ISSN: 2473-4901
v11 n6454

A Topical Examination of the Systems Analysis & Design Course

Neelima Bhatnagar
bhatnagr@pitt.edu
Information Sciences Department
University of Pittsburgh
Greensburg, PA 15601 USA

Kevin J. Slonka
kslonka@francis.edu
Computer Science & Cyber Security Department
Saint Francis University
Loretto, PA 15940 USA

Abstract

The systems analysis & design (SAD) course is a critical course in many majors, specifically Information Systems. Many textbooks exist that are focused on this course, but the content varies wildly between them. This research analyzed major textbooks and job postings focused on systems analysis and design and, by using both quantitative and qualitative analyses, arrived at a final set of 15 themes that should be present in a proper SAD course. Implications are drawn and the groundwork for future studies are laid.

Keywords: systems analysis, design, textbook, competency, knowledge

A Topical Examination of the Systems Analysis & Design Course

Neelima Bhatnagar & Kevin J. Slonka

1. INTRODUCTION

The systems analysis & design (SAD) course continues to be a foundational course in the Information Systems discipline. Despite there being a plethora of textbooks focused on this specific topic, the topics discussed within each vary wildly. For this reason, as well as the related job postings requiring such diverse skills, it is crucial that the SAD course teach the foundational concepts of the discipline but also remain current by teaching the skills required to attain a job in today's market.

The SAD course is a major requirement in many majors across the country, but specifically in the Management Information Systems (MIS) major at an R1 regional campus in Western Pennsylvania. This course often sees enrollments from other majors as well (such as IT). This initial study focuses on analyzing the literature to deduce the proper topics that should be taught in a SAD course, which leads to the singular research question:

RQ 1: What are the core knowledge areas that should be present in a Systems Analysis & Design (SAD) course?

2. THE SAD COURSE CONCEPT

According to the 2020 Information Systems Competency Model published by ACM (Leidig & Salmela, 2021), systems analysis and design remains one of the core competency areas of the Information Systems degree. The model identified eight competencies that every graduate should comprehend in the area of SAD:

1. Explain what systems are and how they are developed
2. Demonstrate the SDLC phases and activities
3. Identify SDLC Models (Agile, Waterfall, V-shaped, iterative, spiral, etc.)
4. Work effectively in a team environment
5. Describe data modeling techniques
6. Describe the role and responsibilities of the participants in the SDLC
7. Explain the common ways projects fail and how to avoid these failures
8. Identify Enterprise Architecture concepts related to SDLC phases (p. 122)

While these competencies are sufficient for generating learning outcomes for a course, they lack the necessary detail to build an entire 15-week course. Additionally, the content of a SAD course "needs to be aligned with [...] the needs of the IS job market" (Leidig & Salmela, 2021, p. 35). The IS2020 competency model not only suggests the core courses for the major but puts the onus of curriculum design on the faculty (2021). They are in charge of ensuring that their course design exhibits the necessary competencies while remaining relevant to the current job market.

The idea of a discrete course in systems analysis has been a mainstay of the major for at least 50 years, with Couger (1972) suggesting the course be offered in a student's senior year. His objectives (competencies) and description are strikingly similar to how ACM currently describes the course:

> Objectives: To identify the decision requirements for the management of an organization. To analyze the design of an information gathering and processing system intended to facilitate decision making and planning and control. To analyze the concept of an information system. To review the approaches and techniques available to evaluate existing systems. To examine the concept of common data base for all functional modules.
>
> Description: Information requirements for an organization. Operational level systems. Tactical level systems. Strategic level systems. Planning for a comprehensive information system. Introduction to the system life cycle. System life cycle management. Basic analysis tools. Defining logical system requirements. Summary and introduction to physical system design (p. 143).

Beyond these core concepts, Siau et al. (2022) expand upon the SAD concepts across the eras. Their analysis brought to light various topics that may not be found in stereotypical SAD courses, such as DevOps, DevSecOps, IoT, Cloud, etc. Also identified is a pattern that SAD methodologies are moving toward higher levels of abstraction (i.e.,

being able to apply a typically-IS methodology to a non-IS field). This uncovers another consideration for the creation of a proper list of SAD topics: the principles are used not only in the IS field but also in other fields, such as medicine, environmental science, and international studies (Sharma et al., 2022; Proporov, 2022; Dutta et al., 2022).

