Software Requirements Specification

for

ETM Advising Database

**Version 1.3 approved**

**Prepared by Holly Fox, Jake Mulligan, Jordan Zenisek, Kevin Godenswager**

**Ohio University Senior Design**

**January 25th, 2019**

**Table of Contents**

[**Introduction**](#_3znysh7) **4**

[**Purpose**](#_2et92p0) **4**

[**Document Conventions**](#_tyjcwt) **4**

[**Intended Audience and Reading Suggestions**](#_3dy6vkm) **4**

[**Product Scope**](#_1t3h5sf) **4**

[**References**](#_4d34og8) **5**

[**Overall Description**](#_2s8eyo1) **5**

[**Product Perspective**](#_17dp8vu) **5**

[**Product Functions**](#_3rdcrjn) **6**

[**User Classes and Characteristics**](#_26in1rg) **6**

[**Operating Environment**](#_lnxbz9) **6**

[**Design and Implementation Constraints**](#_35nkun2) **7**

[**User Documentation**](#_1ksv4uv) **7**

[**Assumptions and Dependencies**](#_44sinio) **7**

[**External Interface Requirements**](#_2jxsxqh) **8**

[**User Interfaces**](#_z337ya) **8**

[**Hardware Interfaces**](#_3j2qqm3) **8**

[**Software Interfaces**](#_1y810tw) **8**

[**Communications Interfaces**](#_4i7ojhp) **8**

[**System Features**](#_2xcytpi) **9**

[**Web Interface**](#_1ci93xb) **9**

[**Database**](#_owvzgietk89q) **10**

[**PDF DARS Parsing Module**](#_wzvobu7gof9s) **10**

[**Other Nonfunctional Requirements**](#_2bn6wsx) **11**

[**Performance Requirements**](#_qsh70q) **11**

[**Safety Requirements**](#_1pxezwc) **11**

[**Security Requirements**](#_49x2ik5) **12**

[**Software Quality Attributes**](#_2p2csry) **12**

[**Business Rules**](#_147n2zr) **12**

[**Other Requirements**](#_3o7alnk) **13**

**Revision History**

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Date** | **Reason For Changes** | **Version** |
| Kevin Godenswager Jordan Zenisek  Holly Fox  Jacob Mulligan | 10-3-2018 | Updated information, grammatical fixes. | 1.1 |
| Kevin Godenswager Jordan Zenisek  Holly Fox | 1-17-2019 | Updated information based on changes that have been made. | 1.2 |
| Kevin Godenswager Jordan Zenisek  Holly Fox | 1-25-2019 | Updated the information throughout document to reflect changes and expand on project requirements. | 1.3 |
| Jordan Zenisek | 4-18-2019 | Final Updates reflecting end product. | 1.4 |

# Introduction

## Purpose

The purpose of this project is to design a web application for Engineering Technology and Management (ETM) advisors. The current procedure is fully manual, and this application will automate the process. This will be done by providing views for a student’s Program of Study, a prospective Program Planner, and Course Loading views which are all covered in this document.

REVISION 4. RELEASE 1.

*<Identify the product whose software requirements are specified in this document, including the revision or release number. Describe the scope of the product that is covered by this SRS, particularly if this SRS describes only part of the system or a single subsystem.>*

## Document Conventions

There are currently no document standards or conventions.

*<Describe any standards or typographical conventions that were followed when writing this SRS, such as fonts or highlighting that have special significance. For example, state whether priorities for higher-level requirements are assumed to be inherited by detailed requirements, or whether every requirement statement is to have its own priority.>*

## Intended Audience and Reading Suggestions

The intention of this document is for the formal explanation of the goals and requirements for this project. It will give developers an understanding of the software requirements, as well as express the views of the project manager and clients. High level requirements are outlined in sections 2 and 3. To fully understand this software, it is recommended to read the full document in order.

*<Describe the different types of reader that the document is intended for, such as developers, project managers, marketing staff, users, testers, and documentation writers. Describe what the rest of this SRS contains and how it is organized. Suggest a sequence for reading the document, beginning with the overview sections and proceeding through the sections that are most pertinent to each reader type.>*

## Product Scope

This software will provide an online web application to assist ETM advisors throughout the advising process. This will provide major advantages in time saved and ease of the advising process, as many of the components of advising process will be automated.

