Department of Computer Science and Information Systems

BSc Digital and Technology Solutions (Software Engineer) Degree Apprenticeship

Final Year Project

Handbook

Contents

1. Introduction

The final year project is a work-based project that is designed to assess the practical skills, knowledge and professional behaviours laid out in the Digital and Technology Solutions Degree Apprenticeship Standard. It will require you to design, implement and test a significant piece of software, or a software component, to at least prototype level, in order to achieve a set of clearly defined business objectives. It will also require you to undertake a significant element of project planning, including a business rationale, risk analysis, project scoping and detailed costing. Throughout the project, you will be required to demonstrate the ability to work independently, to act professionally, to show initiative, and to work with academic rigour.

2. General Information

Module status	Core
Credit value	30 credits
Level	6
Module coordinator	Gordon McIntyre
Duration	2 terms (autumn and spring). Approximately 6 months.

3. Project Selection

You will work with your academic and workplace project supervisors to identify and select a suitable project. The project must be agreed by both supervisors before work can begin. Before a project is signed off, it must be clear that it:

- Can be completed within the stated time constraints.
- Can be completed within extant financial constraints.
- Can be completed given available technical resources (e.g. tools and network access).
- Is at an appropriate level of difficulty (not too challenging and not insufficiently challenging).
- Covers, as far as possible, the full range of skills and knowledge described in the apprenticeship standard (e.g. scoping, planning, costing, analysis, design, implementation, testing, etc.).
- Is original in nature (e.g. has not been done before).
- Satisfies all ethical and legal requirements for software engineering projects (e.g. copyright, confidentiality, etc.).

Previous software engineering projects within DCSIS include the following:

- Design of software to assist in a data re-cabling project at the LSE.
- Development of an examination timetabling system for SOAS.
- Stock control database.
- Children's educational software program.
- Computerised theatre booking system.
- Implementation of SAGE accounting system in a fashion design company.
- Archiving system for financial papers: storage and retrieval.
- An expert system to select racehorses based on breeding.

4. Supervision

4.1 Academic Supervisor

The academic supervisor will provide academic support to the student throughout the duration of the project. This will include:

- Monitoring and ensuring progress.
- Providing academic and technical guidance.
- Providing feedback on submitted project elements (e.g. proposal and draft).
- Grading submitted elements.
- Answering student queries.

To find a suitable academic supervisor for your project, you should consult the DCSIS project supervisor list. However, please be aware that there is a limit to the number of projects a member of staff can supervise in an academic year.

4.2 Workplace Supervisor

The workplace supervisor will provide technical and material support for the student in the workplace. This will include:

- Monitoring and ensuring progress.
- Ensuring availability of required technical and material resources (e.g. servers, tools, platforms, study time).
- Providing technical guidance.
- Answering student queries.

4.3 Supervision Meetings

You should arrange a face-to-face progress meeting w your academic supervisor, usually midway through the project. Other progress meetings can be held via Skype or telephone. General queries can be dealt with via email.

It is your responsibility to maintain contact with your academic supervisor. Students who fail to keep in contact their supervisor, or miss appointments cannot expect to be chased.

You should arrange project supervision meetings with your workplace supervisor, as required.

5. Project Deliverables

5.1 Project Portfolio

The project portfolio comprises the project proposal, the project report and the project software.

5.1.1 Proposal (2000 words, +/- 10%)

The project proposal will provide the following sections:

- Problem Outline
  Provides a short account to the problem that the project will address.
- Solution Outline
  Provides an initial indication of the proposed solution.
- Assumptions
  Details any assumptions that you are making in your initial understanding of the problem.
- Cost-benefit Analysis
  Provides an analysis of the cost of the project in terms of time, resources and money. Weighs the cost of the project against the identified benefits that the project will bring to the commissioning organisation.
- Risk Analysis
  Provides an appraisal of:
  - Things that can go wrong.
  - The likelihood of things going wrong.
  - The consequences of things going wrong.
  - Measures that can be put in place to stop things going wrong.

- Project Management Methodology
  Identifies the project management methodology to be used to manage the project. Provides a clear rationale for the choice of methodology.

- Design Methodology
  Identifies the design methodologies and techniques to be used to model the software. Provides a clear justification for the methodologies and techniques chosen.

- Implementation Technologies
  Provides an initial assessment of the implementation technologies that will be required to complete the project (e.g. HTML, Java, PHP, SQL, etc.).

