

Systems Engineering: Optimizing Creation of Virtual Conversational Human Agents

CAPT Daniel P. Burns, USN(Ret)
Home Ports Solutions. LLC
Los Angeles, California
daniel.p.burns@homeportsolutions.net

Dan M. Davis & Julianne Nordhagen
University of Southern California
Los Angeles, California
dmdavis@acm.org & nordhage@usc.edu

ABSTRACT

Many defense organizations report pressure to replace or augment scarce, expensive, and schedule-constrained professionals who are needed to counsel personnel. One method that has demonstrable capability and has found utility is the use of conversational, computer-directed agents. These can respond within appropriate times to a predetermined list of germane questions. These agents can be either animated or live (via a large number of video clips). Early research and development has focused on laboriously generating question lists, paraphrases, recording scripts, utterances, transcriptions, video edits, validations, tests, and updates. The administrative burden of all of this has begun to overshadow the scientific and technical research effort. As this virtual human capability becomes more acceptable and widely implemented, the skills of Systems Engineering are seen as potential facilitators for optimizing the production process. This paper addresses preliminary experience with enhancing an on-going project via Systems Engineering. These approaches should make the adoption of these agents more practical and economical. The authors discuss the use of the V-Model and other Systems Engineering tools in a way that should enable other researchers to understand what can be expected of such an approach to computer-moderated virtual humans and agents. These management tools and techniques provide a real opportunity in the DoD to manifest the early successes demonstrated in research settings. Optimizing just the transcription phase would save significant amounts of implementation time.

ABOUT THE AUTHORS

Daniel P. Burns is a lifelong Systems Engineer, first with the Active Duty Navy, then SAIC, and small business. He served as Naval Chair and Professor of Practice in Systems Engineering at the Naval Postgraduate School (NPS). Captain Burns served as the as the Military Associate Dean and as acting Dean of the Graduate School of Engineering and Applied Sciences at NPS. His research interests center on human and resource utilization in defense efforts. Currently he is participating in the creation of a new program for Air Force Officers who seek postgraduate degrees. Captain Burns received a BS degree from the U.S. Naval Academy, an MS from the Naval Postgraduate School and an MS from Southern Methodist University. He is currently enrolling Portland State University PhD. program.

Dan M. Davis is a consultant for the University of Southern California (USC), focusing on large-scale distributed DoD simulations. Pre-retirement, for a decade he was the Director of USC's JESPP project for JFCOM. As the Assistant Director of the Center for Advanced Computing Research at Caltech, he managed Synthetic Forces Express, bringing HPC to DoD simulations. Prior experience includes serving as a Director at the Maui High Performance Computing Center and as a Software Engineer at the Jet Propulsion Laboratory and Martin Marietta. He has served as the Chairman of the Coalition of Academic Supercomputing Centers and has taught at the undergraduate and graduate levels. As early as 1971, Dan was writing programs in FORTRAN on one of Seymour Cray's CDC 6500's. He saw duty in Vietnam as a USMC Cryptologist and retired as a Commander, Cryptologic Specialty, U.S.N.R. He received B.A. and J.D. degrees from the University of Colorado in Boulder.

Julianne M. Nordhagen is an Ensign in the U.S. Navy and is attached to the NROTC unit at the University of Southern California, where she is assigned to collaborate on research at the Institute of Creative Technologies (ICT). As an Industrial and Systems Engineering (ISE) undergraduate, Julianne cultivated the analytical and technical skills required for this project, including optimizing the performance of large scale systems and applying computational methods for scheduling and forecasting. She has professional experience using integrated planning software like SAP and NetSuite ERP. She was selected to be one of the mentors for ICT's MentorPal project for the Office of Naval Research. Next April, she will begin her Naval Aviator training at the Naval Air Technical Training Center in Pensacola Florida. Ensign Nordhagen earned an ISE B.S. degree from the University of Southern California.

Systems Engineering: Optimizing Creation of Conversational Virtual Human Agents

CAPT Daniel P. Burns, USN(Ret)

Home Ports Solutions, LLC

Los Angeles, California

daniel.p.burns@homeportsolutions.net

Dan M. Davis & Julianne Nordhagen

University of Southern California

Los Angeles, California

dmdavis@acm.org & nordhage@usc.edu

INTRODUCTION

This paper addresses the issues surrounding the use of Systems Engineering (SE) techniques, tools, and approaches to enhance the conduct of simulation research, specifically Virtual Human (VH) and computer agent implementations. It presents both an overview of SE disciplines and examples of how these have been employed in a program at USC's Institute for Creative Technologies (ICT). ICT has been funded to create a computer-hosted VH mentor series to address STEM careers. The paper has two over-arching goals: analyzing the utility of SE and providing sufficient direction to enable others to see how to apply similar approaches to their own research projects.

The paper is organized in five parts. The first will be a background discussion of the recent advances made in the capabilities of VH technologies. The second section will present a brief survey of System Engineers' tools. The third section will cover the MentorPal project at ICT. The fourth section will focus on analysis of the efforts to use SE for optimization in this environment and a discussion as to future implementation opportunities and efforts. Then discussion appears in a fifth section.

VIRTUAL HUMANS IN THE DEFENSE ENVIRONMENT

Declining budgets, increasing operations tempos, and under-staffed conditions continue to constrain the amount of time that experienced DoD personnel have available to train, advise, and mentor those who follow them (Gould, 2017). Those same conditions create an increased need for such counseling to maintain readiness, improve morale, and enhance retention (Payne & Huffman, 2005). The authors contend that a method for standardizing and disseminating such advice globally at any time would make such information transfer more beneficial.

A virtual conversation is a creation in which an agent is implemented, in this case based on real persons. It attempts to recreate the appearance, voice, feel, and interactions that a live human conversation would produce. Enabled by new technologies, including natural language processing, virtual humans, computer generated imagery (CGI), machine learning, and virtual learning, the uses of virtual reality are spreading across the DoD. With a specialization in VR programming, ICT is the home of myriad simulation projects: SimCoach, New Dimensions in Testimony (NDT), PAL3, MentorPal, and other generalized programs under learning sciences, medical VR, mixed reality, narratives, social stimulation, and virtual humans.

