Middlesex University London

Faculty of Science and Technology

CST3145

Web-Based Mobile App Development

Module Leader: Kai Xu

Term: Autumn 2020

Duration of the module: 24 weeks

Document version: 1

Online location of handbook

This handbook can also be accessed via My Learning at: Insert link

Other formats available

This handbook is available in a large print format. If you would like a large print copy or have other requirements for the handbook, please contact the Disability Support Service disability@mdx.ac.uk.

Disclaimer

The material in this handbook is as accurate as possible at the date of production. You will be notified of any minor changes. If there are any major changes to the module you will be consulted prior to the changes being confirmed. Please check the version number on the front page of this handbook to ensure that you are using the most accurate information.

Other documents

Your module handbook should be read and used alongside your programme handbook and the information available to all students on My Learning, including the Academic Regulations. Your programme handbook can be found on the My Learning programme page.

Contents

1 Welcome

This module teaches you how to build an app that run online and iOS or Android devices using the web technologies that you started in the second year.

2 The Module Team

Kai Xu		
	Role:	Module Leader
	Room:	TG07
	Email:	k.xu@mdx.ac.uk
	Telephone:	15510

Andrew Caruana		
NO PHOTO	Role:	Malta Lecturer
	Room:	
	Email:	a.caruana@mdx.ac.uk
	Telephone:	

3 Communication with the Teaching Team

Students may contact staff via e-mail, phone, and by making an appointment to see them. Due to the COVID19 situation, all communications will be done virtually unless there is an absolute need for face-to-face meeting.

Staff will contact students by e-mail, phone, the My Learning module page and via lectures and seminars. The team may send urgent group and/or individual messages about the module to you by email, so it is important that you read your University email regularly.

In the first instance problems should be dealt with by talking to a member of the module team. You can give feedback on this module to the module leader, your Student Voice Leader, to your personal tutor, and through the end of module evaluation survey.

Our most important consideration is your health, wellbeing and safety as well as our staff and people related to the University. Remember that you as part of #TeamMDX can stay up-to-date with the guidance on Coronavirus at `https://unihub.mdx.ac.uk/coronavirus-covid19`.

4 Module Overview

4.1 Aims

This module aims to develop a good understanding of the latest app programming languages, frameworks, and tools to develop modern software that can be deployed on platforms including desktop, web, and mobile devices (both Android and iOS).

The module will cover the latest programming language standards that are fundamental to app development. Modern programming frameworks will be introduced to simply the otherwise complex development workflow and introduce the ability to target multiple platforms such as Android and iOS.

4.2 Learning Outcomes

4.2.1 Knowledge

On completion of this module, the successful student will be able to:

1. Understand the latest programming standard required for advanced app development;
2. Understand the latest app development frameworks and their strength and weakness;
3. Understand the back end server and database technologies that provide data access and storage;
4. Understand essential app development tools such as version control and dependency management.
5. Understand the methodology of developing platform-independent mobile app and the strength and weakness of existing libraries.

4.2.2 Skills

This module will call for the successful student to demonstrate:

6. Ability to develop efficient and robust app following the latest programming standard.
7. Ability to design and implement modern app utilising the latest software frameworks.
8. Ability to create the server and database required for data access and storage;
9. Ability to deploy, config, and administrate tools essential for app development workflow such as version control and dependency management.
10. Ability to develop platform-independent mobile apps using Web-based technologies.

4.3 Syllabus

- Advanced JavaScript features in $ECMAScript6$;
- Advanced app development with $Vue.js$ framework;
- Version control and issue tracking with Git and $Github.com$;
- REST API services with $Node.js$ and $Express.js$;
- Project dependency management with NPM;
- Progressive web app development with $serviceworker$;

- Mobile app development with $NativeScript$ framework;
- Software testing with libraries such as $Jest$, $Postman$, and $Cypress$;

4.4 Learning and Teaching Strategy

The focus of teaching will be a mixture of lectures and lab-based practical work. Besides introducing core concepts, the lectures will include live coding to demonstrate the new concepts and introduce practical coding skills. Various tools, such as Git, NPM, and Postman will be introduced as they become necessary. The students will work in groups during lab to practice the skills covered in the lecture.

