	1	<input type="checkbox"/>
14	rating	INT	11	<input type="checkbox"/>
15	points	INT	11	<input type="checkbox"/>
16	last_ip	VARCHAR	15	
17	status	VARCHAR	255	
18	status_date	DATETIME		
 19	invited_by	INT	11	<input type="checkbox"/>
20	invdate	DATETIME		
 21	openid	VARCHAR	250	

Рис.3.1. Таблица user

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	user_id	INT	11	<input type="checkbox"/>
3	title	VARCHAR	100	
4	pubdate	DATETIME		
5	allow_who	VARCHAR	10	
6	description	VARCHAR	200	

Рис.3.2. Таблица sit_user_albums

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	title	VARCHAR	200	
3	description	VARCHAR	200	
4	imageurl	VARCHAR	200	
5	p_comment	INT	11	<input type="checkbox"/>
6	p_blog	INT	11	<input type="checkbox"/>
7	p_forum	INT	11	<input type="checkbox"/>
8	p_photo	INT	11	<input type="checkbox"/>
9	p_privphoto	INT	11	<input type="checkbox"/>
10	p_content	INT	11	<input type="checkbox"/>
11	p_karma	INT	11	<input type="checkbox"/>
12	published	TINYINT	1	<input type="checkbox"/>

Рис.3.3. Таблица sit_user_autoawards

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	user_id	INT	11	<input type="checkbox"/>
3	pubdate	DATETIME		
4	title	VARCHAR	250	
5	description	TEXT		
6	imageurl	VARCHAR	100	
7	from_id	INT	11	<input type="checkbox"/>
8	award_id	INT	11	<input type="checkbox"/>

Рис.3.4. Таблица sit_user_awards

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	to_id	INT	11	<input type="checkbox"/>
3	from_id	INT	11	<input type="checkbox"/>
4	logdate	DATETIME		
5	is_accepted	TINYINT	1	<input type="checkbox"/>

Рис.3.5. Таблица sit_user_friends

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	title	VARCHAR	200	
3	alias	VARCHAR	100	
4	is_admin	TINYINT	1	<input type="checkbox"/>
5	access	TEXT		

Рис.3.6. Таблица usit_user_groups

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	user_id	INT	11	<input type="checkbox"/>
3	guest_id	INT	11	<input type="checkbox"/>
4	date	TIMESTAMP		

Рис.3.7. Таблица usit_user_guests

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	user_id	INT	11	<input type="checkbox"/>
3	sender_id	INT	11	<input type="checkbox"/>
4	points	SMALLINT	6	<input type="checkbox"/>
5	senddate	DATETIME		

Рис.3.8. Таблица usit_user_karma

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	to_id	INT	11	<input type="checkbox"/>
3	from_id	INT	11	<input type="checkbox"/>
4	senddate	DATETIME		
5	is_new	TINYINT	1	<input type="checkbox"/>
6	message	TEXT		
7	to_del	TINYINT	1	<input type="checkbox"/>
8	from_del	TINYINT	1	<input type="checkbox"/>

Рис.3.9. Таблица usit_user_msg

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	user_id	INT	11	<input type="checkbox"/>
3	album_id	INT	11	<input type="checkbox"/>
4	pubdate	DATETIME		
5	title	VARCHAR	250	
6	description	TEXT		
7	allow_who	VARCHAR	15	
8	hits	INT	11	<input type="checkbox"/>
9	imageurl	VARCHAR	250	

Рис.3.10. Таблица usit_user_photos

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	user_id	INT	11	<input type="checkbox"/>
3	city	VARCHAR	250	
4	description	VARCHAR	500	
5	showmail	TINYINT	1	<input type="checkbox"/>
6	showbirth	TINYINT	1	<input type="checkbox"/>
7	showicq	TINYINT	1	<input type="checkbox"/>
8	karma	INT	11	<input type="checkbox"/>
9	imageurl	VARCHAR	250	
10	allow_who	VARCHAR	35	
11	signature	VARCHAR	240	
12	signature_html	VARCHAR	300	
13	gender	VARCHAR	1	
14	formsdata	VARCHAR	800	
15	email_newmsg	INT	11	<input type="checkbox"/>
16	cm_subscribe	VARCHAR	4	
17	stats	VARCHAR	500	

Рис.3.11. Таблица usit_user_profiles

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	user_id	INT	11	<input type="checkbox"/>
3	author_id	INT	11	<input type="checkbox"/>
4	pubdate	DATETIME		
5	content	TEXT		
6	usertype	VARCHAR	8	

Рис.3.12. Таблица usit_user_wall

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	user_id	INT	11	<input type="checkbox"/>
3	ip	VARCHAR	15	
4	bandate	DATETIME		
5	int_num	INT	11	<input type="checkbox"/>
6	int_period	VARCHAR	20	
7	status	TINYINT	1	<input type="checkbox"/>
8	autodelete	TINYINT	1	<input type="checkbox"/>

Рис.3.13. Таблица usit_banlist

Следующие категории таблиц рассматривают основной контент для функционирования ИС состоит из следующих таблиц: блоги, категории, клубы пользователей, комментарии пользователей к фотографиям, блогам, статьям, контент, формы, форум, меню, фотоальбомы и фотографии.

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	user_id	INT	11	<input type="checkbox"/>
3	blog_id	INT	11	<input type="checkbox"/>
4	description	VARCHAR	100	
5	startdate	DATETIME		

Рис.3.14. Таблица usit_blog_authors

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	blog_id	INT	11	<input type="checkbox"/>
3	title	VARCHAR	250	
4	description	VARCHAR	1000	

Рис.3.15. Таблица usit_blog_cats

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	user_id	INT	11	<input type="checkbox"/>
3	cat_id	INT	11	<input type="checkbox"/>
4	blog_id	INT	11	<input type="checkbox"/>
5	pubdate	DATETIME		
6	title	VARCHAR	250	
7	feel	VARCHAR	100	
8	music	VARCHAR	100	
9	content	TEXT		
10	content_html	TEXT		
11	allow_who	VARCHAR	20	
12	edit_times	INT	11	<input type="checkbox"/>
13	edit_date	TIMESTAMP		
14	published	TINYINT	1	<input type="checkbox"/>
15	seolink	VARCHAR	255	
16	comments	TINYINT	1	<input type="checkbox"/>
17	comments_count	INT	11	<input type="checkbox"/>
18	rating	INT	11	<input type="checkbox"/>

Рис.3.16. Таблица usit_blog_posts

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	user_id	INT	11	<input type="checkbox"/>
3	title	VARCHAR	250	
4	pubdate	TIMESTAMP		
5	allow_who	VARCHAR	15	
6	view_type	VARCHAR	15	
7	showcats	TINYINT	1	<input type="checkbox"/>
8	ownertype	VARCHAR	15	
9	premod	TINYINT	1	<input type="checkbox"/>
10	forall	TINYINT	1	<input type="checkbox"/>
11	owner	VARCHAR	10	
12	seolink	VARCHAR	255	
13	rating	INT	11	<input type="checkbox"/>
14	comments_count	INT	11	<input type="checkbox"/>

Рис.3.17. Таблица usit_blogs

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	parent_id	INT	11	<input type="checkbox"/>
3	title	VARCHAR	200	
4	description	TEXT		
5	published	TINYINT	1	<input type="checkbox"/>
6	showdate	TINYINT	1	<input type="checkbox"/>
7	showcomm	TINYINT	1	<input type="checkbox"/>
8	orderby	VARCHAR	30	
9	orderto	VARCHAR	4	
10	modgrp_id	INT	11	<input type="checkbox"/>
11	NSLeft	INT	11	<input type="checkbox"/>
12	NSRight	INT	11	<input type="checkbox"/>
13	NSLevel	INT	11	<input type="checkbox"/>
14	NSDiffer	VARCHAR	11	
15	NSIgnore	INT	11	<input type="checkbox"/>
16	ordering	INT	11	<input type="checkbox"/>
17	maxcols	INT	11	<input type="checkbox"/>
18	showtags	TINYINT	1	<input type="checkbox"/>
19	showrss	TINYINT	1	<input type="checkbox"/>
20	showdesc	TINYINT	1	<input type="checkbox"/>
21	is_public	TINYINT	1	<input type="checkbox"/>
22	photoalbum	TEXT		
23	seolink	VARCHAR	200	

Рис.3.18. Таблица usit_category

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	admin_id	INT	11	<input type="checkbox"/>
3	title	VARCHAR	200	
4	description	LONGTEXT		
5	imageurl	VARCHAR	100	
6	pubdate	TIMESTAMP		
7	clubtype	VARCHAR	15	
8	published	TINYINT	1	<input type="checkbox"/>
9	maxsize	INT	11	<input type="checkbox"/>
10	enabled_blogs	TINYINT	1	<input type="checkbox"/>
11	enabled_photos	TINYINT	1	<input type="checkbox"/>
12	rating	INT	11	<input type="checkbox"/>
13	members_count	INT	11	<input type="checkbox"/>
14	photo_premod	TINYINT	1	<input type="checkbox"/>
15	blog_premod	TINYINT	1	<input type="checkbox"/>
16	blog_min_karma	INT	11	<input type="checkbox"/>
17	photo_min_karma	INT	11	<input type="checkbox"/>
18	album_min_karma	INT	11	<input type="checkbox"/>
19	join_min_karma	INT	11	<input type="checkbox"/>
20	join_karma_limit	INT	11	<input type="checkbox"/>
21	create_karma	INT	11	<input type="checkbox"/>
22	is_vip	TINYINT	1	<input type="checkbox"/>
23	ioin cost	FLOAT		<input type="checkbox"/>

Рис.3.19. Таблица usit_clubs

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	target	VARCHAR	32	
3	component	VARCHAR	32	
4	title	VARCHAR	100	
5	target_table	VARCHAR	32	
6	subj	VARCHAR	100	

Рис.3.20. Таблица usit_comment_targets

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	parent_id	INT	11	<input type="checkbox"/>
3	pid	INT	11	<input type="checkbox"/>
4	user_id	INT	11	<input type="checkbox"/>
5	target	VARCHAR	20	
6	target_id	INT	11	<input type="checkbox"/>
7	guestname	VARCHAR	200	
8	content	TEXT		
9	content_bbcode	TEXT		
10	pubdate	TIMESTAMP		
11	published	TINYINT	1	<input type="checkbox"/>
12	is_new	TINYINT	1	<input type="checkbox"/>
13	target_title	VARCHAR	150	
14	target_link	VARCHAR	200	
15	ip	VARCHAR	15	
16	is_hidden	TINYINT	1	<input type="checkbox"/>
17	rating	INT	11	<input type="checkbox"/>

Рис.3.21. Таблица usit_comments

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	category_id	INT	11	<input type="checkbox"/>
3	user_id	INT	11	<input type="checkbox"/>
4	pubdate	DATETIME		
5	enddate	DATE		
6	is_end	TINYINT	1	<input type="checkbox"/>
7	title	VARCHAR	200	
8	description	TEXT		
9	content	LONGTEXT		
10	published	TINYINT	1	<input type="checkbox"/>
11	hits	INT	11	<input type="checkbox"/>
12	rating	INT	11	<input type="checkbox"/>
13	meta_desc	TEXT		
14	meta_keys	TEXT		
15	showtitle	TINYINT	1	<input type="checkbox"/>
16	showdate	TINYINT	1	<input type="checkbox"/>
17	showlatest	TINYINT	1	<input type="checkbox"/>
18	showpath	TINYINT	1	<input type="checkbox"/>
19	ordering	INT	11	<input type="checkbox"/>
20	comments	TINYINT	1	<input type="checkbox"/>
21	is_arhive	TINYINT	1	<input type="checkbox"/>
22	seolink	VARCHAR	200	
23	canrate	TINYINT	1	<input type="checkbox"/>

Рис.3.22. Таблица usit_content

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	content_id	INT	11	<input type="checkbox"/>
3	content_type	VARCHAR	100	
4	group_id	INT	11	<input type="checkbox"/>

Рис.3.23. Таблица usit_content_access

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	form_id	INT	11	<input type="checkbox"/>
3	title	VARCHAR	200	
4	description	VARCHAR	200	
5	ordering	INT	11	<input type="checkbox"/>
6	kind	VARCHAR	30	
7	mustbe	INT	11	<input type="checkbox"/>
8	config	TEXT		

Рис.3.24. Таблица usit_form_fields

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	title	VARCHAR	200	
3	description	TEXT		
4	email	VARCHAR	200	
5	sendto	VARCHAR	4	
6	user_id	INT	11	<input type="checkbox"/>
7	form_action	VARCHAR	200	
8	only_fields	TINYINT	1	<input type="checkbox"/>
9	showtitle	TINYINT	1	<input type="checkbox"/>

Рис.3.25. Таблица usit_forms

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	title	VARCHAR	250	
3	published	TINYINT	1	<input type="checkbox"/>
4	ordering	INT	11	<input type="checkbox"/>
5	seolink	VARCHAR	200	

Рис.3.26. Таблица usit_forum_cats

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	post_id	INT	11	<input type="checkbox"/>
3	filename	VARCHAR	200	
4	filesize	INT	11	<input type="checkbox"/>
5	hits	INT	11	<input type="checkbox"/>
6	pubdate	DATETIME		

Рис.3.27. Таблица usit_forum_files

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	thread_id	INT	11	<input type="checkbox"/>
3	title	VARCHAR	200	
4	description	TEXT		
5	answers	TEXT		
6	options	VARCHAR	250	
7	enddate	DATETIME		

Рис.3.29. Таблица usit_forum_polls

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	thread_id	INT	11	<input type="checkbox"/>
3	user_id	INT	11	<input type="checkbox"/>
4	pinned	TINYINT	1	<input type="checkbox"/>
5	pubdate	DATETIME		
6	editdate	DATETIME		
7	edittimes	INT	11	<input type="checkbox"/>
8	rating	INT	11	<input type="checkbox"/>
9	attach_count	INT	11	<input type="checkbox"/>
10	content	TEXT		
11	content_html	TEXT		

Рис.3.29. Таблица usit_forum_posts

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	forum_id	INT	11	<input type="checkbox"/>
3	user_id	INT	11	<input type="checkbox"/>
4	title	VARCHAR	250	
5	description	VARCHAR	250	
6	icon	VARCHAR	100	
7	pubdate	DATETIME		
8	hits	INT	11	<input type="checkbox"/>
9	closed	TINYINT	1	<input type="checkbox"/>
10	pinned	TINYINT	1	<input type="checkbox"/>
11	is_hidden	TINYINT	1	<input type="checkbox"/>
12	rel_to	VARCHAR	15	
13	rel_id	INT	11	<input type="checkbox"/>
14	post_count	INT	11	<input type="checkbox"/>
15	last_msg	TEXT		

Рис.3.30. Таблица usit_forum_theaeds

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	poll_id	INT	11	<input type="checkbox"/>
3	answer	VARCHAR	300	
4	user_id	INT	11	<input type="checkbox"/>
5	pubdate	DATETIME		

Рис.3.31. Таблица usit_forum_votes

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	category_id	INT	11	<input type="checkbox"/>
3	title	VARCHAR	250	
4	description	VARCHAR	300	
5	access_list	TINYTEXT		
6	moder_list	TINYTEXT		
7	ordering	INT	11	<input type="checkbox"/>
8	published	TINYINT	1	<input type="checkbox"/>
9	parent_id	INT	11	<input type="checkbox"/>
10	NSLeft	INT	11	<input type="checkbox"/>
11	NSRight	INT	11	<input type="checkbox"/>
12	NSDiffer	VARCHAR	15	
13	NSIgnore	INT	11	<input type="checkbox"/>
14	NSLevel	INT	11	<input type="checkbox"/>
15	icon	VARCHAR	200	
16	topic_cost	FLOAT		<input type="checkbox"/>
17	thread_count	INT	11	<input type="checkbox"/>
18	post_count	INT	11	<input type="checkbox"/>
19	last_msg	TEXT		

Рис.3.32. Таблица usit_forums

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	menu	VARCHAR	200	
3	title	VARCHAR	200	
4	link	VARCHAR	200	
5	linktype	VARCHAR	12	
6	linkid	VARCHAR	25	
7	target	VARCHAR	8	
8	component	VARCHAR	100	
9	ordering	INT	11	<input type="checkbox"/>
10	published	TINYINT	1	<input type="checkbox"/>
11	template	VARCHAR	30	
12	access_list	TINYTEXT		
13	iconurl	VARCHAR	100	
14	NSLeft	INT	11	<input type="checkbox"/>
15	NSRight	INT	11	<input type="checkbox"/>
16	NSLevel	INT	11	<input type="checkbox"/>
17	NSDiffer	VARCHAR	40	
18	NSIgnore	INT	11	<input type="checkbox"/>
19	parent_id	INT	11	<input type="checkbox"/>

Рис.3.33. Таблица usit_menu

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	ip	VARCHAR	15	
3	sess_id	VARCHAR	100	
4	lastdate	TIMESTAMP		
5	user_id	INT	11	<input type="checkbox"/>
6	agent	VARCHAR	250	
7	viewurl	VARCHAR	250	

Рис.3.34. Таблица usit_online

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	parent_id	INT	11	<input type="checkbox"/>
3	ordering	INT	11	<input type="checkbox"/>
4	NSLeft	INT	11	<input type="checkbox"/>
5	NSRight	INT	11	<input type="checkbox"/>
6	NSDiffer	VARCHAR	15	
7	NSIgnore	INT	11	<input type="checkbox"/>
8	NSLevel	INT	11	<input type="checkbox"/>
9	title	VARCHAR	100	
10	description	VARCHAR	300	
11	published	TINYINT	1	<input type="checkbox"/>
12	showdate	TINYINT	1	<input type="checkbox"/>
13	iconurl	VARCHAR	100	
14	pubdate	TIMESTAMP		
15	orderby	VARCHAR	20	
16	orderto	VARCHAR	4	
17	public	TINYINT	1	<input type="checkbox"/>
18	perpage	INT	11	<input type="checkbox"/>
19	cssprefix	VARCHAR	20	
20	thumb1	INT	11	<input type="checkbox"/>
21	thumb2	INT	11	<input type="checkbox"/>
22	thumbsqr	TINYINT	1	<input type="checkbox"/>

Рис.3.35. Таблица usit_photo_albums

#	Name	Datatype	Length/Set	Unsigned
1	id	INT	11	<input type="checkbox"/>
2	album_id	INT	11	<input type="checkbox"/>
3	title	VARCHAR	200	
4	description	TEXT		
5	pubdate	TIMESTAMP		
6	file	VARCHAR	200	
7	published	TINYINT	1	<input type="checkbox"/>
8	hits	INT	11	<input type="checkbox"/>
9	showdate	TINYINT	1	<input type="checkbox"/>
10	comments	TINYINT	1	<input type="checkbox"/>
11	user_id	INT	11	<input type="checkbox"/>
12	owner	VARCHAR	10	
13	rating	INT	11	<input type="checkbox"/>

Рис.3.36. Таблица usit_photo_files

3.2. Создание таблицы

В структуру MySQL входят базы данных, таблицы, записи и поля. В базах данных собраны таблицы, в таблицах собраны записи, в записях — поля, в которых, в свою очередь, хранится собственно информация. Такая структура позволяет пользователям выбирать, каким образом обращаться к своей информации.

