Unit 1: Foundations of Web-Based Applications and eLearning Systems

Unit 1 — Moodle Introduction

This unit introduces the conceptual and architectural foundations of web-based applications and eLearning systems. Because the content is highly abstract and terminologically dense, you are strongly encouraged to engage actively with the H5P interactive video, complete the pre-video diagnostic quiz before watching, and post your reflection in the discussion forum after. There are no programming tasks in this unit; the goal is to build accurate mental models before any code is written.

UDL NOTE

Multiple Means of Engagement (UDL Principle 3): The pre-video quiz activates prior knowledge and sets a clear purpose for learning (UDL Guideline 8.1). The discussion forum fosters community and sustained engagement (UDL Guideline 8.3). Optional audio transcripts of all videos are available for learners who prefer listening without visual distraction.

Unit Aim

The aim of Unit 1 is to establish the conceptual architecture of web-based applications and eLearning systems, enabling students to accurately explain how these systems are structured, how their components interact, and what professional, ethical, and accessibility obligations apply to educational web developers.

Intended Learning Outcomes

By the end of Unit 1, you should be able to:

- Explain the conceptual architecture of web-based applications, identifying the function of each component and describing how they interact via the HTTP/HTTPS request-response cycle (Bloom's Level 2 — Understand).
- Distinguish among the major architectural layers of a Learning Management System, specifically Moodle, relating each layer to its educational role (Bloom's Level 2 — Understand).
- Evaluate the eLearning interoperability standards SCORM and xAPI, explaining what each standard enables and why it matters for content portability (Bloom's Level 2 — Understand).
- Identify the ethical and accessibility obligations of educational web developers, including WCAG 2.1 principles and data privacy law (Bloom's Level 1 — Remember).

ID NOTE

Outcome Alignment: Each ILO is tagged with Bloom's Taxonomy level to ensure assessments match cognitive demand. Unit 1 intentionally targets Levels 1–2 (Remember/Understand) because students need conceptual scaffolding before progressing to higher-order creation tasks in Units 2–5. This follows the ADDIE Design principle of sequencing from lower to higher cognitive complexity (Anderson & Krathwohl, 2001).

1.1 What is a Web-Based Application?

A web-based application (also called a web app) is software that runs on a remote server and is accessed by a user through a web browser over the internet or an intranet. Unlike a desktop application — which is installed and runs locally on the user's machine — a web-based application requires no local installation. The user simply opens a browser, navigates to a URL, and interacts with the application.

Definition: Web-Based Application

A software application that is hosted on a web server, accessed via a web browser using the HTTP or HTTPS protocol, and capable of performing complex operations on the server side, storing data in a database, and presenting dynamic content to the user.

MULTIMEDIA

Pedagogically Meaningful Graphic: An annotated diagram illustrating the difference between desktop and web application architecture is included in the H5P Interactive Video at timestamp 01:30. The graphic applies the Signalling Principle (Mayer, 2009) by using arrows and colour to direct learner attention to the key distinction: data resides on the server, not the local device.

Examples of web-based applications include: the University of Rwanda eLearning Platform (Moodle), Google Docs, online banking portals, and social media platforms. In educational contexts, web-based applications enable any-time, any-place learning, automated assessment and feedback, and scalable delivery of course content to thousands of learners simultaneously.

1.1.1 Web Apps vs. Desktop Applications

Characteristic	Desktop Application vs. Web-Based Application
Location of software	Local device (installed) vs. Remote server (no installation)
Update mechanism	Manual download by user vs. Updated centrally by developer
Access device	Must be installed on each device vs. Any device with a browser
Internet required	Not always vs. Yes (or intranet)
Example	Microsoft Word vs. Moodle, Google Docs

1.2 The Client-Server Model

All web-based applications are built on the client-server model. This model describes the relationship between two categories of computing entities: the client, which requests resources or services, and the server, which provides those resources or services in response.

Definition: Client

Any device or application that sends requests to a server in order to retrieve or submit data. In web applications, the client is typically a web browser (Chrome, Firefox, Edge) running on the user's computer or mobile device.

