

Московский авиационный институт
(государственный ТЕХнический университет)

Факультет прикладной математики

Кафедра вычислительной математики и программирования

Курсовой проект по курсу
«Проектирование Баз Данных»

Студент: И. К. Никитин
Преподаватели: А. М. Марасанов
О. О. Бородин

Москва, 2009

Содержание

I	Общее описание	3
1	Глобальное задание	3
2	UseCase-диаграмма	5
3	Интерфейс	7
II	Проектирование	8
4	ER-модель	8
III	Реализация	9
5	Вставка	9
6	Запросы	13
1	Само-комментарии	13
2	Самые обсуждаемые статьи	14
7	Процедуры. Пользовательский модуль.	15
8	Триггеры	18
IV	Словарь терминов	20

Задача

- Выбрать тему лабораторных работ.
- Описать словарь терминов и составить техническое задание.
- Нарисовать UseCase-диаграмму проекта, в соответствие с темой.
- Составить Ег-модель базы данных проекта.
- Создать базу данных под управлением СУБД Oracle.
А в ней написать:
 - Заполняющий скрипт
 - Несколько вложенных запросов, в соответствие с темой.
 - Хранимые процедуры и триггеры.

Часть I

Общее описание

1 Глобальное задание

Требуется написать социальную сеть для Международной Лиги Молодых Журналистов. Необходимо иметь *Личную Страницу* для каждого пользователя.

Пользователи делятся на (роли):

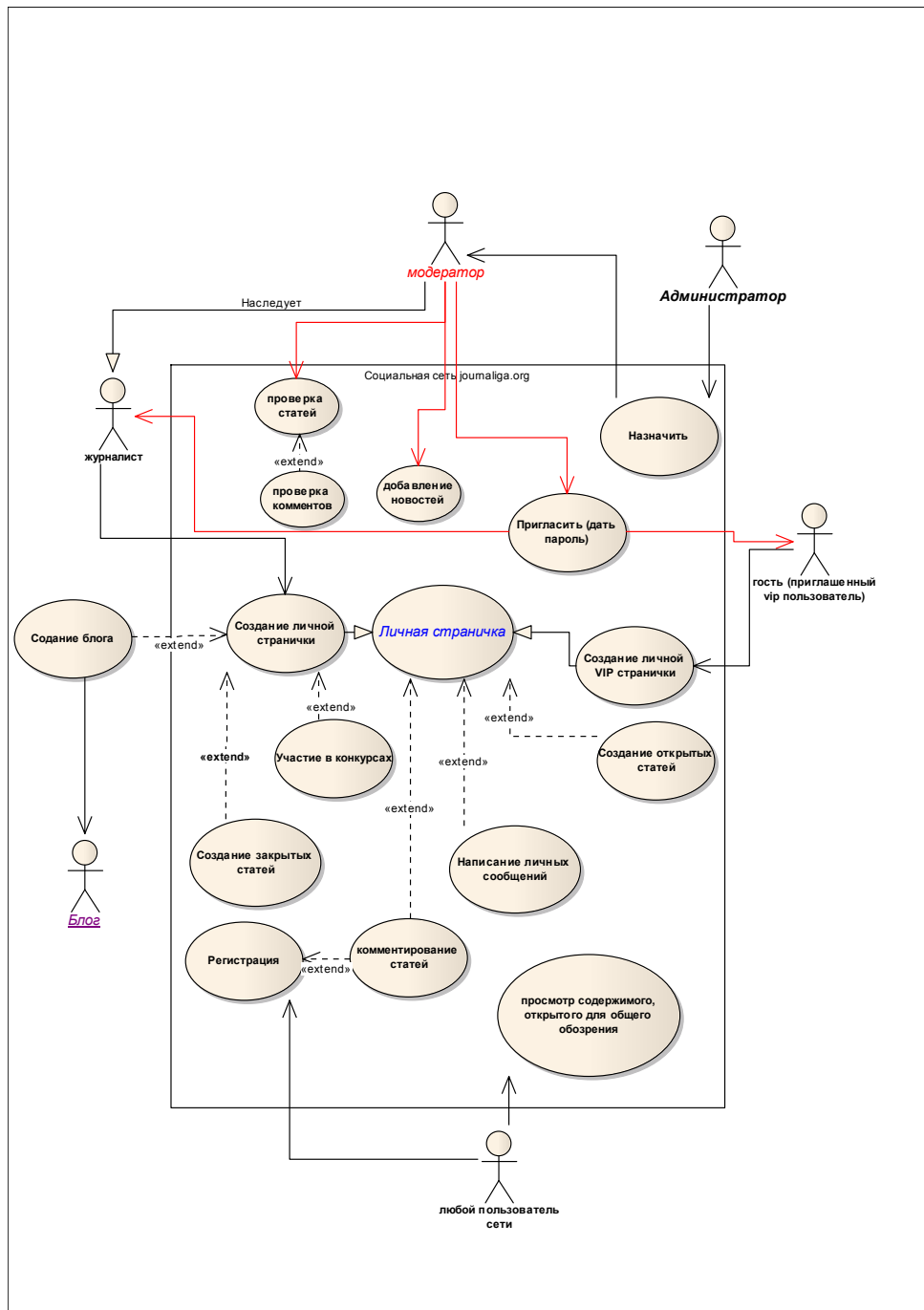
- Не зарегистрированных пользователей.
 - Могут просматривать только открытые статьи, комментарии и личные страницы участников.
 - Зарегистрироваться на общих правах.
- Зарегистрированных пользователей.
 - Могут просматривать и комментировать только открытые статьи, комментарии и личные страницы участников.
- Зарегистрированных журналистов.
 - Могут создавать, просматривать, комментировать любые статьи, комментарии и личные страницы участников.
 - Принимать, отправлять личные сообщения.
 - Участвовать в конкурсах.
- Vip-Гостей
 - Могут создавать, просматривать, комментировать открытые статьи, комментарии и личные страницы участников.
 - Принимать отправлять личные сообщения.
- Модераторов. Кроме того, что обладают всеми правами **Зарегистрированных журналистов**.
 - Создавать аккаунты журналистов.
 - Создавать аккаунты VIP-гостей.
 - Редактировать статьи и комментарии к ним.

