Курсовая работа

I. Введение

В качестве курсовой работы выбрана разработка базы данных, которая могла бы обеспечить работу портала PubChem (https://pubchem.ncbi.nlm.nih.gov).

PubChem является открытой базой данных национального института здоровья (США). Открытость в данном случае обозначает, что каждый может загрузить на портал результаты своих научных работ, и каждый посетитель портала может их использовать.

Разрабатываемая база данных призвана решить следующие задачи:

- Сохранение учетных записей пользователей (таблица **users**) и компаний (таблица **companies**);
- Сохранение результатов научных работ, предоставленных пользователями (таблица **articles**);
- Группировка результатов научных работ для проведения необходимых проверок и упрощения процесса поиска (таблица **partitions**);
- Предоставление пользователям сервисов, аналогичных тем, что представлены на портале PubChem:
  - использование списков выбранных статей для повторного обращения (таблица **favourites**),
  - составление библиографических списков статей для собственных работ (таблица **bibliography**),
  - возможность перелинковки статей между собой (таблица **references_list**);
- Также предусмотрены:
  - возможность прикрепления файлов к текстам статей и постов (таблица **media**)
  - и ведение блогов, необходимых для публикации новостей, анонсов статей и планируемых исследований (таблица **posts**).

В базе данных предусмотрены вспомогательные таблицы:

- **category** (категория компаний: например, лаборатория, журнал или университет);
- **source_types** – типы материалов (пост или статья), с которыми связаны столбцы source_id (в таблицах favourites, bibliography, references_list) и id (таблицы posts и articles);
- **specialization** – специализация, в которой работает автор.
- отдельно необходимо сохранять список авторов в архивной таблице **authors**, т.к. не обязательно каждый автор является пользователем портала; так, например, журнал от лица пользователя – главного редактора – может опубликовать статью научного работника какого-нибудь университета, никогда не регистрировавшегося на портале PubChem.
- Для целей данного проекта также создана таблица для копирования данных удаляемых пользовтелей (archive_user).

Отдельный триггер призван проверять отсутствие одинаковых записей в таблице authors в т.ч. при случайной смене порядка написания имени и фамилии автора; дополнительной проверкой является проверка совпадений по специальности и стране автора.

Используя созданный View Authors_jobs можно выводить список всех работ данного автора и проверять наличие аналогичных его работ по заданной тематике.

II. Связи таблиц в базе данных

Таблица.столбец (ключ)	Связи с Таблицей.столбцом (зависимые)
patitions.id	media.partitions_id
users.id	media.user_id favourites.user_id bibliography.user_id referencies_list posts.user_id articles.user_id
category.id	companies.category_id
companies.id	users.company_id
referencies_list.id	posts.referencies_id articles.references_id
source_types.id	favourites.source_type_id bibliogarphy.source_type_id referencies_list.source_type_id
authors.id	posts.author_id articles.author_id
specialization.id	authors.spec_id

III. Описание таблиц

- partitions – таблица содержит название, описание и ссылку на каждый раздел, в соответствии с которым и хранятся медиа-данные;
- users – логин, имя и фамилию пользователя, email, дату и время регистрации, дату рождения, название компании (если имеет отношение к какой-либо компании из тех, что есть в базе данных); наличие аккаунта в базе данных дает возможность хранить в профиле данные о понравившихся (отложенных) статьях в разделе favorites, а также составлять библиографический список bibliography и делать ссылки references на другие статьи со своей статьи.
- category – категория (название), к которой относится компания; отношение к той или иной категории имеет информационное значение, а также может быть полезно, например, при выполнении рассылок.
- companies – имя, ссылку на страницу с данными, id категории из таблицы category, имя контактного лица, адрес компании, дату обновления информации, телефон; эти данные используются для составления информационной страницы о каждой компании;
- media – имя файла, размер файла, id раздела из таблицы partitions, дату обновления файла, имя пользователя (владельца файла); media-данные сопровождают каждое исследование; деление на разделы происходит в соответствии с правилами раздела и содержанием исследования;
- favourites – имя пользователя, к которому относятся выбранные материалы, id типа материала из таблицы source_types, id материала из соответствующей типу материала таблицы; эта таблица соответствует списку отложенных или понравившихся пользователю статей;
- bibliography - имя пользователя, к которому относятся выбранные материалы, название работы, id типа материала из таблицы source_types, id материала из соответствующей типу материала таблицы; эта таблица соответствует библиографическому списку статей, необходимому каждому исследованию;
- references_list – имя пользователя, название статьи (на которую дается ссылка), id типа материала из таблицы source_types, ссылка, дата публикации, id материала из соответствующей типу материала таблицы; данная таблица соответсвует списку связанных материалов на портале и позволяет дать не только ссылку, но и справочную информацию о ней;
- source_types – название типа материала; основные типы: посты и статьи, которым соответствует отдельные таблицы;
- posts – имя пользователя (автор поста), название поста, описание опубликованной информации, ссылка, id автора опубликованной информации из таблицы author (может не совпадать с автором поста), дата создания, id из таблицы references_list для вывода информации о связанных статьях;
- articles – название статьи, описание статьи, автор статьи, дата публикации, дата обновления, ссылка, id из таблицы references_list для вывода информации о связанных статьях, id пользователя (того, кто опубликовал статью; может не совпадать с автором статьи);
- authors – имя, фамилия, страна, специализация автора;
- specialization – название основной специализации для каждого автора;
- archive_user – по структуре – копия таблицы users.

