Degree projects in Computing

Jürgen Börstler, Niklas Lavesson, Veronica Sundstedt

{jub,nla,vsu}@bth.se

January 28, 2014

Reference material

- Thesis Projects A Guide for Students in Computer Science and Information Systems, M. Berndtsson, J. Hansson, B. Olsson, B. Lundell, Second Edition, Springer, 2008

Developing a project plan

- Introduction - context, problem, related work, relevance, identification of gap
- Aim - the overall goal of the project
- Objectives - how the aim will be achieved
- Research questions
- Method
- Expected outcomes
- Time and activity plan
- Risk management
- References

Choosing a subject area

- Review your knowledge, interests, and skills
- Discuss with fellow students and teachers
- Find topics or areas published on the homepages of researchers at BTH
- Pitch ideas for researchers and teachers
- Review the state-of-the-art or the state-of-practice of an area you are interested in

Choosing a subject area

- ACM Computing Classification System
  - http://www.acm.org/about/class/1998
- Sublevels often correspond to keywords in relevant papers
- Combine areas (interesting problems)
  - Databases and human-computer interaction => interfaces for database systems
- Your interest and skills should guide you

ACM Computing Classification System

- I.3 COMPUTER GRAPHICS
  - I.3.0 General
  - I.3.1 Hardware Architecture (B.4.2)
    - *Graphics processors*
    - *Hardcopy devices* [**]
    - *Input devices*
    - *Parallel processing*
    - *Raster display devices*
    - *Storage devices* [**]
    - *Three-dimensional displays* [**]
    - *Vector display devices* [**]
  - I.3.2 Graphics Systems (C.2.1, C.2.4, C.3)
    - *Distributed/network graphics*
    - *Remote systems* [**]
    - *Stand-alone systems* [**]
  - I.3.3 Picture/Image Generation
    - *Antialiasing* [**]
    - *Bitmap and framebuffer operations*
    - *Digitizing and scanning*
    - *Display algorithms*
    - *Line and curve generation*
    - *Viewing algorithms*
  - I.3.4 Graphics Utilities
    - *Application packages*
    - *Device drivers* [**]
    - *Graphics editors*

- D.2 SOFTWARE ENGINEERING (K.6.3)
  - D.2.0 General (K.5.1)
    - *Protection mechanisms*
    - *Standards*
  - D.2.1 Requirements/Specifications (D.3.1)
    - *Elicitation methods (e.g., rapid prototyping, interviews, JAD)* NEW!
    - *Languages*
    - *Methodologies (e.g., object-oriented, structured)* Revised
    - *Tools*
  - D.2.2 Design Tools and Techniques Revised
    - *Computer-aided software engineering (CASE)*
    - *Decision tables*
    - *Evolutionary prototyping* NEW!
    - *Flow charts*
    - *Modules and interfaces*
    - *Object-oriented design methods* NEW!
    - *Petri nets*
    - *Programmer workbench* [**]
    - *Software libraries*
    - *State diagrams* NEW!
    - *Structured programming* [**]
    - *Top-down programming* [**]
    - *User interfaces*
  - D.2.3 Coding Tools and Techniques Revised
    - *Object-oriented programming* NEW!

ACM Computing Classification System

- H.5 INFORMATION INTERFACES AND PRESENTATION (e.g., HCI) (I.7)
  - H.5.0 General
  - H.5.1 Multimedia Information Systems
    - *Animations*
    - *Artificial, augmented, and virtual realities* Revised
    - *Audio input/output*
    - *Evaluation/methodology*
    - *Hypertext navigation and maps* [**]
    - *Video (e.g., tape, disk, DVI)*
  - H.5.2 User Interfaces (D.2.2, H.1.2, I.3.6)
    - *Auditory (non-speech) feedback* NEW!
    - *Benchmarking* NEW!
    - *Ergonomics*
    - *Evaluation/methodology*
    - *Graphical user interfaces (GUI)* NEW!
    - *Haptic I/O* NEW!
    - *Input devices and strategies (e.g., mouse, touchscreen)*
    - *Interaction styles (e.g., commands, menus, forms, direct manipulation*
    - *Natural language* NEW!
    - *Prototyping* NEW!
    - *Screen design (e.g., text, graphics, color)*
    - *Standardization* NEW!
    - *Style guides* NEW!
    - *Theory and methods*
    - *Training, help, and documentation*
    - *User-centered design* NEW!

