Overall Aim

The overall goal is to produce a working, fully featured, sensible website that sells a product or service online:

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

Deliverables

Main deliverable is a working web site which uses Bootstrap 5.3. The site must be designed from a mobile first perspective and should be fully responsive.

- Your site should work in all the major browsers – you will need to test on Chrome and Edge.
- The site should be designed to work well in all screen sizes, with appropriate content and layout for major breakpoints.

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)

Content

- Home, Contact, About Us, Shop (with product catalogue of at least 6 items), Login/Register
- User Registration - Forms on the site will in include Login, Register, Forgot Password – so a user can register on the site.
- 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

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 5.3 (e.g. carousel on home page of top selling products) or elsewhere – as long it is appropriate to the site.

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%		
	Content makes sense to the site?	2%		Do I know what the site is selling? Is it in Euros?
	If not logged in at checkout does it ask me to login?	2%		
	Additional styling?	1%		To make project less bootstrappy?
	Wireframe resembles finished product?	1%		
Technical			8%	Use of Bootstrap/Javascript/SASS, etc
	Javascript used?	1%		
	CSS/JS in separate files?	1%		
	Checkout increments?	1%		
	Checkout contains items actually added to checkout?	1%		
	SASS used?	1%		Week 9 topic
	Files minified?	1%		Week 11 topic
	Login changes to logout?	1%		
	Additional points?	1%		Points awarded for functionality above what was requested
Forms usage			8%	Required breadth of forms using validation, accessibility
	User can login with test username and password	1%		
	Invalid username and password throws error?	0.5%		
	Checkout form has user details populated?	1.5%		
	Accessible forms?	1%		Labels associated?
	Payment details validated?	1%		
	Good user interaction if payment successful?	1%		
	Checkout to zero if payment successful?	1%		
Project Report			6%	
	All sections addressed?	3%		Thought put into the responses?
	Wireframes	3%		Were wireframes submitted? Was Care and attention put into them?

Tools you will use

- Boostrap 5.3, CSS customisations/SASS
- **Javascript**
- 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().

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 <u>will</u> use Javascript arrays 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 <u>could</u> use Javascript arrays 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. However you will probably just have one overall shop page which displays all of your products on one page.

Plagiarism

- The work needs to be your own. You cannot use Generative AI tools or use code from YouTube, Github repositories, etc.
- The 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.

Things to consider for a good implementation (these are common issues that occur with projects every year)

1. Use Bootstrap – You must use Bootstrap to provide a responsive, reliable site.

2. At least 6/8 products - use relevant image, text description, price. Using GenAI for product images is fine.

3. If you are using an API for products - and they are in USD, then contact page will be a business based in the US.

4. Pay attention to simple detail - eg. if price in EURO then contact us page is in EU..

5. The report - make sure you mention the stuff you have done, that is not obvious - e.g. rewritten the JS from scratch, using API for products, optimising site (e.g. minifying files, optimising images for desktop and mobile) - put your best forward forward

6. If you are uploading source in a zip - INSTEAD of a hosted site on git pages - make sure the link works - to do this - unzip to a different location - does evertything load and link? (Absolute link vs relative)

7. Double check AGAIN the work on a different location from where you have developed it (again – make sure you are not using absolute links in say, image files)

8. From design POV, make CSS changes nearer end of project. Make sure you don't reinvent the wheel, let Bootstrap do its job…

9. If you don't do things suggested in the project template - then say this or leave heading empty….

10. Breakpoints - at least have 3/4 columns, 2 columns, 1 column!!!!

11. Navigation - is checkout "hidden" in the burger menu - if so, this is not a good thing?

12. Wireframes - if your finished product differs from wireframe - explain (in a short way) why this is the case…