- Админа.
 - Создаёт аккаунты модераторов.

;

2 UseCase-диаграмма

(Диаграмма создана в системе Enterprise Architect)



;

3 Интерфейс

Мы не будем здесь приводить требования к интерфейсу системы, т.к.

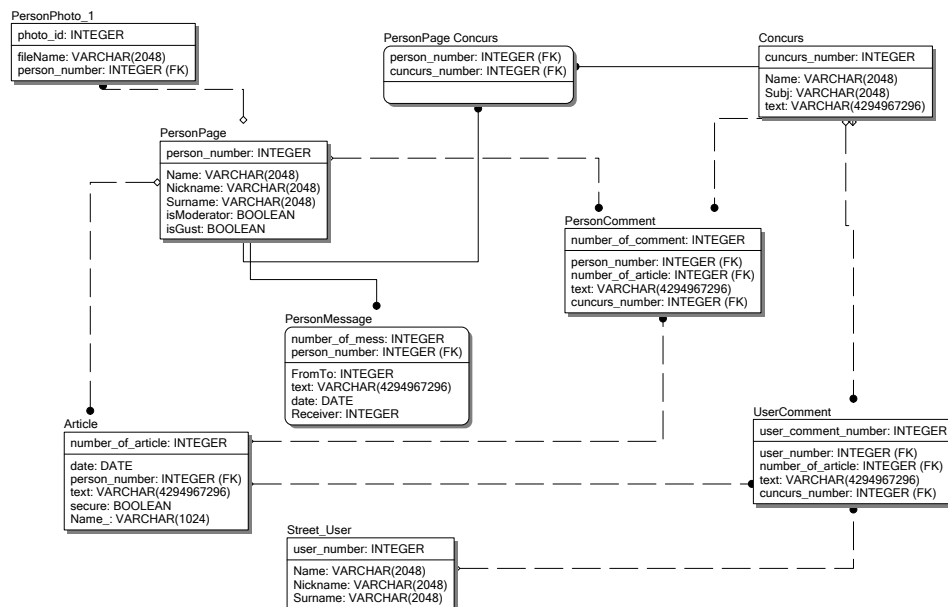
- если не обращать внимания на художественные изыски, то все социальные сети выглядят одинаково
- понятие об интерфейсе следует из описания выше
- это выходит за рамки данных лабораторных работ и относится скорее к проектированию клиента, чем всей системы.

