Instructor: Mohammad Sadoghi
TAs: Sarmishta Burujupalli
Giritheja Sreenivasulu

Due Date: January 29, 2019
Submission Method: Canvas
Score: 15%

Milestone 1: Analyze, Evaluate, & Demonstrate Core Technologies
ECS 165A

In the pre-milestone assignment, you each had the chance to broadly research available technologies needed for building a mobile app. Hopefully, you now have a general understanding of what it takes to create an app from scratch. Hereafter, you will be working in your groups formed in the pre-milestone stage. The main objectives of this stage are threefold. **(S1) Analyze:** To systematically examine the requirements for building an Instagram App. **(S2) Evaluate:** To experiment (hands-on experience beyond merely reading about a topic) and finalize all technologies needed to fulfill your app requirements. **(S3) Demonstrate:** To present that you have a basic working understanding of the technologies, you will be creating the "Hello World!" Instagram App in your chosen technology stack.

Think Long-term, Plan Carefully.
Be curious, Be creative!

In each group, there must be a lead member for every aspect of the project while all members are expected to contribute and learn about all areas; roughly, the four main areas are (1) front-end, UI, and data visualization, (2) app logic and data aggregation/querying, (3) data model and indexing, and (4) database management and tuning.

(S1) Analyze Requirements:

In Milestone 1, each member will serve in the role of the requirement analyst to gather the necessary functionalities for building an Instagram App. These requirements are essential and a prerequisite for assessing the technologies choices (e.g., frontend framework and backend database).

So, what does a requirement analyst do?

You will have to determine what the user expectations are, which are called the functional requirements. For example, one of the main expectations for a user while using applications like Instagram would be to have an option to make or follow friends on the platform and communicate through posts, likes, or via messaging.

To help you get started, here are basic requirements for Instagram App. These merely serve as examples and you are encouraged to think creatively not bound to any accepted social norms. Of course, any useable app would require identifying and managing user profile while facilitating social interactions.

1. **Authorization**
    - The user should be able to create a new account
    - The user should be able to sign-in into the existing account via social networks or personal email
2. **User Profile Management**
    - The user should be able to add personal data, e.g., about me, interests
    - The user should be given options to edit and delete personal data
3. **Post Creation**
    - The user should be able to upload pictures taken from the app or existing images from the gallery
    - The user should be able to mention people by adding tags
4. **Activity Feed Maintenance**
    - The user should be able to follow/unfollow people through which the user feed is adjusted accordingly
    - The user can like and comment on a post
5. **Search Feature**
    - The user can search by username, email, or by hashtag feature

After you have gathered the desired functionality of your application, make a list of features that you plan to implement.

Example: User Profile Management
Q. Will you be giving the feature to make the account private?
If yes, then you would make a note of it in your list of features. You would have to brainstorm with your team members and decide upon the set of services and features that you would provide the user with while using the application.

Now that you have defined and finalized the goals of your system, the next step is to determine what data you have to store in your database to meet your feature needs. The way you would go about this is by first creating an ER (Entity-Relationship) diagram.

Sample steps to create an ER diagram:

1. Identify the entities which are mostly nouns.
2. Identify the attributes related to the entities.
3. Identify the relationship between the entities, usually the verb.
4. Design Choices? Please review Chapter 2!

Notably, ER diagram is an iterative process. It is based on the constraints/features you choose, refinements will be made as you build the application.

(S2) Evaluate Technologies:

Apart from the compiling requirement and drafting your ER Diagram, you are expected to decide on the frontend and backend technologies that you explored in the pre-milestone phase. As a reminder, examples (but not limited to) of frontend are React Native, Android Studio, or Swift; and of backend are SQLite, Firebase, MongoDB, or PostgreSQL. Naturally, your choice of technology must be motivated and justified by your App requirements.

(S3) Demonstrate Basics:

Once you have finalized your technology stack, you need to create a basic live demo which demonstrates your ability to work with these technologies, i.e., building a basic "Hello World!" Instagram App.

We would expect a live demo of the App (e.g., in React Native) which will show "Hello Alice!" Here the text 'Alice' is retrieved from the database (e.g., using SQLite) with at least one table (e.g., User Table). Note this seemingly simple exercise will cut through and demands the basic skeleton all four tiers of your App (1) front-end, UI, and data visualization, (2) app logic and data aggregation/querying, (3) data model and indexing, and (4) database management and tuning. How you wish to design and demonstrate your "Hello Alice!" Instagram App is all up to you, you have complete freedom!

Milestone Deliverables/Grading Scheme: What to submit?

At the end of this milestone, each team needs to prepare a presentation that concisely summarizes the entire progress including analysis of requirements and features, ER diagram, technology choices (and its justification), and a live demo of the "Hello Alice!" App (which must use all the chosen technologies). Also, you will need to submit **the presentations in .pptx, .key, or .pdf format and the live demo code by the due date.**

The actual presentation and evaluation will be scheduled after the milestone due date. Each group will be assigned a dedicated 15-minute timeslot. The presentation must be completed strictly in 10 minutes (no extra time would be granted) followed by a 5-minute Q&A. In Q&A, each team member will be asked questions related to any part of the milestone to ensure every student's participation and understanding of the whole assignment. Groups with five members will receive an 18-minute time slot.

During the 10-minute presentation, each student must present their respective parts, e.g., each team member would take up one component from the basic functionality list or they would focus on the tier of the app they are leading (e.g., front-end, app logic, data model, backend database).

The presentation should include the following (but not limited to):

1. The functionalities and features that you decided upon.
2. The overall system design for the whole application based on the gathered requirements, namely, an executive overview of your Instagram App.
3. The ER model along with your design choices and constraints.
4. The technology stack you are going to use for front-end and back-end.
5. A live demo of the "Hello Bob!" Instagram App based on the technology stack.

Important Note: The presentation slide and live demo must be identical to the materials submitted by the milestone due date.

As noted on the course syllabus, for each milestone, a portion of the grade is devoted to the presented project as a whole on which all members receive the same grade (70% of the grade), but the remaining portion is individualized (30% of the grade), so for each milestone, not all group members may receive the same grade. In each milestone, **a bonus of up to 20% can be gained** to further encourage taking a risk, going the extra mile, and just to be curious & creative.

Late Policy

There will be a 10% penalty for each late day. After three late days, the homework will not be accepted.

Course Policy

In this class, we adopt the UC Davis Code of Academic Conduct available here.

Changelog:

Milestone Handout Version v1: January 20, 2019 (initial posted version)