Lancaster University

SCC 141: Professionalism in Practice

Week 2 – Systems development lifecycle

16th October 2024

Dr Mo El-Haj, Senior Lecturer, School of Computing and Communications

About me…

Lancaster University

My research:

- Natural Language Processing
- Low Resource Language
- LLMs and Language Resources

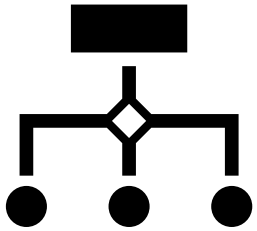
Contact me: m.el-haj@lancaster.ac.uk

Lancaster University

Part 1 – System development lifecycle

Learning objectives

- To **identify** the different stages of the systems development lifecycle – and what each phase involves



- To **compare** the pros and cons of different approaches

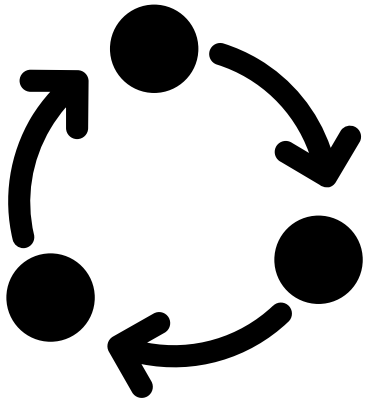
Why are we starting here?

- Starting point for understanding wider context of any development work
- Understanding where and when key decisions are made

What is a system?

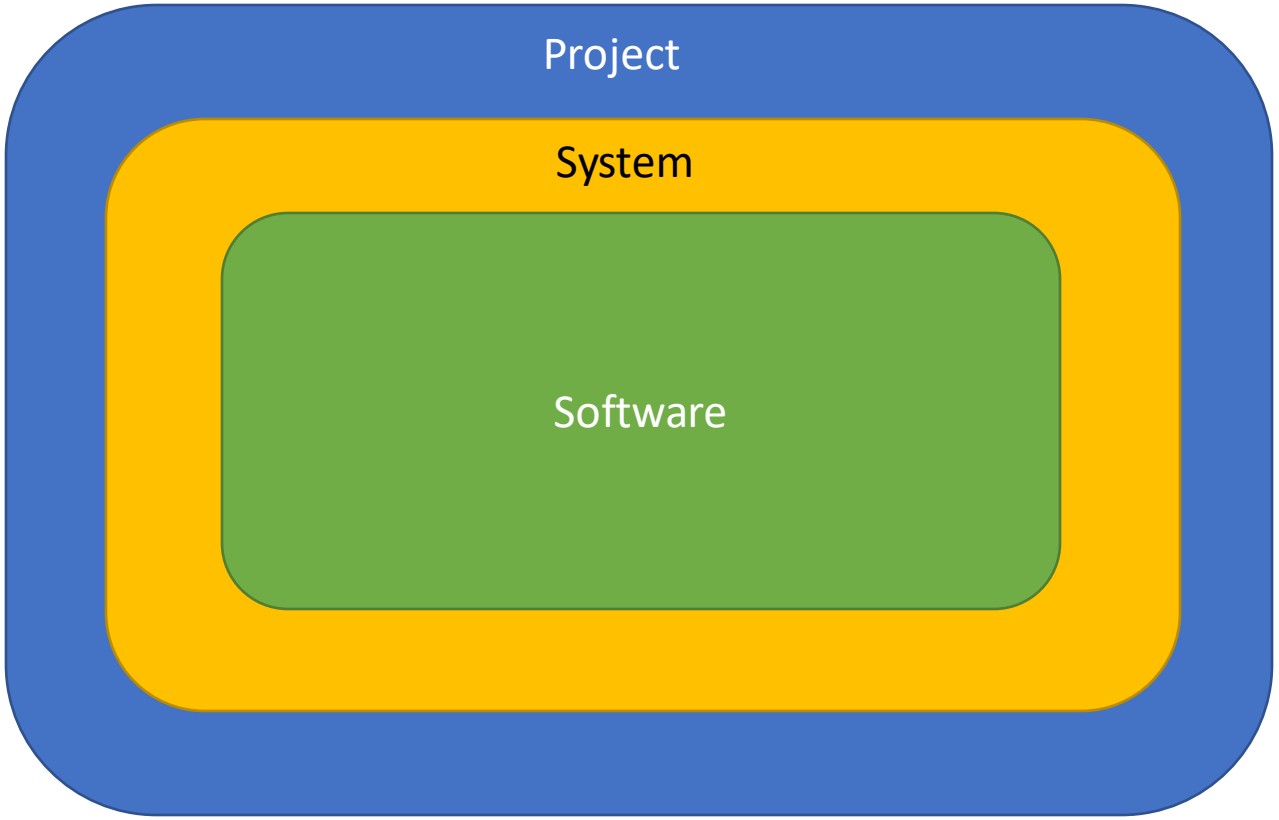
Lancaster University

- A collection of **interrelated parts** that form a whole
- The collection has some **purpose**
- A change in any part can lead to a change in other part(s)