In particular, student’s course history (DARS) will be automatically parsed and uploaded to a database. The web application will interface with this database in order to show a student’s progress in their course of study, (Program of Study view) which an advisor will be able to assess and use in advising a student’s next step.

The information pulled from student’s DARS reports will also be used to create a second, automatically completed ‘Program Planner’ view,’ which will outline the courses a student will need to take within future semesters, planned along a four-year track.

Additionally, the courses planned within the Program Planner view will be compiled across all ETM students to create a third view, the Course Loading view. This will show the projected number of enrolled students in each ETM course by semester through the next four years.

Advisors will have access to the Program of Study and Program Planner belonging to their respective advisees to use as needed and during advising sessions, as well as the overall Course Loading view for the benefit of improving the planning process. Advisors will also be able to add new students, as well as update and adjust student’s data as needed.

*<Provide a short description of the software being specified and its purpose, including relevant benefits, objectives, and goals. Relate the software to corporate goals or business strategies. If a separate vision and scope document is available, refer to it rather than duplicating its contents here.>*

## References

Current ETM documents can be found at: [*https://www.ohio.edu/engineering/engineering-tech/undergradtools/documents.cfm*](https://www.ohio.edu/engineering/engineering-tech/undergradtools/documents.cfm)

*<List any other documents or Web addresses to which this SRS refers. These may include user interface style guides, contracts, standards, system requirements specifications, use case documents, or a vision and scope document. Provide enough information so that the reader could access a copy of each reference, including title, author, version number, date, and source or location.>*

# Overall Description

## Product Perspective

This product is a web application designed to replace the manual system currently used by ETM advisors. It will allow advisors to update a student’s course plan, view their current program of study and program planner, and also provides an estimate for the number of students planning to take an ETM course in future semesters.

This data and backend for this software will be implemented using an SQL-based database hosted on an external server that will hold all information pertaining to the ETM program and necessary components of the web application. Additionally, it will hold relevant student data that will be parsed from their respective DARS reports through the use of a separate Python module.

*<Describe the context and origin of the product being specified in this SRS. For example, state whether this product is a follow-on member of a product family, a replacement for certain existing systems, or a new, self-contained product. If the SRS defines a component of a larger system, relate the requirements of the larger system to the functionality of this software and identify interfaces between the two. A simple diagram that shows the major components of the overall system, subsystem interconnections, and external interfaces can be helpful.>*

## Product Functions

1. Import DARS from PDF and place into database
2. Provide flexibility for future adding or removing of courses and/or programs of study
3. Provide three separate views
   1. Home Page
   2. Course Loading View
   3. Program Planner View
   4. Program of Study View
4. Layout the student’s courses for the entirety of their college career
5. Provide security to protect students data and comply with FERPA guidelines
6. Provide an estimate of the number of students taking a course for a future semester

*<Summarize the major functions the product must perform or must let the user perform. Details will be provided in Section 3, so only a high level summary (such as a bullet list) is needed here. Organize the functions to make them understandable to any reader of the SRS. A picture of the major groups of related requirements and how they relate, such as a top level data flow diagram or object class diagram, is often effective.>*

## User Classes and Characteristics

1. Administrative Advisors
   1. High privilege level and will be able to add and remove other advisors, see all student information, and change ownership.
   2. Expected to use the application on a semi-frequent basis
   3. Not required to have high technical expertise
   4. Highly important to satisfy
2. Advisors
   1. Medium privilege level and will be able to access information pertaining to the students they advise
   2. Expected to mostly use the application during advising week and the Course Loading view during planning periods
   3. Not required to have a high technical expertise
   4. Highly important to satisfy
3. Developers
   1. Highest privilege level and will be able to access all functions of the application, database, and web server
   2. Expected to use for regular maintenance
   3. Required to have high technical expertise
4. Possible Student User class
   1. Low privilege level and will only be able to access their own information
   2. Expected to use before scheduling courses and during advising week
   3. Not required to have a high technical expertise
   4. Less important to satisfy implementation of this user class