IV. Описание табличных данных

- Таблица partititons – все значения в каждой строке должны быть заполнены (не могут быть нулевыми)
  - id тип данных tinyint (разделов не может быть очень много), первичный ключ, авто увеличение на 1, не может быть нулевым;
  - name текстовая строка длиной до 50 знаков, не может бтыь нулевой;
  - description текстовая строка длиной до 255 знаков, не может быть нулевой;
  - href текстовая строка длиной до 255 знаков, не может быть нулевой.
- Таблица users
  - id тип serial (данных может быть много; пользователи могут удаляться, а предоставленные ими данные нужно сохранять, если они не потребовали их удалить),
  - login текстовая строка длиной до 20 знаков, не может быть нулевой;
  - first_name текстовая строка длиной до 50 знаков, не может быть нулевой;,
  - last_name текстовая строка длиной до 50 знаков, не может быть нулевой;
  - email текстовая строка длиной до 125 знаков, не может быть нулевой;
  - registred_at тип данных datetime, значение не может быть нулевым,
  - birthday тип данных datetime, значение не может быть нулевым,
  - company_id тип данных int, значение не может быть отрицательным.
- Таблица category
  - id тип int (их может быть много, но bigint представляется излишним), id всегда будет больше нуля и при этом должен быть заполнен, автоматически увеличиваться на 1 при каждой следующей записи, является первичным ключом.
  - category_name - текстовая строка длиной до 50 знаков, не может быть нулевой;
- Таблица companies
  - id тип int (их может быть много, но bigint представляется излишним), id всегда будет больше нуля и при этом должен быть заполнен, автоматически увеличиваться на 1 при каждой следующей записи, является первичным ключом,
  - name текстовая строка длиной до 100 знаков, не может быть нулевой,
  - href текстовая строка длиной до 255 знаков, не может быть нулевой,
  - category_id тип int, как в таблице category,
  - contact_Name текстовая строка длиной до 255 знаков, не может быть нулевой,
  - Address текстовая строка длиной до 255 знаков, не может быть нулевой,
  - updated_at тип date – важна только дата обновления информации,
  - phone тип int(13) – 13 цифр для сохранения номера телефона (с запасом для кодов городов и добавочных номеров).
- Таблица media
  - Id тип serial auto_increment primary key,
  - user_id bigint unsigned not null – как в таблице users,
  - file_name текстовая строка длиной до 25 знаков, не может быть нулевой,текстовая строка длиной до 255 знаков, не может быть нулевой,
  - file_size_mb тип float, неотрицательное ненулевое значение,
  - Partition_id тип tinyint ненулевое значение – как в таблице partitions,
  - updated_date тип date, значение не может быть нулевым;
- Таблица favourites
  - id тип serial,
  - user_id тип bigint (20 символов), не отрицательный и не нулевой,
  - source_type_id тип tinyint не отрицательный и не нулевой,
  - source_id тип bigint (20 символов), неотрицательный и не нулевой
- Таблица bibliography

