Date	Day	Week	Objective	Activity	Achievements	Supervisor Remark	Appendix
20/11/2023	Monday	Week 9	1) Generate website User Dashboard front-end using WordPress. 2) Generate website User Profile front-end using WordPress.	1) Search template and design ideas for financial analysis website User Dashboard from 2) Create and design test2.fhase.io website's USER DASHBOARD page front-end using Divi theme. 3) Create and design sidebar and header of the test2.fhase.io website's USER DASHBOARD page based on searched User Dashboard template using Divi theme. 4) Create and design test2.fhase.io website's USER PROFILE page front-end using Divi theme.	1) Successfully searched the most suitable Financial analysis website User Dashboard page template. 2) Successfully added USER DASHBOARD page for test2.fhase.io using WordPress Theme Customizer. 3) Successfully added, edit, and modify sidebar and header of USER DASHBOARD page for test2.fhase.io using Divi Builder based on the searched financial analysis website User Dashboard template. 4) Successfully added USER PROFILE page for test2.fhase.io using WordPress Theme Customizer. 5) Successfully edit and modify test2.fhase.io website's USER PROFILE page using Divi Builder.		
21/11/2023	Tuesday		1) Generate website User Dashboard front-end using WordPress. 2) Modify website News page's front-end using WordPress.	1) Redesign test2.fhase.io website's USER DASHBOARD page to be simpler and compatible with laptop, tablet, and mobile displays. 2) Added bar chart and to the test2.fhase.io website's USER DASHBOARD page front-end using Divi theme. 3) Search template and design ideas for news by comparing design 4) Redesign test2.fhase.io website's NEWS page by adding new features, display latest news, and compatible with laptop, tablet, and mobile displays.	1) Removed the sidebar and header due to not compatible with laptop, tablet, and mobile displays. 2) Successfully re-edit, and modify the design of USER DASHBOARD page for test2.fhase.io using Divi Builder based on the searched financial analysis website User Dashboard template. 3) Successfully searched multiple suitable website News page template. 4) Successfully re-edit, and modify the design of NEWS page for test2.fhase.io using Divi Builder by combining the design of multiple website News page template.		
22/11/2023	Wednesday		1) Learned to initiate plugin. 2) Setting WordPress File Upload plugin for the Wordpress Login page.	1) Research tutorials to configure file upload feature in WordPress website using WordPress File Upload plugin from YouTube. 2) Learn to configure and enable the file upload feature in WordPress's USER DASHBOARD page using WordPress File Upload plugin. 3) Learn to configure and enable the image upload feature in WordPress's USER PROFILE page using WordPress File Upload plugin. 4) Modify and edit the design of the file upload feature for both USER DASHBOARD page and USER PROFILE page.	1) Successfully configure and enable the file upload feature in WordPress's USER DASHBOARD page using WordPress File Upload plugin. 2) Confirmed the uploaded files in the USER DASHBOARD page are successfully added to the WordPress File Upload database. 2) Successfully configure and enable the image upload feature in WordPress's USER PROFILE page using WordPress File Upload plugin. 4) Confirmed the uploaded images in the USER PROFILE page are successfully added to the WordPress File Upload database. 5) Successfully edit and modify the design of the file upload feature for both USER DASHBOARD page and USER PROFILE page.		
23/11/2023	Thursday		1) Ensure the images used in test2.fhase.io are Creative Commons Licensed. 2) Ensure all the texting, alignment, border radius, design, sizing, margin, and padding.	1) Search and download Creative Commons License Images from Creative Commons Search Website. 2) Upload the Creative Commons License Images to the Media Library of test2.fhase.io website. 3) Change test2.fhase.io website's images with the downloaded Creative Commons License Images. 4) Modify test2.fhase.io website's heading texts, border design, border sizing, margin, and padding.	1) Successfully searched and downloaded the Creative Commons License Images which compatible with test2.fhase.io website from Creative Commons Search Website. 2) Successfully uploaded the downloaded Creative Commons License Images to the Media Library of test2.fhase.io website. 3) Successfully change all the test2.fhase.io website's images with the downloaded Creative Commons License Images. 4) Successfully edited the heading texts, border design, border sizing, margin, and padding of the test2.fhase.io website to ensure compatibility with laptop, tablet, and mobile displays.		