Значит, нужно создать несколько таблиц. Чтобы это сделать, используем команду CREATE TABLE. Однако, эта команда довольно сильно отличается от простой команды CREATE DATABASE. Ей передается список аргументов следующего вида:

```
CREATE TABLE [имя_таблицы] ([имя_поля]
[тип_данных_поля]([размер]));
```

имя_таблицы — имя создаваемой таблицы.

Заполнение базы данных MySQL Таблица заполняется (данные добавляются) командой INSERT.

```
INSERT INTO имя_таблицы (столбец1, столбец2, ...)
VALUES('знач1','знач2',...);
```

Эту команду используют для вставки записей в таблицу. Сначала указывается имя таблицы, в которую нужно добавить информацию. Затем может идти список столбцов, в которые добавляются данные, а в VALUES указываются значения, которые добавляются в таблицу. Можно опустить список полей, если значения добавляются в каждое в том же порядке, в каком определены поля в таблице.

MySQL типы данных MySQL — реляционная СУБД, которая поддерживает несколько типов данных, которые можно разделить на три категории: числовые данные, дата и время, и данные типа строка. Для более подробной информации смотрите руководство по MySQL.

Тип данных	Использование	Диапазоны
TINYINT	Очень маленькое	Диапазон числа со знаком от –

	целое число	128 до 127. Диапазон числа без знака (unsigned) от 0 до 255.
SMALLINT	Маленькое целое число	Диапазон числа со знаком от -32768 до 32767. Диапазон числа без знака (unsigned) от 0 до 65535.
MEDIUMINT	Среднее целое число	Диапазон числа со знаком от -8388608 до 8388607. Диапазон числа без знака (unsigned) от 0 до 16777215.
INT или INTEGER	Целое число	Диапазон числа со знаком от -2147483648 до 2147483647. Диапазон числа без знака (unsigned) от 0 до 4294967295.
BIGINT	Большое целое число	Диапазон числа со знаком от -9223372036854775808 до 9223372036854775807. Диапазон числа без знака (unsigned) от 0 до 18446744073709551615.
FLOAT	Малое (одинарной точности) число с плавающей запятой. Не может быть числом без знака	Диапазоны от -3.402823466E+38 до -1.175494351E-38, 0 и 1.175494351E-38 до 3.402823466E+38. Если количество знаков после запятой не установлено или ≤ 24 это число с плавающей запятой одинарной точности.
DOUBLE, DOUBLE PRECISION, REAL	Нормальное (двойной точности) число с плавающей запятой. Не может быть числом без знака	Диапазоны от -1.7976931348623157E+308 до -2.2250738585072014E-308, 0 и 2.2250738585072014E-308 до 1.7976931348623157E+308. Если количество знаков после запятой не установлены или $25 \leq$ количество знаков ≤ 53 означает число с плавающей запятой двойной точности.
DECIMAL, NUMERIC	Распакованное число с плавающей запятой	Работает подобно типу данных CHAR: «распакованный» означает, что число хранится в виде строки, используя один символ для каждой цифры-

		значения. Символ десятичной запятой и символ отрицательного числа "-" не учитывается в длину. Если десятичное значение равно 0, значение не будет иметь десятичной запятой или дробной части. Максимальный размер для DECIMAL значение такое же, как и для DOUBLE, но фактический диапазон для данного столбца DECIMAL может быть ограничен в выборе длины и десятичные дроби.
DATE	Дата	Дата в диапазоне от «1000-01-01» до «9999-12-31». MySQL хранит поле типа DATE в виде «YYYY-MM-DD» (ГГГГ-ММ-ДД).
DATETIME	Дата и время	Допустимые диапазоны от «1000-01-01 00:00:00» до «9999-12-31 23:59:59». MySQL хранит поле типа DATETIME в виде «YYYY-MM-DD HH:MM:SS» (ГГГГ-ММ-ДД ЧЧ-ММ-СС).
TIMESTAMP	Дата и время	Диапазон от «1970-01-01 00:00:00» до, примерно, 2037 года. MySQL может хранить поле типа TIMESTAMP в видах «YYYYMMDDHHMMSS» (TIMESTAMP(14)), «YYMMDDHHMMSS» (TIMESTAMP(12)), «YYYYMMDD» (TIMESTAMP(8)) и др.
TIME	Время	Диапазон от «-838:59:59» до «838:59:59». MySQL хранит поле TIME в виде «HH:MM:SS», но позволяет присваивать значения столбцам TIME с использованием либо строки или числа.

YEAR	Год в 2- или 4-хцифровом виде (4 цифры по умолчанию)	Если вы используете 4 цифра, то допустимые значения 1901-2155, и 0000. Если 2 цифры, то 1970-2069 (70-69). MySQL хранит значения поля YEAR в формате «YYYY».
CHAR	Строка фиксированной длины, которая справа дополняется пробелами до указанной длины, при хранении	Диапазон длины от 1 до 255 символов. Завершающие пробелы удаляются, когда значение извлекается. Значения CHAR сортируются и сравниваются без учета регистра в зависимости от кодировки по умолчанию, если не установлен флаг BINARY.
VARCHAR	Строка переменной длины. Примечание: конечные пробелы удаляются при сохранении (в отличие от спецификации ANSI SQL).	Диапазон длины от 1 до 255 символов. Значения VARCHAR сортируются и сравниваются без учета регистра, если не установлен флаг BINARY.
TINYBLOB, TINYTEXT		BLOB или TEXT с максимальной длиной 255 ($2^8 - 1$) символов.
BLOB, TEXT		BLOB или TEXT с максимальной длиной 65535 ($2^{16} - 1$) символов.
MEDIUMBLOB, MEDIUMTEXT		BLOB или TEXT с максимальной длиной 16777215 ($2^{24} - 1$) символов.
LOB, LONGTEXT		BLOB или TEXT с максимальной длиной 4294967295 ($2^{32} - 1$) символов.
ENUM	Перечисление	Строка-объект, который может принимать только одно значение, выбирается из списка значений «значение 1»,

		«значение 2» или NULL. ENUM максимум может иметь 65535 различных значений.
SET	Набор	Строка-объект, который может принимать ноль и более значений, каждую из которых должно быть выбрано из списка значений «значение 1», «значение 2», ... Поле SET может иметь максимум 64 варианта значений.

3.3. Создания тестовых данных

Для проверки работоспособности и производительности базы необходимо определить тестовые данные. Для входных данных рассмотренных ранее создаются тестовые запросы которые реализованы с помощью SQL.

```
INSERT INTO `usit_users` (`id`, `group_id`, `login`, `nickname`, `password`,
`email`, `icq`, `regdate`, `logdate`, `birthdate`, `is_locked`, `is_deleted`,
`is_logged_once`, `rating`, `points`, `last_ip`, `status`, `status_date`, `invited_by`,
`invdate`, `openid`) VALUES
```

```
(1, 2, 'admin', 'Администратор', '12b94e4b3dc7c97219e8f00803c1258b',
'admin@usit.uz', '100200300', '2012-07-23 12:41:57', '2014-04-25 06:57:30',
'1989-12-27', 0, 0, 1, 52, 0, '127.0.0.1', 'Люди были созданы для того, чтобы их
любили, а вещи были созданы для того, чтобы ими пользовались.', '2013-10-
03 21:02:36', 0, '2012-07-09 23:25:59', NULL);
```

```
INSERT INTO `usit_users_activate` (`id`, `pubdate`, `user_id`, `code`) VALUES
```

```
(14, '2013-10-18 12:08:05', 32, '0a9b88485ead7025981685ee7f8ed993'),
(15, '2013-10-18 17:54:09', 33, '5e9343d32ccb95aed0c61f9c8217a260'),
(17, '2013-10-19 01:46:16', 35, 'c92092e68e8065a326084c9ce0f81f5f'),
(18, '2013-10-19 03:00:19', 36, 'c0f023234b59c148c6df6663eebee8da'),
(20, '2013-11-18 06:47:48', 38, 'bfee27b3d17a0dbb32faecd083c7e24f'),
```

(21, '2013-11-18 15:12:55', 39, 'a59a097ba910836292c4e731f4b90de9'),
(22, '2013-11-18 17:28:10', 40, 'a4b1ecdb5e416f429aa2f58bef89baa6');

```
INSERT INTO `usit_user_albums` (`id`, `user_id`, `title`, `pubdate`, `allow_who`,  
`description`) VALUES
```

```
(2, 3, 'Мой фотоальбом', '2012-07-22 20:28:51', 'all', ''),  
(5, 1, 'Мои картинки', '2012-07-13 23:13:37', 'all', ''),  
(6, 10, 'Я!', '2013-05-14 20:43:44', 'all', 'Это Я!');
```

```
INSERT INTO `usit_blog_posts` (`id`, `user_id`, `cat_id`, `blog_id`, `pubdate`,  
`title`, `feel`, `music`, `content`, `content_html`, `allow_who`, `edit_times`,  
`edit_date`, `published`, `seolink`, `comments`, `comments_count`, `rating`)  
VALUES (6, 12, 0, 26, '2013-05-15 09:41:26', 'Python', 'Отличное', '',
```

'Python(англ. python — питон, произносится [paθ(ə)n] — пайтон; в русском языке распространено название питон) — высокоуровневый язык программирования общего назначения, ориентированный на повышение производительности разработчика и читаемости кода. Синтаксис ядра Python минималистичен. В то же время стандартная библиотека включает большой объём полезных функций.\r\n\r\nPython поддерживает несколько парадигм программирования, в том числе структурное, объектно-ориентированное, функциональное, императивное и аспектно-ориентированное. Основные архитектурные черты — динамическая типизация, автоматическое управление памятью, полная интроспекция, механизм обработки исключений, поддержка многопоточных вычислений и удобные высокоуровневые структуры данных. Код в Питоне организовывается в функции и классы, которые могут объединяться в модули (которые в свою очередь могут быть объединены в пакеты).\r\n\r\nЭталонной реализацией Python является интерпретатор CPython, поддерживающий большинство активно используемых платформ. Он распространяется под свободной лицензией PythonSoftwareFoundationLicense, позволяющей использовать его

без ограничений в любых приложениях, включая проприетарные. Есть реализации интерпретаторов для JVM (с возможностью компиляции), MSIL (с возможностью компиляции), LLVM и других. Проект PyPy предлагает реализацию Питона на самом Питоне, что уменьшает затраты на изменения языка и постановку экспериментов над новыми возможностями.

Python — активно развивающийся язык программирования, новые версии (с добавлением/изменением языковых свойств) выходят примерно раз в два с половиной года. Вследствие этого и некоторых других причин на Python отсутствуют стандарт ANSI, ISO или другие официальные стандарты, их роль выполняет CPython.

Python (англ. python — питон, произносится [ˈpaɪθ(ə)n] — пайтон; в русском языке распространено название пито́н) — высокоуровневый язык программирования общего назначения, ориентированный на повышение производительности разработчика и читаемости кода. Синтаксис ядра Python минималистичен. В то же время стандартная библиотека включает большой объём полезных функций.

Python поддерживает несколько парадигм программирования, в том числе структурное, объектно-ориентированное, функциональное, императивное и аспектно-ориентированное. Основные архитектурные черты — динамическая типизация, автоматическое управление памятью, полная интроспекция, механизм обработки исключений, поддержка многопоточных вычислений и удобные высокоуровневые структуры данных. Код в Питоне организовывается в функции и классы, которые могут объединяться в модули (которые в свою очередь могут быть объединены в пакеты).

Эталонной реализацией Python является интерпретатор CPython, поддерживающий большинство активно используемых платформ. Он распространяется под свободной лицензией PythonSoftwareFoundationLicense, позволяющей использовать его без ограничений в любых приложениях, включая проприетарные. Есть реализации интерпретаторов для JVM (с возможностью компиляции), MSIL (с возможностью компиляции), LLVM и других. Проект PyPy предлагает реализацию Питона на самом Питоне, что

уменьшает затраты на изменения языка и постановку экспериментов над новыми возможностями. Python — активно развивающийся язык программирования, новые версии (с добавлением/изменением языковых свойств) выходят примерно раз в два с половиной года. Вследствие этого и некоторых других причин на Python отсутствуют стандарт ANSI, ISO или другие официальные стандарты, их роль выполняет CPython.', 'all', 0, '2013-05-24 21:02:17', 1, 'python', 1, 0, 2);

```
INSERT INTO `usit_category` (`id`, `parent_id`, `title`, `description`, `published`, `showdate`, `showcomm`, `orderby`, `orderto`, `modgrp_id`, `NSLeft`, `NSRight`, `NSLevel`, `NSDiffer`, `NSIgnore`, `ordering`, `maxcols`, `showtags`, `showrss`, `showdesc`, `is_public`, `photoalbum`, `seolink`, `url`, `tpl`, `cost`) VALUES (54, 53, 'Интеллектуальные Системы', "", 1, 1, 1, 'pubdate', 'ASC', 0, 81, 86, 2, "", 0, 1, 1, 1, 1, 1, 0, "", 'предметы/ис', "", 'com_content_view.tpl', ""), (55, 53, 'Проектирование Информационных Систем', "", 1, 1, 1, 'title', 'ASC', 0, 87, 90, 2, "", 0, 2, 1, 1, 1, 1, 0, "", 'предметы/проектирование-ис', "", 'com_content_view.tpl', ""), (56, 54, 'Лекции', "", 1, 1, 1, 'pubdate', 'ASC', 0, 82, 83, 3, "", 0, 1, 1, 1, 1, 1, 0, "", 'предметы/ис/лекции', "", 'com_content_view.tpl', ""), (57, 55, 'Лекции', "", 1, 1, 1, 'pubdate', 'ASC', 0, 88, 89, 3, "", 0, 1, 1, 1, 1, 1, 0, "", 'предметы/проектирование-ис/лекции', "", 'com_content_view.tpl', '0'), (59, 53, 'Мобильные системы связи (МАТ)', '<p>\r\n Методички, Лекции...\r\n</p>', 1, 1, 1, 'pubdate', 'ASC', 0, 91, 94, 2, "", 0, 3, 1, 1, 1, 1, 0, "", 'untitled/untitled', "", 'com_content_view.tpl', '0'), (60, 59, 'Методички', '<p>\r\n Все методички по МАТ\r\n</p>', 1, 1, 1, 'pubdate', 'ASC', 0, 92, 93, 3, "", 0, 1, 1, 1, 1, 1, 0, "", 'untitled/untitled/untitled', "", 'com_content_view.tpl', '0'), (61, 54, 'Тесты', '<p>#160;Тесты по ИС(Со#1074;ершенно секретно)</p>', 1, 1, 1, 'pubdate', 'ASC', 0, 84, 85, 3, "", 0, 2, 1, 1, 1, 1, 0, "", 'untitled/untitled/untitled-61', "", 'com_content_view.tpl', '0');
```

```

INSERT INTO `usit_forums` (`id`, `category_id`, `title`, `description`,
`access_list`, `moder_list`, `ordering`, `published`, `parent_id`, `NSLeft`,
`NSRight`, `NSDiffer`, `NSIgnore`, `NSLevel`, `icon`, `topic_cost`,
`thread_count`, `post_count`, `last_msg`) VALUES
(1000, 0, '-- Корневой форум --', "", "", 1, 0, 0, 1, 74, "", 0, 0, "", 0, 0, 0, ""),
(1, 1, 'Общий форум', "", "", 1, 1, 1000, 2, 7, "", 0, 1, "", 0, 1, 1, '---\npubdate: 2012-
04-04 18:54:53\nid: 29\ncontent_html: "Геосинклиналь <strong>обогащает
магматический монтмориллонит</strong>, что в общем свидетельствует о
преобладании тектонических опусканий в это время. Углефикация
характерна. Порода существенна. Тектогенез, разделенные узкими
линейновытянутыми зонами выветрелых пород, <br />переоткладывает
морской авгит, образуя на границе с Западно-Карельским поднятием
своеобразную систему грабенов. Ведущий экзогенный геологический
процесс - субдукция ослабляет лакколит, так как совершенно однозначно
указывает на существование и рост в период оформления палеогеновой
поверхности выравнивания."\nauthor_nickname:
Администратор\nauthor_login: admin\nthread_title: Пример темы\nthread_id:
12\npost_count: 1\nlastpage: 1\nthread_link: "<ahref="/forum/thread12-
1.html#29">Пример темы</a>"\nuser_link: "<ahref="/users/admin"
title="Администратор">Администратор</a>"\n'),
(1002, 1, 'Подфорум1', "", "", 2, 1, 1, 5, 6, "", 0, 2, "", 0, 0, 0, '---\npubdate: 2013-04-
23 17:50:27\nid: 30\ncontent_html: eewr\nauthor_nickname:
Администратор\nauthor_login: admin\nthread_title: werwe\nthread_id:
13\npost_count: 1\nlastpage: 1\nthread_link: "<ahref="/forum/thread13-
1.html#30">werwe</a>"\nuser_link: "<ahref="/users/admin"
title="Администратор">Администратор</a>"\n'),
(1004, 1, 'Подфорум2', "", "", 1, 1, 1, 3, 4, "", 0, 2, "", 0, 0, 0, ""),
(1028, 3, 'HTML, CSS', 'HTML, CSS', "", "", 1, 1, 1027, 51, 52, "", 0, 2, "", 0, 0, 0, ""),
(1029, 3, 'JavaScript', 'JavaScript', "", "", 2, 1, 1027, 53, 54, "", 0, 2, "", 0, 0, 0, ""),