Another topic that may not be present in many SAD courses is suggested by the Gartner research firm, which predicted that low/no-code toolkits will be used to create more than half of all new business applications (Crumbly & Field, 2020). Such an idea, while not impossible to include in an SAD course, is counter to many of the core competencies that assume a more standard development environment. Similarly, some have begun to implement another competency from the discipline that is not found in many SAD courses but is listed as a competency area in the IS2020 model, business analytics (Pomykalski, 2021). Beyond these topics, Topi & Spurrier (2022) argue that artificial intelligence should be added into the SAD course given its recent prominence in industry.

Possibly the largest trend that appears in literature and industry is the shift from waterfall-style methods to agile methods (Siau et al., 2022; Sharp & Lang, 2021; Kakar, 2023). Despite the seminal work on Agile being published more than 20 years ago (Beck et al., 2001), the majority of the literature on teaching agile methods has only been published since 2016. These publications can be categorized in three ways:

1. use of Agile as a pedagogical approach to teach non-Agile content
2. use of Agile as a pedagogical approach to teach Agile content, and
3. use of non-Agile pedagogical approaches to teach Agile content (Sharp et al., 2020).

Despite the majority of the literature suggesting the use of non-agile approaches, Hassan (2023) suggests doing away with the traditional approach altogether (only mentioning it as a background) and focusing solely on agile. This is in stark contrast to the framework proposed by Topi & Spurrier (2019) that suggests a hybrid approach, teaching agile methods along with the core plan-driven approaches that are paramount to any method. With all of the disagreement on the necessary competencies/topics for a SAD course research must be conducted to fill the gap.

3. METHODOLOGY & RESULTS

A mixed methods approach was used to conduct this study. First, systems analysis and design textbooks from major publishers were identified. This resulted in seven textbooks being utilized for this research. These are listed in Table 1.

Title	Publisher
Systems Analysis and Design	Cengage
Systems Analysis and Design in an Age of Options	Prospect Press
Systems Analysis and Design	MyEducator
Systems Analysis and Design: An Object-Oriented Approach	Wiley
Systems Analysis and Design	Wiley
Modern Systems Analysis and Design	Pearson
Systems Analysis and Design	Pearson

Table 1: List of Textbooks

Next, for each textbook, a spreadsheet was created that listed the chapter titles. Based on the chapter titles, descriptive themes were inductively coded. This was done three times for accuracy purposes, once by each researcher and once together. A total of 28 keywords were recorded. A frequency count of the top twelve keywords, representing the initial list of topics to be taught, is presented in Table 2.

Theme	Freq.
UX/UI	11
Requirements	10
Data Design	9
Overview	9
Operations & Maintenance	8
Modeling	7
Planning	7
Project Management	5
Architecture	4
UML	4
Agile	3
SDLC	3

Table 2: Initial Themes

The remaining keywords had frequency counts as follows: code design, design, and estimation had a frequency count of two; activities, construction, distributed systems, documentation, job role, patterns, quality, resourcing, security,

ISSN: 2473-4901
v11 n6454

specifications, transition, use cases, and use stories had a frequency count of one.

Further analysis of the thematic data was conducted by analyzing an entire year's worth of job postings related to systems analysis and design, which were downloaded into spreadsheet files using an online API service for exporting jobs from Indeed.com. A total of 3826 rows of data were captured. The fields included in the data are listed in Table 3.

Field	Field
Job Title	Hiring Manager (full name, first name, last name, role, LinkedIn URL)
URL	Company Name
Posted-date	Company URL
Country code	Company LinkedIn URL
Remote (Y/N)	Company Industry
Job Location	Company Employee Count
Job Description	Company Revenue (USD)
Salary	Company Description

Table 3: Jobs Data

The Job Title and Job Description fields were qualitatively analyzed using NVivo. In order to conduct an analysis using NVivo, the 13 files were consolidated into a single file that contained only job titles and descriptions. Additional fields captured were not included. A total of 22,432 codes and 310,514 references were generated by using the auto-coding feature, which analyzes the data to provide broad categorizations. A hierarchy chart of the themes was created from those categorizations in order to determine the major themes from the job posting data, which is presented at a high level in Figure 1 simply to show that a total of 18 themes emerged. These include project management, experience, skills, process, data, management, requirements, solutions, business, work, status, customer, communication, teams, plans, systems, development, and product.