Definition: Server

A computer or software system that receives requests from clients, processes them, and returns responses. In web applications, the server hosts the application code and is responsible for accessing the database and generating the appropriate response.

1.2.1 The HTTP/HTTPS Request-Response Cycle

Communication between the client and the server follows a protocol called HTTP (HyperText Transfer Protocol) or its secure version HTTPS. Every interaction between a user and a web application is an exchange of HTTP messages following a request-response pattern:

1. The user performs an action in the browser — for example, submitting an assignment on Moodle.
2. The browser constructs an HTTP request (typically a POST request for form submission) and sends it to the server.
3. The request travels across the internet to the server hosting the web application.
4. The server receives the request, runs the appropriate application code, and queries the database if necessary.
5. The server constructs an HTTP response and sends it back to the browser.
6. The browser receives the response and renders the appropriate content for the user.

INTERACTIVE *Articulate/H5P Interactive Element: At this point in the H5P video, an interactive drag-and-drop activity asks learners to sequence the six steps of the HTTP request-response cycle. This aligns with UDL Guideline 9.2 (Facilitate personal coping skills and strategies) by giving learners immediate corrective feedback before proceeding. The activity is also available as a standalone SCORM package for offline use.*

HTTP Method	Purpose and Educational Example
GET	Requests data from the server (e.g., loading a course page in Moodle). Data is sent in the URL query string. Should not change server state.
POST	Submits data to the server (e.g., submitting a quiz answer or uploading a file). Data is sent in the request body. Changes server state.
PUT / PATCH	Updates an existing resource (e.g., editing a course description). Full replacement (PUT) or partial update (PATCH).
DELETE	Removes a resource from the server (e.g., deleting a student submission). Must be used with caution in educational systems.

Security Note: HTTPS

Educational web applications must use HTTPS rather than plain HTTP. HTTPS encrypts all data transmitted between the browser and the server using TLS, preventing eavesdropping on sensitive information such as student passwords, assessment submissions, and personal data. HTTPS is also a requirement for WCAG 2.1 Level AA compliance and Rwanda's Law No. 058/2021 on Personal Data Protection.

1.3 The Three-Tier Architecture

Modern web applications — including Moodle — are organised according to the three-tier architecture, which separates the application into three distinct functional layers. This separation improves maintainability, scalability, and security.

Tier	Description and Moodle Example
Tier 1: Presentation	The user-facing layer. Consists of HTML, CSS, and JavaScript running in the browser. In Moodle: Bootstrap-based themes, Mustache templates, and the JavaScript-rendered course interface.
Tier 2: Application Logic	The business logic layer. Runs on the server and processes user requests, enforces rules (e.g., quiz time limits, enrolment restrictions), and mediates between the presentation tier and the data tier. In Moodle: PHP activity modules, authentication plugins, and the Moodle API.
Tier 3: Data	The persistence layer. Stores all application data in a relational database. In Moodle: a MySQL or PostgreSQL database containing tables prefixed with mdl_ (e.g., mdl_user, mdl_course, mdl_quiz_attempts), as well as the moodledata file system for uploaded files.

ID NOTE

Advance Organiser (Ausubel, 1968): The three-tier architecture diagram serves as an advance organiser—a conceptual scaffold that helps students organise new knowledge as they encounter HTML (Unit 2), SQL (Unit 3), and PHP/Node.js (Unit 4). It is repeated in units 2–4 with the relevant tier highlighted, reinforcing the overall mental model across the module.

1.4 Learning Management Systems: Moodle

A Learning Management System (LMS) is a web-based application specifically designed for the administration, delivery, tracking, and assessment of educational content. The University of Rwanda uses Moodle (Modular Object-Oriented Dynamic Learning Environment) as its primary LMS.

Definition: Learning Management System (LMS)

A software platform that enables educational institutions to create, deliver, manage, and assess online and blended learning programmes. Key features include course management, user role administration, assignment submission, quiz delivery, discussion forums, and learning analytics.

1.4.1 Moodle's User Role Hierarchy

- Administrator: Full system access. Manages site settings, installed plugins, and user accounts.
- Manager: Can access and manage all courses. Used for programme directors or eLearning coordinators.
- Course Creator: Can create new courses but cannot manage site-level settings.