```

(1006, 2, 'Форум программистов', 'Форум программистов', "", "", 2, 1, 1000, 8, 49, "", 0, 1, "", 0, 0, 0, ""),

(1027, 3, 'Форум web-программистов', 'Форум web-программистов', "", "", 3, 1, 1000, 50, 71, "", 0, 1, "", 0, 0, 0, ""),

(1008, 2, 'C++', 'C++', "", "", 1, 1, 1006, 9, 12, "", 0, 2, "", 0, 0, 0, ""),

(1009, 2, 'C (Си)', 'C (Си)', "", "", 2, 1, 1006, 13, 14, "", 0, 2, "", 0, 0, 0, ""),

(1010, 2, 'Delphi', 'Delphi', "", "", 3, 1, 1006, 15, 16, "", 0, 2, "", 0, 0, 0, ""),

(1011, 2, 'Pascal (Паскаль)', 'Pascal (Паскаль)', "", "", 4, 1, 1006, 17, 18, "", 0, 2, "", 0, 0, 0, ""),

(1012, 2, 'Java', 'Java', "", "", 5, 1, 1006, 19, 20, "", 0, 2, "", 0, 0, 0, ""),

(1013, 2, '.NET', '.NET', "", "", 6, 1, 1006, 21, 22, "", 0, 2, "", 0, 0, 0, ""),

(1014, 2, 'Программирование мобильных систем', 'Программирование мобильных систем', "", "", 7, 1, 1006, 23, 24, "", 0, 2, "", 0, 0, 0, ""),

(1015, 2, 'Python', 'Python', "", "", 8, 1, 1006, 25, 26, "", 0, 2, "", 0, 0, 0, ""),

(1016, 2, 'Basic', 'Basic', "", "", 9, 1, 1006, 27, 28, "", 0, 2, "", 0, 0, 0, ""),

(1017, 2, 'Perl', 'Perl', "", "", 10, 1, 1006, 29, 30, "", 0, 2, "", 0, 0, 0, "");

3.4. Тестирование БД

Тестирование БД осуществляется на языке Sql, с помощью SELECT запросов. Ниже приведение основные тестовые Sql запросы:

```
SELECT `id`, `parent_id`, `title`, `description`, `published`, `showdate`,
`showcomm`, `orderby`, `orderto`, `modgrp_id`, `NSLeft`, `NSRight`, `NSLevel`,
`NSDiffer`, `NSIgnore`, `ordering`, `maxcols`, `showtags`, `showrss`, `showdesc`,
`is_public`, `photoalbum`, `seolink`, `url`, `tpl`, `cost` FROM `usit_category`;
```

```
SELECT `id`, `category_id`, `title`, `description`, `access_list`, `moder_list`,
`ordering`, `published`, `parent_id`, `NSLeft`, `NSRight`, `NSDiffer`, `NSIgnore`,
`NSLevel`, `icon`, `topic_cost`, `thread_count`, `post_count`, `last_msg` FROM
`usit_forums` WHERE `category_id` = 3;
```

```
SELECT `id`, `thread_id`, `user_id`, `pinned`, `pubdate`, `editdate`, `edittimes`,  
`rating`, `attach_count`, `content`, `content_html` FROM `usit_forum_posts`  
WHERE `user_id` = 29
```

3.5. Увеличение производительности сервера базы данных

Обзор оптимизации. Чтобы увеличить скорость системы, необходимо, разумеется, прежде всего разбираться в ее конструкции. Кроме того, нужно знать, какие функции будет выполнять система и какие "узкие места" в ней имеются. Ниже приведен список наиболее часто встречающихся "узких мест":

- Поиск данных на диске. Чтобы найти на диске какой-то фрагмент данных, требуется некоторое время. Для устройств выпуска 1999 года среднее время поиска составляет менее 10мс, так что теоретически можно выполнять приблизительно 100 операций поиска в секунду. Это время можно ненамного уменьшить, заменив диски более новыми. Для одной таблицы поиск на диске оптимизировать очень сложно. Такую оптимизацию можно выполнить путем распределения данных по нескольким дискам.
- Дисковое чтение/запись. После выполнения поиска, когда найдена соответствующая позиция на диске, мы можем считать данные. Для устройств выпуска 1999 года производительность одного диска составляет около 10-20Мб/с. Дисковое чтение/запись легче оптимизировать, чем дисковый поиск, поэтому читать можно параллельно с нескольких дисков.
- Циклы процессора. Когда мы помещаем данные в основную память (или если они уже находятся там), мы должны обработать их, чтобы получить результат. Наличие маленьких по сравнению с объемом ОЗУ

таблиц - наиболее часто встречающийся лимитирующий фактор. Но в этом случае, в общем-то, скорость обработки маленьких таблиц значения не имеет.

- Пропускная способность ОЗУ (memory bandwidth). Когда процессору требуется больше данных, чем может вместить его кэш, узким местом становится пропускная способность памяти. В большинстве систем это узкое место встречается редко, однако о нем нужно знать.

Скорость выполнения запросов INSERT. Время, необходимое для вставки записи, можно грубо разделить на такие промежутки:

- Подсоединение: (3)
- Посылка запроса на сервер: (2)
- Синтаксический анализ запроса: (2)
- Вставка записи: (1 * размер записи)
- Вставка индексов: (1 * число индексов)
- Закрытие: (1)

где числа в скобках пропорциональны полному времени. При этом не учитывается время в начале вставки, требующееся для открытия таблиц (таблицы открываются один раз для каждого конкурентно выполняющегося запроса).

Размер таблицы замедляет вставку индексов в $\log N$ раз (B-деревья).

Некоторые способы ускорения вставки:

- Если с одного клиента одновременно вставляется большое количество строк, используйте операторы INSERT в форме, содержащей множество записей. При этом вставка будет происходить намного быстрее (в некоторых случаях в несколько раз), чем при использовании отдельных операторов INSERT. При добавлении данных в непустую

таблицу можно настроить переменную `myisam_bulk_insert_tree_size` так, чтобы это делалось еще быстрее. При вставке нескольких строк с различных клиентов можно повысить скорость, используя оператор `INSERT DELAYED`.

- Обратите внимание: при использовании таблиц MyISAM можно вставлять строки во время выполнения операторов `SELECT`, если в таблицах нет удаленных строк.
- При загрузке таблицы из текстового файла используйте команду `LOAD DATA INFILE`. При этом обычно вставка будет происходить в 20 раз быстрее, чем при использовании соответствующего количества операторов `INSERT`. Если таблица имеет много индексов, можно проделать некоторую дополнительную работу, чтобы команда `LOAD DATA INFILE` выполнялась еще быстрее. Используйте следующую процедуру:
 1. При необходимости создайте таблицу при помощи оператора `CREATE TABLE` (например, используя `mysql` или `Perl-DBI`).
 2. Выполните оператор `FLUSH TABLES` или команду оболочки: `mysqladmin flush-tables`.
 3. Используйте `myisamchk --keys-used=0 -rq /path/to/db/tbl_name`. После этого индексы не будут использоваться для данной таблицы.
 4. Вставьте данные в таблицу при помощи `LOAD DATA INFILE`. При этом никакие индексы обновляться не будут и, следовательно, скорость будет высокой весьма.
 5. Если вы собираетесь в будущем только лишь читать таблицу, выполните `myisampack` для этой таблицы, чтобы уменьшить ее размер.
 6. Воссоздайте индексы при помощи команды `myisamchk -r -q /path/to/db/tbl_name`. Эта процедура создает индексное дерево в памяти, перед тем как записать его на диск, что гораздо быстрее

за счет исключения большого количества дисковых операций. Индексное дерево, получившееся в результате, к тому же отлично сбалансировано.

7. Выполните оператор `FLUSH TABLES` или команду оболочки: `mysqladmin flush-tables`.

Обратите внимание: команда `LOAD DATA INFILE` также выполняет вышеупомянутую оптимизацию при вставках в пустую таблицу. Главное отличие этой команды от вышеупомянутой процедуры заключается в том, что при помощи `myisamchk` можно выделить намного больше временной памяти для создания индекса, чем MySQL, по вашему мнению, должен выделять для каждого воссоздания индексов. Начиная с MySQL 4.0 можно также использовать команду `ALTER TABLE tbl_name DISABLE KEYS` вместо `myisamchk --keys-used=0 -rq /path/to/db/tbl_name` и `ALTER TABLE tbl_name ENABLE KEYS` вместо `myisamchk -r -q /path/to/db/tbl_name`. Таким образом можно также пропускать шаги `FLUSH TABLES`.

- Можно ускорять операции вставки, выполняемые несколькими операторами, путем установки блокировки таблиц:
- `mysql> LOCK TABLES a WRITE;`
- `mysql> INSERT INTO a VALUES (1,23),(2,34),(4,33);`
- `mysql> INSERT INTO a VALUES (8,26),(6,29);`
- `mysql> UNLOCK TABLES;`

Главный фактор, влияющий на скорость, - то, что буфер индексов сбрасывается на диск только один раз, после завершения всех операторов `INSERT`. Обычно содержимое индексных буферов сбрасывалось бы на диск столько раз, сколько имеется различных операторов `INSERT`. Блокировка не нужна, если можно вставить все строки при помощи одного оператора. Для транзакционных таблиц,

чтобы повысить скорость, следует использовать BEGIN/COMMIT вместо LOCK TABLES. Блокировка также понизит полное время проверки подсоединений (multi-connection tests), но максимальное время ожидания для некоторых потоков повысится (потому что они ожидают снятия блокировки). Например:

поток 1 делает 1000 вставок

потоки 2, 3 и 4 делают 1 вставку

поток 5 делает 1000 вставок

Если блокировка не используется, 2, 3, и 4 завершат выполнение раньше, чем 1 и 5. Если блокировка используется, 2, 3 и 4, видимо, не закончат выполнение раньше, чем 1 или 5, но общее время должно приблизительно уменьшиться на 40%. Так как в MySQL операции INSERT, UPDATE и DELETE очень быстрые, общая производительность будет улучшаться, если добавлять блокировки ко всем командам, делающим более 5 вставок или обновлений подряд. Если делается очень много вставок строк, можно время от времени сопровождать команду LOCK TABLES командой UNLOCK TABLES (после каждых 1000 строк), чтобы позволить другим потокам обращаться к таблице. Результатом всего этого будет получение хорошей производительности. Конечно, для загрузки данных намного более быстрой является команда LOAD DATA INFILE.

Оценка производительности запроса

В большинстве случаев можно оценивать производительность путем подсчета дисковых операций. Для маленьких таблиц можно обычно принимать 1 строку за 1 операцию дискового поиска (поскольку индекс, скорее всего, в кэше). Для больших таблиц можно считать, что (при

использовании индексов типа B++ деревьев) для нахождения строки
потребуется

$$\log(\text{количество_строк}) / \log(\text{длина_индексного_блока} / 3 * 2 / (\text{длина_индекса} + \text{длина_указателя_на_данные})) + 1$$

дисксовая операция для получения строки.

Обычно в MySQL индексный блок занимает 1024 байта, а указатель - 4 байта. Для таблицы, содержащей 500000 строк и имеющей длину индекса 3 (medium integer) потребуется $\log(500,000)/\log(1024/3*2/(3+4)) + 1 = 4$ дисковых операции поиска.

Поскольку вышеупомянутый индекс будет занимать приблизительно $500000 * 7 * 3/2 = 5,2\text{Мб}$ (если учитывать, что индексные буфера обычно заполняются на $2/3$), большая часть индекса, скорее всего, окажется в памяти, и для того, чтобы найти строку, потребуется лишь 1-2 обращения к ОС для чтения.

Для записи, однако, потребуется 4 дисковых запроса (таких, какие рассматривались выше) чтобы найти место для помещения нового индекса, и обычно 2 дисковых операции, чтобы обновить индекс и вставить строку.

Обратите внимание: сказанное выше не означает, что производительность приложения будет ухудшаться в $\log N$ раз! Поскольку все кэшируется в ОС или на SQL-сервере, замедление работы при увеличении таблицы будет незначительным. И лишь после того, как данных станет так много, что они перестанут помещаться в кэш, замедление работы там, где работа приложения сводится только к операциям дискового поиска (количество которых растет в $\log N$), станет гораздо ощутимей. Чтобы избежать этого, следует увеличить индексный кэш так, чтобы он вмещал возросшее количество данных.

Выводы к третьей главе

- В этой главе разработана структура БД;
- Созданы таблицы;
- Созданы тестовые данные;
- Протестирована БД в среде применения.

ГЛАВА 4. БЕЗОПАСНОСТЬ ЖИЗНЕДЕЯТЕЛЬНОСТИ

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

Снижение физических нагрузок в условиях современной жизни, с одной стороны, и недостаточное развитие массовых форм физической культуры среди населения, с другой стороны, приводят к ухудшению различных функций и появлению негативных состояний организма человека.

Для обеспечения нормальной жизнедеятельности организма человека необходима достаточная активность скелетных мышц. Работа мышечного аппарата способствует развитию мозга и установлению межцентральных и межсенсорных взаимосвязей. Двигательная деятельность повышает энергопродукцию и образование тепла, улучшает функционирование дыхательной, сердечно-сосудистой и других систем организма. Недостаточность движений нарушает нормальную работу всех систем и вызывает появление особого состояния – гиподинамии.

Гиподинамия – понижение мышечных усилий, когда движения осуществляются, но при крайне малых нагрузках на мышечный аппарат. В обоих случаях скелетные мышцы нагружены совершенно недостаточно. Возникает огромный дефицит биологической потребности в движениях, что резко снижает функциональное состояние и работоспособность организма. Современному человеку доступны все блага цивилизации: автомобили, магазины на каждом шагу, сидячая работа, интернет. Все это, конечно, хорошо, но проблема в том, что для человеческого организма малоподвижный образ жизни – смерти подобно. Ведь самой природой заложено, что мы должны много и активно двигаться.

Некоторые животные очень тяжело переносят отсутствие движений. Например, цыплята, выращенные в условиях обездвижения в тесных клетках и выпущенные затем на волю, погибали при малейшей пробежке по двору.

Тяжело переносится снижение двигательной активности человеком. Обследование моряков-подводников показало, что после 1,5 месяцев пребывания в море сила мышц туловища и конечностей уменьшалась на 20-40% от исходной, а после 4 месяцев плавания – на 40-50%. Наблюдались и другие нарушения.

Последствия гиподинамии

Еще в древности было замечено, что физическая активность способствует формированию сильного и выносливого человека, а неподвижность ведет к снижению работоспособности, заболеваниям и тучности. Все это происходит вследствие нарушения обмена веществ. Уменьшение энергетического обмена, связанное с изменением интенсивности распада и окисления органических веществ, приводит к нарушению биосинтеза, а также к изменению кальциевого обмена в организме. Вследствие этого в костях происходят глубокие изменения. Прежде всего, они начинают терять кальций. Это приводит к тому, что кость делается рыхлой, менее прочной. Кальций попадает в кровь, оседает на стенках кровеносных сосудов, они склерозируются, т. е. пропитываются кальцием, теряют эластичность и делаются ломкими. Способность крови к свертыванию резко возрастает. Возникает угроза образования кровяных сгустков (тромбов) в сосудах. Содержание большого количества кальция в крови способствует образованию камней в почках.

