

**2014**

**University of North Carolina Wilmington  
Master of Science in  
Computer Science and Information Systems  
Proceedings**

**<https://csbapp.uncw.edu/mscsis>**

ANALYSIS AND IMPLEMENTATION OF A REPORTING SYSTEM FOR  
GRADUATE PROGRAM MANAGEMENT

Linda Oxendine Dance

This is a Capstone Project to be submitted to the University of North Carolina at Wilmington in  
Final Fulfillment of the Requirements for Degree of Master of Science in  
Computer Science and Information Systems

Department of Computer Science and  
Department of Information Systems and Operations Management

University of North Carolina at Wilmington

2014

Committee

---

Dr. Douglas Kline, Chair

---

Dr. Ron Vetter, Professor

---

Dr. Bryan Reinicke, Associate Professor

---

Ms. Karen Barnhill, Cameron School  
of Business Graduate Coordinator

**Abstract:**

A database and reports are created to help manage the Master of Science in Computer Science and Information Systems program at University of North Carolina at Wilmington. The current system provided by the university registrar is limited to reporting only required coursework for individual students. The new database incorporates the required coursework, as well as program pre-requisites and capstone project/thesis progress. Per-student reports now integrate all requirements of the program in one place. In addition, program-level reports display information aggregated across students, such as counts of students at different stages of the capstone. The improved database and new reporting capabilities improve operational decision making, as well as reporting for program review and accreditation purposes.

## Table of Contents

Part I: PROJECT ANALYSIS AND PLANNING.....	8
1 Motivation for Project .....	9
2 Importance of Graduate Program Database.....	13
3 Importance of Project .....	14
4 System Analysis .....	15
Part II: IMPLEMENTATION.....	27
5 Project execution .....	28
6 Conclusion.....	33
7 Works Cited.....	35
8 APPENDIX .....	36
APPENDIX A – Project Charter.....	37
APPENDIX B – Program Admissions and Degree Requirements.....	40
APPENDIX C: Project Staging .....	44
APPENDIX D: Database Diagram – Basic Model (version prior to this project).....	45
APPENDIX E: Database Tables (prior to this project) .....	46
APPENDIX F: Database Diagram of Program Requirements prior to this project.....	47
APPENDIX G: Program Prerequisites already loaded from another project.....	48
APPENDIX H: Program Prerequisites Satisfying Course is already loaded from another project .....	49
APPENDIX I: Data Model Overview for Design and Development .....	50
APPENDIX J: Logs of Meetings (this list is not all inclusive) .....	51
APPENDIX K: Emails to get the account set up on the UNCW production SSRS server .....	52
APPENDIX L: Screen shot of Cisco VPN login:.....	54
APPENDIX N: Database Diagram: Program Requirements .....	56
APPENDIX O: Original scope of project was Capstone Project Management System.....	57
APPENDIX P: Scope of project for Tracking and Reporting System.....	58
APPENDIX Q: Sample Reports .....	59
APPENDIX R: List of Views Created, those with naming convention ld*** and some of the attributes.....	61

APPENDIX S: List of Reports Created .....	76
APPENDIX T: Reports built during learning process and before final edits and deployment to the Report Server .....	87
APPENDIX U: Query Designer for Potential Graduation Report.....	88
APPENDIX V: Banner System Degree Audit.....	89
APPENDIX W: Tables added during this project, with the naming convention of ld_*** .....	90
APPENDIX X: Data Model for Program Required Course Fulfillments.....	91
APPENDIX Y: Prerequisite Courses already loaded prior to project .....	92

**Introduction:**

The University of North Carolina at Wilmington offers students a unique and prestigious opportunity to enhance skills in a well-designed program for a Master’s Degree in Computer Science and Information Systems (MS CSIS). The periodic Program Review lauds the success of the program and the graduating students’ performance and subsequent job opportunities.

Students with Bachelor’s degree in Computer Science or Information Systems, or a Bachelor’s in a related discipline with skills in Computer Science and Information Systems, can apply for acceptance into the program. In evaluating a prospective student’s level of knowledge coming into the program, a list of prerequisites are reviewed by the acceptance committee to outline which of those prerequisites the individual student must complete before being accepted into the program or complete before reaching 18 course hour threshold of Master’s level courses in the program curriculum.

The degree requirements include:

- 6 core courses            18 hours
- 4 elective courses       12 hours
- Capstone/Thesis        6 hours

A total of 36 hours is needed to complete the program.

The problem to solve with this project is to develop and implement an easy single-source tracking and reporting system for the MS CSIS program and students for program prerequisites, core courses, elective courses and capstone/thesis course work. This is done by establishing the processes of systematic data entry, tracking, and reporting on the statuses of prerequisites, core, elective and capstone courses for graduate students in the program from acceptance to program completion. This tracking will include credit hours and completion status in a persistent state, with no workflow. Credit hours will not be tracked for prerequisites.

The program is in need of a formal system of reporting services. In addition, a well-defined data model provides the foundation for building views to facilitate reporting services and ad hoc reporting. Standardized reports will provide administrators with tools to identify:

- The statuses of prerequisites, whether they are incomplete, completed, waived, or otherwise, by student
- Students who have applied for graduation but still have prerequisites to complete
- Program level trends and requirements for operational decision making such as course offerings and staffing
- Statistical data reporting and presentation for program evaluation and accreditation.

Stakeholders are the UNCW MS CSIS Coordinator, the Cameron School of Business Graduate School Administrators, MS CSIS graduate students, and MS CSIS academic advisors.

This project is divided into 2 phases:

- 1) Project Analysis and Planning
  - a) Explore the needs of the program and consider the appropriate technologies and applications.
  - b) Connect to the UNCW Cameron School of Business production database and the reporting services. This will involve setting up a client machine to access the production database, using Microsoft Visual Studio 2008 and SQL Server 2008 R2 with the Business Intelligence module to utilize SQL Servers Reporting Services (SSRS).
  - c) Consider various standardized reports to meet the needs of the project stakeholders.
  - d) Conduct stakeholders' interviews
  - e) Create a project charter
  - f) Create actor diagram to understand the roles and responsibilities of stakeholders
  - g) Develop a data model

- h) Load production data
- 2) Implementation
  - a) Build and test reports
  - b) Deploy Reports to Reports Server
  - c) Success of the project:
    - Alignment of reports with stakeholder needs
    - Accuracy and completeness of data and reports
    - Level of satisfaction of the stakeholders
    - Ability of reports to aid in program evaluation

**Part I: PROJECT ANALYSIS AND PLANNING**

## **1 Motivation for Project**

The Graduate office needs a mechanism for tracking students' progress in the program from application to graduation. There are some limitations with the University Registrar's current system used for tracking, which is Banner. Banner does not have reporting capabilities necessary to be proactive in targeting shortfalls or obstacles for students' course scheduling and preparedness for graduation. The Degree Audit in Banner is the only tool for such tracking currently and this is after the fact and does not alert students or the Graduate office of these shortfalls. Eventually, the new system could be expanded to have an interface for students and alumni.

Also, this new system will add in program course planning and program evaluation.

The final system should have:

- Usability
- Adaptability
- Extensibility

The project has evolved over a period of one and a half years due to challenges in project scope. Working with Dr. Kline and Ms. Barnhill during the MS CSIS program review as a Graduate Assistant, these needs became very apparent. Originally, when considering a total system delivery, the scope proved to be too broad. The system analysis uncovered how large the system could be including:

- Size of project with all the moving parts of the database development
- Technologies to be used for different aspects of the overall system

- Family Educational Rights and Privacy Act of 1974 (FERPA), which provides privacy of educational records for students (Wikipedia, 2014)
- User Login screens
- User input screens
- Reports for Students and Alumni
- Messaging system
- Monitoring and Auditing
- User profiles and security

The scope had to be pared down to be manageable. During the year and one half long incubation period, other students had begun working on different elements of the overall system. A rudimentary database was set up with courses, persons, students and other tables loaded. There is a concurrent project, which focuses on Capstone projects, which is almost complete. It was decided to limit the scope of this project to an easy simple tracking and reporting system to fill in the gaps that are currently not fulfilled by the Banner system.

In summary, the problem:

- Reports on the statuses of Prerequisite courses and Capstone projects
- Deployment on production servers
- Pre-selected aggregate and detailed reporting
- Platform for future reports

The solution is to provide a system with:

- Expand the current data model to capture data to tracked and reported
- Pre-selected reporting, aggregate and detailed
- Ad hoc reporting

- An Extended Degree Audit that integrates the statuses of Prerequisite courses and Capstone projects

One of the stakeholders, Ms. Karen Barnhill, the Coordinator for the Cameron School of Business, manages tracking reporting of data for three graduate programs:

- Master of Science in Business Administration (MBA)
- Master Science in Accountancy (MSA)
- Master of Science in Computer Science and Information Systems (MS CSIS)

This project is focused solely on the MS CSIS program but will be extendable for usability in the other graduate programs

The MS CSIS program is a interdisciplinary program for Computer Science and Informations Systems. It offers advanced study for those with Bachelor degrees in Computer Science or Information Systems and as well as a secondary group of professions with degrees in other related fields, such as business, engineering, accounting, for example. Refer to Appendix B for broader details of the program.

There are 10 Prerequisites and with approval, student can fulfill some of these while in the program. Students must complete all prerequisites before reaching an 18 hour threshold of credits completed in the program graduate core and elective courses. This presents a primary need for data warehousing and reporting services to track and report progress and compliance. See Appendix S for the first report created and develop for this project. The ten prerequisite courses are:

- Two programming courses
- Data structures
- Database

- Software engineering
- System analysis and design or data communications or networking
- Financial accounting
- Marketing
- Finance
- Management

Core courses for the MS CSIS are as follows:

- CSC 532 - Design and Analysis of Algorithms I Credits: (3)
- MIS 534 - Information Security Management Credits: (3)
- CSC 544 - Network Programming Credits: (3)
- CSC 550 - Software Engineering Credits: (3)
- MIS 555 - Database Management Systems Credits: (3)
- MIS 565 - Analysis, Modeling and Design Credits: (3)

See Appendix B. The Banner Degree Audit has this line item and the user can drill down to get the details. This project includes this same presentation and behavior for the Extended Degree Audit. See the line item on the Banner report in Appendix V.

The program offers many non-core courses in computer science and information systems as elective courses, for which the student must fulfill 12 to 15 hours. These courses include such topics and areas of study in project management, artificial intelligence and biometrics. See Appendix B, in the section for Degree Requirements.

Students can choose a research project with 3 hours credit for a thesis for 6 hours to serve as the Capstone “experience”. See Appendix B.

The term “non-standard” was chosen in the development of this project describe the other opportunities the offers for course fulfillment towards earning the degree. These course studies extend the students’ learning beyond standard course offerings:

- CSC or MIS 591 – Directed Independent Study on selected topic, skill, or technology
- CSC or MIS 595 – Research Seminar on selected topic
- CSC or MIS 598- Internship

No more than 9 hours of these non-standard course hours can be applied for degree fulfillment. There is only a periodic review of these hours by the academic advisors and the graduate office. Banner does not provide this compliance testing. This project will provide this reporting and compliance testing for this, which is another high value feature of this project.

## **2 Importance of Graduate Program Database**

The scope of this project is to fulfill the needs of the Graduate Office Coordinator, Academic Advisors and Student for tracking and reporting students’ requirement. To be accurate, a system was needed with additional reporting functionality beyond the Banner system.

The development and maintenance graduate program database is paramount to ensuring the high value reporting, filling the shortfalls of the Banner system and replacing the current system of paper files and Excel files. This tracking and reporting system’s planning, development and execution are simple and easy to follow.

Another important aspect of this database is the ad hoc reporting capabilities for tracking historical data and trends. The data model planning is discussed on section 4.3 and the development is discussed in section 6.11. These discussions convey that the data model process

has been thoughtful and iterative, while being built concurrently with other MS CSIS graduate school database projects.

Pitfalls that can be avoided with tracking and reporting but not currently addressed in Banner System:

- Student expects to graduate but still has a prerequisite to fulfill
- Student expects to graduate in a specific term but a class needed is not offered for the term that they need it
- Student has more hours of non-standard course credits than 9 hours

The tracking and reporting system will be updating as program requirements and course offerings change.

Program administrators can make informed decisions on data and trends for program performance evaluation and planning.

### **3 Importance of Project**

This system provides a foundation for tracking and reporting by delivering an easy solution for:

- **Efficiency**

The reporting system will replace the system of paper files and Excel worksheets.

- **Consistency**

The reporting system will provide a foundation for consistent updating and reporting.

- **Quality**

The Banner Degree Audit is a good source for error checking, also, other sources are reviews by students and academic advisors.

#### **4 System Analysis**

Initially, use cases, events and other aspects of system analysis were explored, but the scope was defined by the following declarations:

- No workflows
- Simple database design, with views, using the Student Id as the key for this system
- No custom user interface or forms
- Data entry directly into database
- Prerequisite courses are track by number count, not by credit hours.
- Standard, simple architecture

This very basic system provides the foundation for a larger, more granular system, if needed, with triggers and messaging, for example.

The simple architecture consists of deployment on the UNCW production SSRS server for reporting purposes and use of the Cameron School of Business MSCSIS production database. See the figure below representing the architecture.

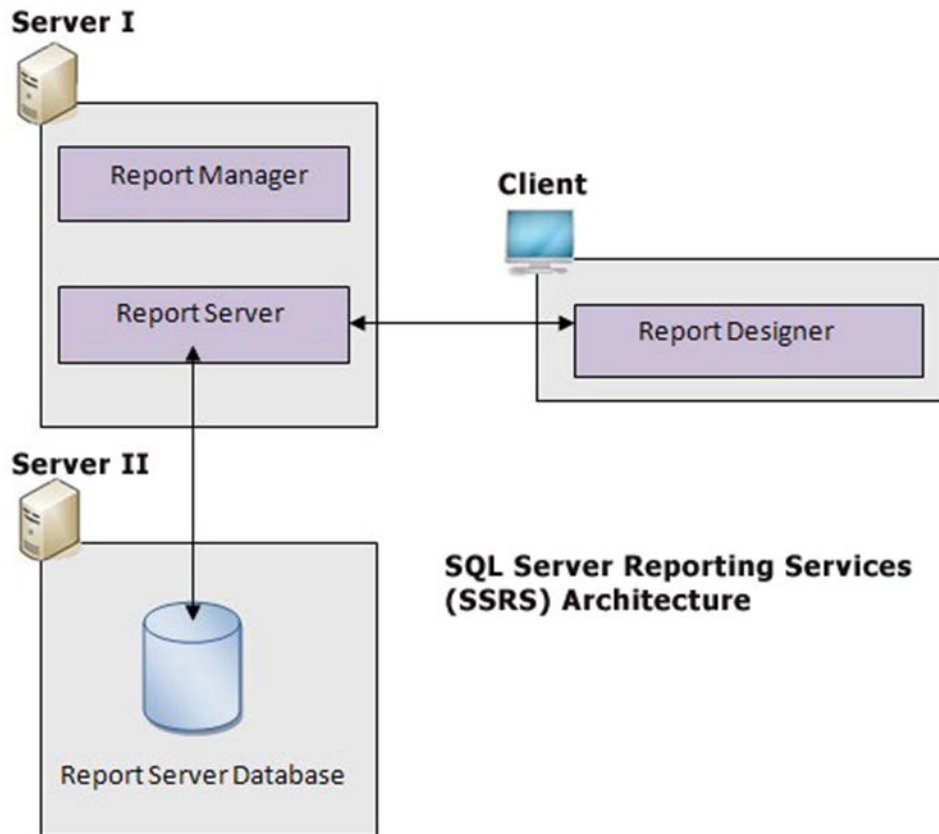


Figure 1: Simple architecture. Server I is the UNCW SSRS server, Server II is the CSB MSCSIS production database that contains the MS CSIS graduate school database.

See Appendix K for the email in which Dr. Kline requested that access to the UNCW production SSRS server for proper credentialing to access reporting services for reports creation, deployment, editing and deletions.

#### 4.1 Project Charter

##### Purpose

Provide easy single-source reporting for MS CSIS program and students for Program prerequisites, core courses, elective courses and Capstone/thesis course work.

