# Activity 1 Questions

1. What is the URL of your Github project?
   1. <https://github.com/hulto/CSEC380-Team4>
2. How did you break up your projects and what are the security ramifications?
   1. We broke up the project tasks into epics, and each of those epics into user stories. We are planning to have each of us handle a user story in each epic. The security ramifications of this are during development of the application, having multiple people working on the same project can lead to lazy coding practices, inconsistent security, or confusion as to the intended functionality of our webapp.
3. How did you choose to break down your Epic into various issues (tasks)?
   1. We chose to treat each main task of our app as a project. For example: Authentication, Upload, and introducing vulnerabilities are projects. Each epic represents the main deadlines associated with the project. Epic 1 encompasses the work associated with activities 1 and 2. Epic 2 encompasses the rest of the project up until the final due date.
4. How long did you assign each sprint to be?
   1. 2 weeks per sprint.
5. Did you deviate from the Agile methodology at all? If yes, what is your reasoning for this?
   1. No
6. How do you ensure that after each issue/milestone that security has been verified? How would you identify such issues in an ideal environment?
   1. Security will be verified by travis unit testing each time a pull request is made. We will also enforce code review on all code being merged into the master branch. We would identify issues in an ideal environment using unit testing with 100% coverage, and 2 person code review for each issue.

# Activity 2 Questions

1. What Web Application security mechanisms are involved in your topology? What security mechanisms would ideally be involved?
   1. The only security mechanism involved in our topology at the moment is a public facing caching server to store larger files and hopefully reduce traffic on the main web server. In an ideal high-traffic scenario, we would have a load balancer splitting the work between multiple different web servers in order to deliver web content even at extremely high traffic levels. We would also prefer a firewall between our services and the public internet, this would help defer any unwanted traffic intended for our public facing server or any of its subsidiaries
2. What testing framework did you choose and why?
   1. We chose to use pytest as our testing framework because it was the technology that our developers were most familiar with. This means that we as an organization are able to generate much more thorough unit tests of our webapp and better adhere to a test-driven development strategy