# Giới thiệu chung về Phương pháp nghiên cứu khoa học

2024

# Introduction (Giới thiệu)

- **Aims**: (Mục đích)
  - To introduce academic projects (đề tài nghiên cứu). The course will take the *computing* for demonstration through out the course.
- **Learning objectives:** (Mục tiêu)
  - Understand what projects are.
  - Understand the different types of academic projects in computing and information sciences.
  - Understand different degree structures and project requirements.
  - Describe the roles different people have in academic projects.

# Introduction

- Projects are a major component of virtually all undergraduate and postgraduate computing and information science courses within universities.
- They require students to draw on a number of separate but highly important skills; surveying literature, report writing, developing and documenting software, presentational skills, time management, project management skills and so on

# Introduction

- Pursuing a project within academia is not the same as performing a project within industry.
  - Students will be expected to look at things much more critically and more deeply than they would elsewhere.
- Academic projects should provide evidence of a much deeper understanding of what you are doing.
  - They require some form of justification and contextualisation.
  - You are not expected to do merely what you are told to do; you are expected to develop your own thoughts, arguments, ideas, and concepts

# Introduction

- Importantly, as a degree student you are expected to **think.**
  - This ‘deeper’ understanding of situations, problems, and events is supported by your *research* skills – skills that are vitally important within academic projects.
- Academic projects are usually a critical component of your degree course.
  - Sometimes they make up a significant component of your final year, and sometimes, particularly at postgraduate level, they may represent *all* of your degree

# Why projects required?

- **Assessment across a number of disciplines simultaneously**.
  - Your project will require you to apply things you have learnt from lots of different areas of your course – both technical and personal skills.

# Why projects required?

- **Allows you to develop new skills**.
  - The project will also enable you to develop skills you might not have covered explicitly in your course so far.
  - These new skills might be technical (*learning a new programming language, development method, design technique, research, etc.*) and personal (*time management, discipline, communication skills, report writing, etc.*).

# Why projects required?

- **Work independently**.
  - Your project might be the first time that you have had to work mainly on your own on a project that is primarily your own work, ideas, and responsibility.

# Why projects required?

- **Make a contribution**.
  - A project will allow you to make some form of contribution.
  - Previously you might have been doing directed coursework, examinations, etc.
  - The project will allow you to produce something that may be used by or benefit others

# What are projects

- In a broad sense: Projects can be defined as ‘something which has a beginning and an end’.

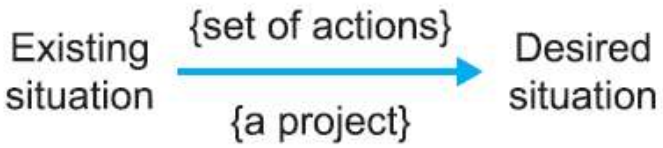


**Figure 1.1** The Meliorist Model

# What are projects

- Computing projects come in all different shapes and sizes, as the field they are drawn from is immense.
  - It is more widely recognised, within academic institutions, that computing projects need to do more than develop a piece of software.
- the nature of CS and IS means that projects are drawn from both the ‘hard’ sciences (natural science) and the ‘soft’ sciences (social science).
  - => projects cover a vast range of topics, from highly technical software development projects to (equally difficult) case studies within information science

# What are projects

- In terms of university degree courses, you will *probably* find that courses entitled ‘Computer Science’ or ‘Artificial Intelligence’ tend to fall more towards the left-hand side of this scale.
- ‘Software Engineering’, ‘Computing’, and ‘E-business’ courses probably fall more towards the centre (with ‘Software Engineering’ to the left and ‘E-business’ to the right).
- Courses entitled ‘Information Science’, ‘Information Technology’, ‘Business and Information Technology’, ‘Business Information Systems’ and ‘Information Systems’ will fall more towards the right-hand side.
- Some courses might fall anywhere along the scale depending on their content – for example, ‘Multimedia’ and ‘Computer Studies’ can mean different things in different institutions