To those unfamiliar with its hurdles, creating a virtual human may seem as simple as remodeling a human using CGI; it turns out that it takes significant study and effort to implement a Virtual Conversation, and the process can consume considerable computing power. The essential elements that go into the creation of a virtual conversation with lifelike abilities include natural language processing, machine learning, VR, CGI, and social stimulation of humans by computer-generated interactions. Natural language processing (NLP), will be the main focus of this discussion, though the same argument concerning the limits of virtual humans can be made with several of the other components. Natural language processing is comprised of the decomposition of language to allow the computer to do useful communications (Chowdhury, 2003). Recent developments in NLP have made significant advances, including breaking down sentences into: parts-of-speech tags, chunks, entity tags, semantic roles, similar words, and the grammatical and semantic elements of a sentence that generate meaning (Collobert & Weston, 2008).

The presentation to the user can take many forms, as shown in Figures 1-3.



Figure 1 – Video of NDT speaker shown in 3-D holographic display

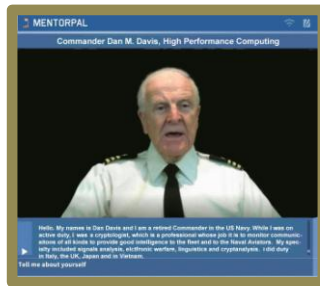


Figure 2 – ICT MentorPAL video clip presented on a 2-D monitor



Figure 3 – ICT Fully Animated CGI in SimCoach

At ICT, progress has been made in training and learning environments (Kenny et al., 2007), multi-party dialogues (Traum & Rickel, 2002), ethics and cooperation (Allwood, Traum, & Jokinen, 2000), health applications (Rizzo et al., 2011), and representation and reasoning (Swartout et al., 2006). Although automated speech recognition (ASR) is far from perfect, prime software makes Virtual Humans practicable. For instance, Google achieved nearly 75% exact recognition of phrases, with only a 16% word error rate (Kudryavtsev, 2017), and the technology will continue to improve. It is also important to note that many of the errors in ASR are caused by slurred speech, cultural slang, and context, stemming from inconsistent definitions and in variability of contexts (Lee, 1988). Although this causes problems when comparing these transcriptions with global data, models can be trained locally to a particular person's voice or a low resource language, using software such as CMU Sphinx from Carnegie Mellon University as has been applied to languages such as Arabic (Satori, Harti & Chenfour, 2007).

As it stands now, the primary issues with natural language processing include machine translation, precision, data storage, efficiency, and computation power; meeting these are the foci of this paper. All of these will continue to improve with time, as hardware, software, data storage, and ease of access are areas within which new research is emerging and seems directly applicable. Specifically, the speed and application of quantum computing will enable significant advances in NLP and its applications: one of which is the critical area of enhancing the realism of virtual conversations.

Within the military context, virtual learning environments provide useful mechanisms for initial training as well as lifelong training. An example of this can be found in PAL3. "The PAL3 system was designed to accompany a learner throughout their career and mentor them to build and maintain skills" (Swartout et al., 2016). Modern learning calls for new methods of information transfer.

The New Dimensions in Testimony project allows interactive conversation with a Holocaust survivor storyteller who has recorded a number of responses (Figure 4), including narratives of his experiences and thoughts (Traum et al., 2015). Advances in storytelling and gaming allow for utilization in learning, entertainment, healthcare, and lifelike training. The SimCoach system was developed to assist Post Traumatic Stress Disorder (PTSD) subjects and for "patients to enter therapy and to seek advice and information regarding their situation (Rizzo et al., 2011). These virtual systems have been shown to have more success, generating deeper levels of confidence than live professionals (Rizzo et al., 2011).



Figure 4 – Holocaust Survivor on Light Stage

SYSTEMS ENGINEERING

The Systems Engineer abstracts the term "system more broadly, seeing it as an amalgamation of staffing, processing, and producing a deliverable item for use. The standard DoD text on the subject, Systems Engineering Fundamentals defines a system as integrated composite of people, products, and processes (Lightsey, 2001). Further most Systems Engineers like to consider many systems in the context of their interrelations with other systems, which has led to the term System of Systems (Keating, 2003).

It may be useful here to consider the meanings of the term “Engineering,” as many in academia differentiate that term from “Science,” with most universities having a separate College or School for the two. Many argue that the fundamental nature of scientific research and the applied nature of engineering implementation are not dichotomous, but fall on two sides of a continuum that is artificially made more remote by their separation on campus. One author offers the visualization of this view, as is shown in Figure 5 (Putz, 2013).



Figure 5 - Engineering/Science Continuum

This paper’s definition of Systems Engineering is: “integrates all the disciplines and specialty groups into a team effort forming a structured development process that proceeds from concept to production to operation ... considers both the business and the technical needs of all customers with the goal of providing a quality product that meets the user needs (INCOSE, 2018).” The consensus is that SE uses a multi-disciplinary approach, applies engineering and management, fosters coherent and comprehensive design, and focuses on life-cycles of systems (IEEE, 1994).

Again, it may be instructive to consider what the DoD says on this topic. In a brief by the Director of Mission Assurance from the Office of the Deputy Assistant Secretary of Defense, the following three goals were listed: 1) To design, develop, construct, and operate complex systems; 2) To forecast their behavior under specific operating conditions; and 3) To deliver their intended function while addressing economic efficiency, environmental stewardship, and safety of life and property. The authors assert that these are goals that are very compatible with the goals of researchers engaged in their activities in any professional community.

A question that may be raised would be the accessibility of SE tools to the journeyman academic, government, or industry researcher. It should be noted that it is not the goal of this paper to create Systems Engineers of all of its readers, but to introduce them to tools they can easily use for their own projects.