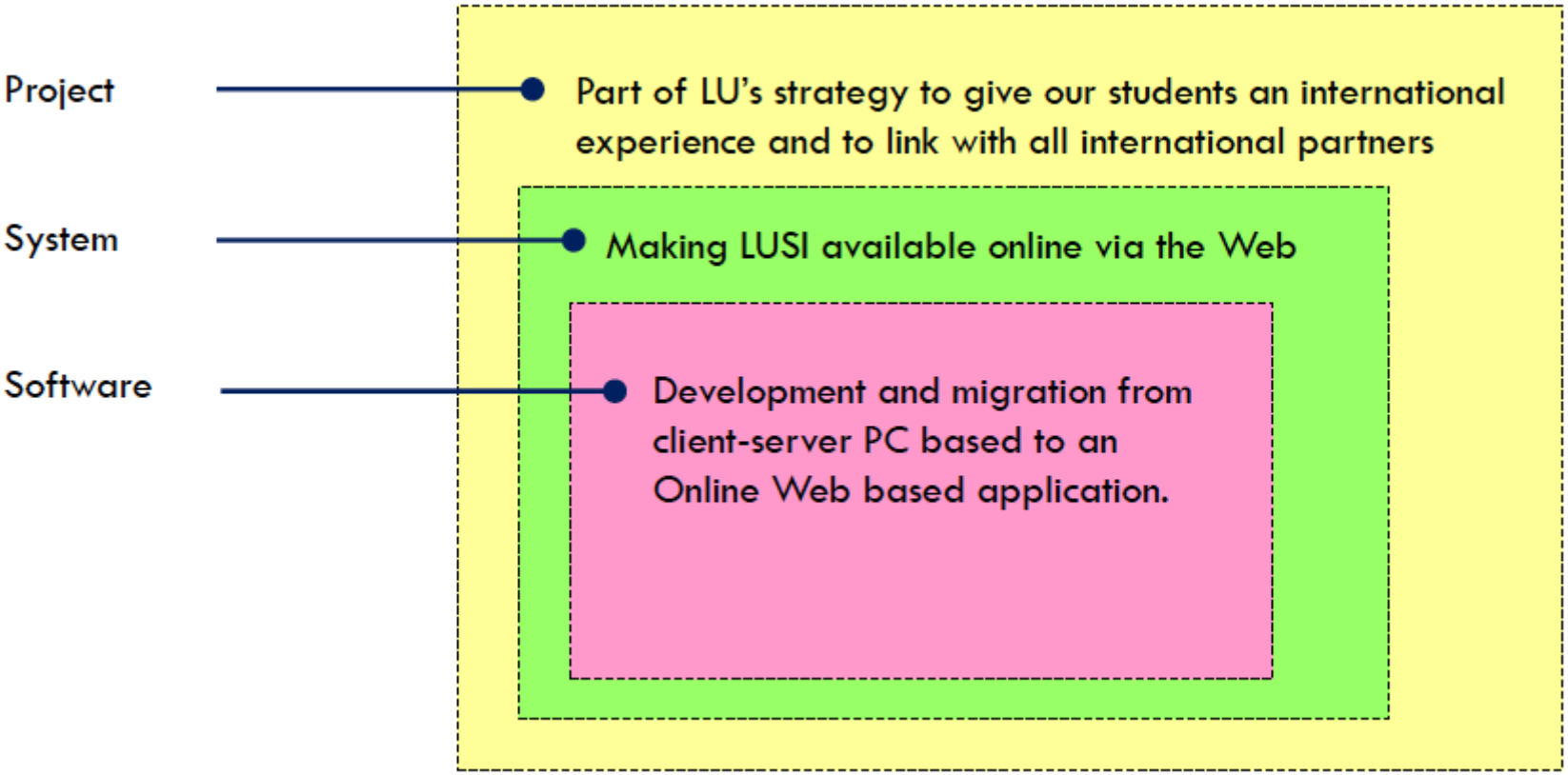


Software, system, project

Lancaster University



Example – LUSI (**L**ancaster **U**niversity's **S**tudent **I**nformation record system)



