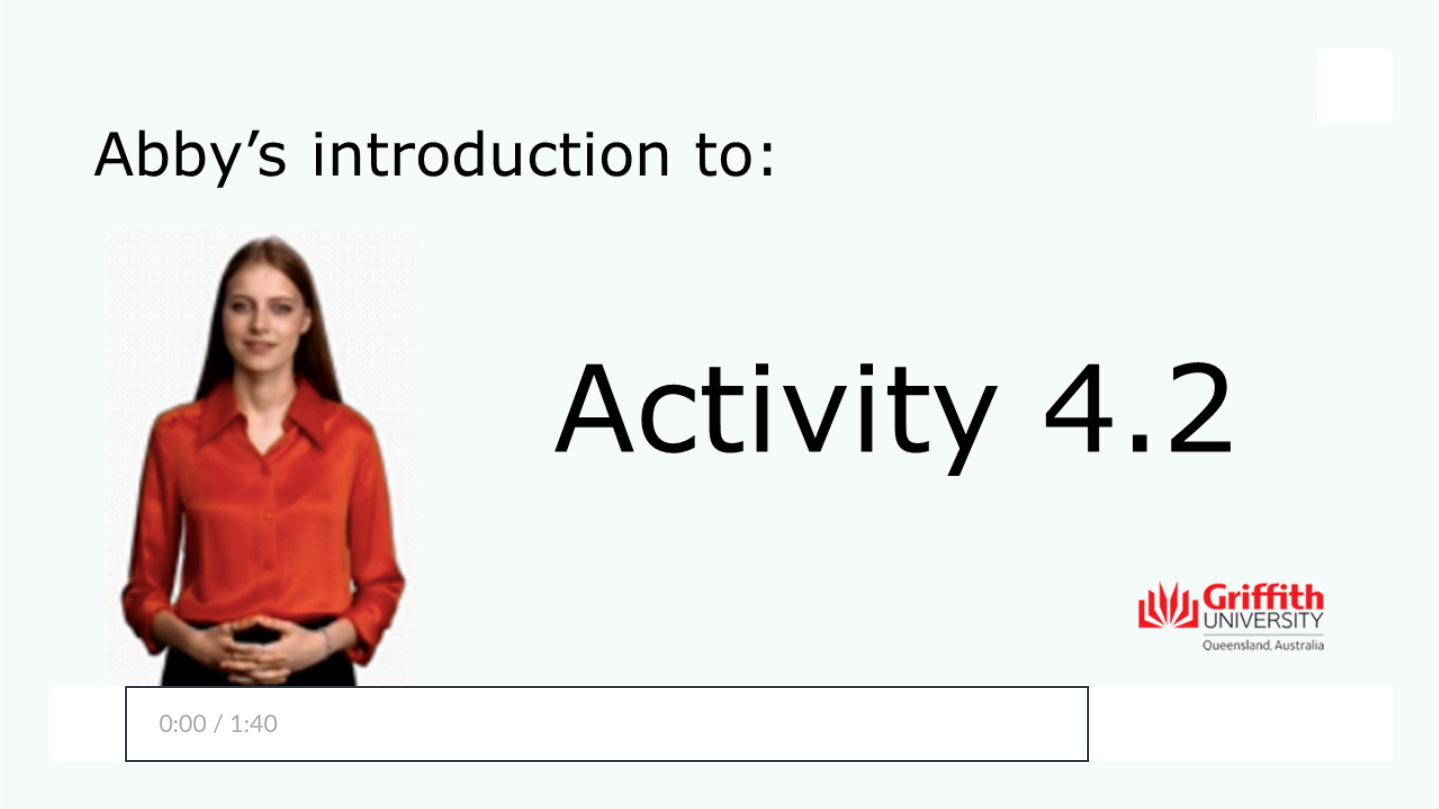
Activity 4.2 Propose and justify adaptation strategies for a given application system to a specific ICT domain

Access course FAQ chatbot (https://lms.griffith.edu.au/courses/24045/pages/welcome-to-the-course-chatbot)

Module 4: Integrate and adapt the application system



What is this activity?

In Activity 4.2, you will propose and justify adaptation strategies for your application system to meet the specific requirements of your chosen ICT domain. Building upon the insights gained from your integration planning in Activity 4.1, you will now critically examine the unique characteristics, constraints, and user needs within your selected domain. By proposing targeted adaptations to your application

system, you will demonstrate your ability to tailor your system to provide maximum value and usability within a specific real-world context.

The final output of Module 4 is a detailed report section that addresses the integration planning for your chosen assignment scenario (https://lms.griffith.edu.au/courses/24045/assignments/93487) . This should include a comprehensive integration plan, risk management strategies, and a clear rationale for the selected integration approach, ensuring a seamless and efficient integration process.

Why is this activity important?

By engaging in this activity, you will develop a deep understanding of the domain-specific requirements and constraints that shape the success of application systems in real-world environments.

Some key benefits of this activity include:

Enhancing the relevance and value of your application system - By carefully examining the specific needs and challenges of an ICT domain, you can propose adaptations that make your system more relevant and valuable to users and stakeholders within that domain.

Improving user experience and adoption - Through proposing adaptations that align with the unique characteristics and expectations of users, you can create an application system that is more intuitive, usable, and likely to be adopted by its target audience.

Demonstrating your ability to think critically and strategically - By justifying your proposed adaptations based on a thorough analysis of domain-specific requirements and constraints, you showcase your ability to think critically and strategically about application system design in real-world contexts.

Developing valuable skills in domain-specific customisation - Through this activity, you will gain practical experience in tailoring application systems to meet the unique demands of specific ICT domains, a valuable skill set that is highly sought after in the field of application system design.

