

UWS 2025-26

Derek Turner

Module: COMP 10020

# INTERNET TECHNOLOGIES

## Internet Technologies 2025-26

Welcome .....	2
Summary of Module .....	2
Course rationale and aims .....	2
Learning Outcomes .....	3
Presentation of Module .....	3
Back up .....	3
Assessment Overview .....	4
Submissions and deadlines .....	4
Resources .....	5
Referencing and Plagiarism .....	5
Module Engagement .....	5
Lecture and Presentation Content .....	6
Activity Calendar .....	7
Assessment Brief .....	8
Coursework 1 (50%) .....	8
Marks Contribution .....	8
Objectives .....	8
Marking Scheme .....	9
Coursework 2 (50%) .....	11
Marks Contribution .....	11
Objectives .....	11
Working in Pairs .....	11
The Scenario .....	11
Originality .....	12
Joint Signatures .....	12
Demonstration .....	12
Documentation .....	13
Marking Scheme .....	16
Module descriptor .....	20

## Welcome

Welcome to the Internet Technologies module. I hope that working as a team we will be able to investigate technologies relevant to web apps and consider some of the pros and cons of various approaches. I am keen that you should regard the lecture and labs as a framework around which you are encouraged to explore other technologies

## Summary of Module

### Course rationale and aims

The background of the course is the huge growth in the range and power of Internet Technologies. There are many things that you should know about before you graduate: your aim should be to impress employers and clients and choose the best solution for them.

The aims of the course are to enable you to

- understand and work with a range of advanced internet technologies.
- implement distributed applications and choose between a range of implementation strategies.
- design distributed applications and analyse trade-offs between different distributed implementation strategies.
- think about the design of an app and consider a range of factors before moving to implementation.

The emphasis in focussing an exploration the technologies is on working through a standard framework learning some of the code from first principles then looking more generally at alternative approaches adapting code from tutorials and templates rather than programming from first principles. I expect you to have a working knowledge of the Internet and how to develop web pages, and a familiarity with scripting, databases, and some programming. I will expect you to use that prior knowledge to understand the technologies presented in the course to gain the abilities mentioned above.

This delivery will mainly focus on “React Router or NextJs” full-stack development based on typescript. This stack includes React as a client-side technology programmed in Typescript, a server based on Express and a MongoDB database. Working in a containerised Linux environment running node “Remix” web applications are typically developed using the VITE testing server. This provides a route to creating a final JavaScript bundle which can be served by a standard web server without the need of a node environment. NextJS is a rival framework which has its own bundler and provides server side programming. The pace of change in current applications is rapid and so latest developments will influence the choice of preferred stack. I hope you will develop confidence in your ability to adapt your coding skills to different situations and languages.

I will present enough material to enable you to pass the module. To get the highest classifications, you will need to be comfortable with the technologies introduced, going beyond this material, using the links and resources I provide as a starting point.

## Learning Outcomes

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

- L1. Understand and work with a range of current internet technologies to specify, create and deploy web apps using appropriate services.
- L2. Demonstrate independent learning in the area of advanced internet technologies.
- L3. Present and report upon group investigations of internet technologies.

## Presentation of Module

This session the presentation will be face-to-face. Labs will be supported on site and material to look at before sessions will be presented on Aula.

Most weeks I will provide a tutorial to have a conversation around the module notes and lab exercises. The lab exercises will need a node environment, I will be using Docker, running on Windows Subsystem for Linux as the working environment for lab exercises, use of Docker is recommended, but is not essential to the completion of coursework. Docker will also run MacOS or Linux, you should still be able to complete these labs on those machines though there may be some minor, but important, differences requiring adaptations to the lab instructions.

Notes are not intended to be self-contained: there are many Internet links in the laboratory sheets, and you are expected to find and consider related Internet resources. Peer learning is encouraged. The notes contain a viewpoint and interpretation of currently available resources, which you may well disagree with, it is important to form your own view and be able to justify it.

Each week there will be a structured lab where you will use resources freely available on the web to explore these technologies. To succeed in the course, you really must work through all the laboratories, using some time outside the lab sessions if necessary. These are step-by-step introductions to some core technologies. If you have practical difficulties with any of them, please contact me and we will endeavour to resolve them.