Some of these themes were discarded, representing the standard language of job postings. The remaining themes were further investigated to see whether these matched the initial set of keywords produced in Table 2 or if any new themes emerged that should be included in a 15-week course.

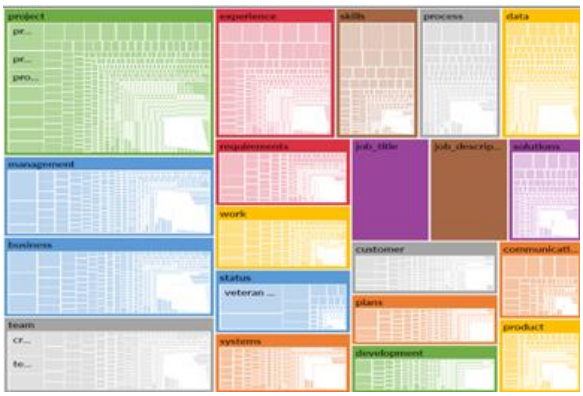

Figure 1: NVivo Results

4. DISCUSSION

The job descriptions analysis revealed that the project management, process, data, management, requirements, business, plans, systems, development, and product themes are addressed by the keywords listed in Table 2, and thus are not new.

Four of the job description themes were related to the job advertisements specifically: experience, skills, work, and status. Additionally, a communication theme was found but did not reveal any further information distinct from what would already be taught when teaching existing themes.

However, there are three new themes found in the job analysis that should be added to the original list. These include solutions, which typically occurs at the end of the systems development process by combining multiple items together with the newly designed system to provide a complete deliverable, the customer theme, which focuses on both internal and external customers who would have a stake in the newly designed system, and teams, which focuses on the people aspect of developing the solution in greater detail than the Project Management theme.

Thus, the final list of topics/keywords/themes that should be covered in an undergraduate systems analysis and design course are presented in Table 4.

Theme	
UX/UI	
Requirements	
Data Design	
Overview	
Operations & Maintenance	
Modeling	
Planning	
Project Management	
Architecture	
UML	
Agile	
SDLC	
Solutions	*New*
Customer	*New*
Teams	*New*

Table 4: Final Topics List

Based on the IS model curriculum of 1997, 2002, 2010, and 2020 (see Appendix A for a summary of the scope and topics; IS 2020 does not list topics, but rather competencies as learning objectives) many of the topics are still relevant and have not changed much.

5. LIMITATIONS & FUTURE RESEARCH

The selection of textbooks is one limitation of this research. Despite the researchers' best efforts to scour the Internet, publishers, retail outlets, and literature reviews to find all current textbooks on the topic of SAD it is still possible that some were not located. Another limitation, or perhaps better stated, design choice, is that this research focused on the SAD course through the lens of an IS degree. SAD courses can be found in business schools, engineering schools, etc. The goals of the researchers may not align with the perspective of the reader.

Future research in this area will be undertaken to determine a sufficient set of learning activities for the SAD course. Now that a comprehensive set of topics has been established, future studies can not only investigate learning activities, expanding on the work of Hsbollah & Hassan (2022) but also develop case studies and longitudinal assessments of this research's effectiveness.

6. REFERENCES

Beck et al. (2001). The agile manifesto. Agile Alliance.

Couger, J. D. (1972). The undergraduate program in information systems development. *ACM '72 Proceedings of the ACM annual conference – Volume 1*, 138-144. https://doi.org/10.1145/800193.805831

Crumbly, J. & Field, D. (2020). Implementing no/low-code development in a systems analysis and design course. *Journal of Strategic Innovation and Sustainability, 15*(5), 116-130.