Contact hours:

- Lecture: 1.5 hrs
  - All the online lectures will be conducted in Zoom;
  - The lectures will be recorded and made available online;
  - The lectures will include live coding sessions, so it is recommended to use a laptop to code along.
- Laboratory: 1.5 hrs
  - All the online labs will be conducted in Zoom;
  - Students will form groups to work on lab tasks;
  - Some of the lab tasks will be marked and needs to be demonstrated in lab.

4.5 Assessment Scheme

The module will be assessed by **coursework only**, and there is no exam. **Students must get overall at least 40% to pass the module**. **It is not required to pass each coursework to pass the module.**

There are three coursework (details in Section 7):

- Coursework 1 (individual): Web App with Vue.js (35%);
- Coursework 2 (individual): Progressive Web App with REST API (35%);
- Coursework 3 (individual): Mobile App with NativeScript (30%).

Each coursework includes 2 marked in-lab tasks, at 5% each. These in-lab tasks will be completed in groups and demonstrated in-lab. Group member not present at the demonstration will receive zero mark, unless evidence for extenuating circumstance is provided.

All the coursework has to be demonstrated. Coursework submission that is not demonstrated by the deadline will receive zero mark. During the demonstrate, part of the work will receive zero mark if it cannot be explained satisfactorily during the demonstration.

4.6 Extension and Late Penalty

All extension must be applied through the *Extenuating circumstances* service (see Section 6.4 for more details). Please do not contact the module leader for extension.

The late penalty is 5% for each day after the deadline. It is the 5% of your final mark. For

example, if you receive 30 for your coursework and are two days late, you coursework mark will be $30 \times (1 - 0.5 \times 2) = 27$.

4.7 Ethics

The teaching, learning, assessment and research activities undertaken in this module have been considered and are **not likely to require ethical approval**.

However, **please seek advice if undertaking the module entails carrying out any research activities involving human participants, human data, animals/animal products, precious artefacts, materials or data systems**. If you submit work that includes data gathered from or about people, this may be treated as academic misconduct and could lead to fail grade being awarded.

Research ethics approval seeks to ensure all research is designed and undertaken according to certain principles of ethical research. These include:

1. Primary concern must be given to the safety, welfare and dignity of participants, researchers, colleagues, the environment and the wider community
2. Consideration of risks should be undertaken before research commences with the aim of minimising risks to those involved i.e. human participants or animal subjects, colleagues, the environment and the wider community, as well as actual or potential risks to those directly or indirectly affected by the research.
3. Informed consent should be freely given by participants, and by a trained person when collecting or analysing human tissue (details on accessing and completing online training for gaining informed consent for HTA purposes can be found below in Section 8).
4. Respect for the privacy, confidentiality and anonymity of participants
5. Consideration of the rights of people who may be vulnerable (by virtue of perceived or actual differences in their social status, ethnic origin, gender, mental capacities, or other such characteristics) who may be less competent or able to refuse to give consent to participate
6. Researchers have a responsibility to the general public and to their profession; as such they should balance the anticipated benefits of their research against potential harm, misuse or abuse which must be avoided
7. Researchers must demonstrate the highest standards of ethical conduct and research integrity. They must work within the limits of their skills, training and experience, and refrain from exploitation, dishonesty, plagiarism, infringement of intellectual property rights and the fabrication of research results. They should declare any actual or potential conflicts of interest, and where necessary take steps to resolve them.
8. When using human tissues for research,Human Tissue Act and Human Tissue Authority (HTA) requirementsmust be met.Pleasecontact the relevant designated person (DP) in your department or the HTA Designated Individual (DI) (Dr Lucy Ghali - L.Ghali@mdx.ac.uk). Further information is provided below inthesection: "Human Tissue Authority Information", see 'GovernanceStructure" document and SOPs etc.
9. Research should not involve any illegal activity, and researchers must comply with all relevant laws.
    - For more information about ethics go to the Middlesex Online Research Ethics (MORE) system which has information and guidance to help you meet the highest standards of ethical research using this link: `https://MOREform.mdx.ac.uk`
    - Information and further guidance on how to complete a research ethics application

form (e.g., video guides and templates) can be found on the MORE MyLearning site [1]: http://mdx.mrooms.net/enrol/index.php?id=12277 (Log in required)

5 Learning Resources

This module has a variety of learning resources available for you to use to support your learning. These include recorded lecture, lecture slides, feedback, and key reading materials. These can be accessed online via the module page. Please visit the module page regularly to make use of these.

