



Assessment Brief

Module Title:	Advanced Web Application Development
Module Code:	CI609
Author(s)/Marker(s) of Assignment	Marcus Winter

Assignment No:	1
Assignment Title:	Advanced web application
Percentage contribution to module mark:	100%
Weighting of component assessments within this assignment:	n/a
Module learning outcomes covered:	LO1. Research concepts, technologies and APIs used in the development of advanced Web applications LO2. Appraise technology choices and design suitable Web application architectures and APIs LO3. Specify, design and develop advanced, integrated Web applications

Assignment Brief and Assessment Criteria: See following pages.

Date of issue:	03/10/2022
Deadline for submission:	09/12/2022 at 3pm Note: Students are allowed to submit work within two weeks of the published deadline, or the last working day immediately prior to the feedback date if this is shorter than two weeks. Late work is capped at the pass mark of 40%.
Method of submission:	e-submission via MyStudies
Date feedback will be provided	20/01/2023

1. A copy of your coursework submission may be made as part of the University of Brighton's and School of Architecture, Technology & Engineering procedures which aim to monitor and improve quality of teaching. You should refer to your student handbook for details.
2. All work submitted must be your own (or your team's for an assignment which has been specified as a group submission) and all sources which do not fall into that category must be correctly attributed. The markers may submit the whole set of submissions to the JISC Plagiarism Detection Service.

Advanced web application development project

In this individual assignment you will architect, design, develop and deploy an advanced Web application. You can choose to either work towards a set application brief (see Options 1-3 below) or to develop your own application idea, subject to approval by module tutors to ensure the technical challenge is appropriate for the intended learning outcomes (Option 4).

For any option you choose, you will carry out required technical research in order to select suitable technologies and tools supporting your development project. To support you in the process, there will be lectures and tutorials where we discuss related concepts and where you can request formative feedback on your work.

You are free to complete this assignment with vanilla web development languages or using client- and/or server-side frameworks and libraries. Either way, must carry out suitable research to inform your technology choices and discuss them in your report or presentation.

Choose ONE of the following options:

Option 1: Zap App

A research team studying threats to Pangolins in the wild commissions you to develop a mobile-optimised progressive web application to record Pangolin sightings and mortalities.

Pangolins are the world's most trafficked mammal and are threatened solely by human impacts. In addition to tracking and poaching, many pangolins in Africa face the risk of being killed on roads and by electric fences. The app should enable game rangers and local communities across Southern Africa to log pangolin sightings and mortalities, enabling the research team to study these threats.

This app should allow users to take a picture or select one from the gallery and select whether they have seen the Pangolin alive or dead. If dead, users should select the type of mortality (fence death: electrocution; fence death: caught on non-electrified fence; road death; other). Users also should have an option to add additional notes, such as fence or road type. The app should automatically pick up the user's location and submit it together with the recorded information.

As user may be in remote areas with no signal when they encounter pangolins, the app should provide key functionality offline. It should be able to store information locally and upload that information later when they have an internet connection.

A client-server solution involving a REST API is preferred to allow other organisations to develop their own client applications against the system.

Related links:

https://developer.mozilla.org/en-US/docs/Web/API/Geolocation_API

<https://developers.google.com/web/fundamentals/media/capturing-images>

https://developer.mozilla.org/en-US/docs/Web/API/Navigator/Online_and_offline_events

Option 2: Skip App

A charity promoting re-use and sustainability commissions you to develop a mobile-optimised progressive web application to share, map and find materials and household goods found in skips or offered for free collection in the street.

By default, users should be presented with a map of the local area showing markers for available objects. When touching an object marker, a detail view should pop up with the object image, short description, a button to mark the object as taken and a button to close the detail view (and revert to the map). Once an object is taken, it disappears from the map at the end of the day.

The map interface should also provide an Add button to upload new objects. After pressing the Add button, users should be presented with a screen where they can take a picture or select one from the gallery and add a short description. The app should automatically pick up the user's location and submit it together with the object image and description. After the user submits the new object, the app should revert to the map interface (now showing an additional marker for the newly submitted object).

A client-server solution involving a REST API is preferred to allow the charity to develop an independent admin and analytics interface against the system.

Related links:

<https://leafletjs.com/>

https://developer.mozilla.org/en-US/docs/Web/API/Geolocation_API

<https://developers.google.com/web/fundamentals/media/capturing-images>

Option 3: Mind App

A mental health charity commissions you to develop an anonymous, secure, mobile-first, progressive web application enabling users to log and review their emotional state.

By default, users should be presented with a screen where they can select a mental state (as smiley faces) and add a comment to further qualify their state or describe possible reasons. The app should automatically add the date, time and current weather (e.g. clear sky, 15C), and store this information together with the user-provided data in local storage.

Users should be able to view past entries like a blog (last entry first, date/time as headings) to support their reflection. They should be able to share this information as encrypted JSON data with a mental health professional via the system's native sharing mechanism.

