DOWNLOAD PDF ENGINEERING SYSTEMS DESIGN AND ANALYSIS (PD)

Chapter 1 : Systems Analysis and Design Tutorial (Software Engineering)

Design Synthesis System Analysis and Control (Balance) Chapter 1 Introduction to Systems Engineering 7 system product by showing how it is broken down.

Table of Contents The key to success in business is the ability to gather, organize, and interpret information. Systems analysis and design is a proven methodology that helps both large and small businesses reap the rewards of utilizing information to its full capacity. As a systems analyst, the person in the organization most involved with systems analysis and design, you will enjoy a rich career path that will enhance both your computer and interpersonal skills. Systems Analysis and Design SAD is an exciting, active field in which analysts continually learn new techniques and approaches to develop systems more effectively and efficiently. However, there is a core set of skills that all analysts need to know no matter what approach or methodology is used. All information systems projects move through the four phases of planning, analysis, design, and implementation; all projects require analysts to gather requirements, model the business needs, and create blueprints for how the system should be built; and all projects require an understanding of organizational behavior concepts like change management and team building. The major goal of systems analysis and design is to improve organizational systems. Often this process involves developing or acquiring application software and training employees to use it. Application software, also called a system, is designed to support a specific organizational function or process, such as inventory management, payroll, or market analysis. The goal of application software is to turn data into information. For example, software developed for the inventory department at a bookstore may keep track of the number of books in stock of the latest best seller. Software for the payroll department may keep track of the changing pay rates of employees. A variety of off-the-shelf application software can be purchased, including WordPerfect, Excel, and PowerPoint. However, off-the-shelf software may not fit the needs of a particular organization, and so the organization must develop its own product. Information systems analysis and design is a method used by companies ranging from IBM to PepsiCo to Sony to create and maintain information systems that perform basic business functions such as keeping track of customer names and addresses, processing orders, and paying employees. The main goal of systems analysis and design is to improve organizational systems, typically through applying software that can help employees accomplish key business tasks more easily and efficiently. As a systems analyst, you will be at the center of developing this software. The analysis and design of information systems are based on: This tutorial builds on our professional experience as systems analysts and on our experience in teaching systems analysis and design in the classroom. This tutorial will be of particular interest to instructors who have students do a major project as part of their course. Each chapter describes one part of the process, provides clear explanations on how to do it, gives a detailed example, and then has exercises for the students to practice. In this way, students can leave the course with experience that will form a rich foundation for further work as a systems analyst. Systems analysis and design is typically taught in one or two semesters. Our tutorial may be used in either situation. The text is appropriate for undergraduate junior or senior curricula at a four-year university, graduate school, or community college. The level and length of the course can be varied and supplemented by using real-world projects, HyperCase, or other materials available on the Instructor Resource Center. Chapters 1â€”3 stresses the basics that students need to know about what an analyst does and introduces the three main methodologies of the systems development life cycle SDLC , agile approaches, and object-oriented analysis with UML, along with reasons and situations for when to use them. These sections show how a variety of emerging information systems, including mobile and wireless technologies, and enterprise systems integrate IT and fit into organizations; how to determine whether a systems project is worthy of commitment; e-commerce project management; and how to manage a systems project using special software tools. The three roles of the systems analyst as consultant, supporting expert, and agent of change are introduced, and ethical issues and professional guidelines for serving as a systems consultant are covered. There is also material on virtual teams and virtual organizations, and the concept of HCI is introduced. The use of open source software OSS is also introduced. Chapter 2 includes how to initially approach an organization by drawing context-level data flow diagrams, using entity-relationship models, and developing use cases and use case scenarios. Chapter 3 introduces expanded material on creating the project charter and introduces writing the systems proposal early in the process, no matter what method of analysis and design has been chosen. Expanded coverage of evaluating software and hardware, and when to use COTS commercial off-the-shelf software , is included. This chapter teaches several methods for forecasting costs and benefits, which are necessary to the discussion of acquiring software and hardware. Chapter 3 helps students evaluate software by assessing trade-offs among creating custom software, purchasing commercial-off-the-shelf COTS software, or outsourcing to an application service provider ASP. Creating a problem definition and determining feasibility are also covered. Chapter 3 guides students in professionally writing and presenting an effective systems proposal, one that incorporates figures and graphs to communicate with users. Chapters 4â€"6 emphasizes the use of systematic and structured methodologies for performing information requirements analysis. Attention to analysis helps analysts ensure that they are addressing the correct problem before designing the system. Chapter 4 introduces a group of interactive methods, including interviewing, joint application design JAD , and constructing questionnaires. Chapter 5 introduces a group of unobtrusive methods for ascertaining information requirements of users. Chapter 6 on agile modeling and prototyping is innovative in its treatment of prototyping as another data-gathering technique that enables the analyst to solve the right problem by getting users involved from the start. Agile approaches have their roots in prototyping, so this chapter begins with prototyping to