Отсутствие мышечной нагрузки снижает интенсивность энергетического обмена, что отрицательно сказывается на скелетных и сердечной мышцах. Кроме того, малое количество нервных импульсов, идущих от работающих мышц, снижает тонус нервной системы, утрачиваются приобретенные ранее навыки, не образуются новые. Все это самым отрицательным образом отражается на здоровье. Следует учесть также следующее. Сидячий образ жизни приводит к тому, что хрящ постепенно становится менее эластичным, теряет гибкость. Это может повлечь снижение

амплитуды дыхательных движений и потерю гибкости тела. Но особенно сильно от неподвижности или малой подвижности страдают суставы.

Характер движения в суставе определен его строением. В коленном суставе ногу можно только сгибать и разгибать, а в тазобедренном суставе движения могут совершаться во всех направлениях. Однако амплитуда движений зависит от тренировки. При недостаточной подвижности связки теряют эластичность. В полость сустава при движении выделяется недостаточное количество суставной жидкости, играющей роль смазки. Все это затрудняет работу сустава. Недостаточная нагрузка влияет и на кровообращение в суставе. В результате питание костной ткани нарушается, формирование суставного хряща, покрывающего головку и суставную впадину сочленяющихся костей, да и самой кости идет неправильно, что приводит к различным заболеваниям. Но дело не ограничивается только этим. Нарушение кровообращения может привести к неравномерному росту костной ткани, вследствие чего возникает разрыхление одних участков и уплотнение других. Форма костей в результате этого может стать неправильной, а сустав потерять подвижность.

Гиподинамия - это деятельность человека с ограниченными мышечными усилиями.

В условиях монотонного труда гиподинамия является фактором, способствующим развитию состояния монотонии.

Монотония - функциональное состояние организма человека, определяемое комплексом психологических и физиологических изменений в организме, возникающее при монотонной работе, т. е. ответная реакция организма на монотонный труд.

В эпоху научно-технической революции развитие современного производства качественно изменило профессиональную деятельность человека. На фоне значительного уменьшения доли тяжелого физического труда существенно возросло количество людей, занятых выполнением простых однообразных операций с ограничением мышечных усилий и общей

подвижности, т.е. работающих в условиях монотонности, гипокинезии и гиподинамии. Проблема монотонности является весьма актуальной и для новых видов труда, важными чертами которых являются функции наблюдения управления и контроля в условиях небольшого количества поступающей информации, значительной деятельности наблюдения, ограниченной двигательной активности.

Монотонность труда, особенно в сочетании с гиподинамией, вызывает целый ряд таких неблагоприятных последствий, как снижение работоспособности, рост травматизма, заболеваемости и текучести кадров и др., которые в конечном итоге приводят к значительному снижению эффективности труда в целом. В связи с этим проблема профилактики развития состояния монотонии является актуальной как в медико-биологическом, так и в социально-экономическом аспектах. В то же время ее решение сдерживается отсутствием единых научно-обоснованных положений, необходимых для конкретных рекомендаций по предупреждению отрицательных последствий монотонного труда в разных отраслях промышленности.

Причины возникновения монотонного труда

Монотонный труд (работа) - это свойство некоторых видов труда, требующих от человека длительного выполнения однообразных, элементарных действий или непрерывной и устойчивой концентрации внимания в условиях дефицита сенсорной информации.

Монотония - особый вид функционального состояния организма человека, развивающегося в процессе монотонной работы.

Основными показателями монотонного труда являются:

- структурное однообразие, простота рабочих приемов (операций);
- непродолжительный временной цикл;
- высокая степень повторяемости;
- отсутствие творческих элементов;
- строго определенный регламент действий;

- принудительный ритм и темп;
- дефицит информации о ходе выполнения работы;
- отсутствие интеллектуально-эмоциональной нагрузки;
- легкая или средняя величина физической нагрузки и дефицит движений, что обуславливает гиподинамию и гипокинезию;
- постоянный фоновый шум;
- стробоскопический эффект люминесцентных ламп и др.;
- ограниченное пространство рабочих действий;
- изоляция, разобщенность рабочих.

В формировании восприятия трудового процесса как монотонного определенную роль играют психологические и физиологические особенности человека, его индивидуальная подверженность к развитию состояния монотонии.

Влияние монотонного труда на организм человека

Монотонность труда сопровождается у многих людей рядом неприятных субъективных ощущений.

Субъективные ощущения проявляются в падении интереса к выполняемой работе, скуке, апатии, невнимательности, сонливости, искаженном чувстве времени ("время длится очень долго"), чувстве усталости и т.п., что в конечном счете обуславливает субъективную оценку работы как неинтересной или даже непривлекательной.

Психофизиологические проявления состояния монотонии свидетельствуют о пониженной психофизиологической активности человека и заключаются в следующем:

- снижению уровня бодрствования (изменение альфа-ритма ЭЭГ);
- снижению тонуса симпатического отдела вегетативной нервной системы (урежение частоты пульса, снижение артериального давления, увеличение аритмии пульса и др.);
- снижению тонуса скелетной мускулатуры.

Состояние монотонии характеризуется также ухудшением рабочих действий, замедлением их и увеличением ошибок в работе. Ухудшение основных параметров профессиональной деятельности, а также психофизиологические проявления состояния монотонии свидетельствуют о том, что в этих условиях работоспособность снижается. Для состояния монотонии и, соответственно, его симптоматики характерны волнообразные колебания: периоды понижения работоспособности сменяются периодами повышения ее. В условиях монотонности человек должен время от времени волевым усилием преодолевать состояние пониженной активности. Эти периодические повышения активности сопряжены с затратой как энергетических, так и функциональных ресурсов и способствуют более быстрому развитию утомления и недовольства работой.

Основными последствиями монотонного труда являются:

- снижение работоспособности и производительности труда;
- ухудшение качества продукции;
- производственный травматизм;
- повышенная заболеваемость;
- снижение творческой инициативы работающих;
- высокая текучесть кадров.

Основное отрицательное последствие гипокинезии - детренированность как отдельных систем (мышечной и сердечно-сосудистой), так и организма в целом. В результате детренированности функциональные системы организма (и в первую очередь - сердечно-сосудистая система) становятся менее устойчивыми к отрицательному влиянию нейро-гуморальных воздействий в ситуациях сильного психо-эмоционального напряжения. Это, вероятно, и является одной из причин существенного возрастания за последнее время нервных и сердечно-сосудистых заболеваний.

Основные принципы оптимизации монотонных видов труда

При разработке мер профилактики развития состояния монотонии необходимо учитывать основные психофизиологические явления, возникающие в организме работающих в условиях монотонного труда и во многом определяющие его отрицательные последствия.

Поэтому разрабатываемые мероприятия должны быть направлены на:

- совершенствование технологических процессов с целью уменьшения влияния монотонности труда;
- обеспечение оптимальной информационной и двигательной нагрузок;
- повышение уровня бодрствования, увеличение эмоционального тонуса и мотивации.

Все это достигается как оптимизацией содержания и условий трудовой деятельности, так и непосредственным воздействием на функциональное состояние организма человека комплекса технологических, организационно-технических и психофизиологических мер. Среди них важнейшее значение имеют:

- автоматизация и механизация однообразных ручных работ;
- совершенствование технологии, оптимизация содержания труда;
- совершенствование организации трудовой деятельности;
- совершенствование организации рабочего места;
- оздоровление условий производственной среды;
- применение психологических и социально-психологических факторов профилактики монотонии;
- разработка системы профориентации;
- рациональное использование вне рабочего времени.

Профилактические мероприятия, проводимые в плане вышеуказанных аспектов, снижают утомление и субъективные ощущения монотонности, положительно влияют на работоспособность и производительность труда, дают положительный эффект.

4.2. Эргономика

Усложнение производственных процессов и оборудования изменили функции человека в современном производстве: возросла ответственность решаемых задач; увеличился объем информации, воспринимаемой работающим и быстродействие оборудования. Работа человека стала сложнее, возросла нагрузка на нервную систему и снизилась нагрузка физическая. В ряде случаев человек стал наименее надежным звеном системы «человек-машина». Возникла задача обеспечения надежности и безопасности работы человека на производстве. Эту задачу решает эргономика и инженерная психология.

Эргономика (от греческого *ergon* - работа и *nomos* - закон) - научная дисциплина, изучающая человека в условиях его деятельности, связанной с использованием машин. Цель эргономики - оптимизация условий труда в системе "человек-машина" (СЧМ). Эргономика определяет требования человека к технике и условия ее функционирования. Эргономичность техники является наиболее обобщенным показателем свойств и других показателей техники.

Эргономика - наука о том, как люди с их различными физическими данными и способами функционирования взаимодействуют с окружающей рабочей средой (оборудованием и машинами, которыми они пользуются). Цель эргономики состоит в том, чтобы обеспечить комфорт, эффективность и безопасность при пользовании компьютерами уже на этапе разработки клавиатур, компьютерных плат, рабочей мебели и др. для устранения физического дискомфорта и проблем со здоровьем на рабочем месте. В связи с тем, что все больше людей проводят много времени перед компьютерными мониторами, ученые многих областей, включая анатомию, психологию и охрану окружающей среды, вовлекаются в изучение правильных, с точки зрения эргономики, условий работы.

Большинство изготовителей мебели не учитывают индивидуальные особенности человеческого тела при проектировании автоматизированных рабочих мест. Обустройство эргономически оборудованного места может потребовать дополнительных расходов. Если позволяет Ваш бюджет, покупайте эргономически разработанную мебель типа стульев, полок и столов, которая может быть приспособлена к Вашим индивидуальным физическим данным.

Инженерная психология - научная дисциплина, изучающая закономерности информационного взаимодействия человека и техники для проектирования, создания и эксплуатации системы "человек-машина". Инженерная психология исследует процессы приема, хранения, переработки и реализации информации человеком. На основании закономерностей психических, психофизиологических процессов и свойств человека она определяет требования к техническим устройствам и построению системы "человек-машина", а также требования к свойствам человека-оператора.

В качестве обобщенных показателей деятельности оператора и системы "человек-машина" инженерная психология использует эффективность, надежность, точность, быстродействие.

Научную основу эргономики составляют анатомия, физиология и психология. Анатомия составляет теоретическую основу антропометрии и биомеханики.

Антропометрия - измерение человека позволяет получить данные, необходимые для правильного расположения органов управления и определения размеров рабочих пространств. Важным моментом при этом является определение границ колебаний размеров, в которых учитывается потребный объем выборки, выражаемый в перцентилях. Так, 90-й перцентиль представляет результаты измерений, показывающих, что 90% измеряемой группы имеют определенные размеры меньше, а 10% больше средних для данной группы. На практике любая конструкция рассчитывается на 90% населения.

Биомеханика - изучает приложение сил телом человека. Она дает рекомендации, как необходимо эффективно прилагать силы: усилие должно создаваться массой тела, а не мышц; наиболее полно должны использоваться мышцы, передвигающие сустав вокруг его центрального участка.

Физиология в эргономике дает закономерности процесса производства энергии организмом человека. Вырабатываемая энергия организма оценивается по потреблению кислорода. Психология вносит в эргономику теорию деятельности человека, основанную на информационной модели человека-оператора; теорию обучения и теорию организации, связанную с проектированием работы.

Задачами эргономики как прикладной дисциплины являются:

- проектирование системы "человек-машина", то есть распределение функций между человеком и машиной;
- проектирование рабочего пространства так, чтобы физическое окружение соответствовало характеристикам человека;
- проектирование окружающей среды в соответствии с требованиями оператора;
- проектирование рабочих ситуаций (продолжительность рабочего дня, перерывы для отдыха и т.п.).

Инженерная психология, как это следует из вышеизложенного, является практически составной частью эргономики, решающая задачи организации системы "человек-машина" путем:

- распределения функций между человеком и машиной;
- анализа функций, выполняемых человеком в системе "человек-машина";
- проектирования системы информации, выбора чувствительного канала;
- конструирования средств управления;
- проектирования рабочих мест;

- обеспечение удобства технического обслуживания машин;
- подбора кадров и их профессиональной подготовки.

Учет, эргономических требований должен осуществляться на всех этапах проектных решений и включает:

- Разработку профессиограммы, определяющей цели и задачи трудовой деятельности, ее психофизиологические характеристики, требования к человеку и технике.
- Анализ и уточнение назначения, принципов действия и конструкции техники, ее характеристик применительно к целям трудовой деятельности.
- Распределение функций между человеком и техникой на основе оценки качества выполнения задач человеком и машиной и общей эффективности системы.
- Установление последовательности выполняемых человеком операций и определение объема и формы представления информации.
- Ориентационную оценку надежностных, временных и точностных требований к деятельности человека.

На основании рассмотренных работ определяется: состав специалистов, их функции и организация работы; состав средств отображения информации, органов управления рабочих мест и пультов управления; компоновка средств отображения информации и органов управления, размещение рабочих мест в производственных помещениях.

Связь человека с окружающей средой и параметрами рабочего места.

Рабочее место - это зона, в которой совершается трудовая деятельность исполнителя или группы исполнителей. Рабочие места могут быть индивидуальными и коллективными, универсальными, специализированными и специальными.

Общие требования, которые должны соблюдаться при проектировании рабочих мест, следующие:

- достаточное рабочее пространство для человека;
- оптимальное положение тела работающего;
- достаточные физические, зрительные и слуховые связи между человеком и машиной;
- оптимальное размещение рабочего места в помещении;
- допустимый уровень действия факторов производственных условий; оптимальное размещение информационного и моторного поля;
- наличие средств защиты от производственных опасностей.

Конструирование должно обеспечивать зоны оптимальной и легкой досягаемости моторного поля рабочего места и оптимальную зону информационного поля рабочего места. Угол обзора по отношению к горизонтали должен составлять 30-40°.

Выбор рабочего положения должен учитывать усилия, затрачиваемые человеком, размах движений, необходимость перемещений, темп операций. Выбор рабочей позы должен учитывать физиологию человека, а параметры рабочего места определяются выбором положения тела при работе (сидя, стоя, переменнo).

Рабочие места для выполнения работ «сидя» организуются при легкой работе и средней тяжести, а при тяжелой - рабочая поза - "стоя".

В конструкции оборудования и организации рабочего места необходимо предусматривать возможности регулирования отдельных элементов, чтобы обеспечить оптимальное положение работающего.

Проектирование оборудования должно обеспечить его соответствие антропометрическим и биомеханическим характеристикам человека на основе учета динамики изменения размеров тела при его перемещении, диапазона движений в суставах.

Для учета в конструкции оборудования антропометрических данных необходимо:

- определить контингент людей, для которых предназначено оборудование;
- выбрать группу антропометрических признаков;
- установить процент работающих, которому должно удовлетворять оборудование;
- определить границы интервала размеров (усилий), которые должны быть реализованы в оборудовании.

При проектировании используют антропометрические размеры тела, причем учитываются различия в размерах тела мужчин и женщин, национальные, возрастные, профессиональные. Для определения границ интервалов, в которых учитывается процент населения, используется система перцентелей. Конструкция оборудования должна обеспечивать возможность использования по меньшей мере для 90% потребителей.

Для работы в положении "сидя" используются различные рабочие сиденья. Различают рабочие сиденья для длительного и кратковременного пользования. Общие требования для сидений длительного пользования следующие: сидение должно обеспечивать позу, уменьшающую статистическую работу мышц; создавать условия для возможности изменения рабочей позы; не затруднять деятельность систем организма; обеспечивать свободное перемещение относительно рабочей поверхности, иметь регулируемые параметры; иметь полумягкую обивку. Для кратковременного пользования рекомендуются жесткие стулья и различного типа табуреты.

В условиях растущей механизации и автоматизации производственных процессов особое значение приобретают средства отображения информации об объекте управления. Широкое использование получила информационная модель, то есть организованная по определенным правилам информация о

состоянии объекта управления. К информационным моделям предъявляются следующие требования:

- содержание информационной модели должно адекватно отображать объект управления;
- информационная модель должна обеспечивать оптимальный информационный баланс;
- форма и композиция информационной модели должна соответствовать задачам трудового процесса и возможностям человека по приему информации.

Практика позволяет наметить последовательность разработки информационной модели: определение задач системы, очередность их решения и источников информации; составление перечня объектов управления и их признаков; распределение объектов по степени важности; распределение функции между автоматикой и человеком; выбор системы кодирования объектов и составление общей композиции модели; определение исполнительных действий человека.

В процессе конструирования информационной модели определяются места размещения средств информации на рабочем месте, выбираются размеры знаков и компоновка. Средства отображения размещаются в поле зрения наблюдателя с учетом оптимальных углов и зон наблюдения. Размеры знаков наблюдения определяются с учетом максимальной точности и скорости восприятия информации, а также яркости знаков, величины контраста, использования цвета. Оптимальной яркостью считаются значения, при которых обеспечивается максимальная контрастная чувствительность. Величина ее будет тем больше, чем меньше размер объекта различения. Оптимальная зона величины контраста равна 60-90%.

В работе глаза имеет место определенная инерционность, что требует учета времени экспозиции зрительного сигнала и временных интервалов для ощущения раздельности сигналов следующих один за другим. В

большинстве случаев время экспозиции сигнала должно быть не менее 50 мс. Каждая разновидность индикаторов имеет свою область использования: индикаторы с подсветкой применяются для отображения качественной информации, требующей немедленной реакции оператора; стрелочные индикаторы используются для чтения измеряемых параметров; интегральные индикаторы для совмещения информации сразу о нескольких параметрах.