Case study

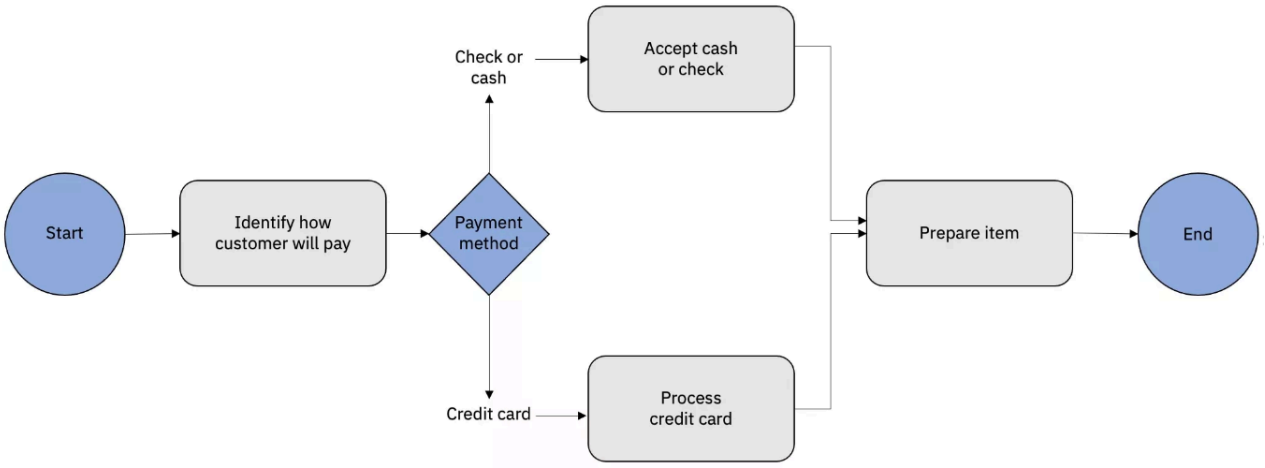
▶ StudySphere - E-Learning Platform

Supporting content for this activity

You should then work through the content elements below. These will reinforce the principles and elements from the StudySphere case study and will provide you with the knowledge and tools that you need to successfully complete this activity.

▼ Supporting content A - Examining business processes and workflows

Techniques for mapping and analysing business processes and workflows within a specific ICT domain



Business Process Modeling Notation (Image source (https://www.ibm.com/blog/bpmn/))

Mapping and analysing business processes and workflows within a specific ICT domain is crucial for understanding how technology can be leveraged to improve efficiency, reduce costs, and enhance service delivery. One technique for mapping these processes is through the use of **Business Process Modeling Notation (BPMN)**, which provides a standardised method for depicting the steps, decisions, and events in a process. BPMN allows stakeholders to visualise the flow of work, identify bottlenecks, and understand the interactions between different parts of the business. By using BPMN, organisations can ensure that the mapping process is consistent and understandable across the entire ICT domain, facilitating better communication and collaboration among team members.

Another technique for analysing business processes is **process mining**, which involves the use of data analytics to uncover insights from event logs. Process mining tools can automatically detect the actual processes that are being executed, as opposed to the processes that are supposed to be executed according to documentation. This technique is particularly useful in ICT domains where digital transactions and logs are abundant. Process mining can reveal deviations, inefficiencies, and opportunities for process optomisation that may not be apparent through traditional mapping methods. By applying process mining, organisations can make data-driven decisions to adapt their ICT systems and align them more closely with the realities of their operational processes.

Value stream mapping (**Image source** (https://www.mangolive.com/value-stream-maspping-lean-management-tools))

In addition to BPMN and process mining, **value stream mapping (VSM)** is a lean manufacturing technique that has been adapted for use in ICT domains. VSM helps identify and eliminate waste in a process, focusing on adding value from the customer's perspective. By applying VSM to ICT workflows, organisations can streamline their operations, reduce cycle times, and improve customer satisfaction. This technique encourages a critical examination of each step in a process, ensuring that every action contributes to the creation of value. When adapting VSM to an ICT context, it is important to consider the unique characteristics of the domain, such as the importance of data integrity and the need for continuous system updates and maintenance.

Finally, **simulation modeling** is a powerful technique for analysing business processes and workflows within an ICT domain. Simulation allows organisations to experiment with different scenarios and process changes without disrupting the actual operations. By creating a digital twin of the current processes, stakeholders can predict the outcomes of various strategies, such as the implementation of new technologies or the reallocation of resources. This technique is particularly useful for complex ICT environments where the interdependencies between systems and processes can lead to unforeseen consequences. Simulation modeling provides a risk-free environment to test and refine adaptation strategies before they are rolled out in the real world.

Best practices for identifying domain-specific process and workflow requirements

Identifying domain-specific process and workflow requirements is a critical step in ensuring that ICT systems are aligned with the unique needs of a particular industry or sector. One best practice is to conduct thorough **domain analysis**, which involves studying the specific characteristics, regulations, and standards of the domain. This includes understanding the domain's terminology, common practices, and the context in which business processes operate. By immersing oneself in the domain, analysts can better identify the nuanced requirements that are essential for effective workflow design.

Another best practice is to engage with **domain experts** and **stakeholders** throughout the requirements identification process. These individuals possess valuable insights into the day-to-day operations and can provide a wealth of information about what works well and what pain points exist within current processes. Workshops, interviews, and focus groups are effective methods for

gathering this information. It is important to ask open-ended questions and encourage honest feedback to uncover hidden requirements that may not be immediately apparent.

Moreover, it is crucial to **leverage existing documentation and artifacts** within the domain, such as process maps, workflow diagrams, and system logs. These materials can serve as a starting point for identifying requirements and can help in understanding the current state of processes and workflows. analysing these documents can reveal patterns, redundancies, and areas for improvement. However, it is important to verify the accuracy of the documentation and supplement it with direct observations and discussions with those who execute the processes.

Lastly, best practices include the use of **requirements management tools** and techniques to systematically capture, organise, and prioritise domain-specific requirements. These tools can help in tracking the relationships between different requirements and ensuring that they are comprehensive and feasible. Additionally, it is important to establish clear criteria for evaluating the importance and feasibility of each requirement. This helps in making informed decisions about which requirements should be included in the system design and which may need to be deferred or rejected due to resource constraints or other considerations. Regularly reviewing and updating the requirements as the understanding of the domain evolves is also essential to maintain their relevance and accuracy.

Strategies for aligning application system design with domain-specific processes and workflows

Aligning application system design with domain-specific processes and workflows is essential for creating ICT solutions that meet the unique needs of a particular industry or sector. One strategy for achieving this alignment is through the adoption of a **user-centered design approach**. This involves engaging with end-users and stakeholders from the outset of the design process to understand their work practices, challenges, and requirements. By incorporating this domain-specific knowledge into the design, developers can create systems that not only automate tasks but also support the complex decision-making and collaborative activities that are characteristic of many domains.

Another strategy is to leverage **domain-specific frameworks, standards**, and **best practices** in the system design. Many industries have established guidelines and protocols that govern how processes and workflows should be structured. Incorporating these into the application design ensures that the system will be compatible with existing domain practices and can facilitate smoother integration with other systems and tools used in the domain. For example, in the healthcare sector, adhering to standards such as HL7 for interoperability can ensure that the application system can effectively exchange information with other healthcare IT systems.