Systems development lifecycle

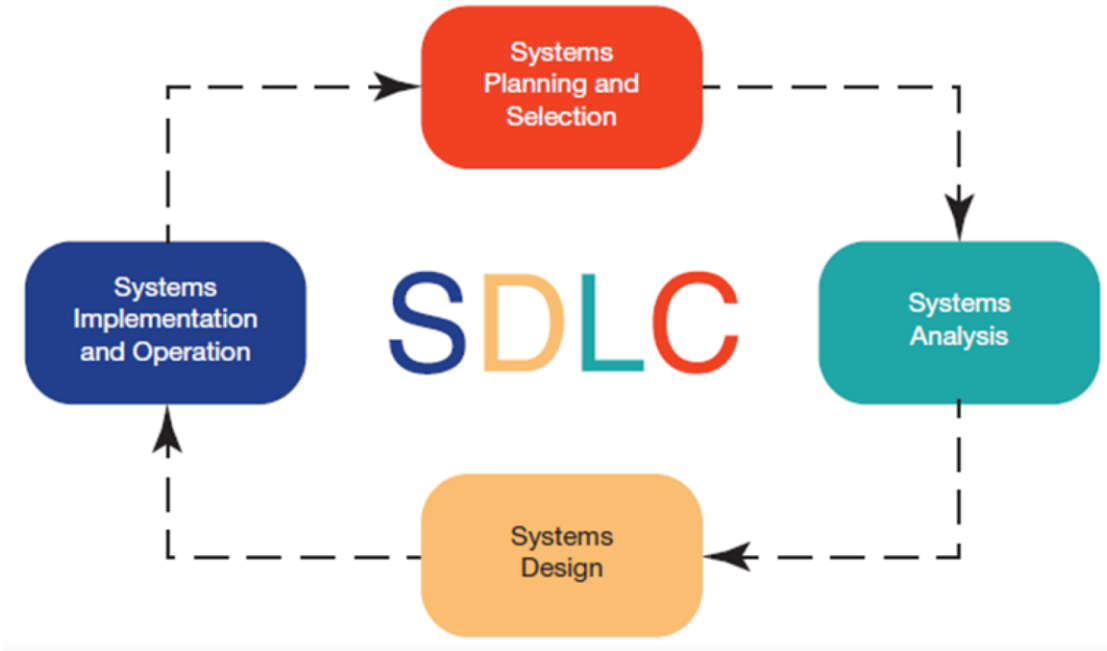
Lancaster University

Systems development lifecycle = **'the process of determining how a system can support business needs, designing the system, building it, and delivering it to users'**

- Dennis et al. (2012) *System Analysis and Design.* Wiley.

Systems development lifecycle stages

1) Planning
2) Analysis
3) Design
4) Implementation



Systems development lifecycle stages

Lancaster University

1) **Planning**
2) Analysis
3) Design
4) Implementation

Stage 1 - Planning

- Understanding why a system **should** be developed and creating plan for **how** it will be developed and delivered
- Often involves a feasibility analysis – technical feasibility, economic feasibility and organizational feasibility
- Working out the resources required and capacity to deliver; making a business case

Stage 1 - Planning

Lancaster University

Cost-benefit analysis:

- Part of stage 1 economic feasibility
  - Development costs (one-time costs)
  - Operational costs (ongoing costs)
  - Tangible benefits (e.g., revenue)
  - Intangible benefits (predicted benefits, that may be harder to quantify)