- Tools
  Provides an initial indication of the tools that will be required to complete the project.

  Must include an account of:

  - Modelling tools.
  - Implementation tools.
  - Testing tools.
  - Configuration management tools (e.g. version control tools).

- Testing
  Provides an initial account of how the software will be tested.

- Outline User Requirements
  Provides an initial account of the functional, non-functional and security requirements for the project. These can be amended as the project progresses.

- Project Plan
  Provides a detailed plan of work for the project. All project tasks should be identified, and decomposed into sub-tasks. Time estimations and dependence relations should be shown. The project plan should be completed using specialist project planning software. It should be presented in GANTT format. Note that the project plan can be finessed as the project progresses.

  Provides an assessment of the resource constraints under which the project will be carried out.

5.1.2 Report (6000 words, +/- 10%)

The project report will contain the following sections:

- Introduction
  Gives a clear background to the project. Details the chosen topic, why the topic is relevant or of interest to you, the aims and objectives of the project.
- Research
  Summarizes the background reading you have done for the project subject area. References any relevant theories, studies or related projects that inform your work. Must include only *authoritative* sources. All sources must be cited.
- Analysis
  Provides a thorough analysis of the problem domain. Analyses and documents in detail the functional, non-functional and security requirements for the project. The requirements should be fully and appropriately documented, actionable, measurable, testable and linked directly to previously identified business needs. Requirements should also be explicitly linked to specific system stakeholders.
- Design
  Provides a complete set of designs for the project software. Design diagrams should be annotated with explanations and textual commentary where appropriate. Should cover the system from all perspectives (e.g. UI, data, functionality, and network).
- Implementation
  Provides a comprehensive account of the implementation of the project. Includes a deployment plan. Makes reference to programme code where appropriate. Shows how your coding practices factor in security concerns. Uses code snippets to exemplify areas of complexity that you have dealt with. Details your use of version control tools to ensure code integrity. Shows how the coding solutions you have produced are optimal solutions. Details any problems you have had to overcome in the course of the implementation.
- Testing
  Provides a detailed and comprehensive test plan that contains a full range of appropriate tests for the software you have built. Includes unit and functional testing (e.g. testing against functional and non-functional requirements). Provides test data and an analysis of that data. Shows how tests have revealed problems, and helped to overcome those problems. Alternatively, makes recommendations for improvements to the software based on test results.
- Conclusion
  Analyses the success or otherwise of the project. Shows where the project has met its

objectives, and where it has failed to meet its objectives. Makes recommendation for further development.

- Reflections
  Details from a personal/reflective perspective:
  - What you have achieved in the course of the project.
  - The main lessons you have learned.
  - Mistakes you made.
  - Things you would do differently next time.
- References
  Contains references for all external sources that you have read during your project, whether they are cited in text or not.
- Appendices
  The appendices section should be used for helpful and supporting material that would otherwise clutter, or break up the main body of the report (e.g. long lists of data, complex graphics, correspondence, etc.).

5.1.3 Software

You must provide a fully working version of your software implemented to at least prototype standard.

5.2 Presentation (30 – 40 minutes)

You will give a presentation focusing on the outcomes of the synoptic project. The presentation will be attended by your academic and workplace mentors. Other Birkbeck and employer staff, and other students may also attend. The presentation will involve a significant Q&A session. You should use Microsoft PowerPoint to prepare the slides for your presentation.

The purpose of the presentation is to:

- Explore aspects of the project work, including how it was carried out.
- Demonstrate the extent to which you have achieved your goals.
- Demonstrate the quality of the work you have produced.
- Show how you have approached your work and dealt with the problems you encountered.
- Allow Birkbeck and employer staff to question you on aspects of your project and your portfolio.
- Demonstrate your ability to design and give an academic presentation.

The presentation will take place following the completion of the project and the submission of the final report. You will be given appropriate notice of their presentation time and venue.

The venue will be at the university or employer location, as appropriate. The presentation and Q&A will typically last 30 – 40 minutes. It will be conducted face to face or in exceptional circumstances via live video link.

6. Writing up

6.1 Presentation

Your work should be presented to a high standard. An easily readable font should be used, as well as realistic font sizes and line spacing settings. Headings should clearly indicate the hierarchy of the document. Sections and paragraphs should be clearly and consistently spaced. Graphics and tables should be readable at normal zoom, and should have a caption. If in doubt about details of presentation, please consult your academic supervisor.

