Summative Assignment

Module code and title	COMP1101 Programming (Black)
Academic year	2025-26
Coursework title	Web Development including Peer Assessment
Coursework credits	10 credits
% of module's final mark	50%
Lecturer	Steven Bradley
Submission date*	Thursday, 29 January, 2026 14:00
Estimated hours of work	20
Submission method	Ultra

Additional coursework files	*None*
Required submission items and formats	Source code (all zipped) - HTML and CSS and any media - Client and server side JavaScript - package.json including test and pretest scripts - jest test cases e.g. app.test.js - documentation of API - demonstration video Should not include \`node_modules\` in submission

* This is the deadline for all submissions except where an approved extension is in place. For assessments carried out in weekly practicals or other scheduled sessions, the date shown will be the Monday of the week in which the assessments take place.

Late submissions received within 5 working days of the deadline will be capped at 40%.
Late submissions received later than 5 days after the deadline will receive a mark of 0.
It is your responsibility to check that your submission has uploaded successfully and obtain a submission receipt.

Your work must be done by yourself (or your group, if there is an assigned groupwork component) and comply with the university rules about plagiarism and collusion. Students suspected of plagiarism, either of published or unpublished sources, including the work of other students, or of collusion will be dealt with according to University guidelines (https://www.dur.ac.uk/learningandteaching.handbook/6/2/4/).

COMP1101 Programming (Black) Summative Assessment 1

Term 1 Programming Exercise Outline

- Submission of code and video by 14:00 29 January 2026
- Submission of peer reviews by 14:00 19 February 2026
- Return by 26 February 2026
- Contributes 50% of module marks
- Includes peer review feedback which you will be allocated
- This is an individual piece of work

Subject-specific Knowledge

- Interaction between JavaScript programs and the Document Object Model (DOM)
- Using control statements to loop and make decisions
- An understanding of the nature of imperative programming in the object-oriented style
- A knowledge and understanding of good programming practice (for example, reuse, documentation and style)
- Building collections of data within a program and using JavaScript Object Notation (JSON)
- Making programs robust through the use of exceptions and exception handling
- A knowledge and understanding of good programming practice (for example, reuse, documentation and style)

Subject-Specific Skills

- an ability to realise solutions to problems as working JavaScript programs
- an ability to apply reuse by exploiting predefined components
- an ability to use software tools related to programming (programming environments, code management, documentation tools, etc.)

Key Skills

- an ability to communicate technical information
- an ability to recognise and apply the principles of abstraction and modelling

Task summary

- Construct a dynamic web site for an application of your choosing

- Use static HTML pages loading dynamic JSON content from server via AJAX
- Server written in nodejs to provide JSON through REST API
- Prepare a 2 minute video demonstrating your code
- Do a code quality review of four other submissions

Dynamic web site

- Choose any application domain as long as it includes at least two kinds of entity e.g.
  - pictures
  - people
  - places
  - events
  - comments
- Could be e.g. club, social, health, gallery
- If you are not sure then ask me

Static HTML loading JSON via AJAX

- 'Single page app': page content loaded as JSON via AJAX
- Can have more than one page e.g. for user and admin
- Should provide clean and simple User Experience (UX)
- Should be responsive i.e. work well on desktop and mobile
- Recommend using front-end framework such as Bootstrap, Foundation
- DO NOT use non-standard language extensions e.g. React, TypeScript

Message sequence chart

Server provides JSON through a REST API

Each entity type (e.g. picture) has

- GET method to list/search (returns a list of ids and names)
- GET method for individual details (includes details of related entities)
- POST method to add new entity
- Document your API in the style of the ChatGPT API
- see other good API docs
- Response provided as JSON
- Content-type needs to be correct
- HTTP codes should be correct: use 200, 400 or 403 (if using authentication)