Часть II

Проектирование

4 ER-модель

Покажем ER-модель для базы данных социальной сети. (Модель создана в системе ERWin7.3)



Замечание: В данной модели комментарии бывают двух типов: к конкурсу, ко статье. Чтобы отличать их использована маркировка числом -1

- Если это комментарий к статье то $concurs_number = -1$
- Если это комментарий к конкурсу то $number_of_article = -1$

Для маленьких проектов такой подход удобен — проект не становится больше. В общем случае это опасно, так как является исключением. Рано или поздно нам потребуется ввести ещё одно исключение (маркировка числом -2). Их количество может возрасти до неуправляемого.

Часть III

Реализация

База данных рассматриваемой социальной сети реализована на СУБД Oracle. SQL-запросы, создающие нужные страницы автоматически генерируются с помощью ErWin. Далее необходимо заполнить с помощью запросов, которые мы напишем сами.

5 Вставка

Для начала операция вставки была написана, как тестовый вариант, чтобы иметь какие-то данные в базе. На самом деле при установке «на боевой сервер» данный скрипт следует удалить.

```
1  -- УДАЛИМ СТАРЫЕ ЗАПИСИ
2  delete from USERCOMMENT;
3  delete from STREET_USER;
4  delete from PERSONPHOTO_1;
5  delete from PERSONMESSAGE;
6  delete from PERSONCOMMENT;
7  delete from ARTICLE;
8  delete from PERSONPAGE_CONCURS;
9  delete from PERSONPAGE;
10 delete from CONCURS;
11 /*
12   Создадим:
13  */
14 -- PERSONPAGE:
15
16 insert into PERSONPAGE(PERSON_NUMBER, NICKNAME, NAME_,
17   SURNAME, ISMODERATOR, ISGUST)
18   values(0, 'w495', 'Илюха', 'Никитин', 1, 0);
19 insert into PERSONPAGE(PERSON_NUMBER, NICKNAME, NAME_,
20   SURNAME, ISMODERATOR, ISGUST)
21   values(1, 'unleo', 'Аня', 'Леонова', 0, 0);
22 insert into PERSONPAGE(PERSON_NUMBER, NICKNAME, NAME_,
23   SURNAME, ISMODERATOR, ISGUST)
24   values(2, 'kiss', 'Алексей', 'Киселев', 0, 1);
25
26 -- Картинки:
```

```

27
28 insert into PERSONPHOTO_1( PHOTO_ID, PERSON_NUMBER, Filename)
29 values(0, 2, '/home/site_media/img/person/swiborg10.png');
30 insert into PERSONPHOTO_1( PHOTO_ID, PERSON_NUMBER, Filename)
31 values(1, 1, '/home/site_media/img/person/swiborg11.png');
32 insert into PERSONPHOTO_1( PHOTO_ID, PERSON_NUMBER, Filename)
33 values(2, 0, '/home/site_media/img/person/swiborg12.png');
34
35 -- Статьи:
36
37 insert into ARTICLE( NAME_ ,NUMBER_OF_ARTICLE,
38 DATE_ , PERSON_NUMBER, TEXT, SECURE)
39 values(-1,
40 to_date('0001/01/01:01:01:01AM',
41 'yyyy/mm/dd:hh:mi:ssam'),
42 1, 'это_текст_статьи_1', 0 );
43
44 insert into ARTICLE( NAME_ ,NUMBER_OF_ARTICLE,
45 DATE_ , PERSON_NUMBER, TEXT, SECURE)
46 values('Статья_0',0,
47 to_date('1998/05/31:12:00:00AM',
48 'yyyy/mm/dd:hh:mi:ssam'),
49 1, 'это_текст_статьи_1', 0 );
50
51 insert into ARTICLE(NAME_ , NUMBER_OF_ARTICLE,
52 DATE_ , PERSON_NUMBER, TEXT, SECURE)
53 values('Статья_2', 2,
54 to_date('1998/05/31:12:00:00AM',
55 'yyyy/mm/dd:hh:mi:ssam'),
56 0, 'для_самокоммента', 0 );
57
58 insert into ARTICLE(NAME_ , NUMBER_OF_ARTICLE,
59 DATE_ , PERSON_NUMBER, TEXT, SECURE)
60 values('Статья_1',1,
61 to_date('1998/05/31:12:00:00AM',
62 'yyyy/mm/dd:hh:mi:ssam'),
63 1, 'это_текст_статьи_2', 1 );
64
65 -- Личный сообщения:
66
67 insert into PERSONMESSAGE(NUMBER_OF_MESS, PERSON_NUMBER,