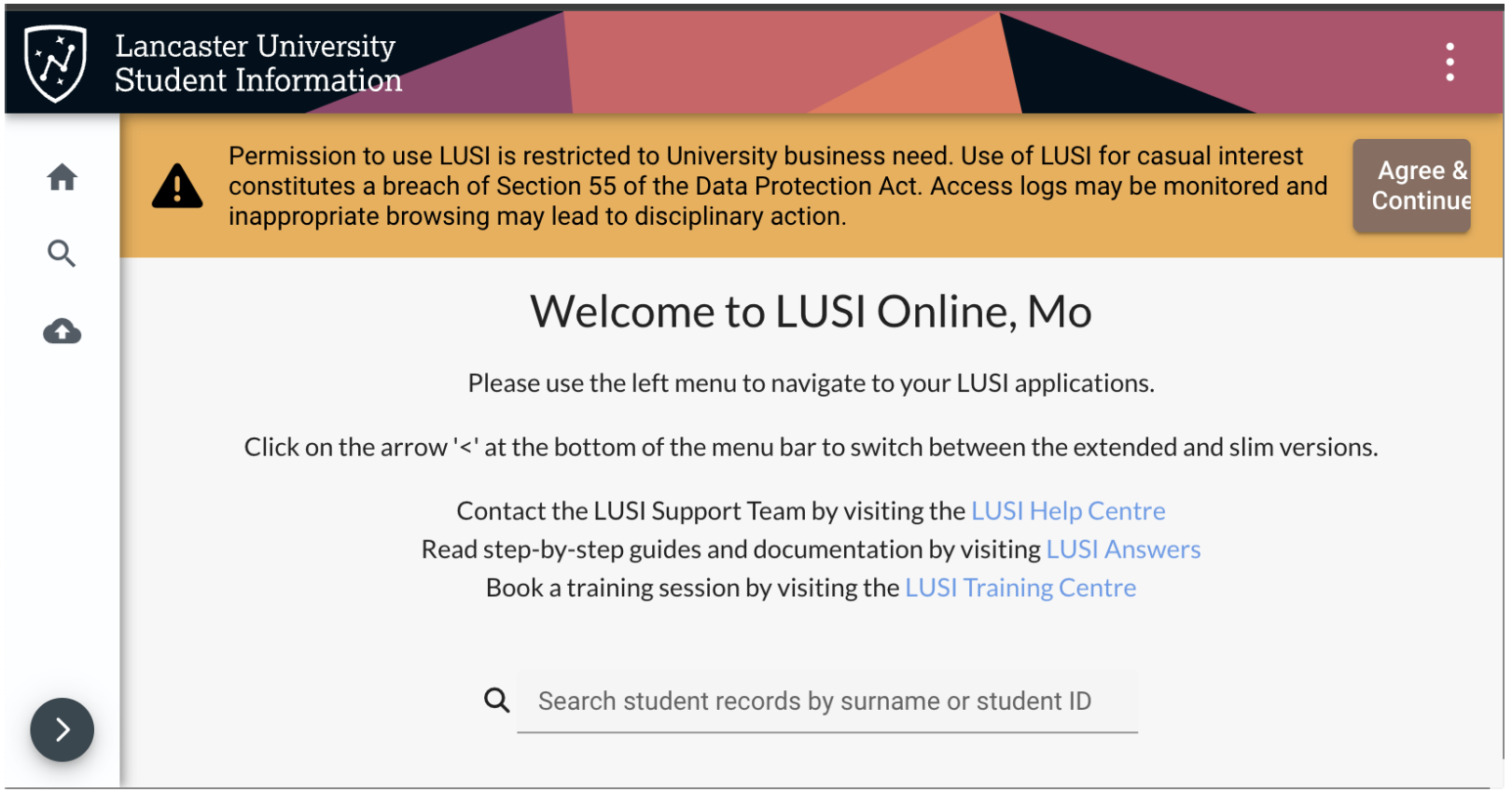
Stage 1 – Planning

Lancaster University

Outputs:

- Goals for new system
- Definition of project's scope
- Assessment of feasibility
- Initial work plan

Stage 1 – Planning: LUSI example



Stage 1 – Planning: LUSI example

Lancaster University

Vision:

- “To enhance and enrich the experience of all Lancaster students through every aspect of the student lifecycle by providing staff and students with efficient and effective digital processes and information to underpin and support the delivery of globally significant research and teaching”