## Back up

I strongly recommend the use of GitHub for code back-up; you should also be able to save to your student one drive accounts. Take care over file management, loss or corruption of files is a cause of distress and always seems to happen when you need the files most, as when preparing an

assessment for submission. It is not a secure backup to have a spare copy of a programme on the same machine as the original, the back-up must be able to survive the demise of the original. Backing up takes time, it is a habit which is easy to allow to lapse, leaving you with out of date back up files and lots of work to do to restore files to the current status. File management and back up should be a core skill for any computer based professional whether these are programmes, audio files or video edits.

In using GitHub there is an extra advantage that some web cloud technologies can load code directly from a GitHub account which eases deployment. GitHub is designed to assist collaboration and since we are all dispersed between class sessions this is valuable.

## Assessment Overview

There are two elements of assessment, both of which should be worked towards gradually over the course of the module.

Coursework1: 50%. This is a group presentation of a topic relating to an Internet Technology. Ideally presentation groups will have four members. This will be presented in class and recorded via a Microsoft Teams meeting. You will be asked to produce a static webpage with guidance notes on the selected topic and a tutorial section. This can be linked to the Aula site to support peer learning. Some of your time in the labs should be spent following through the tutorials generated by other student groups. In this way the module can provide a broader view of technology than could be provided by lectures alone.

Coursework 2: 50%. This is best completed as pairs of students, but individual working is permitted. A document describing a site specification will be generated and this will be supported by a proof-of-concept application. The design document is the main item and should not be overlooked. The code is a proof-of-concept for which I will expect to see customised design aspects in the site and evidence of interaction with a database. I do not expect to see all aspects of the design document implemented in this site. Where possible you should deploy your site to the web.

Code for the app will be stored on GitHub. Students can choose to create this app in any appropriate technology and are not restricted to using those covered in labs, however Microsoft solutions and PHP/Laravel should *not* be used as these topics are covered in other modules. Often technologies described in coursework 1 may be taken up in some aspects of coursework 2.

Please note a WordPress site would not meet this brief. To encourage new learning, you are discouraged from using software frameworks which you are learning in other modules.

## Submissions and deadlines

Turnitin assignments on Aula will be open for submissions for two weeks, 1 week prior to the stated submission date and 1 week after the deadline. Uploading a submission after the stated date and

time (deadline) will incur a 10% penalty deducted from the final grade awarded for that individual assignment.

## Resources

You will be expected to work with freely available code and information from the web.

A key technology which will be featured is Docker <https://www.docker.com/>, this provides containerisation for other applications. Students may use docker in their coursework 2 submissions, but this is *not* a requirement.

## Referencing and Plagiarism

All academic written reporting should correctly reference any information sources used and fully credit any images or other resources incorporated in the report. Details of the Harvard Referencing scheme to be used can be found at the [UWS referencing guide](#).

Within this module you will be providing a slide presentation and a code tutorial with content created by a small group as coursework 1. I do not expect all the code in this to be original, but you must reference any sites or texts where you have drawn inspiration for ideas or sourced code constructs (including functions and classes). In coursework 2 you will be creating a site specification as a pair which should include references to the technologies used.

The work you present at university must be your own. Turnitin will be used as a marking and feedback tool. This will show similarities between submitted work and other sources.

You *may* use AI to help you to research material for your presentation and in some cases AI may be part of the subject of your presentation. However, you should not generate your written submission using AI.

You *may* use AI programming assistants such as github copilot to maintain correct typescript syntax and assist with code debugging, but you *must* state how you have used this software within a declaration which is submitted with your assignment.

## Module Engagement

Students will take cognisance of the University's requirements for Academic Engagement and Attendance. Students who are not engaging with the module may be withdrawn.

In order to engage with this module a student will be expected to:

- View and assimilate online materials according to the module schedule

- Participate in class meetings and conversations
- Contribute meaningfully to group presentations
- Complete the assessment according to the detailed brief

Engage in opportunities to broaden their interest in internet technologies beyond the core of the module materials

## Lecture and Presentation Content

The following topics are intended as an outline plan. This may vary to accommodate the latest trends in technology.