##### Objectives

- Improve decision making by program director, coordinator, department administrators, and students
- Improve management of program data (backup, maintenance, auditability)
- Save time for program director and coordinator
- Improve conversion of data to information (access and reporting)
- System Analysis for Data Model, Data Entry, and Reporting Processes

### **Stakeholders**

- Ms. Karen Barnhill, Graduate Program Coordinator
- Dr. Douglas Kline, MS CSIS Program Director
- Department Administrators
- Academic Advisors
- Students

### **Desired Features**

- Single Relational Database on Enterprise Database Server
- Select pre-written program-level reports
- Ad-hoc reporting capabilities
- Data Entry Forms for Program Coordinator
- Ability to adapt to other graduate programs

### **Critical Success Factors**

- Well-designed relational data model

- Direct data entry into production database
- Program Coordinator buy-in
- Functional reporting capabilities
- Adaptability and scalability

### **Constraints**

- Time
- Integration/Separation from Existing Systems
- Personnel Experience – novice to intermediate developer
- Project completion by April 15, 2014

### **Deliverables**

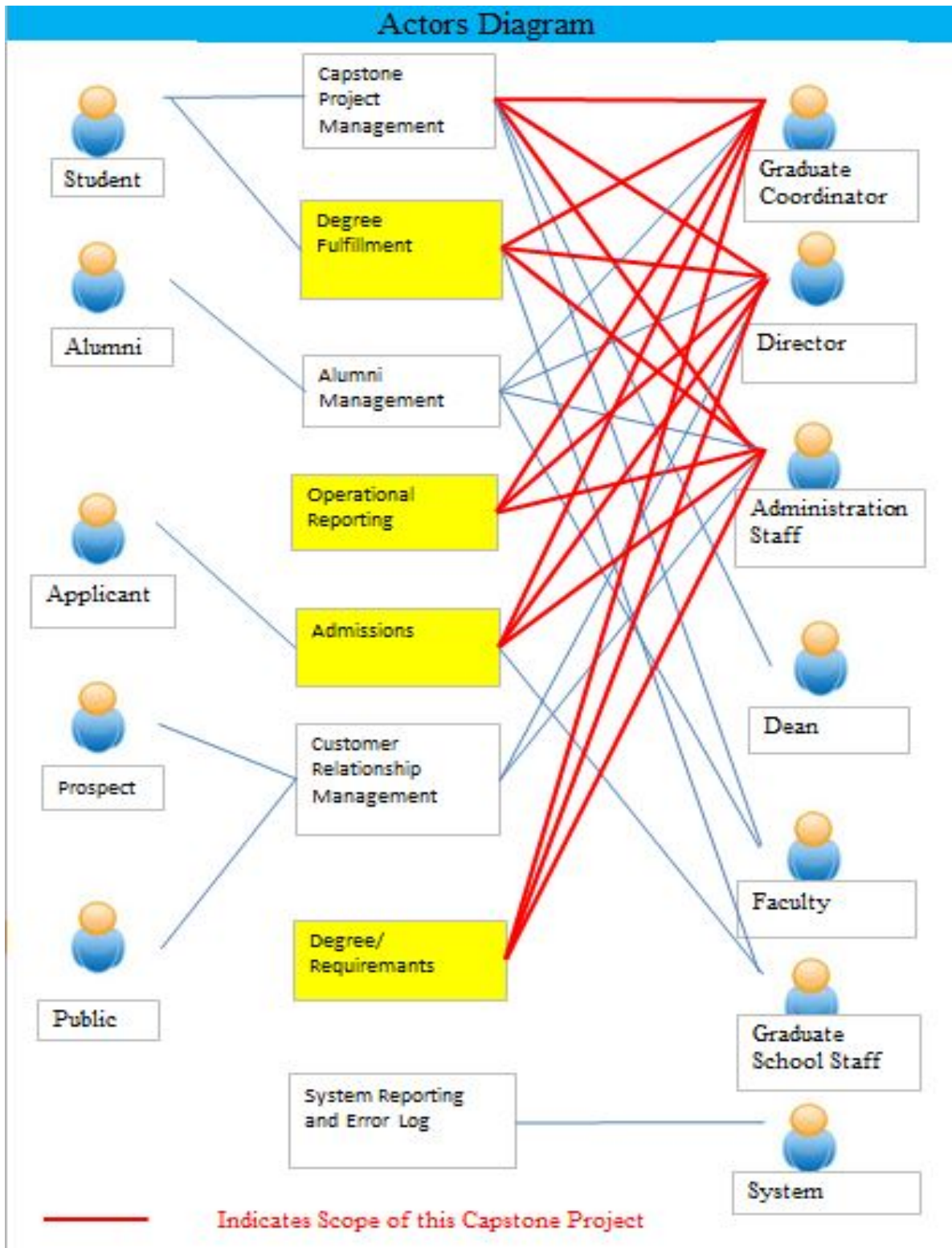
- 1) Design system for tracking and reporting of program requirements
- 2) Develop a well-defined data model
- 3) Migrate current data into new system
- 4) Create standardized reporting system
- 5) Specific reports to be created:
  - a) Total number of students enrolled by term
  - b) Credit hours produced by term/course/department
  - c) Yield by term
  - d) Acceptances by gender
  - e) Admissions by term
  - f) Graduates by term
  - g) Time to graduate

- h) List of students reaching 18 course hour threshold of Master's level courses in the program curriculum but have not completed their prerequisites

## **4.2 Stakeholder Interviews**

Meetings with held with Ms. Barnhill and Dr. Kline during the spring of 2012, discussing the need to develop electronic forms for tracking students' progress for Capstone, prerequisites and other graduate school events. This project originally focused on Capstone but later the focus and scope focus on the course and degree tracking.

Conversations lead to the incremental possibilities of a graduate database and how future enhancements could facilitate a customer relationship management (CRM) for students and alumni. Look at the Actor's Diagram in Figure 1 to see some of the possibilities on a larger scale.



**Figure 1: Actor Diagram**

This project focuses on those processes for degree fulfillment, operational reporting, initial admissions and degree and other requirements.

### **4.3 Data Model**

The states of data model and database tables prior to project are exhibited in Appendices D, E, F, G and H.

The goal of the data model is to be useful at the program level but extendible to other programs. It must provide high value for querying, ad hoc reporting and reporting services. See figure 2 for the model representing the overview of the components of the data model and the iterative nature of the generating and reviewing outcomes while providing the structure for other programs to use the database.

Upon inspection of the data model already in place, the focus appears to be on student events and student statuses. Student fulfillment events to be entered into the production database with either Microsoft Access or SQL Server within the scope of this project are:

- New persons, faculty, students, committee members
- Admissions
- Prerequisites
- Fulfillment of Prerequisite, Core, Elective, Non-standard and Capstone courses (non-standard, meaning an elective course, herein)

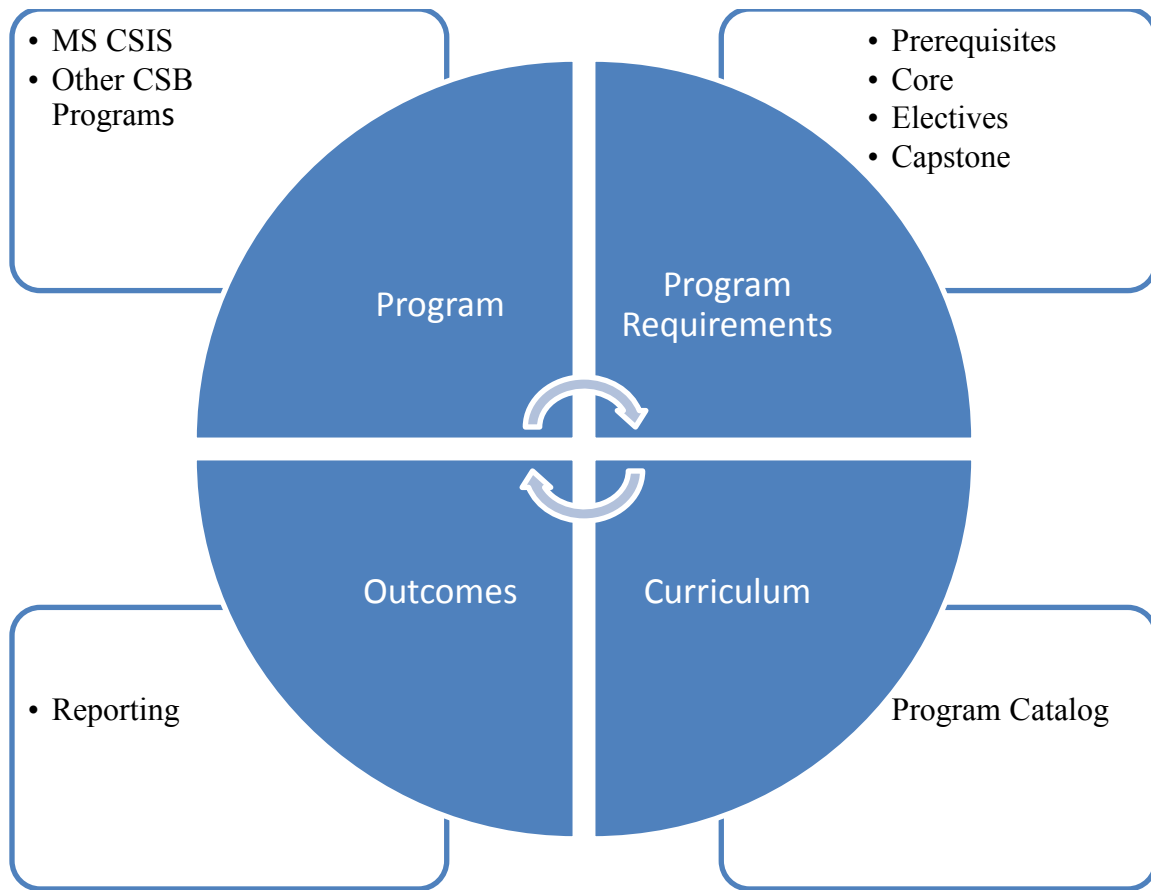


Figure 2: Data Model Overview for Design and Development

The first order for this task is to identify the tables that already exist, capture its current state and then to build on that base model to include tables for and links to the program requirements, tracking attributes, and requirement fulfillment.

Project Features:

- No forms
- No User Interface, direct entry into database through Microsoft Access or SQL Server
- Reporting Services

### Highlights of project:

- Data loading: only data not related to FERPA
- Integrity and simplicity of design
- Data warehouse
- Track Student status with Extended Degree Audit
- Reporting Services

The scope of this project was to develop an easy simple tracking and reporting system to fulfill the shortfalls of the Banner System. This project runs concurrently with other graduate projects using the MS CSIS database.

The project was originally discussed with Dr. Kline in November, 2011 with a focus on Capstone reporting. The MS CSIS program review was conducted in the spring of in 2012. I provided some administrative support for Dr. Kline, as a Graduate Assistant, in compiling statistics for the program review. During this time, discussions continued regarding the program's need for a single database and reporting system. I began working on the Project Charter, Actor Diagram, and other related documents and analyses in 2012. Refer to Appendices O and P to see the changes in the scope of the project from 2012 to 2013. Refer to Appendix A for the final Project Charter in 2013 and Appendix J for a log of meetings.

The resources for this project included:

- Dr. Douglas Kline: provided project guidance, mentoring, project development and learning opportunities.
- Mr. Uche Iheadindu: fellow graduate student who provided guidance on SSRS and Business Intelligence.

Ms. Lori Speakman: UNCW System Administrator, set up credentials and folder on the UNCW SSRS server

#### **4.4 Learning Curve**

There was a slow start on the project due to the learning curve while performing the following tasks:

- Review of the existing data model and database to determine what had been set up and what data had been entered. The data model was expanded and tables and views were added to the database to provide the foundation for high value reporting.
- Setup VPN connection through Cisco AnyConnect Secure Mobility Client with instructions from the UNCW Technology Assistance Center (TAC). Refer to Appendix L for a screenshot of the login screen.
- Getting connected to the CSB MSCSIS database,
- Getting set up with the credentials and folders on the UNCW productions SSRS server for this project. See the email correspondence in Appendix K.
- Working with production database, data modeling had to be more thoughtful, deliberate and concise to support consistency and efficiency and to enhance future work.
- Learn SSRS

#### **4.5 Constraints**

There has been a long incubation period for the project, but short period for analysis, planning, development and execution. Refer to Appendix C for the project staging. There is an upside to this, in that other students have concurrent project so the overall system is being developed in a production environment.

#### **4.6 Success Criteria**

A simple reporting system has been delivered as proposed, providing

- Well-designed relational data model
- Direct data entry into production database
- Program Coordinator buy-in
- Functional reporting capabilities
- Adaptability and scalability

#### **4.7 System Architecture and Technologies**

This system will be designed using the Cameron School of Business (CSB) production server which is interfaced with a Report Server on the same network. Visual Studio 2008 will be utilized as the integrated development environment. Visual Studio 2008 was chosen because the Business Intelligence Development Services (BIDS) Integration Service, Report Services and Analysis Services projects for SQL Server 2008 and SQL Server 2008 R2. Visual Studio 10 does not support the BIDS and other services for SQL 2008. (Microsoft) A data warehouse will be created on the CSB production server named, CSBSQL. SQL Server Reporting Services (SSRS)

#### **4.8 Project Staging**

This project was finally scoped out in the fall of 2013. See Appendix C for the project staging from the fall of 2013 forward. See that much thought had been put into project and the momentum picked up in. The final 8 weeks were compressed with finishing the database and building the reports.

## 4.9 Data Model

The data model and data entered were analyzed and development began with identifying the types of courses and how to track these.

The table, `ld_PrerequisiteFulfillment` is used to track the number of prerequisites fulfilled. The prerequisites were already loaded into the database prior to this project. Refer to Appendix Y. A record for a fulfilled prerequisite is to be entered for each student for each prerequisite as fulfilled. In the absence of a “fulfillment” record, a report can be generated to show which prerequisites are unfulfilled by student or in aggregate.

Similarly, the required program courses are tracked by using data in the `ld_ProgramRequiredCourseFulfillment` table but credit hours are tracked. Refer to the Fulfillment Diagram, in Appendix X.

The data model evolved with each report that was created. Data tables were added as seen in Appendix W and views were added as seen in Appendix R.

The complexity of the views that were built in this data model for aggregate reporting are covered in section **5.2 Design and Development of Reports** in the paper.

## 4.10 Reports

Refer to Appendix Q for several sample reports proposed at the beginning of this project.

## **Part II: IMPLEMENTATION**

## **5 Project execution**

### **5.1 Data Loading**

Some data was loaded into the tables at the beginning of this project. This project did entail entering some non-confidential type information, such as prerequisites fulfilled by current students. Graduate data for the author of this project was entered by the author to compare and verify data and reports against the Banner Degree Audit.

### **5.2 Design and Develop reports**

As reports were created, views were needed. These views hold containers of information, such as GPA or total number of graduate hours, and offer an easier mechanism for reporting functionality. Refer to Appendix S to review the reports built and some of the database challenges to be fulfilled.

List of Views created in this project are listed below and labels are somewhat self-explanatory. See Appendix R:

1. IdAllRequiredCourseAndFulfillments
2. IdCapstoneThesisHoursCompleted
3. IdCommitteeCountsByFaculty
4. IdCoreCoursesCompleted
5. IdCourseGPA
6. IdCreditHoursByTerm
7. IdGradHoursCompleted
8. IdNonCoreCoursesCompleted
9. IdNoOfStudentsByTermYr

10. IdNumberOfCommitteesChairedByPerson
11. IdNumberOfUnfulfilledRequiredCourses
12. IdStudentPrereqcounts
13. IdStudentsAndAllPrereqFulfillments
14. IdStudentsAnAllPrerequisites
15. IdUnfulfilledRequiredCourses

The views developed to facilitate aggregate reporting demonstrate the level of complexity in the data model, please refer to Appendix U, the query designer window for the **Potential Graduation Report**. The report had to be developed over a period of time as the data model was updated. The first aspect to point out is the number of views that were built to capture the desired information. The next aspect is to look at the details of the five views that were built to render the desired output for the report:

- 1) IdCapstoneThesisHoursCompleted: See Appendix R-2 to see how the Capstone/thesis hours completed is calculated.
- 2) IdStudentPrereqCounts
- 3) IdCoreCoursesCompleted: See Appendix R-4 to see how the Core courses completed is calculated.
- 4) IdCourseGPA: See Appendix R-5 to see how the Course GPS is calculated.
- 5) IdGradHoursCompleted: See Appendix R-7 to see how the Graduate hours completed is calculated.

Other reports required similar data modeling to obtain the report results. See Appendix R for other views built and Appendix S for the reports.

### **5.3 Testing**

Testing on real data was limited due to the guidelines of FERPA. However, testing was done based on the author's Banner Degree Audit and Program Department records. Report output is accurate based on author's data and the limited data provided.

### **5.4 Continuous Deployment**

This reporting system is very easy, very basic. As the Graduate Program Coordinator and the Program Director need reports and other ad hoc analytical data, the reporting capabilities can be expanded with very little effort with SSRS.

### **5.5 Summary**

The data model and reporting systems are simple yet sufficient for the needs of the stakeholders, in contrast to the system this project is replacing for the most part, from of Excel and paper files. Also this project has met the need for an Extended Degree Audit, which adds the requirements and fulfillments for Prerequisites and Capstone Projects, which enhances the Degree Audit in the Banner system. The overlapping of the two Degree Audits provides verification of some of the data.

The data model and reporting system provide the foundation for generating statistics, capturing what is happening at a point in time and also, program metrics, the performance indicators.

The challenges and learning opportunities in this project were:

- Learning curve, setting up the connections for the system and installing the necessary tools and applications, but mostly SSRS
- Working in a production environment, ensuring to protect data integrity
- Privacy and confidentiality

- Concurrent projects being developed and implemented in the MSCSIS database, the data model is very dynamic and expanding
- Data model concepts changed to meet the needs of the desired reports
- Developing reports became iterative, with some reports, the data model was revised and views recreated, thus adding functionality
- Report by report, multiple version: some reports are links and subreports of other larger, more complex reports
- Expanding ideas in Reporting System, seeing possibilities but staying within project scope

## **5.6 Reporting Concepts**

The program needs aggregate but also parameterized reporting. This scope of this project is to capture aggregates for program level evaluation but to use the student id as the primary key for the system. In addition to pre-selected reports, the well-defined database is easy to navigate for informative ad hoc reporting.

## **5.7 Types of Reports**

The scope of this project has included the primary emphasis on reporting and exploring different report formats:

- Drill Down – expand to see details at a different level
- Graph
- Subreports – reports within reports

See reports in the Appendix S.

## 5.8 Reports Created and Deployed

List of reports created:

<u>Report Name</u>	<u>Report Type</u>
1. Summary of Prerequisites by Student	Drilldown
2. Prerequisites Done	Tabular
3. Summary of Prerequisites by Student	Drilldown
4. Prerequisites Violations Report, list	Tabular
5. Potential Graduation Report	Tabular
6. Current Students – Capstone Status	Tabular
7. Capstone Committees by Faculty	Drilldown
8. Capstone Committees Chaired and Member Of	Tabular
9. Core Courses Students have not Fulfilled – Drilldown	Drilldown
10. Enrollment by Term	Report and Graph
11. Potential Graduation Report	Tabular
12. Extended Degree Audit	Subreports

The Extended Degree Audit has 4 subreports that listed above for Prerequisites, Core, Elective, Non-standard and Capstone courses. A Prerequisite report was created in the matrix format but not selected as one of the reports to deploy. See Appendix T for list of all reports completed and Appendix S for the reports that were deployed.

