APPENDIX E – System Development Plan

Scope

This Software Development Plan (SDP) establishes the plan for the development, testing and implementation of the Activities-based Cost Estimator (ACE) tool.

System Overview

The ACE system is a new software development effort. The application is a stand-alone tool to estimate the lengths of welds within and between structural assemblies. The system will perform weld identification and classification activities using the ACIS geometry kernel from Spatial. Spatial also offers the 3D InterOp library of data translators to import the majority of file formats required by Northrop Grumman Ship Systems (NGSS). For additional import capability, the AutoDesk RealDWG utility will be used to import AutoCAD-based files: DXF and DWG.

A graphical user interface (GUI) will display the CAD models, their associated structure hierarchy and identified welds. The three dimensional display will be driven by the TechSoft3D HOOPS graphics engine. The two-dimensional GUI (menubar, toolbar, general window interface control) will be created using the Microsoft Foundation Classes (MFC) routines. The GUI will also offer the ability to select welds and assembly parts as well as modify their attributes. The weld data can also be exported into an extensible mark-up language (XML) or comma-delimited format. The compiled data can then feed downstream systems or models.

NGSS, along with input from other shipyards, specified the requirements for the system. The specification states that the tool will supplement ongoing production planning and analysis efforts by collecting weld information for use by shipyard planners and engineers. The System Specification document and System Schematic diagram were developed from the gathered requirements from NGSS as well as other shipyards and shipbuilding entities.

This system specification phase of the ACE development effort was funded by the National Shipbuilding Research Program (NSRP) Business Process Technologies panel.

Document Overview

This SDP identifies the proposed development tasks, testing plan, developmental release schedule and implementation requirements for the Activities-based Cost Estimator (ACE) tool. It defines the personnel, organization and resource requirements for the system development. The SDP does not contain any proprietary information or International Traffic in Arms Regulations (ITAR) restricted data. The remainder of the document has the following sections:

- "*Referenced Documents*" lists all documents referenced by this SDP and used during its preparation.
- "*Overview of Required Work*" describes the scope of the project and system requirements

- "*Software Engineering*" describes the plans for general software development activities.
- "*Software Development Tasks*" describes the system coding tasks and supporting activities.
- "*Project Schedule*" provides the proposed timeline for the major tasks.
- "*Project Organization and Resources*" describes the organization and resources required to accomplish the work.
- "*Risk Management*" provides a concise statement of possible risks and a proposed mitigation plan.
- "*System Cost Breakdown*" proposes a rough budget for the development effort.

Relationship to Other Plans

This SDP and the following supporting documents will guide the development of the ACE tool:

- ACE System Requirements Document, GCRMTC Document
- ACE System Schematic, GCRMTC Document
- ACE System Specification Document, GCRMTC Document

The ACE system will possibly be affected by the results of the AutoGen project and the ongoing software development within the Ship Design Tool Enhancement (SDTE) project funded by the NSRP. The SDTE project is expanding the capabilities of the ShipConstructor design package to integrate it with the common parts catalog, coordinate work with subcontractors by accessing the same database and incorporate more detailed weld specification data. The current plan is to utilize the results of the commercialization effort within the AutoGen project. This decision was based on the proposed capabilities of the ACIS kernel within AutoGen and the lack of access to the weld identification routines within the ShipConstructor Application Programming Interface (API), as of the proposed start date of the development effort (mid-2008).

There were no specified development or implementation standards identified for the creation of the ACE tool.

Referenced Documents

The documents listed below were either used to create this document or are referenced in it:

- ACE System Requirements Document, GCRMTC Document
- ACE System Schematic, GCRMTC Document
- ACE System Specification Document, GCRMTC Document
- Wikipedia web site (http://en.wikipedia.org/wiki/Main_Page)

Overview of Required Work

The objective of the ACE tool, as specified by NGSS, is to extract weld information from supplied CAD files and export that data into shipyard-specified files formats to feed downstream analysis and planning efforts. The development of this tool is guided by the following requirements. At this time, there are no requirements specified for the deliverables schedule, programming standards or development methods.

System Requirements

The following major requirements have been set by NGSS. More detailed information can be found in the ACE System Requirements Document.

