COMP 8157 Advanced Database Topics

Milestone – 01

Team Members:

1. Agneshkumar Arvindbhai Godhani (110005914)
2. Husain Abbas Kagalwala (110009789)
3. Meet Mineshkumar Patel (110023512)
4. Akashkumar Vitthalbhai Akbari (110019594)
5. Kushal Manvar (110019964)

Project Name: PixtaGram – A Real-Time Memory & Story Sharing Application

1. Related existing models: brief description

After doing research, we found out some existing applications related to our project and those are:

- Instagram
- Snapchat
- Hike
- We heart it
- Retrica

1. Brief description about Instagram:

Instagram is among a breed of smartphone apps which have tapped into the creative, image-based online life of young people.

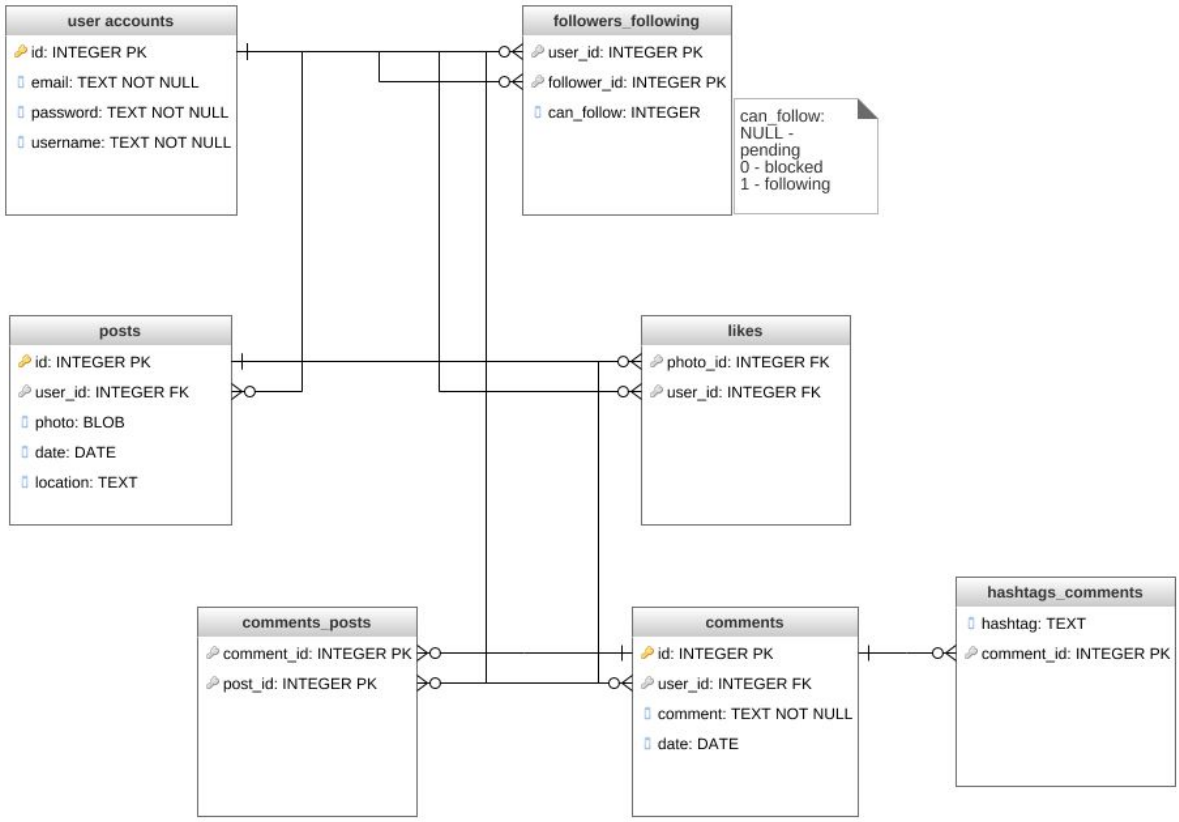
It's likely that you have heard about it in passing, whether from your students, your child or in the media. So, it's about time you got up to speed on the app and what it's all about.

As with other social networking platforms, Instagram users can like, comment on and bookmark others' posts, as well as send private messages to their friends via the Instagram Direct feature. Photos can be shared on one or several other social media sites -- including Twitter, Facebook and Tumblr -- with a single click.