The Report Manager provides credentialing and security controls to the reports.

The primary focus of the project is on the SSRS and the high value reporting. To achieve this, the database had to be expanded to accommodate the types of data requirement for the project's charter.

Reports created are basically as proposed. A few report presentations changes from the initial report concepts due to asking questions about the data and how to best present for the high value end result desired.

## **6 Conclusion**

The motivation and importance of this project are put forth herein. The foundation is laid for replacing the outdated paper and Excel file systems. Ownership of the database and the reporting services will be needed to ensure continuity and establishing important aspects of the system:

- Maintenance Schedule
- Risk Management and Security
- Communication Plan and Management
- Issues Management
- Escalation Procedures
- Future Enhancements for Administrators
- Future Enhancements for Students

Working with a data model being developed concurrently with other projects was challenging but it yields collaborative insight into the data warehousing and reporting. See Appendix W for tables entered.

Working through the reporting requirements for each report resulted in going back and revising a prior report to link to a current report. This iterative process provided opportunities to

test the data model and consider views to facilitate data retrieval. See the list of the fifteen views built in Appendix R. These views, containers for data from multiple tables and can store calculated fields, make report creation much simpler.

When putting the components of SSRS together, Visual Studio and SQL Server Business Intelligence, creating, managing and using reports easier with no programming involved and with “high quality presentation” of the reports on the server. (Larson, 2009)

The Query designer is an excellent tool for building the SQL script behind the scenes to create the dataset and the report wizard guides the user through setting the report template up. See the Query designer in Appendix U. After this, adding report features, such as parameters, links, subreports, becomes easy once the user learns to navigate the tools and the toolbar options.

This project provides high value reporting services for the graduate school administrators and students, with an infrastructure to be further developed in the future with additional module and new internal and external program users. The system is implemented with a live reports server and a well-defined production database. It is ready for use by the stakeholders.

## 7 Works Cited

UNCW Computer Science and Information Systems Graduate website. (2014). Retrieved 2014, from <http://www.csb.uncw.edu/mscsis/index.htm>

Larson, B. (2009). *Microsoft SQL Server 2008 Reporting Services*. New York: McGraw-Hill.

Microsoft. (n.d.). *Microsoft Technet Library*. Retrieved from <http://technet.microsoft.com/en-us/library/ms173767%28v=sql.105%29.aspx>

Wikipedia. (2014, April). *Family Education Rights and Privacy Act*. Retrieved April 2014, from Wikipedia: [http://en.wikipedia.org/wiki/Family\\_Educational\\_Rights\\_and\\_Privacy\\_Act](http://en.wikipedia.org/wiki/Family_Educational_Rights_and_Privacy_Act)

## **8 APPENDIX**

## **APPENDIX A – Project Charter**

(Version 4)

### **Purpose**

Provide easy single-source reporting for MS CSIS program and students for Program prerequisites, core courses, elective courses and Capstone/thesis course work.

### **Objectives**

- Improve decision making by program director, coordinator, department administrators, and students
- Improve management of program data (backup, maintenance, auditability)
- Save time for program director and coordinator
- Improve conversion of data to information (access and reporting)
- System Analysis for Data Model, Data Entry, and Reporting Processes

### **Stakeholders**

- Ms. Karen Barnhill, Graduate Program Coordinator
- Dr. Douglas Kline, MS CSIS Program Director
- Department Administrators
- Academic Advisors
- Students

### **Desired Features**

- Single Relational Database on Enterprise Database Server
- Select pre-written program-level reports
- Ad-hoc reporting capabilities
- Data Entry Forms for Program Coordinator
- Ability to adapt to other graduate programs

### **Critical Success Factors**

- Well-designed relational data model
- Direct data entry into production database
- Program Coordinator buy-in
- Functional reporting capabilities
- Adaptability and scalability

### **Constraints**

- Time

- Integration/Separation from Existing Systems
- Personnel Experience – novice to intermediate developer
- Project completion by April 15, 2014

### **Milestones and target dates**

<u>Milestone</u>	<u>Target Date</u>
Proposal Document to Committee	02/02/2014
Project Proposal	02/28/2013
Project Development	03/05/2013
Project Execution	03/31/2014
Project Testing	04/15/2014
Project Defense	04/25/2014

Project incubation period has been 1 1/2 years prior to Proposal. Time frame from Proposal documentation to Defense will be 12 weeks.

### **Risks & Mitigation Strategies**

- Personal Learning curve
  - Technologies and applications: Visual Studio, SQL Server 2008 R2 and Reporting Services, Visual Basic, TSQL,
  - Reading
  - Textbook, Microsoft SQL Server 2008 Reporting Services by Brian Larson
  - Committee, external expert support
- Client feedback on requirements
  - regular meetings
  - confirming accuracy and integrity of data with existing systems
  - working with stakeholders
- Staff Members
  - Focus on high-value-add features
- Ongoing Maintenance of Project
  - Good documentation
  - Standard tools and techniques

### **Roles and Responsibilities**

- Linda Dance – Developer
- Douglas Kline – Project committee chairman
- Ron Vetter – Project committee member
- Bryan Reinicke – Project Committee member
- Karen Barnhill – Client and Committee Member

### **Location of Interest**

- University of North Carolina at Wilmington (UNCW)
- Computer Information Systems Building

- UNCW Graduate Program Office

### **Actors**

- Database Administrator (DBA)
  - Monitors Database and Reporting Server
  - Write new reports
  - Manage report permissions
  - Updates system as needed
- Student
  - Request status report
- UNCW Graduate Program Office Coordinator
  - View Student's Status of Prerequisites reports
  - Request ad-hoc reports to be created by DBA
  - View program- and detail-level reports
  - Data Entry
- UNCW Graduate Program Office Staff
  - Write ad-hoc reports
  - View program- and detail-level reports
  - Data Entry
- UNCW MSCSIS Graduate Director
  - Write ad-hoc reports
  - View program- and detail-level reports
  - Data Entry

### **Deliverables**

- 6) Design system for tracking and reporting of program requirements
- 7) Develop a well-defined data model
- 8) Migrate current data into new system
- 9) Create standardized reporting system
- 10) Specific reports to be created:
  - a) Total number of students enrolled by term
  - b) Credit hours produced by term/course/department
  - c) Yield by term
  - d) Acceptances by gender
  - e) Admissions by term
  - f) Graduates by term
  - g) Time to graduate
  - h) List of students reaching 18 course hour threshold of Master's level courses in the program curriculum but have not completed their prerequisites

## **APPENDIX B – Program Admissions and Degree Requirements**

Computer Science and Information Systems, M.S.

---

←Return to: [-Interdisciplinary Programs](#)

Graduate Coordinator: Dr. Douglas Kline

The Department of Computer Science in the College of Arts and Sciences and the Department of Information Systems and Operations Management in the Cameron School of Business offer a joint program of study leading to the Master of Science degree in computer science and information systems. This interdisciplinary program is targeted primarily at students who received undergraduate degrees in computer science or information systems and at computer professionals with equivalent academic preparation. A secondary audience for the program is students whose background is in related areas such as business, mathematics, and electrical engineering, or working professionals seeking to migrate to the information technology arena. The interdisciplinary nature of this program provides a unique balance of advanced scientific knowledge, commonly found in the computer science field, and the development of systems and solutions, applied in a business environment, usually considered the focus of the information systems field. This unique blend will provide a foundation for information technology professionals to have a broader perspective of the rapidly expanding and evolving science of technology and how it can be managed and leveraged to support and further commerce and trade activities.

### **Admission Requirements**

---

Applicants seeking admission to the graduate program in computer science and information systems are required to submit the following to the Graduate School:

An application for graduate admission.

Official transcripts of all college work (undergraduate and graduate).

Certificates of training in computer science/information systems if applicable.

Official scores on the Graduate Management Admission Test (GMAT) or Graduate Record Examination (GRE). Scores more than five years old will not be accepted.

Three recommendations from individuals in professionally relevant fields.

Applicant's resume and a letter of interest.

Additional materials may be required (e.g., TOEFL or IELTS scores for international students).

Each applicant must have a strong overall academic record and have successfully completed the undergraduate level prerequisites in computer science and information systems courses or their equivalent: two programming courses, and a course in each of data structures, database, software engineering, analysis and design or data communications or networking, financial accounting, marketing, finance, and management. Deficiencies in a student's undergraduate preparation will be ascertained by the MSCSIS Advisory Committee. Placement tests may be administered to incoming students at the discretion of the advisory committee to assist with the evaluation of deficiencies.

Persons entering the program must have completed a basic core of computer science and information system courses. Professional experience may be accepted for some of the prerequisite coursework. Professional experience and/or coursework from other institutions must be approved by the MSCSIS Advisory Committee. Professional experience and/or technical certifications will be evaluated on a case by case basis for any prerequisite substitution.

Admissions decisions are based upon the examination of several factors, and where other indicators of success warrant, individuals who fall below the established criteria may still be considered for admission.

Deadline for receipt of applications is April 1 for Fall admission and November 1 for Spring admission.

## **Degree Requirements**

---

Programs leading to the Master of Science degree require a minimum of 36 semester hours of graduate study. This includes six required core courses (18 hours) providing a mix of theoretical underpinning, technical skills, and information technology perspectives and elective courses (12 to 15 hours) that provide the opportunity for additional study in a variety of areas to be determined by the student and his/her advisory committee. A research project (3 hours), or a thesis (6 hours), will serve as the capstone experience.

No more than 9 credit hours from those courses cross listed as 400/500 may be applied toward the degree. Graduate courses offered by other departments may be approved by the student's advisory committee. A maximum of six hours of credit may be transferred from another accredited institution. Grades earned on transfer work must be equivalent to a "B" or better, and courses must be acceptable to the student's advisory committee. A minimum of 30 semester hours of graduate study must be completed in residence.

A student must have no less than a 3.0 GPA on all graduate-level courses.

The student must successfully complete an oral defense.

The program shall be completed within five years of the date of first registration for graduate study.

A research project (3 hours), or a thesis (6 hours), will serve as a capstone experience.

No more than 9 credit hours from the list [CSC 591](#), [MIS 591](#), [CSC 595](#), [MIS 595](#), [CSC 598](#), and [MIS 598](#) may be applied toward the degree.

## **Degree Options**

---

A student may select among two options:

<b>Option</b>	<b>Program Coursework</b>	<b>Research Project</b>	<b>Thesis</b>
1	33 hours	3 hours	
2	30 hours		6 hours

### **Option 1 – Research Project**

---

This option requires at least 36 semester hours of graduate credit, with three credit hours for the project ([CSC 594](#) or [MIS 594](#)). Under this option, the student is required to complete a three hour research project under the direction of a graduate advisory committee. This project could involve the development of software, work on a project (potentially part of a team), independent research, or some other scholarly pursuit. The outcome includes a technical paper written by the student and an oral defense acceptable to the student’s advisory committee. In the oral defense, the student is responsible for the domain of the research project as well as the program coursework.

### **Option 2 – Thesis**

---

This option requires at least 36 semester hours of graduate credit, with six credit hours for the thesis ([CSC 599](#) or [MIS 599](#)). Each student must present and defend a thesis, based on original research, acceptable to the student’s advisory committee, prior to graduation. In the oral defense,

the student is responsible for the domain of the research effort as well as the program coursework. The thesis defense is open to the public.

### **Required Core Courses**

---

[CSC 532 - Design and Analysis of Algorithms I](#) Credits: (3)

[MIS 534 - Information Security Management](#) Credits: (3)

[CSC 544 - Network Programming](#) Credits: (3)

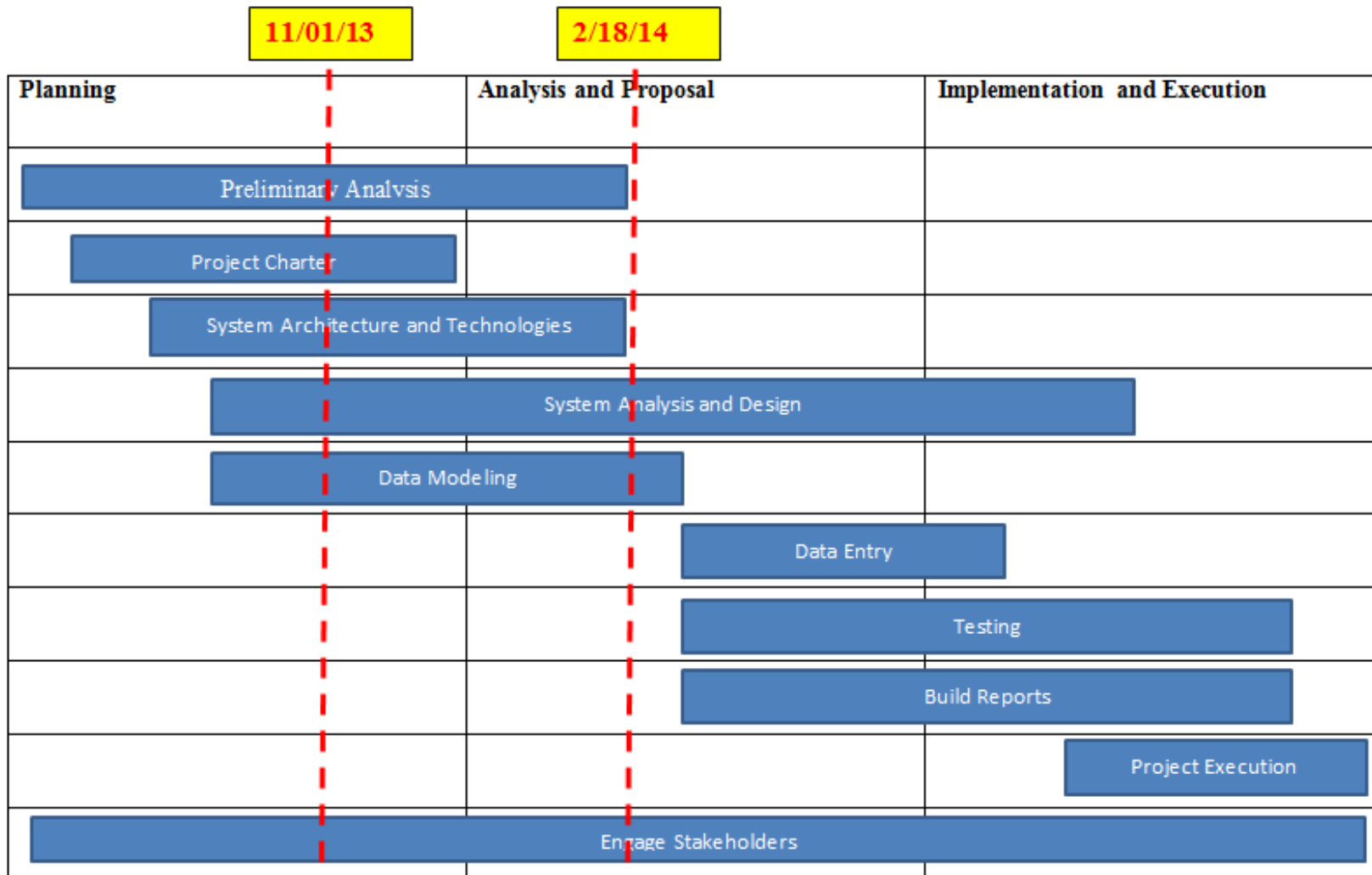
[CSC 550 - Software Engineering](#) Credits: (3)

[MIS 555 - Database Management Systems](#) Credits: (3)

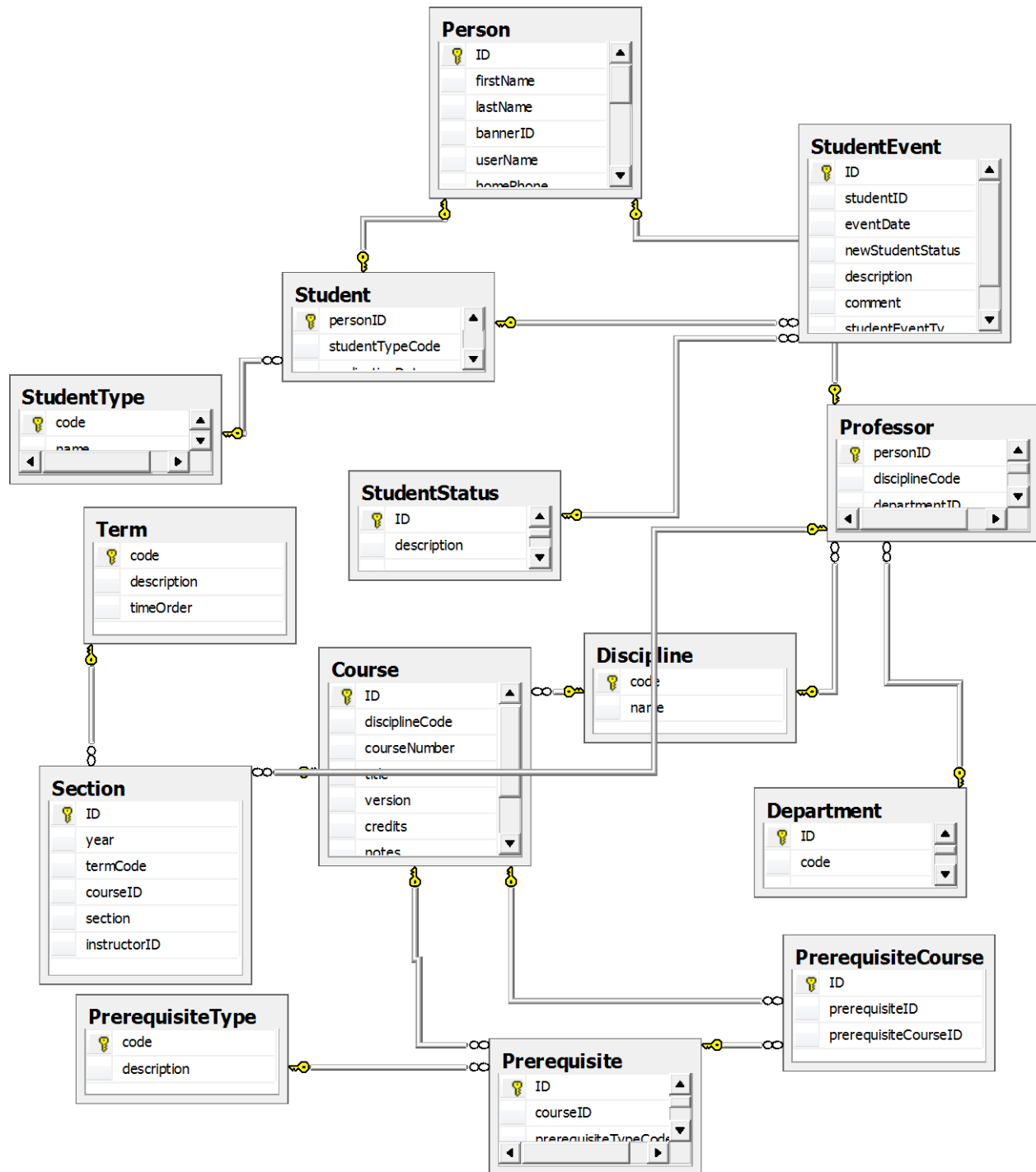
[MIS 565 - Analysis, Modeling and Design](#) Credits: (3)