- Pilot application will be the NGSS-New Orleans panel line
- The system should not be Navy program specific
- Assume all seams/joints to be welded
- Extract weld information from following construction stages
    - Between two individual parts
    - Welds within a structural unit
    - Welds to join previously welded assemblies
- Tool will deliver the following at any level of the assembly
    - Accumulated weld length
    - Weld orientation (e.g. horizontal, vertical, overhand)
    - Weld process (e.g. fillet, butt)
- Export data in XML and/or comma-delimited format

Implementation Requirements

Upon the completion of the development effort, NGSS has required the delivery of a User's Manual for the system and an overall schematic of the tool. The User's Manual will include the instructions for the operation, configuration and maintenance of the tool. The final schematic will build on the ACE System Schematic delivered at the end of the first phase of the project. As part of the final deliverables of the development project, the refined system requirements, use cases and specification will be provided to NGSS.

A training document was not required, but will be considered depending on available resources and identified need from NGSS once the tool has been developed. NGSS personnel will receive training on the system at the time of each interim build release, see the "Develop System" sub-section under the "Software Development Tasks" section.

There are no other documentation requirements for the development of the tool.

Software Engineering

Deliverable Software

The only piece of deliverable software is the ACE tool itself. The system requirements specified that the tool should be a stand-alone application. As such, there will be no other deliverable tools developed under this project. The ACE deliverable will include any required dynamically-linked libraries associated with the procured software discussed in the next section.

Procured Software

As noted, the ACE tool will utilize commercial software libraries as the basis for the geometric interrogation routines, file import translation and 3D display interfaces. The following tools will be procured to support the system development effort:

- **Spatial ACIS:** geometry kernel
- **Spatial 3D InterOp:** file import translation routines for the essential-designated file formats
- **AutoDesk RealDWG:** file import translation routines for the preferred-designated file formats
- **TechSoft3D HOOPS:** high-level 3D display and interface engine

The project team will also utilize previously procured programming language systems (Microsoft .NET) and freeware systems for source code management (e.g. CVS or Mercurial) and code documentation (doxygen).

Non-Development Software

There are no pieces of software that will be procured that do not directly support the development effort.

Software Build Strategy

There are three basic strategies for a software/program development: Grand Design, Incremental, and Evolutionary. The Grand Design method performs each of the following steps only once: determine user needs, compile the requirements, design the system, implement, test, fix and then deliver the system. The Incremental strategy, as the name implies, creates the system through the release of incremental builds. Each iterative release of the system adds capabilities until the system is complete. The user needs and requirements are determined at the start of the effort and are addressed within the scheduled releases. The Evolutionary strategy is similar to the Incremental strategy in that the completed system is reached through a series of build releases. The difference is that only preliminary user needs and requirements are gathered upfront and are later refined along the development path. This strategy can be used when the user needs are not fully understood, but may result in duplicative or wasted development effort.

For the ACE project, the Incremental strategy will be used.

Software Development Tasks

Before any code creation tasks are started, the following tasks are envisioned for the system development phase of the ACE project:

Project Management

The University of New Orleans (UNO) Simulation Based Design Center (SBDC) will act as the overall team lead for the project. Northrop Grumman Ship Systems will provide additional technical management and project management. The following Project Management tasks apply to each project organization:

- plan, manage, and track progress against project tasks,
- manage project budget and project expenditures,
- compile monthly progress reports and deliver to the Principal Investigator,
- participate in quarterly project presentations and scheduled meetings.

This task will also coordinate the collection of additional requirements or proposed applications from other shipyards. The SBDC project team will also attempt to cultivate a commercialization partner for the ACE tool and/or coordinate the integration with other Navy-funded software development efforts, especially the continuing work under the Second-Tier Design Enhancement project.

Setup Development Environment

The goal of this task is to define the development environment components (e.g. software compiler, configuration management system, major system libraries), compile the components into a working system and test using preliminary sample code. The ACE tool will be developed on existing computer platforms and language compilers. As noted, the only procured software will be the Spatial components (ACIS, 3D InterOp and HOOPS) and the RealDWG data translation toolkit.

Setup Test Environment

This task will refine the testing requirements for the system, identify any hardware and software components and compile the components into working test platforms. The current system requirements specify that the tool should run on Windows 2000 and XP platforms. Thus, two personal computer systems will be configured with these operating systems. The SBDC may also check the tool on Vista and Vista 64-bit operating systems.