*<Identify the various user classes that you anticipate will use this product. User classes may be differentiated based on frequency of use, subset of product functions used, technical expertise, security or privilege levels, educational level, or experience. Describe the pertinent characteristics of each user class. Certain requirements may pertain only to certain user classes. Distinguish the most important user classes for this product from those who are less important to satisfy.>*

## Operating Environment

Environments will include desktop web browsers on any PC or device with browser compatibility with Google Chrome and/or Firefox Quantum. For optimal performance computers that use the application should have the latest version of each browser installed.

*<Describe the environment in which the software will operate, including the hardware platform, operating system and versions, and any other software components or applications with which it must peacefully coexist.>*

## Design and Implementation Constraints

A prototype is required by the advising week of Spring semester 2018-2019. Upon completion of the Spring 2018-2019 semester, the client is required to provide their own web server and provide their own maintenance of the application.

The database for this application is required to not waste space with duplicate information.

This application is required to run at relatively fast speeds due to how short advising sessions are.

FERPA guidelines must be met due to the handling of student information.

*<Describe any items or issues that will limit the options available to the developers. These might include: corporate or regulatory policies; hardware limitations (timing requirements, memory requirements); interfaces to other applications; specific technologies, tools, and databases to be used; parallel operations; language requirements; communications protocols; security considerations; design conventions or programming standards (for example, if the customer’s organization will be responsible for maintaining the delivered software).>*

## User Documentation

A paper user manual is required to explain in detail how the application works and to assist in providing maintenance in the future.

A help page is located on the website so any user can access it easily at any time.

*<List the user documentation components (such as user manuals, on-line help, and tutorials) that will be delivered along with the software. Identify any known user documentation delivery formats or standards.>*

## Assumptions and Dependencies

A user is expected to have prior knowledge of how courses work at Ohio University as well as how to use a common web interface, in particular for completing web forms.

This application depends on both a database and web server. Foreseeable issues include connection with the Ohio University mainframe, as access to Single-Sign-On (SSO) will require requests and approval through OIT.

A prototype of the project will be hosted and presented through an Amazon Web Services instance, but this will last a year and the client must create their own account and server through AWS if they would like to continue using it.

The necessary components (web application files, database structure, and DARS parser) will be handed over to the client to host and run as desired.

*<List any assumed factors (as opposed to known facts) that could affect the requirements stated in the SRS. These could include third-party or commercial components that you plan to use, issues around the development or operating environment, or constraints. The project could be affected if these assumptions are incorrect, are not shared, or change. Also identify any dependencies the project has on external factors, such as software components that you intend to reuse from another project, unless they are already documented elsewhere (for example, in the vision and scope document or the project plan).>*

# External Interface Requirements

## User Interfaces

This web application will consist of a number of main pages/views, all of which will follow standard online form and web application style:

1. A login screen - The same screen is seen by all users, allowing secure access to the application.
2. Home screen -- Advisors can view a list of their advisees, from which they can select a student and see more information. Administrative advisors can see a list of all advisors and advisees, of which they can also select in order to view more.
3. Program of Study -- This is designed to replace the current ETM advising sheet, showing all of the courses necessary for an ETM student to complete. There is a unique Program of Study view for each student, showing which courses have already been completed, including the grade and semester taken.
4. Program Planner -- This is designed to show which courses a student has taken and recommends when to enroll in the rest of the courses that must be taken. It is unique to each student and gives a prospective graduation date.
5. Course Loading -- This is a table consisting of ETM courses, projecting the number of students expected to enroll by year and semester through the next four years. The semester shown can be selected by the user from a dropdown menu.
6. Student/Advisor Administration -- This allows for Administrative advisors to add/delete/modify student and advisor accounts and data.