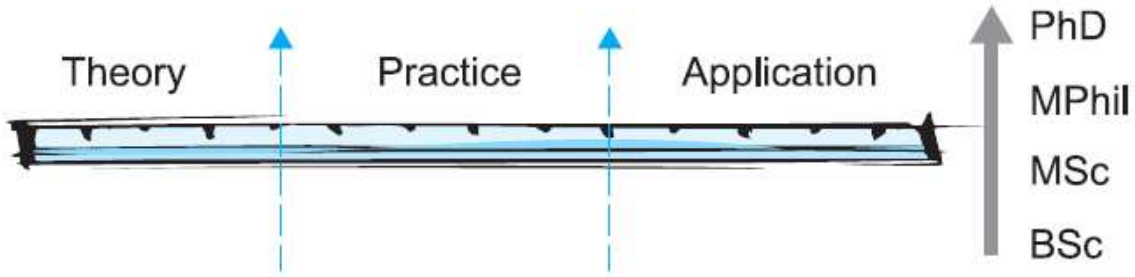


**Figure 1.2** The landscape of computing (adapted from Dawson, 2004)

# Project types

- Research based
- Development
- Evaluation
- Industry based
- Problem solving

# Project types: Research based

- A research based project involves a thorough investigation of a particular area;
  - improving your understanding of that area, identifying strengths and weaknesses within the field, discussing how the field has evolved, and acknowledging areas suitable for further development and investigation.
- This kind of project will involve some form of literature search and review, and would be suitable for taught bachelor's or taught master's courses.
- A research-based project may well have to do more than establish the field of study.
  - For example, having established the field (backward looking), a student in a doctoral program (a PhD, for example) would then be expected to *contribute* to that field (forward looking)

# Development

- This category includes the development of, not only software and hardware systems, but also of process models, methods, algorithms, theories, designs, requirement specifications, and other interim documents.
- For some developments (notably software) you will be required to include requirements documentation, designs, analyses, and fully documented test results along with user manuals or guides

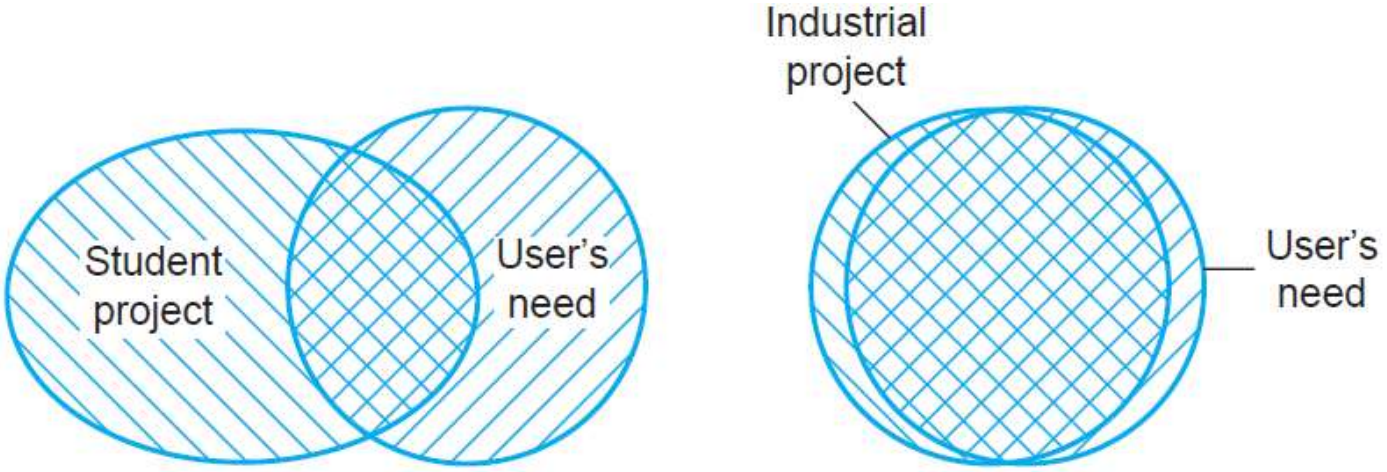


**Figure 1.3** Comparison of student development project and industrial development project

# Evaluation

- This category encompasses all projects that involve some form of evaluation as their main focus.
- For example,
  - comparing several approaches to a particular problem; evaluating two or more programming languages (applied in different contexts or to different problems);
  - analyzing an implementation process within a particular industry; assessing different user interfaces; analyzing a particular concept;
  - considering alternative and new technological approaches to a problem; appraising development methodologies to a problem; and so on.