Davis, G. B., Gorgone, J. T., Couger, J. D., Feinstein, D. L., & Longenecker, H. E. (1997). IS '97. *ACM SIGMIS Database*, *28*(1), 101–194. https://doi.org/10.1145/511965.512222

Dutta, S. J., Knafo, S., & Lovering, I. A. (2022). Neoliberal failures and the managerial takeover of governance. *Review of International Studies, 48*(3), 484-502. https://doi.org/10.1017/S0260210521000619

Gorgone, J., Davis, G., Valacich, J., Topi, H., Feinstein, D., & Longenecker, H. (2003). IS 2002 Model Curriculum and Guidelines for Undergraduate Degree Programs in Information Systems. Communications of the Association for Information Systems, 11, pp-pp. https://doi.org/10.17705/1CAIS.01101

Hassan, N. (2023). Navigating the tensions between traditional and agile development approaches in systems analysis and design courses. *MWAIS 2023 Proceedings*, 1-9.

Hsbollah, H. M. & Hassan, H. (2022). Creating meaningful learning experiences with active, fun, and technology elements in the problem-based learning approach and its implications. *Malaysian Journal of Learning and Instruction, 19*(1), 147-181. https://doi.org/10.32890/mjli2022.19.1.6

Kakar, A. K. (2023). A rhetorical analysis of the agile manifesto on its 20th anniversary. *The Journal of the Southern Association for Information Systems, 10*(1), 1-11. https://doi.org/10.17705/3JSIS.00030

Leidig, P. & Salmela, H. (2021). *IS2020 A competency model for undergraduate programs in information systems.* Association for Computing Machinery. https://www.acm.org/binaries/content/assets/education/curricula-recommendations/is2020.pdf

ISSN: 2473-4901
v11 n6454

Pomykalski, J. J. (2021). Moving to business analytics Re-designing a traditional systems analysis and design course. *Information Systems Education Journal, 19*(6), 55-67.

Priporov, I. E. (2022). System research on the preparation of sunflower seed cake using multimedia devices. *IOP Conference Series. Earth and Environmental Science, 954*(1), 1-8. https://doi.org/10.1088/1755-1315/954/1/012064

Sharma, A., Singh, B., Ethakota, J., Kulkarni, S. P., Rathore, R., & Kulkarni, P. S. (2022). Advanced medical research to control neurological disorder: A systematic analysis. *NeoruQuantology*, *20*(12), 151-160.

Sharp, J., Mitchell, A., & Lang, G. (2020). Agile teaching and learning in information systems education: An analysis and categorization of literature. *Journal of Information Systems Education, 31*(4), 269-281.

Sharp, J. & Lang, G. (2021). IS 2020 and a snapshot of the current state of systems analysis and design courses. *Proceedings of the 2021 AIS SIGED International Conference on Information Systems Education and Research*, 1-19.

Siau, K., Woo, C., Storey, V. C., Chiang, R. H. L., & Chua, C. E. H. (2022). Information systems analysis and design: Past revolutions, present challenges, and future research directions. *Communications of the Association for Information Systems, 50*, 835-856. https://doi.org/10.17705/1CAIS.05037

Topi, H., Valacich, J. S., Wright, R. T., Kaiser, K., Nunamaker, Jr., J. F., Sipior, J. C., & de Vreede, G. (2010). IS 2010: Curriculum Guidelines for Undergraduate Degree Programs in Information Systems. Communications of the Association for Information Systems, 26, pp-pp. https://doi.org/10.17705/1CAIS.02618

Topi, H. & Spurrier G. (2019). A generalized, enterprise-level systems Development process framework for systems Analysis and design education. *Journal of Information Systems Education, 30*(4), 253-265.

Topi, H. & Spurrier, G. (2022). Addressing AI-based capabilities in systems analysis and design: A pedagogical perspective. *Proceedings of the 2022 AIS SIGED International Conference on Information Systems Education and Research*, 1-13.