(UNCW Computer Science and Information Systems Graduate website, 2014)

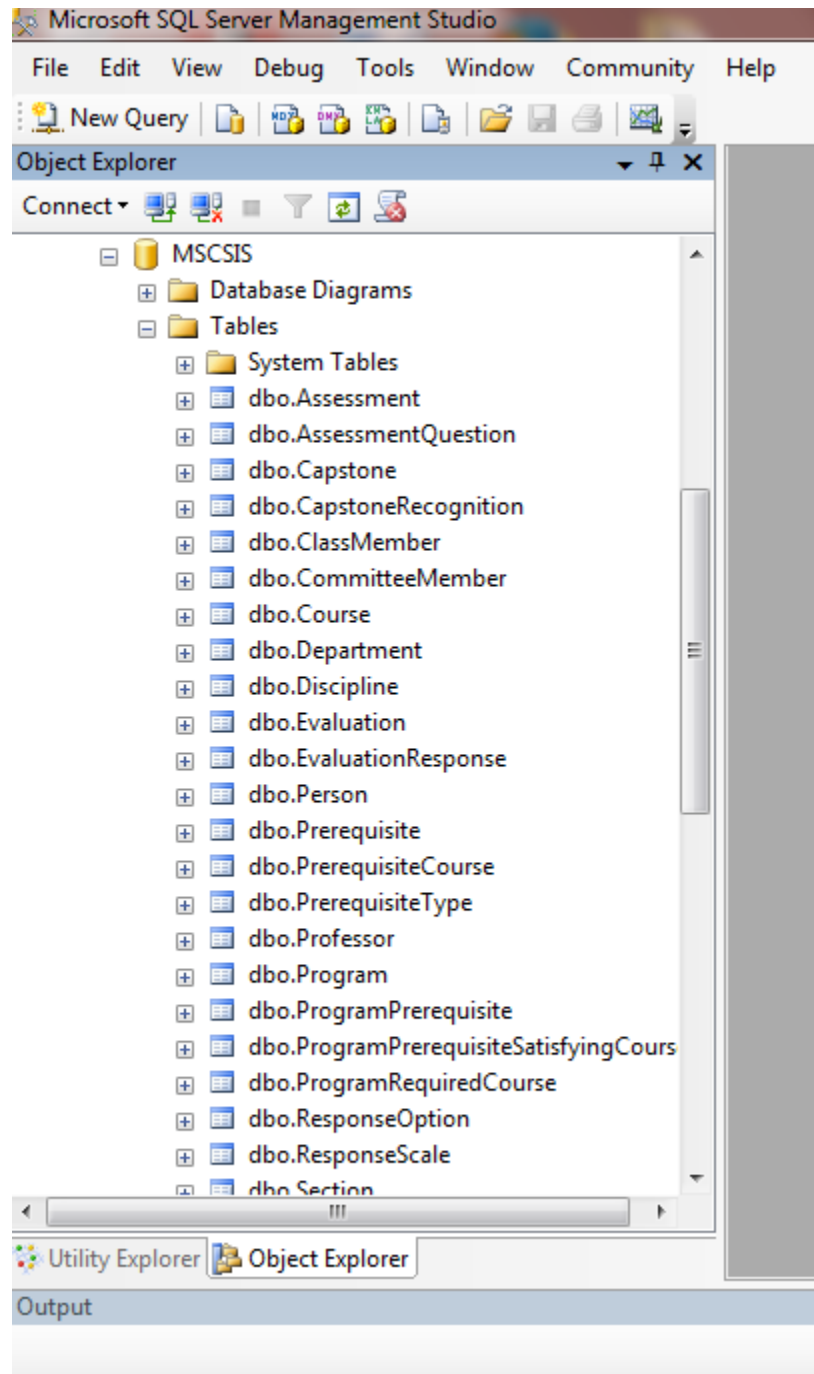
## APPENDIX C: Project Staging



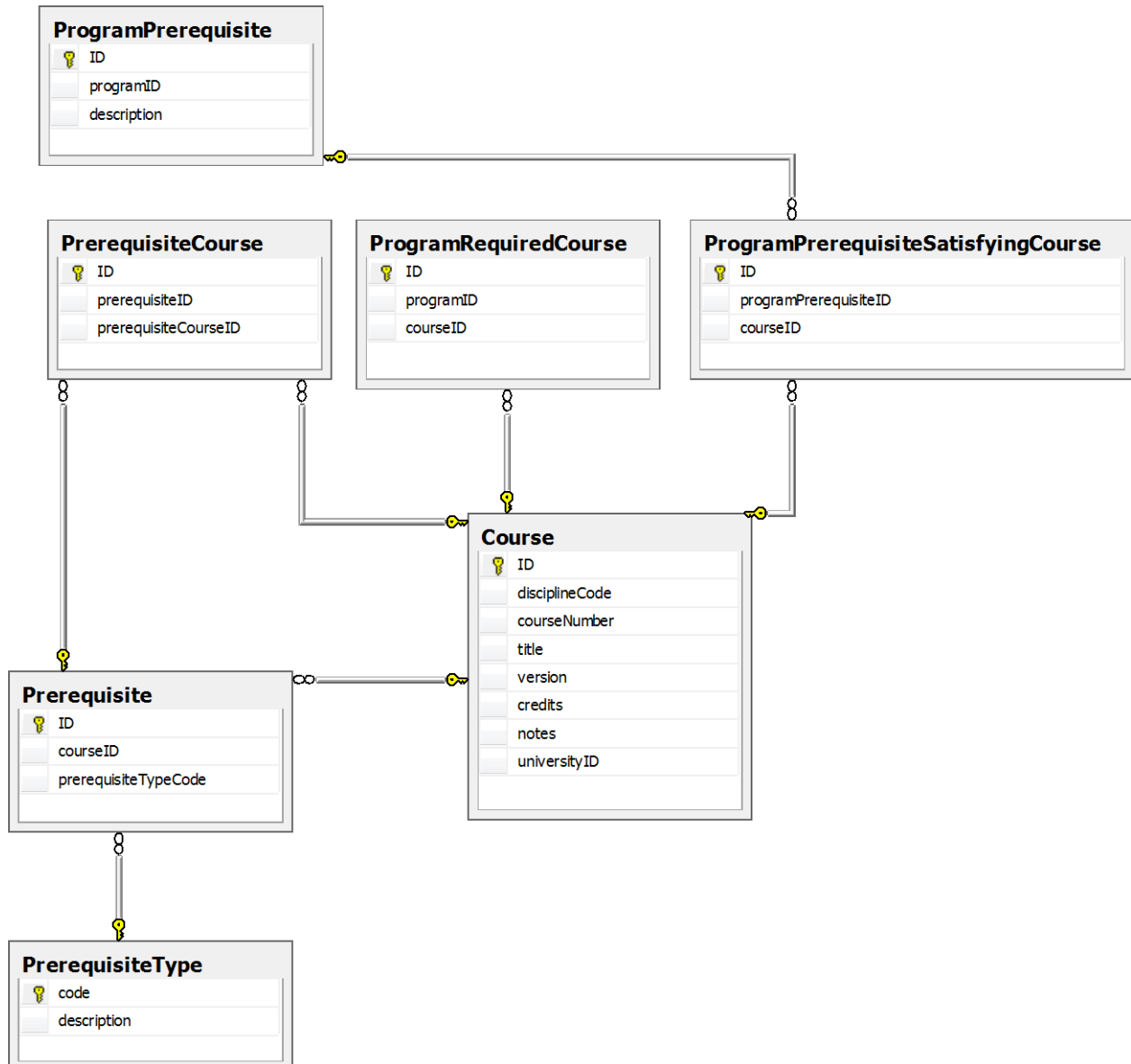
## APPENDIX D: Database Diagram – Basic Model (version prior to this project)



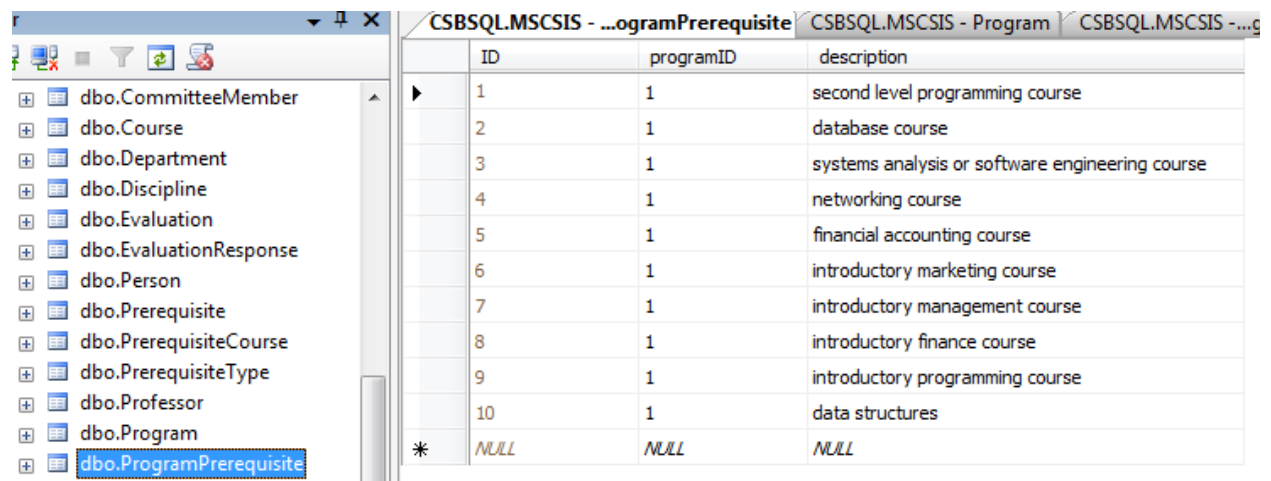
## APPENDIX E: Database Tables (prior to this project)



## APPENDIX F: Database Diagram of Program Requirements prior to this project



## APPENDIX G: Program Prerequisites already loaded from another project



The screenshot shows the SQL Server Enterprise Manager interface. On the left, a tree view displays the database schema, with 'dbo.ProgramPrerequisite' selected. The main pane shows a table with the following data:

ID	programID	description
1	1	second level programming course
2	1	database course
3	1	systems analysis or software engineering course
4	1	networking course
5	1	financial accounting course
6	1	introductory marketing course
7	1	introductory management course
8	1	introductory finance course
9	1	introductory programming course
10	1	data structures
*	NULL	NULL

## APPENDIX H: Program Prerequisites Satisfying Course is already loaded from another project

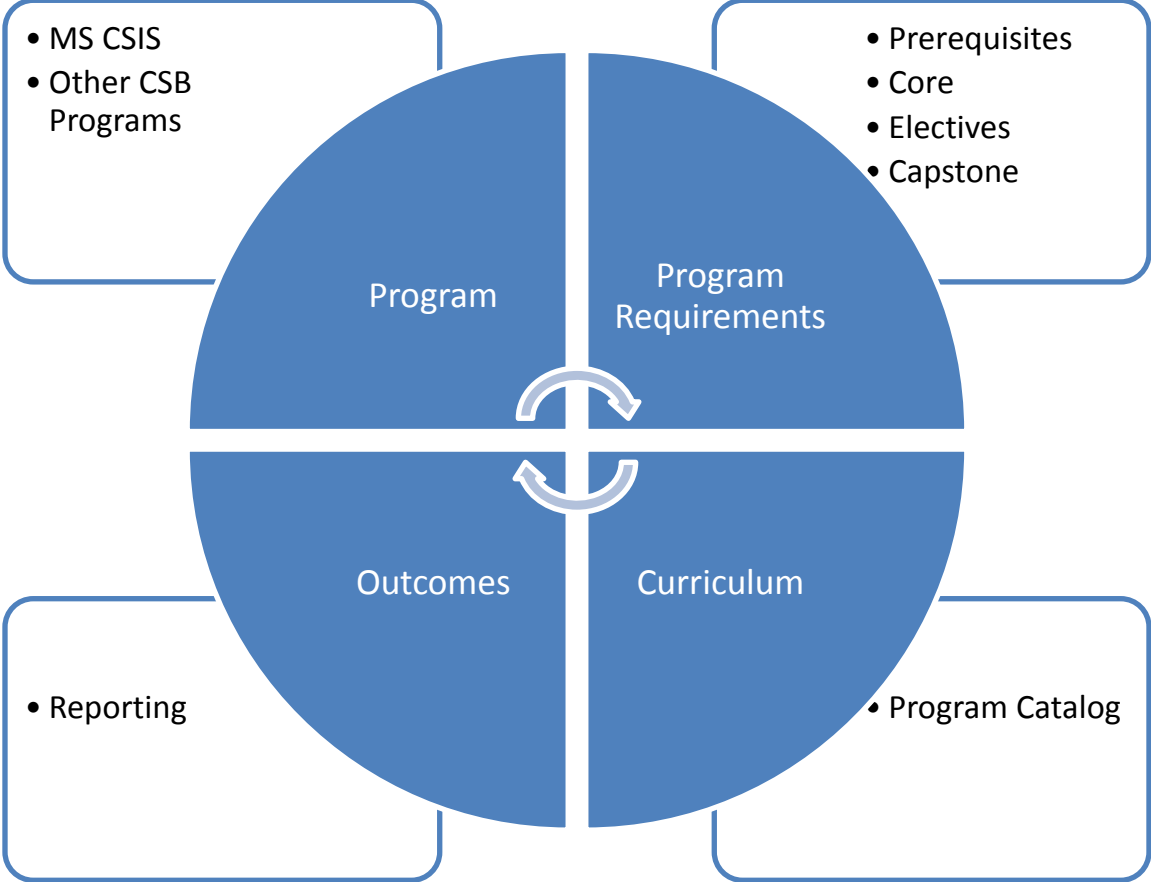
The screenshot displays the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows the database structure, with the 'dbo.ProgramPrerequisiteSatisfyingCourse' table selected. The table's columns are listed as follows:

- ID (PK, int, not null)
- programPrerequisiteID (FK, int, not null)
- courseID (FK, int, not null)

The main window shows a query result grid for the 'CSBSQL.MSCSIS - ...eSatisfyingCourse' table. The grid contains 15 rows of data and a final row with NULL values. The columns are ID, programPrerequisiteID, and courseID.

ID	programPrerequisiteID	courseID
1	1	3
2	1	5
3	2	6
4	2	7
5	3	8
6	3	9
7	4	10
8	4	11
12	5	12
13	6	13
14	7	14
15	8	15
9	9	2
*	NULL	NULL

**APPENDIX I: Data Model Overview for Design and Development**



**APPENDIX J: Logs of Meetings** (this list is not all inclusive)

Dates	Met with	Discussion
Nov 2011	Dr. Kline	Explored a Capstone Project for a Capstone Tracking and Reporting system for the UNCW MSCSIS Program.
1/9/12	Dr. Kline	
2/3/12	Dr. Kline	Discussed Program needs for tracking and reporting services. Review current system. Considered the events and milestones that the data model will support. Basic project output: <ul style="list-style-type: none"> <li>• Data Model</li> <li>• Forms</li> <li>• Reporting</li> </ul>
2/10/12	Dr. Kline	Discussed scope of project. The MS CSIS program could use some of this reporting for the program review that was being conducted at the time. Considered the overview of the future system, built in modules, will for provide reporting capabilities.
3/23/12	Dr. Kline	Discussed program review and updating Excel worksheets. Reporting system is needed.
4/9/12	Ms. Barnhill	Met to discuss the tracking of Capstone projects including the forms, signatures for the Capstone events forms for establishing a committee, scheduling Proposals and Defenses and the challenges of the process and the tracking system.
Spring 2012	Ms. Barnhill	Met several times to discuss Capstone forms and Capstone tracking.
Spring 2012	Dr. Kline	Met weekly to discuss program review and Capstone tracking.
10/19/12	Dr. Kline	Project Charter
10/26/12	Dr. Kline	Actor Diagram
9/17/13	Dr. Kline	Building out machine with Business Intelligence for SQL Server 2008. Put prerequisite data in. Data Model not highly normalized. Need on demand reporting for better decision making.
10/25/13	Dr. Kline	See Tab A. Also, received deployment settings for the Report Server.
2/18/14	Dr. Kline	Mock up reports for Proposal.
2/27/14	Dr. Kline	Review Proposal presentation and Powerpoint slides.
2/28/14	Committee	Proposal
3/27/14	Dr. Kline	Building views.
4/3/14	Dr. Kline	Add subtype to course table for non-standard courses, account for incompletes, review drill through parameters.
4/9/14	Dr. Kline	Capstone by Faculty reports
4/14/14	Dr. Kline	Project documentation
4/21/14	Dr. Kline	Project documentation and Defense

## APPENDIX K: Emails to get the account set up on the UNCW production SSRS server

SQL Reporting Services (SSRS) account?

