Computer Science
Information Technology
Project

Also Specialist Modules

Lecture 2
AIMS

This session aims to get you to think more about:

- What makes a good project
- What makes a good set of objectives
- How to extend a basic idea
Slides available on-line

The slides from this lecture are available on line

Choose Page Scaling: Multiple pages per sheet
Note

• Project as a series of increasingly challenging objectives
• Process and Assessment
• Don't look for an easy project
• Need to Reflect
• It is a Computer Science or Information Technology project, not a Business, not Engineering, not....
Choosing a Project

• Free choice of topic and the issues you set out to address (within reason)
• Suggestions available (but limited number)
• Even if you choose a project from a list, you define the advanced objectives for your project and how you are going to meet them
• So you set the objectives for your project, which is DANGEROUS
• Don’t look for an easy project
Choosing a Project

Your project will be assessed as a CS – IT project

Your project must be a Computer Science or an Information Technology Project depending on the route of your degree.

• As a guide: it must be based on one or more modules run on our programme.

• Do not get confused with some modules that contain material that may look as not computer science.
  - for example Computer Architecture is not Electronics, Information Technology for Business is not Business, etc.

• Sessions on your work that are not CS or IT will not count on your marks and possibly extended use will reduce your mark.
So what makes a good project?

- practical problem to solve / issue to address
- realistic idea for solving the problem / addressing the issue
- understanding when to stop
- a set of milestones for judging progress
- a plan for presenting the results
- understanding of how to evaluate your work and reflect on it
- motivation to carry the project through
- ability to achieve the goals you have set
Success, mediocrity and failure

- good practical work + good written work
  - Good grade – up to 1st class project

- good practical work + poor written work
  - Poor mark. Possibly a bare pass?

- poor practical work + good written work
  - Virtual Project? What is the report about? Possibly Fail – Possible Cheating?

- poor practical work + poor written work
  - Failure
We expect you to

• Set realistic objectives
• Include more ambitious and try some of them
• Work independently
• Foresee potential problems
• React to changing circumstances
• Present results coherently
• Recognize and overcome difficulties
• Identify essentials
• Evaluate your own work
Things that can go wrong

Being:

Vague,
Over-ambitious without been also realistic,
Under-ambitious,
Distracted, ...

Not working hard enough

Remember an average student needs 300 hours of
productive work for an average mark
Vagueness

- Being vague about what you are going to do
- Failing to properly define the problem you intend to address
- Failing to decide precisely what you will produce
Not Planning

• Not planning your practical work before you do it
• Not deciding what you will put in your report until it’s time to write it
• Not thinking about how you will judge the quality of your work according to marking criteria

If you are going to get a 1st class mark, you should have done a project worthy of a 3rd class by the time of Interim Project Report.
Scope

• Setting the scope of the project
  - too wide (over-ambitious) and concentrating on these from the beginning
  - too narrow (under-ambitious)

• Concentrating on learning tools and techniques rather than solving a problem or addressing an issue

• Losing motivation
Not Working

- Not doing any work
  - about 300 hours productive work for an average student to get an average grade
  - only about 10 hours of “class” contact

If you want a guide of time needed, think that of second year (Level 5) modules required 300 hours each for an average student to get an average grade, and that included Lectures, Tutorials, Labs and self study.
Wasting your Life

• Concentrating on the **quantity** of time you are spending rather than the **quality** of the work you are doing

Many students say:

“*I spent 10 hours for that.*”

Very few say:

“*I build that and test and validate it.*”

We may find one or two who evaluate their effort by the **quality** of the product they produce. Most think on **quantity** not on **quality**.
Aims, Project Objectives, System Requirements and Personal Goals

- An aim is a general, overall thing you would like to accomplish.
- An objective is specific, concrete and testable.
- A personal goal is something you want to learn or to achieve. Personal goals do not usually make sensible project objectives.
- System requirements are not project objectives; they can be deliverables.

Based on notes from UH core team (James Malcolm, Steve Hunt, Colin Egan)
Minor Modifications to allow for IST cases, for example deadlines
Objectives

Your project objectives should indicate

- How much work you intend to do;
- What kind of work it will be;
- What standard you expect to achieve;
- What you intend to produce.

Project is a series of increasingly challenging objectives
Tasks & Deliverables – 1

Your project proposal should indicate

- Give us a list of tasks you will perform and the deliverables in order to pass your project.