- D.3 PROGRAMMING LANGUAGES
  - D.3.0 General
    - *Standards*
  - D.3.1 Formal Definitions and Theory (D.2.1, F.3.1, F.3.2, F.4.2, F.4.3)
    - *Semantics*
    - *Syntax*
  - D.3.2 Language Classifications
    - *Applicative (functional) languages* Revised
    - *Concurrent, distributed, and parallel languages*
    - *Constraint and logic languages* NEW!
    - *Data-flow languages*
    - *Design languages*
    - *Extensible languages*
    - *Macro and assembly languages*
    - *Microprogramming languages* [**]
    - *Multiparadigm languages* NEW!
    - *Nondeterministic languages* [**]
    - *Nonprocedural languages* [**]
    - *Object-oriented languages*
    - *Specialized application languages*
    - *Very high-level languages*
  - D.3.3 Language Constructs and Features (E.2)
    - *Abstract data types*
    - *Classes and objects* NEW!
    - *Concurrent programming structures*
    - *Constraints* NEW!

Formulating a problem

- Identify a problem you want to explore
- Find problems of general interest
    - What you want and can do
    - Read the literature, what has been done before?
    - Talk to your supervisor/external partner
    - Is it worthwhile?
    - Write down arguments for why it is relevant
        - NOT I would like to do it because it is interesting
        - NOT - I would like to learn more about
        - Ideally a gap in the literature

Example projects

- Descriptive projects
  - Present state-of-the-art
- Theory oriented projects
  - Extend or compare existing theoretical models
- Applied projects
  - Conducting experiments, implementations
- Comparison of theory and practice

Identifying gaps

- Identify what has been done before
- Search relevant conferences and journals
- Talk to researchers at BTH
- Talk to external partners

Developing objectives

- Objectives are formulated so that fulfilling them leads to the aim being satisfied
  - Aim: change a car colour from green to red
  - O1: Change color of everything that is not red
  - O2: Get appropriate red paint
  - O3: Make sure car is operationable after painting

Choosing a method

- Interview
- Case study
- Survey
- Implementation
- Experiment
  - Controlled, Quasi, Natural, Observational

Interviews

- Open interview (qualitative research)
  - Clear purpose/issues are not planned in advance
  - Directed to open up important issues
  - Avoid leading/closed questions
- Closed interview
  - Fixed set of questions, pre-structured
  - Repeatability advantage
  - Limited questions, drawback?
  - Survey research/statistics (more suitable)

Case studies

- In-depth exploration of a phenomenon in its natural setting
  - Organisation, department, group, individual, etc.
  - Example: a specific software development organisation and their use of a tool
  - Aim to generalise
  - Your bias, the details, and a representative case

Surveys

- Statistical techniques to analyse responses
- Limited means to reach a large number of respondents
- Difficult to investigate complicated issues
- Impossible to clarify questions
- Sample strategies to ensure (stratified) random selection
- Is the right person answering?
- Motivation for participation low

Implementations

- Developing new solutions
  - Software architecture, method, procedure, algorithm, etc.
  - To evaluate if it possess the proposed advantages
  - Goal is to demonstrate solution has certain properties of behaves in a specific way
  - Instrument for measuring
- Compare with existing solutions
- Good software development practice
  - Validity (implementation reflect the solution you propose)
  - Realiability (robustness of the implementation)
    - Performance, bugs, etc.