### Section 1: Client side of the stack (weeks 1-5)

- 1.1: Introduction, setup and first React code
- 1.2: Introduction to Docker and development of CW2 scenario
- 1.3: UI design. React consumes JSON from a standard source
- 1.4: SQL Rest and GraphQL. Using postman to create a mock API
- 1.5: Mobile design. An introduction to React Native

### Section 2: Student Presentations (weeks 6-7)

- 2.1: Group presentations: Part 1
- 2.2: Group Presentations: Part 2

### Section 3: Server side of the stack (week 8 – 12)

- 3.1: API building. Remix application: part 1
- 3.2: Hosting. Remix application: part 2
- 3.3: Scaling. Remix application: part 1
- 3.4: GDPR. Remix application: part 2

## Activity Calendar

2025-26

Term 1

Week	Month	Mon	Tue	Wed	Thur	Fri	Sat	Sun	Comment
0	Sept	8	9	10	11	12	14	14	Induction
1	Sept	15	16	17	18	19	20	21	Intro to React
2	Sept	22	23	24	25	26	27	28	Intro to Docker
3	Oct	29	30	1	2	3	4	5	UI design
4	Oct	6	7	8	9	10	11	12	API
5	Oct	13	14	15	16	17	18	19	Intro to React Native
6	Oct	20	21	22	23	24	25	26	Student Presentations
7	Nov	27	28	29	30	31	1	2	Student Presentations
8	Nov	3	4	5	6	7	8	9	Full Stack Application
9	Nov	10	11	12	13	14	15	16	Full Stack Application
10	Nov	17	18	19	20	21	22	23	Full Stack Application
11	Nov	24	25	26	27	28	29	30	Full Stack Application
12	Dec	1	2	3	4	5	6	7	
13	Dec	8	9	10	11	12	13	14	
14	Dec	15	16	17	18	19	20	21	

T1 induction/teaching  
T1 ends

Monday 8th September 2025 Teaching from Monday 15<sup>th</sup> September 2024  
Friday 19th December 2025

T2 Teaching Commences

Monday 19th January 2026

Spring Break

Monday 6th April – Friday 10th April 2026

T2 Continues

Monday 13th April 2025

University Holiday

Friday 3rd April (Good Friday) Monday 6th April 2026 (Easter Monday)

University Holiday

Monday May 4<sup>th</sup> 2026 (Early Spring Bank Holiday)

T2 Ends

Friday 8th May 2026

<https://www.uws.ac.uk/media/7760/term-dates-2025-2026.pdf>



## Assessment Brief

### Coursework 1 (50%)

Scheduled into class time weeks 6 & 7.

This work will be presented live in class and recorded via Microsoft Teams. The written element will be submitted via Aula. Feedback will be provided verbally and through Turnitin.

### Marks Contribution

This coursework will contribute 50% towards your final mark for the module. The work will be marked from 100% and this will be weighted to determine the final mark.

### Objectives

Working in small groups students will research an aspect of internet technologies selected from a list presented in the lecture notes, which is not covered in the main presentation of a module in the current year of their programme. In normal circumstances all students in the group will receive the same mark. In the event of a group break down work will be marked individually.

Students will create an organised PowerPoint presentation in class supported with Microsoft Teams which will be recorded.

After the presentation, a slide set with supporting notes and tutorial lab instructions will be mounted on GitHub and the link to this submitted via the Aula VLE. Lab time will be allocated to allow students to through tutorials from this collection to gain a benefit of peer learning.

Students will be expected to make a presentation which provides:

- a brief introduction/overview for an audience encountering a topic for the first time.
- simple examples which demonstrate their interaction with the technology.
- show how the technology elements work in a system with other components.
- describe a scenario where the technology might be used.
- describe pros and cons of use.
- reference notes and a tutorial guide for fellow students

Code examples may be based on reading, and sources must be fully credited. There should be a small amount of original input to the code which could be simple examples of contextualization.

## Marking Scheme

Marks will be allocated to the presentation, wiki reference and tutorial. Please ensure that the written section makes a clear distinction between notes and tutorial. You may, if you wish to, use videos in the tutorial.

Grade	Presentation (50%)
A1	Exceptional: As A3 conveying a depth of subject knowledge appropriate to a seminar.
A2	Outstanding: As A3 and outstanding in one area of verbal or visual presentation or response to questions.
A3	Excellent: A well-structured presentation with progressive examples, strong visual elements and good coordination between presenters. Good response to questions using prepared follow-on materials.
B1	Very Good: A well-structured presentation with appropriate examples and strong visual elements. Good response to questions.
B2	Good: A well-structured presentation with appropriate examples
C	Basic: Essential information on web technology is conveyed
D	Below threshold: Conveys uncertainty in knowledge of the topic.
E	Well below threshold: Patchy and inaccurate account of topic.