Component Training

After the development environment is operational, the SBDC development team will begin training on the major components of the system, including: ACIS, 3D InterOp and HOOPS. This may be accomplished through formal training sessions; consultation with technical support or services representatives; or self training using the vendor manuals. The path will be determined by the available resources and schedule.

One option provided by Spatial is a new program called the Application Graphics Manager. The goal of the program is to accelerate the development of commercial tools that utilize the ACIS kernel and HOOPS graphics display engine by providing higher-level routines and program templates to speed the creation of the base application. The SBDC may utilize this system to decrease the development time of the interface and data translation components.

Use Case Development

To ensure that the interface meets the user requirements and is efficient, the SBDC will guide the development and refinement of detailed use cases with the NGSS personnel who will be using the tool. The goal is to develop a near click-by-click directive for the development of the two-dimensional graphical user interface (i.e. windows environment) and the three-dimensional graphical display interaction (i.e. display requirements and mouse interaction). An iterative approach will be used where SBDC personnel capture the needs of the users, document those requirements and deliver the use cases to NGSS for additional review. Once the final use cases are developed, the SBDC will extract additional, more detailed system requirements and specifications from the documents.

Gather Test Data

In parallel with the use case development and environment setup tasks, NGSS personnel will be responsible for collecting the sample files to be used during the development and testing of the system. The requirement is the provide CAD models that accurately represent the "real" format and content of NGSS design files. NGSS will provide the content of each file, the naming conventions used, identified standard parts, and preferred manufacturing orientation. The collection should include, at a minimum, one assembly in all of the required file formats and a complete series of files representing a large structural unit in one format.

Detailed System Design

When all of this preliminary data has been collected, the SBDC will refine the current system specification, schematic and development plan to accommodate any new components. At that time, the additions will be reviewed for possible risks and their effect on project resources and schedule; to prioritize the incorporation of these new requirements; or eliminate them for this phase of development.

NGSS will also specify the required format for the XML export file. With this information and additional input from the use cases, the SBDC will then design the output file formats as well as the lexicon and format for the keyword-based Rules, Configuration File and Project Save file formats. This information will also feed the design of the internal data structures for the parts, weld and file lists data structures with the associated object-oriented classes.

Develop System

Once these tasks are completed, this SDP proposes five (5) interim releases of the code in addition to a final version at the end of the project:

- **Basic Capabilities**: first level of GUI capabilities, import of a single part file, identification of all welds within the assembly and export of the lengths to either the XML or CSV format.
- **Multiple Assemblies**: ability to import multiple assemblies, identify as loose group of parts or previously welded structure, address bugs/fixes identified after previous release.
- **Modify/Query**: ability to query and modify weld and part attributes; select welds or parts within the respective display areas or the 3D display window; ability to import multiple file formats; and address bugs/fixes identified after previous release.

- **Rules**: identification of standard parts; ability to apply leave loose rules; editable rules for part orientation and process identification; and address bugs/fixes identified after previous release.
- **Orientation and Roll-Up**: user interface to specify primary and secondary manufacturing orientations; roll-up weld lengths at any level in the part hierarchy; and address bugs/fixes identified after previous release.

The final release is not scheduled to include any new capabilities. The associated development time for compilation of the final release is purely to address concerns or bugs raised in previous releases. Each interim release will include the following components:

- Internal testing within the SBDC test environment. This includes testing across the different hardware systems and using the varied set of file formats and CAD content. The primary goal is to identify bugs on the code and exercise the interface to ensure that it meets the requirements for the release. This task will span the entire length of the individual build schedule.
- Release of the tool to NGSS for review. Once the tool is ready for release, the SBDC will host a short meeting to present the tool to NGSS personnel, introduce the new capabilities offered in this release, demonstrate how they are used and identify known issues or limitations within the build. This meeting should not take more than ½ day; more than likely this will simply be an hour or two in length.
- NGSS review of the tool, After the tool has been released to NGSS, the planners will have a week to exercise the system. The goal is to ensure that the interface is effective and efficient within the shipyard environment.
- Gather shipyard comments (short meeting to gather feedback from shipyard personnel; identify bugs or new required capabilities; prioritize and schedule the incorporation of these fixes in subsequent releases or postpone for future development phases)

Create Supporting Documents

As noted previously, the SBDC will create a User's Manual for the system. This will include directions on how to use the tool, the interface, configuration capabilities and guidelines for editing the Rules file. If required and resources allow, the SBDC will also develop training documentation and export the automated documentation for code from the doxygen system.