[Recipients will require the user's PIN (see below) to view the encrypted data in a separate viewer app. PIN sharing and the separate viewer app are not part of this assignment.]

In order to promote trust among users and ensure privacy of this sensitive information, all data is stored locally rather than transmitted to a server. Access to the app should be secured with a PIN, which users can set when first starting the app from their home screen. The same PIN should be used to encrypt log data before it is stored or transmitted.

Related links:

<https://openweathermap.org/api>

<https://github.com/afitzek/crypto-js>

<https://developer.mozilla.org/en-US/docs/Web/API/Navigator/share>

Option 4: Own App

Develop your own web application idea. Your idea should have sufficient technical challenge to demonstrate that you meet the learning outcomes of this module. Write an informal project proposal (max 2 pages) outlining the general idea, functionality and preliminary technical research carried out to make sure the project is feasible and you can deliver it in the allocated time.

Report or Presentation

You can choose one of the following assessment types to document and critically reflect on your development project:

- a) A written report discussing the web application architecture and design, technology choices, implementation details and deployment (approx. 3,000 words)

- OR -

- b) A slideshow presentation to the academic tutors discussing the web application architecture and design, technology choices, implementation details and deployment (25 minutes including questions).

Deliverables

Your web application **must be hosted on your personal university web space**

e.g. <https://username.brighton.domains/ci609/assignment1/index.html>

The assignment must be submitted online, through the CI609 assessment area on MyStudies. Please submit the following:


1. **In the text field** of the submission form, submit the **URL** of your web application.
Make sure the link works! (If your web application requires users to login, also include test user **login details** in case any user registration procedure fails).
2. A **single ZIP file** with the following name and content:

Use the **exact** names, folder structure and file types as specified below.
Marks will be deducted in the presentation category for non-conformance.

ZIP file name:

<lastname>_<firstname>_<studentnumber>_CI609_assignment.zip




ZIP folder structure, content and file names:

 report

Report or presentation slides for your development project,
as a single PDF file named *report.pdf* -or- *presentation.pdf*

 project

Complete source code, assets, manifests, configuration files, build scripts, database exports, API specifications, etc. used to create your web application.

Use relevant sub-folders inside the project folder for the different parts of your application ( client,  server,  database, etc.)

Marking criteria (equal weighting for all criteria)

GRADE	Independent research	Fitness for purpose	Technical achievement	User interface	Report or presentation
A (70% and above)	Excellent: In-depth, critical research into key aspects relevant to the specification, design and development of the web app.	Excellent: Application fully meets the brief (options 1-3 above or option 4 agreed with tutor) and runs reliably without errors.	Excellent: Appropriate application architecture and technology choices, high quality code, professionally built and deployed.	Excellent: Fully responsive, well presented, meeting usability and accessibility guidelines to the highest standard.	Excellent: Well-structured and convincing discussion, covers all relevant aspects, presented to the highest standard, submission fully meets the brief.
B (60-69%)	Good: Thorough research into key aspects relevant to the specification, design and development of the web app.	Good: Application fully meets the brief (options 1-3 above or option 4 agreed with tutor) and generally runs without errors.	Good: Suitable application architecture and technology choices, good quality code, successfully deployed.	Good: Responsive, well presented, meeting usability and accessibility guidelines to a high standard.	Good: Overall well-structured and convincing discussion, covers all relevant aspects, presented to a high standard, submission fully meets the brief.
C (50-59%)	Satisfactory: Competent research into aspects relevant to the specification, design and development of the web app.	Satisfactory: Application mostly meets the brief (options 1-3 above or option 4 agreed with tutor) and generally runs without errors.	Satisfactory: Acceptable application architecture and technology choices, some issues with code quality, successfully deployed.	Satisfactory: Responsive, fairly well presented, meeting most usability and accessibility guidelines.	Satisfactory: Fairly well structured and convincing discussion, covers most relevant aspects, well presented, submission mostly meets the brief.
D (40-49%)	Adequate: Research into several aspects of the project.	Adequate: Application meets some of the brief (options 1-3 above or option 4 agreed with tutor) and mostly runs without errors.	Adequate: Shortcomings in architecture, technology choices, code quality or deployment.	Adequate: Responsive, some usability or accessibility problems.	Adequate: Some weakness in structure and/or discussion, covers most relevant aspects to some extent, submission in some aspects meets the brief.
E (30-39)%	Unsatisfactory: Little research into relevant aspects of the project.	Unsatisfactory: Serious shortcomings in functionality and/or robustness.	Unsatisfactory: Serious shortcomings in architecture, technology choices, code quality and/or deployment.	Unsatisfactory: Serious problems with responsiveness, usability and/or accessibility.	Unsatisfactory: Weak structure and/or discussion and/or presentation and/or submission does not meet the brief.
F (0-29%)	Work submitted, but criteria for grade E not met.	Work submitted, but criteria for grade E not met.	Work submitted, but criteria for grade E not met.	Work submitted, but criteria for grade E not met.	Work submitted, but criteria for grade E not met.