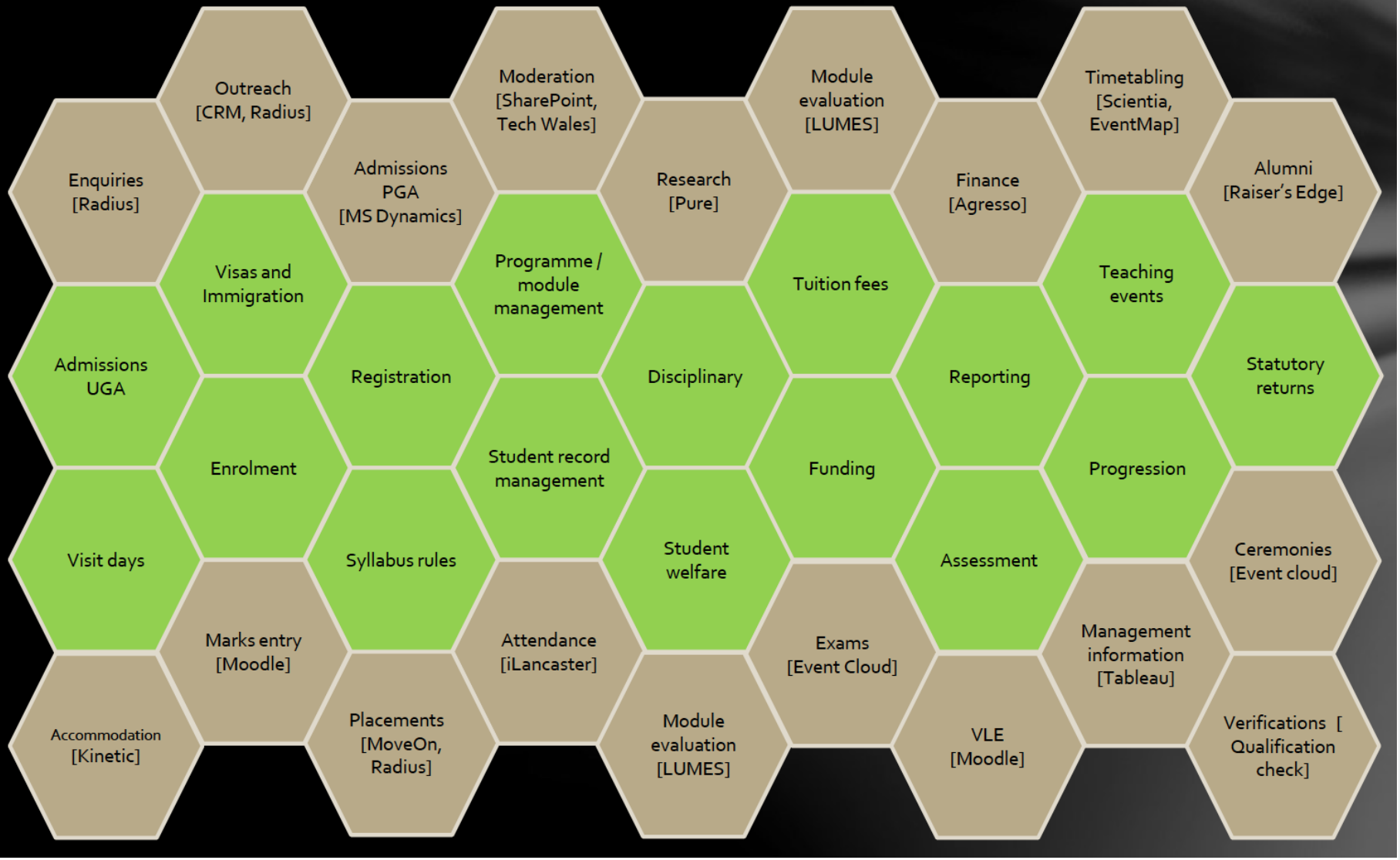
Stage 1 – Planning: LUSI example

Lancaster University

Business case:

- Enhancing student experience
- Streamlining administrative processes
- Moving away from paper-based processes
- Expand provision of LUSI to overseas campuses

LUSI core processes and integrations



Systems development lifecycle stages

Lancaster University

1) Planning
2) **Analysis**
3) Design
4) Implementation

Stage 2 – Analysis

Lancaster University

Understanding **who** will use the system, **what** the system will do, and **where/when** it will be used

1. **Understand** the **existing** situation
2. **Identify improvements**
3. **Define requirements**

We'll talk more about requirements later in week 3.

Stage 2 – Analysis

Lancaster University

1) Analysis: problem analysis

2) Requirements determination: what the system should do and what characteristics it should have

3) Requirements gathering: gathering data from users

Stage 2 – Analysis: problem analysis

- Breaking a whole into its parts to understand the parts' functions and inter-relationships
- May involve rich pictures; images that attempt to capture everything that is relevant to a complex situation or system

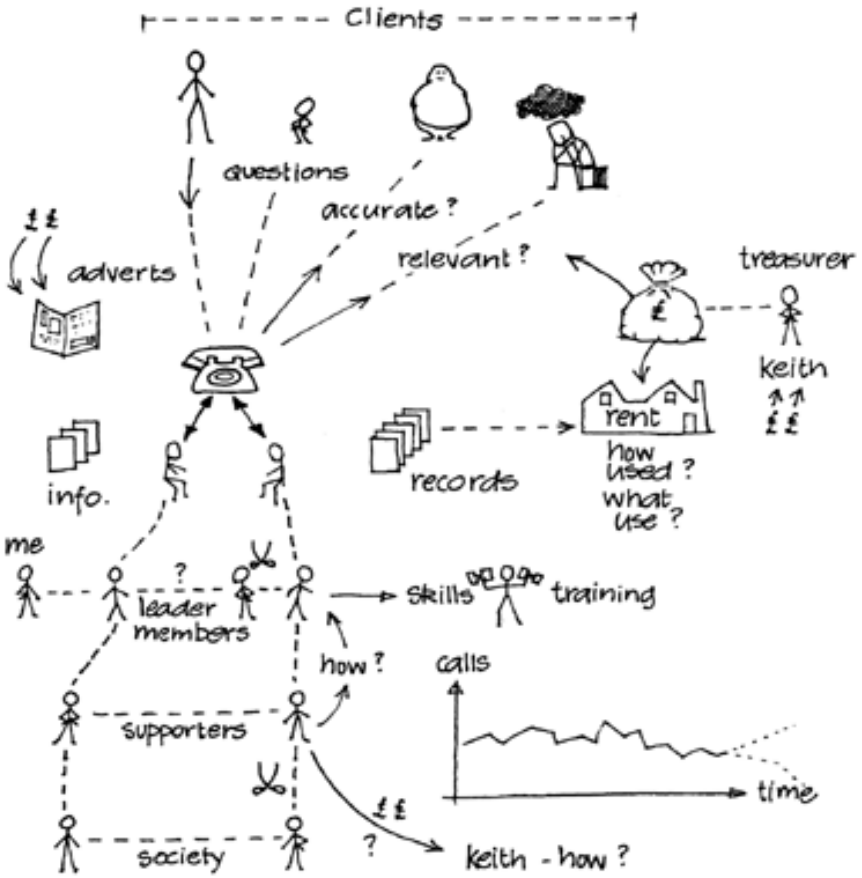


Image source: http://systems.open.ac.uk/materials/T552/pages/rich/richAppendix.html

Stage 2 – Analysis: why are rich pictures helpful?