Furthermore, it is important to design application systems with **flexibility** and **scalability** in mind. Domain-specific processes and workflows are subject to change due to technological advancements,

regulatory updates, or shifts in industry practices. An application system that is modular and easily adaptable can accommodate these changes without the need for extensive reengineering. This can be achieved through the use of **service-oriented architecture** (SOA), microservices, or other design patterns that promote loose coupling between system components. By building in this adaptability, the application system can continue to align with domain-specific processes and workflows over time, ensuring its longevity and value to the organisation.

Examples of process and workflow adaptations in various ICT domains

Process and workflow adaptations in various ICT domains often involve tailoring technology solutions to fit the unique requirements and constraints of each sector. Here are some examples:

1. **Healthcare**: Electronic Health Records (EHR) systems have been adapted to comply with healthcare regulations such as HIPAA in the United States, which mandates patient data privacy and security. Workflows in EHR systems are designed to ensure that only authorised personnel can access sensitive patient information and that there is an audit trail for all actions taken. Additionally, these systems are integrated with other healthcare technologies like medical imaging and laboratory systems to provide a comprehensive view of a patient's health.
2. **Finance**: In the finance sector, workflow adaptations often revolve around compliance with financial regulations such as Know Your Customer (KYC) and Anti-Money Laundering (AML) directives. Banking software, for example, includes processes for verifying customer identities and monitoring transactions for suspicious activities. These adaptations help financial institutions to prevent fraud and comply with legal requirements.
3. **Manufacturing**: Enterprise Resource Planning (ERP) systems in the manufacturing domain are adapted to handle complex supply chain processes, inventory management, and production scheduling. These systems often integrate with Internet of Things (IoT) devices on the production floor to provide real-time data on machine performance and production status, allowing for just-in-time manufacturing and predictive maintenance workflows.
4. **Retail**: Point of Sale (POS) systems in the retail sector have been adapted to support omnichannel sales strategies, integrating online and offline shopping experiences. They often include workflows for managing loyalty programs, processing returns, and handling various payment methods, including mobile payments and contactless transactions.

5. **Education**: Learning Management Systems (LMS) have been adapted to support diverse educational processes, from online course delivery and student assessment to tracking attendance and managing grades. These systems often include features for collaboration and communication, tailored to the needs of educators and students in different learning environments.
6. **Government**: Public sector ICT systems are adapted to handle citizen data securely and to comply with open government and transparency regulations. Workflow adaptations may include processes for handling freedom of information requests, digital document management, and e-voting systems that ensure the integrity and accessibility of electoral processes.
7. **Transportation and Logistics**: Fleet management systems are adapted to optomise routes, monitor vehicle performance, and manage driver schedules. These systems often integrate with GPS and telematics data to provide real-time tracking and predictive analytics for maintenance and logistics planning.
8. **Energy**: In the energy sector, smart grid technologies are adapted to manage the distribution of electricity more efficiently, incorporating renewable energy sources and demand response mechanisms. Workflows are designed to handle real-time data from sensors and smart meters, enabling grid operators to respond to fluctuations in supply and demand.

Each of these examples demonstrates how ICT systems can be adapted to the specific processes and workflows of different domains, enhancing efficiency, compliance, and overall performance within those sectors.

▼ Supporting content B - Identifying data and information requirements

Overview of common data and information requirements in different ICT domains

In the realm of ICT, various domains have distinct **data and information requirements** that are crucial for the effective functioning of their respective systems and applications. For instance, in the domain of **healthcare**, data requirements often include patient records, diagnostic information, treatment plans, and real-time health monitoring data. This information is critical for providing accurate diagnoses, personalised treatment, and improving patient outcomes. Privacy and security are paramount in this domain, necessitating robust data protection measures to comply with regulations such as the Health Insurance Portability and Accountability Act (HIPAA).

The **finance** domain, on the other hand, demands precise and timely financial data, including transaction records, market trends, credit scores, and risk assessments. This data is essential for making informed investment decisions, managing portfolios, and complying with financial reporting

standards. The finance sector also places a high emphasis on data security to protect sensitive financial information from cyber threats and to maintain customer trust.

In the domain of **education**, data requirements focus on student performance, learning analytics, curriculum development, and resource management. Educational institutions collect data on student attendance, grades, assessments, and feedback to personalise learning experiences and improve educational outcomes. The integration of technology in education also necessitates the secure storage and management of student data, adhering to privacy laws such as the Family Educational Rights and Privacy Act (FERPA) in the United States.

Each **ICT** domain's unique data and information requirements dictate the need for specialised systems and applications that can handle the specific challenges and demands of that field. Understanding these requirements is fundamental to developing effective ICT solutions that meet the needs of users and comply with relevant regulations.

Best practices for assessing domain-specific data and information needs

Assessing domain-specific data and information needs is a critical step in the development and adaptation of ICT systems. Best practices in this area involve a **systematic approach** that ensures the identification of all necessary data elements and the understanding of how they will be used within the specific domain. One of the primary best practices is **stakeholder engagement**. This involves consulting with domain experts, end-users, and other stakeholders to gather insights into the data requirements. Through interviews, workshops, and surveys, stakeholders can provide valuable information on the types of data they need, the frequency of data updates, and the desired outcomes of data analysis. This collaborative approach ensures that the assessment is comprehensive and aligned with the actual needs of the domain.

Data modelling (Image source (https://www.thedataschool.com.au/john-lyu/data-modeling-101-a-beginners-guide-to-success/))

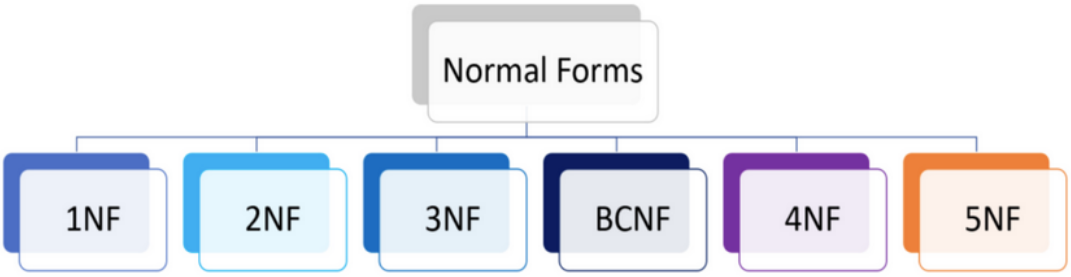
Another best practice is the use of **data modeling and analysis techniques**. These techniques help in understanding the relationships between different data elements and how they contribute to the domain's objectives. By creating data models, one can visualise the data flow, identify gaps or redundancies, and ensure that the data architecture is efficient and scalable. Additionally, analysing existing data within the domain can reveal patterns, trends, and areas for improvement, which can inform the assessment of future data needs.