Page 1 should be a cover sheet. This should contain your name, the name of your degree, the title of the project module, the project title, the project date and a word count. It should also contain the following declaration:

University of London [year]. This report is the result of my own work except where explicitly stated in the text. The report may be freely copied and distributed provided the source is explicitly acknowledged.

Page 2 should be a table of contents. The table of contents should be indexed to show sections and subsections, and page numbers.

The report should be bound using the plastic 'comb' method with a transparent upper cover and a heavy card backing cover. Print Shop in ULU provide this service (and also photocopying) at reasonable rates. Your name and the year in brackets should appear along the spine of the binder

6.2 Drafting and Proofreading

The project report should undergo a drafting and editing process prior to submission. You will have a chance to submit a draft version of the report for comment prior to submission.

Proofreading by a third party is acceptable provided it constitutes no more than advice on the following: spelling and punctuation, formatting, grammar and syntax. If you have doubts about the quality of your written English, you are encouraged to use a third-party proofreading service.

6.3 Writing Style

All written elements, apart from the reflective writing section in the report, should be written in standard academic English. The reflective account can be written from a personal perspective.

6.4 Research

6.4.1 References

You must provide references for all source material that you have read in the course of your project (e.g. books, journal articles, web sources, etc.). This includes cited material and non-cited material. References should be written in the Harvard style.

6.4.2 Citation

All information from external sources that you include in your written work must be clearly and correctly cited. This includes any paraphrases or quotations. Citation should be done in a standard academic style.

6.4.3 Plagiarism

Plagiarism is the presentation of another person's thoughts, words, ideas, designs or programs as though they were your own. It is a serious examination offence, and treated very seriously by the College. Direct quotation from the work of others (published or unpublished) must always be clearly identified as such by being placed in quotation marks, with a full reference to the source provided. Where a summary of someone else's writing or ideas is made, i.e. expressed in different words to the original, reference to the source must still be provided. A series of short quotations from several different sources, if not clearly identified as such, constitutes plagiarism just as much as an unacknowledged long quotation from a single source.

All submissions will be carefully scrutinized by staff for signs of plagiarism, and written submissions will be checked using plagiarism detection software.

7. Submission

7.1 Submitting your Proposal

Your project proposal should be submitted as a Microsoft Word document. The proposal document should be named *username_bsc_dts_project_proposal_year.docx*. The proposal should be submitted via Moodle before the deadline given by your academic supervisor.

7.2 Submitting your Report

Your project report should be submitted as a Microsoft Word document. The report document should be named *username_bsc_dts_project_report_year.docx*. The report should be submitted via Moodle before the deadline given by your academic supervisor.

Your report should also be submitted in hard copy (see details above) to the programme office before the deadline given by your academic supervisor.

7.3 Submitting your Software

7.3.1 Web-based Applications

For web-based applications, you must provide an accessible URL. The URL should be clearly indicated on the cover page of your report. Failure to provide a URL or the provision of a non-functioning URL may lead to a significant loss of marks. If login credentials are also required, you must also provide these.

7.3.2 Desktop Applications

For desktop applications, you must provide an executable file. This must be submitted on a USB stick. The USB stick should be submitted along with your report to the programme office before the deadline given by your academic supervisor.

7.3.3. Exceptions

Where a piece of software requires a specific platform to operate, or specific tool to run then an arrangement must be made with your academic supervisor for you to provide access to the platform and/or tool required. In extreme cases, where it is impossible to provide the means to run the software, an arrangement must be made with your supervisor for a live demonstration of the software.

7.4 Penalties for Late Submission

All submissions are subject to a deadline and the late deadline. You should always try to aim to submit before the deadline. If you cannot submit before the deadline, you should try to submit before the late deadline. However, submissions made after the deadline and before the late deadline will be subject to a grading cap of 40%. Any submission made after the late deadline will be awarded 0%.

8. Assessment

8.1 Assessment Weightings

The following weightings will apply to project elements:

Project Portfolio	85%	Proposal	20%
		Report	50%
		Software	30%
Project Presentation	15%		

8.2 Assessment Criteria

8.2.1 Project Portfolio

Proposal

The following criteria will apply to the marking of the proposal.

