# Assignment 2 – Application Systems Design Report

**Weight: 80%**

**Rationale:**

The purpose of this assignment is to demonstrate your ability to analyse and design a solution for an application system proposal.

**Group Work:**

Being able to collaborate on tasks, especially in virtual environments, is an important skill to master to ensure your success at university and in your later careers. To facilitate your development of communication and collaboration skills, you will be undertaking this assignment in a **group of three** (3) students. Group members must be enrolled in the same course code (i.e., 1803ICT students group together and 7610ICT students may form groups).

**Task:**

You will produce an analysis and design report for an innovative application system proposal chosen from the following five scenario options. Choose just **one** scenario that your group will use for this assignment.

The scenario options are:

1. Virtual Reality Shopping Experience
2. AI-Powered Music Collaboration Platform
3. Smart Home Automation System
4. Gamified Task Management Platform for Freelancers
5. Blockchain-Based Digital Identity Management System

Full descriptions of the scenarios are provided on the “Assignment 2 Scenarios” page in the Welcome module of the course site. Before choosing a scenario, ensure you carefullyread the details of the one that interests you most.

You will incrementally produce your assignment report by applying the concepts and techniques covered each week to your chosen scenario, and documenting the output of the analysis and design. It will be necessary to return to your chosen scenario’s description weekly as you work through this assignment.

For each week’s course module, you are required to produce a corresponding section of your report, as outlined below. For each report section we have provided a list of possible areas you might address. You **do not need to include all items**, instead choose as many or as few that you feel will best evidence your analysis and design for your chosen scenario.

Remember, your finished report should be logically structured, and each section should be comprehensive and form a coherent unity.

Refer to the Marking Rubric for further guidance.

### 1 Gather and Analyse Requirements

**Analyse complex requirements gathering scenarios and evaluate appropriate techniques. Conduct a comprehensive requirements analysis for your chosen scenario and create a detailed requirements document.**

Some areas you might address include:

* **Understand the Scenario**
  + Read and analyse the description of your chosen scenario.
  + Identify the key stakeholders, the domain of the project, and any initial constraints or challenges.
* **Select Appropriate Techniques**
  + Based on the scenario, for **one** given stakeholder group, choose the most suitable requirements gathering technique (e.g., interviews, workshops, surveys, document analysis, use case analysis, stakeholder analysis).
  + Justify the selected technique based on the scenario's specific needs and the characteristics of the stakeholder group.
* **Plan the Requirements Gathering Activity**
  + Outline a plan for how you will conduct the requirements gathering activity (e.g., schedule meetings with stakeholders, prepare interview guides or survey questions, and set up any necessary tools or materials).
* **Create a Detailed Requirements Document**
  + Write a comprehensive requirements document that includes:
    1. An introduction to the project and its objectives.
    2. A description of the stakeholder involvement and requirements gathering process.
    3. A detailed list of functional requirements (what the system should do).
    4. A list of non-functional requirements (qualities the system should have, such as performance, security, and usability).
    5. Any system models, use cases or user stories that help illustrate the requirements.
    6. A traceability matrix to show how requirements relate to project goals and stakeholder needs.

### 2 Design System Architecture and Select Application Type

**Design a comprehensive system architecture and justify the selected application type for your scenario.**

Some areas you might address include:

* **Select Application Type**
  + Based on the requirements and the nature of the scenario, decide on the most appropriate application type (e.g., web application, mobile application, desktop application, IoT application).
  + Justify your choice by explaining how the selected application type meets the specific needs of the scenario.
* **Choose the Technology Stack**
  + Select the technology stack that will be used to build the system.
  + Consider factors such as programming languages, frameworks, databases, and any other tools or services that will be integrated.
  + Justify your choices based on their suitability for the application type and the system requirements.
* **Design the System Architecture**
  + Sketch the high-level architecture of the system, including the main components and their interactions.
  + Decide on the architecture pattern (e.g., monolithic, microservices, serverless) that best fits the scenario.
  + Consider the data flow, API design, and any external services or third-party integrations.
* Detailed Component Design
  + For each component in the architecture, provide a detailed design that includes the responsibilities, interfaces, and internal structure.
  + Ensure that the design supports the required functionality and meets the non-functional requirements.
* Consider Deployment and Operations
  + Outline the deployment strategy for the system, including infrastructure needs and automation processes.
  + Discuss how the system will be monitored, maintained, and scaled.

