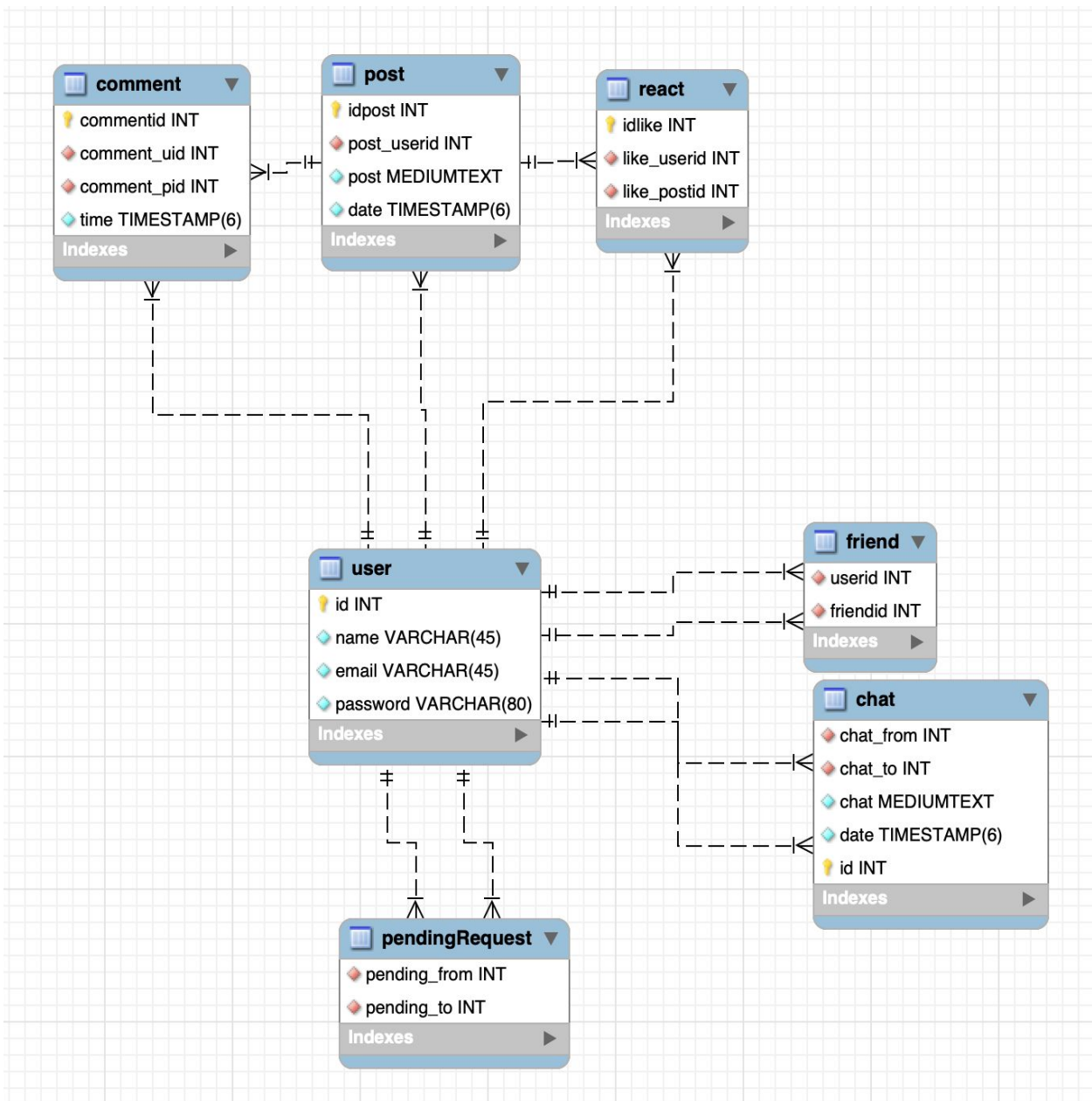


BÁO CÁO CUỐI KỲ
NHÓM 20 : SIMPLIFIED SOCIAL NETWORK
INT 2207 7
THÀNH VIÊN: NGUYỄN HOÀNG VIỆT - 18020063

1. Các chức năng hiện tại

- Đăng ký, đăng nhập và lưu thông tin đăng nhập, xoá khi web được cập nhật mới
- Đăng bài, like, comment được bài của mình và friend
- Gửi yêu cầu, chấp nhận từ chối yêu cầu kết bạn
- Tìm kiếm thông tin người dùng
- Trang cá nhân
- Chatbox (Chưa tối ưu, xảy ra mất dữ liệu)