Finally, it is essential to consider the **ethical**, **legal**, and **regulatory** aspects of data management within the domain. Best practices include conducting a thorough review of the relevant **laws**, **standards**, and **industry best practices** to ensure compliance. This may involve data protection regulations, such as the General Data Protection Regulation (GDPR) in the European Union, or industry-specific standards. Understanding these requirements helps in designing data governance frameworks that not only meet the domain's information needs but also protect the privacy and security of the data. By adhering to these best practices, organisations can effectively assess and address the domain-specific data and information needs, leading to more informed decision-making and improved outcomes within the ICT domain.

Techniques for designing data models and information architectures that meet domain-specific requirements

Designing data models and information architectures that meet domain-specific requirements is a complex task that requires a deep understanding of the domain's unique characteristics and data needs. One of the key techniques in this process is **domain analysis**, which involves studying the domain to identify the entities, attributes, and relationships that are relevant to the domain. This analysis helps in creating a conceptual model that accurately represents the domain's data structures and how they interact. By using domain-specific terminology and concepts, the data model can be made more intuitive and useful for stakeholders within that domain.

Another important technique is the use of **data modeling tools and methodologies**, such as the **Entity-Relationship (ER)** model or the **Unified Modeling Language (UML)**. These tools provide a standardised way to visualise and document data models, making it easier to communicate the design to different stakeholders, including developers, data analysts, and domain experts. They also facilitate the identification of data integrity rules and constraints that are critical for maintaining the quality and accuracy of the domain-specific data.



Database normalisation (Image source (https://guides.visual-paradigm.com/a-comprehensive-guide-to-database-normalization-with-examples/))

Furthermore, it is crucial to design the information architecture with **scalability** and **flexibility** in mind. As domains evolve and new data requirements emerge, the data model should be able to accommodate these changes without significant restructuring. This can be achieved by designing modular and extensible architectures that allow for the addition of new data elements or the modification of existing ones without disrupting the overall system. Techniques such as **normalisation** in relational databases or the use of microservices in software architecture can support this need for adaptability.

Finally, it is important to **validate** the data models and information architectures with **stakeholders** and through **prototyping**. Stakeholder feedback is invaluable for ensuring that the design meets the domain's requirements and is aligned with the users' workflows and expectations. Prototyping allows for the early identification of potential issues and the opportunity to make adjustments before the full implementation. By iteratively refining the design based on feedback and testing, the data models and information architectures can be fine-tuned to better serve the domain-specific needs, resulting in more effective and user-friendly ICT systems.

Examples of data and information adaptations in various ICT domains

Data and information adaptations in various ICT domains often involve tailoring the way data is collected, stored, processed, and presented to meet the specific needs and regulatory requirements of each domain. Here are some examples:

1. **Healthcare**:
    - **Electronic Health Records (EHRs)**: Adaptation involves ensuring that EHR systems can handle a wide range of medical data types, including text, images, audio, and video, while also complying with health data privacy laws like HIPAA.
    - **Telemedicine**: Adapting data transmission protocols to securely transfer patient health information between remote locations and healthcare providers.
    - **Wearable Health Devices**: Designing data models that can integrate continuous streams of biometric data from wearable devices into patient records.
2. **Finance**:

- **Algorithmic Trading**: Adapting data models to process real-time market data feeds at high speeds for algorithmic trading systems.
- **Regulatory Compliance**: Adjusting data storage and reporting systems to meet the requirements of regulations such as the Sarbanes-Oxley Act (SOX) and the Dodd-Frank Wall Street Reform and Consumer Protection Act.
- **Fraud Detection**: Developing advanced analytics models that can detect patterns indicative of fraudulent activities within financial transaction data.

3. **Education**:

   - **Learning Management Systems (LMS)**: Adapting LMS to track and analyse student performance data to personalise learning experiences and provide actionable insights for educators.
   - **Online Assessment Tools**: Designing data models that can securely store and analyse results from online exams, ensuring academic integrity and compliance with privacy laws.
   - **Student Information Systems**: Adapting systems to integrate data from various sources, such as attendance, grades, and behavioural records, to provide a holistic view of student progress.

4. **Manufacturing**:

   - **Supply Chain Management**: Adapting data models to optomise the flow of materials and components across the supply chain, incorporating real-time data from sensors and IoT devices.
   - **Predictive Maintenance**: Developing data models that can analyse machine performance data to predict maintenance needs and prevent downtime.
   - **Quality Control**: Implementing data capture systems that can record and analyse quality metrics throughout the manufacturing process.

5. **Retail**:

   - **Customer Relationship Management (CRM)**: Adapting CRM systems to integrate data from various touchpoints (e.g., social media, online transactions, in-store purchases) to provide a 360-degree view of the customer.
   - **Inventory Management**: Designing data models that can handle real-time inventory updates from multiple locations and sales channels.
   - **Personalised Marketing**: Using customer data to create targeted marketing campaigns that adapt to individual preferences and behaviours.

6. **Transportation and Logistics**:

   - **Fleet Management**: Adapting data models to track vehicle locations, fuel consumption, and maintenance schedules in real time.

- **Route optomisation**: Developing algorithms that can analyse traffic patterns and road conditions to optomise delivery routes.
- **Cargo Tracking**: Implementing IoT-based systems to monitor the condition and location of cargo in transit.

In each of these examples, the adaptations are driven by the unique challenges and opportunities presented by the domain-specific context, ensuring that the ICT systems are not only functional but also add value to the domain's operations and decision-making processes.

▼ Supporting content C - Assessing performance and scalability needs

Techniques for evaluating performance and scalability requirements within a specific ICT domain

When evaluating performance and scalability requirements within a specific ICT domain, it is crucial to employ techniques that are tailored to the unique characteristics and demands of that domain. One such technique is **benchmarking**, which involves comparing the performance of the application system against industry-standard benchmarks or similar systems within the same domain. This can provide insights into how well the system performs in comparison to competitors or established standards, and where improvements may be necessary. Additionally, **stress testing** is a valuable method for determining the limits of the system's scalability by subjecting it to extreme workloads and identifying at what point performance begins to degrade. This information is critical for understanding the system's capacity and planning for future growth.