Структуру и динамику управляемого объекта обычно представляют с помощью микросхемы. В ряде случаев используется табло для отображения информации и восприятия ее коллективом операторов.

При проектировании рабочего места должны учитываться правила экономики движений: при работе двумя руками движения их должны быть одновременными и симметричными; движения должны быть плавными и закругленными, ритмичными и привычными для работающего. Конструкция оборудования должна учитывать правила, касающиеся скорости и точности рабочих движений. Например, наиболее быстрое движение к себе; в горизонтальной плоскости скорость рук больше, чем вертикальной; точность движений лучше в положении сидя, чем стоя и т.д. Органы управления, используемые на рабочем месте, должны соответствовать общим требованиям эргономики: направление движения органов управления должно соответствовать движению связанного с ним индикатора; соответствие расположения органов управления последовательности работы оператора; удобство использования; создание в органах управления механического сопротивления и т.п. Помимо этого, к каждому виду органов управления соответствует своя область использования и особые требования к размерам, форме, усилию и т.п.

На автоматизированном рабочем месте оператора-связиста (оператор в диспетчерской) в общем случае используются:

- средства отображения информации индивидуального пользования (блоки отображения, устройства сигнализации и так далее);

- средства управления и ввода информации (пульт дисплея, клавиатура управления, отдельные органы управления и так далее);
- устройства связи и передачи информации (модемы, телеграфные и телефонные аппараты):
- устройства документирования и хранения информации (устройства печати, магнитной записи и так далее);
- вспомогательное оборудование (средства оргтехники, хранилища для носителей информации, устройства местного освещения).

На автоматизированном рабочем месте должна быть обеспечена информационная и конструктивная совместимость используемых технических средств, антропометрических и психофизиологических характеристик человека.

При организации рабочего места должны быть учтены не только факторы, отражающие опыт, уровень профессиональной подготовки, индивидуально-личностные свойства операторов-связистов, но и факторы, характеризующие соответствие форм, способов представления и ввода информации психофизиологическим возможностям человека.

При оптимизации процедур взаимодействия операторов-связистов с техническими средствами в условиях автоматизации эргономические факторы выступают в качестве основных, обуславливающих вероятностно-временные характеристики и напряженность работы. Эти факторы являются чувствительными к вариациям индивидуально-личностных свойств оператора.

Рабочая мебель должна быть удобной для выполнения планируемых рабочих операций. Конструкция рабочей мебели: стола, стула имеет огромное значение для создания здоровых условий и высокопроизводительного труда. Рабочая мебель конструируется с учетом антропометрических данных человека, технических, эстетических и экономических факторов.

В комплекте рабочей мебели большое значение имеет конструкция производственного стула, так как от него зависит поза работника, а следовательно, и затрата энергии и степень его утомляемости. Рабочее сиденье должно иметь требуемые размеры, соответствующие антропометрическим данным человека и быть подвижным. Наиболее удобны стулья и кресла с регулируемым наклоном спинки и высотой сиденья. Изменяя высоту сиденья от уровня пола и угол наклона спинки, можно найти положение, наиболее соответствующее трудовому процессу и индивидуальным особенностям работника.

Как правило, все поверхности письменных и рабочих столов должны быть на уровне локтя при рабочем положении человека. При выборе высоты стола необходимо учитывать сидит человек во время работы или стоит.

Неудобная высота стола снижает эффективность работы и вызывает быстрое утомление. Отсутствие достаточного пространства для коленей и ступней вызывает постоянное раздражение работника. Минимальная рабочая высота стола должна быть не менее 725 мм. Как показывает практика, для рабочего среднего роста высота рабочего стола принимается 800 мм. Для работника другого роста можно изменить высоту рабочего стула или положение его подножки так, чтобы расстояние от предмета обработки до глаз рабочего по высоте было равным примерно 450 мм.

Размещение технических средств и кресла оператора в рабочей зоне должно обеспечивать удобный доступ к основным функциональным узлам и блокам аппаратуры для проведения технической диагностики, профилактического осмотра и ремонта; возможность быстро занимать и покидать рабочую зону; исключение случайного приведения в действие средств управления и ввода информации; удобную рабочую позу и позу отдыха. Кроме того, схема размещения должна удовлетворять

требованиям целостности, компактности и технико-эстетической выразительности рабочей позы.

Дисплей должен размещаться на столе или подставке так, чтобы расстояние наблюдения на экране не превышало 700 мм (оптимальное расстояние 450 – 500 мм). Экран дисплея по высоте должен быть расположен так, чтобы угол между центром экрана и горизонтальной линией взгляда составлял 20°. В горизонтальной плоскости угол наблюдения экрана не должен превышать 60°. Пульт дисплея должен быть размещен на столе или подставке так, чтобы высота клавиатуры пульта по отношению к полу составляла 650 - 720 мм. При размещении пульта на стандартном столе высотой 750 мм необходимо использовать кресло с регулируемой высотой сиденья (450 - 380 мм) и подставку для ног.

Документ (бланк) для ввода оператором данных рекомендуется располагать на расстоянии 450 - 500 мм от глаза оператора, преимущественно слева, при этом угол между экраном дисплея и документом в горизонтальной плоскости должен составлять 30-40°. Угол наклона клавиатуры должен быть равен 15°.

Экран дисплея, документы и клавиатура пульта дисплея должны быть расположены так, чтобы перепад яркостей поверхностей, зависящий от их расположения относительно источника света, не превышал 1 : 10 (рекомендуемое значение 1 : 3). При номинальных значениях яркостей изображения на экране 50 - 100 кд/м² освещенность документа должна составлять 300 - 500 лк.

Рабочее место следует оборудовать таким образом, чтобы движения работника были бы наиболее рациональные, наименее утомительные.

Устройства документирования и другие, нечасто используемые технические средства, рекомендуется располагать справа от оператора в зоне максимальной досягаемости, а средства связи слева, чтобы освободить правую руку для записей.

Выводы к четвертой главе:

Эргономика (от греч. *ergon* - работа и *nomos* - закон) - область научноприкладных исследований, находящихся на стыке технических наук, психологии и физиологии труда, в которой разрабатываются проблемы проектирования, оценки и модернизации систем «человек-техника-среда».

В отличие от инженерной психологии и психологии труда эргономика изучает взаимодействие человека и техники не только в сфере производства, но и в сферах досуга и быта.

Одно из важнейших направлений в эргономике, например, связано с проектированием предметного мира для людей с физическими недостатками. Наиболее интенсивно эргономика развивалась в рамках военно-промышленного комплекса.

Активное воздействие на развитие эргономики оказали возросшая конкуренция в производстве товаров и услуг и принятие законодательства, защищающего права потребителя.

Юридическая ответственность за качество и безопасность своих изделий побудила предпринимателей привлекать эргономистов к разработке новых изделий еще на стадии проектирования.

Одно из основных направлений эргономики связано с изменением менталитета инженеров, проектировщиков, конструкторов, которых призывают ориентироваться не на собственные представления о физических и психических характеристиках человека, а на точное знание того, как человек видит, слышит, думает, как далеко он может дотянуться рукой, насколько он может согнуться, как повлияет определенный уровень вибрации на его зрительное восприятие, эмоциональные реакции и т.д.

С этой целью создаются многостраничные справочники по конкретным характеристикам человека и его возможностям действовать в тех или иных средах, проводится большая работа по стандартизации эргономических норм и требований, вводятся в инженерные вузы дополнительные курсы по

эргономике и человеческим факторам.

В настоящее время в мировой практике эргономистов готовят на базе высшего технического образования и в рамках высшего психологического образования. Так, в технических университетах, институтах введена специальность – «эргономика».

На факультетах психологии классических университетов специализацию в области эргономики обеспечивают кафедры психологии труда, инженерной психологии, психофизиологии.

ЗАКЛЮЧЕНИЕ

В ходе выполнения ВКР работы были выполнены следующие задачи:

– Установлена система регистрации/авторизации пользователей;

Введены:

- Система восстановления паролей;
 - Индикатор система присутствия на сайте – пользователи находящиеся на сайте, отмечаются статусом «online»;
 - Система пользовательских профилей;
- Разработана система друзей – возможность установки связи между зарегистрированными пользователями и система личных сообщений – внутренняя почта для зарегистрированных пользователей с уведомлением на email;
- Подключены ленты активности — позволяет пользователям, находящимся в связи отслеживать события друг друга;
- Разработан форум, который позволяет пользователям создавать обсуждения по интересам;
- Построена система фотоальбомов – кроме своих фотографий, пользователи обладают возможностью создания альбома для отслеживания своих результатов и успехов;
- Разработана база для системы комментариев;
- Создана база системы обмена – пользователи могут делиться фотографиями, документами и т.п.

Разработанная база данных отвечает условиям целостности и непротиворечивости запросов. После завершения процесса проектирования, разработанная база была протестирована в действии, которая показала стабильное функционирование в самой системе.

СПИСОК ЛИТЕРАТУРЫ

1. Постановление от 21.03.2012 г. № ПП-1730 «О мерах по дальнейшему внедрению и развитию современных информационно-коммуникационных технологий»
2. Постановление от 01.02.2012г. № 24 «О мерах по созданию условий для дальнейшего развития компьютеризации и информационно-коммуникационных технологий на местах»
3. Владимирова Л.П. Организация профессиональной деятельности учителей в сети / Материалы XVI Международной конференции «Применение новых технологий в образовании». Троицк, 2005. – 189 с.
4. Charles Bell Expert MySQL; Alma Classics - Москва, 2012. - 550 с.
5. Paul DuBois MySQL; Addison-Wesley Professional - Москва, 2008. - 506 с.
6. Philip J. Pratt, Mary Z. Last A Guide to MySQL; БХВ-Петербург, Арлит - Москва, 2012. - 304 с.
7. Robert Sheldon Beginning MySQL®; М.:Гарант - Москва, 2005. - 864 с.
8. How To Do Everything With Php And Mysql; Московский педагогический государственный университет - Москва, 2011. - 400 с.
9. Дронов Владимир PHP 5/6, MySQL 5/6 и Dreamweaver CS4. Разработка интерактивных Web-сайтов; БХВ-Петербург - Москва, 2009. - 544 с.
10. Дюбуа, Поль MySQL; М.: Вильямс; Издание 2-е - Москва, 2010. - 185 с.
11. Кузнецов Максим , Симдянов Игорь MySQL 5; БХВ-Петербург - Москва, 2010. - 502 с.
12. Симдянов, И.В.; Кузнецов, М.В. MySQL 5; БХВ-Петербург - Москва, 2006. - 884 с.
13. Ульман, Ларри MySQL; М.: ДМК Пресс - Москва, 2009. - 352 с.
14. Аткинсон, Леон MySQL. Библиотека профессионала; М.: Вильямс, 2008. - 624 с.

15. Грофф, Джеймс; Вайнберг, Пол SQL: полное руководство; Киев: BHV, 2005. - 608 с.
16. Нанда, А. и др. Oracle PL/SQL для администраторов баз данных; Символ, 2008. - 496 с.
16. Стоунз, Ричард; Мэттью, Нейл PostgreSQL. Основы; СПб: Символ-Плюс, 2007. - 640 с.
17. Фейерштейн, С.; Прибыл, Б. Oracle PL/SQL для профессионалов; СПб: Питер, 2005. - 941 с.
18. Шнайдер, Роберт Microsoft SQL Server 6.5. Проектирование высокопроизводительных баз данных; М.: Лори, 2010. - 361 с.
19. Яргер, Р.Дж.; Риз, Дж.; Кинг, Т. MySQL и mSQL: Базы данных для небольших предприятий и Интернета; СПб: Символ-Плюс, 2013. - 560 с.

Приложение

```
-- База данных: `usit`

-- Структура таблицы `usit_actions`

CREATE TABLE IF NOT EXISTS `usit_actions` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `component` varchar(20) NOT NULL,
  `name` varchar(20) NOT NULL,
  `title` varchar(100) NOT NULL,
  `message` varchar(255) NOT NULL,
  `is_tracked` tinyint(1) NOT NULL,
  `is_visible` tinyint(1) NOT NULL,
  PRIMARY KEY (`id`),
  KEY `name` (`name`,`is_visible`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=31 ;

-- Структура таблицы `usit_actions_log`

CREATE TABLE IF NOT EXISTS `usit_actions_log` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `action_id` int(11) NOT NULL,
  `pubdate` datetime NOT NULL,
  `user_id` int(11) NOT NULL,
  `object` varchar(100) NOT NULL,
  `object_url` varchar(255) NOT NULL,
  `object_id` int(11) NOT NULL,
  `target` varchar(100) NOT NULL,
  `target_url` varchar(255) NOT NULL,
  `target_id` int(11) NOT NULL,
  `description` varchar(1000) DEFAULT NULL,
  `is_friends_only` tinyint(1) NOT NULL DEFAULT '0',
  `is_users_only` tinyint(1) NOT NULL DEFAULT '0',
  PRIMARY KEY (`id`),
```

```

KEY `action_id` (`action_id`,`user_id`),

KEY `object_id` (`object_id`),

KEY `target_id` (`target_id`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=160 ;

-- Структуратаблицы `usit_banlist`

CREATE TABLE IF NOT EXISTS `usit_banlist` (

  `id` int(11) NOT NULL AUTO_INCREMENT,

  `user_id` int(11) NOT NULL,

  `ip` varchar(15) NOT NULL,

  `bandate` datetime NOT NULL,

  `int_num` int(11) NOT NULL,

  `int_period` varchar(20) NOT NULL,

  `status` tinyint(1) NOT NULL DEFAULT '1',

  `autodelete` tinyint(1) NOT NULL DEFAULT '0',

  PRIMARY KEY (`id`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=2 ;

-- Структуратаблицы `usit_banners`

CREATE TABLE IF NOT EXISTS `usit_banners` (

  `id` int(11) NOT NULL AUTO_INCREMENT,

  `position` varchar(100) NOT NULL DEFAULT 'banner_top',

  `typeimg` varchar(10) NOT NULL DEFAULT 'image',

  `fileurl` varchar(250) DEFAULT NULL,

  `hits` int(11) NOT NULL,

  `clicks` int(11) NOT NULL,

  `maxhits` int(11) NOT NULL,

  `maxuser` int(11) NOT NULL,

  `user_id` int(11) NOT NULL DEFAULT '1',

  `pubdate` datetime DEFAULT NULL,

  `title` varchar(250) DEFAULT NULL,

  `link` varchar(250) DEFAULT NULL,

```

```

`published` tinyint(1) NOT NULL DEFAULT '1',

PRIMARY KEY (`id`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=5 ;

-- Структура таблицы `usit_banner_hits`

CREATE TABLE IF NOT EXISTS `usit_banner_hits` (

`id` int(11) NOT NULL AUTO_INCREMENT,

`banner_id` int(11) DEFAULT NULL,

`ip` varchar(16) DEFAULT NULL,

`pubdate` datetime DEFAULT NULL,

PRIMARY KEY (`id`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=1 ;

-- Структура таблицы `usit_blogs`

CREATE TABLE IF NOT EXISTS `usit_blogs` (

`id` int(11) NOT NULL AUTO_INCREMENT,

`user_id` int(11) NOT NULL,

`title` varchar(250) NOT NULL,

`pubdate` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,

`allow_who` varchar(15) NOT NULL,

`view_type` varchar(15) NOT NULL DEFAULT 'list',

`showcats` tinyint(1) NOT NULL DEFAULT '1',

`ownertype` varchar(15) NOT NULL DEFAULT 'single',

`premod` tinyint(1) NOT NULL DEFAULT '0',

`forall` tinyint(1) NOT NULL DEFAULT '0',

`owner` varchar(10) NOT NULL DEFAULT 'user',

`seolink` varchar(255) NOT NULL,

`rating` int(11) NOT NULL DEFAULT '0',

`comments_count` int(11) NOT NULL DEFAULT '0',

PRIMARY KEY (`id`),

KEY `seolink` (`seolink`),

KEY `user_id` (`user_id`),

```

```

KEY `pubdate` (`pubdate`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=42 ;

-- Структура таблицы `usit_blog_authors`
CREATE TABLE IF NOT EXISTS `usit_blog_authors` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `user_id` int(11) NOT NULL,
  `blog_id` int(11) NOT NULL,
  `description` varchar(100) NOT NULL,
  `startdate` datetime NOT NULL,
  PRIMARY KEY (`id`),
  KEY `user_id` (`user_id`),
  KEY `blog_id` (`blog_id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=12 ;

-- Структура таблицы `usit_blog_cats`
CREATE TABLE IF NOT EXISTS `usit_blog_cats` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `blog_id` int(11) NOT NULL,
  `title` varchar(250) NOT NULL,
  `description` varchar(1000) NOT NULL,
  PRIMARY KEY (`id`),
  KEY `blog_id` (`blog_id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=14 ;

-- Структура таблицы `usit_blog_posts`
CREATE TABLE IF NOT EXISTS `usit_blog_posts` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `user_id` int(11) NOT NULL,
  `cat_id` int(11) NOT NULL,
  `blog_id` int(11) NOT NULL,
  `pubdate` datetime NOT NULL,
  `title` varchar(250) NOT NULL,