Main Features of Instagram are listed below:

- Instagram Stories
- Instagram Direct
- Instagram Questions
- Hyperlink Username and Hashtags in Your Bio

Database Schema for Instagram (Lower-Level):



Instagram Database Information:

Instagram use Postgres for handling user, media, friendship etc., and use Cassandra for handling fraud detection, users' feed, Direct inbox, and activities.

They have multiple data centers and each data centers hold their replicas of database Cassandra and Postgres also services (mostly core implemented in Django and message broker Rabitmq), all datacenters DBS are sync with each other.

Also, they don't use global Memcached, every datacenter has its own Memcached caches.

This is a brief overview of Instagram application and their database schema's.

2. Brief description about Hike Messenger

Hike Messenger, also called Hike Sticker Chat, is an Indian freeware, cross-platform instant messaging (IM), Voice over IP (VoIP) application which was launched on 12 December 2012 by Kavin Bharti Mittal and is now owned by Hike Private Limited.

Hike can work offline through SMS and has multi-platform support. The app registration uses standard one-time password (OTP) based authentication process. With abundance of low-cost data, Hike decided to go from a single super app strategy to multiple app approach, so that it can focus more on the core messaging capabilities.

Main features of Hike Messenger:

- Hike Stickers
- Text to Stickers
- Blue Packets
- Hike Run

Technology stack behind Hike Messenger:

- Nginx
- MongoDB
- React Native
- Gunicorn
- Amazon S3
- Amazon EC2
- MySQL

Here, is the brief overview of Hike Messenger application which we are taking into consideration while developing our application based on picture's and stories sharing.

2. Draft step by step process of your task

We are developing this project, in order to learn the latest technology which has gained boost in the I.T market (i.e. MERN) stack, which consist of **MongoDB, Express.js, React.js and Node.js** for the implementation.

Our application named PixtaGram is a clone of one of the famous mobile application named **Instagram**, the main purpose for developing this system is to learn real-time integration with **MongoDB** in order to learn new database and to make ourselves familiar with **Cloudfoundary** that is being used for handling images that are uploaded by the user's in our application.

The road-map for our Database Project is presented to you as:

Step No 1: Laying out the main fundamentals and core technologies which are being used.

In the initial phase, when we got the **approval from the professor**, we all team-members came together in virtual meet-up and discussed about the **core concepts and idea's** that we wanted to touch upon by developing this project. We, worked on finalizing the technology that we wanted to learn during this course-work and we all decided to finalize ourselves on MERN stack and to learn MongoDB. Also, we developed the rough idea of the features that our web application would be consisting of having features such as stories and post sharing, Followers building, privacy and password protection features.

Step No 2: Developing prototype for our web application

In this phase of our project (**during reading week**), we all group members started working on developing **front-end user interface** for our web application. We used technology such as Bootstrap 4, HTML 5, CSS 3 and jQuery for developing the front-end framework for our project. We split up in two groups and worked on different modules and pages of our web application and at the end of the reading week we were all set with the **Graphical User Interface** of our web application and now we have to work on the database setup and combining front-end and back-end together.

Step No 3: Designing Database and Entering Dummy Data's for Testing

In this phase, we are now going to design our database model on MongoDB and then enter dummy contents and posts into our database system for testing purpose. We group members have split up in pair of 2 and 3 to work in collaboration for the development of the database. **Agnesh** and **Husain** are working on fields and entities that are required in order to build the structure of the database and **Akashkumar**, **Meet** and **Kushal** are working in collaboration on integration of database to our front-end system and forming dummy data for the contents.

Step No 4: Testing overall system with Stress Test, Load Test and User-Interface and Mobility Testing

In this phase, we are planning to test each and every module of our project thoroughly so that when we are trying to release our web application, we are good to go with all the features running up to date and there is no bug in the overall system.

We are firstly going to test the User-Interface of our application by making various other classmate use our application and gather their feedbacks on our interface design and check the running application on various browsers and handheld devices to get a good knowledge of how our application is working on various screen-sizes.

And, finally we would test our application for the database consistency because, at a single time many user's might be using our application, so the database should work perfectly fine and there should not be any inconsistency in the data updating or inserting mechanisms.

Thus, once this all features are tried and tested, we would host our application on a local server for the course project and we would display or record the video of our application and present it to our professor.

So, this is the overall breakdown of the development process of our application during the course-work and the different existing models which we are studying in order to get more information on the latest technologies and methodologies that are being used in them to make the work cleaner and more efficient for the usage.