A Short List of Systems Engineering Tools:

The authors feel the following could be useful in virtual conversation development and other research. They are offered as being easy to implement and analyze. The next section will look at “real life” implementations. The authors have found a useful on-line source of information by a British Group: Burge, Hughes and Walsh (BHW, 2018). Should the reader be familiar with these SE tools or should the reader wish to see which tools were actually implemented or considered for this virtual reality and conversationally enabled project, then skipping ahead to the MentorPal section below is suggested and then coming back when the reader needs more explication of the SE tool set. The selection of the tools was made by all three of the authors based on their experience and training in the Navy and in civilian academia.

The “V-Model” – grasping the life cycle:

This is one of SE’s best known tools. In application, its value lies in mapping the life cycle of the entire life of a project (Figure 6). The concept is based on the idea that there are varying stages to a project and that there is a balance between the left arm of the V and the testing of that stage will be balanced by the confirming activities on the right arm. Much of its value is that attention needs to be paid to each stage of a project. It also reminds participants that everything on the left arm will need to be shown on the right arm. This elicits two important questions: First, “Can we do what we are promising?” and Second, “Can we design a function on the right arm that will show we did just that?”.

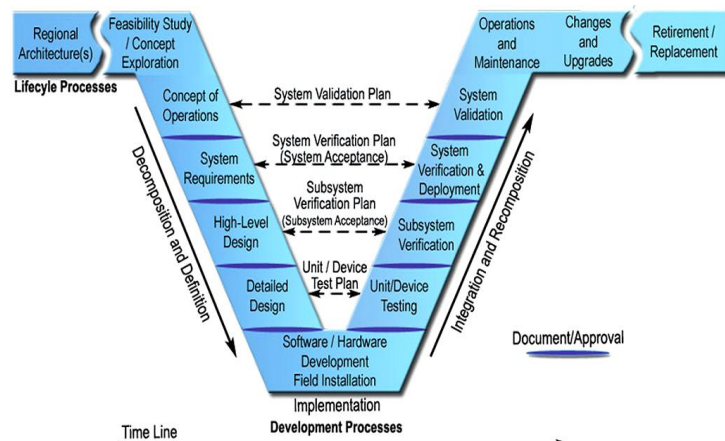


Figure 6 - Classic V-Model (FHWA, 2018)

Stakeholder Diagram – grouping interested parties into mutually focused groups:

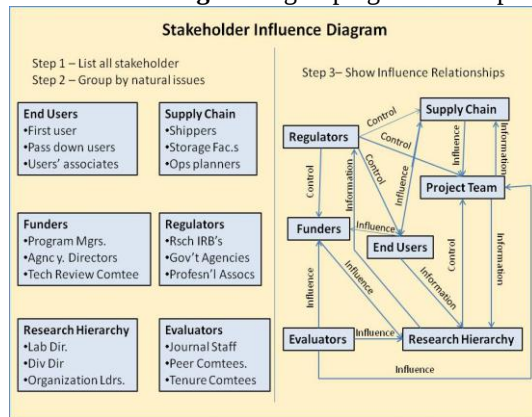


Figure 7 - Stakeholder Diagram

Most of the projects on which the authors have served were at least vaguely aware of the people who were interested in its outcome, the general needs of the end users, or the political entities who would want to be informed of its progress and success. Some of the more astute research managers may even have had an organized list in their minds of who needed to be satisfied, if not actually served, by the project. SE practitioners urge a more formal, but not onerous, process: the creation of an informal, but graphic, representation of such a list. This is often called a "Stakeholders Diagram." An example of such a diagram is included as Figure 7. As the creation of such is reasonably obvious, there will be no further description of steps required. Like taking notes before an exam that cannot go into the test room with the student, just the process of creation is often revelatory. This is best developed in team brain-storming session.

Articulating Requirements – translating customer desires into technical system requirements:

A critical part of any effort is understanding what is desired and what is required of the team. This mandates a rigorous and disciplined process for deciphering, clarifying, specifying, and documenting all requirements. Careful attention is required to prevent “mission creep.” Ambiguities and inconsistency must be resolved as soon as possible. Also unnecessary requirements must be recognized and removed. A careful reading of all controlling documents should be followed by a systematic listing of all requirements. These then should be reviewed by all of the functional groups within the effort, as a requirement that is well recognized may be unfamiliar to other groups. The reviewers should remember not only the end goal that is envisioned, but the specifications, contractual imperatives, and legal constraints that may drive the system design. Careful attention to these details may have significant impact on project success and in preventing cost over-runs. Getting the funder’s agreement on the completeness of the requirements list may be central to compensation adjustments if new requirements are imposed later.

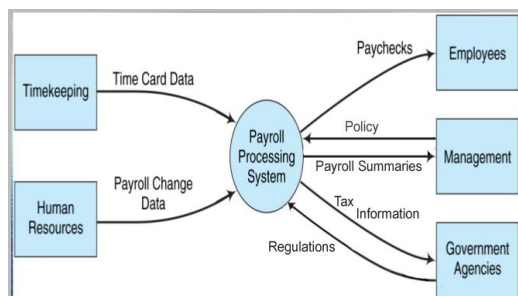


Figure 8 - Context Diagram (Slide Share, 2018)

Context Diagram – graphing the mutual impacts of external entities and the system:

This tool is designed to highlight how the central concept under consideration fits in with all of the rest of its environment. Its major use is in identifying both the boundaries of the concept and the necessary interfaces that will be required as it will be implemented. It is critical to know what needs to be done, what should not be done and in what ways to associate harmoniously with adjacent entities. An example featuring a payroll function is shown in Figure 8. It shows both the interfacing entities, but identifies the required interfaces that must be defined and implemented.

Pugh Matrix – plotting design criteria satisfaction against different design options:

Any time a new concept is being developed, there are a number of design issues that arise: which approach to use, what programming language to employ, what platform to implement, *etc.* Stuart Pugh of Strathclyde University is a mainstay of the Total Design methodology (Pugh, 1996) and he is credited with developing a matrix to display the relative strengths and weaknesses of various alternatives for each criterion, *e.g.* ease of use, cost, time, training, *etc.* Then each cell is evaluated, usually starting with a baseline concept which is variously scored as zero or “S”, then the alternatives are scored as either plus or minus in the basic instantiation. The matrix can have positive or negative values in more elaborate forms. Table 2 shows typical weighted scoring.