```

```

`feel` varchar(100) NOT NULL,

`music` varchar(100) NOT NULL,

`content` text NOT NULL,

`content_html` text NOT NULL,

`allow_who` varchar(20) NOT NULL,

`edit_times` int(11) NOT NULL DEFAULT '0',

`edit_date` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,

`published` tinyint(1) NOT NULL DEFAULT '1',

`seolink` varchar(255) NOT NULL,

`comments` tinyint(1) NOT NULL DEFAULT '1',

`comments_count` int(11) NOT NULL DEFAULT '0',

`rating` int(11) NOT NULL DEFAULT '0',

PRIMARY KEY (`id`),

KEY `seolink` (`seolink`),

KEY `blog_id` (`blog_id`),

KEY `user_id` (`user_id`),

FULLTEXT KEY `title` (`title`),

FULLTEXT KEY `content_html` (`content_html`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=41 ;

```

```

-- Структура таблицы `usit_board_cats`

CREATE TABLE IF NOT EXISTS `usit_board_cats` (

  `id` int(11) NOT NULL AUTO_INCREMENT,

  `parent_id` int(11) NOT NULL,

  `ordering` int(11) NOT NULL DEFAULT '1',

  `NSLeft` int(11) NOT NULL,

  `NSRight` int(11) NOT NULL,

  `NSDiffer` varchar(15) NOT NULL,

  `NSIgnore` int(11) NOT NULL,

  `NSLevel` int(11) NOT NULL,

```

```

`title` varchar(100) NOT NULL,
`description` varchar(300) NOT NULL,
`published` tinyint(1) NOT NULL,
`orderform` tinyint(1) DEFAULT '1',
`showdate` tinyint(1) NOT NULL DEFAULT '1',
`pubdate` datetime NOT NULL,
`orderby` varchar(20) NOT NULL DEFAULT 'title',
`orderto` varchar(4) NOT NULL DEFAULT 'asc',
`public` tinyint(1) NOT NULL,
`perpage` int(11) NOT NULL DEFAULT '15',
`maxcols` int(11) NOT NULL DEFAULT '1',
`thumb1` int(11) NOT NULL DEFAULT '64',
`thumb2` int(11) NOT NULL DEFAULT '400',
`thumbsqr` int(11) NOT NULL,
`uplimit` int(11) NOT NULL DEFAULT '10',
`is_photos` tinyint(1) NOT NULL DEFAULT '1',
`icon` varchar(200) DEFAULT 'folder_grey.png',
`obtypes` text NOT NULL,
`form_id` int(11) NOT NULL,
PRIMARY KEY (`id`),
KEY `parent_id` (`parent_id`),
KEY `NSLeft` (`NSLeft`,`NSRight`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=11 ;
-- Структуратаблицы `usit_board_items`
CREATE TABLE IF NOT EXISTS `usit_board_items` (
`id` int(11) NOT NULL AUTO_INCREMENT,
`category_id` int(11) NOT NULL,
`user_id` int(11) NOT NULL,
`obtype` varchar(50) NOT NULL,
`title` varchar(250) NOT NULL,

```

```

`content` text NOT NULL,
`formdata` text NOT NULL,
`city` varchar(100) NOT NULL,
`pubdate` datetime DEFAULT NULL,
`pubdays` int(11) NOT NULL,
`published` tinyint(1) NOT NULL,
`file` varchar(250) NOT NULL,
`hits` int(11) NOT NULL,
`is_vip` tinyint(1) NOT NULL DEFAULT '0',
`vipdate` datetime NOT NULL,
`ip` int(10) unsigned NOT NULL,
PRIMARY KEY (`id`),
KEY `category_id` (`category_id`),
KEY `user_id` (`user_id`),
KEY `obtype` (`obtype`),
KEY `city` (`city`),
KEY `ip` (`ip`),
FULLTEXT KEY `title` (`title`),
FULLTEXT KEY `content` (`content`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=7 ;
-- Структуратаблицы `usit_cache`
CREATE TABLE IF NOT EXISTS `usit_cache` (
`id` int(11) NOT NULL AUTO_INCREMENT,
`target` varchar(10) NOT NULL,
`target_id` varchar(255) NOT NULL,
`cachedate` datetime NOT NULL,
`cachefile` varchar(80) NOT NULL,
PRIMARY KEY (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=1 ;

```

```

-----

--
-- Структура таблицы `usit_category`
CREATE TABLE IF NOT EXISTS `usit_category` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `parent_id` int(11) DEFAULT NULL,
  `title` varchar(200) NOT NULL,
  `description` text NOT NULL,
  `published` tinyint(1) NOT NULL,
  `showdate` tinyint(1) NOT NULL DEFAULT '1',
  `showcomm` tinyint(1) NOT NULL DEFAULT '1',
  `orderby` varchar(30) NOT NULL DEFAULT 'date',
  `orderto` varchar(4) NOT NULL DEFAULT 'asc',
  `modgrp_id` int(11) NOT NULL,
  `NSLeft` int(11) NOT NULL,
  `NSRight` int(11) NOT NULL,
  `NSLevel` int(11) NOT NULL,
  `NSDiffer` varchar(11) NOT NULL,
  `NSIgnore` int(11) NOT NULL,
  `ordering` int(11) NOT NULL,
  `maxcols` int(11) NOT NULL DEFAULT '1',
  `showtags` tinyint(1) NOT NULL DEFAULT '1',
  `showrss` tinyint(1) NOT NULL DEFAULT '1',
  `showdesc` tinyint(1) NOT NULL,
  `is_public` tinyint(1) NOT NULL,
  `photoalbum` text NOT NULL,
  `seolink` varchar(200) NOT NULL,
  `url` varchar(100) NOT NULL,
  `tpl` varchar(50) NOT NULL DEFAULT 'com_content_view.tpl',

```

```

`cost` varchar(5) NOT NULL,

PRIMARY KEY (`id`),

KEY `seolink` (`seolink`),

KEY `parent_id` (`parent_id`),

KEY `NSLeft` (`NSLeft`,`NSRight`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=62 ;

```

```
-- Структура таблицы `usit_clubs`
```

```

CREATE TABLE IF NOT EXISTS `usit_clubs` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `admin_id` int(11) NOT NULL DEFAULT '1',
  `title` varchar(200) NOT NULL,
  `description` longtext NOT NULL,
  `imageurl` varchar(100) NOT NULL,
  `pubdate` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
  `clubtype` varchar(15) NOT NULL DEFAULT 'public',
  `published` tinyint(1) NOT NULL DEFAULT '1',
  `maxsize` int(11) NOT NULL,
  `enabled_blogs` tinyint(1) DEFAULT '-1',
  `enabled_photos` tinyint(1) DEFAULT '-1',
  `rating` int(11) NOT NULL,
  `members_count` int(11) NOT NULL DEFAULT '1',
  `photo_premod` tinyint(1) NOT NULL,
  `blog_premod` tinyint(1) NOT NULL,
  `blog_min_karma` int(11) NOT NULL,
  `photo_min_karma` int(11) NOT NULL,
  `album_min_karma` int(11) NOT NULL DEFAULT '25',
  `join_min_karma` int(11) NOT NULL,
  `join_karma_limit` int(11) NOT NULL,
  `create_karma` int(11) NOT NULL,

```

```

`is_vip` tinyint(1) NOT NULL DEFAULT '0',

`join_cost` float NOT NULL,

PRIMARY KEY (`id`),

KEY `pubdate` (`pubdate`),

KEY `admin_id` (`admin_id`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=26 ;

-- Структуратаблицы `usit_comments`

CREATE TABLE IF NOT EXISTS `usit_comments` (

`id` int(11) NOT NULL AUTO_INCREMENT,

`parent_id` int(11) NOT NULL,

`pid` int(11) NOT NULL,

`user_id` int(11) NOT NULL,

`target` varchar(20) NOT NULL,

`target_id` int(11) NOT NULL,

`guestname` varchar(200) NOT NULL,

`content` text NOT NULL,

`content_bbcode` text NOT NULL,

`pubdate` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,

`published` tinyint(1) NOT NULL DEFAULT '1',

`is_new` tinyint(1) NOT NULL DEFAULT '1',

`target_title` varchar(150) NOT NULL,

`target_link` varchar(200) NOT NULL,

`ip` varchar(15) NOT NULL,

`is_hidden` tinyint(1) NOT NULL DEFAULT '0',

`rating` int(11) NOT NULL DEFAULT '0',

PRIMARY KEY (`id`),

KEY `user_id` (`user_id`),

KEY `target_id` (`target_id`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=24 ;

-- Структуратаблицы `usit_comment_targets`

```

```

CREATE TABLE IF NOT EXISTS `usit_comment_targets` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `target` varchar(32) NOT NULL,
  `component` varchar(32) NOT NULL,
  `title` varchar(100) NOT NULL,
  `target_table` varchar(32) NOT NULL,
  `subj` varchar(100) NOT NULL,
  PRIMARY KEY (`id`),
  UNIQUE KEY `target` (`target`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=11 ;

```

-- Структура таблицы `usit_components`

```

CREATE TABLE IF NOT EXISTS `usit_components` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `title` varchar(200) NOT NULL,
  `link` varchar(200) NOT NULL,
  `config` text NOT NULL,
  `internal` int(11) NOT NULL,
  `author` varchar(200) NOT NULL DEFAULT 'InstantCMS team',
  `published` tinyint(1) NOT NULL DEFAULT '1',
  `version` varchar(6) NOT NULL DEFAULT '1.10',
  `system` int(11) NOT NULL DEFAULT '0',
  PRIMARY KEY (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=29 ;

```

-- Структура таблицы `usit_content`

```

CREATE TABLE IF NOT EXISTS `usit_content` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `category_id` int(11) NOT NULL,
  `user_id` int(11) NOT NULL DEFAULT '1',
  `pubdate` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',
  `enddate` date NOT NULL,

```

```

`is_end` tinyint(1) NOT NULL,
`title` varchar(200) NOT NULL,
`description` text NOT NULL,
`content` longtext,
`published` tinyint(1) NOT NULL DEFAULT '1',
`hits` int(11) NOT NULL DEFAULT '0',
`rating` int(11) NOT NULL DEFAULT '0',
`meta_desc` text NOT NULL,
`meta_keys` text NOT NULL,
`showtitle` tinyint(1) NOT NULL DEFAULT '1',
`showdate` tinyint(1) NOT NULL DEFAULT '1',
`showlatest` tinyint(1) NOT NULL DEFAULT '1',
`showpath` tinyint(1) NOT NULL DEFAULT '1',
`ordering` int(11) NOT NULL,
`comments` tinyint(1) NOT NULL DEFAULT '1',
`is_arhive` tinyint(1) NOT NULL,
`seolink` varchar(200) NOT NULL,
`canrate` tinyint(1) NOT NULL DEFAULT '1',
`pagetitle` varchar(255) NOT NULL,
`url` varchar(100) NOT NULL,
`tpl` varchar(50) NOT NULL DEFAULT 'com_content_read.tpl',
PRIMARY KEY (`id`),
KEY `seolink` (`seolink`),
KEY `category_id` (`category_id`),
KEY `user_id` (`user_id`),
FULLTEXT KEY `title` (`title`),
FULLTEXT KEY `content` (`content`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=65 ;
-- Структуратаблицы `usit_content_access`
CREATE TABLE IF NOT EXISTS `usit_content_access` (

```

```

`id` int(11) NOT NULL AUTO_INCREMENT,

`content_id` int(11) NOT NULL,

`content_type` varchar(100) NOT NULL,

`group_id` int(11) NOT NULL,

PRIMARY KEY (`id`),

KEY `content_id` (`content_id`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=1 ;

-- Структура таблицы `usit_cron_jobs`

CREATE TABLE IF NOT EXISTS `usit_cron_jobs` (

`id` int(11) NOT NULL AUTO_INCREMENT,

`job_name` varchar(50) NOT NULL,

`job_interval` smallint(6) NOT NULL DEFAULT '1',

`job_run_date` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,

`component` varchar(20) NOT NULL,

`model_method` varchar(100) NOT NULL,

`custom_file` varchar(250) NOT NULL,

`is_enabled` tinyint(1) NOT NULL DEFAULT '1',

`is_new` smallint(6) NOT NULL DEFAULT '1',

`comment` varchar(200) NOT NULL,

`class_name` varchar(50) NOT NULL,

`class_method` varchar(50) NOT NULL,

PRIMARY KEY (`id`),

KEY `job_name` (`job_name`,`is_enabled`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=10 ;

-- Структура таблицы `usit_downloads`

CREATE TABLE IF NOT EXISTS `usit_downloads` (

`id` int(11) NOT NULL AUTO_INCREMENT,

`fileurl` varchar(250) NOT NULL,

`hits` int(11) NOT NULL,

PRIMARY KEY (`id`)

```

```

) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=1 ;

-- Структура таблицы `usit_event_hooks`
CREATE TABLE IF NOT EXISTS `usit_event_hooks` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `event` varchar(50) NOT NULL,
  `plugin_id` varchar(30) NOT NULL,
  PRIMARY KEY (`id`),
  KEY `event` (`event`,`plugin_id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=77 ;

-- Структура таблицы `usit_faq_cats`
CREATE TABLE IF NOT EXISTS `usit_faq_cats` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `parent_id` int(11) NOT NULL,
  `title` varchar(250) NOT NULL,
  `description` text NOT NULL,
  `published` tinyint(1) NOT NULL DEFAULT '1',
  PRIMARY KEY (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=6 ;

-- Структура таблицы `usit_faq_quests`
CREATE TABLE IF NOT EXISTS `usit_faq_quests` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `category_id` int(11) NOT NULL,
  `pubdate` datetime NOT NULL,
  `published` tinyint(1) NOT NULL,
  `quest` text NOT NULL,
  `answer` text NOT NULL,
  `user_id` int(11) NOT NULL,
  `answeruser_id` int(11) NOT NULL,
  `answerdate` datetime NOT NULL,
  `hits` int(11) NOT NULL,

```

```

PRIMARY KEY (`id`),

KEY `category_id` (`category_id`),

FULLTEXT KEY `quest` (`quest`,`answer`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=6 ;

-- Структуратаблицы `usit_filters`

CREATE TABLE IF NOT EXISTS `usit_filters` (

`id` int(11) NOT NULL AUTO_INCREMENT,

`title` varchar(100) NOT NULL,

`description` text NOT NULL,

`link` varchar(100) NOT NULL,

`published` tinyint(1) NOT NULL,

PRIMARY KEY (`id`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=8 ;

-- Структуратаблицы `usit_filter_rules`

CREATE TABLE IF NOT EXISTS `usit_filter_rules` (

`id` int(11) NOT NULL AUTO_INCREMENT,

`title` varchar(200) NOT NULL,

`find` varchar(200) NOT NULL,

`replace` text NOT NULL,

`published` tinyint(1) NOT NULL DEFAULT '1',

PRIMARY KEY (`id`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=1 ;

-- Структуратаблицы `usit_forms`

CREATE TABLE IF NOT EXISTS `usit_forms` (

`id` int(11) NOT NULL AUTO_INCREMENT,

`title` varchar(200) NOT NULL,

`description` text NOT NULL,

`email` varchar(200) NOT NULL,

`sendto` varchar(4) NOT NULL DEFAULT 'mail',

```

```

`user_id` int(11) NOT NULL DEFAULT '1',

`form_action` varchar(200) NOT NULL DEFAULT '/forms/process',

`only_fields` tinyint(1) NOT NULL DEFAULT '0',

`showtitle` tinyint(1) NOT NULL DEFAULT '1',

PRIMARY KEY (`id`),

KEY `title` (`title`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=5 ;

-- Структуратаблицы `usit_form_fields`

CREATE TABLE IF NOT EXISTS `usit_form_fields` (

`id` int(11) NOT NULL AUTO_INCREMENT,

`form_id` int(11) NOT NULL,

`title` varchar(200) NOT NULL,

`description` varchar(200) NOT NULL DEFAULT "",

`ordering` int(11) NOT NULL,

`kind` varchar(30) NOT NULL,

`mustbe` int(11) NOT NULL,

`config` text NOT NULL,

PRIMARY KEY (`id`),

KEY `form_id` (`form_id`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=27 ;

-- Структуратаблицы `usit_forums`

CREATE TABLE IF NOT EXISTS `usit_forums` (

`id` int(11) NOT NULL AUTO_INCREMENT,

`category_id` int(11) NOT NULL,

`title` varchar(250) NOT NULL,

`description` varchar(300) NOT NULL,

`access_list` tinytext NOT NULL,

`moder_list` tinytext NOT NULL,

`ordering` int(11) NOT NULL,

`published` tinyint(1) NOT NULL DEFAULT '1',

```

```

`parent_id` int(11) NOT NULL,
`NSLeft` int(11) NOT NULL,
`NSRight` int(11) NOT NULL,
`NSDiffer` varchar(15) NOT NULL,
`NSIgnore` int(11) NOT NULL,
`NSLevel` int(11) NOT NULL,
`icon` varchar(200) NOT NULL,
`topic_cost` float NOT NULL DEFAULT '0',
`thread_count` int(11) NOT NULL DEFAULT '0',
`post_count` int(11) NOT NULL DEFAULT '0',
`last_msg` text NOT NULL,
PRIMARY KEY (`id`),
KEY `category_id` (`category_id`),
KEY `parent_id` (`parent_id`),
KEY `NSLeft` (`NSLeft`,`NSRight`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=1040 ;