5.1 Reading List

Your online reading list can be accessed from the My Study area of UniHub (http://readinglists.mdx.ac.uk/lists/78D0F586-A45D-60DB-32A7-5D5EC274302B). This highlights recommended reading for this module. The course website has many links to other online resources.

6 Expectations of studying this module

6.1 Engagement

Engaging with online and remote learning and activities is integral to your success. Middlesex University supports students, enabling them to achieve their full potential.

We provide this support through a number of strategies, all of which provide our students with a supportive learning environment online, remotely, face-to-face, or blended.

Further information on engaging with your programme will be available at your Induction and updates online at UniHub https://unihub.mdx.ac.uk/study/assessment/attendance

6.2 Professional Behaviour

The programme of study you are undertaking is underpinned by developing professional behaviour and attitude. You are expected to behave in a professional, supportive manner to your peers and teachers. You must come to sessions prepared and ready to contribute where appropriate. Please remember that your University ID should be carried with you always and you must be able to identify yourself if asked to do so. Please conduct your email communication with fellow students, tutors and all relevant staff in a formal and courteous manner

[1] Middlesex University Definition of Research document can be located on this site.

6.3 Academic Integrity and Misconduct

Academic misconduct is a breach of the values of academic integrity, and can occur when a student cheats in an assessment, or attempts to deliberately mislead an examiner that the work presented is their own when it is not. It includes, but is not limited to, plagiarism, self-plagiarism commissioning or buying work from a third party or copying the work of others, breach of examination room rules.

Students who attempt to gain unfair advantage over others through academic misconduct will be penalised by sanctions, according to the severity of the offence, which can include exclusion from the University. Links to the relevant University regulations and additional support resources can be found here:

- Becoming a successful student Course which includes Academic Integrity Access to course (You will have to log into to MyUniHub and then MyLearning to access the course): `https://mdx.mrooms.net/mod/lesson/view.php?id=877307`
- Section F: Academic Integrity and Misconduct `https://www.mdx.ac.uk/about-us/policies/university-regulations`
- Referencing & Plagiarism: Suspected of plagiarism? `http://libguides.mdx.ac.uk/c.php?g=322119&p=2155601`
- Referencing and avoiding plagiarism `http://unihub.mdx.ac.uk/your-study/learning-enhancer online-resources/referencing-and-avoiding-plagiarism`
- The MDXSU Advice Service offers free and independent support face-to-face in making an appeal, complaint or responding to any allegations of academic or non-academic misconduct `https://www.mdxsu.com/advice`

6.4 Extenuating circumstances:

There may be difficult circumstances in your life that affect your ability to meet an assessment deadline or affect your performance in an assessment. These are known as extenuating circumstances or ECs. Extenuating circumstances are exceptional, seriously adverse and outside of your control. Please see link for further information and guidelines `https://unihub.mdx.ac.uk/your-study/assessment-and-regulations/extenuating-circumstances`

7 Assessment

Formative assessment: Formative assessment is completed during your year of study and provides the opportunity to evaluate your progress with your learning. Classroom assessment is one of the most common formative assessment techniques although other activities and tasks may be used. Formative assessments help show you and us that you are learning and understanding the material covered in this course and allow us to monitor your progress towards achieving the learning outcomes for module. Although formative assessments do not directly contribute to the overall module mark they do provide an important opportunity to receive feedback on your learning.