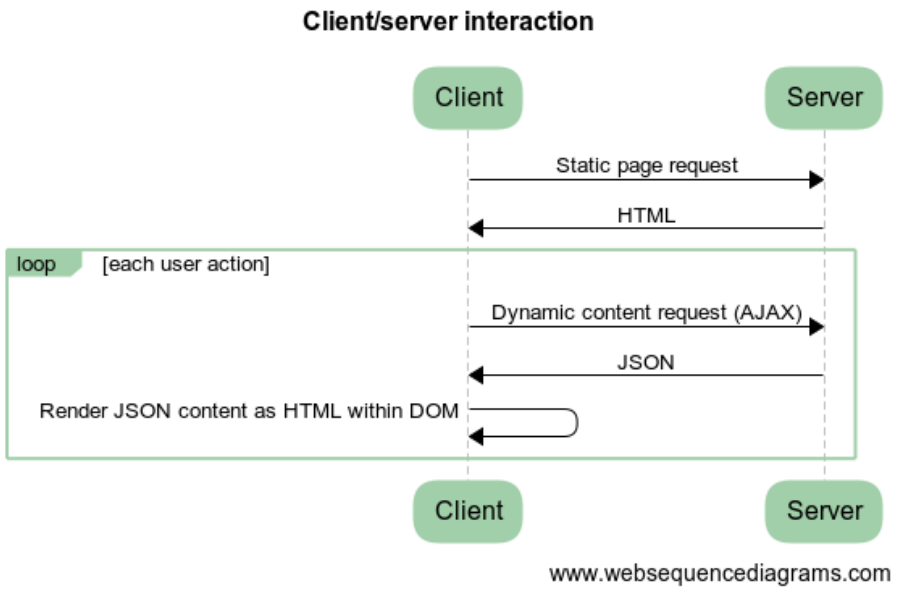
Client/server interaction
Client
Server
Static page request
HTML
loop
[each user action]
Dynamic content request (AJAX)
JSON
Render JSON content as HTML within DOM
Client
Server
www.websequencediagrams.com

Figure 1: Message Sequence Chart showing Client server interaction with AJAX

Server written in nodejs

- Use npm for management
- Make sure you use –save or –save-dev option with packages you add
- Write jest test cases: run with `npm test`
- Recommend using express

Submission

Source code (all zipped)

- HTML and CSS and any media
- Client and server side JavaScript
- package.json including test and pretest scripts
- jest test cases e.g. app.test.js
- documentation of API
- demonstration video

Should not include `node_modules` in submission

Assessment Criteria

Equally weighted 9% each

- Client-side functionality
- Client-side quality
- Server-side functionality
- Server-side quality
- Video presentation

Client-side functionality criteria

- User Experience (UX): clean layout and minimal clicks/entry required
- App complexity: entities can be listed and edited
- 'Single page' style: asynchronous updates
- Staff reviewed

Client-side quality criteria

- Standards compliant (HTML5, JavaScript not React)
- Responsive to different viewport sizes
- Gracefully handles server disconnection
  - useful error messages
  - recommences on server restart
- Peer reviewed; staff moderated

Server-side functionality criteria

- More than one entity type, with relationships
- REST API provides each entity with appropriate GET/POST methods
- Installs with `npm install`
- Starts with `npm start`
- Staff reviewed

Server-side quality criteria

- Successful jest tests with good coverage (run with `npm test`)
- Testing includes content-type and HTTP code
- Completeness of API documentation
- Peer reviewed; staff moderated

Video Presentation

- Submit a 2 minute (max) video demonstrating your software
- Include demonstration of how to start the program
- All functionality will be assessed by what is demonstrated in the video
- If it is not demonstrated in the video, you will not get a mark for it
- Quality of video presentation will be marked separately from functionality:
  - Structure; Visual Presentation; Audio explanation

- Lose 10% of marks for every 10 seconds over 2 minutes
- Staff reviewed

Peer Review Marking

5% of the module marks are awarded for peer reviews

- Completion of all reviews on time
- Professional and helpful reviews
- The average student tends to get about 65%
- 65% is on the good/very good boundary of the marking conventions p15

How to do the assignment

- Design HTML
- Design web service
- Join with Fetch
- Read the FAQ

COMP1101 Programming (Black) Summative Assessment 1 FAQs

General

Can I use code from elsewhere and how should I reference it?