```

-- Структура таблицы `usit_forum_cats`

```

CREATE TABLE IF NOT EXISTS `usit_forum_cats` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `title` varchar(250) NOT NULL,
  `published` tinyint(1) NOT NULL DEFAULT '1',
  `ordering` int(11) NOT NULL,
  `seolink` varchar(200) NOT NULL,
  PRIMARY KEY (`id`),
  KEY `seolink` (`seolink`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=4 ;

```

-- Структура таблицы `usit_forum_files`

```

CREATE TABLE IF NOT EXISTS `usit_forum_files` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `post_id` int(11) NOT NULL,

```

```

`filename` varchar(200) NOT NULL,

`filesize` int(11) NOT NULL,

`hits` int(11) NOT NULL,

`pubdate` datetime NOT NULL,

PRIMARY KEY (`id`),

KEY `post_id` (`post_id`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=1 ;

-- Структура таблицы `usit_forum_polls`

CREATE TABLE IF NOT EXISTS `usit_forum_polls` (

`id` int(11) NOT NULL AUTO_INCREMENT,

`thread_id` int(11) NOT NULL,

`title` varchar(200) NOT NULL,

`description` text NOT NULL,

`answers` text NOT NULL,

`options` varchar(250) NOT NULL,

`enddate` datetime NOT NULL,

PRIMARY KEY (`id`),

KEY `thread_id` (`thread_id`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=1 ;

-- Структура таблицы `usit_forum_posts`

CREATE TABLE IF NOT EXISTS `usit_forum_posts` (

`id` int(11) NOT NULL AUTO_INCREMENT,

`thread_id` int(11) NOT NULL,

`user_id` int(11) NOT NULL,

`pinned` tinyint(1) NOT NULL DEFAULT '0',

`pubdate` datetime NOT NULL,

`editdate` datetime NOT NULL,

`edittimes` int(11) NOT NULL,

`rating` int(11) NOT NULL DEFAULT '0',

`attach_count` int(11) NOT NULL DEFAULT '0',

```

```

`content` text NOT NULL,

`content_html` text NOT NULL,

PRIMARY KEY (`id`),

KEY `thread_id` (`thread_id`,`pubdate`),

KEY `user_id` (`user_id`),

FULLTEXT KEY `content` (`content`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=31 ;

-- Структура таблицы `usit_forum_threads`

CREATE TABLE IF NOT EXISTS `usit_forum_threads` (

`id` int(11) NOT NULL AUTO_INCREMENT,

`forum_id` int(11) NOT NULL,

`user_id` int(11) NOT NULL,

`title` varchar(250) NOT NULL,

`description` varchar(250) NOT NULL,

`icon` varchar(100) NOT NULL,

`pubdate` datetime NOT NULL,

`hits` int(11) NOT NULL,

`closed` tinyint(1) NOT NULL,

`pinned` tinyint(1) NOT NULL,

`is_hidden` tinyint(1) NOT NULL DEFAULT '0',

`rel_to` varchar(15) NOT NULL,

`rel_id` int(11) NOT NULL,

`post_count` int(11) NOT NULL DEFAULT '0',

`last_msg` text NOT NULL,

PRIMARY KEY (`id`),

KEY `user_id` (`user_id`),

KEY `forum_id` (`forum_id`),

KEY `rel_id` (`user_id`),

FULLTEXT KEY `title` (`title`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=14 ;

```

```

-- Структура таблицы `usit_forum_votes`
CREATE TABLE IF NOT EXISTS `usit_forum_votes` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `poll_id` int(11) NOT NULL,
  `answer` varchar(300) NOT NULL,
  `user_id` int(11) NOT NULL,
  `pubdate` datetime NOT NULL,
  PRIMARY KEY (`id`),
  KEY `user_id` (`user_id`),
  KEY `poll_id` (`poll_id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=1 ;

```

```

--
-- Структура таблицы `usit_menu`
CREATE TABLE IF NOT EXISTS `usit_menu` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `menu` varchar(200) NOT NULL,
  `title` varchar(200) NOT NULL,
  `link` varchar(200) NOT NULL,
  `linktype` varchar(12) NOT NULL DEFAULT 'link',
  `linkid` varchar(25) DEFAULT '-1',
  `target` varchar(8) NOT NULL DEFAULT '_self',
  `component` varchar(100) NOT NULL,
  `ordering` int(11) NOT NULL DEFAULT '1',
  `published` tinyint(1) NOT NULL,
  `template` varchar(30) NOT NULL,
  `access_list` tinytext NOT NULL,
  `iconurl` varchar(100) NOT NULL,

```

```

`NSLeft` int(11) NOT NULL,
`NSRight` int(11) NOT NULL,
`NSLevel` int(11) NOT NULL,
`NSDiffer` varchar(40) DEFAULT NULL,
`NSIgnore` int(11) NOT NULL,
`parent_id` int(11) NOT NULL,
PRIMARY KEY (`id`),
KEY `parent_id` (`parent_id`),
KEY `NSLeft` (`NSLeft`,`NSRight`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=46 ;

```

```
-- Структуратаблицы `usit_modules`
```

```

CREATE TABLE IF NOT EXISTS `usit_modules` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `position` varchar(20) NOT NULL,
  `name` varchar(200) NOT NULL,
  `title` varchar(200) NOT NULL,
  `is_external` tinyint(1) NOT NULL,
  `content` text NOT NULL,
  `ordering` int(11) NOT NULL DEFAULT '1',
  `showtitle` int(11) NOT NULL DEFAULT '1',
  `published` tinyint(1) NOT NULL DEFAULT '1',
  `user` int(11) NOT NULL DEFAULT '0',
  `config` text NOT NULL,
  `original` int(11) NOT NULL DEFAULT '1',
  `css_prefix` varchar(30) NOT NULL,
  `access_list` tinytext NOT NULL,
  `cache` int(11) NOT NULL,
  `cachetime` int(11) NOT NULL DEFAULT '1',
  `cacheint` varchar(15) NOT NULL DEFAULT 'HOUR',
  `template` varchar(35) NOT NULL DEFAULT 'module.tpl',

```

```

`is_strict_bind` tinyint(1) NOT NULL DEFAULT '0',
`author` varchar(100) NOT NULL DEFAULT 'InstantCMS team',
`version` varchar(6) NOT NULL DEFAULT '1.0',
PRIMARY KEY (`id`),
KEY `position` (`position`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=93 ;
-- Структуратаблицы `usit_modules_bind`
CREATE TABLE IF NOT EXISTS `usit_modules_bind` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `module_id` int(11) NOT NULL,
  `menu_id` int(11) NOT NULL,
  `position` varchar(20) NOT NULL,
  PRIMARY KEY (`id`),
  KEY `position` (`position`),
  KEY `module_id` (`module_id`),
  KEY `menu_id` (`menu_id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=724 ;
-- Структуратаблицы `usit_ns_transactions`
CREATE TABLE IF NOT EXISTS `usit_ns_transactions` (
  `IDTransaction` int(10) unsigned NOT NULL AUTO_INCREMENT,
  `TableName` tinytext,
  `Differ` tinytext,
  `InTransaction` bit(1) DEFAULT NULL,
  `TStamp` timestamp NULL DEFAULT NULL,
  PRIMARY KEY (`IDTransaction`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=1 ;
-- Структуратаблицы `usit_online`
CREATE TABLE IF NOT EXISTS `usit_online` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `ip` varchar(15) NOT NULL,

```

```

`sess_id` varchar(100) NOT NULL,

`lastdate` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,

`user_id` int(11) NOT NULL,

`agent` varchar(250) NOT NULL,

`viewurl` varchar(250) NOT NULL,

PRIMARY KEY (`id`),

UNIQUE KEY `sess_id` (`sess_id`),

KEY `user_id` (`user_id`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=9344 ;

-- Структура таблицы `usit_photo_albums`

CREATE TABLE IF NOT EXISTS `usit_photo_albums` (

  `id` int(11) NOT NULL AUTO_INCREMENT,

  `parent_id` int(11) NOT NULL,

  `ordering` int(11) NOT NULL DEFAULT '1',

  `NSLeft` int(11) NOT NULL,

  `NSRight` int(11) NOT NULL,

  `NSDiffer` varchar(15) NOT NULL,

  `NSIgnore` int(11) NOT NULL,

  `NSLevel` int(11) NOT NULL,

  `title` varchar(100) NOT NULL,

  `description` varchar(300) NOT NULL,

  `published` tinyint(1) NOT NULL,

  `showdate` tinyint(1) NOT NULL DEFAULT '1',

  `iconurl` varchar(100) NOT NULL,

  `pubdate` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,

  `orderby` varchar(20) NOT NULL DEFAULT 'title',

  `orderto` varchar(4) NOT NULL DEFAULT 'asc',

  `public` tinyint(1) NOT NULL,

  `perpage` int(11) NOT NULL DEFAULT '15',

  `cssprefix` varchar(20) NOT NULL,

```

```

`thumb1` int(11) NOT NULL DEFAULT '96',
`thumb2` int(11) NOT NULL DEFAULT '480',
`thumbsqr` tinyint(1) NOT NULL DEFAULT '1',
`showtype` varchar(10) NOT NULL DEFAULT 'lightbox',
`nav` tinyint(1) NOT NULL DEFAULT '1',
`uplimit` int(11) NOT NULL,
`maxcols` int(11) NOT NULL DEFAULT '4',
`orderform` tinyint(1) NOT NULL DEFAULT '1',
`showtags` tinyint(1) NOT NULL DEFAULT '1',
`bbcode` tinyint(1) NOT NULL DEFAULT '1',
`user_id` int(11) NOT NULL DEFAULT '1',
`is_comments` tinyint(1) NOT NULL,
PRIMARY KEY (`id`),
KEY `user_id` (`user_id`),
KEY `parent_id` (`parent_id`),
KEY `NSLeft` (`NSLeft`,`NSRight`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=1050 ;
-- Структура таблицы `usit_photo_files`
CREATE TABLE IF NOT EXISTS `usit_photo_files` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `album_id` int(11) NOT NULL,
  `title` varchar(200) NOT NULL,
  `description` text NOT NULL,
  `pubdate` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
  `file` varchar(200) NOT NULL,
  `published` tinyint(1) NOT NULL,
  `hits` int(11) NOT NULL,
  `showdate` tinyint(1) NOT NULL DEFAULT '1',
  `comments` tinyint(1) NOT NULL DEFAULT '1',
  `user_id` int(11) NOT NULL DEFAULT '1',

```

```

`owner` varchar(10) DEFAULT 'photos',
`rating` int(11) NOT NULL DEFAULT '0',
PRIMARY KEY (`id`),
KEY `album_id` (`album_id`),
KEY `user_id` (`user_id`),
KEY `owner` (`owner`),
FULLTEXT KEY `title` (`title`,`description`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=34 ;

-- Структуратаблицы `usit_plugins`
CREATE TABLE IF NOT EXISTS `usit_plugins` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `plugin` varchar(30) NOT NULL,
  `title` varchar(255) NOT NULL,
  `description` text NOT NULL,
  `author` varchar(255) NOT NULL,
  `version` varchar(15) NOT NULL,
  `plugin_type` varchar(10) NOT NULL,
  `published` tinyint(1) NOT NULL DEFAULT '0',
  `config` text NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=24 ;

-- Структуратаблицы `usit_polls`

CREATE TABLE IF NOT EXISTS `usit_polls` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `title` varchar(200) NOT NULL,
  `pubdate` date NOT NULL DEFAULT '0000-00-00',
  `answers` text NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=4 ;

```

```

-- Структура таблицы `usit_polls_log`
CREATE TABLE IF NOT EXISTS `usit_polls_log` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `poll_id` int(11) NOT NULL,
  `answer` varchar(300) NOT NULL,
  `user_id` int(11) NOT NULL,
  `ip` varchar(15) NOT NULL,
  PRIMARY KEY (`id`),
  KEY `user_id` (`user_id`),
  KEY `poll_id` (`poll_id`),
  KEY `ip` (`ip`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=17 ;

-- Структура таблицы `usit_price_cats`
CREATE TABLE IF NOT EXISTS `usit_price_cats` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `title` varchar(200) NOT NULL,
  `description` text NOT NULL,
  `published` tinyint(1) NOT NULL DEFAULT '1',
  PRIMARY KEY (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=10 ;

-- Структура таблицы `usit_price_items`
CREATE TABLE IF NOT EXISTS `usit_price_items` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `category_id` int(11) NOT NULL,
  `title` varchar(230) NOT NULL,
  `price` float NOT NULL,
  `published` tinyint(1) NOT NULL DEFAULT '1',
  `canmany` int(11) NOT NULL DEFAULT '1',
  PRIMARY KEY (`id`),
  FULLTEXT KEY `title` (`title`)
)

```

```
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=892 ;
```

```
-- Структура таблицы `usit_ratings`
```

```
CREATE TABLE IF NOT EXISTS `usit_ratings` (
```

```
  `id` int(11) NOT NULL AUTO_INCREMENT,
```

```
  `item_id` int(11) NOT NULL,
```

```
  `points` int(11) NOT NULL,
```

```
  `ip` varchar(20) NOT NULL,
```

```
  `target` varchar(20) NOT NULL,
```

```
  `user_id` int(11) NOT NULL DEFAULT '1',
```

```
  `pubdate` datetime NOT NULL,
```

```
  PRIMARY KEY (`id`),
```

```
  KEY `user_id` (`user_id`),
```

```
  KEY `item_id` (`item_id`)
```

```
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=25 ;
```

```
-- Структура таблицы `usit_ratings_total`
```

```
CREATE TABLE IF NOT EXISTS `usit_ratings_total` (
```

```
  `id` int(11) NOT NULL AUTO_INCREMENT,
```

```
  `target` varchar(32) NOT NULL,
```

```
  `item_id` mediumint(9) NOT NULL,
```

```
  `total_rating` int(11) NOT NULL,
```

```
  `total_votes` int(11) NOT NULL,
```

```
  PRIMARY KEY (`id`),
```

```
  KEY `item_id` (`item_id`),
```

```
  KEY `target` (`target`,`item_id`)
```

```
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=19 ;
```

```
-- Структура таблицы `usit_rating_targets`
```

```
CREATE TABLE IF NOT EXISTS `usit_rating_targets` (
```

```
  `id` int(11) NOT NULL AUTO_INCREMENT,
```

```
  `target` varchar(32) NOT NULL,
```

```
  `component` varchar(32) NOT NULL,
```

```

`is_user_affect` tinyint(4) NOT NULL,
`user_weight` smallint(6) NOT NULL,
`target_table` varchar(32) NOT NULL,
`target_title` varchar(70) NOT NULL,
PRIMARY KEY (`id`),
UNIQUE KEY `target` (`target`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=8 ;

-- Структура таблицы `usit_sanction`
CREATE TABLE IF NOT EXISTS `usit_sanction` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `date` datetime NOT NULL COMMENT 'Дата выдачи ограничения',
  `free_date` datetime NOT NULL COMMENT 'Дата снятия ограничения',
  `user_id` int(11) NOT NULL COMMENT 'Идентификатор пользователя',
  `access` text NOT NULL COMMENT 'Список ограничений',
  `who_gave_id` int(11) NOT NULL COMMENT 'Идентификатор выдавшего ограничение',
  `reason` text NOT NULL COMMENT 'Причина ограничений',
  PRIMARY KEY (`id`),
  KEY `user_id` (`user_id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 COMMENT='Компонент: Санкции, автор: lokanaft'
AUTO_INCREMENT=4 ;

-- Структура таблицы `usit_sanction_notice`
CREATE TABLE IF NOT EXISTS `usit_sanction_notice` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `date` datetime NOT NULL,
  `user_id` int(11) NOT NULL,
  `who_gave_id` int(11) NOT NULL,
  `reason` text NOT NULL,
  PRIMARY KEY (`id`),
  KEY `user_id` (`user_id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=5 ;

-- Структура таблицы `usit_search`

```

```

CREATE TABLE IF NOT EXISTS `usit_search` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `session_id` varchar(100) NOT NULL,
  `date` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
  `pubdate` datetime DEFAULT NULL,
  `title` varchar(250) NOT NULL,
  `description` varchar(500) NOT NULL,
  `link` varchar(200) NOT NULL,
  `place` varchar(100) NOT NULL,
  `placelink` varchar(200) NOT NULL,
  PRIMARY KEY (`id`),
  KEY `session_id` (`session_id`),
  KEY `date` (`date`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=62 ;

```

-- Структура таблицы `usit_seo_page`

```

CREATE TABLE IF NOT EXISTS `usit_seo_page` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `note` text NOT NULL,
  `component` varchar(50) NOT NULL,
  `request_uri` varchar(200) NOT NULL,
  `target_uri` varchar(200) NOT NULL,
  `page_title` varchar(200) NOT NULL,
  `meta_desc` text NOT NULL,
  `meta_keys` text NOT NULL,
  `published` int(11) DEFAULT '1',
  PRIMARY KEY (`id`),
  KEY `request` (`request_uri`),
  KEY `target` (`target_uri`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=1 ;