Summative assessment: Summative assessment is used to check the level of learning at the end of the course. It is summative because it is based on accumulated learning during the course. The point is to ensure that students have met the learning outcomes for the

course and are at the appropriate level. It is the summative assessment that determines the grade that you are awarded for the module.

There are 3 assessment components in this module:

- Coursework 1 (week 8): Web App with Vue.js (35%);
- Coursework 2 (week 15): Progressive Web App with REST api (35%);
- Coursework 3 (week 22): Mobile App with NativeScript (30%).

7.1 Coursework 1: Web App with Vue.js (35%)

Deadline: week 8

7.1.1 Task

For this coursework, you need to create the front-end of a fictitious web app that allows students and their parents to look for after school classes and activities.

The second coursework will add the backend and make the app work offline, and the third coursework will turn it into a mobile app.

7.1.2 Requirements

- The coursework does not require any backend storage such as a (MongoDB) database. All the data must be stored locally using local storage.
- You can use external CSS library such as Bootstrap. Make sure the library file or online link is included in the submission.
- Any JavaScript library is not allowed if it duplicates or replaces features provided Vue.js. Check with the tutor if not sure.

A submission will receive zero mark if it fails any of the requirements below:

- It must be implemented using $vue.js$ framework;
- The code must be stored in a $GitHub$ repository with at least 10 commits;
- The app must be available online using $GitHubPages$;
- The web app must be demonstrated.

Coursework code will not receive any mark, even it works fine if:

- It cannot be explained satisfactorily during the in-lab demonstration, i.e., student cannot explain what the code does.
- It is not implemented with Vue.js when it is possible to do so. This includes using plain JavaScript or other libraries.

7.1.3 Marking criteria

Group in-lab tasks (10%):

- Display the information of one lesson (3%):

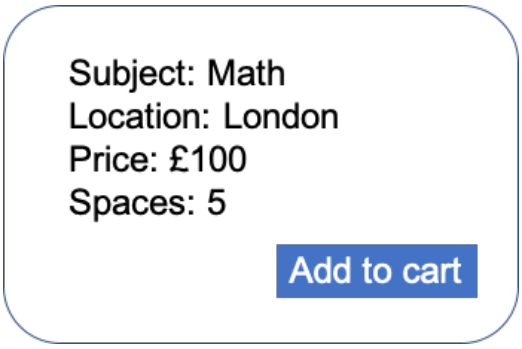


Figure 1: Coursework 1 - lab work

  - The information must include at least: subject, location, price, and number of spaces;
  - All the lesson information must be stored in a `Vue` object;
- The lesson must have an 'Add to Cart' button (4%):
  - The 'Number of Space' is reduced by one after clicking the 'Add to Cart' button if it was greater than zero;
  - No change happens after clicking the 'Add to Cart' button if the 'Number of Space' equals zero.
  - The interaction must be implemented using `v-on`.
- The code is stored GitHub repository and can be accessed online using GitHub Pages (3%):
  - The code must be stored in a GitHub repository with at least 5 commits;
  - The Vue.js app must be accessible through GitHub Pages.

Individual task (25%):

- Display lessons (5%)
  - There should be at least 10 lessons and each lesson has 5 spaces;
  - Each lesson should have at least:
    * Subject;
    * Location;
    * Price;
    * Space (how many spaces are left);
    * A image or icon.
  - The list of lessons must be stored as an array of `JSON` objects, one object for each lesson, in a separate JavaScript file, such as `lessons.js`;
  - `v-for` has to be used for the display of the lesson list.
- Sort (5%)
  - User can choose to sort the lessons by one of the following attributes: subject, location, price, or availability;
  - There must be an option to sort in ascending or descending order, regardless of the attribute selected;
- Add to cart (5%)
  - Each lesson must have a 'Add to Cart' button;
  - The button is only visible when there is still space available;
  - Clicking the button once will add one space to the shopping cart, reducing the remaining space by one;
- Shopping cart (5%)
  - The shopping cart button should only be visible after at least one lesson is added