### 3 Design User Experience (UX) and Conduct Usability Testing

**Create a high-fidelity UX design prototype and develop a comprehensive usability testing plan for your application system scenario.**

Some areas you might address include:

* Research and Gather Inspiration
  + Review the details of the application scenario, including the target users, the purpose of the application, and any specific requirements or constraints.
  + Study similar applications to understand common UX patterns and trends.
  + Look for design inspiration and best practices that align with the application scenario.
* Define User Flows and Journey Maps
  + Map out the user flows and create journey maps to visualise the steps users will take within the application.
  + Identify key interactions and potential pain points.
* Create Wireframes
  + Develop initial wireframes to layout the basic structure of the application's interfaces.
  + Focus on the placement of elements and the overall navigation.
* Design a High-Fidelity Prototype
  + Using a UX design tool (e.g., Sketch, Adobe XD, Figma), create a high-fidelity prototype that includes detailed designs, interactions, and animations.
  + Ensure the prototype reflects the branding, colour scheme, typography, and visual elements that align with the application's identity.
* Develop a Usability Testing Plan
  + Outline the goals and objectives of the usability testing.
  + Define the target user demographics for testing and the method of recruitment.
  + Choose the usability testing methods (e.g., think-aloud protocol, task-based testing, interviews).
  + Develop a test script with specific tasks and questions for participants.
* Prepare Testing Materials
  + Create a consent form and any necessary pre-test questionnaires.
  + Prepare a moderator guide with instructions for facilitating the testing sessions.
  + Set up the testing environment and ensure all equipment (e.g., computers, recording devices) is ready.
* Conduct Usability Testing (variable time, depending on the number of participants)
  + Recruit participants according to the defined demographics.
  + Conduct the usability testing sessions, observing participants as they interact with the prototype and complete the tasks.
  + Take notes and record sessions for later analysis.
* Analyse Usability Testing Results
  + Review the recordings and notes from the usability testing.
  + Identify patterns in user behaviour, common issues, and areas of confusion.
  + Prioritise findings based on their impact on user experience.

### 4 Integrate and Adapt the Application System

**Develop a detailed integration plan for your application system scenario and existing infrastructure. Propose and justify adaptation strategies for your application system.**

Some areas you might address include:

* Assess the Existing Infrastructure
  + Evaluate the current infrastructure, including hardware, software, networks, and any other relevant components.
  + Identify the strengths, weaknesses, and limitations of the existing setup.
* Analyse the Application System
  + Review the design and specifications of the application system that needs to be integrated.
  + Understand the system's requirements, dependencies, and document any specific features that may impact integration.
* Identify Integration Points
  + Determine where and how the application system will interface with the existing infrastructure.
  + Identify data exchange points, service interactions, and any other integration touchpoints.
* Develop an Integration Strategy
  + Outline a high-level strategy for integrating the application system with the existing infrastructure.
  + Consider using middleware, APIs, or other integration tools and platforms.
* Propose Adaptation Strategies
  + Suggest changes or adaptations to the application system or infrastructure that may be necessary to facilitate integration.
  + Justify these adaptations based on technical compatibility, performance requirements, security considerations, and business goals.
* Plan the Integration Phases
  + Break down the integration process into phases or stages.
  + Define the scope, objectives, and deliverables for each phase.
  + For each phase, outline the specific implementation steps required.
  + Include tasks such as coding, configuration, testing, and deployment.
* Address Data Migration and Synchronisation
  + Develop a plan for migrating data from the existing system to the new application system or for synchronising data between the two.
* Consider Risk Management and Contingency Planning
  + Identify potential risks associated with the integration and propose mitigation strategies.
  + Develop contingency plans for possible issues that may arise during the integration process.
* Estimate Resources and Timeline
  + Estimate the resources needed for the integration, including personnel, time and budget.
  + Create a timeline that outlines the key milestones and deadlines for each phase of the integration.

### 5 Optimise Performance, Security, and Privacy

This section is in two parts.

**A. Identify performance bottlenecks and recommend evidence-based optimisation techniques for your application system scenario.**

Some areas you might address include:

* Review System Architecture
  + Examine the architectural diagram and documentation to understand the flow of data and interactions between system components.
  + Identify potential single points of failure or areas where data processing could be intensive.
* Analyse Common Bottlenecks
  + For each component (e.g., database, server, network), list possible performance issues that typically arise.
  + Consider scenarios such as high-traffic periods, data spikes, or resource-intensive operations.
* Simulate Load and Profiling
  + Describe how you would simulate load on the system using hypothetical tools.
  + Outline the steps for profiling the system to identify where resources are being consumed.
* Recommend Monitoring Tools
  + List and briefly describe monitoring and profiling tools that could be used to track system performance in a real-world scenario.
* Propose Caching Strategies
  + Suggest where caching could be implemented within the system to improve response times.
  + Discuss the types of caching (e.g., database query caching, content caching) and their potential benefits.
* Code Optimisation and Resource Allocation
  + Identify areas in the hypothetical codebase that could be optimised for better performance.
  + Discuss how resource allocation could be adjusted to handle performance demands.
* Security and Performance Trade-offs
  + Consider any security measures that might impact performance and suggest ways to mitigate these effects.