Table 1 - Typical Format of a Weighted Pugh Matrix

	Weight	Baseline	Alt A	Alt B	Alt C	Alt D
Criteria						
One	2	0	+2	-5	+4	+6
Two	1	0	-2	+3	-2	+4
Three	3	0	+3	+6	-1	-2
Plus Total	Weighted		+13	+21	+8	+16
Minus Total	Weighted		-2	-10	-5	-6
Total Score	Weighted		+11	+11	-3	+10

Table 1 - Typical Format of a Weighted Pugh Matrix

	Weight	Baseline	Alt A	Alt B	Alt C	Alt D
Criteria						
One	2	0	+2	-5	+4	+6
Two	1	0	-2	+3	-2	+4
Three	3	0	+3	+6	-1	-2
Plus Total	Weighted		+13	+21	+8	+16
Minus Total	Weighted		-2	-10	-5	-6
Total Score	Weighted		+11	+11	-3	+10

The Analytic Hierarchy Process – quantifying the decision-making process:

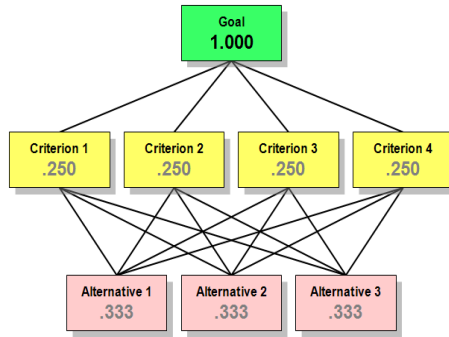


Figure 9 - AHP Chart (Tucker, 2017)

Another way to approach major decisions is the Analytic Hierarchy Process, commonly referred to as the AHP. It decomposes a single major challenge into many sub-areas for consideration, and then displays them as a hierarchy to better enable the decision makers to understand the values and interrelationships of the various components of the decision. It relies upon assigning numerical values to the various components to enforce a quantitative analysis of issues that are often subjective. It was developed in the latter part of the 20th Century by Professor Thomas Saaty of the University of Pittsburg. (Saaty, 1980) As well as the visual aid provided by the process, there have been significant quantitative analyses developed for its implementation. A simple hierarchy is provided in Figure 9, with hypothetical numerical values inserted.

System Requirements Model – putting requirements into a framework: Often referred to as Requirements Analysis, this is the process of identifying, characterizing, and articulating all of the requirements of the task ahead. It begins by dichotomizing Operational Requirements (the end goal and effort *raison d'être*) and the Functional Requirements (the individual functions needed to achieve that goal.) Then Functional Requirements are further defined by their constraints, referred to as Non-Functional Requirements (size, energy needs, etc.). The actual implementation of this process in any efforts has to be tailored to that effort, but it is essential that the process be rigorous and that it ferrets out any unidentified requirements and that it be revisited periodically to ascertain if any hitherto unidentified requirements have surfaced.

Design Structure Matrix – identifying subsystem connections:

A Design Structure Matrix (DSM) is an interesting way to get a grasp of how the various design elements interact. First, the interactions of the various entities are plotted on a flow chart to identify all connectivity. The classic way to view this in a coherent manner is to plot major sub-elements of the overall effort down the vertical axis, and then repeat the same along the horizontal axis. This produces a two dimensional graph like the one on the left, Figure 10 (DSM-Org, 2018). Then, the team goes through the matrix and, if the direction of the interaction is from the entity on the vertical axis to the entity on the horizontal axis, an “X” is placed in the intersecting cell in the bottom/left half, and if the flow is the other way, an “X” is placed in the cell in the upper/right half. Even just the identification of these interconnections is helpful on several fronts. It helps define when careful interface specifications are needed, it assists in creating a valid documentation of record for the design, and it is useful for future analyses. With this matrix, any behavior will not be seen *in vacuo*, but will be taken within the interface of all of the relevant subsystems.

		1	2	3	4	5	6	7	8	9	10	11	12	13
Customer Requirements	1			X	X	X								
Wheel Torque	2					X	X		X		X			X
Pedal Mech. Advantage	3					X		X		X		X	X	X
System Level Parameters	4			X	X		X		X	X	X	X	X	X
Rotor Diameter	5			X				X				X		X
ABS Modular Display	6													
Front Lining Coef. of Friction	7					X								X
Piston-Rear Size	8			X	X		X				X	X		X
Caliper Compliance	9						X							
Piston- Front Size	10			X	X		X	X	X			X		X
Rear Lining Coef of Friction	11					X								X
Booster - Max. Stroke	12													
Booster Reaction Ratio	13			X	X		X		X		X	X		

Figure 10 - DSM of Automobile Brakes

N² Analysis – understanding system interconnection problems:

While some authors see this analysis as synonymous with the DSM analyses method above, the authors of this paper come from a different community and see N² analyses as having another and equally critical impact on mission success, that of the scalability of any function when large numbers become involved. In all “big data” analyses, the fact of literally tens of trillions of pieces of data being involved create a real issue of memory capacity, processor power, and computational platform foundation, digital or quantum (Lucas, 2015). However, battlefield simulations work brings about another N² issue, that of the exponential burden if a huge number of entities not only exist, but must communicate with each other (Gottschalk, 2010). This analysis must be done on small models, then extended as best we can. The Gottschalk paper lays out the mind numbing numbers of transactions involved. The question posed to each team member on any project should be, “If the data or inputs grow in the future, will that impact performance? What kind of data growth is expected in the field?”. Note that the growth in entities which were originally coded to enable all entities to talk to all other entities, would have basically 10M entities with open communications with all the other entities, or 10 quadrillion paths.