Experiments

- True, or controlled, experiments compare the results obtained from an experimental group and a control group
  - One or more variables are manipulated in the experimental group to investigate causal relationships
  - The objective is to determine whether the variables cause a certain effect (that is not present in the control)
- Natural, or quasi, experiments rely only on observations of the variables of some system
  - Natural experiments are sometimes called observational studies and are used to investigate correlations
  - Cause-effect cannot be identified only correlation

Hypothesis testing

- Hypotheses are defined so that they can be tested and rejected (falsifiability)
- Hypotheses can thus be rejected but never accepted or proven
- If a hypothesis has been tested systematically without being rejected
  - The evidence supports the hypothesis
  - Example: the theory of evolution is based on systematically tested hypotheses from a wide array of fields

Statistical hypothesis testing

- A method for making decisions based on observation
- Attempts to reject a particular hypothesis, the null hypothesis
    - Null hypothesis (H0)
        - H0: there is no difference in taste between coke and diet coke
    - Alternate hypothesis (H1)
        - H1: there is a difference
- In statistics, a result is called statistically significant if it is unlikely to have occurred by chance, according to a pre-determined threshold probability, the significance level
- Beware of fishing expeditions: more tests increase the likelihood that something appears by chance

Experimental design

- An experimental design can be constructed once the objective of the experiment is defined, e.g. to test
  - Algorithm A is more efficient than algorithm B
  - Noise has a negative impact on classification accuracy
  - Users are more productive with software A than with software B

Experimental design

- An experimental design comprises, e.g.
  - A dependent variable (that may show potential effect)
  - Independent variables (that may have potential cause)
  - Confounding variables (excluded variables that also cause effect)
  - Experimental type (single/multi-factor, within/between subjects, etc.)
  - Sample (a random subset of the population)

Example

- Suppose we want to compare the predictive accuracy of models generated by two different algorithms, A and B
  - Null hypothesis: A and B perform equally well
  - Alt. Hypothesis: A and B differ in accuracy
- We collect two data samples
  - Accuracy of A and B models on 30 data sets
- We conduct a non-parametric statistical test to decide whether to reject the null hypothesis

Data collection

- The data collection depends on
  - Type: quantitative/qualitative/mixed
  - Availability
  - Need and conditions
- What type of data do you need for your project? How much do you need?
- How much data are available and within what time frame?

Choice of analysis method

- Most research methods require specific types of analysis, e.g.
  - Experiments, meta studies, and quantitative surveys: statistical analysis
  - Interviews: e.g. discourse or content analysis
- Each analysis method has caveats and limitations
- The chosen analysis method should be compatible with the data and the research method

Synthesis

- During the analysis
  - Complex phenomena, objects, components, ideas, etc. are broken down into pieces
  - Each piece is analysed using the appropriate method
- During the synthesis
  - The results and conclusions from analysing the pieces are glued together to form a coherent and new understanding, view, or contribution
  - Taken together, what does all that mean?

Conclusions

- The conclusions section should
  - be short, concise, and meaningful
  - describe the main conclusions that can be drawn from analysing your results
  - Not include any new content or citations
- The most common order to read a thesis or paper
  - Title, abstract, conclusions, the rest...
  - Make your reader continue reading

Describing a contribution

- Each contribution should be clearly identified
  - Type: scientific, societal, engineering, etc.
  - Description and potential impact/implication
- The aim of a Master's thesis (or equivalent) is to contribute new knowledge to the studied field
  - In reality, many students do not succeed but still graduate
  - Real contributions should make it possible to publish the thesis as a research paper

Academic writing

- Chapter 14 in Berndtsson et al. (2008)
- Please refer to the recommended reading
- The text should be
  - Clear, concise, and to the point
  - Free from grammatical error
  - Based on credible sources and coherent arguments
- The reader is always right

Presentation and defense

- Chapter 12 in Berndtsson et al. (2008)
- The presentation should be
  - Clear, concise, well rehearsed, and interesting to listen to
- The defense should be
  - Lead by the opponent and the examiner
  - Based on scientific reasoning and logic

Acknowledgments

- Tony Gorschek, Robert Feldt, Andrew Moss
- http://www.princeton.edu/~achaney/tmve/wiki100k/docs/Statistical_hypothesis_testing.html