[Kline, Douglas](#)

Sent: Thursday, August 01, 2013 12:47 PM

To: [TAC](#)

Cc: [Dance, Linda Oxendine](#); [Barnhill, Karen](#)

Hello,

Ms. Linda Dance ([lod2457@uncw.edu](mailto:lod2457@uncw.edu)) is working with me on a project as part of her degree requirements for the MSCSIS program.

As part of her project, Ms. Dance will be creating some reports from data in the MSCSIS database on the CSBSQL database server.

These reports will be used by myself and the Cameron graduate programs office in administering the MSCSIS program.

Rather than build and maintain our own SSRS server, we would like to have the reports hosted on a production SSRS server.

Here are the basic requirements:

Access to create, edit, delete reports for these people, through Windows domain authentication, i.e., using UNCW logins

- Ms. Linda Dance
- Myself, Douglas Kline, [klined@uncw.edu](mailto:klined@uncw.edu)
- Ms. Karen Barnhill, Graduate Programs Coordinator, [barnhillk@uncw.edu](mailto:barnhillk@uncw.edu)

Ability to connect to CSBSQL as the data source

A “common” area for the three people above – we should all be able to work on the same set of reports

Ability to author reports through the SSRS web interface

Ability to author reports through the Visual Studio Business Intelligence Design Studio

This is probably a somewhat unusual request – I’m happy to meet to discuss the project.

Thanks,

Douglas M. Kline, Ph.D.

Director, MS Computer Science and Information Systems

<http://www.csb.uncw.edu/mscsis/>

Associate Professor, Information Systems

Cameron School of Business

UNC Wilmington

<http://www.csb.uncw.edu/>

Office Hours for Spring 2013

MW 9:30-noon, 1-3:30

910-962-7552

<http://www.csb.uncw.edu/people/klined>

From: Speakman, Lori  
Sent: Friday, October 25, 2013 11:41 AM  
To: Speakman, Lori; Kline, Douglas  
Cc: Dance, Linda Oxendine  
Subject: RE: INC 32971

In addition, in your VS project; this should be the deploy settings:

OverWriteDataSources False  
TargetDataSourceFolder Data Sources  
TargetReportFolder CSB/MSCSIS  
TargetServerURL <https://reports.uncw.edu/reportserver>

Thanks and have a great day!

Lori Speakman  
System Administrator

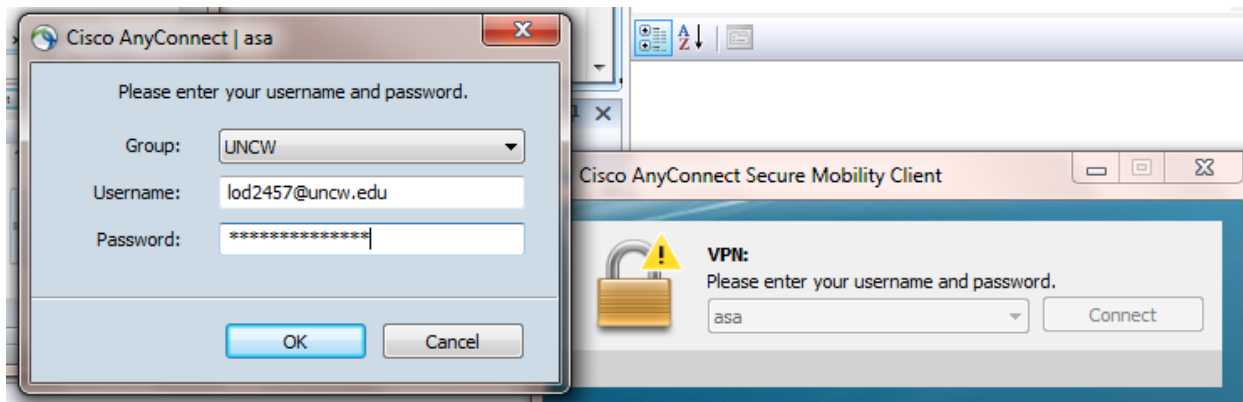
ITS/Department of Operations & Systems Administration  
UNC-Wilmington  
910-962-4184

[speakmanl@uncw.edu](mailto:speakmanl@uncw.edu)<<mailto:speakmanl@uncw.edu>>

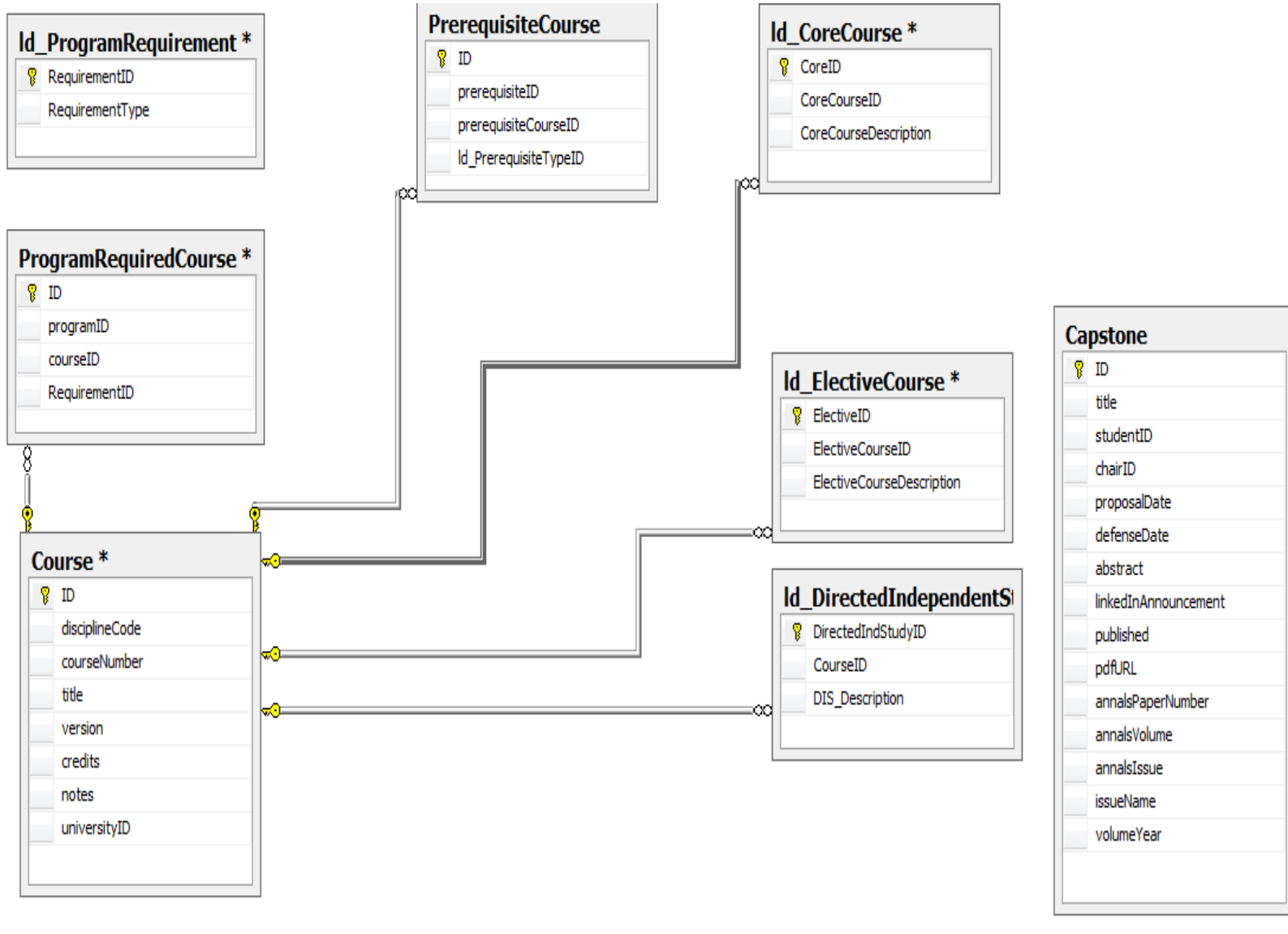
AskTAC for self-service solutions and immediate assistance!

<https://asktac.uncw.edu>

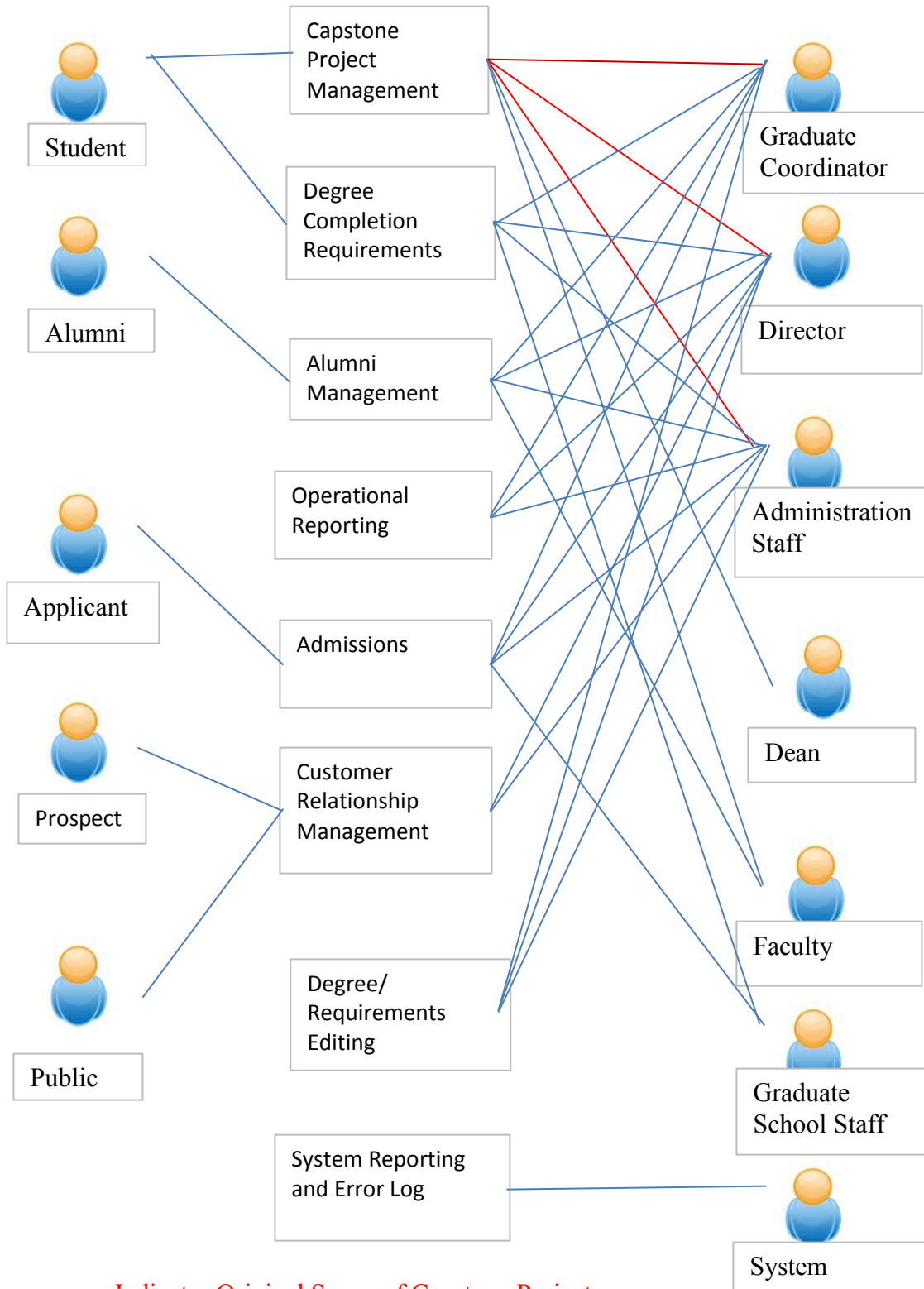
**APPENDIX L: Screen shot of Cisco VPN login:**



## APPENDIX N: Database Diagram: Program Requirements



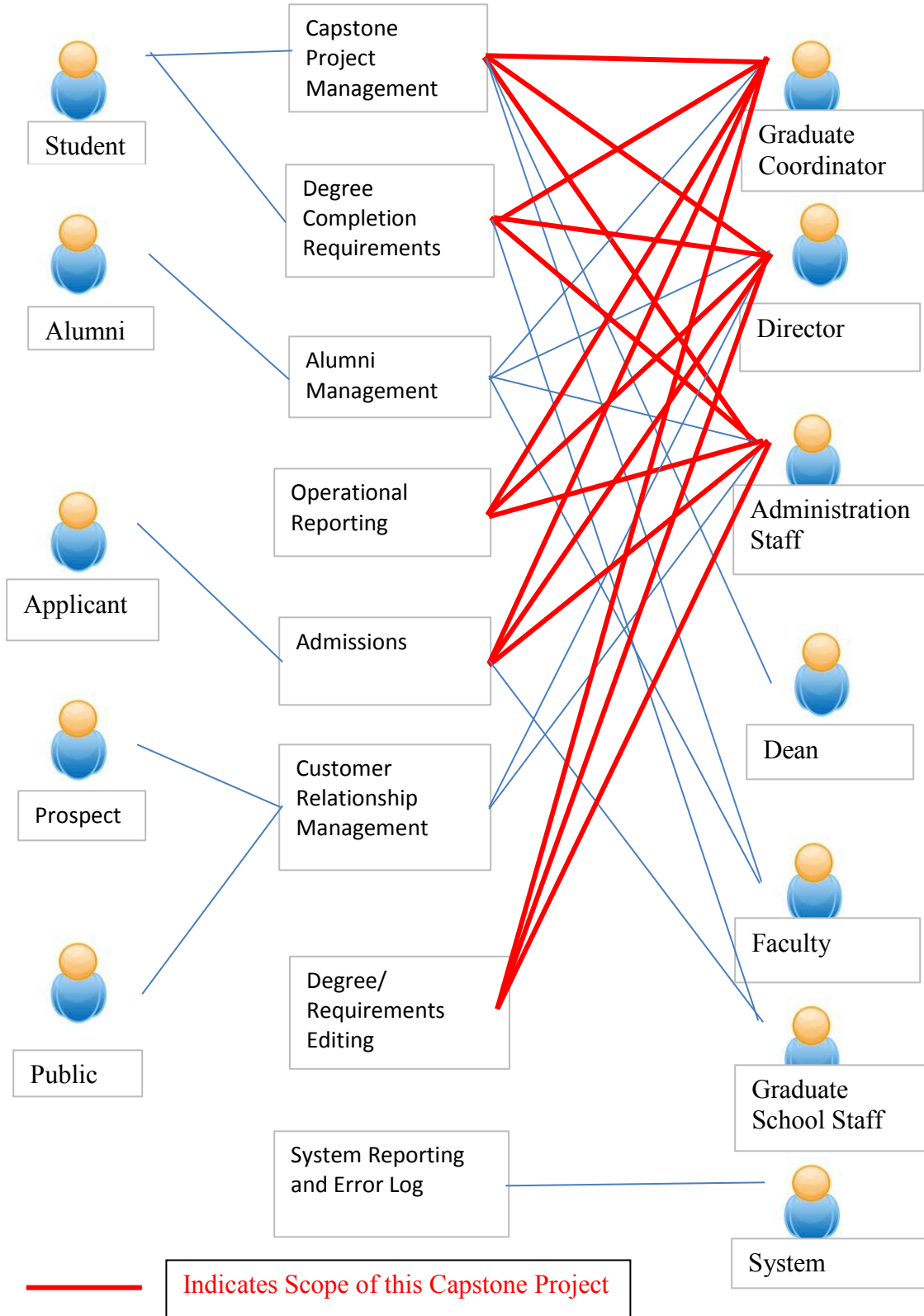
APPENDIX O: Original scope of project was Capstone Project Management System  
**Actor Diagram**



————— Indicates Original Scope of Capstone Project

# APPENDIX P: Scope of project for Tracking and Reporting System

## Actor Diagram



**APPENDIX Q: Sample Reports**

**UNCW MS CSIS Program**

**Applicants who have Prerequisites to Fulfill**

**1/31/14**

<u>Applicant</u>	<u>Prerequisite to Fulfill</u>	<u>Course Hours</u>	<u>ALERT: Approaching Program Course 18 Hour Threshold</u>
Jane Smith			
	332 Data Structures	3	
	455 Database Management	3	
	350 Financial Accounting	3	
	365 Marketing Concepts	<u>3</u>	
Total Hours Prerequisites to Fulfill		12	
Total # Prerequisites to Fulfill		4	
<b>Program Course Hours Completed to Date</b>			<b>15 *****</b>

Thomas Delaney

	455 Database Management	3	
	350 Financial Accounting	3	
	365 Marketing Concepts	<u>3</u>	
Total Hours Prerequisites to Fulfill		9	
Total # Prerequisites to Fulfill		3	
Program Course Hours Completed to Date			12