- Mental tool to understand a scenario
- Useful for discussing with other people – can cut across jargon, etc.
- Helpful for identifying stakeholders
- Suitable for any domain

Stage 2 - Analysis: requirements determination

Lancaster University

- Requirements determination changes high level strategic objectives into more precise statements of what a system should do
- Requirements are simply statements of **what the system should do and characteristics the system should have**

Stage 2 – Analysis: Requirements gathering

Lancaster University

- Getting more insight into the requirements for the system by gathering data from users

Stage 2– Analysis: **PACT** questions

Lancaster University

- **P**eople – what are the intended users' characteristics and skills?
- **A**ctivities – how is the activity currently carried out? Why? What can be improved?
- **C**ontext – what is the environment of the activity?
- **T**echnology – what tools are being used currently? How might new developments be used?

Stage 2– Analysis: PACT considerations

Lancaster University

People:

- Language
- Levels of skill and expertise
- Cognitive characteristics – attention, perception, memory
- Physical characteristics – physical abilities, accessibility
- Emotions – satisfaction, frustration, things being aesthetically pleasing
- Infrequent vs. frequent users

Stage 2– Analysis: PACT considerations

Lancaster University

Activities:

- Goals, tasks and actions
- Frequent or infrequent tasks?
- Well-defined or vague goals
- Continuous or interrupted?
- Individual or cooperative?
- Time requirements (e.g., how fast a response is needed)
- Error tolerance

Stage 2– Analysis: PACT considerations

Lancaster University

Context:

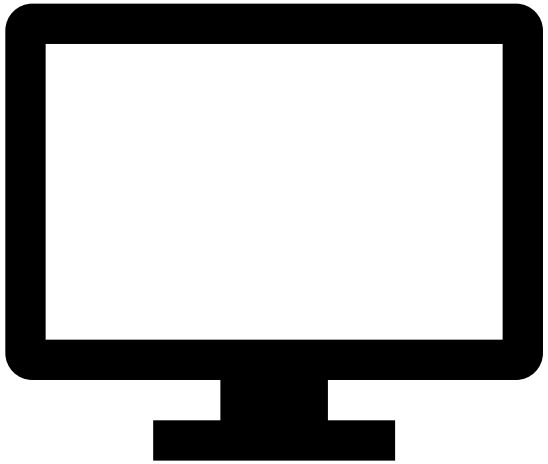
- Physical environment
- Social environment
- Organisational context
- When and where activities happen

Stage 2– Analysis: PACT considerations

Lancaster University

Technology:

- Input – data, commands
- Output
- Communications (between people, between devices, speed, etc.)
- Size of screen
- Networked?



Stage 2 – Analysis: PACT – example (Library)

People: users will be students and staff. Likely to want something fast and convenient. Should be easy to use for non-frequent users

Context: library may be busy at times; should be quick process

Technology: database should be automatically updated to show when a book is taken out or returned

Activities: taking books out, returning books, paying fines

Systems development lifecycle stages

Lancaster University

1) Planning
2) Analysis
3) **Design**
4) Implementation

Stage 3 - Design

Lancaster University

"System design is the determination of the overall system architecture [...] that will satisfy the system's essential requirements"

- Dennis et al. (2012) *System Analysis and Design.* Wiley.

Stage 3 - Design

Lancaster University

- **How** will the system operate; deciding **how** to build it
- Involves design of architecture and interface, development of database and file specifications
- Output: system specification

We'll talk more about user-centered design principles and methods in Week 4.

Stage 3 – Design: interface design

Lancaster University

- User interface design defines how users will interact with the system and what kind of inputs and outputs the system accepts and produces

➢ **Navigation mechanisms**: how user gives instructions to the system (e.g., buttons, menus)

➢ **Input mechanisms:** how the system captures information (e.g., forms)

➢ **Output mechanism:** how the system provides information to the user (e.g., reports)