```

```

68     TEXT, FROMTO, DATE_)
69     values(0, 0, 'у_нас_есть_Алексей', 1,
70         to_date('1998/05/31:12:00:00AM',
71             'yyyy/mm/dd:hh:mi:ssam') );
72
73     insert into PERSONMESSAGE(NUMBER_OF_MESS, PERSON_NUMBER,
74         TEXT, FROMTO, DATE_)
75     values(1, 1, 'Алексей!', 1,
76         to_date('1998/05/31:12:00:00AM',
77             'yyyy/mm/dd:hh:mi:ssam') );
78
79 -- Конкурсы:
80
81     insert into CONCURS( CONCURS_NUMBER, TEXT, NAME_, SUBJ)
82     values(-1, 'Empty',
83         'Empty', 'Empty');
84
85     insert into CONCURS(CONCURS_NUMBER, TEXT, NAME_, SUBJ)
86     values(1, 'это_текст_конкурса',
87         'это_имя_конкурса', 'это_тема_конкурса');
88
89     insert into CONCURS(CONCURS_NUMBER, TEXT, NAME_, SUBJ)
90     values(2, 'aaaaaaaaaaa!!!!',
91         'Допиши_сайт', 'кто_допишет_наконец_этот_сайт');
92
93 -- Участие:
94
95     insert into PERSONPAGE_CONCURS(
96         PERSON_NUMBER, CONCURS_NUMBER)
97     values(0, 2);
98
99     insert into PERSONPAGE_CONCURS(
100         PERSON_NUMBER, CONCURS_NUMBER)
101     values(0, -1);
102
103     insert into PERSONPAGE_CONCURS(
104         PERSON_NUMBER, CONCURS_NUMBER)
105     values(1, 1);
106
107 -- Комментарии:
108

```

```

109  insert into PERSONCOMMENT(
110      NUMBER_OF_COMMENT, PERSON_NUMBER,
111      NUMBER_OF_ARTICLE, CONCURS_NUMBER, TEXT)
112      values(10, 0, 1, -1, 'текст_самокоммента_для_статьи');
113
114  insert into PERSONCOMMENT(
115      NUMBER_OF_COMMENT, PERSON_NUMBER,
116      NUMBER_OF_ARTICLE, CONCURS_NUMBER, TEXT)
117      values(0, 0, 1, -1, 'текст_комментария_для_статьи_--_самокоммента');
118
119  insert into PERSONCOMMENT(
120      NUMBER_OF_COMMENT, PERSON_NUMBER,
121      NUMBER_OF_ARTICLE, CONCURS_NUMBER, TEXT)
122      values(1, 0, -1, 1, 'текст_комментария_для_конкурса');
123
124  -- ЛЮДИ :
125
126  -- STREET_USER:
127
128  insert into STREET_USER( USER_NUMBER, NAME_,
129      NICKNAME, Surname)
130      values(0, 'Иван', 'Странник', 'Факов');
131
132  -- USERCOMMENT:
133
134  insert into USERCOMMENT( USER_COMMENT_NUMBER, USER_NUMBER,
135      NUMBER_OF_ARTICLE, CONCURS_NUMBER, TEXT)
136      values(0, 0, 1, -1, 'комментарий_юзера_к_статье');
137
138  insert into USERCOMMENT( USER_COMMENT_NUMBER, USER_NUMBER,
139      NUMBER_OF_ARTICLE, CONCURS_NUMBER, TEXT)
140      values(1, 0, -1, 1, 'комментарий_юзера_к_конкурсу');
141
142  commit;

```

Этот скрипт не выводит ничего полезного. Результатом его выполнения являются созданные в базе записи.