These are the ones that will be judged to accept your proposal.

When a proposal is accepted, it is accepted that it is good enough for a pass.
Tasks & Deliverables – 2

Your project proposal should:

- Give us a list of tasks you will perform and the deliverables in order to obtain a better mark. Read the marking criteria to help you.
- Try for a 1st class mark, in order to get good mark. I am sure you will all do so.
- If your target is set low, then you will probably fail
How your target affects your mark

Case 1: You aim for a pass (3\textsuperscript{rd} class mark)
- This is 49 at the most
- If you are marked by 80\% that means that you will get
  \(0.8\times49=39\) ie: \textit{fail}

Case 2: You aim for a 2\textsuperscript{nd} upper class mark
- This is 69 at the most
- If you are marked by 57\% that means that you will get
  \(0.57\times69=39\) ie: \textit{fail}

Case 3: You aim for a 1\textsuperscript{st} class mark
- If you are marked by 40\% you will get \(0.4\times100=40\) ie: \textit{Pass}

\textbf{So: Put your target to 100}
How do I develop my initial idea?

Your topic needs to be turned into a sequence of concrete and achievable objectives.

- Use the project suggestions or
- Look at the way the Project Suggestions are formulated to see how a project is broken down into stages and objectives are set out. We can accept a proposal of your own if it is as good as the ones given by us.
- Search on the WEB

Based on notes from UH core team (James Malcolm, Steve Hunt, Colin Egan)

Minor Modifications to allow for IST cases, for example deadlines
Setting your own objectives

Your objectives should be SMART:

- Simple,
- Meaningful,
- Achievable,
- Relevant, and
- Testable

At level 3 you need to be able to cope with conflicting information, evaluate alternatives
Simple

Simple: Each objective identifies a single, simply stated, goal, even though the project as a whole will not be simple.
Meaningful

Meaningful: Objectives should be written in language that others find easy to understand, using technical terminology only where absolutely necessary, avoiding jargon, slang and acronyms.
Achievable

Achievable: It should be possible for you to achieve the goal in the time available and with the resources at your disposal. But remember that an investigative project can set up a hypothesis to test; if it turns out to be false, you have still achieved an objective of testing the hypothesis.
Relevant

Relevant: The goal should be directly related to your project; not be a personal goal. Thus, to "learn C++" might be necessary, but it is not an objective.
Testable

Testable: It should be possible to say whether or not it has been achieved, for example: "I will create a data dictionary".
Objectives exercise:

A number of statements. Are these objectives? Some. You may not be sure about

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Read a book about how to design databases (normalisation, etc).</td>
</tr>
<tr>
<td>2.</td>
<td>Go to the Library and spend time looking for relevant papers.</td>
</tr>
<tr>
<td>3.</td>
<td>Read the paper &quot;How java programs interact with virtual machines at the micro-architectural level&quot;.</td>
</tr>
<tr>
<td>4.</td>
<td>Download the latest Java ME developer tools from Sun, including the Wireless Toolkit.</td>
</tr>
<tr>
<td>5.</td>
<td>Install Dreamweaver, and get it talking to mySQL.</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>6.</td>
<td>Talk to a friend who is familiar with the requirements for systems like that being done for the project.</td>
</tr>
<tr>
<td>7.</td>
<td>Go to the Library and make a list of relevant papers.</td>
</tr>
<tr>
<td>8.</td>
<td>Learn how to use Dreamweaver, and use it to create a &quot;Home Page&quot; for the site.</td>
</tr>
<tr>
<td>9.</td>
<td>To increase my knowledge of Web/WAP programming.</td>
</tr>
<tr>
<td>10.</td>
<td>System should be scalable in the event of pursuing the project in the real world.</td>
</tr>
</tbody>
</table>
1. Look at the statements on your sheet, and mark them "+" for **objective** or "-" for **not** (as appropriate).

2. If you are not sure, put "?"

3. Find 2 others and find out whether you agree. Discuss the disagreements!

4. When I bring the discussion to a close, mark any statements where the discussion caused you to change your mind!!
Where were the disagreements?