**B. Perform a comprehensive security and privacy audit for your application system and propose well-justified improvements.**

Some areas you might address include:

* Identify Personal or Sensitive Data
  + Identify the types of data the system would handle, especially personal or sensitive information.
* Map Assets and Data Flow
  + List hypothetical critical assets (data, software, hardware) that the system would contain.
  + Map out the speculative flow of data within the system, including storage, processing, and transmission points.
* Evaluate Hypothetical Security Measures
  + Deduce the potential security measures that would be in place, such as authentication, authorisation, encryption, and secure coding practices.
  + Consider the security technologies or protocols that would likely be used.
* Analyse Privacy Implications
  + Prepare a privacy policy and data handling procedures based on the data types identified.
  + Assess how personal data would be collected, used, stored, and shared, ensuring compliance with privacy laws.
  + Justify how your privacy policy and data handling procedures comply with the requirements of privacy laws.
* Identify Potential Vulnerabilities and Threats
  + Use a checklist or framework (e.g., OWASP Top 10) to speculate on security vulnerabilities and threats the system could face.
  + Consider technical vulnerabilities and organisational weaknesses that could be exploited.
* Suggest Security Enhancements
  + For each potential vulnerability or threat, propose specific improvements or additional security controls.
  + Justify recommendations with reasons and potential benefits, considering the balance between security, usability, and cost.
* Recommend Privacy Improvements
  + Advocate for enhancements to the system's privacy practices, such as data minimisation, pseudonymisation, or privacy-enhancing technologies.
  + Suggest ways to strengthen user consent mechanisms and data subject rights fulfilment processes.

### 6 Plan for Maintenance and Evolution

Create a comprehensive maintenance and evolution plan for your application system scenario. If possible, critically analyse real-world case studies of similar application system maintenance and evolution, and derive best practices.

Some areas you might address include:

* Maintenance Strategy
  + Outline a basic strategy for routine maintenance, including regular updates, backups, and monitoring.
* Propose an Evolution Roadmap
  + Write a high-level roadmap for system evolution, focusing on key enhancements and technology upgrades that align with stakeholder feedback.
* Estimate Resources
  + Prepare an estimate of the resources required for maintenance and evolution, including time and budget.
* Define Change Management
  + Outline a simple process for managing changes, including testing and deployment steps.
* Documentation and Training
  + Describe the processes for updating documentation and training materials as part of the evolution process.
* Outline Monitoring and Evaluation
  + Suggest tools and metrics for monitoring system performance and user satisfaction of the system updates and enhancements.
* Document Risk Mitigation Strategies
  + List potential risks during maintenance and updates, and describe the basic strategies for mitigating them.
* Create a Schedule
  + Prepare a detailed schedule for the regular maintenance activities.
  + Develop a concise timeline for the most critical evolution activities.
* Review Case Studies for Similar Systems
  + Analyse case studies to understand how similar systems have been maintained and evolved.
  + Derive Best Practices: Identify best practices and lessons learned from the case studies.

### 7 Address Ethical Considerations

Analyse complex ethical implications of application system design decisions and propose mitigation strategies. Develop a comprehensive set of ethical guidelines for your application system design, considering diverse stakeholder perspectives

Some areas you might address include:

* Review Ethical Theories
  + Provide a summary overview of the key ethical theories and principles relevant to technology and system design.
  + Outline the primary ethical issues that commonly arise in application system design.
* Analyse Design Decisions
  + Briefly describe a few hypothetical scenarios where design decisions led to ethical dilemmas.
  + Analyse the design decisions using the ethical theories and principles.
* Identify Stakeholder Perspectives
  + Revisit stakeholders from Q1 and consider what their interests, concerns, and expectations might be regarding ethical considerations for the application system.
* Assess Ethical Implications
  + Write an evaluation of the ethical implications of your chosen scenario.
  + Consider the potential impact on stakeholders and discuss the foreseeability of outcomes.
* Analyse System Design
  + Review the design of your scenario’s application system, focusing on features, data handling, user interactions, and any other relevant aspects.
  + Identify potential ethical issues that could arise from the system's design.
* Suggest Mitigation Strategies
  + Propose targeted mitigation strategies for each identified ethical issue.
  + Ensure strategies are practical and take into account the technical, social, and legal aspects of the system’s design.
* Document Analysis and Proposals
  + Write a succinct report that summarises the ethical analysis and proposed mitigation strategies.
  + Use your chosen scenario to illustrate key points.
* Best Practice Guidelines
  + Based on the ethical impact analysis and proposed mitigation strategies, write a concise set of best practice guidelines for ethical application system design.
* Research Ethical Principles
  + Review literature on ethical principles commonly applied to technology and system design, such as privacy, security, fairness, transparency, and accountability.
  + Summarise these principles and their relevance to your application system scenario.
* Write Ethical Guidelines
  + Write ethical guidelines that address identified issues and respect the diverse stakeholder perspectives.
  + Make sure the guidelines strike a balance between the interests of different stakeholders.
  + Ensure the guidelines cover key areas such as data privacy, security measures, bias and fairness, user consent, and transparency.

