

Generic Syllabus of Website Development I

Kazuo Sugihara

June 25, 2021

This “**generic syllabus**” is designed to be independent of a platform and/or a programming language used. It should be customized by a coach for a particular platform and/or a programming language that the coach chooses.

Prerequisites: Programming Basics in Any Programming Language

| Items | weeks |
|--|-------|
| 0. HTML & CSS Basics This section is only for completely newbies, based on the following tutorial. http://probono.dyndns.org/~sugihara/courses/Myanmar/ Trainees who know basics of HTML and CSS skip this and join from the next section. | 3 |
| 1. Getting Ready (Work Environment Setup) (a) Set Up Virtual Machine and Guest OS (if a chosen platform is different from the trainees’ platform e.g., VMware Workstation, VirtualBox, QEMU for Using Linux on Windows 10 PC) (b) Install IDE (including either a compiler or an interpreter, code editor, SDK, etc) (c) Set Up HTTP Server (either Apache or Microsoft IIS) (d) Set Up DBMS Server (either MySQL or Microsoft SQL) (e) Install Graphics Editors (either Photoshop & Illustrator or GIMP & Inkscape) | 2 |
| 2. Project Requirements Comprehension Requirements of a product to be developed are given in either way (or both). (a) Written documents of requirements specifications (b) Prototype of the product developed by a coach as an example | 2 |
| 3. Project Management Plan Discussion among the coach and all trainees about an organization of the product: (a) Each trainee tries to break down the product into modules (b) Each trainee decides tasks to be carried out, based on the structure of the modules (c) Each trainee creates a project plan, according to dependencies among tasks Sample Projects Listed Below (Just as Examples) | 2 |
| 4. Work on Each Task for the Project * The length varies in a wide range, depending on project(s) | * |
| 5. Documentation for Other Trainees to Review Each Product | 2 |
| 6. Functional Testing and Performance Evaluation by Peers | 2 |
| 7. Delivery (Documentation & Demo) of the Final Product | 2 |

Repeat 3–7 if there are multiple projects

Sample Project 1: *Web Forms for Website Registration*

Create the following Web forms that a user signs up for a website.

(a) Form1 titled “Sign Up”

It contains:

- 3 text fields “Username”, “Password”, and “Verify Password”
- Two buttons “Submit” and “Clear”
- Icon “Show” next to the “Password” field

Upon clicking the “Submit” button, execute the following actions.

- (a) If values of the two “Password” fields are not identical, display an error message and have a user retry to input a password again.
- (b) If a value of “Username” already exists in a database, display a warning for changing it.
- (c) If the value of “Username” is new, go to Form2 by passing the values of “Username” and “Password”.

Upon clicking the “Clear” button, discard all the input values and erase values in all fields.

Upon clicking the “Show” icon, toggle the masking property of the 2 password fields.

- For the default state of masking a password value, each character is displayed as a bullet.
- For the disabled masking state, display characters of a password instead of bullets.

(b) Form2 titled “Contact Information”

It contains:

- Text fields “First Name” and “Last Name”
- Text field “Email Address”
- Text fields “Street Line 1”, “Street Line 2”, “City”, “State”, “Postal Code”, “Country”
- Drop-down menu “Tel Country Code” (e.g., +95 for Myanmar)
- Text field “Phone Number”
- Two buttons “Submit” and “Cancel”

Upon clicking the “Submit” button, execute the following actions.

- (a) If a value of “Email Address” already exists in the database, display an error message.
“You already signed up with this email address. You cannot sign up again with it.”
- (b) If all 4 values of “First Name”, “Last Name”, “Tel Country Code” and “Phone Number” already exist in the database, display an error message
“You already signed up with this phone number. You cannot sign up again with it.”
- (c) Otherwise, go to Form3 by passing all the input values.

Upon clicking the “Cancel” button, discard all the input values and display Form1.

(c) Form3 titled “Review & Confirmation”

It displays all values that the user has given so far.

It contains two buttons: “Confirm” and “Cancel”.

- Upon clicking the “Submit” button, add all the input values in Form1 and Form2 into the database and then display a Web page with a sign-up confirmation message, e.g.
“You have successfully signed up with the username XXXXXX. Thank you.”
where XXXXXX is an actual username.
- Upon clicking the “Cancel” button, discard all the input values and display Form1.

Security Remarks:

- (a) A password must NOT be stored in a database as a plaintext. Instead of the password itself, a “hash” value of the password is stored into the database. A choice of a so-called hash function used to hash a password is critical. For example, the hash function MD5 is a poor choice even though it is widely used. Recommended hash functions are argon2, scrypt, bcryptor, PBKDF2, SHA3, or any stronger one.
- (b) Simply storing a hash value of a password is not secure enough. The attack technique called “Rainbow Table” makes it possible to get a keyword for its hash value. So, it is recommended to add so-called “salt” randomly generated string to a password before hash. Furthermore, it is recommended to use another technique called “pepper”.

Sample Project 2: Web Form for Website Authentication

Assuming that you developed a module for website registration (Sample Project 1), develop a module for authentication (i.e., login module).

It consists of the following single Web form.

Form4 titled “Log In”

Extra of the Project 2: Automatic Session Timeout

Try to develop a complete component of sign-up & authentication with an automatic “session timeout” such that after a successful login, authentication is automatically nullified (automatic logout) when there is no activity on the current Web page for a certain period of time (e.g., 10 min).

Open Source IAM (Identity & Access Management)

- <https://auth0.com/>
- <https://www.keycloak.org/>
- <https://medevel.com/iam-systems-10-identity/>

Sample Project 3: Website for Browsing & Searching Items of an Online Shop

Build a website of an online shop selling items of 3 categories “Cosmetics”, “Shoes”, and “Books” with the following functionality.

1. A customer can browse items in each category or search items with keywords, and view a description & images of an item, where the information of each item is stored in a database.
2. A customer can select an item and add the item with its quantity into a cart.
3. When a customer clicks the “Cart” button, the viewer can views a list of items in the cart with an invoice of the items.
4. A store manager (after authentication) can add items and their information into the database.

Extra of the Project 3: Expand the Cart module with the functionality of checkout after authentication

There are free open source shopping carts such as Prestashop, OpenCart, and Zen Cart.

<https://infobeat.com/top-10-free-open-source-shopping-carts/>

Licenses of Open Source Codes

The coach may give trainees an open source product relevant to a project and suggest them to learn, customize, and/or reuse the open source product. Remember to tell the trainees to observe the license terms of the open source product, (e.g., BSD License, GNU GPL, Apache License, and Creative Commons License) when they use and/or modify code of the open source product.