Final Deliverables

At the end of each quarter, the SBDC will deliver a quarterly report containing the following information

- Project Synopsis,
- Budget Status,
- Accomplishments During Current Quarter,
- Proposed Activities for Next Quarter,
- Collaborative Efforts,
- Project Timeline.

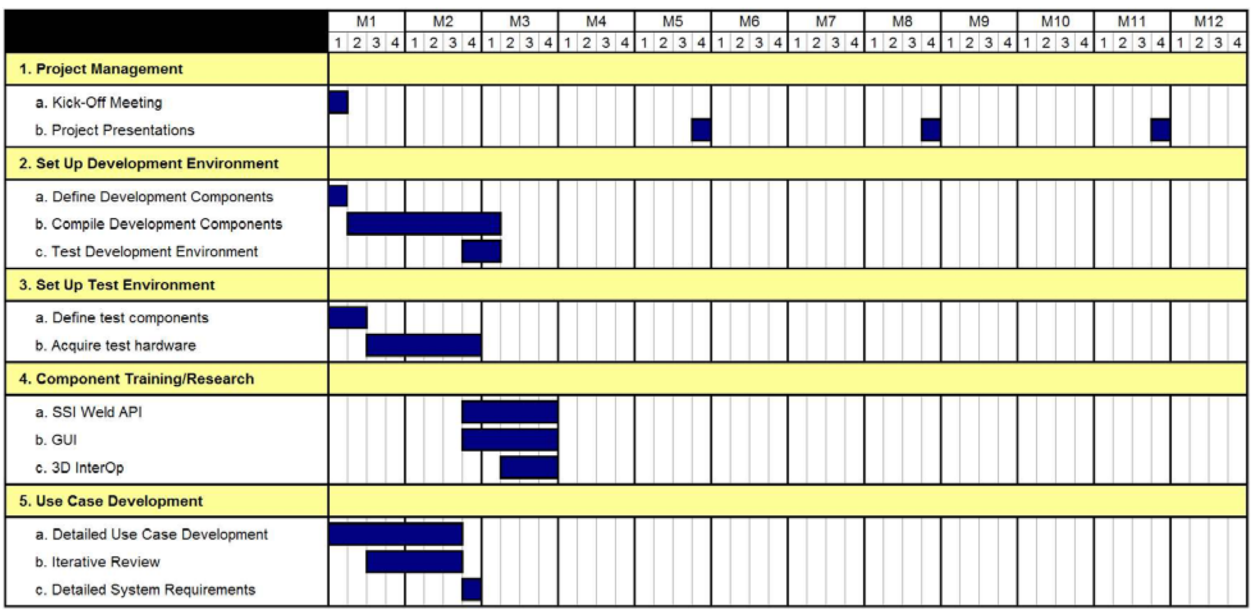
At the end of the project, the SBDC will deliver a Final Report detailing the project objectives, progress made against those objectives with details on project conduct, results, benefits (quantitative and qualitative), conclusions and any recommendations. The report will have the following general format:

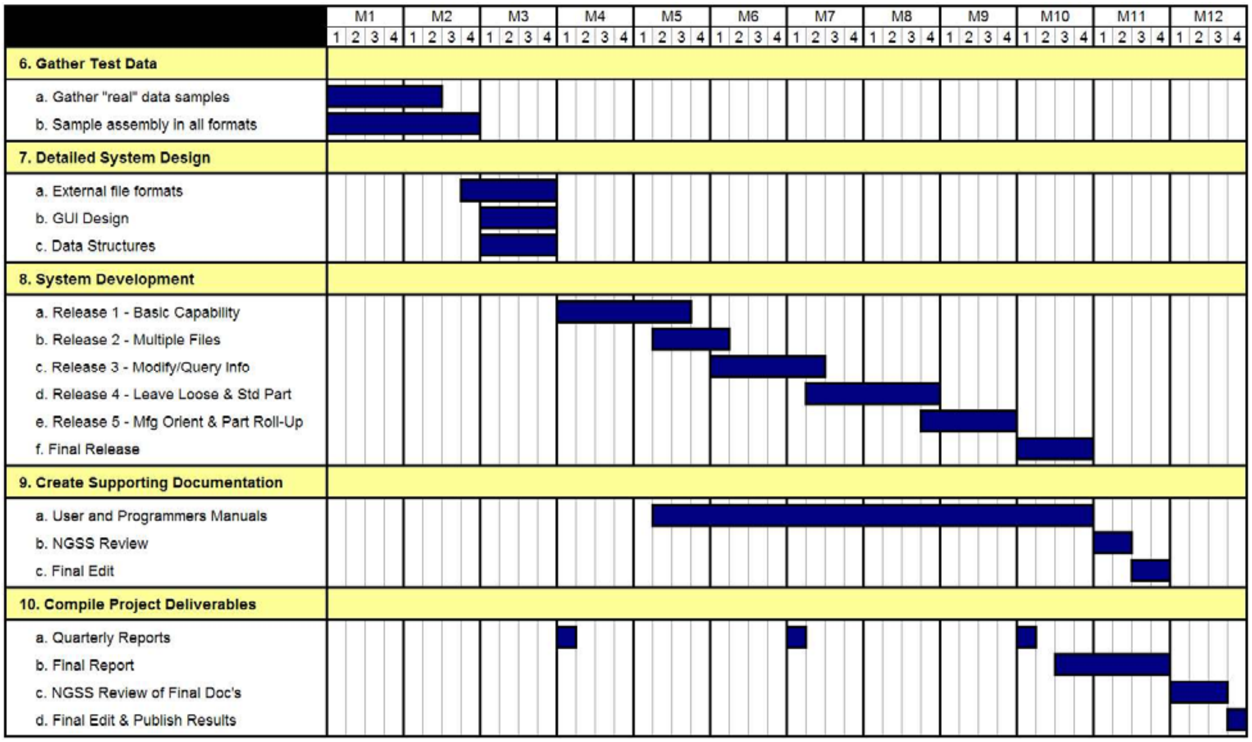
- Introduction
- Objectives
- Method of Procedure, including initial task outline with any variations
- Discussion of Results, including progress against tasks, discussion of compiled data, future system development plan and proposed benefits
- Conclusions
- Recommendations
- Appendices, potentially including compiled system requirements, use cases, updated system schematic and specifications

With the consent of NGSS, the system specifications, use cases and investigation results will be posted on the GCRMTC web site (content will be cleaned or simplified to protect NGSS proprietary information). The SBDC Project Team will also present the progress of the project a multiple industry meetings and conferences.

Project Schedule

The development phase of the project was planned as a 12-month effort. The proposed timeline is provided in the following figures.





Project Organization and Resources

Project Organization

The overall responsibility for the project shall be with the Principal Investigator (University of New Orleans). He shall manage progress against the objectives, schedule, budget and milestones of the project. He shall rely on the technical and financial leads within both organizations to report on progress, identification of potential risk, asses the impact of that risk on the project and development of mitigation plans.

The NGSS Technical Lead and the UNO Development Lead will be responsible for the collection and dissemination of gathered system requirements and specification information as well as progress and feedback on the build releases. The Business Lead and the UNO P.I. will prepare, monitor, control, maintain and administer the project's contractual responsibilities to the funding agency. A high-level personnel relationship chart is provided in the following figure.