Another important technique is **profiling**, which involves analysing the system's resource usage and identifying bottlenecks or inefficiencies. By using profiling tools to monitor CPU usage, memory consumption, I/O operations, and network traffic, developers can pinpoint specific areas of the application that may be causing performance issues or limiting scalability. This targeted approach allows for more effective optomisation efforts, as resources can be focused on the most critical areas for improvement. Furthermore, **capacity planning** is essential for assessing scalability needs. By forecasting future demand based on historical usage data and growth trends within the ICT domain, organisations can proactively scale their infrastructure to meet expected increases in load, ensuring that the system remains performant and responsive even as user base or data volume grows.

Lastly, **simulation** and **modeling techniques** can be employed to predict how the system will behave under various load conditions without the need for real-world testing. This can be particularly useful in scenarios where it is impractical or too costly to conduct large-scale tests. By creating a virtual representation of the system and simulating different usage patterns, potential performance issues and scalability limitations can be identified and addressed before they occur in the actual

production environment. This proactive approach helps in designing a more robust and efficient system that is better prepared to handle the specific demands of the ICT domain.

Best practices for designing application systems that meet domain-specific performance and scalability needs

Designing application systems that meet domain-specific performance and scalability needs requires a deep understanding of the unique requirements and constraints of the ICT domain in question. One best practice is to start with a thorough **analysis** of the domain's **performance** and **scalability** demands, which may involve consulting with domain experts, reviewing industry standards, and examining the workload patterns that are typical for the domain. This analysis should inform the selection of appropriate hardware, software, and network infrastructure that can support the expected levels of concurrency, data processing, and transaction rates.

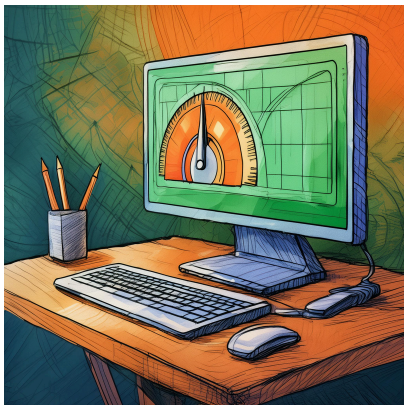
Another best practice is to adopt a **modular and scalable architecture** that allows the system to grow and adapt over time. This often means designing the application with a microservices approach, where each component of the system is a standalone service that can be scaled independently. This not only facilitates scaling but also makes it easier to replace or update individual components without disrupting the entire system. Additionally, leveraging Containerisation and orchestration tools like Docker and Kubernetes can provide the flexibility needed to manage and scale these microservices efficiently.

Performance tuning (**Image source** (https://www.devart.com/dbforge/postgresql/studio/postgresql-performance-tuning-and-optimization.html))

To ensure that the system meets performance expectations, it is crucial to implement **continuous monitoring** and **performance tuning**. This involves setting up real-time monitoring tools that track key performance indicators (KPIs) such as response times, throughput, and error rates. By analysing this data, developers can identify trends and potential bottlenecks before they become critical issues. Regular performance testing, including load testing and stress testing, should be conducted to

validate that the system can handle the expected workloads and to identify areas for optomisation. Finally, establishing a culture of performance awareness and continuous improvement can help ensure that the application system remains aligned with the evolving needs of the ICT domain.

Strategies for optomising application system performance and scalability in domain-specific contexts



Optomising application system performance and scalability in domain-specific contexts requires a strategic approach that considers the unique challenges and opportunities presented by the ICT domain. One key strategy is to focus on **workload-specific optomisations**. This involves analysing the typical workloads of the domain and tailoring the system's resources and algorithms to handle these workloads more efficiently. For example, in a domain with high transaction rates, optomising database access patterns and caching strategies can significantly improve performance. Similarly, in data-intensive domains, implementing efficient data processing pipelines and utilising technologies like in-memory databases or distributed file systems can enhance both performance and scalability.

Another strategy is to leverage **domain-specific hardware and software accelerators**. Many ICT domains have specialised technologies designed to boost performance for particular types of workloads. For instance, using GPUs for parallel processing can dramatically accelerate computations in scientific or AI-driven domains. Similarly, adopting domain-specific languages or frameworks that are optomised for certain tasks can lead to more efficient code execution. By integrating these accelerators into the application system design, developers can achieve performance gains that are not possible with general-purpose solutions.

Lastly, implementing **adaptive scaling strategies** is crucial for maintaining optimal performance as domain demands fluctuate. This involves using auto-scaling mechanisms that can dynamically adjust the system's resources in response to real-time usage patterns. For example, cloud-based applications can benefit from auto-scaling policies that provision additional compute instances during peak usage times and scale down during periods of lower demand. Additionally, designing the system to support **multi-tenancy** and **resource sharing** can improve utilisation and scalability, ensuring that resources are allocated efficiently across different domain-specific applications or user groups. By combining these strategies, organisations can create application systems that are not only highly performant but also resilient to the changing needs of their specific ICT domain.

Examples of performance and scalability adaptations in various ICT domains

Performance and scalability adaptations in various ICT domains often involve specialised techniques and technologies to meet the unique demands of each sector. Here are some examples:

1. **E-commerce**:
   - **Adaptation**: Implementing a distributed cache system like Redis or Memcached to handle product catalogs and user sessions, reducing database load and improving response times.
   - **Example**: Amazon's use of DynamoDB for their shopping cart service, which allows for high availability and scalability during peak shopping seasons.
2. **Financial Services**:
   - **Adaptation**: Utilising low-latency messaging systems such as Kafka or RabbitMQ for real-time transaction processing and event streaming.
   - **Example**: High-frequency trading platforms that use FPGA (Field-Programmable Gate Array) accelerators to execute trades within microseconds.
3. **Healthcare**:
   - **Adaptation**: Deploying a private cloud infrastructure with robust security measures to ensure patient data privacy and compliance with regulations like HIPAA.
   - **Example**: Electronic Health Record (EHR) systems that use blockchain for secure and immutable patient records, ensuring scalability without compromising data integrity.
4. **Telecommunications**:
   - **Adaptation**: Employing network function virtualisation (NFV) and software-defined networking (SDN) to dynamically allocate network resources based on traffic demands.
   - **Example**: 5G networks that use SDN to manage network slices, providing scalable and Customised services for different types of users and applications.
5. **Media and Entertainment**:
   - **Adaptation**: Using content delivery networks (CDNs) to distribute streaming media content across multiple servers geographically, reducing latency and improving user experience.
   - **Example**: Netflix's global CDN, which ensures high-quality video streaming by caching content close to users and dynamically adjusting bitrates based on network conditions.
6. **Internet of Things (IoT)**:
   - **Adaptation**: Implementing edge computing to process data closer to the source, reducing latency and bandwidth requirements.