Using pre-existing code modules on the server side is recommended. As long as it is installed properly with npm so that the dependency is included in your package.json file you don't need to do anything else.

If you re-use, adapt or modify code taken from other sources (e.g. StackOverflow) then put a comment in your code pointing to the URL of the source. If you include code from a LLM (e.g. ChatGPT) include a comment with the prompt you used to create it and the LLM you used (including version)

What about using copyrighted content?

Work legally. Make sure you follow the terms of the license i.e. provide attribution if required. There are some exceptions to copyright but you might want to share your work publicly (after the June exam board) so be careful not to assume everything is for private study. There is plenty of material out there that you are allowed to use.

Client-side functionality criteria

Do we need an authentication aspect, like username and password?

Not recommended: if you are going to do authentication you should do it properly, and it's very unlikely that you could write something yourself that would be robust. My usual advice would be to use an external authentication service (e.g. through firebase), but that requires have tokens etc embedded into your code. Sharing that with others would most likely be outside the terms and conditions of its use. So if you want to do authentication, just include a function to authenticate, which maybe pops up an alert saying 'this page is private' or something like that.

Are there marks for complexity?

Yes, but only up to what is requested in the brief. So, for example, if you provide more than three entities you don't gain marks, but you will lose marks if you have fewer than two entities.

Should we use CDNs for things like bootstrap?

You can but you don't have to. It will not affect your marks, only your download time (positively) and offline access (negatively).

What is the best way to attach event listeners to multiple buttons?

Use a class if buttons are doing the same action on different things (e.g. delete): use the event parameter in the event handler to extract the object. If the actions are all different then add by hand with ids. Do it after DOM is loaded to make sure that all the elements exist, possibly by using `defer` attribute on your script tag.

Client-side quality criteria

Can I use react to build the web-site?

No: peer reviewers probably don't know what this is, and so may find it difficult to assess the quality of your solution. React is commonly used, but is not a standard. Part of the quality assessment criteria is standards compliance.

Can I use jQuery to build the web-site?

Not recommended: in years gone by jQuery was essential to make things like event handling and AJAX work across browsers. However these days modern browsers offer things like the Fetch API which handle this. Internet Explorer support is not the issue it once was, because it now has such a small market share, and polyfills are available to make it look like it has Fetch. Some systems still do use jQuery (e.g. Bootstrap 4) but many have removed it as a requirement (e.g. Bootstrap 5) so don't use it for a new project.

Will I get marked on documentation of the client-side code?

No

What do you mean by 'gracefully handles server disconnection'

Basically your client-side code should do something sensible if the connection to the server goes down (as it might do if it were connected via the internet). It should display an informative message to the user, and maybe try again later. You can test this by stopping the server and trying to interact with it through the client. Once the server is started up again the client should be able to carry on as before. So follow the sequence: start server, load page, stop server, get helpful message, restart server, works again (without reloading page).

Is it possible to use TypeScript or should we just stick with normal JavaScript?

TypeScript needs transpilation tools to be installed and uses different code quality rules so for this assignment you should stick with plain JavaScript - you are free to do what you want in the second assignment.

Do Alt tags added by users need to be unique?

No

Does it have to be in British English?

No

Server-side functionality criteria

Are we allowed to include additional modules via NPM that provide additional functionality?

Yes this is definitely a good idea, as it makes the code easier to read, more robust and more maintainable. Bear in mind that some frameworks (e.g. React) that are installed via npm essentially use a different (non-standard) language, so are difficult to read for non-experts and should not be used.

How are we advised to save the data, i.e using database?

If this were for real then a database of some kind would be the best way to store data. However, databases are not part of this course. My recommendation would be to write functions for saving the state, which just write to file a JSON string representing the state. This would not work with multiple users but would be enough for simple testing.

How to store entities? Two different files?

Not necessarily, one is fine, as long as it contains all of the relevant details and relationships.