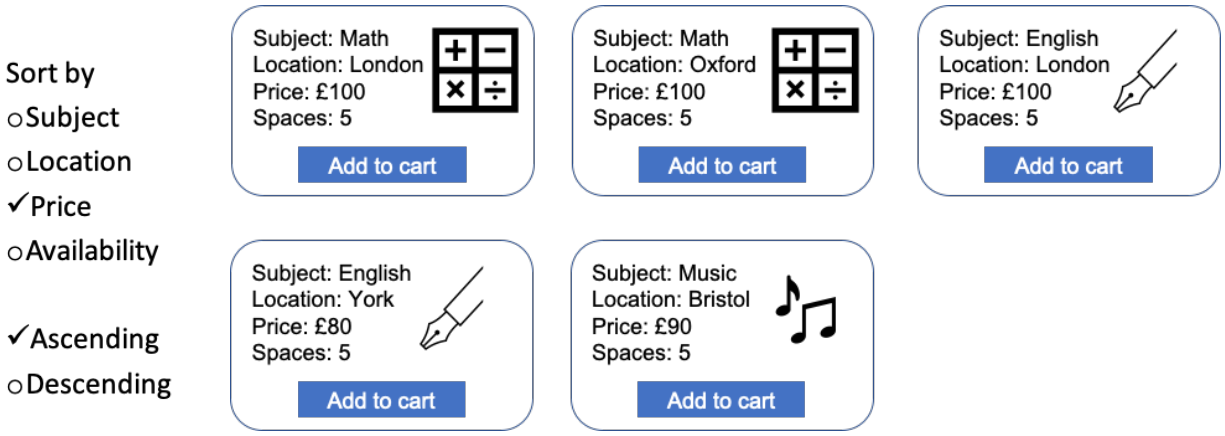


Figure 2: Coursework1 - Lesson List and Sort

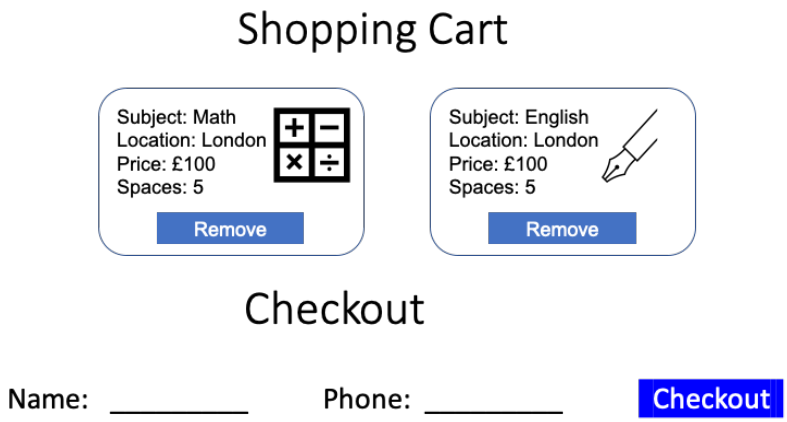


Figure 3: Coursework1 - Shopping Cart and Checkout

to cart;
  - Clicking the shopping cart button should go to a new page, with a button to go back to the lesson page;
  - The shopping cart should show all the lessons added;
  - User should be able to remove lesson from shopping cart.
- Checkout (5%)
  - The checkout is part of the shopping cart page;
  - A user must provide the following information before he/she can check out:
    - Name;
    - Phone;
  - The 'Name' must be letters only and the 'Phone' must be numbers only;
  - The 'checkout' button is only visible after both valid 'name' and 'phone' are provided;
  - Clicking the 'checkout' button should display a message confirming the order has been submitted.

7.2 Coursework 2: REST API and Progressive Web Apps (PWA) (35%);

Deadline: week 15

7.2.1 Task

For this coursework, you need to

- Create the back-end (server and database) for the app built in Coursework 1;
- Make the app 'progressive', i.e., it can work without internet connection.