1. Read a book about how to design databases (normalisation, etc).
2. Go to the library and spend time looking for relevant papers.
3. Read the paper "How java programs interact with virtual machines at the microarchitectural level".
4. Download the latest Java ME developer tools from Sun, including the Wireless Toolkit.
5. Install Dreamweaver, and get it talking to mySQL.
6. Talk to a friend who is familiar with the requirements for systems like that being done for the project.
7. Go to the Library and make a list of relevant papers.
8. Learn how to use Dreamweaver, and use it to create a "Home Page" for the site.
9. To increase my knowledge of Web/WAP programming.
10. System should be scalable in the event of pursuing the project in the real world.

Based on notes from UH core team (James Malcolm, Steve Hunt, Colin Egan)
Minor Modifications to allow for IST cases, for example deadlines
Now turn these into "proper" objectives

Write down what you think these objectives should say.

Next you will find some changes I propose. Many can disagree. Please use the allocated project time slot (see module guide and project time table) to discuss your disagreement.
My Answers!

1. Read a book about how to design databases (normalisation, etc).

becomes

Design and implement a fully normalised database for the required system.
My Answers!

2. Go to the Library and spend time looking for relevant papers.

becomes

Produce an annotated bibliography of papers about database design.
My Answers!

3. Read the paper "How Java programs interact with virtual machines at the micro-architectural level".

becomes

Compare the results obtained with those described in the paper "How Java programs interact with virtual machines at the micro-architectural level".

Based on notes from UH core team (James Malcolm, Steve Hunt, Colin Egan)
Minor Modifications to allow for IST cases, for example deadlines
My Answers!

4. Download the latest Java ME developer tools from Sun, including the Wireless Toolkit.

becomes

Install the Java ME developer tools and Wireless Toolkit, and write and test a "hello world" program.
My Answers!

5. Install Dreamweaver, and get it talking to mySQL.

becomes

Implement a simple page to list all the products in the database.
My Answers!

6. Talk to a friend familiar with the requirements for systems like that being done for the project.

becomes

Create a first draft requirements catalogue.

or (much better)

Design and implement a database to support the requirements identified, and enter suitable test data to explore these requirements.
My Answers!

7. Go to the Library and make a list of relevant papers.

*is objective, but even better would be*

Produce an annotated bibliography of papers about database design.
My Answers!

8. Learn how to use Dreamweaver, and use it to create a "Home Page" for the site.

*is objective*
My Answers!

9. To increase my knowledge of Web/WAP programming.

becomes

Produce a WAP front-end to the system
My Answers!

10. System should be scalable in the event of pursuing the project in the real world.

becomes

Measure the performance of the system as the transaction rate increases

or (not quite so good)

Document steps taken to ensure that the system is scalable.
Managing your time

1. Set aside time for project work on your weekly timetable
   a. use your ‘project time’ slots to work on your project
   b. don’t let other work interfere with progress on your project
   c. don’t let your project interfere with the progress on other modules
Managing your time (cont)

2. Make an action plan
   a. check it regularly
   b. notice when you are slipping behind
   c. change it when necessary
   d. document any slipping or changes – you will need them in your report

If you are going to get a 1st class mark, you should have done a project worthy of a 3rd class by the time of Interim Project Report.
Managing your time (cont)

3. Conduct a short feasibility study before you start your project:
   a. do a "hello world" project ASAP.
   b. Iterate and go on

4. Start planning your final report as early as possible, and leave plenty of time for writing up your work
Managing your time (cont)

5. Meet every week with your supervisor 10-15 minutes
   a. Prepare the meeting agenda – weekly report. Email it to your supervisor and to project’s email well in advance.
   b. Make an appointment – verify on the day
   c. Go to the meeting with your attendance form, and ask your supervisor to sign it. Submit it within 24 hours to CS office
   d. Be brief and precise. Do not ask for solutions unless you are desperate – you are loosing marks
   e. Arrange for next meeting

If you need more than 20 minutes, then something is wrong.
Putting together a Project

- Find a problem to solve
- Think about how you might solve it
- Make sure it is something you are interested in working on
- Work out a set of ‘high level’ objectives for your project
Putting together a Project (cont)

- Check that you fully understand your project objectives (try them on someone else)
- Check that your objectives are achievable
- Make sure that the resources you require will be available when you need them, or preferably NOW.
The importance of getting started NOW

- The hardest part of any project is deciding what you will do
- If you put off the decision, or keep things vague, you
  - make the decision harder
  - make starting on the practical work harder

Based on notes from UH core team (James Malcolm, Steve Hunt, Colin Egan)
Minor Modifications to allow for IST cases, for example deadlines
The importance of getting started NOW (cont)

- Students who leave it until late to decide what they are going to do tend not to do well in their projects
- Students who make up their minds early have more opportunities to revise their projects later

If you are going to get a 1\textsuperscript{st} class mark, you should have done a project worthy of a 3\textsuperscript{rd} class by the time of Interim Project Report.
Extending a basic project

Suppose you can only think of a simple system...