24/11/2023	Friday		1) Learned to create Login and Signup page integrated with Firebase Authentication using Angular Framework. 2) Learn to setup a new Angular project integrate with Firebase Authentication.	1) Research Firebase Authentication tutorials from YouTube and Firebase Doc. 2) Research tutorials to integrate Firebase Authentication with Angular project from YouTube and 3) Create a new Firebase project Firebase Analysis1 and setup Firebase Realtime database and Firebase Authentication.	1) Successfully learned to set up Firebase Authentication using Firebase console step-by-step. 2) Successfully learned and acknowledge the features available using Firebase Authentication. 3) Successfully learned to integrate Firebase Authentication with an Angular project step-by-step. 4) Successfully created a new Firebase project Firebase Analysis1 and setup Firebase Realtime database and Firebase Authentication.		
25/11/2023	Saturday						
26/11/2023	Sunday						
27/11/2023	Monday	Week 10	1) Search for a tutorial to guide the installation and import of @angular/fire in an Angular project. 2) Create a new Angular project with correct installation of @angular/fire.	1) Research tutorials on connecting Firebase Authentication to an Angular application on YouTube. 2) Create a new Angular project PhaseSignup with CSS stylesheet format using npm command "ng new PhaseSignup". 3) Install Firebase packages in Angular project PhaseSignup using command "npm install @angular/fire firebase" based on YouTube tutorial "How to Connect Firebase Auth to an Angular App". 4) Add Firebase configurations in Angular project PhaseSignup's environments/environment.ts. 5) Import and register AngularFireModule, AngularFireAuthModule, AngularFireStorageModule, AngularFirestoreModule, AngularFireDatabaseModule in Angular project PhaseSignup's app.module.ts.	1) Successfully learned to connect Firebase Authentication to an Angular application from YouTube. 2) Successfully created a new Angular project PhaseSignup with CSS stylesheet format. 3) Successfully installed the Firebase packages in Angular project PhaseSignup. 4) Successfully import and register AngularFireModule, AngularFireAuthModule, AngularFireStorageModule, AngularFirestoreModule, AngularFireDatabaseModule in Angular project PhaseSignup's app.module.ts.		
28/11/2023	Tuesday		1) Search tutorial to implement complete Firebase Authentication in Angular 16. 2) Step-by-step implement complete Firebase Authentication in the Angular project PhaseSignup.	1) Step-by step implement Firebase Authentication in Angular project PhaseSignup based on the tutorial "Implementing Full Firebase Authentication in Angular 16: A Comprehensive Tutorial" 2) Create user interface file and auth service file that cover sign-in, sign-up, password reset, email verification, and route protection for Angular project PhaseSignup. 3) Create .ts file and .html file for sign-in and sign-up in Angular project PhaseSignup.	1) Successfully created the user interface file and auth service file that cover sign-in, sign-up, password reset, email verification, and route protection for Angular project PhaseSignup. 2) Successfully created the .ts file and .html file for sign-in and sign-up in Angular project PhaseSignup.		
29/11/2023	Wednesday		1) Finish implement complete Firebase Authentication in Angular project PhaseSignup based on . 2) Test user authentication in Firebase console.	1) Create .ts file and .html file for forgot-password, verify-email, and dashboard in Angular project PhaseSignup. 2) Run the Angular project PhaseSignup using command "ng serve --open". 3) Check the Firebase Authentication of Angular project PhaseSignup by register a new account and login into the created account.	1) Successfully created the .ts file and .html file for forgot-password, verify-email, and dashboard in Angular project PhaseSignup. 2) Successfully run the Angular project PhaseSignup using command "ng serve --open". 3) Successfully added new user in the Firebase Authentication for the Angular project PhaseSignup. 4) Unsuccessfully added user information in the Firebase Realtime database for the Angular project PhaseSignup.		
30/11/2023	Thursday		1) Search for solutions to resolve issues with integrating Firebase Authentication with Firebase Realtime Database. 2) Learned to implement Firebase Realtime database in the new Angular project PhaseLogin	1) Research tutorials on integrating Firebase Authentication with Firebase Realtime Database on YouTube. 2) Solve the problems in integrating Firebase Authentication and Firebase Realtime database based on searched tutorials. 3) Learn to implement Firebase Realtime database based on the YouTube tutorial "Angular firebase crud - Angular firebase create read update delete with Realtime database". 4) Create a new Angular project PhaseLogin with CSS stylesheet format using npm command "ng new PhaseLogin". 5) Install Firebase packages in Angular project PhaseLogin using command "npm install @angular/fire firebase" based on YouTube tutorial "How to Connect Firebase Auth to an Angular App".	1) Unsuccessfully resolve the issues related to the integration of Firebase Authentication and Firebase Realtime Database due to an incompatible Angular version. 2) Successfully learned to implement Firebase Realtime database with an Angular project step-by-step. 3) Successfully create a new Angular project PhaseLogin with CSS stylesheet format using. 4) Successfully installed the required Firebase packages in Angular project PhaseLogin for the implementation of Firebase Realtime database.		
