BCS LEVEL 6 PROFESSIONAL GRADUATE DIPLOMA IN IT

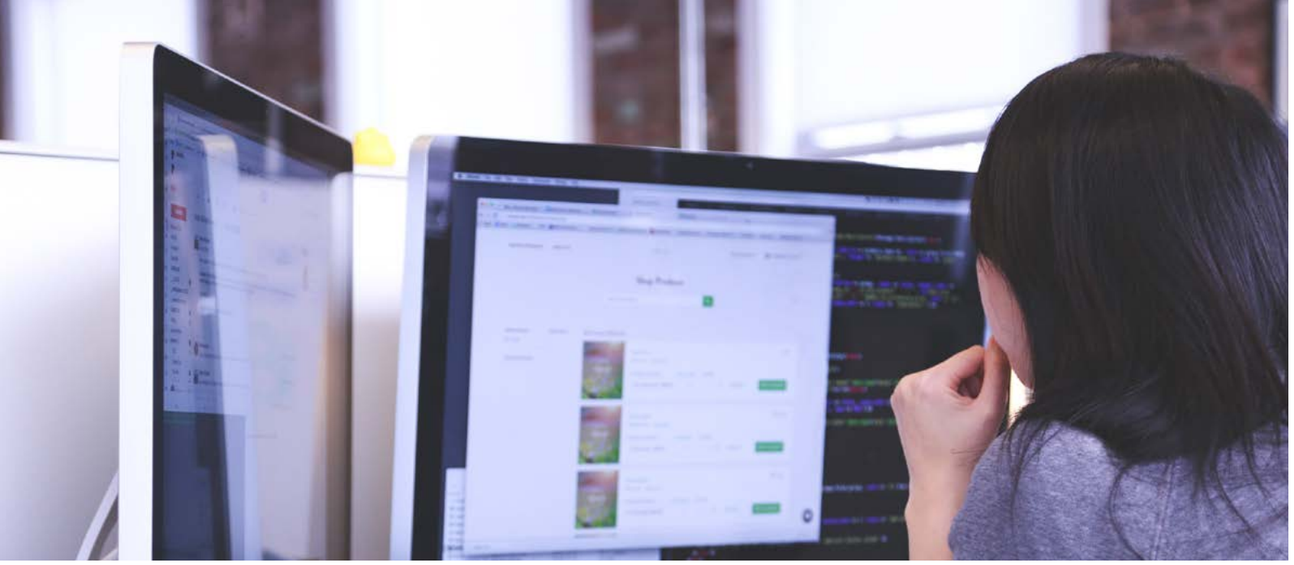
SYSTEM DESIGN METHODS

SYLLABUS

THIS QUALIFICATION WILL BE RETIRING IN 2026

September 2023 v3.1

CONTENTS



Introduction

The final stage within the BCS three-stage Higher Education Qualification programme, the Level 6 Professional Graduate Diploma (PGD) enables candidates who have already achieved the Level 5 Diploma in IT to gain depth of knowledge and expertise in their field.

Our modules have been created in-line with the SFIAPlus framework and latest developments in the industry, giving you a competitive edge in the IT job market and showing your dedication to the industry. You will have the opportunity to learn about topics such as advanced database management, network information systems, web engineering and programming paradigms, as well as to build upon knowledge and skills developed during the Level 5 Diploma.
To successfully achieve the qualification, candidates need to complete:

- One core module (Professional Project in IT)
- Four optional modules

Depending on entrance conditions, completing the Level 6 PGD in IT may support entry onto a Master's degree course at selected global universities.

System Design Methods optional module

The System Design Methods module is an optional module that forms part of the Level 6 PGD in IT – the final stage within the BCS three-stage Higher Education Qualification programme.

Candidates will develop their understanding of the framework for building information systems and how these methods vary, ranging from informal guidelines to mathematical approaches, as well as which methods are applicable to which classes of application. This module is designed for candidates to explore methods appropriate for new classes of applications, and is especially relevant to those who are currently monitoring the implementation of a new method in their day-to-day role, or who are creating one for a specific set of circumstances. The overall aim of the module is to develop candidates' understanding of system design methods, their critical practitioner's knowledge of at least one method, and a general familiarity with several others across a range of application areas.

Qualification Suitability and Overview

Candidates must have achieved the Diploma in IT or have an appropriate exemption in order to be entered for the Professional Graduate Diploma (PGD). Candidates can study for this PGD by attending a training course provided by a BCS accredited Training Provider or through self-study, although it is strongly recommended that all candidates register with an approved centre. Studying with an approved centre will deliver significant benefits.