Systems development lifecycle stages

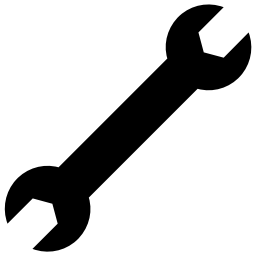
Lancaster University

1) Planning
2) **Analysis**
3) Design
4) **Implementation**

Stage 4 – Implementation

Lancaster University

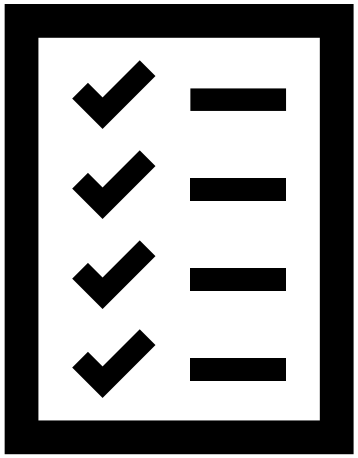
- Actually building the system
- Testing the system
- System construction, installation and support plan (maintainability)
- Sometimes maintenance is identified as a separate phase



Stage 4 – Implementation: testing

Lancaster University

1) Unit testing – testing of each unit or program module separately
2) Integration testing – checking that things that should work together do so without error
3) Acceptance testing – does the system meet requirements
4) User testing - system tested with users



Stage 4 – Maintenance

- Not just about maintaining existing functionality; may often involve building new functionality
- LUSI: new releases approximately every two weeks
- LUSI new functions in July 2025:

→More complex searches and saved searches

→LUSI Ideas Wall, where staff users suggest improvements

Lancaster University

Part 2: Different models for the system development lifecycle

Waterfall development

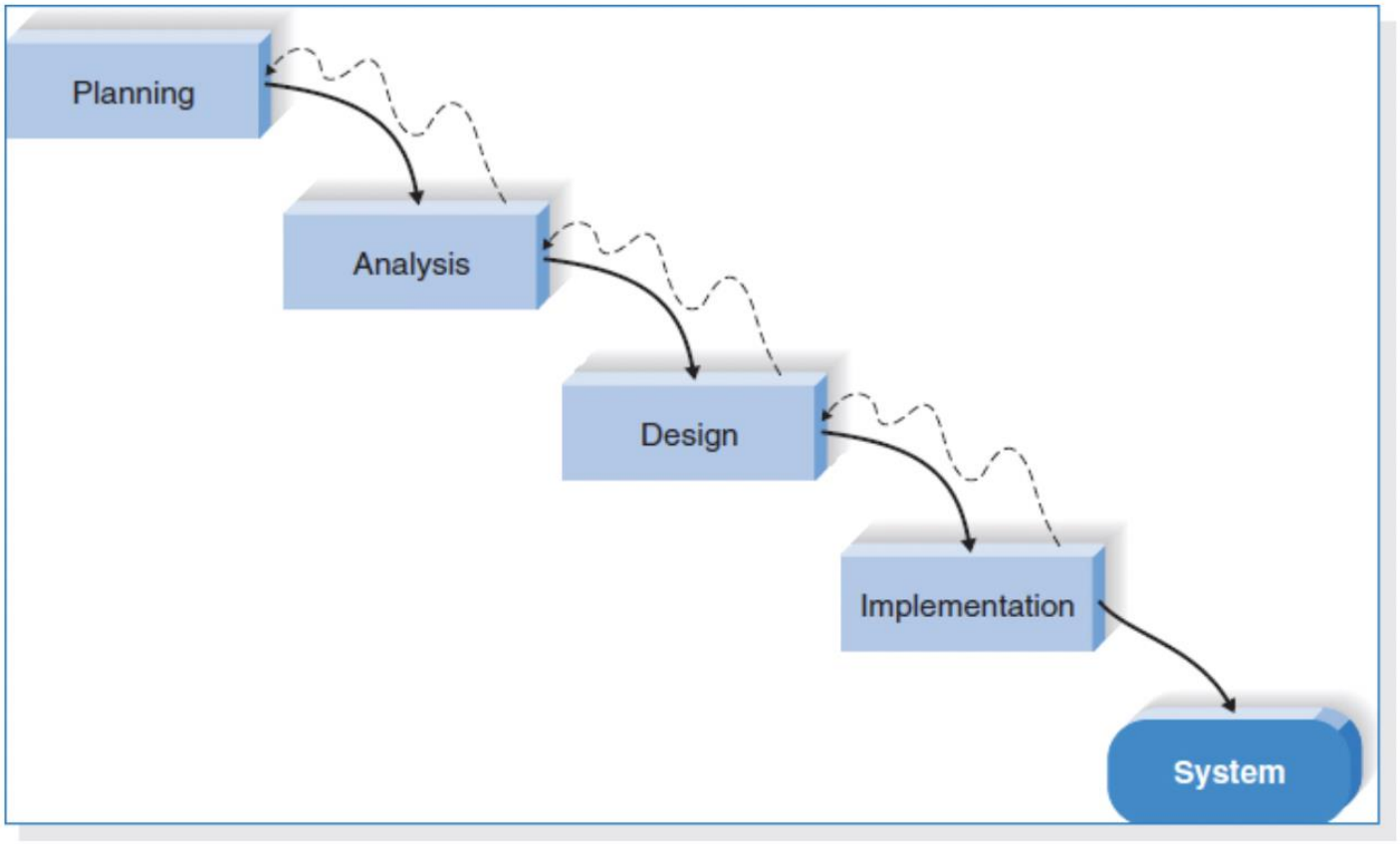


Figure taken from A. Dennis, R. M. Roth and B. H. Wixom (2012) *Systems analysis and design,* Wiley.