7.2.2 Requirements

A submission will receive zero mark if it fails any of the following requirements:

- The backend server must use 'Node.js'; Apache or Xampp is not allowed;
- The data must be stored in 'MongoDB'; any other database is not allowed;
- All database access, such storing and retrieving data, must be achieved through 'REST API'; any other type of access is not allowed, including direct database access;
- The REST API must be developed with 'Express.js';
- The front-end data access must be achieved with 'promise' using 'fetch' function; 'xmlHttpRequest' is not allowed.
- The code must be hosted in a GitHub repository with at least 10 commits. You can continue using the same repository created for cw1, in which case 10 new commits are required.
- The web app must be demonstrated. Coursework code will not receive any mark, even it works fine if It cannot be explained satisfactorily during the in-lab demonstration, i.e., student cannot explain what the code does.

7.2.3 Marking Criteria

Group in-lab tasks (10%): (details to come).

Individual task (25%):

- Data storage and access (8%):
  - Store in MongoDB order information (2%) - minimal fields: name, phone number, lesson ID, and number of space.
  - Store in MongoDB course information - minimal fields: topic, price, location, and space (2%);
  - Connect to MongoDB with native Node.js driver for MongoDB (2%);
  - Access database in the front end using 'promise' with 'fetch()' function (2%).
- REST API (9%):
  - Retrieve all lessons (3%);
  - Save a new order when it is submitted (3%);
  - Retrieve all the orders by the same user with name and phone number (3%);
- Progressive Web Apps (8%):
  - Make the app installable (2%):
    - Can be added to home screen;
    - Run in full-screen mode;
  - Use service worker to cache app files (2%):
    - Each course must have an image, with at least 10 different images;
    - Always load app from cache first.

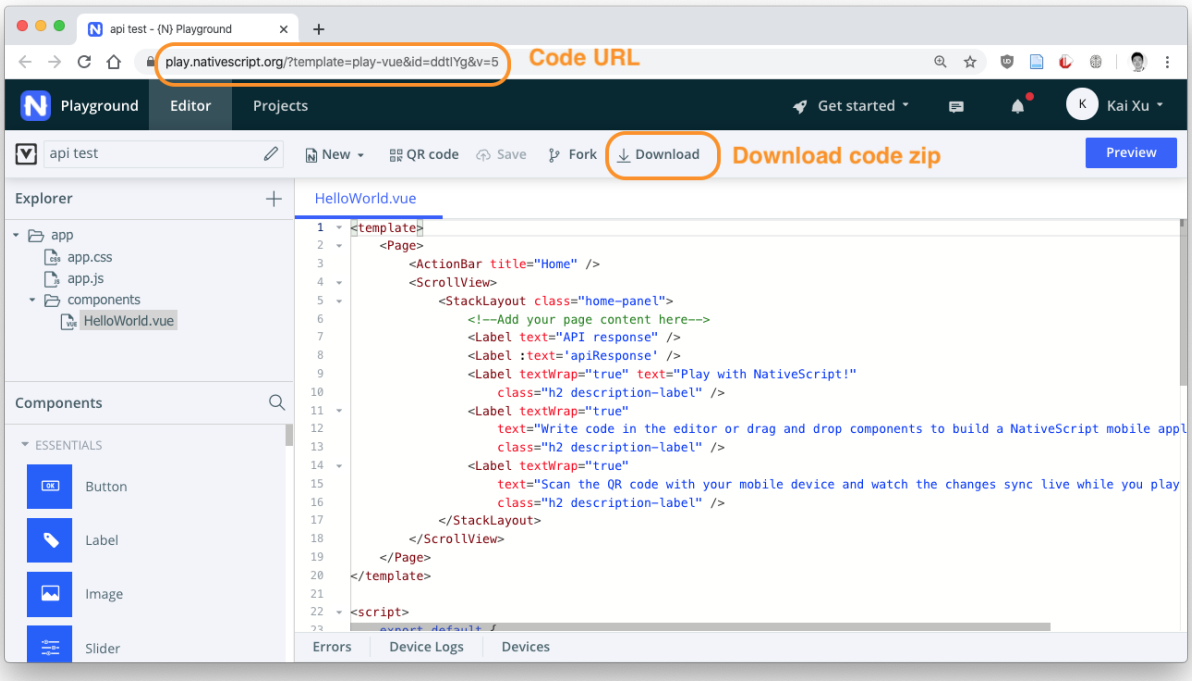


Figure 4: NVS Playground

– Send user notification whenever data is retrieved from cache (2%);
– Progressive loading: Only load an image when it appears in browser window (2%)

7.3 Coursework 3: Mobile App with NativeScript (30%);

Deadline: week 22

7.3.1 Task

For this coursework, you need to create an native mobile app using

- NativeScript-Vue Playground (`https://play.nativescript.org/`) and
- the front end and back end created for Coursework 1 and Coursework 2 respectively.