**Total # of Applicants as of the date of this report = 2**

**Total Hours Prerequisites to Fulfill = 21**

**Total # Prerequisites to fulfill = 7**

**Total Program Course Hours Completed to date by these Applicants = 37**

*(Applicant for this report is defined as Applicant who has applied but has not yet been accepted into the MS CSIS Program.)*

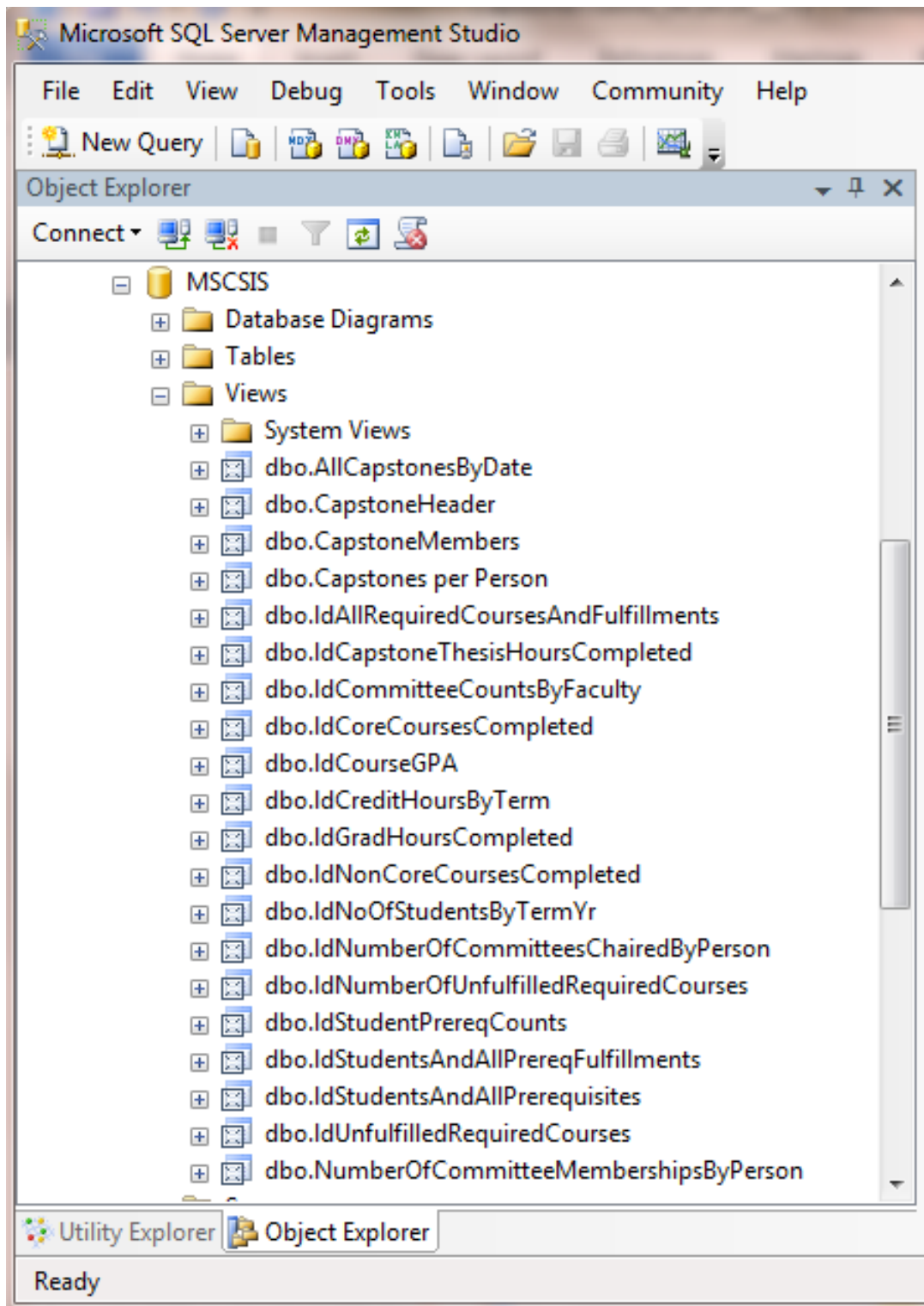
**UNCW MS CSIS Program**  
**Prerequisites to be fulfill Applicants**  
**1/31/14**

<u>Prerequisites</u>	<u>Number of Applicants</u>
332 Data Structures	1
455 Database Management	2
350 Financial Accounting	2
365 Marketing Concepts	2

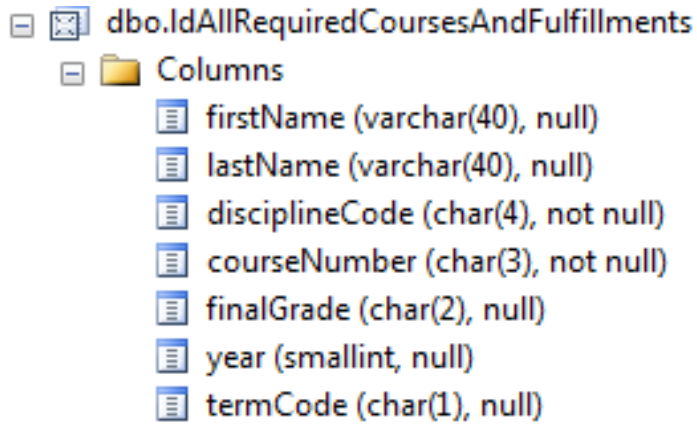
Total # of Prerequisites to be fulfilled by Applicants = 7

## APPENDIX R: List of Views Created, those with naming convention Id\*\*\* and some of the attributes

Overview:



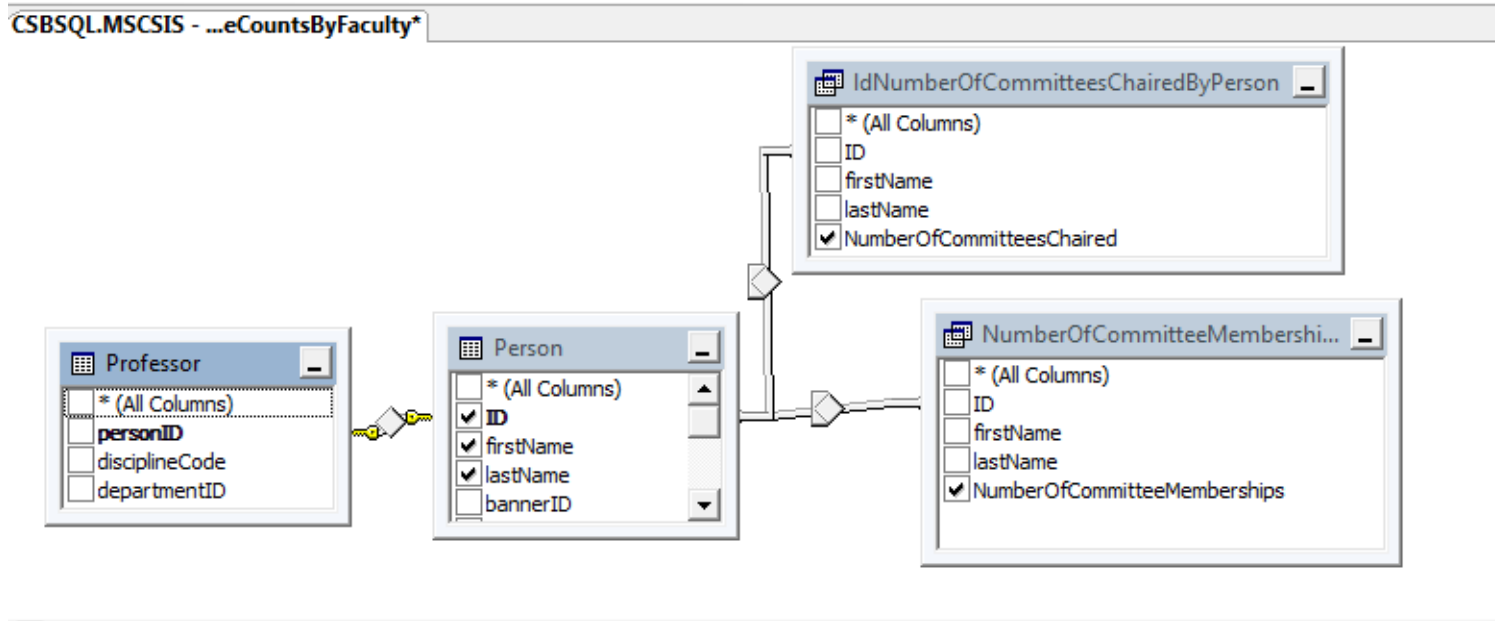
1) IdAllRequiredCoursesAndFulfillments



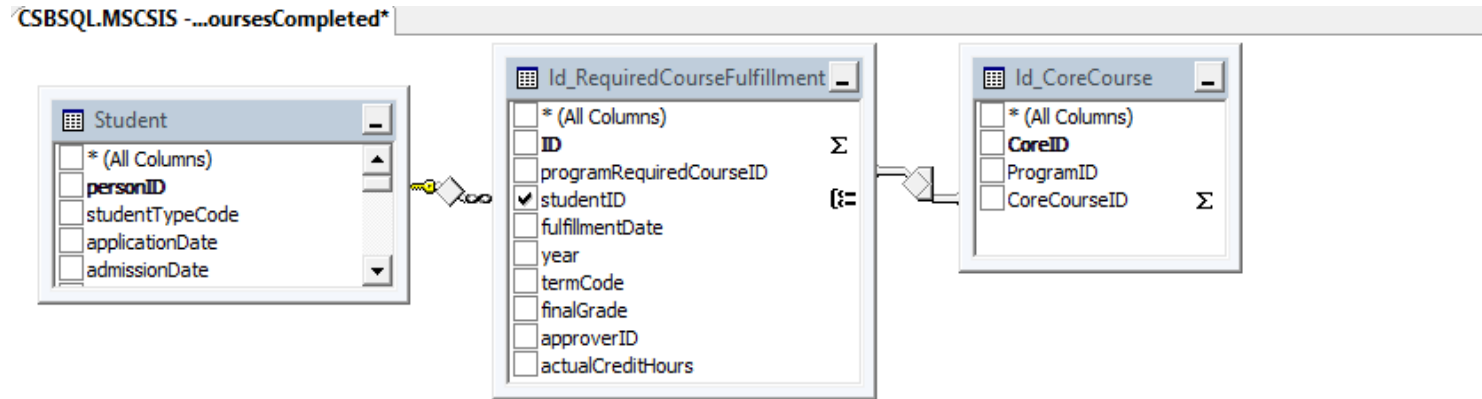
2) IdCapstoneThesisHoursCompleted

Column	Alias	Table	Output	Sort Type	Sort Order	Group By
studentID		Id_RequiredCourseFulfillment	<input checked="" type="checkbox"/>			Group By
ID		Course	<input checked="" type="checkbox"/>			Group By
actualCreditHours	[Capstone/Thesis Hours Completed]	Id_RequiredCourseFulfillment	<input checked="" type="checkbox"/>			Sum
			<input type="checkbox"/>			

3) IdCommitteeCountsByFaculty: This view uses the view, IdNumberOfCommitteesChairedByPerson.



#### 4) IdCoreCoursesCompleted



Column	Alias	Table	Output	Sort Type	Sort Order	Group By	Filter
studentID		Id_RequiredCourseFulfillment	<input checked="" type="checkbox"/>			Group By	
CoreCourseID	[Core Courses Required]	Id_CoreCourse	<input checked="" type="checkbox"/>			Count	
ID	[Core Course Completed]	Id_RequiredCourseFulfillment	<input checked="" type="checkbox"/>			Count	
			<input type="checkbox"/>				

```

LECT  dbo.Id_RequiredCourseFulfillment.studentID, COUNT(dbo.Id_CoreCourse.CoreCourseID) AS [Core Courses Required], COUNT(dbo.Id_RequiredCourseFulfillment.ID)
      AS [Core Course Completed]
.OM   dbo.Id_RequiredCourseFulfillment INNER JOIN
      dbo.Student ON dbo.Id_RequiredCourseFulfillment.studentID = dbo.Student.personID RIGHT OUTER JOIN
      dbo.Id_CoreCourse ON dbo.Id_RequiredCourseFulfillment.programRequiredCourseID = dbo.Id_CoreCourse.CoreCourseID
:OUP BY  dbo.Id_RequiredCourseFulfillment.studentID
  
```

studentID	Core Courses Requi...	Core Course Compl...
126	6	6
146	3	3
150	1	1



6) IdCourseHoursByTerm is designed to show aggregate hours by term without the details by student or course.

CSBSQL.MSCSIS - ...editHoursByTerm\*

The screenshot displays a query window with three tables: Term, Id\_RequiredCourseFulfillment, and Course. The Term table has columns code, description, and timeOrder. The Id\_RequiredCourseFulfillment table has columns ID, programRequiredCourseID, studentID, fulfillmentDate, year, termCode, finalGrade, approverID, and actualCreditHours. The Course table has columns ID, disciplineCo, courseNumt, title, version, credits, notes, universityIC, includeInGP, and nonStandar. The query window shows a SELECT statement with columns termCode, year, and TotalCreditHours (SUM of credits). The query is grouped by termCode and year. The query is:

```
SELECT  dbo.Id_RequiredCourseFulfillment.termCode, dbo.Id_RequiredCourseFulfillment.year, SUM(dbo.Course.credits) AS TotalCreditHours
FROM    dbo.Course RIGHT OUTER JOIN
        dbo.Id_RequiredCourseFulfillment ON dbo.Course.ID = dbo.Id_RequiredCourseFulfillment.programRequiredCourseID FULL OUTER JOIN
        dbo.Term ON dbo.Id_RequiredCourseFulfillment.termCode = dbo.Term.code
GROUP BY  dbo.Id_RequiredCourseFulfillment.termCode, dbo.Id_RequiredCourseFulfillment.year
```

Column	Alias	Table	Output	Sort Type	Sort Order	Group By	Filter
termCode		Id_Require...	<input checked="" type="checkbox"/>			Group By	
year		Id_Require...	<input checked="" type="checkbox"/>			Group By	
credits	TotalCreditHours	Course	<input checked="" type="checkbox"/>			Sum	

7) IdGradHoursCompleted is to be used for several reports, the Potential Graduation Report and the Extended Degree Audit.

CSBSQL.MSCSIS -...dHoursCompleted\*

Column	Alias	Table	Output	Sort Type	Sort Order	Group By	Filter
personID		Student	<input checked="" type="checkbox"/>	Ascending	1	Group By	
firstName		Person	<input checked="" type="checkbox"/>			Group By	
lastName		Person	<input checked="" type="checkbox"/>			Group By	
actualCreditHours	[Graduate Hours Completed]	Id_RequiredCourseFulfillment	<input checked="" type="checkbox"/>			Sum	

```

SELECT TOP (100) PERCENT dbo.Student.personID, dbo.Person.firstName, dbo.Person.lastName, SUM(dbo.Id_RequiredCourseFulfillment.actualCreditHours)
AS [Graduate Hours Completed]
FROM      dbo.Student LEFT OUTER JOIN
          dbo.Person ON dbo.Student.personID = dbo.Person.ID RIGHT OUTER JOIN
          dbo.Id_RequiredCourseFulfillment ON dbo.Student.personID = dbo.Id_RequiredCourseFulfillment.studentID
GROUP BY dbo.Student.personID, dbo.Person.firstName, dbo.Person.lastName
ORDER BY dbo.Student.personID
  
```

personID	firstName	lastName	Graduate Hour...
126	Linda	Dance	36

- 8) IdNonCoreCoursesCompleted is designed to output Elective courses completed. This view is used in the Potential Graduation Report and the Extended Degree Audit.