1/12/2023	Friday		1) Generate new Firebase project and setup Realtime Database in Firebase console. 2) Implement CRUD methods with Realtime Database in Angular project PhaseLogin's Sign Up and Login.	1) Create a new Firebase project Firebase Analysis and setup a Firebase Realtime database. 2) Add Firebase configurations of Firebase Analysis in Angular project PhaseLogin's environments/environment.ts. 3) Create .ts file and .html file for user registration feature in Angular project PhaseLogin that allow to create a new user's account methods in Firebase Realtime database. 4) Create a feature in .ts file for user registration feature in Angular project PhaseLogin that allow user to navigate to user login feature if they have existing account. 5) Create .ts file and .html file for user login feature in Angular project PhaseLogin that read and validate existing user's email and password in Firebase Realtime database. 6) Create a feature in .ts file and .html file for user login feature in Angular project PhaseLogin that read and validate existing user's email and password in Firebase Realtime database.	1) Successfully created a new Firebase project Firebase Analysis and setup a Firebase Realtime database. 2) Successfully created the .ts file and .html file for user registration feature in Angular project PhaseLogin that allow to create a new user's account methods in Firebase Realtime database. 3) Successfully created a feature in .ts file for user registration feature in Angular project PhaseLogin that allow user to navigate to user login feature if they have existing account. 4) Successfully created the .ts file and .html file for user login feature in Angular project PhaseLogin that read and validate existing user's email and password in Firebase Realtime database. 5) Successfully created a feature in .ts file and .html file for user login feature in Angular project PhaseLogin that read and validate existing user's email and password in Firebase Realtime database.		
2/12/2023	Saturday						
3/12/2023	Sunday						
4/12/2023	Monday	Week 11	1) Implement CRUD methods with Realtime Database in Angular project PhaseLogin's User Profile. 2) Implement CRUD methods with Realtime Database in Angular project PhaseLogin's Edit User Profile Form.	1) Create .ts file and .html file for display user profile feature in Angular project PhaseLogin that allow to display all the user's informations in Firebase Realtime database based on the retrieved userid. 2) Create features in .ts file for display user profile feature in Angular project PhaseLogin that allow the user to navigate to edit user profile feature and sign-out feature. 3) Create .ts file and .html file for edit user profile feature in Angular project PhaseLogin that allow the user to update the user's informations methods in Firebase Realtime database based on the retrieved userid. 4) Create features in .ts file for edit user profile feature in Angular project PhaseLogin that allow the user to save the edited user's informations or allow the user to cancel the edit process and return to user profile.	1) Successfully created the .ts file and .html file for display user profile feature in Angular project PhaseLogin that allow to display all the user's informations in Firebase Realtime database based on the retrieved userid. 2) Successfully created features in .ts file for display user profile feature in Angular project PhaseLogin that allow the user to navigate to edit user profile feature and sign-out feature. 3) Successfully created the .ts file and .html file for edit user profile feature in Angular project PhaseLogin that allow the user to update the user's informations methods in Firebase Realtime database based on the retrieved userid. 4) Successfully created features in .ts file for edit user profile feature in Angular project PhaseLogin that allow the user to save the edited user's informations or allow the user to cancel the edit process and return to user profile.		
5/12/2023	Tuesday		1) Implement router with PhaseLogin Angular project. 2) Test Firebase Realtime Database connection with PhaseLogin Angular project. 3) Redesign PhaseLogin Angular Project using CSS and Bootstrap icon.	1) Modify the routes of user registration, login, display user profile, and edit user profile features in Angular project PhaseLogin. 2) Edit the background color, border style, border radius, width, height, margin, content alignment, and padding in the .css file of Angular project PhaseLogin. 3) Edit the background color, border radius, text color, and padding of the submit button in the .css file of Angular project PhaseLogin. 4) Added the Bootstrap icons link copied from Bootstrap website to the PhaseLogin .html files.	1) Successfully modify the routes of user registration, login, display user profile, and edit user profile features in Angular project PhaseLogin. 2) Successfully edited the background color, border style, border radius, width, height, margin, content alignment, and padding in the .css file of Angular project PhaseLogin. 3) Successfully edited the background color, border radius, text color, and padding of the submit button in the .css file of Angular project PhaseLogin. 4) Successfully added the Bootstrap icons link copied from Bootstrap website to the PhaseLogin .html files according to the purposes and features.		