7.3.2 Submission

- All the Playground code files in a zip, which can be downloaded from NSV Playground (see Figure 4);
- A text file containing the URL to the code in Playground (see screen shot below);
- The back end node.js and express.js code files;
- A mongodb dump;
- Add everything to one zip file (max 50MB);

No GitHub repository is required, as the code is in NSV Playground.

7.3.3 Requirements

A submission will receive zero mark if it fails any of the following requirements:

- The mobile app must be built with NativeScript-Vue;
- The code must be hosted at NativeScript Playground `https://play.nativescript.org/`;
- The work must be demonstrated.

7.3.4 Marking Criteria

Group in-lab tasks (10%): (details to come).

Individual task (20%):

- Display lessons (5%)
  - There should be at least 10 lessons and each lesson has 5 spaces;
  - Each lesson should have at least:
    - Subject;
    - Location;
    - Price;
    - Space (how many spaces are left);
    - A image or icon.
  - The list of lessons should be stored in a separate `JSON` file;
- Sort (5%)
  - Sort must be in a 'side drawer';
  - User can choose to sort the lessons by one of the following attributes: subject, location, price, or availability;
  - There must be an option to sort in ascending or descending order, regardless of the attribute selected;
- Add to cart (5%)
  - Each lesson must have a 'Add to Cart' button;
  - The button is only enabled when there is still space available;
  - Clicking the button once will add one space to the shopping cart, reducing the remaining space by one;
- Shopping cart (5%)
  - The shopping cart button is enabled even the shopping cart is empty;
  - Clicking the shopping cart button should go to a new tab, with a button to go back to the lesson tab;
  - The shopping cart should show all the lessons added;
  - User should be able to remove lesson from shopping cart.