Functional Modeling – establishing functionality and needed interconnections:

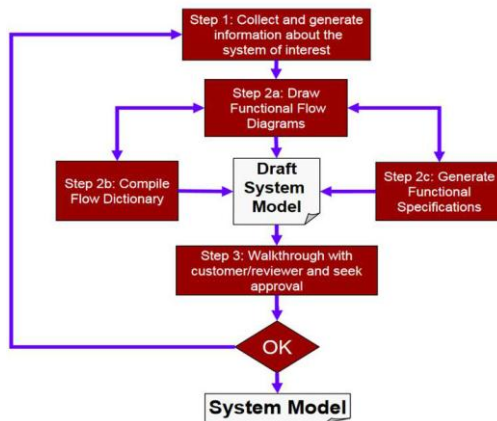


Figure 11 - Functional Modeling Process

This is one of the most critical methods in the SE toolbox. Unfortunately, some authors maintain that attempts to diagram any complex systems are more likely to create confusion and mislead the analysts into thinking they understand issues that are actually hidden from them under the clutter of the high-level representation. Some authors maintain that breaking the modeling into small subsystems analyses which they see as more illuminating. (Burge, 2011) The data visualization issue is an interesting one; many there are others interested in this area and are urging technical advances as potential solutions. (Amburn, 2016) in the interim, the Burge article cited above presents an approach with concomitant warnings to make sure the user can understand the relationship of the sub-system to the entire project. This approach is diagrammed in Figure 11.

Requirement Relationship Analysis – structuring the relationship of the requirements:

Relating the requirements to sponsors' needs and program managers' visions is a critical function that can be assisted by Systems Engineering tools. One researcher from Carnegie Melon University proposes a number of activities to help define the relationship of requirements. (Tran, 1999) He recommends a studied approach to understanding the following: Mission definition, Performance, Physical parameters, Use requirements, Anticipation of the use of the system, Operational deployment or distribution, Operational life cycle, Effectiveness factors, and Environment Definition. The authors agree with him on his focus and especially point out his mention of the Operational Life-Cycle.

Implementation Means Analysis – enumerating implementation alternatives

For every goal in the process, there will likely be a plethora of ways to accomplish that goal. Prior to analyzing the best way to pursue the achievement of those goals, it is considered useful to have a process for identifying those alternatives. Within the authors' experience, this is usually done by internal processes, but can be productively formalized by creating a spread sheet listing all of the subcomponent goals of the effort on the vertical axis of the spread sheet, then enumerating all of the conceivable means for accomplishing that goal horizontally across the page. This is another activity that may lend itself well to a group brain-storming session. The output of these creative and expansive sessions can then be filtered to respond to external and internal constraints. The remaining alternatives can then be analyzed and an optimized path to their implementation can be generated. The process also has the likelihood of alerting the team to other conflicting goals or mutually unacceptable activities. Figure 12 below is a trivial example of a spreadsheet's representing the alternatives to designing a fork lift truck. After the comprehensive review of alternative approaches and means, the rest of the analysis should be centered on the analysis of the most effective alternative using the analytic methods recommended elsewhere in this paper.

Goals	Means				
Power Source	Electric	Gas	Diesel	LPG	Steam
Power Delivery	Gears	Belts	Chains	Hydraulic	Cog Rail
Power Interface	Wheels	Propeller	Hover Craft	Skids	Tracks
Steering	Rear	Front	All	Thrusters	Guide Wire
Stopping	Disk	Drum	Electric	Thruster	Hydraulic
Lifting	Hydraulic	Cable	Screw	Chain	
Operator	Front Seat	Rear Seat	Remote	Standing	

Figure 12 - Listing Alternatives to sub-Goals

Failure Mode and Effects Analysis (FMEA) – listing potential causes and effects of failure:

There are two primary reasons to do a rigorous analysis of failures, observed and potential. The first is to guide design and the second is to provide amelioration of any negative effects of failure. The formal process is laid out in US Armed Forces Military Procedures document MIL-P-1629 (Army, 1949). At the highest level, it involves identifying the steps in a process, failure modes, failure causes, and failure effects. Most FMEA envisions creating a number of documents enumerating these steps and in tables and charts. An example of a simple form for this analysis is offered in Figure 13.

FMEA			
Function	Failure Mode	Effect	Proposed Action
describe the piece	describe the potential failure	describe the potential effects	proposed solutions to avoid the failure mode or to avoid the effects

Figure 13 – Sample Form for FMEA

Having done so, the failures can either be mitigated by design or offset by fall-back procedures.

MENTORPAL

The DoD has identified the increasing need for technical personnel, both in uniform and in a wide range of civilian capacities. To meet that need, potential candidates putatively would be well served to have access to thoughtful and experienced technical personnel who can clearly communicate the careers that may be offered to them (Nye, 2017). There are several factors that militate against that availability of effective mentoring. Some of these factors were mentioned by ICT's education professionals who have given input into the MentorPal project, Figure 14. While anecdotal, they are mentioned here. Many of the counselors noted that a large portion of the population lives in widely dispersed small school districts around the country, many students live in lower Socio-Economic Status (SES) districts in which few technical professionals live, many students are not amenable to face-to-face mentoring by professionals they find intimidating, and many who are available lack skills for it.



Figure 14 – MentorPal: Users & Navy CPO

ICT proposed to the Office of Naval Research that most of these problems could be ameliorated by using their conversational VH technology and providing compelling, engaging, and validating mentoring which could be delivered by locally hosted programs or web-delivered client/server service. It was proposed as being able to provide conversation-like mentoring to student in any locality or of any socio-economic status, in a non-threatening way, and by mentors who have been vetted as being “on-message” and personally engaging and insightful. The program has been funded and initial mentor videotaping of mentors and a computer/human interface design has produced a working prototype. Various versions of this prototype, running on a Microsoft Surface tablet computer, have been assessed by a small number of pre-college students. Input from the students affirmed both the need for the mentorships and the openness to the on-line mentor.