What do you mean by an 'entity type'?

If this were object oriented programming they would be referred to classes. It is a 'type' or 'kind' of thing. The different entity types will have different information stored for them. If your app is about poets and their poems then you would have two entitity types: 'poet' and 'poem'. For each poet you might want to store their id, name(s), date of birth, url of image. For each poem you might want to store an id, the title, date of writing, and the text. The relationships between the entities would be of the form poet1 authored poem2.

Exactly how you store the information about the relationship is up to you: you could store the author id with the poem, or store a list of poems ids with the author, or have a separate store relating the two. In either case, when you get the details of an entity you should include everything, including the relationships.

Do we need to set up the http server so it's not on the localhost?

No, although you might like to do that after the June exam board if you would like other people to use your app.

Do GET/POST have to be exactly what it says?

They should: list (GET), details of one (GET), add one (POST). If not sure, ask me.

Server-side quality criteria

Is the testing solely on the server.js file or on the other js files the website uses also

You only need to test the server side javascript: client-side testing is a whole different can of worms.

Does automatic testing have to test every component?

Yes, to get full marks you need to have detailed testing of every GET and POST method you write. You can get some marks by partial testing.

How do you test if it works on another machine?

The main reasons for it not to work on another machine are if you use absolute paths, or use a back (Windows) or forward (Mac/Linx) to separate directories in the path. `__dirname__` is your friend, and use path.sep and path.join. You can always use a CIS lab machine to test whether it works on another machine. Many are dual-boot so you can try out Windows and Linux.

Are we allowed to use an API documentation generator like Postman or must we create our own documentation from scratch?

Yes, if you have a good tool for API generation such as https://learning.postman.com/docs/postman/api-documentation/documenting-your-api/ that would be fine, and a good idea.

Will adding comments to the API be sufficient for API documentation, or would it be preferable to use something like postman or ‘https://www.npmjs.com/package/node-api-doc-generator’ (an npm package)?

You should not have to read the code to see the API documentation. You could write it in HTML by hand, or you could use a tool to do that. The postman tool looks good and well maintained, but the npm tool looks less good: not updated in a long time, virtually no history of maintenance.

What do we need in the API documentation?

For the API documentation the ideal is something like the documentation of the ChatGPT API which lists the methods in the API and then provides the details for each method including details of parameters and response.

Do I need to put comments in my code?

May people view good comments as important in code quality, but some argue that could be should be self-explanatory, and that comments are problematic because they tend not to be updated as the code is maintained. The use of comments is up to you, and is not included in the assessment for this piece of work. The only time you *should* write a comment is if you’ve taken a code snippet from elsewhere (e.g. Stack Overflow, ChatGPT) and included it directly in your code. In this case you definitely should write a comment explaining where you got the code from. This will remove the risk of you being accused of academic misconduct/plagiarism. Also it is good professional practice, as the best solution to a problem may change over time, so it is helpful for people reading your code to know why you did it in a particular way, and to check if there are any updated ways of solving the problem.

Video Presentation

You mentioned that for every 10 seconds after 2 minutes we will be losing 10% of our marks, does this apply if our video is anywhere from 2 mins 1s to 2 mins 9 seconds long ?

Yes, basically it’s 1% per second.

What information are we supposed to include in the video?

The video only needs to cover the points that are not peer assessed. Treat it as a sales pitch for the criteria listed under client-side functionality and server-side functionality. You don’t need to show every single thing that your site does, just show how it meets the requirements. So things like HTML validation, automated testing and the API documentation do not need to be covered. It is difficult to keep videos anonymous, so your peer assessors will not see the video.

What software should we use to record the presentation, and where could we find it?

The university provides panopto (which is used for sharing lecture recordings) for recording and simple editing and gives some instructions for use. Freely available desktop tools include OBS Studio and daVinci Resolve for recording and editing, which are both powerful but have more of a learning curve. MacOS provides QuickTime player and iMovie for recording and editing. TechRadar have a list of screen recorder software for other alternatives.