ISSN: 2473-4901
v11 n6454

APPENDIX A

		SCOPE	TOPICS
IS 1997	IS'97.7 Analysis and Logical Design	This course provides an understanding of the system development and modification process. It enables students to evaluate and choose a system development methodology. It emphasizes the factors for effective communication and integration with users and user systems. It encourages interpersonal skill development with clients, users, team members, and others associated with development, operation and maintenance of the system. Object oriented analysis and design. Use of data modeling tools. Development and adherence to life cycle standards.	Life cycle phases: requirements determination, logical design, physical design, test planning, implementation planning, and performance evaluation; communication, interpersonal skills, interviewing, presentation skills; group dynamics; risk and feasibility analysis; group-based approaches: project management, joint application development (JAD), structured walkthroughs; object oriented design; software production and reviews; prototyping; database design; software quality metrics; application categories; software package evaluation and acquisition; professional code of ethics.
IS 2002	IS 2002.7 Analysis and Logical Design	This course examines the system development and modification process. It emphasizes the factors for effective communication and integration with users and user systems. It encourages interpersonal skill development with clients, users, team members, and others associated with development, operation, and maintenance of the system. Structured and object oriented analysis and design, use of modeling tools, adherence to methodological life cycle and project management standards.	Life cycle phases: requirements determination, logical design, physical design, and implementation planning; interpersonal skills, interviewing, presentation skills; group dynamics; risk and feasibility analysis; group-based approaches: project management, joint application development (JAD), and structured walkthroughs; structured versus object oriented methodologies; RAD, prototyping; database design; software package evaluation, acquisition, and integration; global and inter-organizational issues and system integration; professional code of ethics.
IS 2010	IS 2010.6 Systems Analysis & Design	This course discusses the processes, methods, techniques and tools that organizations use	Identification of opportunities for IT-enabled organizational change

ISSN: 2473-4901
v11 n6454

		SCOPE	TOPICS
		to determine how they should conduct their business, with a particular focus on how computer-based technologies can most effectively contribute to the way business is organized. The course covers a systematic methodology for analyzing a business problem or opportunity, determining what role, if any, computer-based technologies can play in addressing the business need, articulating business requirements for the technology solution, specifying alternative approaches to acquiring the technology capabilities needed to address the business requirements, and specifying the requirements for the information systems solution in particular, in-house development, development from third-party providers, or purchased commercial-off-the-shelf (COTS) packages .	• Business process management • Analysis of business requirements o Business process modeling o Information requirements • Structuring of IT-based opportunities into projects • Project specification • Project prioritization • Analysis of project feasibility o Operational o Tangible costs and benefits (financial and other measures such as time savings) o Intangible costs and benefits such as good will, company image o Technical o Schedule o Legal o Cultural (organizational and ethnic) • Fundamentals of IS project management in the global context • Using globally distributed communication and collaboration platforms • Analysis and specification of system requirements o Data collection methods o Methods for structuring and communicating requirements o Factors affecting user experience o User interface design o System data requirements o Factors affecting security o Ethical considerations in requirements specification • Different approaches to implementing information systems to support business requirements o Packaged systems; enterprise systems o Outsourced development o In-house development • Specifying implementation alternatives for a specific system

ISSN: 2473-4901
v11 n6454

		SCOPE	TOPICS
			• Impact of implementation alternatives on system requirements specification • Methods for comparing systems implementation approaches • Organizational implementation of a new information system • Different approaches to systems analysis & design: structured SDLC, unified process/UML, agile methods
IS 2020	A3.4.1 Systems Analysis and Design	Examines various systems development methodologies and modeling tools with an emphasis on object-oriented systems development methods, software development life cycle (SDLC), and agile software development while emphasizing analytical techniques to develop the correct definition of business problems and user requirements. Topics should also include design, project management standards, information gathering, effective communication and interpersonal skill development.	1. Explain what systems are how they are developed. 2. Demonstrate the SDLC phases and activities 3. Identify SDLC Models (Agile, Waterfall, V-Sha[ed, iterative, spiral, etc.) 4. Work effectively in a team environment. 5. Describe data modeling techniques. 6. Describe the role and responsibilities of the participants in the SDLC. 7. Explain the common ways projects fail and how to avoid these failures. 8. Identify Enterprise Architecture concepts related to SDLC phases.