### 8 Research and Apply Emerging Technologies

Propose an additional emerging technology that could further enhance your application system scenario, demonstrating innovation and critical thinking

Some areas you might address include:

* Research Emerging Technologies
  + Investigate and summarise recent technological advancements and trends that could be relevant to application system design.
  + Focus on areas such as artificial intelligence, machine learning, blockchain, Internet of Things (IoT), edge computing, 5G networks, and quantum computing.
* Identify Application Enhancement
  + Review the design of your application system and identify areas that could be improved or features that could be added by including an additional emerging technology.
  + Consider the system's functionality, user experience, security, scalability, and any other relevant factors.
* Select an Emerging Technology
  + Choose an additional emerging technology that seems most promising for enhancing your application system.
  + Describe the technology's maturity, potential impact, and how well it aligns with your system needs and goals.
  + Provide a rationale for selected additional emerging technology.
* Conceptualise Integration
  + Brainstorm how the additional emerging technology could be integrated into your application system.
  + Sketch out a high-level design that incorporates the new technology, considering both the technical and user experience aspects.
* Assess Feasibility and Impact
  + Critically evaluate the feasibility of integrating the additional emerging technology.
  + Consider factors such as cost, required expertise, time to implementation, and potential risks.
  + Assess the potential impact on the system's performance, user base, and market position.
* Develop a Prototype or Proof of Concept
  + If possible, create a prototype or proof of concept that demonstrates the integration of the additional emerging technology into your application system.
  + This could be a mock-up, a small-scale implementation, or a detailed simulation, depending on the emerging technology and tools available.
* Reflect on Innovation and Critical Thinking
  + Reflect on the process of selecting and integrating the emerging technology.
  + Describe how this exercise has demonstrated innovation in applying and integrating emerging technologies in application systems design.
  + Justify how critical thinking and problem-solving has been used to evaluate the additional technology’s suitability and impact on your application system.

### 9 Reflect on Learning and Professional Development

Reflect on your learning journey throughout the course, identifying areas of strength and opportunities for improvement. Develop a professional development plan to address identified gaps and to stay current with industry trends and best practices.

**IMPORTANT**

This is an individual task. **Do not** complete it as a collaborative effort within your groups.  
**All students must submit their own separate document for this section.**

Refer to the Assignment 2 submission page and Marking Rubric for further guidance.

**UPDATED**

This question has been revised and reduced. You will complete it during your Module 9 workshop activities.

A template is provided (download from the activity 9.2 page of the course site) which you will use for preparing your Professional Development Plan. Activity 9.2 lists the steps to take to complete this question. The separate tasks include:

* Develop a Professional Development Plan
  + Outline a plan that addresses the identified gaps and opportunities for improvement.
  + Set specific, measurable, achievable, relevant, and time-bound (SMART) goals for your professional development.
  + Include a variety of learning activities such as online courses, workshops, seminars, and self-study.
  + Consider networking opportunities, such as joining professional associations or attending industry conferences.
* Portfolio
  + Include a link to your personal portfolio.
  + Refer to the “Building a portfolio for assignment 2” page in the Welcome module of the course site for more information.

### Submission Information

**Report submission:**

All diagrams must be part of a single file. Additional files and attachments will not be marked.

All tables, diagrams, and charts must be accompanied by a **one paragraph description** (minimum 100 words) which explains the rationale and logic. Note that presentation, spelling, and grammar are extremely important aspects of your document. Be sure to proofread your work prior to submission.

1. Carefully check your work against the assignment **Marking Rubric** to ensure your report is complete, i.e., it contains all elements which will be assessed.
2. Compile all your diagrams together with your report into a single .PDF file.  
   **Do not** submit a Word or OpenOffice document.
3. Do not compress (i.e., zip) your assignment file.
4. Only one student from each group needs to submit a copy of their group’s assignment report.
5. Submit the assignment report online using the submission upload link on the Assignment 2 page of the course web site.

**Professional Development Plan submission:**

1. Save your Professional Development Plan as a single .PDF file. **Do not** submit a Word or OpenOffice document.
2. Do not compress (i.e., zip) your file.
3. Every student must submit their own document.
4. Submit your Professional Development Plan document online using the submission upload link on the Assignment 2 page of the course web site.