Database Management System

Course Project

Project No: 1b

MockMail Database

GithubLink: https://github.com/JindalMohit/ClassroomMail

Anmol Anand (15114013)

Chirag Maheshwari (15114020)

Mohit Jindal (15114046)

Nitish Bansal (15114048)

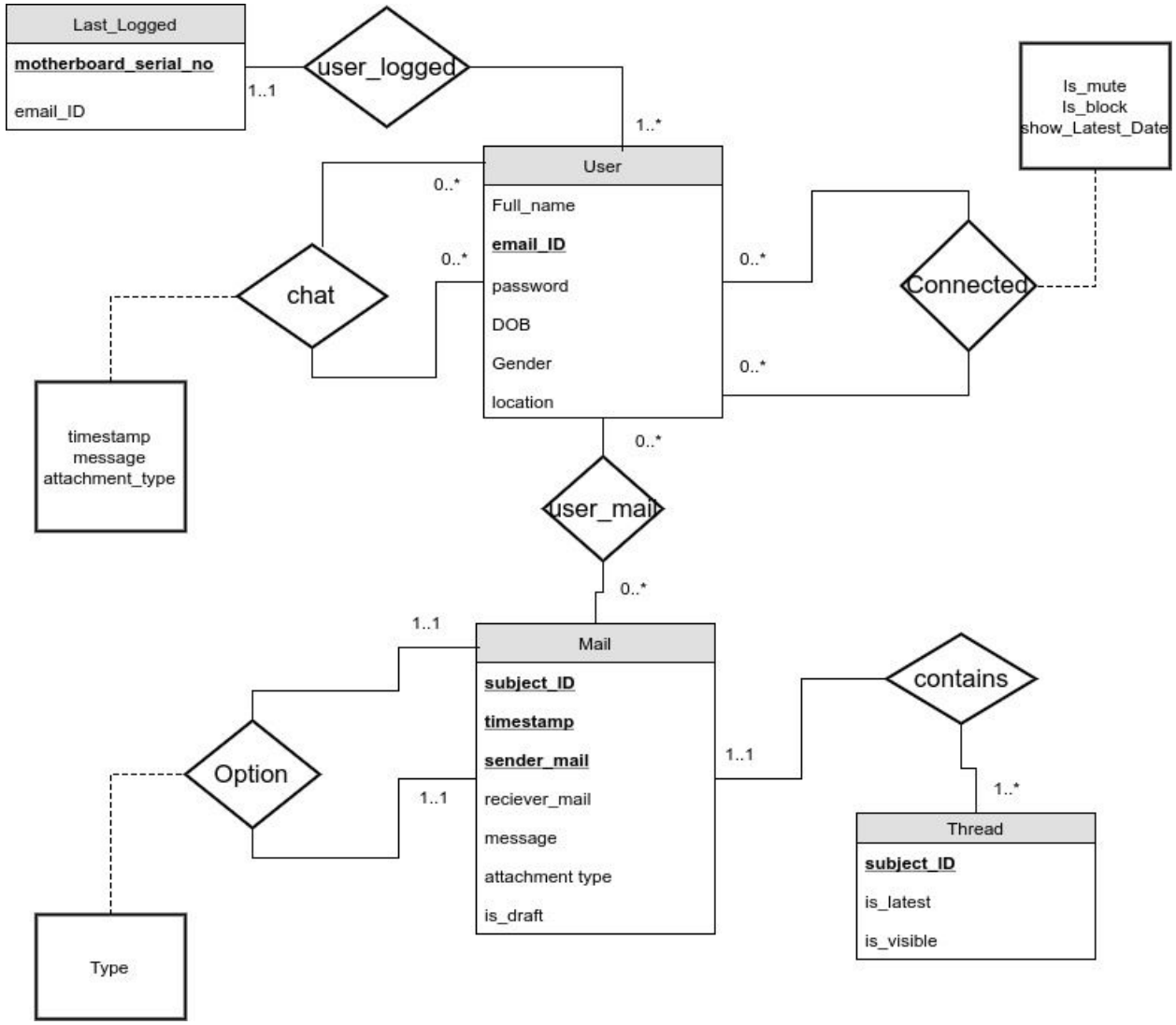
Sahil Grover (15114061)

Before we begin with the project, here are the list of ASSUMPTIONS made while implementing the project.