The procedure for creating such a conversational VH mentor involves a number of steps, in addition to the development and tuning of the software code that makes the program function. Literally volumes have been written about how to system engineer large code programs, running all the way from general, almost philosophical approaches, e.g. Professor Fred Brooks book (Brooks, 1995) to more detailed and didactic tomes (Pressman & Maxim, 2005). However, for the more prosaic support functions, no such tomes are familiar to the simulation computer scientists that would assist in organizing their activities. Project personnel, basing their approach on lessons learned from previous ICT efforts, knew they needed a process for the production of the videos. Therefore, with more courage than confidence, they set up a series of steps to produce the videos: familiarization with the career field to be covered, generation of questions to be posed to the mentor, selection of mentors, setting up the recording facilities, doing the recording, tearing down the facilities, editing the videos, time-stamping the resultant files, correcting voice recognition transcriptions, and validating the results. These are represented in column one of Table 1. The rest of the columns show time estimates, in staff-hours, by two researchers, offering different configurations of the evolutions. This procedure was followed in response to SE methods and it produced insights into the subject processes.

Table 2 – Task List and Estimated Times for Video Series.

Estimated time to produce a Mentor	Units = Hours			
	N.J. Kaimakis	D.M. Davis	DMD 4X Staffing	DMD 2nd Mentor
Study the new topic including interviewing professionals to help design questions	15	60	60	20
Create a new question list, including paraphrases and rankings of questions	35	80	80	
(Need 60 new questions out of 360 total for all subsequent Mentors)				20
Identify, select, recruit and train a single talking head/mentor	20	40	40	40
Set up/tear down the recording facilities (at USC or remote) first time - 1-2 hours	1	2	2	2
subsequent times 0.25 - 0.5 hour per session	1.25	2.5	2.5	2.5
Do the recording of the subject mentor ~360 - 400 questions (mentor reads own questions)	18	18		18
30 questions per hour, 2-3 minutes per question, includes getting into standard clothing				
Recording if question read by researcher, with up to two others monitoring (2-4X staff hours)			72	
Edit the resultant video file 1-4 hours of work for 6 hours of video	3	12	12	12
plus 1 hour per 1 hour of video for after effects to render green screen / other edits	18	18	18	18
Timestamp video files for processing 1 hour per 2 hours of recording	9	9	9	9
Fixing transcriptions after voice recognition processing 2 hours per 1 hour of recording	36	36	36	36
Validating questions at various stages of development by presenting to sample users (4 tests)	16	16		16
Validation if sessions are conducted by additional researchers (2-4X staff hours)			64	
Totals	172.25	293.5	395.5	193.5

It was at this point that those actually doing the production began to notice and decry the significant amount of time that was being invested in very tedious and unrewarding assignments. Optimization was done *post hoc* and *ad hoc*, as the researchers had not thought that the small number of iterations justified a lot of time being invested in the process to enhance the production. However, it was realized that implementation by the Navy for a plethora of professions would benefit from such an effort; further, the success of any consideration for other counseling uses would be more promising if such advocacy were to be accompanied by quantification of the price of production. It was at this point that the team made the decision to seek out the input from the Systems Engineering community.

SE TOOLS APPLICABLE TO MENTORPAL

When considering the application of some of the tools outlined above, it may be illuminating to see how they were used in the project identified earlier, the MentorPal project at ICT. Sometimes these concepts were applied formally and sometimes they were applied informally without identifying them by commonly recognized SE title. Nevertheless, their benefits may aid the reader in seeing how they would be useful in the reader's own environment.

V-Model

Within the authors' experience in MentorPal, the major impact of the V-Model imagery was the recognition that the project had to keep in mind the presence of a future long after the end of the research and proof-of-concept testing. Actually, all of the specifications on the left leg of the "V" needed to be considered, not only in terms of the right leg of the "V," but also in the terms of the maintainability and sustainability of the capability being researched. As discussed above, the team kept in mind the necessity of the later testing and achievement measurements.

Stakeholder Diagram

MentorPal is funded to pursue a general increase in the availability of STEM trained personnel for U.S. Navy accessions in the uniformed services and the civilian support organizations. But the team recognized that the higher level goal of national defense made them include other stakeholders who are critical to the economic strength of the nation. Considering these stakeholders quickly took on a hierarchy, recognizing the funders and immediate university supervisors had a much greater stake in the project than someone in the general public. The stakeholder analysis was conducted informally, sitting around a conference table, but all of the authors think a more formalized documentation would have been, and in the future will be, a more effective approach.

Context Diagram

This is one area that was more parochially visited than innovatively created. The major context under consideration in this case was in terms of computing software. The MentorPal "lives inside" the PAL3 program (Personal Assistant for Life Long Learning) (Swartout, 2016), which is a Navy project designed to make access to interactive training assets easier for hand-held, portable computers. It also involves an animated agent that aids in directing the student to the appropriate modules, tracking progress, forming teams, and producing records for the supervisory personnel. It both provided capabilities to MentorPal, but at the same time, added requirements, e.g. need for network connectivity. Other contexts may have been missed, again emphasizing the need to follow the formal process.

Articulating Requirements

This is an area that requires much more emphasis. Often on small projects, it has been the authors' observation that the urge to get going toward a vague vision of questionable consensus to quickly skips over the necessity of carefully articulating requirements. Fortunately in MentorPal, there was a very collaborative program management from the funding agency, so the staff has been able to work diligently to provide the basic program and articulate the requirements by successive steps. The negative aspects of this methodology have been compensated by the indefatigable efforts of the student Research Assistants and a significant amount of volunteer time by professionals.

System Requirements Model

There was a need to trichotomize the requirements in the MentorPal project, but this was accomplished with reasonably perishable documentation, *i.e.* white board diagrams, captured with smart-phone cameras. But even these exercises were not particularly formal. The reason for the formality of documenting the System Requirements Model is to make sure nothing is left out and it organizes them into the three major categories: Operational, Functional and Non-Functional Requirements. The Operational Requirements were: to provide an on-line mentor to assist students in considering STEM careers especially in the Navy. Functional Requirements included answering high percentage of questions, producing compelling responses, avoiding negative imagery, engaging students, keeping latencies low,

and maintaining a quick conversational pace. Non-Functional Requirements were to use MS Surface tablet computers, run within the PAL3 program, use English, keep programming cost reasonable, limit equipment needed for videotaping, and documenting procedures for archiving, and final reports. Having discussed all of these issues, in retrospect, formality of this process has shown itself to be useful.