Candidates are required to become a member of BCS, The Chartered Institute for IT, to sit and be awarded the qualifications. Candidates may apply for a four-year student membership that will support them throughout their studies.

The Level 6 PGD is suitable for professionals wishing to gain an advanced formal IT qualification, and this module may be particularly relevant for candidates who are interested in career opportunities such as engineering consulting or project management, as well as systems design.

Total Qualification Time (Certificate)	Guided Learning Hours (Module)	Assessment Time (Exam)
1414 hours	**250 hours**	**Three hours**

SFIA Levels

This award provides candidates with the level of knowledge highlighted within the table, enabling candidates to develop the skills to operate successfully at the levels of responsibility indicated.

Level	Levels of Knowledge	Levels of Skill and Responsibility (SFIA)
K7		Set strategy, inspire and mobilise
K6	Evaluate	Initiate and influence
K5	Synthesise	Ensure and advise
K4	Analyse	Enable
K3	**Apply**	**Apply**
K2	**Understand**	**Assist**
K1	**Remember**	**Follow**

SFIA Plus

This syllabus has been linked to the SFIA knowledge skills and behaviours required at Level 6

DESN4

Designs components using appropriate modelling techniques following agreed architectures, design standards, patterns and methodology. Identifies and evaluates alternative design options and trade-offs. Creates multiple design views to address the concerns of the different stakeholders of the architecture and to handle both functional and non-functional requirements. Models, simulates or prototypes the behaviour of proposed systems components to enable approval by stakeholders. Produces detailed design specification to form the basis for construction of systems. Reviews, verifies and improves own designs against specifications.

Further detail around the SFIA Levels can be found at www.bcs.org/levels.

Learning Outcomes

Upon completion of this module, candidates will be able to:

- Identify the weaknesses and limitations of proposed design methods
- Develop expertise in selecting a system design method (or combination of methods) appropriate to a given environment, identifying and considering all relevant factors.
- Assist in planning and managing the introduction of a system design method into an existing development environment.
- Assist in planning and implementing monitoring procedures to evaluate the effectiveness of a method in practice.
- Identify areas in which changes to the method might usefully be introduced.

Syllabus

1. Basic elements of system design

Learners will be able to:

1.1 Explain basic system life cycle models.

Indicative content

a. Waterfall
b. V-model
c. Spiral model
d. Prototyping
e. Incremental
f. Reuse-oriented
g. Relationship between activities and life cycle phases
h. Deliverables associated with each phase
i. Relationship between life cycle models and system design methods

Guidance

Candidates will be expected to explain different life cycle models, corresponding life cycle phases and the relationship between life cycle models and system design methods.

1.2 Explain and demonstrate use of graphical notations and techniques.

Indicative content

a. Rich pictures
b. Data flow diagrams
c. Entity life history diagrams
d. Entity relationship diagrams
e. State transition diagrams
f. State charts
g. Enquiry access paths (EAPs)
h. UML diagrams

Guidance

Candidates should be able to explain different modelling techniques, their purpose and applications in systems development

1.3 Explain and demonstrate formal notations.

Indicative content

a. Notation based on mathematical logic and algebra.

Guidance

Candidates should have a basic understanding of formal methods.

1.4 Explain the techniques for validation and verification.

Indicative content	Guidance
a. Reviews b. Inspections c. Walkthroughs d. Automatic techniques	Candidates are expected to understand various validation and verification techniques

2. Constructing a method

Learners will be able to:

2.1 Explain the idea of virtual machine underlying a design method.

Indicative content	Guidance
a. Simulation of computer systems b. Use as a systems design method	Candidates should be able to explain the concept of a virtual machine and how this underlies a systems design method by providing a simulation of a computer system from which a design for an actual computer system can be created.

2.2 Explain the idea of virtual machine underlying a design method.

Indicative content	Guidance
a. E.g.: i. SSADM method is based on the traditional waterfall model and structured techniques for systems modelling	Candidates should be able to explain the role played by various modelling techniques in modelling different aspects or elements of information systems.

2.3 xplain categories of methods.

Indicative content	Guidance
a. Categories of methods, e.g.: i. Structured development (e.g. SSADM) ii. Object-oriented development (e.g. RUP) iii. Agile development iv. RAD development v. Component-based development vi. 'Soft' methods (e.g. SSM)	Candidates should understand the similarities and differences between different categories of methods and explain corresponding advantages and disadvantages.

3. Selecting a method

Learners will be able to:

3.1 Describe technical factors of matching a method to an application.

Indicative content	Guidance
a. Suitability of the method for use with existing software development environment b. Life cycle coverage c. Interfacing with other methods d. Tool support e. Comparison frameworks e.g. NIMSAD	Candidates should understand various technical factors suitable for method selection. They should be able to match a method to a given application and project and understand the reasons for comparing methods. They are also expected to understand the importance of comparison frameworks.