# Industry based

- An industry-based project involves solving a problem within either an organisation or another university department.
  - The difference in this case is that you undertake the project for an actual client, which carries with it a number of benefits as well as drawbacks.
- The sponsor does not ‘hijack’ the project – that is, force it into a direction the company wishes it to go, regardless of whether it is suitable for your academic work or your course

# Problem solving

- A problem-solving project can involve developing a new technique to solve a problem, improving the efficiency of existing approaches or an evaluation of different approaches or theories in different situations.
- It might also involve applying an existing problem-solving technique or theory to a new area.
- In these cases, some form of evaluation would be expected:
  - for example, did your new approach work well or did you discover reasons why it was unsuitable for problems of this nature? Why does one approach or theory work better in some situations than in others?

# Programming in computing projects

- Although you are on a computing course of one kind or another, you may or may not be expected to write a program.
- Sometimes programming is the main emphasis of your project, for example, if you are on a software engineering course.
- At other times, you may need to write a program as a ‘vehicle’ for testing and demonstrating one thing or another, for instance, to test out some ideas, demonstrate a technique or algorithm, or evaluate some human computer interaction concepts.
- Whatever the case, as a computing student you will naturally be expected to produce code that is of acceptable *quality*

# Degree requirements

- Educational institutions throughout the world have many different degree structures that take students from school/college level education through undergraduate study to doctoral degrees (sometimes referred to as *higher education*).
- Where necessary, a differentiation will be made between *taught* degrees (such as bachelor's and master's degrees – BScs and MScs for example) – which are predominantly based around taught courses that include some project element – and *research* degrees (such as MPhils and PhDs), which are primarily research-based projects that lead to some contribution to knowledge

# Degree requirements

- A project undertaken as part of a taught course (bachelor's or master's degree) does not need to make great breakthroughs in science.
  - It might involve the development of some software, it could be an extended case study, or it might be research-based.
- This is not the case for research degrees, however.
  - An MPhil should produce work that could be published (perhaps with some reworking), whereas students pursuing a PhD will probably be expected to have published some of their work before completing the final thesis

# Degree requirements

- An ability to work independently with minimum supervision;
- An ability to draw on existing knowledge and identify additional knowledge needed for your study;*
- An ability to critically evaluate advanced literature (journal papers);*
- An ability to conceive original ideas';
- An ability to plan your work effectively;*
- An ability to select and use appropriate hardware, software, tools, methods and techniques;*
- An ability to present your work effectively in written and oral forms;*
- An ability to critically evaluate your own work and justify all aspects of it;*
- An ability to identify areas of further research in your chosen area.

# Stakeholders

- Stakeholders are any individuals who are involved with your project. The most important person in your project is ***you.***
  - You are responsible for the overall completion of your project, meeting milestones, achieving objectives, satisfying users, satisfying the examiners, and so forth.
- Your project has many other stakeholders, and they all have significant roles to play.
  - They include your *supervisor(s)*, your *user(s)*, your *client(s)*, your *examiner(s)* and any software *testers/evaluators* you might employ as part of your system evaluation (if you have one in your project)

# Summary

- The field of computing ranges from ‘hard’ theoretical computer science, through practical software implementation, to ‘softer’ areas of information systems concerned with the use and the effect of IT.
- Computing projects tend to fall into one of the following five categories: *researchbased, developments, evaluation, industry-based* or *problem solving.*
- Your project will have a number of *stakeholders,* the most important of which is **you.** Others include your supervisor(s), client(s), user(s), examiner(s), software testers and evaluators.

# Questions