Implementation Means Analysis

This is a critical issue in every project. There are a number of decision to be made and getting these represented for consideration was effectively carried out in this project by having a spirited brain-storming session, again with documentation on white boards, later recorded for archiving. Some of the issues are indicated below in Table 3.

Table 3 - Implementation Means Analysis

Requirements	Alternative 1	Alt. 2	Alt. 3	Alt. 4
Hand held	Tablet	Smart Phone	Smart Watch	iPad
Question input	Keyboard	Mic	Preloaded Quest.s	Follow-ups
Mentor	Volunteer	Paid	ICT Employees	Recruiters

Failure Mode and Effects Analysis

In MentorPal, there was significant attention paid to fault-tolerance during the design stage. This was largely centered on getting the highest possible level of correct answers. Little attention was paid to the possibilities of failure that rapidly became apparent during early tests with live students. Two failures caused the students to rapidly lose interest, losing the “personal” quality. The first of these was the absence of sufficient audio gain in high noise areas and the second was the disconcerting impact of too many unanswerable questions in a row. While both of these could, and are, amenable to resolution by the team, their being raised in a text environment brought home the need for the team to more fully identify, characterize, and plan remediation for other failures. At least with the population that was being observed, there was very little tolerance for anything that destroyed the illusion of conversation.

DISCUSSION

Impact of SE on MentorPal to Date

One of the major impacts on MentorPal at this stage in its development is the increased awareness of all or the professionals of the necessity to formalize and document the SE-enable analyses of the project. Just applying several of the above has resulted in two very important achievements.

First, in trying to satisfy the requirements articulation functions listed above, it was discovered that there was no extant version of a high-level flow chart for the code. This occasioned the production of a chart and that process alone led to the discovery of several salient changes in the code that will further increase capabilities and reduce crashes, which had been observed about once an hour.

Second, in the Implementation Means study noted above, one aspect stuck out and that was the time required for the transcription of the video-clips, both for projection on the output window and for analysis of the questions to which it would be the most appropriate answer. Due to the implementation of this SE induced work, two major decisions have been reached, 1) the use on the screen will be reconsidered and 2) the transcription duties will be reassessed, mainly in terms of machine voice recognition, level of precision needed, and use of written script for the mentors. These impacts will save nearly one staff week per video and, perhaps more importantly, will dramatically increase staff morale, as the transcription process is as tedious as it is time consuming.

The experiences presented above are both illuminating and reasonably intuitive; no advanced mathematical analyses are required. When this was all observed and considered, it seemed that there were good reasons to consider the application of some of these techniques and approaches in most other research projects.

Further, larger projects with even more detailed and onerous procedures, it may warrant the recruitment of a System Engineer as a member of the research team. But the authors maintain that these tools can be applied by the team members themselves, with rewards of savings and time outweighing the time and effort to implement the SE tools.

The authors assert that the data above justify the careful consideration of the adoption of some of the tools. They conclude that the use of SE is prudent, cost-effective and accessible by the vast majority of research teams.

It will require discipline and it will seem disruptive at first, but the authors feel it will be worthwhile. The suggestion is that the research management personnel lead and encourage the formalizing and documenting as shown above.

Finally, the authors assert that it would be prudent to review many of the sources cited above to keep abreast with new techniques in System Engineering, as this field is increasingly useful in managing any endeavor.

ACKNOWLEDGEMENTS

The authors want to acknowledge the leadership and support from William Swartout, PhD. His counsel for us and patience with us were invaluable. Further credit is due to Julia Campbell, Ed. D., Madhusudhan Krishnamachari, Kayla Carr, Nicholas Kaimakis, Samuel Breck and Joseph Gunderson for their contributions. It cannot go without mention that this is a much improved paper due to the effort of and input from our Track Chair, Julian Abich, PhD. Much of the work described above was conducted in response to Office of Naval Research contract: MentorPal: Growing STEM Pipelines with Personalized Dialogs with Virtual STEM Professionals, N00014-16-R-FO03 as well as NPCEditor and PAL3, under Army contract W911NF-14-D-0005. The opinions expressed herein are the authors' own and do not necessarily reflect those of the Department of the Navy, the Department of the Army or the U.S. Government.

REFERENCES

- Allwood, J., Traum, D., & Jokinen, K. (2000). Cooperation, Dialogue and Ethics. *International Journal of Human-Computer Studies*, 53(6), 871-914.
- Amburn, E.P., Burns, D.P., Davis, D.M. & Lucas, R.F. (2016). Facilitating V-Model Analyses: Data Visualization for Test and Evaluation. *The ITEA Journal of Test and Evaluation*, 37(3), 246-256.
- Army, (1949). *Procedure for Performing a Failure Mode Effect and Criticality Analysis*. United States Military Procedure. MIL-P-1629.
- BHW, (2018). *System Engineering Tools and Techniques*. Burge Hughes Walsh Limited. Retrieved 06 February 2018 from URL: <https://www.burgehugheswalsh.co.uk/systems-engineering/tools.aspx>.
- Brooks, F. (1995), *The Mythical Man-Month*. Anniversary Edition, Boston, MA: Addison-Wesley Professional.
- Burge, S. (2011). *Functional Modeling, The System Engineering Toolbox*. Retrieved 12 January 2018 from: <https://www.burgehugheswalsh.co.uk/Uploaded/1/Documents/Functional-Modelling-Tool-Draft.pdf>.
- Chowdhury, G. (2003). Natural language processing. *Annual review of information science and technology*, 37(1), 51-89.
- Collobert, R., & Weston, J. (2008, July). A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning." In *Proceedings of the 25th International Conference on Machine learning* (pp. 160-167).
- FHWA, (2018). *Systems Engineering for Intelligent Transportation Systems*, Federal Highway Administration. Retrieved on 11 February 2018, from URL: <https://ops.fhwa.dot.gov/publications/seitsguide/section3.htm>.
- Gottschalk, T. D., Yao, K-T., Wagenbreth, G. & Davis, D. M., (2010), Distributed and Interactive Simulations Operating at Large Scale for Transcontinental Experimentation, in the *Proceedings of the IEEE/ACM Distributed Simulations and Real Time Applications 2010 Conference*.
- Gould, J. (2107). Military chiefs warn of force-readiness struggles, *Defense News*, Retrieved 15 January 2018 from: <https://www.defensenews.com/congress/2017/02/07/military-chiefs-warn-of-force-readiness-struggles/>.
- IEEE, (1994)., *Standard for Application and Management of the Systems Engineering*. Process, IEEE Standard 15288.1 [Final Draft], 26 September 1994.
- Keating, C., Rogers, R., Unal, R., Dryer, D., Sousa-Poza, A., Safford, R., ... & Rabadi, G. (2003). System of Systems Engineering. *Engineering Management Journal*, 15(3), 36-45.
- Kenny, P., Hartholt, A., Gratch, J., Swartout, W., Traum, D., Marsella, S., & Piepol, D. (2007). "Building Interactive Virtual Humans for Training Environments." In *Proceedings of the Interservice/Industry Simulation, Training and Education Conference*, Orlando, Florida, 2007 (Vol. 174).