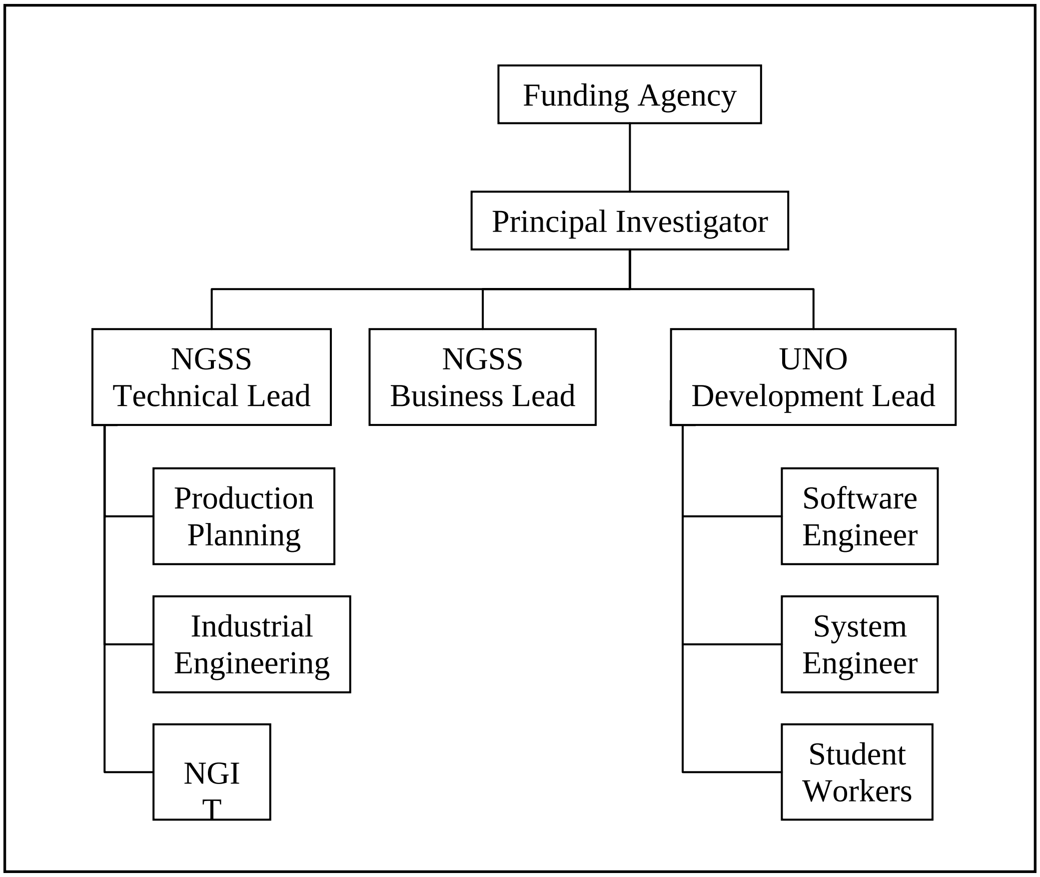


Figure 1 Proposed personnel hierarchy.

Project Resources

Personnel Resources

The University of New Orleans Simulation Based Design Center (SBDC) will act as the Principal Investigator for the project and provide the primary software development resources. Northrop Grumman Ship Systems will provide additional project management effort as well as primary systems specification and build release testing labor. At the SBDC, one person will be assigned project management duties at approximately half time. Approximately 2.5 full-time equivalents will form the core system design and development team. When possible and efficient, student workers will be used to supplement the development effort. An additional half-time equivalent is required for environment development, maintenance and hardware support.

The primary personnel resources at NGSS will be project management (proposed half-time to shared full time equivalent) as well as a myriad of engineering personnel, in finite time blocks, to support the detailed specification of the system and review of the build releases.

Developer Facilities

This project will use the facilities available at the University of New Orleans Simulation Based Design Center located at the UNO/NGSS Maritime Technology Center of Excellence. The project will take advantage of the SBDC Conference Room, Classrooms and Visualization Theater to meet with shipyard personnel and present the progress on the project. The Center will utilize the existing infrastructure and software libraries as well as take advantage of academic and University programs for site license versions of software to minimize project costs.

Shipyard Furnished Equipment, Software and Services

The only required, shipyard-supplied products are the sample files for development testing. At this time, the only requirement for shipyard services is the labor hours to

- Specify the XML format,
- Compile sample data files,
- Detail the specification of the tool,
- Refine use cases to drive the interface design,
- Supply feedback from build release reviews, and
- Review reporting documentation for proprietary and ITAR related data.

Other Required Resources

The two sources of "other required resources" will involve the training of the developers on the Spatial suite of tools and participation by other shipyards and entities to add requirements to the system and to cultivate a commercialization partner. To minimize development time, formal training at the vendor site is being considered as well as a new program at Spatial to "kick-start" software development. As noted previously, the Application Graphics Manager and Scheme program were developed to utilize Spatial Technical Services personnel to quickly generate the internal data linkages and interfaces to accelerate the creation of a commercial software application.

Risk Management

According to Wikipedia.com, risk management is the activity that integrates the recognition of risk, assessing its potential, developing strategies to manage it and mitigating its impact using project resources.

