2DB3 Assignment 1

MacID : lahira2 , Student Number : 400480017 , Name : Aditya Lahiri

~~A group of friends want to start a website focused on festivals and other events, this similar to how IMDB is a website focused on films. Central in this website will be the events, their details, and user-generated content (e.g., discussions and reviews)~~. For each event, the website should maintain basic information such that users can search for specific events. For each event, we keep a name, description, start date and time and end date and time, and location (street, number, city, province/state, country). Due to recent experiences, not all events have a location, however: the website will also support virtual events for which a link and instructions are provided to connect(for virtual event type) (e.g., a link to a streaming service or a registration link for accessing a video conference). Events can be part of a series (e.g., yearly iterations of a festival). For these events, we store the series they belong to and the edition. ~~For example, the Toronto International Film Festival is held yearly and the edition of 2024 will be the 49th edition~~. Events can have one-or-more activities (~~e.g., performances on a music festival, screenings on a film festival~~). For each activity, there is a begin time and date and duration. Optionally, the activity can be tagged with additional location information (~~e.g., Main Stage for a music festival with concurrent performances; or Room 5 for a film festival held in a cinema with 7 rooms~~). Per activity, the website can list the participants (e.g., artists). For each such participant, their role is maintained (e.g., performing artist or organizer). Participants can participate in multiple ways on a single event. ~~Consider an tournament for the utmost competitive game "Minesweeper". This tournament event will be called "Ultimate Minesweeper Competition 2024" and is the 17th edition of this event. The event consists of the following main activities~~:

1. ~~Before the finale, there are online qualification rounds~~.
2. ~~Starting Friday May 5th, the main event will start in a massive arena in Ontario. During the main event, there will be a three-day final competition. On Friday, 4 brackets of 4 players will compete to determine the best two players per group that will advance to the quarterfinals. For this day, there will be four activities~~ (one for each bracket). The participants for each activity will be the players of that bracket and the commentators for that bracket.
3. On Saturday, there will be four quarterfinals (8 players, 4 players advance to the semifinals). For this day, there will be four activities (one for each quarterfinals). The participants for each activity will be the two players of that quarterfinal and the commentators for that quarterfinal.
4. On Sunday, there will be three activities: the two semifinals and the finals. ~~The finals will have a halftime show during which famous artist will perform (to further add to the spectacle). The participants for the finals will be the artist, the two finalists, and the commentators for the finals~~.

Per participant, we maintain the name and a small biography. Given a participant, ~~one should be able to figure out the activities (and the events those activities belong to)~~ that participant partook in. For user-generated content, the website will keep track of users that have a unique username, a unique e-mail address, and a password. Users ~~can log in using either their~~ username or their e-mail address. Users can rate events with a score (between zero and ten). ~~Users can change the rating and the website only keeps track of the last rating provided by the user~~. Finally, users can write text contributions. These text contributions are either long-

form reviews of events, reactions on other contributions, or public messages ~~users place on their public profile. These text contributions can be searched without knowing the exact type of the contribution (e.g., one can search for any user-generated content containing the word "Minesweeper").~~ For each text contribution, we store the publication date and the main textual content. In addition, both long-form reviews and public messages have titles and can be placed in user-defined categories (e.g., hobby, work, or 2022). Other users can like and dislike each text contributions once.

We have derived an ER-Diagram that can represent all information present in the description. The resultant ER-Diagram can be found in Figure 1. Next, we provide a high-level overview of the ER-Diagram, discuss the choices made, and present all constraints.

Each event is represented by an entity **Event** with attributes name (of type TEXT), description (of type TEXT) , start_time (of type TIMESTAMP) , end_time (of type TIMESTAMP) .Since there can be incidences wherein event names could possibly be the same, it is more reliable to use an automatically generated identifiers eid (of type INTEGER), as primary keys. Since events can be held virtually too due to recent experiences, we introduce an ISA-hierarchy with root **Event** and specializations **Physical Event** and **Virtual Event** with addition of five simple attribute attributes , 4 of which are of type TEXT namely street , city , state and country and one of type INTEGER called number, this allows for accurate storage of location allowing for categorization later for **Physical Event** and addition of two simple attributes link (of type TEXT) and instruction (of type TEXT) for **Virtual Event.**
Since series can't exist without an **Event** we model a weak entity **Series**. Since **Event** can be part of series we create an identifying relationship Is_Series to **Event** and with the partial key being an automatically generated identifier sid since edition is not unique enough as a key

Each event can have one or more activities hence we model a weak entity **Activity** with an identifying one or more relationship Activity_Of to **Event** and with partial key aid (of type INTEGER) which is also an automatically generated identifier used because begin_time and duration are not uniquely identifiable keys. Each **Activity** has a begin_time (of type TIMESTAMP), duration (of type DECIMAL , this allows for taking minutes into account upto a certain degree) and location (of type TEXT) which is optional and defaults to a null value if empty, these are stored as simple attributes of weak entity **Activity** .