- **Example**: Smart city infrastructure that uses edge computing to analyse traffic data in real-time, optomising traffic flow without overloading central servers.

7. **Artificial Intelligence and Machine Learning**:
   - **Adaptation**: Training models on GPU or TPU clusters to accelerate the processing of large datasets.
   - **Example**: Google's use of Tensor Processing Units (TPUs) for training and deploying machine learning models at scale, enabling services like Google Photos to recognise and categorise images efficiently.
8. **Big Data Analytics**:
   - **Adaptation**: Adopting distributed storage and processing frameworks like Hadoop or Spark to handle and analyse large volumes of data.
   - **Example**: Apache Hadoop's MapReduce programming model, which allows for the parallel processing of large datasets across clusters of commodity hardware.

Each of these adaptations is tailored to the specific performance and scalability needs of the ICT domain, leveraging technologies and strategies that are best suited to the particular challenges and workloads they face.

▼ Supporting content D - Evaluating security and privacy considerations

Overview of common security and privacy considerations in different ICT domains

In the realm of ICT, **security and privacy considerations** are paramount across various domains, each presenting unique challenges and requirements. In the **healthcare** domain, for instance, the protection of personal health information is critical, with stringent regulations such as the Health Insurance Portability and Accountability Act (HIPAA) in the United States mandating the safeguarding of patient data. Similarly, in the **financial** sector, the security of transactions and the privacy of customer information are of utmost importance, necessitating compliance with standards like the Payment Card Industry Data Security Standard (PCI DSS).

Secure sockets layer (**Image source** (https://whatismyipaddress.com/ssl))

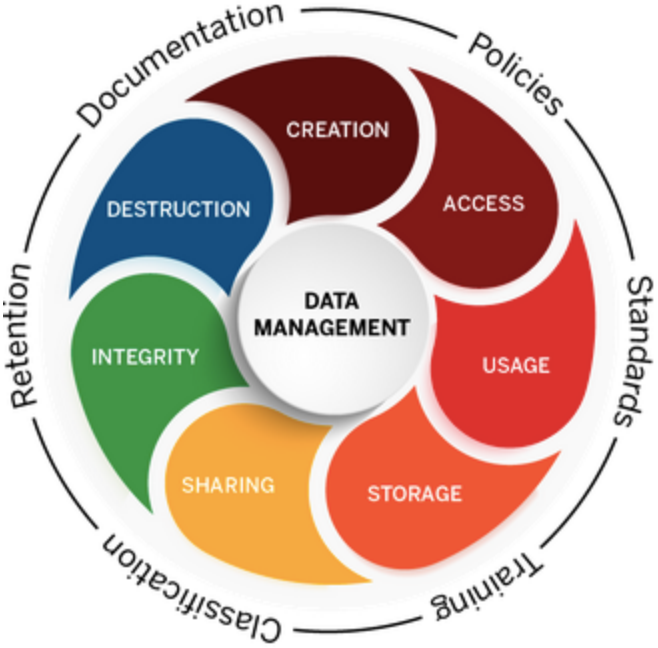
The **e-commerce** domain faces its own set of security and privacy challenges, with the need to secure online transactions and protect customer data from breaches. This domain often relies on technologies such as Secure Sockets Layer (SSL)/Transport Layer Security (TLS) for encrypting data in transit and employs measures like two-factor authentication to enhance account security. Additionally, e-commerce platforms must adhere to privacy laws such as the General Data Protection Regulation (GDPR) in the European Union, which imposes strict rules on data handling and consumer rights.

In the **educational** sector, the security of learning management systems and the privacy of student data are key concerns. Institutions must ensure that educational records are protected from unauthorised access and that student privacy is maintained in accordance with laws like the Family Educational Rights and Privacy Act (FERPA) in the U.S. Furthermore, with the rise of online learning and the use of various educational technologies, there is an increased risk of data breaches and the need for robust cybersecurity measures to safeguard against threats.

Best practices for assessing and addressing domain-specific security and privacy requirements

Assessing and addressing domain-specific security and privacy requirements is a critical process that involves understanding the **unique risks** and **compliance obligations** associated with each ICT domain. Best practices for this process include conducting a thorough **risk assessment** to identify potential vulnerabilities and threats specific to the domain. For example, in the healthcare sector, this might involve evaluating the security of electronic health record systems and ensuring they are HIPAA-compliant. Similarly, in the finance domain, a risk assessment would focus on the security of payment systems and customer data, aligning with PCI DSS standards.

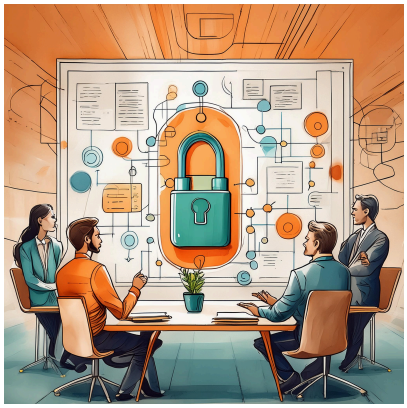
Once the risks are identified, the next best practice is to implement **domain-specific security controls** tailored to mitigate these risks effectively. This could involve the adoption of advanced encryption technologies, access controls, and intrusion detection systems that are appropriate for the domain's data sensitivity and regulatory environment. For instance, in e-commerce, this might mean using multi-factor authentication to protect customer accounts and employing robust data encryption for transactions.



Security and privacy policy (Image source (https://informationsecurity.iu.edu/program/index.html))

Furthermore, it is essential to establish and maintain a comprehensive **security and privacy policy** that not only outlines the controls in place but also includes procedures for regular audits, employee training, and incident response planning. These policies should be continuously updated to adapt to new threats and changes in regulations. Additionally, fostering a **culture of security** awareness within the organisation can help prevent security breaches and ensure that all stakeholders are vigilant about protecting sensitive information in their respective domains.