- Problem Outline, Solution Outline and Assumptions (5%)
  Provides the reader with a clear sense of the problem context and a proposed solution. Provides the reader with a clear and realistic account of any assumptions that are being made on the part of the author. Excellently written. Direct. Precise. No ambiguity or circumlocution.
- Cost Benefit Analysis (15%)
  Provides a clear analysis of the cost of the project in terms of time, resources and money. Weighs the cost of the project against the identified benefits that the project will bring to the commissioning organisation. Excellently written. Direct. Precise. No ambiguity or circumlocution. Any relevant data is included. Uses and references established techniques for CBA.

- Risk Analysis (15%)
  Provides a clear and coherent appraisal of (1) Things that can go wrong during the project (2) The likelihood of things going wrong. (3) The consequences of things going wrong (4) Measures that can be put in place to stop things going wrong. Excellently written. Direct. Precise. No ambiguity or circumlocution. Any relevant data is included. Uses and references established RA techniques.

- Choice of Project Methodologies, Techniques, Technologies and Tools (10%)
  - Details project management methodology chosen. Provides rationale for choice of project management methodology.
  - Details project management methodology chosen Provides rationale for choice of design methodologies and techniques.
  - Provides rationale for implementation technologies and tools (This should include system design tools, implementation tools, testing tools, configuration management tools, and version control tools).
  - References best-practice and, where appropriate, supported by theory.
  - Excellently written. Direct. Precise. No ambiguity or circumlocution.

- Testing (15%)
  Provides an initial account of how the software will be tested. Appropriate tests chosen for application being tested. References best-practice and, where appropriate, supported by theory.

- User Requirements (15%)
  Provides an initial set of functional, non-functional and security requirements for the software. No major omissions. Each requirement is discrete and precise.

- Project Plan (10%)
  Provides a detailed plan of work for the project. All project tasks should be identified, and decomposed into sub-tasks. Time estimations and dependence relations should be shown. Tolerances should be detailed. The project plan should be completed using specialist project planning software. It should be presented in GANTT format.
- Research (5%)
  Evidence of detailed research in form of detailed references. References appropriate and properly formatted in Harvard Style. Citations done correctly.
- Spelling, Grammar and Presentation (5%)
  Excellent spelling and grammar throughout. Standard academic English used. Excellent paragraphing and sub-division of the report. All required elements present (e.g. cover sheet, page numbers, TOC, etc.). Presented to a high standard. Graphics and tables properly labelled and indexed.
- Completeness (5%)
  The work falls within the required word count. The work contains no circumlocution.

8.2.2 Report

The following criteria will apply to the marking of the report.

- Introduction (5%)
  Details the topic, the background to the project, why the topic is relevant or of interest, what the author hopes to achieve, the aims and objectives of the project. Has a clear narrative. Follows a standard introduction format with general statements, specific statements and a clear thesis statement. Excellently written. Clear, direct and precise. No ambiguity or circumlocution.
- Research (10%)
  Summarizes the background reading you have done for the project subject area. References any relevant theories, studies or related projects that inform your work. Includes only authoritative sources. All sources are clearly and properly cited. Excellently written. Clear, direct and precise. No ambiguity or circumlocution.

Evidence of detailed research in form of detailed references. References appropriate and properly formatted. Citations done correctly.

- Analysis (15%)
  Provides a thorough analysis of the problem domain. Analyses and documents functional non-functional and security requirements. Requirements should be fully and appropriately documented, actionable, measurable, testable and linked directly to previously identified business needs. Requirements should be explicitly linked to specific system stakeholders. Excellently written. Clear, direct and precise. No ambiguity or circumlocution.
- Design (15%)
  Provides a complete and appropriate set of designs for the project software. Design diagrams should be annotated with explanations and textual commentary where appropriate. Designs should be without error. They should cover the system from all perspectives (e.g. UI, data, functionality, and network). Graphics should be of professional quality, and should be properly incorporated into the report. Excellently written. Clear, direct and precise. No ambiguity or circumlocution.
- Implementation (15%)
  - Provides a comprehensive account of the implementation of the project. Makes reference to programme code where appropriate. Uses code snippets to exemplify areas of complexity that you have dealt with. Shows how the solutions you have produced are optimal solutions.
  - Accounts for any deviations from the original project plan. Shows how deviations were managed in accordance with your chosen project management methodology.
  - Details any problems you encountered in the course of the implementation, and how they were overcome. Shows how problems were escalated. References your chosen project management methodology.
  - Excellently written. Clear, direct and precise. No ambiguity or circumlocution.