6/12/2023	Wednesday		1) Shared CRUD Login and Signup with Firebase progress with Organisation Supervisor. 2) Learned to implement Firebase Realtime Database in file upload feature. 3) Implement CRUD methods with Realtime Database in Angular project PhaseLogin's Upload File Form.	1) Upload the progress of the Angular project PhaseLogin to WhatsApp group in purpose of design checking by organization supervisor. 2) Research tutorials to learn to upload files using CRUD method and Firebase Realtime database from YouTube. 3) Create .ts file and .html file for upload file feature in Angular project PhaseLogin that allow to logged user to upload files and store the file under the user's userid in Firebase Realtime database. 4) Create a form using HTML that allow logged user to upload file and named their uploaded file.	1) Successfully uploaded the progress of the Angular project PhaseLogin to WhatsApp group in purpose of design checking by organization supervisor. 2) Successfully learned to upload files using CRUD method and Firebase Realtime database from YouTube. 3) Successfully created the .ts file and .html file for upload file feature in Angular project PhaseLogin that allow to logged user to upload files and store the file under the user's userid in Firebase Realtime database. 4) Successfully created an upload file form using HTML that allow logged user to upload file and named their uploaded file.		
7/12/2023	Thursday		1) Implement CRUD methods with Realtime Database in Angular project PhaseLogin's Display Upload File feature. 2) Implement CRUD methods with Realtime Database in Angular project PhaseLogin's Edit and Delete Upload File features.	1) Modify .ts file and .html file for display user profile feature in Angular project PhaseLogin to display a of files that logged user had uploaded to the Firebase Realtime database. 2) Create a toggle form in the .ts file for display user profile feature in Angular project PhaseLogin that allow user to edit the uploaded file and the name of the uploaded file. 3) Create a delete file feature in the .ts file for display user profile feature in Angular project PhaseLogin that allow user to delete the unwanted file from the Firebase Realtime database. 4) Test the upload, delete, and edit file features of the Angular project PhaseLogin.	1) Successfully modified the .ts file and .html file for display user profile feature in Angular project PhaseLogin to display a of files that logged user had uploaded to the Firebase Realtime database. 2) Successfully created a toggle form in the .ts file for display user profile feature in Angular project PhaseLogin that allow user to edit the uploaded file and the name of the uploaded file. 3) Successfully created a delete file feature in the .ts file for display user profile feature in Angular project PhaseLogin that allow user to delete the unwanted file from the Firebase Realtime database. 4) The Angular project PhaseLogin is successfully allow its user to upload file, delete unwanted file, and edit the existing file.		
8/12/2023	Friday		1) Search for solutions to resolve issues with the Edit and Delete Upload File features in the Angular project PhaseLogin. 2) Meeting with organization supervisor. 3) Understanding Laravel workframe and install Laravel using composer and node.js.	1) Research tutorials to address the issue of updating files when saving a toggle form that allows users to edit both the uploaded file and its name. 2) Joined an online meeting hosted by Mr. Devi Williasen Anggara. 3) Install and setup Main File for Laravel using Composer, Xampp, and PHP database.	1) Successfully solved the update file problem when saving the toggle form that allow user to edit the uploaded file and the name of the uploaded file. 2) Acknowledged the use of Larave workframe and PHP database in generating user registration and user login of Fhase.io. 3) Successfully installed and setup Main File for Laravel using Composer, Xampp, and PHP database.		
9/12/2023	Saturday						
10/12/2023	Sunday						
11/12/2023	Monday	Week 12	1) Learn to create a menu with a form using Laravel 2) Create menu with two submenus for the Laravel dashboard sidebar.	1) Research tutorials to learn to create a menu with a form using Laravel from YouTube. 2) Create routes by creating a new navigation item 'Blank' based on YouTube tutorial 'Laravel 10 - Add User menu, show user list, and create user form - Part 5'. 3) Create routes by creating submenu contents 'Upload Form' and 'File List' based on YouTube tutorial 'Laravel 10 - Add User menu, show user list, and create user form - Part 5'.	1) Successfully learned to create a menu with a form using Laravel from YouTube. 2) Successfully created routes by creating a new navigation item 'Blank' based on YouTube tutorial 'Laravel 10 - Add User menu, show user list, and create user form - Part 5'. 3) Successfully created routes by creating submenu contents 'Upload Form' and 'File List' based on YouTube tutorial 'Laravel 10 - Add User menu, show user list, and create user form - Part 5'.		