Each **Event** and **Activity** have participants hence we model a weak entity **Participant** with two relationships :

1. Activity_Participant to **Activity** with partial key pid (of type INTEGER)
2. Participant_Of (identifying relationship) to **Event** also with partial key pid (of type INTEGER) pid is introduced as an automatically generated identifier as no clear identifier is provided in the text and none of the other attributes can be used as unique identifying keys. The participant has a name and roles which are stored in simple attributes of the weak entity **Participant**, name (of type TEXT) and role (of type TEXT) respectively. In addition to name and role a small biography of the participant is stored in a simple attribute biography (of type TEXT).

The website also has users which are described by the entity **User** with attributes username (of type TEXT) , email (of type TEXT) and password (of type TEXT). Since it has been mentioned that the website has unique usernames and email we can choose one of those as

primary keys, in our case since users use email and passwords to log in , we have chosen email as the primary key to maintain uniformity. **User** can rate an event with a rating from 1 through 10 and we represent that information through the relation Rated_Event to **Event** with an attribute rating (of type INTEGER) to store the rating the user has provided on that event.

All users are allowed to create text contributions, and since text contributions are dependent on the user we introduce a weak entity **Text Contribution** with an identifying relationship to **User** called Written_by with partial key con_id (of type INTEGER) which is an automatically generated identifier introduced because of the lack of a clearly specified identifier. Text Contributions have other simple attributes namely publication_date(of type TIMESTAMP) and text_content(of type TEXT). These text contributions are either long-form reviews of events, reactions on other contributions, or public messages users place on their public profile, therefore to satisfy this requirement we introduce an ISA-hierarchy with root **Text Contribution** and specializations **Message** , **Review** and **Reaction**. Message and Review both have an additional simple attribute title (of type TEXT) to store the title of the contributions. We also have defined an entity category with an attribute of type TEXT called category_name as primary key, this ensures that we do not need to have attributes for each **Review** and **Message** category, it also ensures we do not have duplicate categories. Categories are not weak entities since we have assumed that empty categories can be defined without a text contribution being set to it.

We have introduced three additional relationships :

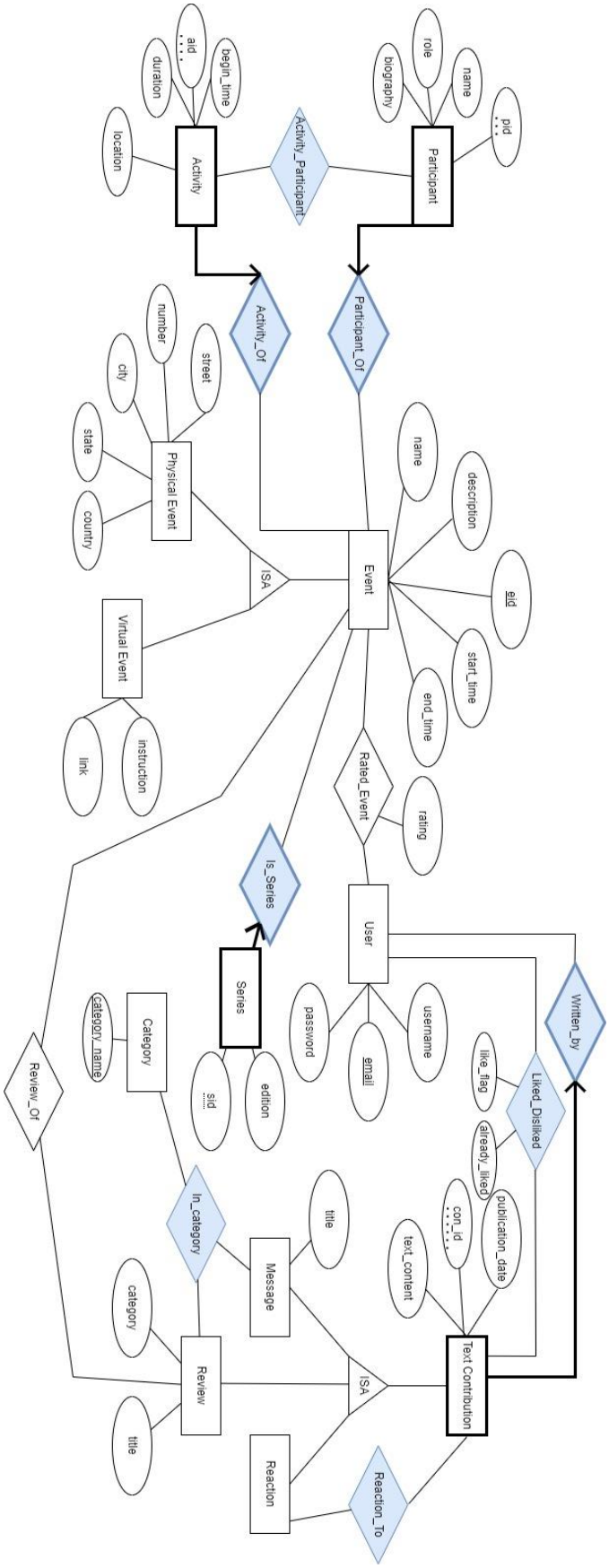
1. Review_Of to **Event** : This stores the relation between a Text Contribution of specialization Review with Event and shows if a review is for a certain event.
2. Reaction_To to **Text Contribution** : Stores the relation between a Text Contribution of specialization Reaction with any Text Contribution.
3. In_category which is a ternary relationship to **Review** and **Message** to check categories to which they would belong.