6 Запросы

Покажем, что созданная база успешно, работает. Кроме того напишем какой-то реально полезный скрипт. Ниже приведённые скрипты не планировались изначально, но в рамках лабораторной работы позволяют закрепить навыки SQL. Вопрос применимости их в реальной жизни — открыт, но вряд ли ими буду часто пользоваться. Однако, эти запросы весьма интересны.

1 Само-комментарии

Найдём все само-комментарии своих статей. Это может, оказаться полезным для статистики. Интересно, что так легко найти людей, болтающих самих с собой.

```
1 select distinct
2     personpage_.NICKNAME as "Имя_пользователя",
3     article_.TEXT as "Текст_статьи",
4     comment_.TEXT as "Текст_комментария"
5 from
6     PERSONCOMMENT comment_,
7     ARTICLE article_
8 join PERSONPAGE personpage_ on personpage_.PERSON_NUMBER =
9     article_.PERSON_NUMBER
10 where (comment_.CONCURS_NUMBER = -1 ) and
11        (article_.NUMBER_OF_ARTICLE != -1 ) and
12        (comment_.PERSON_NUMBER = article_.PERSON_NUMBER );
```

Ответ сервера (через sqlplus):

Имя пользователя

Текст статьи

Текст комментария

w495

для самокоммента

текст комментария для статьи -- самокоммента

w495

для самокоммента

текст самокоммента для статьи

2 Самые обсуждаемые статьи

Напишем запрос, который позволит найти наиболее обсуждаемые статьи в сообществе журналистов.

```
1 select
2   article_.NAME_ as "Название_статьи",
3   cc as "Комменты_от_журналистов" from ARTICLE article_,
4   (
5     select
6       article_.number_of_article na,
7       count(comment_.number_of_comment) cc
8     from
9       PERSONCOMMENT comment_,
10      ARTICLE article_
11    where
12      (comment_.CONCURS_NUMBER = -1 ) and
13      -- Заметим: что CONCURS_NUMBER = -1
14      -- исключение и его использование не желательно
15      (article_.NUMBER_OF_ARTICLE != -1 ) and
16      (article_.number_of_article = comment_.number_of_article )
17
18    Group by comment_.number_of_article, article_.number_of_article
19  )
20  where na = article_.number_of_article
```

Ответ сервера (через sqlplus):

Название статьи

Комменты от пользователей

Статья 1

2

Не трудно убедиться по листингам, что ответ верный

7 Процедуры. Пользовательский модуль.

По заданию лабораторных работ нужно описать несколько хранимых процедур. Но мы поступим иначе. Мы опишем пакет. Это намного удобнее и проще. Кроме того, при реализации «боевого варианта» это избавит, по крайней мере, от пересечения имён. Хотя с точки зрения экономии памяти пакеты — плохи.

Опишем заголовок пакета.

```
1 create or replace package
2 /**
3     Пакет пользовательских процедур
4 **/
5 User_Package is
6     /*
7         Процедуры:
8     */
9     procedure add_user (
10         a_string_NICK varchar2,
11         a_string_NAME varchar2,
12         a_string_Surname varchar2 );
13     procedure internal_add_user (
14         a_string_NICK varchar2,
15         a_string_NAME varchar2,
16         a_string_Surname varchar2 );
17 end User_Package;
```

Пакет содержит всего лишь две процедуры, но в рамках лабораторных работ больше и не требуется. Тем более, остальные процедуры можно создать автоматизированно.