It is inevitable within a project that problems will arise. When they do, the consequences must be evaluated for their potential impact on the successful completion of the project. The risks to the delivered system, the schedule and the budget must be determined. The project management team will be responsible for continually assessing the potential for risk within the project, identifying those risks, prioritizing their potential impact and developing mitigation plans to eliminate or decrease that impact on the project.

For example, there is at least on potential risk addressing the broad applicability of the tool across multiple shipyards. The naming convention used by NGSS is most assuredly different than those used at other shipyards. The naming convention is used to infer the part hierarchy and the identification of standard parts. Hard coding the translation factors into the software for NGSS will limit its applicability to other customers, restricting its market potential. To minimize this obstacle to commercialization, an interface tool may need to be developed to allow the user to configure a naming convention map for the parts identification and hierarchy or create that capability within the rules file (possibly within the Rules or Configuration support files).

Systems Cost Breakdown

Development Budget

The primary component of the development budget is labor. At UNO, the labor includes project management time; creation of the development and testing environments, coordination of data gathering with NGSS, compilation of detailed specifications; translating that information into a refined plan; and then creating and debugging the software. In addition to project management components, the primary NGSS labor driver is the users' time to review the tool, test it for effectiveness, meet with UNO personnel to transfer identified issues and coordinate the mitigation of these problems.

To complete the development environment, license fees may need to be paid to the vendors of the component platforms. Current commercial licensing fees for the Spatial suite of tools is provided in the following table.

Table 1 Commercial annual license fees for Spatial products

Tool Component	Annual Fee
ACIS Kernel w/HOOPS	$40,163
3D InterOp	
CATIA V5 Reader	$4,968
ProE Reader	$1,850
Inventor Reader	$1,850
IGES Reader	$851
STEP Reader	$851
TOTAL	**$50,533**

Additional 3D InterOp modules can be purchased for CATIA V5 write and manufacturing options as well as file import routines for Unigraphics, Solid Works, Parasolid and VDA-FS file formats. The prices shown are the publicized commercial fees for 1-9 seats of the Spatial tools. However, UNO is currently in negotiations with Spatial to try to decrease these costs for the development phase of the project.

If the RealDWG toolset is required, then UNO will need to pay $5,000 for the first-year of the development phase. If the development phase extends beyond 12 months, then each additional year requires a $2,500 licensing fee. Where possible, UNO will utilize freeware or shareware items to minimize development costs. For example, there are a number of freeware configuration management systems that can be used as well as the doxygen tool for documenting code.

At this time, there are no plans for major hardware purchases to support the testing environment.

The project management and technology transfer tasks of the project will require travel expenses to present project progress and to demonstrate the system at industry conferences and meetings. It is envisioned that there will be one major travel in each quarter of the project: (1) to introduce the project and to solicit additional input on requirements and potential commercialization partners, (2) report on progress and present the final system specification and design, (3) report on progress and demonstrate the latest version of the code, (4) final report presentation and demonstration of the tool. At most, two personnel are allocated for each trip. Additional travel will be required for the development team to attend formal training course on the Spatial suite of tools. Depending on the quantity of students, it may be more cost efficient to perform the training at the UNO SBDC site, as opposed to traveling to the vendor site.

The following proposed budget is for the development portion only. The proposed NGSS budget for the development effort was not delivered in time for the release of this document for the project deliverables.

Table 2 Estimated development budget. Labor is encumbered.

Budget Component	Amount
Labor (3.5 fulltime equivalents)	$322,560
Supplies	$1,000
Software Licenses	$56,000
Travel	$8,000
Indirect (26%)	$100,766
TOTAL	**$488,326**

Implementation & Maintenance Costs

Since the tool was specified and designed to operate independently from other shipyard systems or other applications, there should be little to no direct implementation costs. The only upfront and possibly continuing costs will be the licensing fees from the commercial vendors for the use of the kernels.

If NGSS does not require continued updates to the RealDWG library or access to their Technical Support services, there are no additional fees for the use of the import capability. Otherwise, NGSS would need to pay $2,500 annually to maintain and update the file import capabilities within RealDWG.

Under the normal licensing agreement, NGSS would need to pay the full annual fees for the Spatial products, as shown in Table 1. However, UNO is currently in negotiations with Spatial to try to decrease or completely eliminate these costs for continued use of the ACE tool. As with the RealDWG license, the no-fee option would eliminate future system updates and technical support from Spatial.