*<Describe the logical characteristics of each interface between the software product and the users. This may include sample screen images, any GUI standards or product family style guides that are to be followed, screen layout constraints, standard buttons and functions (e.g., help) that will appear on every screen, keyboard shortcuts, error message display standards, and so on. Define the software components for which a user interface is needed. Details of the user interface design should be documented in a separate user interface specification.>*

## Hardware Interfaces

Currently there are no hardware interfaces in place.

*<Describe the logical and physical characteristics of each interface between the software product and the hardware components of the system. This may include the supported device types, the nature of the data and control interactions between the software and the hardware, and communication protocols to be used.>*

## Software Interfaces

Currently the only software advising tool is the Ohio University DARS reports, but due to lack of access, the application requires a pdf version of a DARS to be uploaded. The DARS are parsed by a module that communicates with the application database which is hosted with MySQL. The parser is a separate module so that if the DARS are changed in the future, the parser can be updated/replaced without affecting the application.

*<Describe the connections between this product and other specific software components (name and version), including databases, operating systems, tools, libraries, and integrated commercial components. Identify the data items or messages coming into the system and going out and describe the purpose of each. Describe the services needed and the nature of communications. Refer to documents that describe detailed application programming interface protocols. Identify data that will be shared across software components. If the data sharing mechanism must be implemented in a specific way (for example, use of a global data area in a multitasking operating system), specify this as an implementation constraint.>*

## Communications Interfaces

The web application is required to interface with standard web browsers, be accessible through HTTP, and communicate consistently with the database and hosting web server.

FTP is required to upload student information via their DARs, which is executed by upload of DARs PDFs through the web application.

*<Describe the requirements associated with any communications functions required by this product, including e-mail, web browser, network server communications protocols, electronic forms, and so on. Define any pertinent message formatting. Identify any communication standards that will be used, such as FTP or HTTP. Specify any communication security or encryption issues, data transfer rates, and synchronization mechanisms.>*

# System Features

*<This template illustrates organizing the functional requirements for the product by system features, the major services provided by the product. You may prefer to organize this section by use case, mode of operation, user class, object class, functional hierarchy, or combinations of these, whatever makes the most logical sense for your product.>*

## Web Interface

* + 1. Description and Priority

Priority: HIGH

This is how advisors access the software. It is through the web interface that advisors are able to view student’s Programs of Study, Program Planner, and other relevant information. It is also how they are able to enter and update course information, grades, and all other data relating to student advising.

* + 1. Stimulus/Response Sequences

This is the reading and writing of data by advisors which includes but is not limited to:

* + Entering/updating grades
  + Entering/updating courses
    1. Functional Requirements

The web interface is required to be maintainable and easy for user to understand.

## Database

* + 1. Description and Priority

Priority: HIGH

This is where all data related to the ETM major, relevant courses, and students is stored.

* + 1. Stimulus/Response Sequences

Requesting and writing of data.

This consists of developer adjustments to ETM major requirements and course information.

Additionally, it consists of advisors and ETM administrators requesting and updating data such as relevant student information.

* + 1. Functional Requirements

The database is required to be a reliable source of data for this application. It also must have the capacity to hold all of the required data and have the ability to be maintained in the future.

## PDF DARS Parsing Module

* + 1. Description and Priority

Priority: HIGH

This module allows administrators to quickly and effectively add all relevant student information to the database without the need for manual entry.

* + 1. Stimulus/Response Sequences

This module accesses a local folder containing all ETM student DARS reports in the form of PDFs. It parses relevant data, which is then uploaded to the software database, either in the form of new student entries, or updating of old information.

* + 1. Functional Requirements

The DARS parsing module is required to be a separate module so that updates in the future d not affect the rest of the application. This must be a reliable way to parse data from the current DARS and perform well even when given a large file of DARS.

Description and Priority

*<Provide a short description of the feature and indicate whether it is of High, Medium, or Low priority. You could also include specific priority component ratings, such as benefit, penalty, cost, and risk (each rated on a relative scale from a low of 1 to a high of 9).>*

Stimulus/Response Sequences

*<List the sequences of user actions and system responses that stimulate the behavior defined for this feature. These will correspond to the dialog elements associated with use cases.>*

Functional Requirements

*<Itemize the detailed functional requirements associated with this feature. These are the software capabilities that must be present in order for the user to carry out the services provided by the feature, or to execute the use case. Include how the product should respond to anticipated error conditions or invalid inputs. Requirements should be concise, complete, unambiguous, verifiable, and necessary. Use “TBD” as a placeholder to indicate when necessary information is not yet available.>*

# Other Nonfunctional Requirements

## Performance Requirements

This product is required to perform at a quick speed due to the fact that advising meetings are short. The response time, however, does depend on the network speed of the client and server.