3.2 Explain non-technical factors suitable for selecting a method.

Indicative content	Guidance
a. How widely used is the method b. Documentation and training c. Availability of staff d. How is the method supported e. Standardisation f. Track record	Candidates should understand various non-technical factors suitable for method selection. They should be able to match a method to a given application and project.

4. Introducing a method

Learners will be able to:

4.1 Explain how to carry out and evaluate a pilot.

Indicative content	Guidance
a. The role of a pilot project when introducing a new method b. Motivating staff who will be using the method c. Role of consultants d. Education and training	Candidates will be expected to explain how to select a project for the pilot and how to gather data and user feedback for evaluating it.

4.2 Explain reverse engineering.

Indicative content	Guidance
a. Reverse engineering of existing systems to fit in with the new method	Candidates will be expected to explain reverse engineering and related concepts and to explain how existing systems can be reverse engineered to fit in with the new method.

4.3 Describe potential obstacles.

Indicative content	Guidance
a. Discussion of what can go wrong when introducing a new systems design method	Candidates should demonstrate a good understanding of what can go wrong when introducing a new systems design method e.g. increased project time or cost.

5. Evaluating and tuning

Learners will be able to:

5.1 Explain the statistical process control.

Indicative content	Guidance
a. Statistical process control as applied to the software development process b. Appropriate software metrics, e.g.: i. Strengths ii. Any dangers inherent in their use	Candidates should demonstrate a good understanding of software metrics.

5.2 Describe the use of metrics in the improvement of software development process.

Indicative content	Guidance
a. Metrics that can be used to measure productivity/quality of the software development process, e.g. the time taken to develop a given unit of code	Candidates should demonstrate a good understanding of software metrics suitable for software process improvement.

5.3 Describe the relationship between structured and object-oriented methods and software quality assurance.

Indicative content	Guidance
a. Checking design artefacts produced via systems design methods	Candidates should have a sufficient understanding of software quality assurance activities and techniques and be able to 'allocate' them to different stages or phases of a system development method.

5.4 Explain how to assess the benefits of introducing a new method.

Indicative content	Guidance
a. Methods for assessing the benefits of introducing a new systems design method, e.g.: i. Time ii. Cost iii. Quality	Candidates should demonstrate a good understanding of measuring the productivity, cost and quality benefits of introducing a new systems design method, such as reduced project timescales and costs, improved quality, and fewer implementation and maintenance problems.

Examination Format

This module is assessed through completion of an invigilated written exam.

Type	Three written questions from a choice of five, each with equal marks
Duration	Three hours
Supervised	Yes
Open Book	No (no materials can be taken into the examination room)
Passmark	10/25 (40%)
Delivery	Paper format only

Adjustments and/or additional time can be requested in line with the BCS reasonable adjustments policy for candidates with a disability or other special considerations.

Question Weighting

Candidates will choose three questions from a choice of five. All questions are equally weighted and worth 25 marks.

Recommended Reading

Primary texts

Title: Information Systems Development: Methodologies, Techniques & Tools (fourth edition)
Author: D. Avison and G. Fitzgerald
Publisher: McGraw-Hill
Date: 2006
ISBN: ISBN: 978-0077114176

Title: Software Engineering (tenth edition)
Author: I. Somerville
Publisher: Addison Wesley
Date: 2015
ISBN: 978-0133943030

Title: Software Engineering: A Practitioner's Approach (seventh edition)
Author: R. S. Pressman and D. Ince
Publisher: McGraw Hill
Date: 2009
ISBN: 978-0071267823

Title: Managing the Software Process (SEI) (seventh edition)
Author: W. S. Humphrey
Publisher: Addison Wesley
Date: 1989
ISBN: 978-0201180954

Additional Texts

Title:	Software Design (third edition)
Author:	D. Budgen
Publisher:	Addison-Wesley
Date:	2003
ISBN:	978-201722192

Using BCS Books

Accredited Training Organisations may include excerpts from BCS books in the course materials. If you wish to use excerpts from the books you will need a license from BCS. To request a license, please contact the Head of Publishing at BCS outlining the material you wish to copy and its intended use.

Document Change History

Any changes made to the syllabus shall be clearly documented with a change history log. This shall include the latest version number, date of the amendment and changes made. The purpose is to identify quickly what changes have been made.

Version Number	Changes Made
Version 1.0 **July 2021**	Document Creation

CONTACT

For further information please contact:

BCS
The Chartered Institute for IT
3 Newbridge Square
Swindon
SN1 1BY

T +44 (0)1793 417 445

www.bcs.org