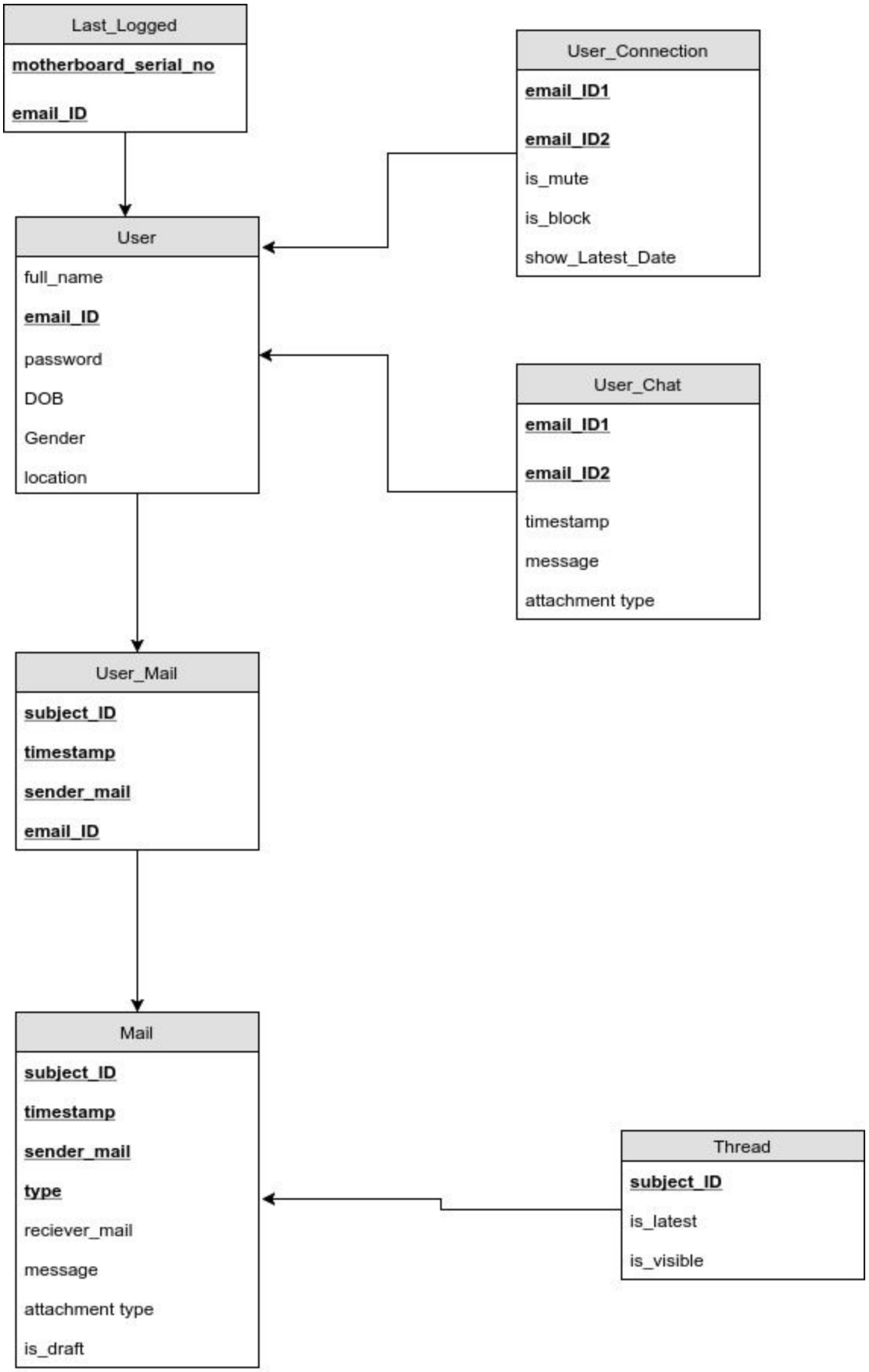
ASSUMPTIONS:

1. **A person CANNOT send multiple messages/mails at same point of time.**
2. **Motherboard serial numbers are unique**
3. **Reply mails and Forward mails are considered to be in same thread as their parent.**
4. **Each time a user views a mail, a new tuple is added with new timestamp (even if user had read that mail earlier).**

Entity Relationship Diagram (ER Diagram)



Schema Diagram



Functional Dependencies & Candidate Keys

Functional Dependencies:

1. **User** (email_ID, full_name, password, DOB, Gender, location)
    - email_ID -> full_name, password, DOB, Gender, location
2. **Last_Logged** (email_ID, motherboard_serial_no)
    - No FD
3. **User_Connection** (email_ID1, email_ID2, is_mute, is_block, show_latest_date)
    - {email_ID1, email_ID2} -> is_mute, is_block, show_latest_date
4. **User_Chat** (email_ID1, email_ID2, timestamp, message, attachment_type)
    - {email_ID1, email_ID2} -> timestamp, message, attachment_type
5. **User_mail** (subject_ID, timestamp, sender_mail, email_ID)
    - No FD
6. **Mail** (subject_ID, timestamp, sender_mail, type, receiver_mail, message, attachment_type, is_draft)
    - {subject_ID, timestamp, sender_mail, type} -> receiver_mail, message, attachment_type, is_draft
7. **Thread** (subject_ID, is_latest, is_visible)
    - subject_ID -> is_latest, is_visible

Candidate Keys

1. **User:** {email_ID}
2. **Last_Logged:** {email_ID, motherboard_serial_no}
3. **User_Connection:** {email_ID1, email_ID2}
4. **User_Chat:** {email_ID1, email_ID2}
5. **User_mail:** {subject_ID, timestamp, sender_mail, email_ID}
6. **Mail:** {subject_ID, timestamp, sender_mail, type}
7. **Thread:** {subject_ID}

Minimal Cover

Since there is no FD in any relation, in which LHS is not a candidate key, the above written dependencies are also the minimal cover for the schema.

1NF, 2NF, 3NF, BCNF 4NF, 5NF?

Observation: In all our tables, we have FDs of type X -> A, X ∈ Candidate Key.

1. **1NF:**
    a. Since all our columns are atomic, the tables are in 1NF.

2. **2NF:**

    a. Since X ∈ Candidate Key, no non-prime attribute can be dependent on a subset of candidate key.

3. **3NF:**

    a. Since X ∈ Candidate Key, there cannot be a functional dependency from non-prime attribute to non-prime attribute

4. **BCNF:**

    a. Since X ∈ Candidate Key ∈ Super Key, this means all our tables are in **BCNF.**

Since there are no multi-valued dependencies or Join Dependencies, the schema is in **5NF**