provide a proper context for understanding, and then takes up the agile approach. The values and principles, activities, resources, practices, processes, and tools associated with agile methodologies are presented. This chapter also includes material on rapid application development RAD for human information-requirements gathering and interface design. Chapters 7â€"10 details the analysis process. It builds on the previous two parts to move students into analysis of data flows as well as structured and semi-structured decisions. It provides step-by-step details on how to use structured techniques to draw data flow diagrams DFDs. Chapter 7 provides coverage of how to create child diagrams; how to develop both logical and physical data flow diagrams; and how to partition data flow diagrams. Chapter 8 features material on the data repository and vertical balancing of data flow diagrams. Chapter 8 also includes extensive coverage of extensible markup language XML and demonstrates how to use data dictionaries to create XML. Chapter 9 includes material on developing process specifications. A discussion of both logical and physical process specifications shows how to use process specifications for horizontal balancing. Chapter 9 also covers how to diagram structured decisions with the use of structured English, decision tables, and decision trees. In addition, the chapter covers how to choose an appropriate decision analysis method for analyzing structured decisions and creating process specifications. Push technologies are also introduced. This part concludes with Chapter 10 on object-oriented systems analysis and design. This chapter includes an in-depth section on using unified modeling language UML. Through several examples and Consulting Opportunities, this chapter demonstrates how to use an object-oriented approach. Consulting Opportunities, diagrams, and problems enable students to learn and use UML to model systems from an object-oriented perspective. Students learn the appropriate situations for using an object-oriented approach. This chapter helps students to decide whether to use the SDLC, the agile approach, or object-oriented systems analysis and design to develop a system. Chapters 11â€"14 covers the essentials of design. It begins with designing output, because many practitioners believe systems to be output driven. The design of Web-based forms is covered in detail. Particular attention is paid to relating output method to content, the effect of output on users, and designing good forms and screens. Chapter 11 compares advantages and disadvantages of output, including Web displays, audio, DVD, and electronic output such as email and RSS feeds. Designing a Web site for e-commerce purposes is emphasized, and output production and XML is covered. Chapter 12 includes innovative material on designing Web-based input forms, as well as other electronic forms design. Also included is computer-assisted forms design. Chapter 12 features in-depth coverage of Web site design, including guidelines on when designers should add video, audio, and animation to Web site designs. The chapter also covers uses of Web push and pull technologies for output designs. There is detailed consideration of how to create effective graphics for corporate Web sites and ways to design effective onscreen navigation for Web site users. Coverage of intranet and extranet page design is also included. Consideration of database integrity constraints has been included as well, in addition to how the user interacts with the computer and how to design an appropriate interface. The importance of user feedback is also found in these topics. How to design accurate data entry procedures that take full advantage of computer and human capabilities to assure entry of quality data is emphasized here. Students are shown the relevance of database design for the overall usefulness of the system, and how users actually use databases. It introduces HCI, discussing its importance in designing systems that suit individuals and assisting them in achieving personal and organizational goals through their use of information technology. The concepts of usability, fit, perceived usefulness, and perceived ease of use are introduced, as is the Technology Acceptance Model TAM , so that systems students can knowledgeably incorporate HCI practices into their designs. Chapter 14 also features material on designing easy onscreen navigation for Web site visitors. The chapter presents innovative approaches to searching on the Web, highlights material on GUI design, and provides innovative approaches to designing dialogs. Chapter 14 articulates specialized design considerations for ecommerce Web sites. Mashups, new applications created by combining two or more Web-based application programming interfaces, are also introduced. Chapter 14 also includes extensive coverage on how to formulate queries, all within the framework of HCI. Chapters 15 and 16 concludes the tutorial. Chapter 15 focuses on designing accurate data entry procedures and includes material on managing the supply chain through the effective design of business-to-business B2B ecommerce. Chapter 16 emphasizes taking a total quality approach to improving software design and maintenance. In addition, material on system security and firewalls is included. Testing, auditing, and maintenance of systems are discussed in the context of total quality management. This chapter helps students understand how service-oriented architecture SOA and cloud computing are changing the nature of information systems design. In addition, students learn how to design appropriate training programs for users of the new system, how to recognize the differences among physical conversion strategies, and how to be able to recommend an appropriate one to a client. Chapter 16 also presents techniques for modeling networks, which can be done with popular tools such as Microsoft Visio. Material on security and privacy in relation to designing ecommerce applications is included. Coverage of security, specifically firewalls, gateways, public key infrastructure PKI , secure electronic translation SET , secure socket layering SSL , virus protection software, URL filtering products, email filtering products, and virtual private networks VPN , is included. Additionally, current topics of interest to designers of ecommerce applications, including the development and posting of corporate privacy policies, are covered. Important coverage of how the analyst can promote and then monitor a corporate Web site is included in this section, which features Web activity monitoring, Web site promotion, Web traffic analysis, and audience profiling to ensure the effectiveness of new ecommerce systems. Techniques for evaluating the completed information systems project are covered systematically as well.

