Team Project

Ashesi Web Development 2025

Percentage of Course: 25% of course evaluation

Date Set: 24th Sept 2025

Last Modified: ---

Due Date: 30th November 2025

Assessment: All team members must participate in the group project. We will assess individual team members based on peer feedback and GitHub contribution. You will receive an individual grade.

Assignment Summary

This assignment runs for most part of the semester. You will work together to define and implement a database powered web application with HTML, CSS, JavaScript, PHP, MySQL and React. Your project topic / area must be approved by the faculty before you start development. We have several intermediate points where deliverables are due. The final deliverable is due: 30th November 2025

Assignment Tasks / Deliverables and Dates

Task	Date Due
Topic Submission on Canvas	29th Sept
Case Scenario	3rd October
Team Project Website	10th October
Sprint 1 First Slice and Demo	~~24th October~~ 5th November
Sprint 2 Second Slice and Database Diagram	~~7th November~~ TBD
Sprint 3 Third Slice and Final Design	21st November
Final Project Due	30th November

Grading Rubric:

Deliverable and Weight	Elements of Rubric
Team Project Website	Includes pages that are well organized and easy to understand concerning the specific information called for.
Team Component Sprint 1	The project is judged according to the following: • All elements of the sprint are complete and coherent. • Some general functionality was implemented, esp. in HTML and CSS. All frontend template completed • Code was committed and stored to GitHub
Team Component Sprint 2	The Following elements contribute to the score: • All elements of the sprint were completed • The database design was submitted and adequate

	• The team's presentation provided a coherent overview of the project. • Additional functionality was logically implemented, especially with work on the "backend" or PHP aspects of the program
Team Component Sprint 3	• All elements of the sprint were completed and coherent. • The project could be accessed and was deployed to a live web server. • Some data driven pages were implemented and are functional
Team Component Final Project	• All final deliverables were submitted; and were complete and coherent. • The project can be built and run locally based on instructions in the README. • The project is live and deployed on a web server • The project exhibits the ability to create, read, update, and delete data from the web front end; and provides some analysis or computed presentation and action from that data. • The presentation is coherent, compelling, and is a strong demonstration of the work of the team
Individual Assessment	• The individual assessment provides a room for members in the team to get different grades based on their overall contribution. The percentage will depend on the relative difference of the individual's contribution vs. the team. So, a team member who contributes significantly above or below their team members based on analysis of code contributed, individual feedback, GitHub contribution, and instructor observation may have an individual assessment that is a higher proportion of their grade.

Defining a Case / Scenario

The first part of your assignment is to develop the case for the website in additional detail. For this portion of the assignment:

- describe the business, organization, or group of people that will use this website. If a business, describe its products or services and what its purpose is
- describe the problem that the website will solve for the entity you are building for.
- What are the 3-4 functions that will need to be built for those problems
- Describe the roles or types of users involved in using the website.

Build a simple 4 Panel Storyboard of the key functionality provided to your user.

The Team Project Website

Create a live WordPress/any free CMS website for your team project. At the minimum, complete the following pages:

- About project.

- The case / scenario you are building and supporting functionality. Include the storyboard from the previous assignment (and add any new storyboards you create).
- **Include (if different) the storyboard for the first piece of functionality you will build**
- A page for each member with Profile Picture & Bio, Contact information and Role in project.
- A link to the GitHub repository of the project

Sprints and Sprint Deliverables

Approximately every week we require that the project be in a working state that can be shown to the teaching team and class; as well as defining specific information about the project. The project at each of these points should be checked into GitHub; and there should be sufficient instructions such that a member of the faculty team could build and run the website on a computer with Apache, MySQL, and PHP.

Common Sprint Deliverables

Each of the three sprints require team to deliver information on their team website that has the following:

- A **description of the functionality** completed in total by the team as of the end of the sprint.
- A description—**user guide**--of how to use the currently implemented functionality. You may decide on relevant screens to link to pages of your user guide to help them as they go along.
- A short **summary page by team member** of what each team member's contribution was to the sprint (specific functions or activities each contributed.
- A link to the milestone/commit in GitHub of their project supporting the deliverable. **All code and tests for the project should be maintained in GitHub.**
- **A page on testing strategy** and tests. What automated (unit) tests are written for the project? How can those tests be run? (optional)
- **A retrospective page** collectively written describing what went well, what did not go well, and what the team is changing about how they work in the next sprint

The link to the Sprint page overview will be submitted in Canvas.

Sprint 1 Deliverables

- All common sprint deliverables.
- **A page on architecture** detailing:
    - What are the major pages that are currently planned for the website?
    - What are the major tables, or PHP functions implemented or envisioned to support the pages?
- Which of those pages are implemented; and which are planned
- A description of any frontend libraries / frontend choices. For example, is a CSS layout library being chosen? Will the team use jQuery or another major front-end framework?
- The team will be able to do a demo of their first functional slice with running code that gives a sense of the value of the website.

A successful initial sprint will have several pages and complete a story providing some useful function for the target user.

Sprint 2 Deliverables

- All Common sprint deliverables
- Updated Architecture Page
- An Entity Relationship Diagram describing the database architecture.
- A demo.

The Database Entity Relationship Diagram

Develop an Entity Relationship Diagram that captures the detailed requirements (entities, relationship between entities and relationship types) for the database system of your project. For each entity, there should be an associated written list of all the attributes that the entity possesses. Remember to indicate all primary and foreign key fields for each of the tables using suitable and consistent notation.

Create a database making sure that appropriate fields are defined as key, and that other suitable data integrity rules are enforced.

The name of the database should be your group's initials and yeargroup. E.g. if the group name is 'The Developers Craft', then our database name would be 'TDC2024'. (Hint: make sure you create the tables in an appropriate order – for instance, those that have foreign keys cannot be created first – why? Think about it!).

Sprint 3 Deliverables

- All Common sprint deliverables
- Final Architecture Page
- **A page which has a flow diagram** describing how the pages/functions of the system relate to each other on the website
- A page describing **any remaining functionality** that is intended to be delivered by for the final.

Final Deliverables

Individual Report from each member combined into 1 document

- Key Contribution and role
- Features worked on and completed, and a short description of those features.
- ~~Which unit tests or testing strategies were employed were written to support the features~~
- A description of your most important algorithm or design
- Challenges
- Lessons Learned

Copy of Presentation Slide

- Link to project on GitHub
- Link to WordPress or CMS
- Link to live server

- Indicate which Team member can start the VM if not started to be contacted by Faculty team to evaluate.
- README with instructions on how to set up a server running the product
- If you do not have GITHUB, then submit a zip source file

PowerPoint Guide (This is just a guide – you do not have to do all that is specified here)

- Background
    - describe the business, organization, or group of people that will use this website. If a business, describe its products or services and what its purpose is
    - describe the problem that the website will solve for the entity you are building for.
    - Describe the roles or types of users involved in using the website.
- Functionality
    - What are the 3-4 major functions that were built for the problems articulated
- Architecture and design
    - High level system overview / architecture (3 tier)
    - Modules / layers of implementation
- Implementation
    - Tools, Flowchart, classes, libraries, framework, APIs, Programming Languages, etc.
    - (optional) A summary slide (one per team) describing each members role and contribution.
- Testing / Validation
    - Any unit or system testing, or validation strategies employed
- Demo (4 Minutes) - Place IP/URL address on the slide.
    - Show a realistic scenario from the point of view of one or more of your users