

CIS557 Project

Photo-sharing Social Network APP

Setup: GitHub Project

We will use **GitHub projects** and **Extreme Programming (XP)** methodology when implementing this project. As a reminder, XP provides 29 simple rules to be followed in terms of **Planning, Managing, Designing, Coding, and Testing**.

1. You will create and configure a **project in your GitHub repository**. (see useful links below).
2. You should create a **wiki page in your GitHub repository** listing and describing your user stories and story points.
3. Each user story should be listed in the **GitHub's tracker** as an **issue**. You must label your issues and assign them to specific member(s) of your team

Implementation: User Stories

You will implement the following user stories (see table below).

- It is up to you to define how you will implement them and the details of the user interface.
- You must provide your user points for each user story.
- The number (level) in parentheses are used for grading.
- Each feature must be fully tested: unit tests, functional tests, integration tests.
- You must use Jest for testing, Selenium for test automation, and Travis for continuous integration
- Your unit test should achieve the highest code coverage possible
- You will use MongoDB or MySQL as database engine
- You will use Node.JS, or Ruby on Rails, or Django as webserver
- Your implementation will use a REST or a GraphQL API for frontend/backend communication
- You can use a 3rd party software, packages, or modules (like JQuery, ReactJS or Express) when implementing your apps
- You should explain your design decisions in your GitHub wiki page
- You should make sure that your web application does not crash during usage/testing
- You must deploy your app on Heroku
- Each team member GitHub contribution will be used when grading the project

Features		
#	User Story (US)	Story Points
1	User registration(0)	
2	Login/Auth (0)	
3	User profile page(0)	
4	Create post / Photo upload (0)	
5	Activity feed (1)	
6	Photo likes & unliking (1)	
7	Follow/unfollow users (1)	
8	Comments (2)	
9	Editing/Deleting Posts (photo) & Comments (2)	
10	Follower suggestions (3)	
11	Tagging photos / @mentions in comments (3)	
12	Interactive API documentation using Swagger (3)	
13	Live update / Browser's notification (4)	
14	Privacy / Visibility control on photos (4)	
15	Pagination / 'infinite scroll' (4)	

Validation:

- Your JavaScript must be clean, readable, and **ESLint error and warning-free**. For this assignment, we will **use the Airbnb Javascript style**.
- Your HTML file(s) must pass validation at <http://validator.w3.org>.
- Your CSS files(s) must pass validation at <http://www.css-validator.org/>.

Grading:

You should schedule a demo with your TA during the last week of class.

Assuming that each user story is implemented completely (description of US and issues in GitHub project, tests, implementation, and deployment) and that all user stories within each level are implemented, your work will be graded as follows:

- **Level 0 will get up to 60%**
- **Level 1 will get up to 75%**
- **Level 2 will get up to 85%**
- **Level 3 will get up to 95%**
- **Level 4 will get 100% and up.**

Milestones:

1. **Level 0 features due October 22nd. 5 points penalty for late submissions**
2. **Level 1 features due November 12th. 5 points penalty for late submissions**
3. **Level 2 features due November 26th. 5 points penalty for late submissions**
4. **Level 3 features due December 3rd. No late submissions allowed**
5. **Level 4 features due December 3rd. No late submissions allowed**

Disclaimer:

This is a live document and changes may be made in the future.