**Questions**

An instructor in the Fisheries & Wildlife department has a course set to begin in three months. Along with standard coursework in the Canvas course, the instructor wants to provide a web interface allowing students to memorize taxonomic information about various species of fishes, as well as keep the information up-to-date. The instructor has produced thousands of digital images as well as realistic 3D models for some species. Describe how you would develop a web site to present the provided images and 3D models that would be effective for learning.

**Answer**

**Assumptions:**

1. We have decided against using a content management system like Drupal.
2. The project will be built from scratch.
3. While this outline contains 7 steps the general overview of the project would be the following:
   1. Application design
   2. Development
   3. Testing
      1. Unit Testing would also be used during all phases of development
   4. Deployment

**Step 1: Initial Design and Analysis**

1. **Information Gathering**
   1. **Backend:** The first step I would take given their availability would be to speak with the Instructor and gather information related to his expectations for this project. Right away I would be seeking to understand his goals for this current project but also trying to see if this site may grow to be used Department wide in Fisheries and Wildlife or Campus Wide for both online and on campus courses.
   2. **Front End:** I would also seek to gain a very broad understanding of the look and feel they are going for. I would bring samples of similar sites and OSU sites to show and make sure the Professor has an idea of where the project is heading in regards to visual aspects and user interface.
2. **Begin Coding Design and Analyze code reusability for future projects**
   1. After gathering all the preliminary information on the backend and front end I would begin to layout the logic of the site from a very high level.
   2. This would include looking for coding patterns that can be bundled into classes and functions, analysis for building and MVC style site with the view layer separate so the code could be used for future projects.
3. **Discuss building a web application interface for future projects**
   1. I would also look meet with staff and discuss the need for meeting the initial deadline but developing a web application so that future versions could support mobile devices and tablets. This would be different then responsive web design but actually developing the native (or potentially non-native) mobile applications.

**Step 2: Create Project Timeline**

1. Three months could be a potentially quick timeframe for a project depending on the project team size therefore the timeline would be rather brisk with aspects of Agile Development as we stay in communication with project leaders (the Professor, Fisheries and Wildlife Department and the Extended Campus).
   1. Application design (1 Week)
      1. During this time, we would also be building parts of the code that we know we need like the database and major components.
   2. Development (7 Weeks)
   3. Testing and Finalizing (3 Weeks)
   4. Deployment (1 Week)

**Step 3: Design and Application Development**

1. Frameworks, Libraries and Tools
   1. Version Control
      1. Most likely would use Git and either GitHub or BitBucket as a repository
   2. CSS
      1. Implement existing OSU CSS style
      2. Most likely utilize SASS
   3. Server
      1. Dependency Manager
         1. Composer for PHP
      2. Framework
         1. This would be something I would discuss but with PHP I would lean towards Laravel because it has a large community, I have experience in it and it provides a nice structure for developing a RESTful API. It also provides nice options for database migrations which would be useful if the project or team grows.
   4. Client
      1. Dependency Manager
         1. NPM for JavaScript
      2. Framework
         1. I am open to any but if I were tasked with choosing I would go with React JS with a Flux architecture or Angular with a MVC architecture.
         2. React really excels at creating fast single page web applications. Given the short time frame I would lean towards Angular do to the fact that it provides the MVC and not just the View Component like React does.
2. Database Design
   1. For handling the files, I would build the MySQL tables to handle links to the file storage system. Security would be important and issues like SQL injection would be all considered.

**Step 4: Testing and Launch**

1. The final stage would be a launch where the system would go live and be tested. The more users to find problem areas here would be better.