Опишем тело пакета.

```
1 create or replace package body User_Package
2 is
3     cnt integer;
4     NICKNAME_fail exception;
5     NAME_fail exception;
6     SURNAME_fail exception;
7     /*
8         Процедуры:
```

```

9  */
10 procedure add_user (
11     a_string_NICK varchar2,
12     a_string_NAME varchar2,
13     a_string_Surname varchar2 ) as
14  /*
15     Добавляет пользователя—журналиста.
16  */
17  begin
18     internal_add_user (a_string_NICK, a_string_NAME, a_string_Surname );
19
20     exception
21         when NICKNAME_fail then dbms_output.put_line('Ошибочный_ник');
22         when NAME_fail then dbms_output.put_line('Ошибочное_имя');
23         when SURNAME_fail then dbms_output.put_line('Ошибочная_фамилия');
24         when OTHERS then dbms_output.put_line('Ошибка_--X!'); rollback;
25 end;
26
27 procedure internal_add_user (
28     a_string_NICK varchar2,
29     a_string_NAME varchar2,
30     a_string_Surname varchar2 ) as
31  /*
32     Добавляет пользователя—журналиста. Низкоуровнево.
33  */
34     user_id integer;
35  begin
36     cnt:= cnt + 1;
37     if (a_string_NICK is null) then
38         raise NICKNAME_fail;
39     end if;
40     if (a_string_NAME is null) then
41         raise NAME_fail;
42     end if;
43     if (a_string_Surname is null) then
44         raise SURNAME_fail;
45     end if;
46     insert into PERSONPAGE(PERSON_NUMBER, NICKNAME,
47         NAME_,Surname, ISMODERATOR, ISGUST)
48     values(cnt, a_string_NICK ,
49         a_string_NAME, a_string_Surname, 0, 0);

```

```

50     commit;
51 end;
52 begin
53     cnt:= 16;
54     -- это не самый лучший способ задавать id пользователей
55     -- но так, проще
56     -- первый 16 номеров являются тестовыми
57 end User_Package;

```

Данный пример наглядно показывает как можно работать с исключениями.

Ответ сервера (через sqlplus):

```
SQL> call User_Package.add_user('vasp', 'Василий', 'Пупкин');
```

Вызов завершен.

```
SQL> select c.NAME_ from PERSONPAGE c where c.NICKNAME = 'vasp';
NAME_
-----
```

Василий

```
SQL> select c.PERSON_NUMBER,
           2 c.NAME_ from PERSONPAGE c where c.NICKNAME = 'vasp';
PERSON_NUMBER
-----
```

NAME_

17

Василий

```
SQL> call User_Package.add_user('', 'Василий', 'Пупкин2');
```

Ошибочный ник

Вызов завершен.

8 Триггеры

В принципе в данном случае, нет необходимости в триггерах. Все функции по контролю вставки может взять на себя пользовательский модуль. Но на всякий случай, мы опишем один важный триггер.

```
1 create or replace trigger PERSONPAGE_T
2 before insert on PERSONPAGE
3 for each row
4 /*
5     Специальный триггер для особо одарённых пользователей
6 */
7 declare
8     NICKNAME_fail exception;
9     NAME_fail exception;
10    SURNAME_fail exception;
11 begin
12     if (:new.NICKNAME is null) then
13         dbms_output.put_line('Пустой_ник');
14         raise NICKNAME_fail;
15     end if;
16
17     if (:new.NAME_ is null) then
18         dbms_output.put_line('Пустое_имя');
19         raise NAME_fail;
20     end if;
21     if (:new.SURNAME is null) then
22         dbms_output.put_line('Пустая_фамилия');
23         raise SURNAME_fail;
24     end if;
25 end;
```

Ответ сервера (через sqlplus):

```
SQL> insert into PERSONPAGE
(PERSON_NUMBER, NICKNAME, NAME_,Surname, ISMODERATOR, ISGUST)
values(666, '' , 'Василий', 'Пупкин', 0, 0);
Пустой ник
insert into PERSONPAGE
(PERSON_NUMBER, NICKNAME, NAME_,Surname, ISMODERATOR, ISGUST)
values(666, '' , 'Василий', 'Пупкин', 0, 0)
*
ошибка в строке 1:
ORA-06510: PL/SQL: необработанное исключение, определенное пользователем
ORA-06512: на "W495.PERSONPAGE_T", line 8
ORA-04088: ошибка во время выполнения триггера 'W495.PERSONPAGE_T'
```

Часть IV

Словарь терминов

Веб-сайт (от англ. website: web - «паутина», «сеть» и site — «место», букв. «место в сети») или просто сайт – в компьютерной сети объединённая под одним адресом (доменным именем или IP-адресом) совокупность документов частного лица или организации. По умолчанию подразумевается, что сайт располагается в сети Интернет. Все веб-сайты Интернета в совокупности составляют Всемирную паутину. Для прямого доступа клиентов к веб-сайтам на серверах был специально разработан протокол HTTP. Веб-сайты иначе называют Интернет-представительством человека или организации. Когда говорят «своя страничка в Интернет», то подразумевается целый веб-сайт или личная страница в составе чужого сайта. Кроме веб-сайтов в сети Интернет так же доступны WAP-сайты для мобильных телефонов.