- Teacher / Instructor: Full access within assigned courses. Can create activities, grade submissions, and access student analytics.
- Non-editing Teacher: Can view course content and grade submissions but cannot modify course structure.
- Student: Can access enrolled courses, submit activities, and receive feedback.

UDL NOTE *Multiple Means of Expression (UDL Principle 2): In the accompanying forum task, students may demonstrate understanding of Moodle role hierarchies through text posts, annotated screenshots, or a short screen-recording narration. This flexibility accommodates learners with different communication preferences and technological skills (CAST, 2018, Guideline 5.2).*

1.4.2 Key Moodle Activity Types

Activity Type	Description and Educational Use
Quiz	Automatically graded assessments with multiple question types (MCQ, short answer, matching). Supports time limits, adaptive mode, and instant feedback.
Assignment	Allows students to submit text online or upload files. Supports group submission and grading with rubrics.
Workshop	Peer-assessment activity. Students submit work and review peers' submissions using a defined rubric.
Forum	Asynchronous discussion tool. Supports standard, Q&A, and blog formats. Essential for Discussion and Collaboration learning activities.
H5P Activity	Interactive content module supporting video quizzes, interactive presentations, drag-and-drop activities, and more.
SCORM Package	Allows upload of externally authored SCORM-compliant courses that communicate completion and score data back to Moodle.

1.5 eLearning Interoperability Standards

Interoperability standards define how eLearning content packages communicate with Learning Management Systems. Without standards, a course developed in one authoring tool could not function on a different LMS, and learner progress data would be lost when platforms change. Two standards are essential knowledge for educational web developers: SCORM and xAPI.

1.5.1 SCORM (Sharable Content Object Reference Model)

SCORM is the dominant legacy eLearning interoperability standard. A SCORM-compliant course is packaged as a ZIP file containing HTML, CSS, JavaScript, and a manifest file (imsmanifest.xml) that describes the course structure to the LMS. When a learner completes a SCORM module, the content uses a JavaScript API defined by the standard to send completion status, score, and time-on-task data back to the LMS.

Version	Characteristics

SCORM 1.2	Most widely supported version. Limited data model (completion and score only). Still in use at many institutions.
SCORM 2004	Richer data model and sequencing rules. Less widely implemented by LMS vendors. Moodle supports both versions.

1.5.2 xAPI (Experience API / Tin Can API)

xAPI is the modern successor to SCORM. Unlike SCORM—which can only track learning that occurs within the LMS—xAPI can record any learning experience from anywhere: a mobile app, a game, a simulation, a workplace activity, or a website. xAPI expresses every learning event as a simple Actor-Verb-Object statement sent to a Learning Record Store (LRS).

Characteristic	SCORM vs. xAPI
Scope	Only within the LMS vs. Anywhere (LMS, mobile, simulation, workplace)
Data format	SCORM JavaScript API calls vs. JSON statements to a REST API
Storage	LMS database vs. Learning Record Store (LRS)
Richness	Completion, score, time vs. Any verb, any object, any context
Moodle support	Full (built-in) vs. Partial (via plugins and H5P)

1.6 Ethics and Accessibility in Educational Web Development

1.6.1 Web Content Accessibility Guidelines (WCAG 2.1)

Principle	Definition and Example
Perceivable	All information must be presentable to users in ways they can perceive. Example: Providing text alternatives for images (alt text) and captions for video.
Operable	All user interface components must be operable. Example: All functionality must be achievable using the keyboard alone (no mouse required).
Understandable	Information and UI operation must be understandable. Example: Form fields must have clear labels; error messages must be descriptive.
Robust	Content must be robust enough to be interpreted by a wide variety of current and future user agents, including assistive technologies. Example: Valid HTML that screen readers can parse correctly.

1.6.2 Data Privacy: Rwanda Law No. 058/2021

- Data Minimisation: Collect only the data that is strictly necessary for the declared purpose.