*<If there are performance requirements for the product under various circumstances, state them here and explain their rationale, to help the developers understand the intent and make suitable design choices. Specify the timing relationships for real time systems. Make such requirements as specific as possible. You may need to state performance requirements for individual functional requirements or features.>*

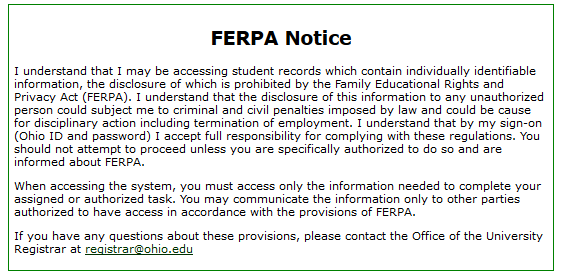
## Safety Requirements

There is a possibility of data loss due to errors in network traffic. To safeguard against this, it is highly recommended to use a web server that will back-up data regularly.

*<Specify those requirements that are concerned with possible loss, damage, or harm that could result from the use of the product. Define any safeguards or actions that must be taken, as well as actions that must be prevented. Refer to any external policies or regulations that state safety issues that affect the product’s design or use. Define any safety certifications that must be satisfied.>*

## Security Requirements

All data must be protected and secure by the guidelines of FERPA. This product is required to meet the current security as used by Oasis (the current advising center).

**

*<Specify any requirements regarding security or privacy issues surrounding use of the product or protection of the data used or created by the product. Define any user identity authentication requirements. Refer to any external policies or regulations containing security issues that affect the product. Define any security or privacy certifications that must be satisfied.>*

## Software Quality Attributes

This software must be adaptable to the future needs of advising. The software must be easy to use if the user meets the required specifications in 2.7 Assumptions and Dependencies.

*<Specify any additional quality characteristics for the product that will be important to either the customers or the developers. Some to consider are: adaptability, availability, correctness, flexibility, interoperability, maintainability, portability, reliability, reusability, robustness, testability, and usability. Write these to be specific, quantitative, and verifiable when possible. At the least, clarify the relative preferences for various attributes, such as ease of use over ease of learning.>*

## Business Rules

Administrative advisors are able to remove other advisors and change formatting of the web pages including adding/removing courses on top of the regular advisor access.

Advisors have a role where they are able to enter students grades, add new students with DARS, and update their course plan.

If students are implemented as users, they will have a read-only view of their own data.

*<List any operating principles about the product, such as which individuals or roles can perform which functions under specific circumstances. These are not functional requirements in themselves, but they may imply certain functional requirements to enforce the rules.>*

# Other Requirements

There are currently no additional requirements for this project.

*<Define any other requirements not covered elsewhere in the SRS. This might include database requirements, internationalization requirements, legal requirements, reuse objectives for the project, and so on. Add any new sections that are pertinent to the project.>*

**Appendix A: Glossary**

SRS currently has no need for a Glossary, but terms will be added upon request.

*<Define all the terms necessary to properly interpret the SRS, including acronyms and abbreviations. You may wish to build a separate glossary that spans multiple projects or the entire organization, and just include terms specific to a single project in each SRS.>*

**Appendix B: Analysis Models**

Credits for the Web-server diagram:

<http://javascript-coder.com/html-form/how-to-make-a-form.phtml>

*<Optionally, include any pertinent analysis models, such as data flow diagrams, class diagrams, state-transition diagrams, or entity-relationship diagrams*.>

**Appendix C: To Be Determined List**

*<Collect a numbered list of the TBD (to be determined) references that remain in the SRS so they can be tracked to closure.>*