Chapter 2 : Systems Engineering and Analysis - SEBoK

Design must consider not only the technical aspects of the system, as reflected by the traditional engineering disciplines with their concerns about materials and the forces of nature, but also.

Model-based systems engineering A graphical representation relates the various subsystems or parts of a system through functions, data, or interfaces. Any or each of the above methods are used in an industry based on its requirements. For instance, the N2 chart may be used where interfaces between systems is important. Part of the design phase is to create structural and behavioral models of the system. Once the requirements are understood, it is now the responsibility of a systems engineer to refine them, and to determine, along with other engineers, the best technology for a job. At this point starting with a trade study, systems engineering encourages the use of weighted choices to determine the best option. A decision matrix , or Pugh method, is one way QFD is another to make this choice while considering all criteria that are important. The trade study in turn informs the design, which again affects graphic representations of the system without changing the requirements. In an SE process, this stage represents the iterative step that is carried out until a feasible solution is found. A decision matrix is often populated using techniques such as statistical analysis, reliability analysis, system dynamics feedback control , and optimization methods. Other tools[edit] Systems Modeling Language SysML , a modeling language used for systems engineering applications, supports the specification, analysis, design, verification and validation of a broad range of complex systems. The following areas have contributed to the development of systems engineering as a distinct entity: Cognitive systems engineering Cognitive systems engineering CSE is a specific approach to the description and analysis of human-machine systems or sociotechnical systems. CSE has since its beginning become a recognized scientific discipline, sometimes also referred to as cognitive engineering. The concept of a Joint Cognitive System JCS has in particular become widely used as a way of understanding how complex socio-technical systems can be described with varying degrees of resolution. The more than 20 years of experience with CSE has been described extensively. Control engineering Control engineering and its design and implementation of control systems , used extensively in nearly every industry, is a large sub-field of systems engineering. The cruise control on an automobile and the guidance system for a ballistic missile are two examples. Control systems theory is an active field of applied mathematics involving the investigation of solution spaces and the development of new methods for the analysis of the control process. Industrial engineering Industrial engineering is a branch of engineering that concerns the development, improvement, implementation and evaluation of integrated systems of people, money, knowledge, information, equipment, energy, material and process. Industrial engineering draws upon the principles and methods of engineering analysis and synthesis, as well as mathematical, physical and social sciences together with the principles and methods of engineering analysis and design to specify, predict, and evaluate results obtained from such systems. Interface design Interface design and its specification are concerned with assuring that the pieces of a system connect and inter-operate with other parts of the system and with external systems as necessary. Interface design also includes assuring that system interfaces be able to accept new features, including mechanical, electrical and logical interfaces, including reserved wires, plug-space, command codes and bits in communication protocols. This is known as extensibility. Systems engineering principles are applied in the design of network protocols for local-area networks and wide-area networks. Mechatronic engineering Mechatronic engineering , like systems engineering, is a multidisciplinary field of engineering that uses dynamical systems modeling to express tangible constructs. In that regard it is almost indistinguishable from Systems Engineering, but what sets it apart is the focus on smaller details rather than larger generalizations and relationships. As such, both fields are distinguished by the scope of their projects rather than the methodology of their practice. Operations research Operations research supports systems engineering. The tools of operations research are used in systems analysis, decision making, and trade studies. Several schools teach SE courses within the operations research or industrial engineering department, [25] highlighting the role systems engineering plays in complex projects. Operations research , briefly, is concerned with the optimization of a process under multiple constraints. Performance is usually defined as the speed with which a certain operation is executed, or the capability of executing a number of such operations in a unit of time. Performance may be degraded when operations queued to execute is throttled by limited system capacity. For example, the performance of a packet-switched network is characterized by the end-to-end packet transit delay, or the number of packets switched in an hour. The design of high-performance systems uses analytical or simulation modeling, whereas the delivery of high-performance implementation involves thorough performance testing. Performance engineering relies heavily on statistics , queueing theory and probability theory for its tools and processes. Program management and project management Program management or programme management has many similarities with systems engineering, but has broader-based origins than the engineering ones of systems engineering. Project management is also closely related to both program management and systems engineering. Proposal engineering Proposal engineering is the application of scientific and mathematical principles to design, construct, and operate a cost-effective proposal development system. Basically, proposal engineering uses the " systems engineering process " to create a cost effective proposal and increase the odds of a successful proposal. Reliability engineering Reliability engineering is the discipline of ensuring a system meets customer expectations for reliability throughout its life; i. Next to prediction of failure, it is just as much about prevention of failure. Reliability engineering applies to all aspects of the system. It is closely associated with maintainability , availability dependability or RAMS preferred by some , and logistics engineering. Reliability engineering is always a critical component of safety engineering, as in failure modes and effects analysis FMEA and hazard fault tree analysis, and of security engineering. Risk Management Risk Management , the practice of assessing and dealing with risk is one of the interdisciplinary parts of Systems Engineering. In development, acquisition, or operational activities, the inclusion of risk in tradeoff with cost, schedule, and performance features, involves the iterative complex configuration management of traceability and evaluation to the scheduling and requirements management across domains and for the system lifecycle that requires the interdisciplinary technical approach of systems engineering. Systems Engineering has Risk Management define, tailor, implement, and monitor a structured process for risk management which is integrated to the overall effort. The "System Safety Engineering" function helps to identify "safety hazards" in emerging designs, and may assist with techniques to "mitigate" the effects of potentially hazardous conditions that cannot be designed out of systems. Scheduling Scheduling is one of the systems engineering support tools as a practice and item in assessing interdisciplinary concerns under configuration management. In particular the direct relationship of resources, performance features, and risk to duration of a task or the dependency links among tasks and impacts across the system lifecycle are systems engineering concerns. Security engineering Security engineering can be viewed as an interdisciplinary field that integrates the community of practice for control systems design, reliability, safety and systems engineering. It may involve such sub-specialties as authentication of system users, system targets and others: Software engineering From its beginnings, software engineering has helped shape modern systems engineering practice. The techniques used in the handling of the complexities of large software-intensive systems have had a major effect on the shaping and reshaping of the tools, methods and processes of Software Engineering.