CSBSQL.MSCSIS - ...oursesCompleted\*

Column	Alias	Table	Output	Sort Type	Sort Order	Filter	Or...	Or...
ID		Person	<input checked="" type="checkbox"/>					
firstName		Person	<input checked="" type="checkbox"/>					
lastName		Person	<input checked="" type="checkbox"/>					
year		Id_RequiredCourseFulfillment	<input checked="" type="checkbox"/>					
termCode		Id_RequiredCourseFulfillment	<input checked="" type="checkbox"/>					
finalGrade		Id_RequiredCourseFulfillment	<input checked="" type="checkbox"/>					
disciplineCode		Course	<input checked="" type="checkbox"/>					
courseNumber		Course	<input checked="" type="checkbox"/>			NOT IN ('594', '599')		
CoreID		Id_CoreCourse	<input checked="" type="checkbox"/>			IS NULL		
nonStandardC...		Course	<input checked="" type="checkbox"/>					
actualCredith...		Id_RequiredCourseFulfillment	<input checked="" type="checkbox"/>					

```

SELECT  dbo.Person.ID, dbo.Person.firstName, dbo.Person.lastName, dbo.Id_RequiredCourseFulfillment.year, dbo.Id_RequiredCourseFulfillment.termCode,
        dbo.Id_RequiredCourseFulfillment.finalGrade, dbo.Course.disciplineCode, dbo.Course.courseNumber, dbo.Id_CoreCourse.CoreID, dbo.Course.nonStandardCourse,
        dbo.Id_RequiredCourseFulfillment.actualCreditHours
FROM    dbo.Id_RequiredCourseFulfillment INNER JOIN
        dbo.Person ON dbo.Id_RequiredCourseFulfillment.studentID = dbo.Person.ID INNER JOIN
        dbo.Course ON dbo.Id_RequiredCourseFulfillment.programRequiredCourseID = dbo.Course.ID LEFT OUTER JOIN
        dbo.Id_CoreCourse ON dbo.Course.ID = dbo.Id_CoreCourse.CoreCourseID
WHERE   (dbo.Id_CoreCourse.CoreID IS NULL) AND (dbo.Course.courseNumber NOT IN ('594', '599'))
  
```

9) IdNoOfStudentsByTermYr

Column	Alias	Table	Output	Sort Type	Sort Order	Group By	Filter
termCode		Id_RequiredCourseFulfillment	<input checked="" type="checkbox"/>			Group By	
year		Id_RequiredCourseFulfillment	<input checked="" type="checkbox"/>			Group By	
*	[# Students]		<input checked="" type="checkbox"/>			Count	

```

SELECT  dbo.Id_RequiredCourseFulfillment.termCode, dbo.Id_RequiredCourseFulfillment.year, COUNT(*) AS [# Students]
FROM    dbo.Course RIGHT OUTER JOIN
        dbo.Term INNER JOIN
        dbo.Id_RequiredCourseFulfillment ON dbo.Term.code = dbo.Id_RequiredCourseFulfillment.termCode LEFT OUTER JOIN
        dbo.Student ON dbo.Id_RequiredCourseFulfillment.studentID = dbo.Student.personID AND dbo.Term.code = dbo.Student.admissionTerm AND
        dbo.Term.code = dbo.Student.graduationTerm ON dbo.Course.ID = dbo.Id_RequiredCourseFulfillment.programRequiredCourseID
GROUP BY  dbo.Id_RequiredCourseFulfillment.termCode, dbo.Id_RequiredCourseFulfillment.year
    
```

10) IdNumberOfCommitteesChairedByPerson

CSBSQL.MSCSIS - ...ChairedByPerson\*

Column	Alias	Table	Output	Sort Type	Sort Order	Group By	Filter
ID		Person	<input checked="" type="checkbox"/>			Group By	
firstName		Person	<input checked="" type="checkbox"/>			Group By	
lastName		Person	<input checked="" type="checkbox"/>			Group By	
ID	NumberOfCommitteesChaired	Capstone	<input checked="" type="checkbox"/>			Count	

```

SELECT  dbo.Person.ID, dbo.Person.firstName, dbo.Person.lastName, COUNT(dbo.Capstone.ID) AS NumberOfCommitteesChaired
FROM    dbo.Professor INNER JOIN
        dbo.Person ON dbo.Professor.personID = dbo.Person.ID INNER JOIN
        dbo.Capstone ON dbo.Professor.personID = dbo.Capstone.chairID
GROUP BY  dbo.Person.ID, dbo.Person.firstName, dbo.Person.lastName
    
```

11) IdNumberOfUnfulfilledRequiredCourses

CSBSQL.MSCSIS -...RequiredCourses\*

IdUnfulfilledRequiredCourses

- \* (All Columns)
- firstName
- lastName
- disciplineCode
- courseNumber
- finalGrade
- year
- termCode
- personID

Column	Alias	Table	Output	Sort Type	Sort Order	Group By
personID		IdUnfulfilledRequiredCourses	<input checked="" type="checkbox"/>	Ascending	2	Group By
firstName		IdUnfulfilledRequiredCourses	<input checked="" type="checkbox"/>			Group By
lastName		IdUnfulfilledRequiredCourses	<input checked="" type="checkbox"/>	Ascending	1	Group By
personID	NumberOfUnfulfilledRequiredCourses	IdUnfulfilledRequiredCourses	<input checked="" type="checkbox"/>			Count

## 12) IdStudentPrereqCounts

CSBSQL.MSCSIS - ...entPrereqCounts\*

Column	Alias	Table	Output	Sort Type	Sort Order	Group By	Filter	Or...	Or
personID		Student	<input checked="" type="checkbox"/>	Ascending	1	Group By			
firstName		Person	<input checked="" type="checkbox"/>			Group By			
lastName		Person	<input checked="" type="checkbox"/>			Group By			
ID	TotalPrereqsRequired	ProgramPrerequisite	<input checked="" type="checkbox"/>			Count			
ID	PrereqsDone	Id_PrerequisiteFulfillment	<input checked="" type="checkbox"/>			Count			

```

SELECT TOP (100) PERCENT dbo.Student.personID, dbo.Person.firstName, dbo.Person.lastName, COUNT(dbo.ProgramPrerequisite.ID) AS TotalPrereqsRequired,
COUNT(dbo.Id_PrerequisiteFulfillment.ID) AS PrereqsDone
FROM   dbo.Student INNER JOIN
      dbo.Program ON dbo.Student.programID = dbo.Program.ID INNER JOIN
      dbo.ProgramPrerequisite ON dbo.Program.ID = dbo.ProgramPrerequisite.programID INNER JOIN
      dbo.Person ON dbo.Student.personID = dbo.Person.ID LEFT OUTER JOIN
      dbo.Id_PrerequisiteFulfillment ON dbo.Student.personID = dbo.Id_PrerequisiteFulfillment.studentID AND
      dbo.ProgramPrerequisite.ID = dbo.Id_PrerequisiteFulfillment.programPrerequisiteID
GROUP BY dbo.Student.personID, dbo.Person.firstName, dbo.Person.lastName
ORDER BY dbo.Student.personID

```

### 13) IdStudentsAndAllPrereqFulfillments

CSBSQL.MSCSIS - ...ereqFulfillments\*

The diagram shows three tables and their relationships:

- Person**: Columns include ID, firstName, lastName, bannerID.
- IdStudentsAndAllPrereqFulfillments**: Columns include personID, admissionYear, ID, description, programID.
- Id\_PrerequisiteFulfillment**: Columns include ID, studentID, programPrerequisiteID, note.

Relationships:

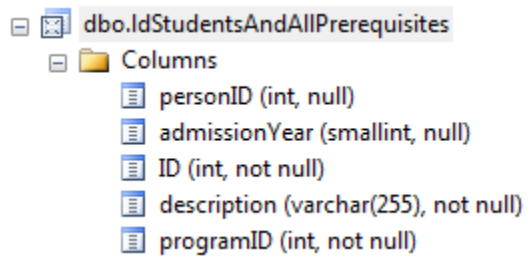
- Person (ID) is connected to IdStudentsAndAllPrereqFulfillments (personID) via a one-to-one relationship.
- IdStudentsAndAllPrereqFulfillments (ID) is connected to Id\_PrerequisiteFulfillment (ID) via a one-to-many relationship.

Column	Alias	Table	Output	Sort Type	Sort Order	Filter	Or...
ID	PersonID	Person	<input checked="" type="checkbox"/>				
firstName		Person	<input checked="" type="checkbox"/>				
lastName		Person	<input checked="" type="checkbox"/>				
ID	PrereqID	IdStudentsAndAllPrerequisites	<input checked="" type="checkbox"/>				
description		IdStudentsAndAllPrerequisites	<input checked="" type="checkbox"/>				
fulfillmentDate		Id_PrerequisiteFulfillment	<input checked="" type="checkbox"/>				
note		Id_PrerequisiteFulfillment	<input checked="" type="checkbox"/>				
approverPers...		Id_PrerequisiteFulfillment	<input checked="" type="checkbox"/>				
programID		IdStudentsAndAllPrerequisites	<input checked="" type="checkbox"/>				
personID		IdStudentsAndAllPrerequisites	<input type="checkbox"/>	Ascending	1		

```

SELECT TOP (100) PERCENT dbo.Person.ID AS PersonID, dbo.Person.firstName, dbo.Person.lastName, dbo.IdStudentsAndAllPrerequisites.ID AS PrereqID,
    dbo.IdStudentsAndAllPrerequisites.description, dbo.Id_PrerequisiteFulfillment.fulfillmentDate, dbo.Id_PrerequisiteFulfillment.note,
    dbo.Id_PrerequisiteFulfillment.approverPersonID, dbo.IdStudentsAndAllPrerequisites.programID
FROM    dbo.Person INNER JOIN
    dbo.IdStudentsAndAllPrerequisites ON dbo.Person.ID = dbo.IdStudentsAndAllPrerequisites.personID LEFT OUTER JOIN
    dbo.Id_PrerequisiteFulfillment ON dbo.IdStudentsAndAllPrerequisites.personID = dbo.Id_PrerequisiteFulfillment.studentID AND
    dbo.IdStudentsAndAllPrerequisites.ID = dbo.Id_PrerequisiteFulfillment.programPrerequisiteID
ORDER BY dbo.IdStudentsAndAllPrerequisites.personID
    
```

## 14) IdStudentsAndAllPrerequisites



dbo.IdStudentsAndAllPrerequisites
Columns
personID (int, null)
admissionYear (smallint, null)
ID (int, not null)
description (varchar(255), not null)
programID (int, not null)

15) IdUnfulfilledRequiredCores is designed to output records where the required course fulfillment field is null.

CSBSQL.MSCSIS -...RequiredCourses\*

The diagram shows the following table structures and relationships:

- Person**: ID, firstName, lastName, bannerID
- Student**: personID, studentTypeCode, applicationDate, admissionDate
- Id\_RequiredCourseFulfillment**: ID, programRequiredCourseID, studentID, fulfillmentDate
- Course**: ID, disciplineCode, courseNumber, title
- Program**: ID, title, catalog, universityID
- ProgramRequiredCourse**: ID, programID, courseID

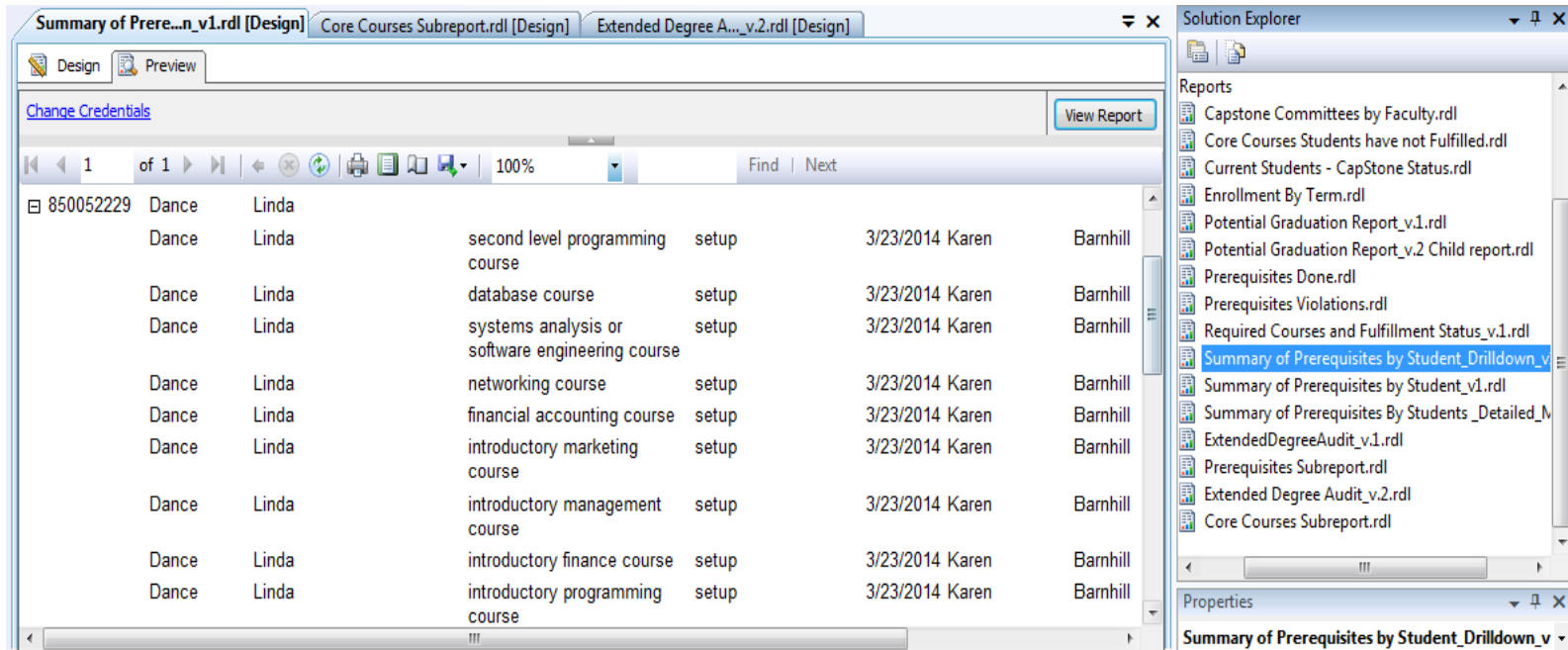
Relationships:

- Person (1) to Student (1)
- Student (1) to Id\_RequiredCourseFulfillment (1)
- Id\_RequiredCourseFulfillment (1) to Course (1)
- Program (1) to ProgramRequiredCourse (1)
- ProgramRequiredCourse (1) to Id\_RequiredCourseFulfillment (1)

Column	Alias	Table	Output	Sort Type	Sort Order	Filter	Or...	Or...	Or...
SELECT	TOP (100) PERCENT	dbo.Person.firstName, dbo.Person.lastName, dbo.Course.disciplineCode, dbo.Course.courseNumber, dbo.Id_RequiredCourseFulfillment.finalGrade, dbo.Id_RequiredCourseFulfillment.year, dbo.Id_RequiredCourseFulfillment.termCode, dbo.Student.personID							
FROM		dbo.Student INNER JOIN							
		dbo.Person ON dbo.Student.personID = dbo.Person.ID INNER JOIN							
		dbo.Program ON dbo.Student.programID = dbo.Program.ID INNER JOIN							
		dbo.ProgramRequiredCourse ON dbo.Program.ID = dbo.ProgramRequiredCourse.programID INNER JOIN							
		dbo.Course ON dbo.ProgramRequiredCourse.courseID = dbo.Course.ID LEFT OUTER JOIN							
		dbo.Id_RequiredCourseFulfillment ON dbo.Student.personID = dbo.Id_RequiredCourseFulfillment.studentID AND							
		dbo.Course.ID = dbo.Id_RequiredCourseFulfillment.programRequiredCourseID							
WHERE		(dbo.Id_RequiredCourseFulfillment.ID IS NULL)							
ORDER BY		dbo.Person.lastName, dbo.Student.personID							

## APPENDIX S: List of Reports Created

1. Summary of Prerequisites by Student  
Report type: Drilldown



- First, created simple report “Summary of Prerequisites by Student”.
- Second, created this drilldown report.
- Lastly, created a “Summary of Prerequisites by Student in a Matrix format.

These were exercises to work with the data in different report formats.

2. Prerequisites Done  
Report type: Tabular

The screenshot displays a report viewer window with the following data:

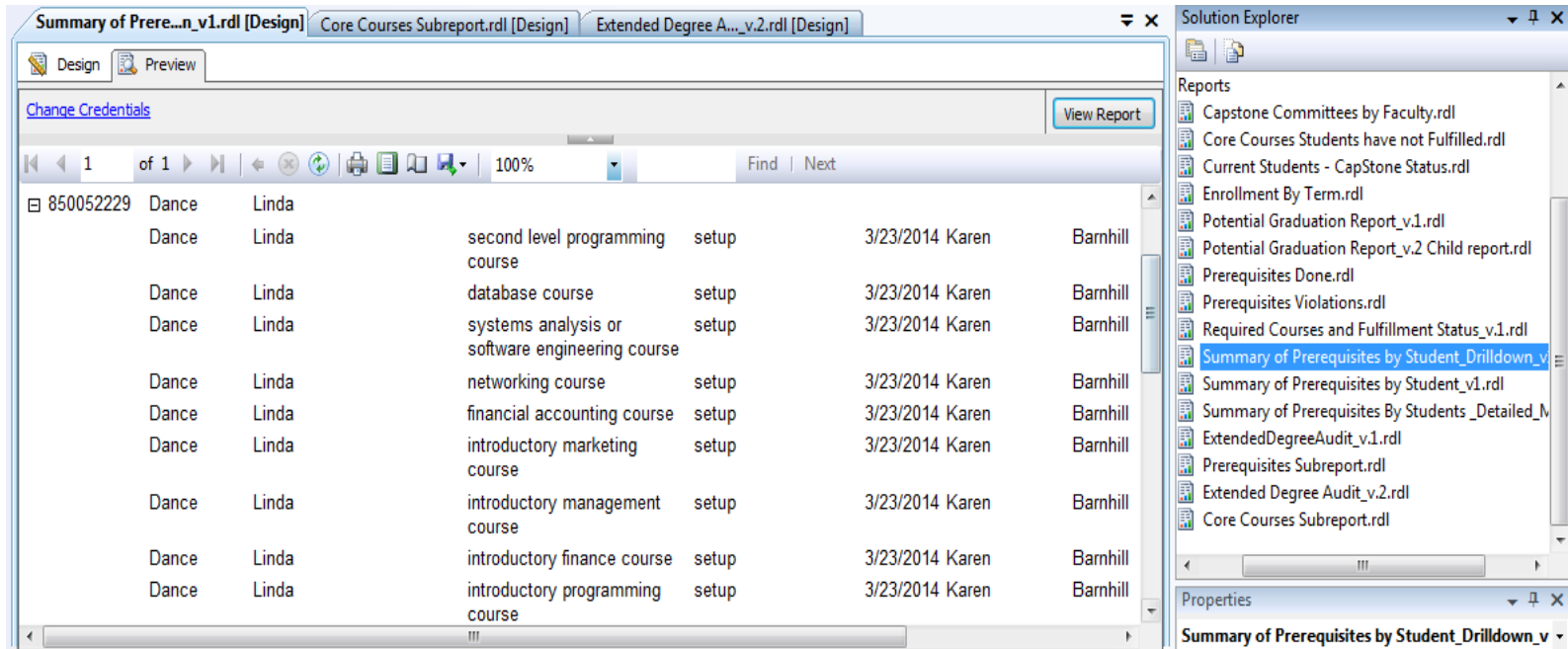
last Name	first Name	Total Prereqs Required	Prereqs Done
	Chris	10	0
	Ryan	10	0
	Eric	10	0
	Alisha	10	0
	Shaun	10	0
	Daniel	10	0
	Allen	10	0
	Ryan	10	0
	James	10	0
	Josh	10	0
	Max	10	0

The Solution Explorer on the right lists the following reports:

- MSCSIS.rds
- Shared Datasets
- Enrollment By Term Chart.rsd
- Reports
  - Capstone Committees by Faculty.rdl
  - Core Courses Students have not Fulfilled.rdl
  - Current Students - CapStone Status.rdl
  - Enrollment By Term.rdl
  - Potential Graduation Report\_v1.rdl
  - Potential Graduation Report\_v2 Child report.rdl
  - Prerequisites Done.rdl
  - Prerequisites Violations.rdl
  - Required Courses and Fulfillment Status\_v1.rdl
  - Summary of Prerequisites by Student\_Drilldown\_v1.rdl
  - Summary of Prerequisites by Student\_v1.rdl
  - Summary of Prerequisites By Students\_Detailed\_v1.rdl
  - ExtendedDegreeAudit\_v1.rdl
  - Prerequisites Subreport.rdl

This is a simple report but it is also a link in the Potential Graduation Report.