Grade	Online information (20%)
A1	Exceptional: A go-to reference on the topic.
A2	Outstanding: Well formatted and written detailed authoritative referenced sources incorporating useful visuals or tables and a cheat sheet
A3	Excellent: Well formatted and written detailed authoritative referenced sources incorporating useful visuals or tables.
B1	Very Good: Well formatted and written detailed authoritative referenced sources
B2	Good: Detailed information drawn from several authoritative referenced sources
C	Basic: Basic information drawn from 3 appropriate referenced sources

D	Below threshold: Insufficient sources, lack of personal insight or poor written style
E	Well below threshold: Not a practical learning resource

Grade	Online Tutorial Lab Instructions (30%)
A1	Exceptional: An excellent learning resource
A2	Outstanding: tutorial with a high degree of originality establishes how to set up platform and gives a step-by-step introduction to basic topic which is easy to follow. Code for a meaningful application is developed.
A3	Excellent: tutorial establishes how to set up platform and gives a step-by-step introduction to basic topic which is easy to follow. Code for a meaningful application is developed.
B1	Very Good: tutorial establishes how to set up platform and gives a step-by-step introduction to basic topic which is easy to follow.
B2	Good: tutorial gives a step-by-step introduction to basic topic which is easy to follow.
C	Basic: tutorial gives basic information but is not structured in a way which is easy to follow.
D	Below threshold: tutorial steps are inadequate to achieve a successful completion
E	Well below threshold: inaccurate or incomplete tutorial

Any marks visible on Moodle before the School Assessment Board have met are provisional and subject to ratification by moderator and external examiner.

## Coursework 2 (50%)

Group Submission

Sunday 7th December 2025

Please submit one copy electronically via Aula clearly identifying the student group. If there is a problem, contact me.

### Marks Contribution

This coursework will contribute 50% towards your final mark for the module. The work will be marked from 100% and this will be weighted to determine the final mark.

### Objectives

Your task is to create a **specification document** for a site and to implement a **proof-of-concept site**.

The site design is to provide solution for a realistic scenario that requires choices between internet technologies and implementation strategies. The documentation must be **detailed, structured** and **cover all specified aspects** discussed in class. The proof of concept will be a working web app which illustrates the design of the site and some interaction with a database, it is not a full implementation of the design document.

You may use any appropriate technology to implement the site and you are not restricted to using the Remix stack or Docker containerisation. Do not use Microsoft solutions or PHP/Laravel as these are covered in other modules.

**Note:** Your solution must be original with code stored on GitHub and preferably deployed on the internet.

### Working in Pairs

Although the coursework is intended to be developed in pairs, individual submissions will be accepted, but no allowance can be given for solitary efforts. Exceptionally the site may be developed by three persons if even pairings are not possible.

### The Scenario

[This scenario is fictitious. Do not contact any real organisations involved.]

Your employer is a software house that has been invited to submit a tender for the provision of a web app to support a group planning to develop "Paisley Highland Games".

Your company is keen that a solution for a job site which will have a generic pattern which can be easily applied to other potential customers.

You may design a site to support this fictional client and use this as a model for your design documentation and your proof-of-concept site.

The games site documentation will require as a minimum: A description of the scope and context of the fictional client, descriptions the proposed competition and the facility for groups or individuals to apply to participate, an administrator administrator role should be provided to maintain the site.

You should review a range of Highland Games and other sporting sites at different scales to suggest ideas for your own site.

Your boss is keen to get a site specification as an internal document with a proof-of-concept site which could be shown to potential clients.

## **Originality**

You should attempt to create a fresh approach that is original and impresses senior colleagues in this small design company). You may of course exchange ideas with other groups and seek assistance from teaching staff, but directly plagiarised documents or designs will attract a penalty.

## **Joint Signatures**

In the (unlikely) event of members of a group disagreeing as to the relative weight of their contributions, each group member is required to sign to what they believe is theirs and their partner's percentage contribution to the work. The module coordinator will have regard to this in allocating a marks weighting and this will be highlighted to the moderator and external examiner.

Normally members will be adjudged to have contributed equally, although not necessarily in the same area, signatures are not required in this case. By contrast, if the group agrees that one has contributed less than the other, say 25% of their possible contribution, then that member signs to a 25% effort, while the other member signs to a 75% contribution. Marks allocated to individuals will be weighted accordingly.

