Workflow of the Blog project

1. Create MySQL database and tables
   1. Tables
      1. Users
         1. Id auto-increment
         2. First Name varchar(50)
         3. Last name varchar(50)
         4. Email varchar(100)
         5. Password varchar(512)
         6. Token varchar(1024)
         7. Created at date and time
         8. Profile image
      2. Blogs
         1. Id auto increment
         2. User id foreign key
         3. Title varchar(1024)
         4. Description blob
            1. Image
            2. Video

Upload

Embedded

* + - 1. Created at date
      2. Modified at date
    1. Comments
       1. Id auto increment
       2. User id foreign key
       3. Blog id foreign key
       4. Comment text
       5. Created at date
    2. Friends
       1. Id auto increment
       2. User id foreign key
       3. Friend id foreign key
       4. Request status
          1. Boolean

0 pending

1 accepted

2 rejected

* + - * 1. Created at date
        2. Accepted at/ rejected at date

1. Creation of all API in node JS and express
2. Frontend
   1. Homepage
      1. Header
         1. Login/signup
         2. Logo
      2. Body
         1. List of blogs by date description
         2. Created by
         3. Created at
         4. Profile image
         5. Comments
         6. Pagination
      3. Footer
3. Inside of the subsection field
   1. Account
      1. Profile
         1. Edit profile
         2. Profile image
         3. Name
         4. Email
         5. Change password
            1. New password
            2. Confirm password
         6. Manage friends
            1. User list

Friends

Pending request

* + - 1. Manage blogs
         1. List view

Edit

Delete

View

* + - * 1. Add blog

Commands to CREATE tables

1. Users
   1. create table users(id int not null auto\_increment primary key,fname varchar(100),lname varchar(100),email varchar(1024),password varchar(1024),token varchar(1024),pic varchar(1024),created\_at timestamp);
   2. desc users;
   3. +------------+---------------+------+-----+-------------------+-----------------------------+
   4. | Field | Type | Null | Key | Default | Extra |
   5. +------------+---------------+------+-----+-------------------+-----------------------------+
   6. | id | int(11) | NO | PRI | NULL | auto\_increment |
   7. | fname | varchar(100) | YES | | NULL | |
   8. | lname | varchar(100) | YES | | NULL | |
   9. | email | varchar(1024) | YES | | NULL | |
   10. | password | varchar(1024) | YES | | NULL | |
   11. | token | varchar(1024) | YES | | NULL | |
   12. | pic | varchar(1024) | YES | | NULL | |
   13. | created\_at | timestamp | NO | | CURRENT\_TIMESTAMP | on update CURRENT\_TIMESTAMP |
   14. +------------+---------------+------+-----+-------------------+-----------------------------+
2. Blog
   1. create table blogs(id int not null auto\_increment primary key,userid int,title varchar(1024),blogDesc blob,image varchar(1024),audio varchar(1024),video varchar(1024),created\_at timestamp,modifiedat datetime,foreign key(userid) references users(id));
   2. desc blogs;
   3. +------------+---------------+------+-----+-------------------+-----------------------------+
   4. | Field | Type | Null | Key | Default | Extra |
   5. +------------+---------------+------+-----+-------------------+-----------------------------+
   6. | id | int(11) | NO | PRI | NULL | auto\_increment |
   7. | userid | int(11) | YES | MUL | NULL | |
   8. | title | varchar(1024) | YES | | NULL | |
   9. | blogDesc | blob | YES | | NULL | |
   10. | image | varchar(1024) | YES | | NULL | |
   11. | audio | varchar(1024) | YES | | NULL | |
   12. | video | varchar(1024) | YES | | NULL | |
   13. | created\_at | timestamp | NO | | CURRENT\_TIMESTAMP | on update CURRENT\_TIMESTAMP |
   14. | modifiedat | datetime | YES | | NULL | |
   15. +------------+---------------+------+-----+-------------------+-----------------------------+
3. Comments
   1. create table comments(id int not null auto\_increment primary key,userid int,blogid int,comment text,createdat timestamp,foreign key(userid) references users(id),foreign key(blogid) references blogs(id));
   2. desc comments;
   3. +-----------+-----------+------+-----+-------------------+-----------------------------+
   4. | Field | Type | Null | Key | Default | Extra |
   5. +-----------+-----------+------+-----+-------------------+-----------------------------+
   6. | id | int(11) | NO | PRI | NULL | auto\_increment |
   7. | userid | int(11) | YES | MUL | NULL | |
   8. | blogid | int(11) | YES | MUL | NULL | |
   9. | comment | text | YES | | NULL | |
   10. | createdat | timestamp | NO | | CURRENT\_TIMESTAMP | on update CURRENT\_TIMESTAMP |’
4. API flow
   1. Users
      1. Insert user details // http://localhost:8080/insert/registerUser
      2. fetch login details and token insert //http://localhost:8080/fetch/authUser
      3. logout with token delete //http://localhost:8080/delete/logout
   2. blogs
      1. insert blog details http://localhost:8080/insert/bloginsert
      2. fetch blog details http://localhost:8080/fetch/fetchBlog
      3. update blog details http://localhost:8080/update/updateblog
      4. delete blog details http://localhost:8080/delete/deleteblog
      5. commands for blogs fetch details:
      6. select concat(u.fname," ",u.lname) as " full name",u.email,b.title,b.blogDesc,b.image,b.audio,b.video,b.created\_at,b.modifiedat from blogs b inner join users u on u.id = b.userid order by b.created\_at desc;
   3. comments
      1. insert comment
      2. fetch comment

Frontend work

1. Header
   1. Logo
   2. account
      1. Blog
         1. Edit blog
         2. Delete blog
         3. View blog
         4. Add blog
      2. Profile
         1. Profileimage
         2. Edit profile