Веб-форум – класс веб-приложений для организации общения посетителей веб-сайта. Термин соответствует смыслу исходного понятия «форум».

Гость сайта – лицо, специально приглашённое в социальную сеть, для обсуждения какого-либо вопроса, но не являющееся полноценным участником социальной сети. Такое лицо не может создавать, просматривать, комментировать, закрывать статьи и комментарии к ним, а также создавать блоги связанные со страницей участника.

Комментарий — сообщение, которое является реакцией на некоторую статью (или комментарием в общем смысле), конкурс или другой комментарий. В данном случае комментарии бывают двух типов:

- к статье
- к конкурсу

Конкурс (от латинского concursus) — соревнование, соискательство нескольких лиц в области искусства, наук, спорта и прочего, с целью выделить наиболее выдающегося (или выдающихся) конкурсанта-претендента на победу.

В данной социальной этот термин обозначает – примерно, то же самое и что и «группы» в большинстве остальных, но учитывая контекст, они являются своеобразными конкурсами для журналистов. В общем случае конкурс имеет:

- Список участников.
- Последовательность комментариев.

Остальные атрибуты пока в процессе проектирования.

Концептуальная модель (англ. conceptual model) – это определённое множество понятий и связей между ними, являющихся смысловой структурой рассматриваемой предметной области. Концептуальная модель – модель предметной области, состоящей из перечня взаимосвязанных понятий, используемых для описания этой области, вместе со свойствами и характеристиками, классификацией этих понятий, по типам, ситуациям, признакам в данной области и законов протекания процессов в ней. (Толковый словарь по искусственному интеллекту)

Контент (англ. content — содержимое) – любое информационно значимое наполнение информационного ресурса (например, веб-сайта) – тексты, графика, мультимедиа – вся информация, которую пользователь может загрузить на диск компьютера с соблюдением соответствующих законностей, как правило, только для личного пользования.

Личное сообщение – некоторое, чаще всего текстовое сообщение, от одного участника социальной сети, другому.

Степень конфиденциальности достаточно высокая – модератор не имеет право просматривать содержание сообщений других пользователей.

Личная страничка – страница, формально принадлежащая одному из участников социальной сети, которая содержит (*в идеале*):

- **Ф.И.О.**
- Фотография
- **Страна, Город (область, район)**
- Дата рождения
- **Цель регистрации**
- Список статей
- Участие в конкурсах (только для не гостей)
- Род занятий
 - Вуз, факультет
 - Место работы
 - Специализация
- Достижения
 - Журналистская практика

- Научная практика
- Проекты
- Награды
- «О себе» – текстовая информация произвольного содержания
- Связь
 - Написать личное сообщение
 - Телефон
 - Факс
 - E-mail
 - IM (с указанием типа/протокола)
 - Почтовый адрес
 - Ссылка на личный блог/сайт

Социальная сеть – интерактивный многопользовательский веб-сайт, контент которого наполняется самими участниками сети. Социальная сеть направлена на построение сообществ в Интернете из людей со схожими интересами и/или деятельностью. Связь осуществляется посредством сервиса внутренней почты или мгновенного обмена сообщениями.

Статья – это произведение журналистики, в котором автор ставит задачу проанализировать общественные ситуации, процессы, явления прежде всего с точки зрения закономерностей, лежащих в их основе. Размер статей должен быть субъективно не большим.

В данной социальной этот термин обозначает – примерно, то же самое и что и «заметки» в большинстве остальных, но учитывая контекст, такие текстовые блоки, все таки являются статьями в общем смысле.

Выводы

В результате проведённых лабораторных работ были закреплены знания SQL, получен опыт работы с СУБД Oracle и программирования на диалекте PL/SQL.

Я считаю, что самое главное в этих работах — то что мы должны были сами выбирать предметную область проектировать системы. Последнее, оказалось нетривиальной задачей, тем более, что здесь не существует готовых решений, а ошибки на этапе проектирования могут понести за собой тяжёлые последствия.