Techniques for designing application systems that meet domain-specific security and privacy needs



Designing application systems that meet domain-specific security and privacy needs requires a thorough understanding of the regulatory landscape, the types of data being handled, and the potential threats that the system may face. One key technique is to adopt a **security-by-design** approach, where security and privacy considerations are integrated into the system from the ground up, rather than being treated as an afterthought. This involves involving security and privacy experts in the design process and conducting privacy impact assessments to identify and mitigate risks early on.

Another technique is to leverage **domain-specific security standards** and **frameworks**. For example, in the healthcare domain, applications should be designed with HIPAA regulations in mind, ensuring that patient data is protected through appropriate access controls, encryption, and audit mechanisms. Similarly, financial applications should adhere to standards like PCI DSS, which

mandate specific security controls for handling payment card data. By designing applications to meet these standards, organisations can ensure that they are addressing the security and privacy needs of their specific domain.

Additionally, it is crucial to design applications with **flexibility** and **scalability** in mind, as security threats and regulatory requirements are constantly evolving. This means building in the ability to update security measures and privacy policies without significant disruption to the system. Techniques such as modular architecture, where components can be updated or replaced independently, can facilitate this. Furthermore, implementing **application programming interfaces (APIs)** that allow for the integration of additional security layers or privacy-enhancing technologies can future-proof the system against emerging threats and compliance demands.

Examples of security and privacy adaptations in various ICT domains

In the **healthcare** domain, security and privacy adaptations include the use of electronic health records (EHRs) systems that are compliant with HIPAA regulations. These systems often feature role-based access controls to ensure that only authorised personnel can view sensitive patient information. Additionally, healthcare providers may implement secure messaging platforms for communicating patient data, and use encryption technologies to protect data both at rest and in transit.

In the **financial** sector, security and privacy adaptations involve the adoption of multi-factor authentication for online banking platforms to prevent unauthorised access. Financial institutions also employ advanced encryption standards for transaction processing and may use tokenisation to secure credit card information. Furthermore, they often have robust fraud detection systems in place to identify and prevent unauthorised transactions.

For **e-commerce** platforms, security and privacy adaptations include the use of SSL/TLS certificates to encrypt data during online transactions, ensuring that customer information is protected from interception. E-commerce sites may also implement address verification systems (AVS) and card security code (CSC) checks to prevent fraud. Additionally, they often provide privacy settings that allow customers to control how their personal information is used and shared.

In the **educational** sector, adaptations include the implementation of secure learning management systems (LMS) that comply with FERPA regulations, ensuring the privacy of student records. Educational institutions may also use secure email services for communicating sensitive information and employ network monitoring tools to detect and prevent unauthorised access to student data.

Finally, in the **public sector**, government agencies may use secure data centers with strict access controls and employ cybersecurity measures to protect sensitive citizen data. They may also utilise blockchain technology for secure record-keeping and implement strict data governance policies to ensure compliance with privacy laws such as GDPR.

▼ Supporting content E - Analysing user characteristics and expectations

Techniques for researching and analysing user characteristics and expectations within a specific ICT domain

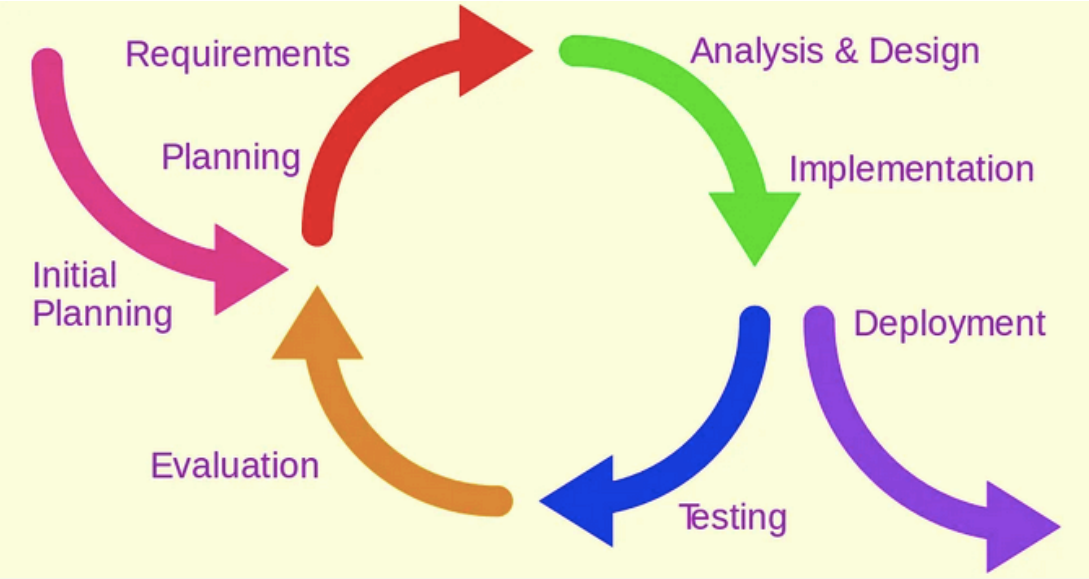
Researching and analysing user characteristics and expectations within a specific ICT domain is crucial for the successful adaptation and development of application systems. One technique for gathering this information is through the use of **surveys** and **questionnaires**. These tools can be designed to target specific user groups within the ICT domain, such as healthcare professionals, financial analysts, or educators. By asking targeted questions about their current use of technology, desired features, and pain points, developers can gain valuable insights into the needs and expectations of their users. This data can then be used to inform the design and functionality of the application system, ensuring that it meets the specific requirements of the domain.

Another effective technique is conducting **focus groups** and **interviews**. These methods allow for a deeper understanding of user characteristics and expectations by providing a platform for open-ended discussion. Focus groups can bring together users from the specific ICT domain to discuss their experiences and expectations in a group setting, while interviews offer a more personalised approach to explore individual user needs and preferences. Both techniques can uncover nuanced insights that might not be captured through quantitative methods alone. The qualitative data collected from these interactions can be invaluable for identifying user-centric design considerations and for validating findings from other research methods.

Usability testing (**Image source** (https://www.hotjar.com/usability-testing/methods/))

Lastly, **usability testing** is a critical technique for analysing user characteristics and expectations. By observing users as they interact with a prototype or existing system, developers can directly assess how well the application meets the needs of its target audience within the ICT domain. This can be done through controlled lab settings or through remote user testing. Usability testing provides immediate feedback on the system's ease of use, effectiveness, and overall satisfaction. The data collected from these tests can be used to make iterative improvements to the application system, ensuring that it aligns with the specific characteristics and expectations of users within the ICT domain.