2. Bảng liên kết cơ sở dữ liệu và liên kết khóa ngoài



Foreign Key	Ref Table Col
comment.comment_uid	user.id
comment.comment_pid	post.idpost
post.post_userid	user.id
react.like_userid	user.id
react.like_postid	post.idpost
friend.userid	user.id
friend.friendid	user.id
chat.chat_from	user.id
chat.chat_to	user.id
pendingRequest.pending_to	user.id
pendingRequest.pending_from	user.id

3. Danh sách các query MySQL

- Đăng nhập: "SELECT * FROM user WHERE email = '"+email+"'"
 - i. Nếu result có length = 0, báo lỗi từ chối đăng nhập
 - ii. Nếu result có length = 1, trường password của result sẽ được so sánh với password người dùng nhập vào bằng bcrypt. Nếu sai báo lỗi từ chối đăng nhập nếu không, đưa vào màn hình chính
- Đăng ký:
 - i. "SELECT * FROM user WHERE email = '" + email + "'" : Kiểm tra xem email đã được sử dụng chưa, báo lỗi nếu email đã được sử dụng
 - ii. "INSERT INTO user (name,email,password) VALUE ('" + name + "', '" + email + "', '" + hash + "')" : được gọi sau khi kiểm tra email không tồn tại. hash được sinh ra nhờ việc mã hoá password bằng bcrypt và đưa vào bảng user
- Lấy các bài đã đăng của bản thân và bạn bè cho trang chính:
 - i. "SELECT friendid FROM friend WHERE userid = '" + id + "'" : Danh sách bạn bè của người dùng
 - ii. "SELECT if(exists(select * from react where post.idpost = react.like_postid and react.like_userid = '1'),1,0) reacted ,post.*,user.* FROM post inner join user on post.post_userid = user.id WHERE post_userid IN (?) ORDER BY date DESC" : lấy toàn bộ các post từ giá trị trả lại của query trước và kiểm tra xem người dùng đã like bài đăng đó chưa
- Lấy bài đăng cho trang cá nhân :
 - i. "SELECT if(exists(select * from react where post.idpost = react.like_postid and react.like_userid = '1'),1,0) reacted ,post.*,user.* FROM post inner join user on post.post_userid = user.id WHERE post_userid = '" + id + "'" ORDER BY date DESC"

- Khi người dùng đăng bài mới:
 - i. "INSERT INTO post (post_userid,post,date) " + "VALUE (" + id + ", " + post + ", FROM_UNIXTIME(" + time*0.001+"))" : time được sinh ra tại thời điểm query được gọi
- Kết bạn :
 - i. "INSERT INTO friend (userid,friendid) VALUE (" + id1 + ", " + id2 + ")"
 - ii. "DELETE FROM pendingRequest WHERE pending_from = " + id1 + " AND pending_to = " + id2 + ""
- Gửi, xoá và kiểm tra yêu cầu kết bạn
 - i. "SELECT * FROM pendingRequest,user WHERE pendingRequest.pending_to = " + id + " AND user.id = pendingRequest.pending_from" : tất cả các request đến người dùng
 - ii. "SELECT * FROM pendingRequest WHERE pending_from = " + from + " AND pending_to = " + to + "" : kiểm tra xem đã gửi yêu cầu kết bạn đến người đó chưa
 - iii. "INSERT INTO pendingRequest (pending_from,pending_to) VALUE (" + from + ", " + to + ")" : Yêu cầu kết bạn mới
 - iv. "DELETE FROM pendingRequest WHERE pending_from = " + from + " AND pending_to = " + to + "" : từ chối kết bạn
- Like/Unlike:
 - i. "insert into react(like_userid,like_postid) value (" + uid + ", " + postid + ")"
 - ii. "delete from react where like_userid = " + uid + " and like_postid = " + postid + ""
- Comment
 - i. "select * from comment inner join user on comment.comment_uid = user.id where comment.comment_pid = " + pid + " order by time desc limit " + from + "," + amount : lấy amount comment và gửi trả lại cho react
 - ii. "insert into comment(comment_uid,comment_pid,comment,time) value (" + uid + ", " + pid + ", " + comment + ", from_unixtime(" + time* 0.001 +"))" : comment mới, time được lấy vào lúc query được xử lý
- Lấy danh sách chat
 - i. "select * from (select if(chat_to = " + uid + ",chat_from,chat_to) as fid, max(date) as maxdate from chat group by fid order by max(date) desc) as alias inner join user on user.id = alias.fid order by alias.maxdate desc" : lấy tất cả những người có chat, được sắp xếp theo thời gian
 - ii. "SELECT * FROM friend,user WHERE user.id = friend.friendid AND friend.userid = " + uid + "" : friendlist, bổ sung những người ko có chat
 - iii. "SELECT * FROM chat WHERE (chat_from = " + uid + " AND chat_to = " + lastChatID + ") OR (chat_from = " + lastChatID + " AND chat_to = " + uid + ") ORDER BY date ASC" : tất cả chat của người chat cuối cùng theo thời gian
- Khi chat
 - i. "INSERT INTO chat(chat_from,chat_to,chat,date)" + "VALUE (" + from + ", " + to + ", " + chat + ", FROM_UNIXTIME(" + time * 0.001 +"))" : time được lấy lúc server xử lý query
- Khi đổi chat:

- i. "SELECT * FROM chat WHERE (chat_from = "" + uid + "" AND chat_to = "" + eid + "") OR (chat_from = "" + eid + "" AND chat_to = "" + uid + "") ORDER BY date ASC"

4. Thông tin hệ thống:

- Backend :
 - i. NodeJS v12.16.1
 - ii. npm 6.13.4
 - iii. "bcrypt": "^4.0.1" : mã hoá và kiểm tra password
 - iv. "body-parser": "^1.19.0" : lấy param của request
 - v. "cookie-parser": "^1.4.5" : không dùng
 - vi. "cors": "^2.8.5" : cho SocketIO
 - vii. "dotenv": "^8.2.0" : cho Heroku
 - viii. "express": "^4.17.1" : xương sống cho Backend
 - ix. "mysql": "^2.18.1" : kết nối với DB
 - x. "nodemailer": "^6.4.6" : Mã xác thực email , đang được xây dựng và thiết kế
 - xi. "socket.io": "^2.3.0" : Chat
- Backend connection :
 - i. NODE_ENV : 'production' : db4free.net
 - ii. NODE_ENV: '!production' : localhost:3306
 - iii. PORT: process.env.PORT || 8080
 - iv. SocketIO : nghe chung port với server
- Frontend:
 - i. NodeJS v12.16.1
 - ii. npm 6.13.4
 - iii. "axios": "^0.19.2" : gửi request đến backend
 - iv. "bootstrap": "^4.5.0" : dự phòng khi kết nối có lỗi, không thể tải được css
 - v. "react": "^16.13.1" : khung xương sống cho Frontend
 - vi. "react-bootstrap": "^1.0.0" : Bootstrap cho React
 - vii. "react-dom": "^16.13.1", Cùng react
 - viii. "react-router-dom": "^5.1.2" : Chia path
 - ix. "react-scripts": "3.4.1", cùng react
 - x. "socket.io-client": "^2.3.0" : kết nối đến backend cho việc chat
- Frontend connection:
 - i. "proxy": "<http://localhost:8080>" : localhost testing
 - ii. NODE_ENV : 'production' : '<https://arcane-everglades-60566.herokuapp.com/>'
 - iii. NODE_ENV: '!production' : 'http://localhost:8080'

5. Screenshot

Email or phonenumber Password

Username

Email

Password

Testing account : Email:a@a Password:a

New post

a

5days ago

aaf

a

1 months ago

dfs

a

1 months ago

ffff

a

New post

Post

a

5days ago

aaf

Like Comment Share

a

1 months ago

dfs

Like Comment Share

a

1 months ago

ffff

Like Comment Share

a

a@b

Search

Name : a

Email : a@b

Requested

User profile

a

5days ago

aaf

Like Comment Share

a

1 months ago

dfs

Like Comment Share

a

1 months ago

ffff

Like Comment Share

a

1 months ago

dfssw

Like Comment Share