Chapter 3 : Systems engineering - Wikipedia

solution to the design problem is a system having specified properties (able to launch a snowball feet), whereas the solution to the analysis problem consisted of the properties of a given system (the height of the snowball).

Page 4 Share Cite Suggested Citation: The National Academies Press. The essential products of the systems engineering process and their programmatic use are described in this section. The purpose is to organize information and knowledge to assist those who manage, direct, and control the planning, development, and operation of the systems necessary to accomplish the mission Sage, The systems engineering process should be conducted in a way that includes consideration of alternative system configurations. The result should be a set of traceable requirements that may be used in design and procurement and in system verification and validation, a baseline description of the physical system, and a baseline description of the operational concept. This should also include a set of documented interfaces to ensure compatibility of different parts of the system as they are developed. Several terms used in systems engineering are defined below for the convenience of the reader. System verification is a two-step process to assure, first, that system design successfully captures the full set of system requirements, and second, that the system hardware and software fully implement the design. System validation is the process of assuring that, once the system is developed, its operational concept will meet the original system requirements Sage, Baseline descriptions, both of design of the physical system and of the functions the system is supposed to perform, once built, are essential to the process of modifying the system as new information or experience is obtained. Configuration management and change control are important quality assurance steps that ensure changes to the baseline occur in a planned manner and are thoroughly documented, so that implications for system performance are understood. The direction of desirable changes is specified through configuration control Sage, The systems engineering process provides value to the development, management, and implementation of a large program by ensuring: This value of the systems engineering process may be realized in a number of ways, including: From Westinghouse Hanford Company, a Figure 3. Department of Energy, a Figure 2. Although not necessarily performed for the purpose of reducing program costs, a sound systems engineering approach improves the ability of managers of large engineering programs to deliver a sound design and operational concept with reduced risk of cost growth. A sound systems engineering approach, appropriately implemented, should result in effective integration of environmental remediation and waste management efforts across the entire Hanford Site. The hallmarks of a well-integrated program are consistency of approach throughout an organization, and a smooth flow of information both up and down the management chain. In such a program, work done by individual units is compatible with the objectives and goals of the larger organization, and individual projects are clearly related to the objectives of the organizational units in which they occur vertical integration. Within organizational layers, individual units are aware of the efforts of others in related domains and work to assure that their own activity complements that being done by other units horizontal integration.