

### **Plagiarism**

- The work needs to be your own. You should not use Generative AI tools to generate code, or use code from YouTube, GitHub repositories, etc. You can use Generative AI to help you in the debugging process for code, doing code reviews, etc. You can also use it for generating product images.
- All code will be reviewed carefully by the lecturer.
- You may need to explain your approach, reasoning and code with the lecturer via a Teams call – prior to the work being given a mark.

#### **A. Overall Aim**

The overall goal for the project is to produce a prototype that shows a working, fully featured, **sensible** website that sells a product or service online:

- selling a product online
- making a reservation/booking
- getting a delivery

The prototype you build is NOT what you delivered in Year 1, Semester 2 of “*Web Design*”. It may use content, graphics and styling – but it will be implemented using the technical stack we have introduced – namely Node, Express, Handlebars, Bootstrap. Submissions not using these core elements will lose marks. Please note *Section F – Things that will make you lose marks*

#### **B. Deliverables**

Main deliverable is a working web site which uses Node, Express, Handlebars and Bootstrap 5.x. The site must be designed from a mobile first perspective and should be fully responsive.

- The site should be designed to work well in all screen sizes, with appropriate content and layout for major breakpoints (mobile/tablet/desktop).

The key dates for these deliverables are noted specifically in Moodle, all of these deliverables will also be uploaded to Moodle (zip files)

- Wireframe and Project Proposal
- Fully completed project (source files + Project report)

### **C. Content**

- Home, Contact, About Us, Shop (with product catalogue of at least 6 items), Login/Register
- User Registration - Forms on the site will include Login. You might also consider “Register” – if your JavaScript functionality allows for the creation of an account. “Forgot Password” is considered out of scope.
- Other forms relating to user registration could include personal details and billing address/delivery address – as long as these are relevant to the service being delivered.
- Check out page

### **D. Design and Layout**

- Menu must be obvious and appropriate for the viewport size.
- Images should be delivered in appropriate formats and resolutions.
- You can include other advanced features from Bootstrap 5.x (e.g. carousel on home page of top selling products) or elsewhere – as long as it is appropriate to the site.

**E. Detailed Marking Rubric (30% of your final exam result)**

Area	Rubric		Mark	Description
Design & Layout			8%	Responsiveness, content, flow, accessibility
	Responsive at different sizes?	2%		3/4 cols (desktop), to 2 cols (tablet) to 1 col stacked (mobile). Needs to be demonstrable preferably on shopping page but responsiveness applies to complete site.
	User flow makes sense?	2%		Have you followed suggestions from Section H?
	Content makes sense to the site?	2%		Do I know what the site is selling? Is it in Euros? What is the copy (text content) like? Do images makes sense and are relevant? Do they look real?
	Additional custom styling?	2%		To make project look less like a "Bootstrap" site
Technical			10%	Use of Bootstrap/JavaScript/GitHub/SASS, etc
	SASS used?	1%		Week 9 topic – generated CSS being used by actual prototype
	Files minified?	1%		Week 11 topic – minified includes being used by actual prototype
	GitHub usage	2%		Commits show good, incremental addition of functionality over lifetime of development (late November onwards)
	Functionality beyond sample code provided	6%		Points awarded for sensible functionality above what was requested – such as things mentioned in Section I, Point 1
Forms usage			6%	Required breadth of forms using validation, accessibility
	User Login	1%		Login works. Labels associated, alert messages hidden and shown appropriately for user login failure. Changes to 'Logout' etc. if success
	Checkout/Payment	2%		Labels associated, card validation, appropriate success/error messages.
	User Details	3%		Labels associated, appropriate error messages, details populated to checkout
Project Report			6%	
	All sections addressed?	4%		Thought put into the responses?
	Wireframes	2%		Were wireframes submitted? Was Care and attention put into them?



## F. Things that will make you lose marks

- Plagiarising code/non attribution of images
- Not making GitHub repository public
- Not using Node, Express, Handlebars, Bootstrap
- Developing functionality that do not make sense
- Giving me the project you developed for Year 1, Semester 2 – ‘Web Design’
- Not uploading the materials I asked you to upload
- Not reading *Section I – Final Checklist*

## G. Tools you will use

- Bootstrap 5.x, CSS customisations/SASS
- Node/Express/Handlebars/JavaScript
- Persisting (saving) Data
  - Use LocalStorage object(getItem,setItem) for persisting data and state from page to page on your site. You will use this in conjunction with JSON.stringify() and JSON.parse().
  - OR/AND Handlebars objects

## H. User Flow

Outlined below is the user journey that is expected from your project. You may refine this yourselves to add extra functionality and this is also fine.