- Purpose Limitation: Use collected data only for the purpose for which it was collected; do not repurpose it without consent.
- Security by Design: Build security controls (encryption, access controls, input validation) into the application from the start.
- Transparency: Inform users clearly what data is collected, why, and how it will be used (privacy notice).
- Right to Erasure: Users have the right to request deletion of their personal data.
- Breach Notification: Security breaches affecting personal data must be reported to the National Cyber Security Authority (NCSA) and affected individuals within defined timeframes.

Unit 1 Assessments — ABC Learning Design Framework

ASSESSMENT

Formative and Summative Balance: Unit 1 includes both ungraded formative activities (pre-video quiz, H5P video) and graded summative activities (Forum, Concept Map, Unit Quiz). This approach follows the Assessment for Learning principle—formative tasks help students identify gaps before being formally assessed (Black & Wiliam, 1998).

Activity	ABC Type	Description	Bloom's Level	Weight
Pre-Video Quiz	Acquisition / Investigation	A 5-question ungraded Moodle Quiz to activate prior knowledge before watching the instructional video. Students identify what they already know about web applications and LMS systems.	Remember (L1)	Ungraded
H5P Interactive Video	Acquisition / Practice	A 10-minute narrated instructional video covering the client-server model, three-tier architecture, Moodle structure, SCORM, xAPI, and WCAG 2.1. Two embedded comprehension-check questions require learner response during playback.	Understand (L2)	Formative
Discussion Forum — E-tivity 1	Discussion / Production	Post a minimum 200-word reflection: 'Choose a feature of Moodle you use regularly and describe which of the three architectural tiers is involved at each step.' Respond to at least one peer's post with constructive academic engagement.	Understand (L2)	5% (ICA)

Concept Map — E-tivity 2	Production / Investigation	Submit a concept map (hand-drawn, Canva, or draw.io) showing the relationships among: web-based application, client, server, HTTP, three-tier architecture, LMS, SCORM, xAPI, WCAG, and data privacy. Include at least one labelled connection between each pair of adjacent concepts.	Understand (L2)	5% (ICA)
Unit 1 Formative Quiz — E-assessment 1	Practice	A 10-question automatically graded Moodle Quiz covering all ILOs of Unit 1. Immediate feedback provided. Students may attempt twice; the higher score is recorded.	Remember / Understand	5% (ICA)
SCORM Scenario Activity — E-tivity 3	Practice / Acquisition	A SCORM package authored in Articulate Storyline presenting four branched scenarios (e.g., 'A student's SCORM score does not appear in Moodle—what could cause this?'). Learners choose responses and receive contextualised feedback. Completion is tracked automatically by Moodle.	Understand / Apply (L2–L3)	Formative

End-Unit Assessment 1: Moodle Quiz — Foundations of Web Applications

Ready-to-Use Moodle Quiz — Copy & Paste Questions

The following questions are formatted for direct entry into Moodle's Question Bank (Quiz > Edit Quiz > Add Question). Use the 'Multiple Choice' question type unless otherwise specified. Correct answers are marked with ✓. Set: Attempts allowed = 2, Grading method = Highest grade, Shuffle answers = Yes.

Question & Answer	Moodle Format Notes
Q1. Which protocol ensures encrypted data transmission between a student's browser and the Moodle server? ✓ A) HTTPS B) HTTP C) FTP D) SMTP Feedback: HTTPS uses TLS/SSL encryption to protect all data in transit, including student passwords and assessment submissions. HTTP sends data in plaintext and must not be used for educational applications handling personal data.	