Best practices for designing application systems that meet domain-specific user needs and expectations



Iterative design (**Image source** (https://www.interaction-design.org/literature/article/design-iteration-brings-powerful-results-so-do-it-again-designer))

Designing application systems that meet domain-specific user needs and expectations requires a deep understanding of the unique challenges and requirements of the target ICT domain. One best practice is to adopt a **user-centered design (UCD)** approach, which places the end-user at the core of the design process. This involves conducting thorough user research, as previously discussed, to inform the design decisions at every stage. By creating **personas** and **user scenarios** based on real data, designers can ensure that the application system addresses the specific needs and expectations of the domain's users. Additionally, **iterative design** and **prototyping**, with frequent user feedback loops, help to refine the application system and make it more responsive to user requirements.

Another best practice is to leverage **domain-specific design patterns and standards**. Each ICT domain often has established conventions and user interface elements that are familiar to its practitioners. For example, in the healthcare domain, certain symbols and terminologies are widely recognised and expected. By incorporating these domain-specific elements into the design,

application systems can be more intuitive and easier to adopt. Furthermore, adhering to **industry standards** can ensure compatibility with other systems and tools within the domain, enhancing the overall user experience.

Lastly, it is crucial to prioritise **accessibility** and **inclusivity** in the design of application systems. Domain-specific users may have varying levels of technical proficiency, physical abilities, and preferences for interaction. Designing with accessibility in mind not only ensures that the application system is usable by a broader audience but also often leads to a better experience for all users. This includes considering factors such as screen reader compatibility, keyboard navigation, and providing options for Customisation. By following these best practices, designers can create application systems that not only meet but exceed the expectations of domain-specific users, leading to higher user satisfaction and adoption rates.

Strategies for creating user-centered designs that align with domain-specific user preferences and behaviours

Creating user-centered designs that align with domain-specific user preferences and behaviours is essential for the success of any application system within an ICT domain. The first strategy is to conduct comprehensive **user research**, which involves ethnographic studies, interviews, and surveys to gather insights into how users within the specific domain interact with technology. This research helps in identifying patterns in user behaviour, preferences, and pain points, which are critical for informing the design process. By understanding the unique context in which users operate, designers can tailor the application system to better fit into the users' workflow and enhance their productivity.

Design thinking (**Image source** (https://www.interaction-design.org/literature/article/what-is-design-thinking-and-why-is-it-so-popular))

A second strategy is to employ **design thinking methodologies**, which are iterative processes that encourage ideation, prototyping, and testing with real users. This approach allows for the rapid exploration of various design solutions and the refinement of ideas based on user feedback. By involving domain-specific users in the design process through workshops and usability testing, designers can ensure that the application system not only meets the users' expectations but also delights them with intuitive and efficient features that cater to their specific needs.

Another key strategy is to create **design prototypes** that are representative of the domain-specific context. These prototypes should be tested in real-world scenarios to validate their effectiveness. This could involve creating clickable prototypes or minimum viable products that users can interact

with in their actual work environment. Observing users as they engage with these prototypes provides invaluable data on how well the design aligns with their preferences and behaviours. This feedback loop is crucial for making informed design decisions and iterating towards a final product that resonates with the target user base.

Lastly, it is important to establish a **collaborative relationship** with domain experts and users throughout the design process. This can be achieved by setting up user panels, advisory boards, or continuous feedback channels. Such collaborations ensure that the design team stays informed about the evolving needs and expectations within the domain. Moreover, it fosters a sense of ownership among users, making them more likely to embrace the final application system. By integrating these strategies, designers can create user-centered designs that are not only functional but also deeply aligned with the unique characteristics and behaviours of domain-specific users.

Examples of user-centered adaptations in various ICT domains

User-centered adaptations in various ICT domains involve tailoring technology to meet the specific needs, preferences, and behaviours of users within those domains. Here are some examples:

1. **Healthcare**: Electronic Health Records (EHR) systems have been adapted to include clinical workflow optomisations that reduce the time physicians spend on data entry, allowing them to focus more on patient care. These adaptations often include Customisable templates for common tasks, voice recognition for notes, and integration with other medical devices to automatically import data, reflecting the needs of healthcare professionals for efficient and accurate patient record management.
2. **Education**: Learning Management Systems (LMS) have been adapted to provide personalised learning experiences. For instance, adaptive learning algorithms adjust the content difficulty based on a student's performance and learning pace. Additionally, these systems often include features like gamification, which aligns with the preferences of younger users who are more engaged by interactive and rewarding educational experiences.
3. **Finance**: Financial analysis software has been adapted to include domain-specific features such as real-time data feeds, advanced charting tools, and predictive analytics. These features cater to the needs of financial analysts who require up-to-date information and sophisticated tools to make informed decisions quickly. The user interfaces of these applications are often highly

Customisable, allowing users to create a workspace that aligns with their analysis processes and preferences.

4. **Retail**: Point of Sale (POS) systems have been adapted to include customer-facing displays that allow customers to see the transaction in real-time, select their preferred payment method, and even complete the purchase themselves if they prefer a self-checkout experience. This adaptation reflects the changing behaviours of retail customers who value convenience and control over their shopping experience.
5. **Transportation**: Navigation apps have been adapted to provide features that cater to the needs of professional drivers, such as truck-specific routing that accounts for vehicle size and weight restrictions, and real-time traffic updates that help avoid delays. These adaptations are in response to the preferences of transportation professionals who require reliable and efficient route planning tools.
6. **Manufacturing**: Industrial IoT (Internet of Things) platforms have been adapted to provide user interfaces that are accessible and operable in noisy, dirty, or hands-free environments. This includes the use of voice commands, gesture controls, and ruggedised touch screens, which align with the behaviours and needs of factory workers who may not be able to interact with devices in traditional ways.

In each of these examples, the adaptations are a direct result of understanding the unique characteristics and expectations of users within a specific ICT domain and designing technology that enhances their productivity, efficiency, and overall user experience.

This activity is complete when you have

- Engaged with the AI tutor in the StudyShere case study and participated in class discussion to share your experiences and learn from others.
- Documented your analysis and recommendations for the StudyShere case study in a short report (1-2 pages, or a copy of the chat transcript), which will form part of your **portfolio (https://lms.griffith.edu.au/courses/24045/pages/building-a-portfolio-for-assignment-2)** .
- Applied the concepts of Activities 4.1 and 4.2 to your **application system design report (https://lms.griffith.edu.au/courses/24045/assignments/93487)** .