    - id тип serial,
    - user_id тип bigint (20 символов), не отрицательный и не нулевой,
    - survey_name текстовая строка длиной 10 символов, не может быть нулевой,
    - source_type_id тип tinyint, не отрицательный и не нулевой,
    - source_id тип bigint (20 символов), не отрицательный и не нулевой;
- Таблица references_list,
    - id тип serial,
    - user_id тип bigint, не отрицательный и не нулевой,
    - name текстовая строка длиной до 255 символов, не может быть нулевой,
    - source_id тип bigint, не отрицательный и не нулевой – как в таблицах posts и articles (столбцы id),
    - source_type_id тип tinyint, не может быть нулевым – как в таблице source_types,
    - href текстовая строка длиной до 512 символов, не может быть нулевой,
    - publication_date тип date, значение не может быть нулевым;
- Таблица source_types
    - id тип tinyint – как в таблице source_types,
- name – текстовая строка длиной до 10 символов.
- Таблица posts
    - id тип serial,
    - user_id тип bigint, не отрицательный, не может быть нулевым,
    - topic текстовая строка длиной до 255 символов, не может быть нулевым,
    - description текстовая строка длиной до 1000 символов, значение по умолчанию – пустая строка (описание могут создать отдельно, м.б. позже);
    - href текстовая строка длиной до 128 символов, не может быть нулевой;
    - authors_id тип bigint – как в таблице authors, не может быть нулевым;
    - created_at тип date, значение не может быть нулевым;
    - referencies_id тип bigint, не отрицательный и не нулевым;
- Таблица articles
    - id тип serial,
    - topic – текстовая строка длиной до 255 символов,
    - description – текстовая строка длиной до 1000 символов,
    - author_id – тип bigint – как в таблице authors,
    - created_at – тип date, значение не может быть нулевым,
    - updated_at – тип date, значение не может быть нулевым,
    - href – текстовая строка длиной до 128 символов, не может быть нулевой;
    - references_id - тип bigint, не отрицательный и не нулевым – как в таблице references_list;
    - user_id тип bigint до 20 символов, не нулевой и не отрицательный;
- Таблица authors
    - id bigint, не отрицательный и не нулевой,
    - first_name – текстовая строка длиной до 50 символов, не может быть нулевой,
    - last_name – текстовая строка длиной до 50 символов, значение по умолчанию - ‘’,
    - country - текстовая строка длиной до 20 символов, не может быть нулевой,
    - spec_id – тип int, значение не может быть нулевым.
- Таблица specialization
    - id тип int, автоувеличение на 1, primary key,
    - spec – текстовая строка длиной до 50 символов, не может быть нулевой;
- Таблица archive_user является **архивной** копией удаленных данных таблицы users.

V. Внешние ключи.

Для поддержания связей между таблицами без участия пользователя были созданы внешние ключи. Было принято решение сохранять все значения в связанных таблицах при удалении внешних ключей, кроме случаев использования личных предпочтений пользователей (таблицы favourites и bibliography), т.к. в остальных случаях данные могут требоваться другим пользователям.

Список внешних ключей базы даных

articles_author_id_idx
articles_created_at_idx
articles_description_idx
articles_href_idx
articles_references_id_idx
articles_topic_idx
articles_updated_at_idx
articles_user_id_idx
companies_href_idx
companies_name_idx
posts_author_id_idx
posts_created_at_idx
posts_description_idx
posts_href_idx
posts_references_id_idx
posts_topic_idx
posts_user_id_idx
references_list_href_idx
references_list_name_idx
references_list_publication_date_idx
references_list_source_id_idx
references_list_source_type_id_idx
references_list_user_id_idx
media_file_name_idx
media_file_size_mb_idx
users_company_id_idx
users_email_idx
users_first_name_idx
users_last_name_idx

VI. Построение индексов

1. Часто используемые таблицы (прогноз):
    1. users;
    2. companies;
    3. posts;
    4. articles;
    5. bibliography;
    6. favourites;
    7. references_list.
    8. media
2. Часто используемые столбцы в часто используемых таблицах:
    1. users.first_name, users.last_name, users.company_id, users.email – все сравнительно редко изменяемые;
    2. companies.name, companies.href – все сравнительно редко изменяемые;
    3. posts – все, при этом они сравнительно редко изменяются после добавления;
    4. articles – все, при этом сравнительно редко изменяются после добавления;
    5. favourites.source_type_id, favourites.source_id – могут часто изменяться при активной работе;
    6. bibliography.survey_name, bibliography.source_type_id, bibliography.source_id – могут часто изменяться при активной работе;
    7. references_list – все, при этом сравнительно редко изменяются;
    8. media.file_name, media.file_size_mb – сравнительно редко изменяются.

Простматриваются возможности создания составных индексов:

- favourites.source_type_id + favourites.source_id – только вместе они указывают на конкретный id в таблице материалов posts или articles;
- bibliography.source_type_id, bibliography.source_id – только вместе они указывают на конкретный id в таблице материалов posts или articles;
- references_list.source_type_id, references_list.source_id – только вместе они указывают на конкретный id в таблице материалов posts или articles.

Для сравнительно небольшой базы данных, принято решение построить индексы для наиболее часто используемых записей, которые обновляются не часто, а также применить составные индексы. В реальной работе эффективность применения индексов может быть оценена отдельно.