## **Demonstration**

There will be the opportunity to present work in progress and to receive verbal feedback during lab session.

## Documentation

The site specification for this coursework should be based on the following structure:

### Table of Contents

Section headings (auto generated)

Table of figures (auto generated)

### Introduction

Restate the scenario with details of your selected competition theme

### Overview

Give an overview of the app function you envisage bearing in mind the following points.

#### Background

- Review the data sources your app will use
  - Public, Private, Licenced
  - APIs
  - User input
- Review features of competitive web apps picking out common core features and useful advanced features

#### Core Functions

- Focusing onto the services which are essential to the apps function consider:
  - What makes the proposed app relevant/distinctive?
  - What business processes needed
  - Consider integration with existing services
  - What profile information is required? Where will it be stored?
  - Development time and cost
  - Will the proposed app meet the brief?

#### Advanced Functions

State what features you regard as the basic core of the app which would be essential in an initial launch version of the app. Identify further advanced features which could be developed over a longer timescale after initial launch.

- Planned features to be added after the app has launched
- What group are these for and what value do they add
- How does the app need to change to accommodate these?
- International aspects

#### Data Protection

What features in the app design and implementation will allow it to comply with GDPR?

- GDPR features

- Nature of personal data
- Design for consent
- Users can see their own personal data
- Processing and Deletion of Data
- Data security
- Authentication

## Implementation

### User Interface

Describe the site layout and show some wireframes of typical pages

- Sketch what the apps UI would look like for data entry
- Sketch the verification response to data entry
- Sketch what the apps UI would look like for content display
- Consider the view for all user groups
- Discuss design tools used to mock-up the interface

### Technology Stack

Describe the technology stack you propose to use for the app. This may be a MERN stack, but you are free to use other technologies, particularly those which you researched in CW1

- What technologies are appropriate to use in the stack and justify a choice
  - Client software
  - Servers
  - Database
  - Storage
  - Plug in
  - Other

### Data organisation

Provide an outline of what database tables/collections will be needed.

- What is the schema within these tables?
- What relations exist between tables and fields?
- Diagrams

How will an API be arranged to access the data? Give some examples.

- What UI forms will generate queries?
- Are queries using SQL, REST or GraphQL?
- Has the design process been supported by mock servers?

## Hosting

### Services

What services would you propose using to host the app. Describe some of the features of the service and state how costs are calculated.

- Single server (Apache, Nginx, IIS), multiple server, virtual server, platform as a service, load balanced cluster
- Distribution of data
- Consider the benefits of cloud services
  - Compare available services
  - Features and costs
  - Ease of use

### Scalability

What factors (in general terms) may affect scalability if usage grows over time? Can your hosting solution handle this?

- What aspects of the app might cause scalability issues over time?
- Number of users
- Amount of data
- Strategies: more servers, microservices
- Cost implications

### Tracking and statistics

*Briefly* state what features your host may provide for tracking site usage.

- Usage and flow through the site
- What is valid to be tracked and what should not be tracked

## Conclusions

All reports must draw conclusions. Yours should say why your approach to the brief is the most appropriate.

## References

All references must use Harvard format and be cited in the text following the UWS referencing guide.

A proof-of-concept site should be made. This does not need to have all the features mentioned in your report, but it should be available online. It should just convince the client of your ability to deliver what you are offering.



## Marking Scheme

The following are guidelines only, and a given grade may be awarded for other work that is deemed suitable.

Generally, each criterion depends on the preceding criteria being met:

Grade	Report: Overview (20%)
A1	Exceptional: detailed background research is applied to the discussion core and advanced functions and GDPR has been considered in the context of the app design. Excellent discussion in all sections describing an app with commercial potential.
A2	Outstanding: detailed background research is applied to the discussion core and advanced functions and GDPR has been considered in the context of the app design. Excellent discussion in all sections.
A3	Excellent: detailed background research is applied to the discussion core and advanced functions and GDPR has been considered in the context of the app design. Excellent discussion in two sections.
B1	Very Good: detailed background research is applied to the discussion core and advanced functions and GDPR has been considered in the context of the app design.
B2	Good: detailed background research is applied to the discussion core and advanced functions and GDPR has been considered discussed at a general level.
C	Basic: each section, of background, core and advanced functions and GDPR has been considered discussed at a general level
D	Below threshold: Section represents some preparation, but is incomplete in coverage of background, core and advanced functions and GDPR
E	Well below threshold: Section does not reflect adequate planning to produce an app based on the scenario