Waterfall pros and cons

Lancaster University

Advantages

- Requirements identified at the start; limited changes to requirements as project goes on
- Well suited to systems that have high security needs
- Clear deliverables
- Easy to arrange tasks as thing progress one phase at a time

Disadvantages

- Time consuming
- Inflexible/not dynamic
- No working software until late
- Doesn’t adjust to changing requirements
- Difficult to measure progress within stages
- High overheads

Rapid application development

Lancaster University

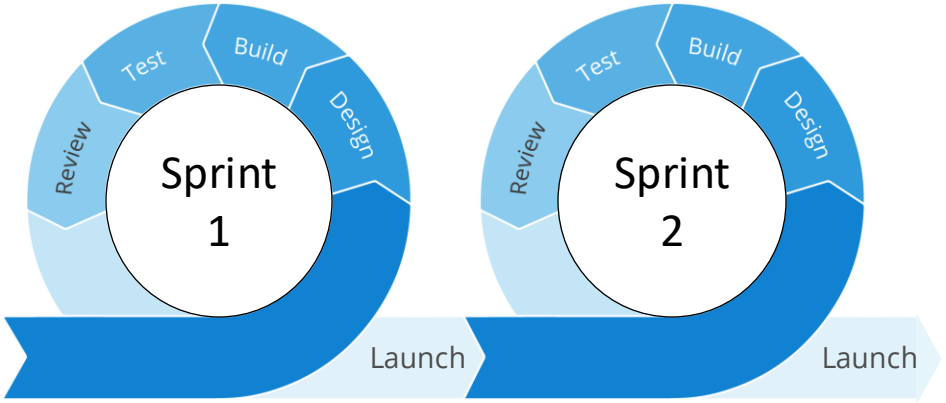
- An alternative approach to waterfall

Examples include:

- Iterative development
- System prototyping
- Throwaway prototyping
- Agile – e.g., XP, Scrum

Agile approach

- Feature oriented not activity oriented
- Rapid development and delivery
- Work in small iterations
- Deliver in each iteration
- Review and adapt
- Make changes



Rapid application development pros and cons

Lancaster University

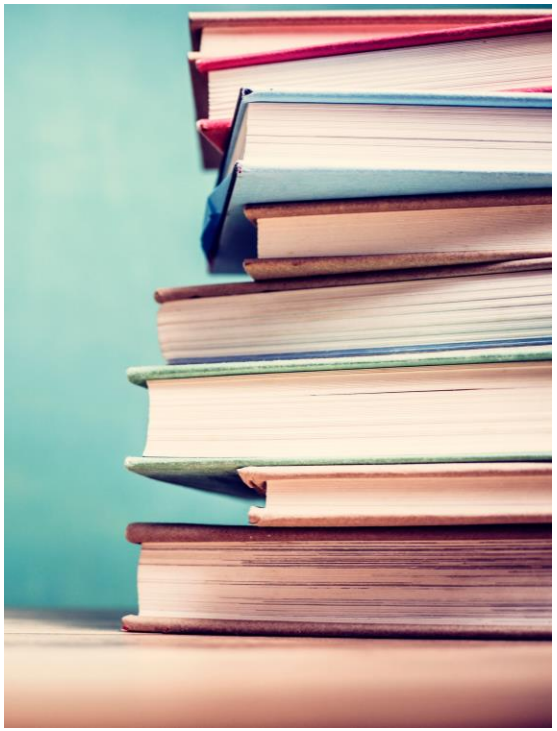
Advantages

- Cheaper and easier to make changes during process
- Flexible
- Requirements can change and be more adaptive
- Often useful when users struggle to articulate requirements
- Get user feedback earlier
- Quicker delivery of working software

Disadvantages

- Can be more challenging to integrate at the end
- Planning can be difficult
- Can be harder to manage
- Less control
- Can be difficult to scale for large systems
- Less documentation

Some recommended reading



- A. Dennis, R. M. Roth and B. H. Wixom (2012) *Systems analysis and design,* Wiley.
- Y. Rogers, H. Sharp and J. Preece (2011) *Interaction Design: beyond human-computer interaction,* Wiley.
- J. Nielsen (1993) *Usability engineering.* Academic Press.
- I. Somerville (2016) *Software engineering*. Pearson

Lancaster University

Thank you! Any questions?