12/12/2023	Tuesday		1) Create .blade.php file to display upload file form when click on 'Upload File' submenu. 2) Create .blade.php file to display uploaded file list when click on 'File List' submenu. 3) Test if 'Upload File' submenu display upload file form and 'File list' submenu display uploaded file list.	1) Create a form.blade.php file under views. 2) Create upload file form for form.blade.php file using HTML and CSS. 3) Create form.blade.php file using command 'php artisan make form. 4) Create upload file form for form.blade.php file using HTML and CSS. 5) Create FormController.php for Laravel using command 'php artisan make:controller FormController' and add routes in web.php. 6) Check on the Laravel app using command 'php artisan serve'.	1) Successfully created form.blade.php file under views. 2) Successfully created upload file form for form.blade.php file using HTML and CSS. 3) Successfully created form.blade.php file using command 'php artisan make form. 4) Successfully created upload file form for form.blade.php file using HTML and CSS. 5) Successfully created FormController.php for Laravel using command 'php artisan make:controller Form' and add routes in web.php. 6) Both 'Upload File' and 'File list' submenus successfully display both upload file form and uploaded file list.		
13/12/2023	Wednesday		1) Learn to implement CRUD methods into Larave's submenus 'Upload File' and 'File List'. 2) Implement CRUD methods in submenu 'Upload File'.	1) Research tutorials on implementing CRUD methods in Larave on YouTube. 2) Create file model and configure migrations for upload file using command 'php artisan make:model File -m'. 3) Create fileUpload() function in FormController.php to create new table in PHP database.	1) Successfully learned to implement CRUD methods in Larave through tutorials on YouTube. 2) Successfully created file model and configure migrations for upload file using command 'php artisan make:model File -m'. 3) Successfully created fileUpload() function in FormController.php to create new table in PHP database.		
14/12/2023	Thursday		1) Implement CRUD methods in submenu 'File List'. 2) Implement 'Delete File' and 'Edit File' on the upload file list in submenu 'File List'.	1) Create fetchData() function in FormController.php to read all tables available in PHP database. 2) Create editFile() and updateFile() functions in FormController.php and create editFile.blade.php in views to edit specific data in PHP database. 3) Create confirmDelete() and deleteFile() functions in FormController.php and create confirmDelete.blade.php in views to delete specific data in PHP database.	1) Successfully created fetchData() function in FormController.php to read all tables available in PHP database. 2) Successfully created editFile() and updateFile() functions in FormController.php and editFile.blade.php in views to edit specific data in PHP database. 3) Successfully created confirmDelete() and deleteFile() functions in FormController.php and confirmDelete.blade.php in views to delete specific data in PHP database.		
15/12/2023	Friday		1) Learned to solve issues related to the implementation of CRUD methods in Larave's submenus 'Upload File' and 'File List'. 2) Solve problems with implement of CRUD methods in Larave's submenus 'Upload File' and 'File List'.	1) Research tutorials on implementing CRUD methods in Larave's submenus 'Upload File' and 'File List' on YouTube. 2) Reinstall and setup Main File for Larave using Composer, Xampp, and PHP database due to problem with login with Larave demo serve. 3) Create FileUploadController.php for Larave using command 'php artisan make:controller FileUploadController' and add routes in web.php. 4) Create File.php model for Larave using command 'php artisan make:model File'. 5) Create createForm() and fileUpload() functions in FileUploadController.php and create file-upload.blade.php in views to create new data in PHP database.	1) Successfully learned to address issues related to the implementation of CRUD methods in Larave's submenus 'Upload File' and 'File List' through tutorials on YouTube. 2) Successfully reinstall and setup Main File for Larave using Composer, Xampp, and PHP database due to problem with login with Larave demo serve. 3) Successfully created FileUploadController.php for Larave using command 'php artisan make:controller FileUploadController' and add routes in web.php. 4) Successfully created File.php model for Larave using command 'php artisan make:model File'. 5) Successfully created createForm() and fileUpload() functions in FileUploadController.php and create file-upload.blade.php in views to create new data in PHP database.		
16/12/2023	Saturday						
17/12/2023	Sunday						