- Kudryavtsev, A. (2016). Automatic Speech Recognition Services Comparison. *Grid Designer's Blog*. N.p., 11 Jan. 2016. Retrieved 20 Apr. 2017 from: <http://blog-archive.griddynamics.com/2016/01/automatic-speech-recognition-services.html>.
- Lee, K. F. (1988). Automatic speech recognition: the development of the SPHINX system (Vol. 62). New York, NY.: *Springer Science & Business Media*.
- Lightsey, B. (2001). *Systems Engineering Fundamentals*. Defense Acquisition University Press, Fort Belvoir, Virginia. DTIC document a606327. Retrieved 12 January 2018 from: <http://www.dtic.mil/dtic/tr/fulltext/u2/a606327.pdf>.
- Lucas, R.F., Davis, D.M., & Burns, D.P. (2015). System of Systems Complexity Addressed by Practical Adiabatic Quantum Computing. *The ITEA Journal of Test and Evaluation*, 36(4), 311-321.
- Nye, B., Swartout, W., Campbell, J., Krishnamachari, M., Kaimakis, N. and Davis, D. (2017). MentorPal: Interactive Virtual Mentors Based on Real-Life STEM Professionals. in the Proceedings of the *Interservice/Industry Simulation, Training and Education Conference*.
- Payne, S. C., & Huffman, A. H. (2005). A longitudinal Examination of the Influence of Mentoring on Organizational Commitment and Turnover. *Academy of Management Journal*, 48(1), 158-168.
- Pressman, R. S. & Maxim, B. (2005). *Software Engineering: a Practitioner's Approach*. Basingstoke, UK: Palgrave Macmillan.
- Pugh, S., & Clausing, D. (1996). *Creating Innovative Products using Total Design: the Living Legacy of Stuart Pugh*. Boston, Mass.: Addison-Wesley Longman Publishing Co., Inc.
- Putz, K., (2013), *What's the Difference Between Science and Engineering*. Helix, Connecting Science to You, Retrieved on 06 February 2018 from URL: <https://helix.northwestern.edu/blog/2013/12/what-difference-between-science-and-engineering#comment-100551>.
- Rizzo, A., Lange, B., Buckwalter, J. G., Forbell, E., Kim, J., Sagae, K., ... & Parsons, T. (2011). "SimCoach: an Intelligent Virtual Human System for Providing Healthcare Information and Support." *International Journal on Disability and Human Development*, 10(4), 277-281.
- Satori, H, Harti, M., & Chenfour, N. (2007). Introduction to Arabic Speech Recognition using CMU Sphinx System. *arXiv preprint arXiv:0704.2083*.
- Saaty, T. L. (1980). *The Analytic Hierarchy Process: Planning, Priority Setting, Resources Allocation*, 281. New York: McGraw.
- Swartout, W. R., Gratch, J., Hill Jr., R W., Hovy, E., Rickel, J., & Traum, D. (2006). Toward Virtual Humans. *AI Magazine*, 27(2), 96.
- Slide Share. (2018), *Context Diagram*, Slide Share LinkedIn Corporation, Retrieved 08 February 2018 from: <https://www.slideshare.net/infinityrulz/context-diagram-3043445>.
- Swartout, W. R., Nye, B. D., Hartholt, A., Reilly, A., Graesser, A. C., VanLehn, K., ... & Wang, L. (2016, March). Designing a Personal Assistant for Life-Long Learning (PAL3). In *The Twenty-Ninth International Flairs Conference*.
- Tran, E. (1999). *Requirements and Specifications*, Carnegie-Melon University Technical Report. Retrieved on 06 February 2018 from: http://users.ece.cmu.edu/~koopman/des_s99/requirements_specs/.
- Traum, D., & Rickel, J. (2002). Embodied Agents for Multi-party Dialogue in Immersive Virtual Worlds. In Proceedings of the first *International Joint Conference on Autonomous Agents and Multi-agent Systems*: part 2 (pp. 766-773). ACM.
- Traum, D., Jones, A., Hays, K., Maio, H., Artstein, R., Debevec, P. & Jungblut, K. (2015, November). New Dimensions in Testimony: Digitally Preserving a Holocaust Survivor's Interactive Storytelling. In *International Conference on Interactive Digital Storytelling* (pp. 269-281). Cham, Switzerland: Springer International Publishing.
- Tucker, G. (2017). *Analytic Hierarchy Process*, The Future of Service Management. Retrieved from the internet on 08 February 2018 from: <https://www.itsminfo.com/analytic-hierarchy-process/>.
- USC. (2018). *Industrial and Systems Engineering (BS)*, USC Catalogue, University of Southern California. Retrieved 06 February 2018 from: http://catalogue.usc.edu/preview_program.php?catoid=2&poid=1514.