Q2. In Moodle's three-tier architecture, which tier is responsible for enforcing quiz time limits? ✓ A) Application Logic (Tier 2) B) Presentation (Tier 1) C) Data (Tier 3) D) Client (browser) Feedback: Quiz rules (time limits, adaptive mode, attempt restrictions) are business logic enforced by Moodle's PHP modules running in Tier 2.	
Q3. A lecturer at UR exports a completed eLearning module from Articulate Storyline as a ZIP file and uploads it to Moodle. Which eLearning standard does this package use? ✓ A) SCORM B) xAPI C) LTI D) AICC Feedback: Articulate Storyline exports SCORM-compliant packages as ZIP files containing imsmanifest.xml, which Moodle reads to import and track the module.	
Q4. Which WCAG 2.1 success criterion requires that every image on an eLearning page has a text alternative? ✓ A) 1.1.1 Non-text Content B) 1.4.3 Contrast (Minimum) C) 2.1.1 Keyboard D) 3.1.1 Language of Page Feedback: WCAG 2.1 SC 1.1.1 requires that all non-text content has a text alternative (alt attribute) so screen reader users receive equivalent information.	
Q5. Rwanda's Law No. 058/2021 requires that a university web application must notify the NCSA within a defined timeframe if there is a data breach. Which data protection principle does this represent? ✓ A) Breach Notification B) Data Minimisation C) Purpose Limitation D) Right to Erasure Feedback: The Breach Notification principle mandates that processors inform both the regulator (NCSA) and affected individuals promptly when a security breach compromises personal data.	
Q6. In an xAPI statement, what does the 'Verb' represent? ✓ A) The learning action performed (e.g., 'completed', 'passed') B) The learner's name C) The course title D) The Learning Record Store address Feedback: An xAPI statement follows the Actor-Verb-Object pattern. The Verb is a URI-based descriptor of what the learner did (e.g., http://adlnet.gov/expapi/verbs/completed).	
Q7. A student visits the Moodle course page. Their browser sends a request. Which HTTP method is used? ✓ A) GET B) POST C) PUT D) DELETE Feedback: GET requests retrieve data from the server without modifying it. Loading a Moodle course page is a read-only operation.	
Q8. Which Moodle role can create activities and grade submissions but cannot create new courses? ✓ A) Teacher B) Administrator C) Manager D) Course Creator Feedback: The Teacher role has full access within assigned courses but cannot create courses at the site level. Only Course Creators, Managers, and Administrators can create courses.	
Q9. Why does Moodle use the mdl_ prefix for all database table names? ✓ A) To allow multiple Moodle installations to share one database without table name conflicts B) To improve query performance C) To comply with SCORM standards D) Because MySQL requires it for PHP applications Feedback: The mdl_ prefix (configurable) allows Moodle to be installed alongside other applications in the same database by namespacing its tables.	
Q10. [Short Answer] In your own words, define the term 'three-tier architecture' and name one example of each tier in Moodle. Model Answer: Three-tier architecture separates a web application into three layers: (1) Presentation—the user-facing layer (Moodle's Bootstrap-based theme and HTML/CSS interface); (2) Application Logic—the server-side processing layer (Moodle's PHP modules that enforce quiz rules and manage enrolments); and (3) Data—the persistence layer (Moodle's MySQL database storing tables such as mdl_user and mdl_course).	

End-Unit Assessment 2: Concept Map Rubric

Grading Rubric — Unit 1 Concept Map (5% ICA)

Upload to Moodle: Assignment > Advanced Grading > Grading Guide. This rubric defines transparent evaluation criteria aligned to the ILO targeting Bloom's Level 2 (Understand).

Criterion	Distinction (75–100%)	Merit (60–74%)	Pass (50–59%)	Fail (<50%)
Conceptual Accuracy	All 10 required concepts correctly represented with accurate labels and definitions shown through connections.	8–9 concepts correctly represented with minor definitional inaccuracies.	6–7 concepts represented; some connections are inaccurate or missing.	Fewer than 6 concepts represented; significant conceptual errors present.
Relationship Labelling	All connections are labelled with precise relational language (e.g., 'communicates via', 'stores data in'). Labels are academically accurate.	Most connections labelled; relational language is generally accurate.	Some connections labelled; labels are vague or partially inaccurate.	Few or no connections labelled; relationships unclear.
Academic Language	Terminology is consistently precise and academically appropriate throughout the map.	Mostly precise terminology with one or two informal or imprecise terms.	Terminology is inconsistent; several informal usages.	Terminology is largely informal or incorrect.
Visual Clarity	Map is clearly structured, legible, and logically organised. Hierarchy or grouping is visible.	Map is mostly clear with minor layout issues.	Map is readable but organisation is unclear in places.	Map is difficult to read or understand.