Grade	Report: Implementation (20%)
A1	Exceptional: each of the section, of user interface, technology stack and data organisation are discussed at an excellent level with a high level of detail and supporting diagrams and appropriate use of design tools. Alternative technology stacks have been compared. The app appears commercially viable.
A2	Outstanding: each of the section, of user interface, technology stack and data organisation are discussed at an excellent level with a high level of detail and

	supporting diagrams and appropriate use of design tools. Alternative technology stacks have been compared.
A3	Excellent: each of the section, of user interface, technology stack and data organisation are discussed at an excellent level with a high level of detail and supporting diagrams and appropriate use of design tools.
B1	Very Good: each of the section, of user interface, technology stack and data organisation are discussed at a good level with good detail and supporting diagrams and appropriate use of design tools.
B2	Good: each of the section, of user interface, technology stack and data organisation are discussed at a good level with good detail and supporting diagrams.
C	Basic: each of the sections, of user interface, technology stack and data organisation are discussed at a general level.
D	Below threshold: Section represents some preparation, but is incomplete in coverage of user interface, technology stack and data organisation
E	Well below threshold: Section does not reflect adequate planning to produce an app based on the scenario

Grade	Report: Hosting (10%)
A1	Exceptional: An appropriate host has been identified and features of the service described at a detailed level. Detailed consideration has been given to scale-ability and tracking and the hosting solution is a very good match for the proposed app.
A2	Outstanding: An appropriate host has been identified and features of the service described at a detailed level. Detailed consideration has been given to scale-ability and tracking.
A3	Excellent: An appropriate host has been identified and features of the service described at a detailed level. Good consideration has been given to scale-ability and tracking.
B1	Very Good: An appropriate host has been identified and features of the service described at a detailed level. Good consideration has been given to scale-ability or tracking
B2	Good: An appropriate host has been identified and features of the service described at a detailed level.

C	Basic: An appropriate host has been identified and features of the service described at a general level.
D	Below threshold: Section represents some preparation, but the features of the hosting service are not fully described.
E	Well below threshold: Section does not reflect adequate planning to host an app based on the scenario

<b>Grade</b>	<b>Report: Presentation (10%)</b>  <b>Structure, intro, conclusion, presentation Referencing</b>
A1	Exceptional: A document which might satisfy the brief in a professional context.
A2	Outstanding: the document matches the structure of the brief and is well written, clearly expressed excellently laid out. Diagrams are of a high standard. Citations and References follow UWS guidelines. Section headings and figure captions are used in TOC. Conclusion is well justified.
A3	Excellent: the document matches the structure of the brief and is well written, clearly expressed excellently laid out. Citations and References follow UWS guidelines. Section headings and figure captions are used in TOC. Conclusion is well justified.
B1	Very Good: the document matches the structure of the brief and is well written, clearly expressed and presented. Citations and References follow UWS guidelines. Section headings and figure captions are used in TOC.
B2	Good: the document matches the structure of the brief and is generally well written and presented. Citations and References follow UWS guidelines.
C	Basic: the document matches the structure of the brief and is generally well written and presented.
D	Below threshold: the document matches the structure of the brief but is marred by significant omissions, poor grammar, errors or poor formatting.
E	Well below threshold: the document does not match the structure of the brief.

<b>Grade</b>	<b>Site: (40%)</b>
A1	Exceptional: Reflects the quality of a commercially developed application.
A2	Outstanding: extends the minimum functionality required by the scenario in several areas with good visual design and implementation and a high degree of originality
A3	Excellent: extends the minimum functionality required by the scenario in several areas with good visual design and implementation
B1	Very Good: extends the minimum functionality required by the scenario with good visual design and appropriate technology
B2	Good: fulfils the minimum functionality required by the scenario with good visual design and appropriate technology
C	Basic: fulfils the minimum functionality required by the scenario
D	Below threshold: incomplete, not meeting the minimum functionality required by the scenario
E	Well below threshold: inappropriate or non-working app

Any marks visible on Aula before the School Assessment Board have met are subject to ratification by moderator and external examiner.

## **Module descriptor**

A full copy of the module descriptor may be found on Aula.