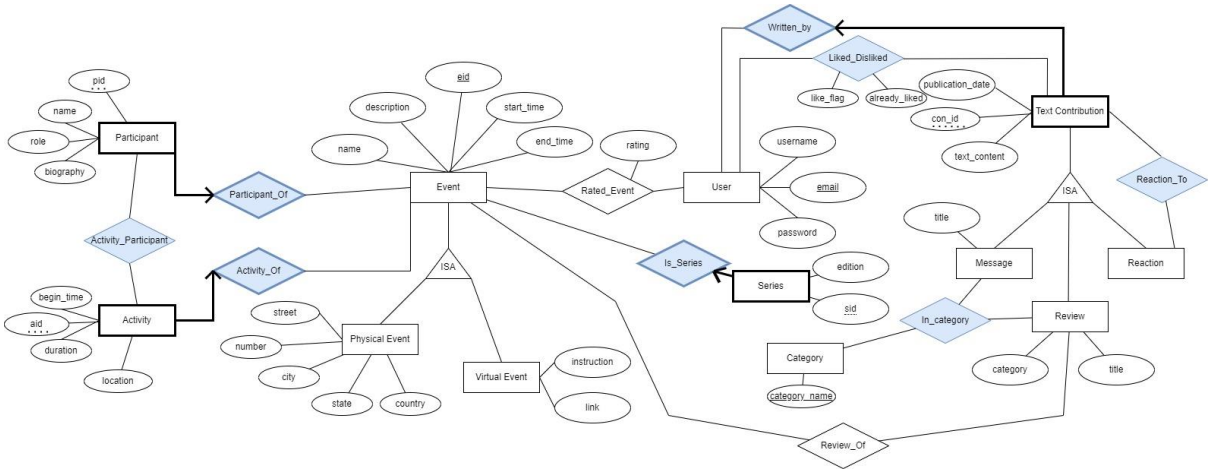
We have chosen for a minimal method that can store all the like and dislike data for a Text Contribution by a **User** using the relationship Liked_Disliked with a simple attribute like_flag (of type INTEGER) and another simple attribute already_liked (of type BOOLEAN) to store the whether the user has already liked this post.like_flag has a default initialized value specifying neither like and dislike and an appropriately chosen value for like and another for dislike which will be set according to the users interaction with the text contribution (Example : 0 for no interaction , 1 for liked , 2 for disliked).

Things not specified in the ER Diagram

- Not specified whether every participant has to take part in every activity of an event therefore we select the option that there is optional participation in activities of an event, necessitating the definition of the Activity_Participant relationship
- All start times must be before end times.
- The ER diagram does not show/check for unique username and email.
- We have not expressed the fact that users can change their ratings and the websites ability to keep track of the newest rating
- We also do not show a relation between **Message** and **User** to express that it is posted on a users profile since a relationship Written_by already exists between **Text Contribution** and **User**.

- The check to ensure that a user can't like/dislike their own Text contribution has not been shown in the ER diagram
- It has not been mentioned whether a contribution can have multiple categories we have assumed so.
- The check for whether a user has already liked a specific post is not shown.




Written_by
Liked_Disliked
like_flag
already_liked
publication_date
con_id
text_content
Text Contribution
Reaction_To
ISA
Reaction
Message
title
pid
name
role
biography
Participant
Participant_Of
Activity_Participant
Activity_Of
Activity
begin_time
aid
duration
location
eid
description
name
start_time
end_time
Event
rating
Rated_Event
User
username
email
password
Is_Series
Series
edition
sid
ISA
Physical Event
street
number
city
state
country
Virtual Event
instruction
link
In_category
Review
category
title
Category
category_name
Review_Of