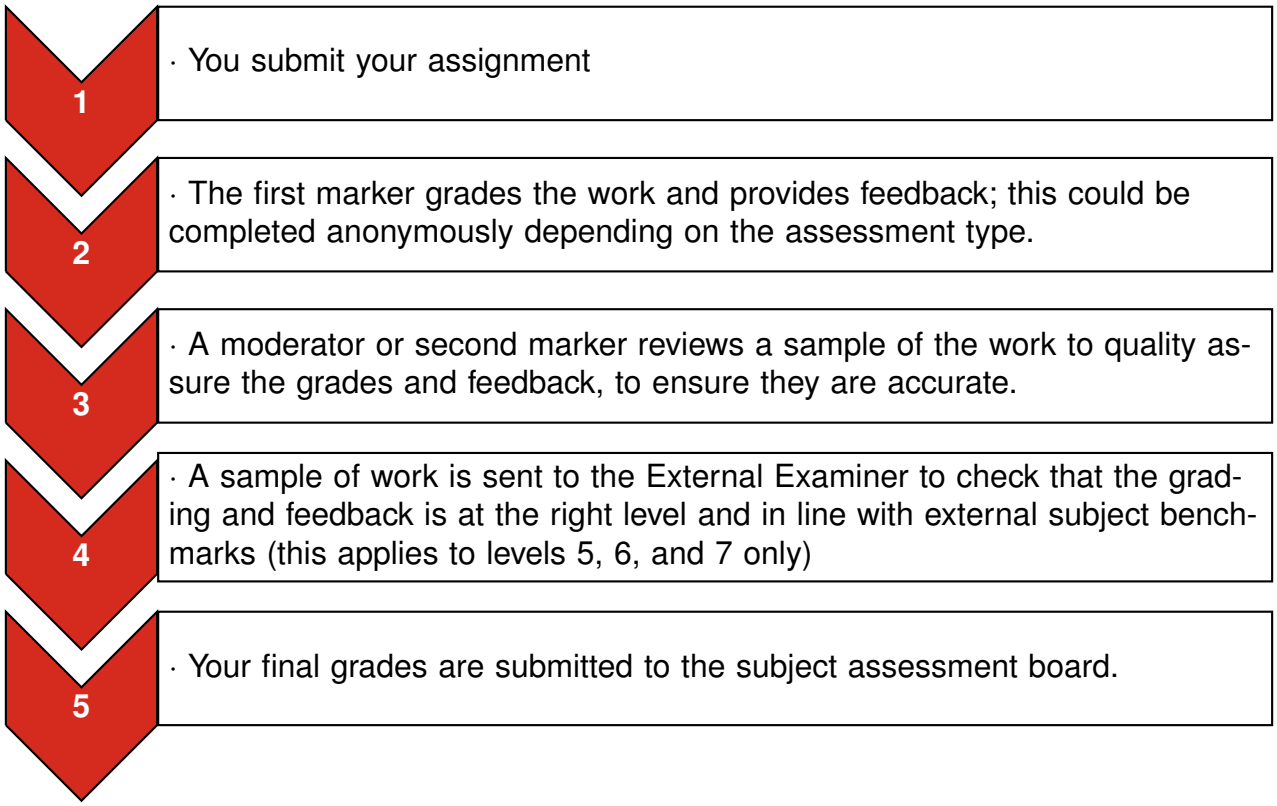
7.4 Feedback on your assignments

Oral feedback will be provided during demonstration, and written feedback will be provided in unihub in electronic form.

Feedback will normally be provided within 15 working days after the lab demonstration.

7.5 How is your assignment mark agreed?

Below is an overview of the marking process for your module assessment. Further information on the role of external examiners can be found at http://unihub.mdx.ac.uk/your-study/ensuring-quality/external-examiners.



7.6 Anonymous Marking Assessment Policy

We have worked with the MDXSU to create an anonymous marking policy, in response to student feedback. Anonymous marking ensures that your identity (your name, student number and other personal/identifiable information) is not made available to academics when they are marking your work. This means that you can have confidence that your assessments will be marked fairly and consistently. However, there are some forms of assessment for which anonymity cannot be guaranteed and these are recognised in the policy. We believe that it is important to provide you with the support and guidance needed to help you develop and prepare for your final assessments (those which count towards your final grades i.e. summative assessments). Therefore, anonymous marking will not apply to learning activities and assessments that do not contribute to your final grades (i.e. formative assessments). If you require further information and support to understand how anonymous marking works in your programme modules please contact the Module Leader for more information.

The Anonymous Marking Assessment Policy is available at: https://www.mdx.ac.uk/__data/assets/pdf_file/0037/563599/anonymous-marking-assessment-policy.pdf

8 Learning Planner

Week	Lecture and labs	Assessment and feedback
1	Module overview	
2	JavaScript vs. Vue.js	
3	Vue instance and Chrome tools	
4	Add to cart and GitHub/Pages	
5	Inventory and checkout page	
6	Forms	CW1 lab work presentation
7	Sorting	
8		Coursework 1 due
9	Node.js and Fetch	Coursework 1 lab demonstration
10	Express.js and Async JavaScript	Coursework 1 lab demonstration
11	REST API	
Christmas break		
12	CW1 sample solution	
13	Progressive web app 1	CW2 lab work presentation
14	Progressive web app 2	
15		Coursework 2 due
16	Single file component	Coursework 2 lab demonstration
17	NativeScript-Vue playground - Todo app 1	Coursework 2 lab demonstration
18	NativeScript-Vue playground - Todo app 2	
19	NativeScript-Vue - REST API 1	
20	NativeScript-Vue - REST API 2	CW3 lab work presentation
21	NativeScript-Vue - side drawer	
22		Coursework 3 due
23		Coursework 3 lab demonstration
24		Coursework 3 lab demonstration