CUSTOMER  ORDER  PRODUCT

Based on notes from UH core team (James Malcolm, Steve Hunt, Colin Egan)
Minor Modifications to allow for IST cases, for example deadlines
How can you extend this?

- Extend the requirements
- Measure what you have built
- Compare alternatives
- Focus on the process
Extend the requirements for a better mark (some examples only)

- Allow the company to nominate salesmen responsible for different customers, and empower them to offer different discounts to their customers, depending on circumstances.
- Allow price changes (without changing old billing information) – if not already core!
- Time limited special offers (3 for 2, etc).
Extend the requirements some examples only

- Special offers based on purchase history.
- Allow kits of parts, with instructions, possibly for a special price; allow customers to edit their order.
- Allow leasing deals (where a customer gets a cheap lease on a piece of capital equipment in return for promising to buy supplies for it).
- Produce monthly reports comparing sales with targets for particular products/salesmen.
Extend the requirements some examples only

• Allow "call off orders" where a customer commits to buying a large quantity of a short shelf life item (to get a good price), but it is only delivered in small batches when requested.

• Stock control, supplier orders, choosing supplier depending on what mix of products is low on stock, increasing order level slightly to get a better discount.
Extend the requirements some examples only

- CRM facilities: exchanging contacts with other systems in the company (targetted, not just a database dump)
- Data warehousing: spotting trends, common purchasing patterns, etc
- Implement your system as a Web Service
- Interact with Google Maps
Propose further work

Building Community

- Allow customers to talk to each other (not just a chat facility)
- Customer comments on product quality
- Customer hints on how to use products
Marketing

• targeted emails

e.g.

Other people who bought a widget also bought one of these. Why don't you buy a widget cleaning kit? Your widget is now 6 months old and could do with cleaning (only E20!)

And other rubbish that marketing people like, but I call them SPAM
Measure what you have built

• Proper evaluation of the user interface and act on the results
• Measure performance of your database as number of customers goes up
• Generate realistic test data automatically, to allow large scale testing
Compare alternatives

- Build the same system with a different database
- Build the same system using XML
- Use a different tool/language to build the GUI
- Use a different approach to constructing the user interface

AND reflect on the differences
Focus on the process

- Take a scientific or software engineering approach to the task
- Document properly the design (*Sorry I forgot that everyone is doing that!*
Sources of help

• Use books on Software Engineering
  o SE is required for any kind of project – not only for IS or SE projects.

• Use papers not just books.
  o Read the assessment criteria of the module. You will see that you cannot get an A grade if you have not use papers in your references.

• Use Studynet, Electronic Libraries, Public Libraries.
Sources of help

DO NOT use for referencing

- lecturer’s notes,
- product manuals,
- company web sites or white papers (for justifying the use of their product).
- unsolicited web sites, - (you do not know the validity of the information)
- Wikipedia and similar sites.

Based on notes from UH core team (James Malcolm, Steve Hunt, Colin Egan)
Minor Modifications to allow for IST cases, for example deadlines
Detailed project specification / plan (DPS/P)

First you need to prepare your Detailed Project Specification / Plan (DPS/P) and get started.

- Due: See Project Calendar
- Worth: 2%
- Hard copy Submission and by email
Reminder of the process

- If you have given us an Initial Proposal you would have by now a supervisor. He will help you with the DPP.

Either wise

- You will be allocated a Project Supervisor after the DPP has been handed in and meet weekly in term time for about 10-15 minutes.

If you do not submit a DPP by one week after the deadline, you will be considered as not enrolled for project.
Again: Meeting your Supervisor

Find your tutor’s allocated time for project meetings. Ask Valy and Eleana.
Meet weekly in term time for about 10-15 mins in your tutor’s allocated time

- Weekly report on what you have done. Email to your supervisor and to project’s email
- Discuss next steps
- Fill attendance meeting form every time and submit to the office within 24 hours.
The end

Don’t forget

Go, read the suggestions, get ideas and choose a project!

Time is passing