- Testing (15%)
  Provides a comprehensive test plan that contains a full range of appropriate tests for the software you have built. Includes functional and unit testing. Provides test data and a thorough analysis of that data. Shows problems revealed by testing, and how you have overcome those problems. Makes recommendations for improvements to the software based on test results. Data well-presented. Excellently written. Clear, direct and precise. No ambiguity or circumlocution.

- Conclusion (5%)
  Analyses the success or otherwise of the project. Shows where the project has met business needs, and where it has failed to meet its business needs. Makes recommendation for further development. Excellently written. Clear, direct and precise. No ambiguity or circumlocution.
- Reflections (10%)
  Details from a personal and reflective perspective (1) What you have achieved in the course of the project (2) The main lessons you have learned (3) Mistakes you have made (4) Things you would do differently next time. Excellently written. Clear, direct and precise. No ambiguity or circumlocution.
- Spelling, Grammar and Presentation (5%)
  Excellent spelling and grammar throughout. Standard academic English used. Excellent paragraphing and sub-division of the report. All required elements present (e.g. cover sheet, page numbers, TOC, etc.). Presented to a high standard. Graphics and tables properly labelled and indexed.
- Completeness (5%)
  The work falls with the required word count. The work contains no circumlocution.

8.2.3 Software

The following criteria will apply to the marking of the project software.

- Functional Requirements (30%)
  Meets all the functional requirements (as detailed in the project report).
- Non-functional Requirements (25%)
  Meets all the non-functional requirements (as detailed in the project report).
- Security Requirements (15%)
  Meets all the security requirements (as detailed in the project report).
- Fitness for Purpose (15%)
  Software is fit for purpose. Works as expected. Bug and error free.
- Quality of Coding (15%)
  Software is optimally coded (e.g. programmatic code, markup, SQL, etc.).

8.2.4 Presentation

The following criteria will apply to the marking of the presentation.

- Content (40%)
  Addresses all the required topics. (1) Sets out the business needs that drove the project. (2) Sets out the extent to which the business needs were met. (3) Provides a realistic assessment of standard of work produced. (4) Explores key aspects of the project. (5) Details any issues arising from the project.
- Organisation (20%)
  Well-structured. Clear introduction. Clear conclusion. Clear progression through slides. Clear narrative. Information well sequenced. Meets time restriction (Not too long or too short).
- Presentation Skills (30%)
  Maintains eye contact. Has clear, even-paced and audible voice. Uses visual aids appropriately. Engages audience. Deals with audience questions in a precise, controlled and clear manner.
- Use of Technology (10%)
  Slides not overloaded. Slides well-organised and easy to read. Graphics well-presented. No spelling or grammar errors. Student uses AV equipment competently.

8.3 Marking

All projects are marked in accordance with the Birkbeck Marking and Moderation Policy.

Each project is first marked by the project supervisor and second marked by another DCSIS academic staff member who has not been involved in teaching on the programme. In the event of disagreement between the supervisor and the second marker, a third marker and/or the External Examiner will also consider the project. Projects may be reviewed by the External Examiner, although in practice usually only a sample are reviewed.

9. Ethics

9.1 Intellectual Property Rights

In the course of the project, you may have course to use digital or non-digital assets that belong to a third party. Before you use such assets, you should be fully aware of the UK laws governing the use and exploitation of intellectual property. You should also be aware of the College policy on intellectual property rights.

If you have any questions regarding your use of third-party digital or non-digital assets in your project, you should contact your academic supervisor and discuss matters with him/her.

9.2 Confidentiality

Your project may contain confidential material that your employer does not want disclosed to the general public. This could include sensitive data, business ideas, programming code, etc. In such cases, your employer may have reservations about allowing you to submit your work to the College for assessment. In such cases, we are willing to make special arrangements for submission to take place in a manner that does not compromise confidentiality. This may include signing a confidentiality agreement, and/or assessing software at the employer workplace under supervision.

Please make sure that you consult with your employer with regards confidentiality issues relating to your project before you begin. If there are confidentiality issues, please raise these with your academic supervisor at the earliest possible opportunity, so that appropriate arrangements can be put in place for assessment to take place.

10. Suggested Reading

Dawson Christian W, Projects in Computing and Information Systems: A Student's Guide, Addison-Wesley, 2005

Weaver Philip, Success in Your Project: A guide to student system development projects, Prentice Hall, 2004

Alley M, The Craft of Scientific Writing, (3rd Edition), Springer, 1998