```

-- Структура таблицы `usit_stats`

```

CREATE TABLE IF NOT EXISTS `usit_stats` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `ip` varchar(20) NOT NULL,
  `logdate` datetime NOT NULL,
  `page` varchar(200) NOT NULL,
  `agent` varchar(60) DEFAULT 'unknown',
  `refer` varchar(200) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=962 ;

```

-- Структура таблицы `usit_subscribe`

```

CREATE TABLE IF NOT EXISTS `usit_subscribe` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `user_id` int(11) NOT NULL,
  `target` varchar(20) NOT NULL,
  `target_id` int(11) NOT NULL,
  `pubdate` datetime NOT NULL,
  PRIMARY KEY (`id`),
  KEY `user_id` (`user_id`),
  KEY `target_id` (`target_id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=4 ;

```

-- Структура таблицы `usit_tags`

```

CREATE TABLE IF NOT EXISTS `usit_tags` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `tag` varchar(250) NOT NULL,
  `target` varchar(25) NOT NULL,
  `item_id` int(11) NOT NULL,
  PRIMARY KEY (`id`),
  KEY `tag` (`tag`),
  KEY `item_id` (`item_id`)

```

```
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=508 ;
```

```
-- Структура таблицы `usit_uc_cart`
```

```
CREATE TABLE IF NOT EXISTS `usit_uc_cart` (
```

```
  `id` int(11) NOT NULL AUTO_INCREMENT,
```

```
  `user_id` int(11) NOT NULL,
```

```
  `session_id` varchar(50) NOT NULL,
```

```
  `item_id` int(11) NOT NULL,
```

```
  `pubdate` datetime NOT NULL,
```

```
  `itemscount` int(11) NOT NULL,
```

```
  PRIMARY KEY (`id`)
```

```
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=19 ;
```

```
-- Структура таблицы `usit_uc_cats`
```

```
CREATE TABLE IF NOT EXISTS `usit_uc_cats` (
```

```
  `id` int(11) NOT NULL AUTO_INCREMENT,
```

```
  `parent_id` int(11) NOT NULL,
```

```
  `title` varchar(200) NOT NULL,
```

```
  `description` text NOT NULL,
```

```
  `published` tinyint(1) NOT NULL DEFAULT '1',
```

```
  `fieldsstruct` text,
```

```
  `view_type` varchar(20) NOT NULL DEFAULT 'list',
```

```
  `fields_show` int(11) NOT NULL DEFAULT '10',
```

```
  `showmore` tinyint(1) NOT NULL DEFAULT '1',
```

```
  `perpage` int(11) NOT NULL DEFAULT '20',
```

```
  `showtags` tinyint(1) NOT NULL DEFAULT '1',
```

```
  `showsort` tinyint(1) NOT NULL DEFAULT '1',
```

```
  `is_ratings` tinyint(1) NOT NULL,
```

```
  `orderby` varchar(12) NOT NULL DEFAULT 'pubdate',
```

```
  `orderto` varchar(10) DEFAULT 'desc',
```

```
  `showabc` tinyint(1) NOT NULL DEFAULT '1',
```

```
  `shownew` tinyint(1) NOT NULL,
```

```

`newint` varchar(100) NOT NULL,

`filters` int(11) NOT NULL,

`is_shop` tinyint(1) NOT NULL,

`NSLeft` int(11) NOT NULL,

`NSRight` int(11) NOT NULL,

`NSLevel` int(11) NOT NULL,

`NSDiffer` int(11) NOT NULL,

`NSIgnore` int(11) NOT NULL,

`ordering` int(11) NOT NULL,

`is_public` tinyint(1) NOT NULL,

`can_edit` int(11) NOT NULL DEFAULT '0',

`cost` varchar(5) NOT NULL,

PRIMARY KEY (`id`),

KEY `NSLeft` (`NSLeft`,`NSRight`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=1001 ;

-- Структура таблицы `usit_uc_cats_access`

CREATE TABLE IF NOT EXISTS `usit_uc_cats_access` (

`cat_id` int(11) NOT NULL,

`group_id` int(11) NOT NULL,

KEY `cat_id` (`cat_id`,`group_id`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8;

-- Структура таблицы `usit_uc_discount`

CREATE TABLE IF NOT EXISTS `usit_uc_discount` (

`id` int(11) NOT NULL AUTO_INCREMENT,

`title` varchar(150) NOT NULL,

`cat_id` int(11) NOT NULL,

`sign` tinyint(4) NOT NULL,

`value` float NOT NULL,

`unit` varchar(10) NOT NULL,

`if_limit` int(11) NOT NULL,

```

```

PRIMARY KEY (`id`),

KEY `cat_id` (`cat_id`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=4 ;

-- Структура таблицы `usit_uc_items`

CREATE TABLE IF NOT EXISTS `usit_uc_items` (

`id` int(11) NOT NULL AUTO_INCREMENT,

`category_id` int(11) NOT NULL,

`title` varchar(200) NOT NULL,

`pubdate` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',

`published` tinyint(1) NOT NULL DEFAULT '1',

`imageurl` varchar(200) NOT NULL,

`fieldsdata` text NOT NULL,

`hits` int(11) NOT NULL,

`is_comments` tinyint(1) NOT NULL,

`tags` varchar(200) NOT NULL,

`rating` float NOT NULL,

`meta_desc` varchar(250) NOT NULL,

`meta_keys` varchar(250) NOT NULL,

`price` float NOT NULL,

`canmany` int(11) NOT NULL DEFAULT '1',

`user_id` int(11) NOT NULL,

`on_moderate` int(11) NOT NULL,

PRIMARY KEY (`id`),

KEY `category_id` (`category_id`),

FULLTEXT KEY `title` (`title`,`fieldsdata`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=8 ;

-- Структура таблицы `usit_uc_ratings`

CREATE TABLE IF NOT EXISTS `usit_uc_ratings` (

`id` int(11) NOT NULL AUTO_INCREMENT,

`item_id` int(11) NOT NULL,

```

```

`points` int(11) NOT NULL,
`ip` varchar(16) NOT NULL,
PRIMARY KEY (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=1 ;

-- Структура таблицы `usit_uc_tags`
CREATE TABLE IF NOT EXISTS `usit_uc_tags` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `tag` varchar(200) NOT NULL,
  `item_id` int(11) NOT NULL,
  PRIMARY KEY (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=1 ;

-- Структура таблицы `usit_upload_images`
CREATE TABLE IF NOT EXISTS `usit_upload_images` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `target_id` int(11) NOT NULL DEFAULT '0',
  `session_id` varchar(50) NOT NULL,
  `fileurl` varchar(250) NOT NULL,
  `target` varchar(25) NOT NULL DEFAULT 'forum',
  `component` varchar(32) NOT NULL,
  PRIMARY KEY (`id`),
  KEY `target_id` (`target_id`),
  KEY `session_id` (`session_id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=1 ;

-- Структура таблицы `usit_users`
CREATE TABLE IF NOT EXISTS `usit_users` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `group_id` int(11) NOT NULL DEFAULT '1',
  `login` varchar(100) NOT NULL,
  `nickname` varchar(100) NOT NULL,
  `password` varchar(100) NOT NULL,

```

```

`email` varchar(100) NOT NULL,

`icq` varchar(15) NOT NULL,

`regdate` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',

`logdate` datetime NOT NULL DEFAULT '0000-00-00 00:00:00',

`birthdate` date NOT NULL DEFAULT '0000-00-00',

`is_locked` tinyint(1) NOT NULL,

`is_deleted` tinyint(1) NOT NULL,

`is_logged_once` tinyint(1) NOT NULL DEFAULT '0',

`rating` int(11) NOT NULL,

`points` int(11) NOT NULL,

`last_ip` varchar(15) NOT NULL,

`status` varchar(255) NOT NULL,

`status_date` datetime NOT NULL,

`invited_by` int(11) DEFAULT NULL,

`invdate` datetime DEFAULT NULL,

`openid` varchar(250) DEFAULT NULL,

PRIMARY KEY (`id`),

KEY `login` (`login`),

KEY `email` (`email`),

KEY `birthdate` (`birthdate`),

KEY `group_id` (`group_id`),

KEY `invited_by` (`invited_by`),

KEY `openid` (`openid`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 ROW_FORMAT=DYNAMIC AUTO_INCREMENT=49 ;

-- Структуратаблицы `usit_users_activate`

CREATE TABLE IF NOT EXISTS `usit_users_activate` (

`id` int(11) NOT NULL AUTO_INCREMENT,

`pubdate` datetime NOT NULL,

`user_id` int(11) NOT NULL,

`code` varchar(50) NOT NULL,

```

```

PRIMARY KEY (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=31 ;

-- Структура таблицы `usit_user_albums`
CREATE TABLE IF NOT EXISTS `usit_user_albums` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `user_id` int(11) NOT NULL,
  `title` varchar(100) NOT NULL,
  `pubdate` datetime NOT NULL,
  `allow_who` varchar(10) NOT NULL,
  `description` varchar(200) NOT NULL DEFAULT "",
  PRIMARY KEY (`id`),
  KEY `user_id` (`user_id`),
  KEY `allow_who` (`allow_who`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=8 ;

-- Структура таблицы `usit_user_autoawards`
CREATE TABLE IF NOT EXISTS `usit_user_autoawards` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `title` varchar(200) NOT NULL,
  `description` varchar(200) NOT NULL,
  `imageurl` varchar(200) NOT NULL,
  `p_comment` int(11) NOT NULL,
  `p_blog` int(11) NOT NULL,
  `p_forum` int(11) NOT NULL,
  `p_photo` int(11) NOT NULL,
  `p_privphoto` int(11) DEFAULT NULL,
  `p_content` int(11) NOT NULL,
  `p_karma` int(11) NOT NULL,
  `published` tinyint(1) NOT NULL DEFAULT '1',
  PRIMARY KEY (`id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=3 ;

```

```

-- Структура таблицы `usit_user_awards`
CREATE TABLE IF NOT EXISTS `usit_user_awards` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `user_id` int(11) NOT NULL,
  `pubdate` datetime NOT NULL,
  `title` varchar(250) NOT NULL,
  `description` text NOT NULL,
  `imageurl` varchar(100) NOT NULL,
  `from_id` int(11) NOT NULL,
  `award_id` int(11) NOT NULL,
  PRIMARY KEY (`id`),
  KEY `user_id` (`user_id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=5 ;

-- Структура таблицы `usit_user_clubs`
CREATE TABLE IF NOT EXISTS `usit_user_clubs` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `user_id` int(11) NOT NULL,
  `club_id` int(11) NOT NULL,
  `role` varchar(20) NOT NULL DEFAULT 'member',
  `pubdate` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP,
  PRIMARY KEY (`id`),
  KEY `user_id` (`user_id`),
  KEY `club_id` (`club_id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=12 ;

-- Структура таблицы `usit_user_files`
CREATE TABLE IF NOT EXISTS `usit_user_files` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `user_id` int(11) NOT NULL,
  `filename` varchar(250) NOT NULL,
  `pubdate` datetime NOT NULL,

```

```

`allow_who` varchar(50) NOT NULL,

`filesize` int(11) NOT NULL,

`hits` int(11) NOT NULL,

PRIMARY KEY (`id`),

KEY `user_id` (`user_id`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=1 ;

-- Структура таблицы `usit_user_friends`

CREATE TABLE IF NOT EXISTS `usit_user_friends` (

`id` int(11) NOT NULL AUTO_INCREMENT,

`to_id` int(11) NOT NULL,

`from_id` int(11) NOT NULL,

`logdate` datetime NOT NULL,

`is_accepted` tinyint(1) NOT NULL,

PRIMARY KEY (`id`),

KEY `to_id` (`to_id`),

KEY `from_id` (`from_id`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=35 ;

-- Структура таблицы `usit_user_groups`

CREATE TABLE IF NOT EXISTS `usit_user_groups` (

`id` int(11) NOT NULL AUTO_INCREMENT,

`title` varchar(200) NOT NULL,

`alias` varchar(100) NOT NULL,

`is_admin` tinyint(1) NOT NULL,

`access` text NOT NULL,

PRIMARY KEY (`id`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=11 ;

-- Структура таблицы `usit_user_groups_access`

CREATE TABLE IF NOT EXISTS `usit_user_groups_access` (

`id` int(11) unsigned NOT NULL AUTO_INCREMENT,

`access_type` varchar(60) NOT NULL,

```

```

`access_name` varchar(100) NOT NULL,

PRIMARY KEY (`id`),

UNIQUE KEY `access_type` (`access_type`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=15 ;

-- Структура таблицы `usit_user_guests`

CREATE TABLE IF NOT EXISTS `usit_user_guests` (

  `id` int(11) NOT NULL AUTO_INCREMENT,

  `user_id` int(11) NOT NULL DEFAULT '0',

  `guest_id` int(11) NOT NULL DEFAULT '0',

  `date` timestamp NOT NULL DEFAULT CURRENT_TIMESTAMP ON UPDATE CURRENT_TIMESTAMP,

  UNIQUE KEY `user_id` (`user_id`,`guest_id`),

  KEY `id` (`id`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=178 ;

-- Структура таблицы `usit_user_invites`

CREATE TABLE IF NOT EXISTS `usit_user_invites` (

  `id` int(11) NOT NULL AUTO_INCREMENT,

  `code` varchar(32) NOT NULL,

  `owner_id` int(11) NOT NULL,

  `createdate` datetime NOT NULL,

  `is_used` tinyint(1) NOT NULL DEFAULT '0',

  `is_sended` tinyint(1) NOT NULL DEFAULT '0',

  PRIMARY KEY (`id`),

  KEY `code` (`code`,`owner_id`,`is_used`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=1 ;

-- Структура таблицы `usit_user_karma`

CREATE TABLE IF NOT EXISTS `usit_user_karma` (

  `id` int(11) NOT NULL AUTO_INCREMENT,

  `user_id` int(11) NOT NULL,

  `sender_id` int(11) NOT NULL,

```

```

`points` smallint(6) NOT NULL,
`sendeddate` datetime NOT NULL,
PRIMARY KEY (`id`),
KEY `user_id` (`user_id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=67 ;

-- Структура таблицы `usit_user_msg`
CREATE TABLE IF NOT EXISTS `usit_user_msg` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `to_id` int(11) NOT NULL,
  `from_id` int(11) NOT NULL,
  `sendeddate` datetime NOT NULL,
  `is_new` tinyint(1) NOT NULL DEFAULT '1',
  `message` text NOT NULL,
  `to_del` tinyint(1) NOT NULL DEFAULT '0',
  `from_del` tinyint(1) NOT NULL DEFAULT '0',
  PRIMARY KEY (`id`),
  KEY `to_id` (`to_id`),
  KEY `from_id` (`from_id`),
  KEY `to_del` (`to_del`),
  KEY `from_del` (`from_del`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=92 ;

-- Структура таблицы `usit_user_photos`
CREATE TABLE IF NOT EXISTS `usit_user_photos` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `user_id` int(11) NOT NULL,
  `album_id` int(11) NOT NULL,
  `pubdate` datetime NOT NULL,
  `title` varchar(250) NOT NULL,
  `description` text NOT NULL,
  `allow_who` varchar(15) NOT NULL DEFAULT 'all',

```

```

`hits` int(11) NOT NULL,

`imageurl` varchar(250) NOT NULL,

PRIMARY KEY (`id`),

KEY `user_id` (`user_id`),

KEY `album_id` (`album_id`)

) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=15 ;

-- Структура таблицы `usit_user_profiles`

CREATE TABLE IF NOT EXISTS `usit_user_profiles` (

`id` int(11) NOT NULL AUTO_INCREMENT,

`user_id` int(11) NOT NULL,

`city` varchar(250) NOT NULL,

`description` varchar(500) NOT NULL,

`showmail` tinyint(1) NOT NULL DEFAULT '0',

`showbirth` tinyint(1) NOT NULL DEFAULT '0',

`showicq` tinyint(1) NOT NULL DEFAULT '0',

`karma` int(11) NOT NULL DEFAULT '0',

`imageurl` varchar(250) NOT NULL,

`allow_who` varchar(35) NOT NULL DEFAULT 'all',

`signature` varchar(240) NOT NULL,

`signature_html` varchar(300) NOT NULL,

`gender` varchar(1) NOT NULL,

`formsdata` varchar(800) NOT NULL,

`email_newmsg` int(11) NOT NULL DEFAULT '1',

`cm_subscribe` varchar(4) NOT NULL DEFAULT 'both',

`stats` varchar(500) NOT NULL,

PRIMARY KEY (`id`),

KEY `user_id` (`user_id`),

KEY `city` (`city`),

KEY `description` (`description`(333)),

KEY `formsdata` (`formsdata`(333)),

```

```
KEY `gender` (`gender`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 ROW_FORMAT=DYNAMIC AUTO_INCREMENT=49 ;

-- Структура таблицы `usit_user_wall`

CREATE TABLE IF NOT EXISTS `usit_user_wall` (
  `id` int(11) NOT NULL AUTO_INCREMENT,
  `user_id` int(11) NOT NULL,
  `author_id` int(11) NOT NULL,
  `pubdate` datetime NOT NULL,
  `content` text NOT NULL,
  `usertype` varchar(8) NOT NULL DEFAULT 'users',
  PRIMARY KEY (`id`),
  KEY `user_id` (`user_id`),
  KEY `author_id` (`author_id`)
) ENGINE=MyISAM DEFAULT CHARSET=utf8 AUTO_INCREMENT=18 ;
```