- Navigate to the home page of the site – know immediately from the content and the logo on the screen what this site is selling/producing/offering.
- I look to the contact details for this store – do they look reputable? Do they provide address contact details/map to show where they are located. Is there a contact form so I can get in touch with them about my order?
- I look to the About section of the site – I want to find out a little more about this company? Who are they?
- Finally I browse to the Shop or product catalogue on the site. It has at least 6 offerings for sale – so I know they must be genuine.
- The products/services are laid out well. There is a title, image, short description; where relevant there are a few images of the product.
- Depending on the product/service being offered – there may be options associated with it; for example small, medium, large or maybe there are different colours? Or maybe the quantity? But you should only add these options if you want to get top marks!!!
- I’m happy with the product I have chosen – so now I click “Add to Cart”; I see that the Items is now set at 1 in my cart – and the cost of the total items has also jumped accordingly.
- I go to the checkout. I see that my delivery address has been defaulted in automatically from my address given in my personal details.
- I enter in dummy credit card detail

- I click “Submit Payment” – and then a message appears on screen “Thank you for your order”.

### *Background to the User Flow*

- You could use LocalStorage/JSON to store details associated with a “registered” user – username, email, password, address, etc.; this means for example that during a session the user could pretend to update their address – and this change would be saved for the duration of the session.
- For the user login screen – to easily allow testing for a registered user – use the value field for username and password to automatically default in a valid username and password for using the site; if these details were adjusted while testing – then the user would not be able to see personal details, etc..
- You could use LocalStorage/JSON to store details of the products in your catalogue.
- You will have a minimum of 6 products or so to sell. You can have an individual product page for each item if you wish – if you decide that you wish to display more detail of the product. However you will probably just have one overall shop page which displays all of your products on one page – with “Add to Cart” at this level.

## **I. Final checklist**

*Things to consider for a good implementation (these are common issues that occur with projects every year) – You must ensure that you read this before you submit your work...*

1. Potential ways to get extra marks could include examples such as listed below – there could be others:
  - a. Storing product data in a JSON file so it is not hardcoded.
  - b. Storing user details in a JSON file so it is not hardcoded.
  - c. Allowing the user to see the individual items in the checkout – this is really only possible if you use try to do (a) above.
  - d. Allowing the user configurability to the product (e.g size, colour)
  - e. Allowing the product to have a separate product page
  - f. Allowing the user to modify quantities in the checkout page
  - g. Building a registration feature – it is not provided in sample code. Would probably involve storing details in a JSON file.
2. Use Bootstrap – You must use Bootstrap to provide a responsive site.
3. At least 6 products - use relevant image, text description, price. Using GenAI for product images is fine. Mention in report you have done this.
4. If you are using images from existing companies or websites, you should give attribution appropriately in footer, etc. You should also mention this in your report. If your work is in a public GitHub repo or you are using as part of a job application process, you may be asked about how you sourced images, have they been attributed correctly, do they have creative commons licence, etc.
5. If you are using an API for products - and they are in USD, then contact page will be a business based in the US.
6. Pay attention to simple detail – e.g. if price in EURO then contact us page is in EU..

7. Final Report - make sure you mention the stuff you have done, detail what is not obvious to me as I review your work. For example, rewriting JavaScript code from scratch, using API for products, optimising site (e.g. minifying files, optimising images for desktop and mobile). Put your best forward and be enthusiastic, sharing and informative about what you have developed!
8. The source uploaded to moodle in a zip file - make sure it works - unzip to a different location - does everything load and link?
9. Github repo – make sure it is public. I do not want to have to email you about this....
10. From design POV, make CSS changes nearer end of project. Make sure you don't reinvent the wheel, let Bootstrap do its job. You should not be modifying core bootstrap files, add your own custom.scss or custom.css
11. If you don't do things suggested in the project template - then state this. Do not leave an empty heading. This will make me think you have neglected to complete it. For example, if you do not include link to GitHub repo – this suggests two things to me. You forgot to include it, which means I have to email you over Christmas holidays – which would be very annoying for both of us. If it does not exist – let me know in report!
11. Breakpoints - at least have 3/4 columns, 2 columns, 1 column. By columns, I mean product "cards". You should allow me to test 3/4 columns on the desktop, say 2 for tablet and definitely just 1 (stacked) on mobile devices.
12. Navigation - is checkout "hidden" in the burger menu - if so, this is not a good thing?
13. Wireframes - if your finished product differs from wireframe - explain (in a short way) why this is the case...