3. Summary of Prerequisites by Student  
Report type: Drilldown



- First, created simple report “Summary of Prerequisites by Student”.
- Second, created this drilldown report.
- Lastly, created a “Summary of Prerequisites by Student in a Matrix format.

These were exercises to work with the data in different report formats.

4. Prerequisites Violations Report

Report Type: Tabular, list

Prerequisites Violations						
last Name	first Name	Prereqs Done	Total Prereqs Require	Graduate Hours	admission	admission Year
[lastName]	[firstName]	[PrereqsDone]	[TotalPrereqsRequired]	[Graduate Hours]	[sionTerm]	[admissionYear]

5. Potential Graduation Report:

Potential Graduation Report\_v.1

ID	last Name	first Name	GPA	Graduate Hours Completed	Prereqs Done	Total Prereqs Required	Core Course Completed	Core Courses Required	Capstone Thesis Hours Completed
126	Dance	Linda	3.567	36	10	10	6	6	6

6. Current Students – Capstone Status

Current Students - CapStone Status

student Id	last Name	first Name	Capstone ID	Proposal Scheduled Date	Proposal Completed Date	Defense Scheduled Date	Defense Completed Date
126	Dance	Linda	899	2/28/2014	02/28/2014	04/22/2014	

## 7. Capstone Committees by Faculty

Capstone Committees by Faculty

Committee Member	Member Order	capstone ID title	first Name	last Name	# of Committees
Adhar		Gur			
Badarinathi		Ravija			
Brown		Jeff			
Burrus		Robert			
Ciner		Cetin			
Clark		Ulku			
Cloud		Avery			
Cummings		Jeffrey			
Dennison		Darwin			

8. Capstone Committees Chaired and Member Of

Capstone Committees By Person\_Detailed

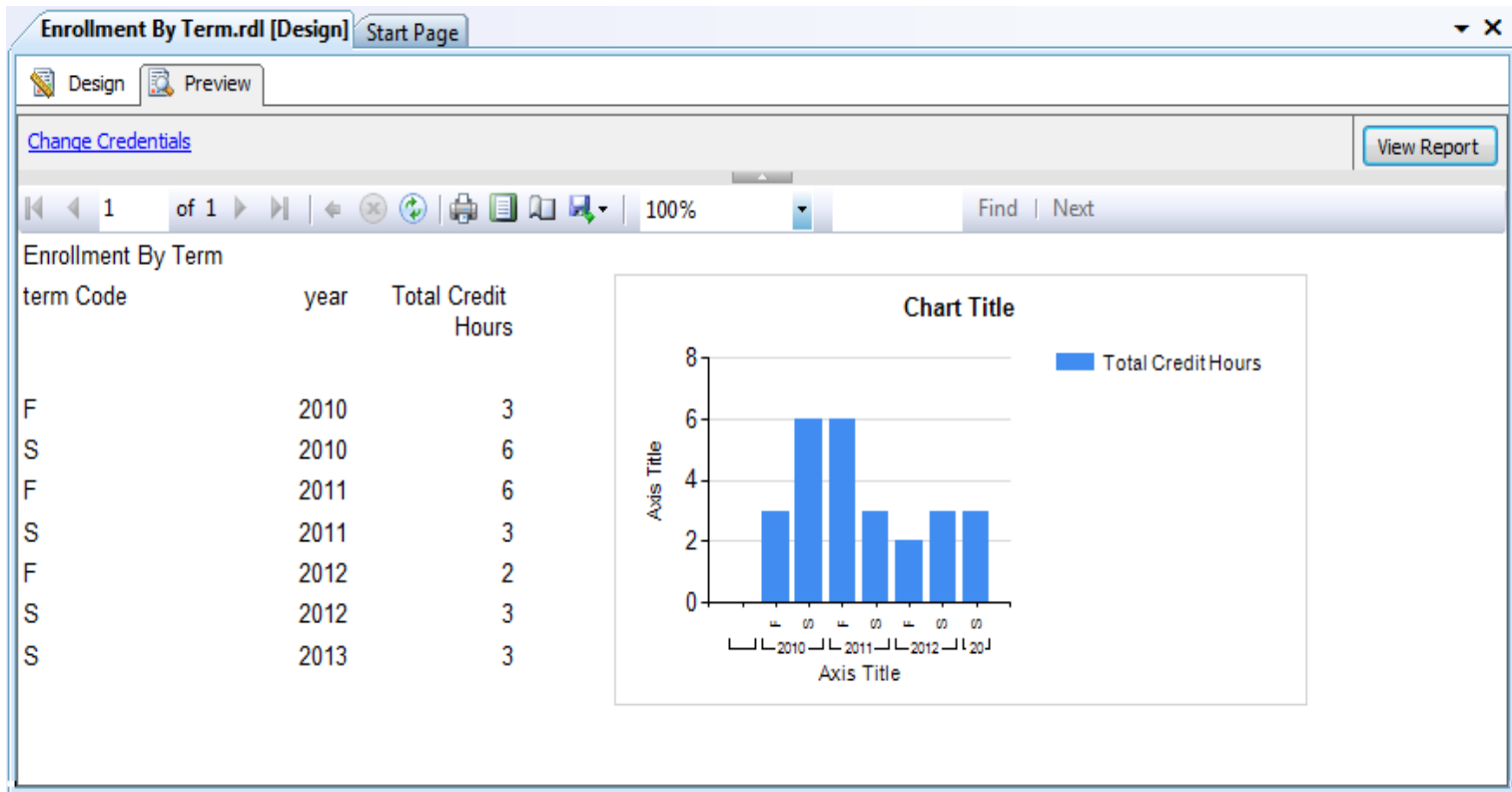
first Name	last Name	Chair Count	Member Count
	Kline	8	12
Tom	Janicki	14	16
Ron	Vetter	13	23
Charles	Laymon	0	3
Karl	Ricanek	8	4
Gene	Tagliarini	2	3
Jeff	Brown	0	6
Kevin	Matthews	0	2
Royce	Nobles	0	1
Ulku	Clark	2	6
Curry	Guinn	9	9
..	..	~	.

9. Core Courses Students have not Fulfilled - Drilldown

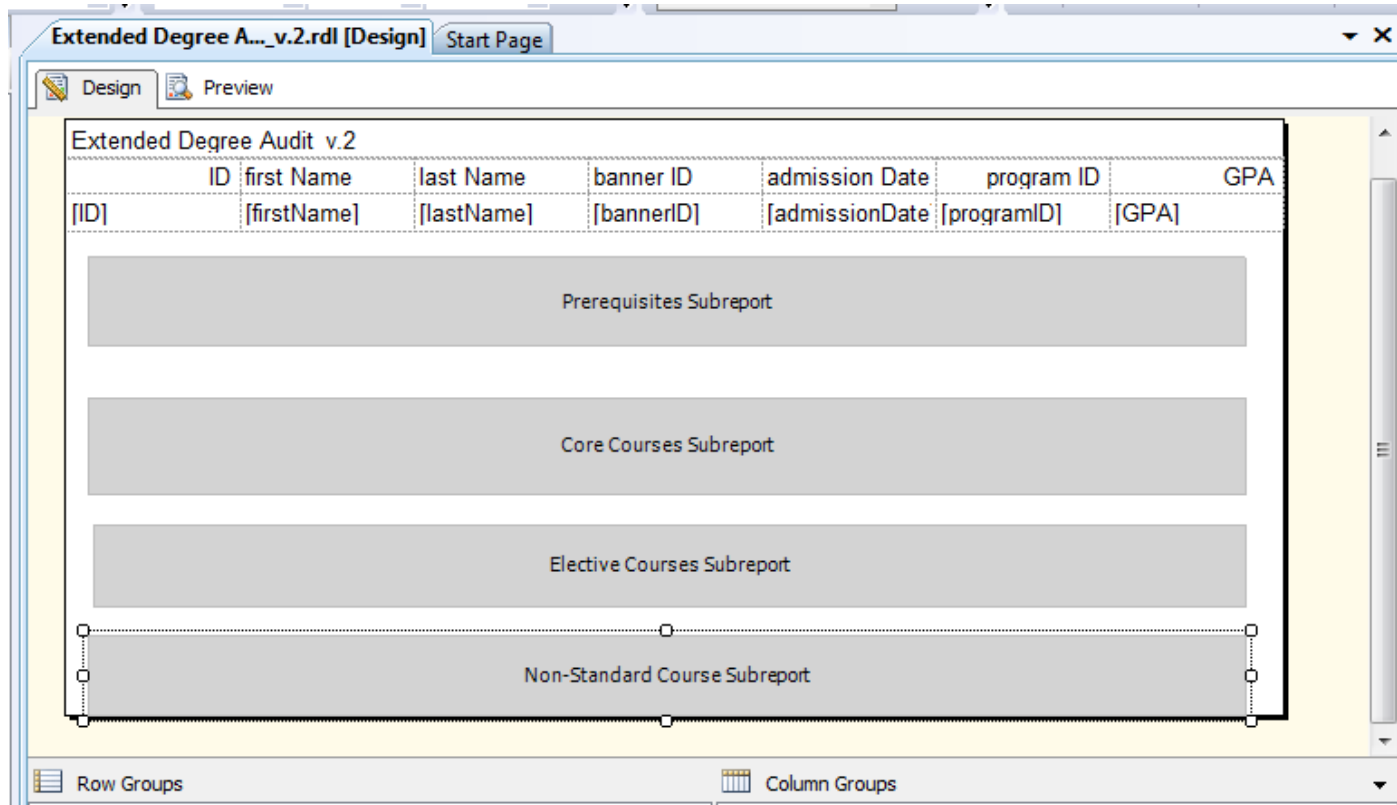
The screenshot shows a report viewer window titled "Core Courses Stude...filled.rdl [Design] Start Page". The window has a "Design" tab and a "Preview" tab. Below the tabs is a "Change Credentials" link and a "View Report" button. The report content is displayed in a table with the following columns: "course Number", "title", "person ID", "fulfillment Date", "fulfillment Date last Name", and "first Name". The table contains six rows of data, all with a "person ID" of 1. The "course Number" values are 532, 534, 544, 550, 555, and 565. The "fulfillment Date" and "fulfillment Date last Name" columns are empty. The "first Name" column is also empty. The report viewer includes navigation controls (back, forward, search) and a "Find | Next" button.

course Number	title	person ID	fulfillment Date	fulfillment Date last Name	first Name
532		1			
534		1			
544		1			
550		1			
555		1			
565		1			

### 10. Enrollment by Term

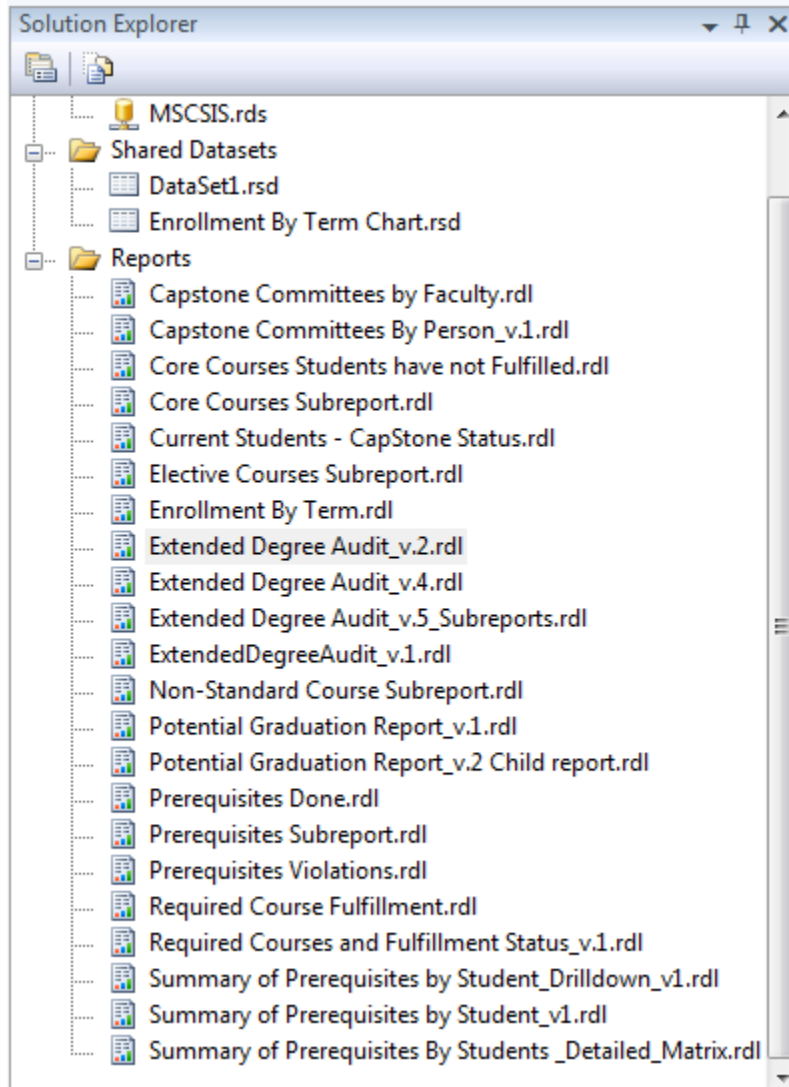


## 10. Extended Degree Audit

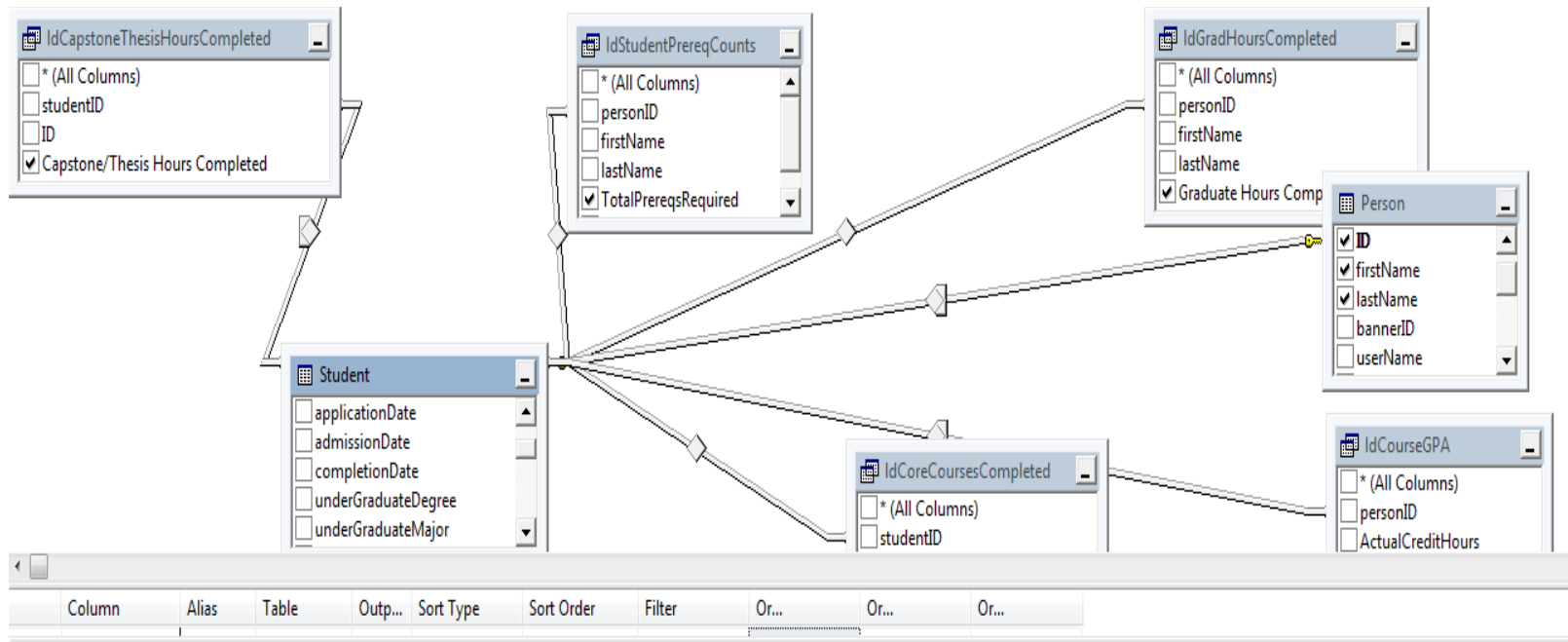


When going to preview, then next screen prompts for a student id once and all subreports are loaded.

## APPENDIX T: Reports built during learning process and before final edits and deployment to the Report Server



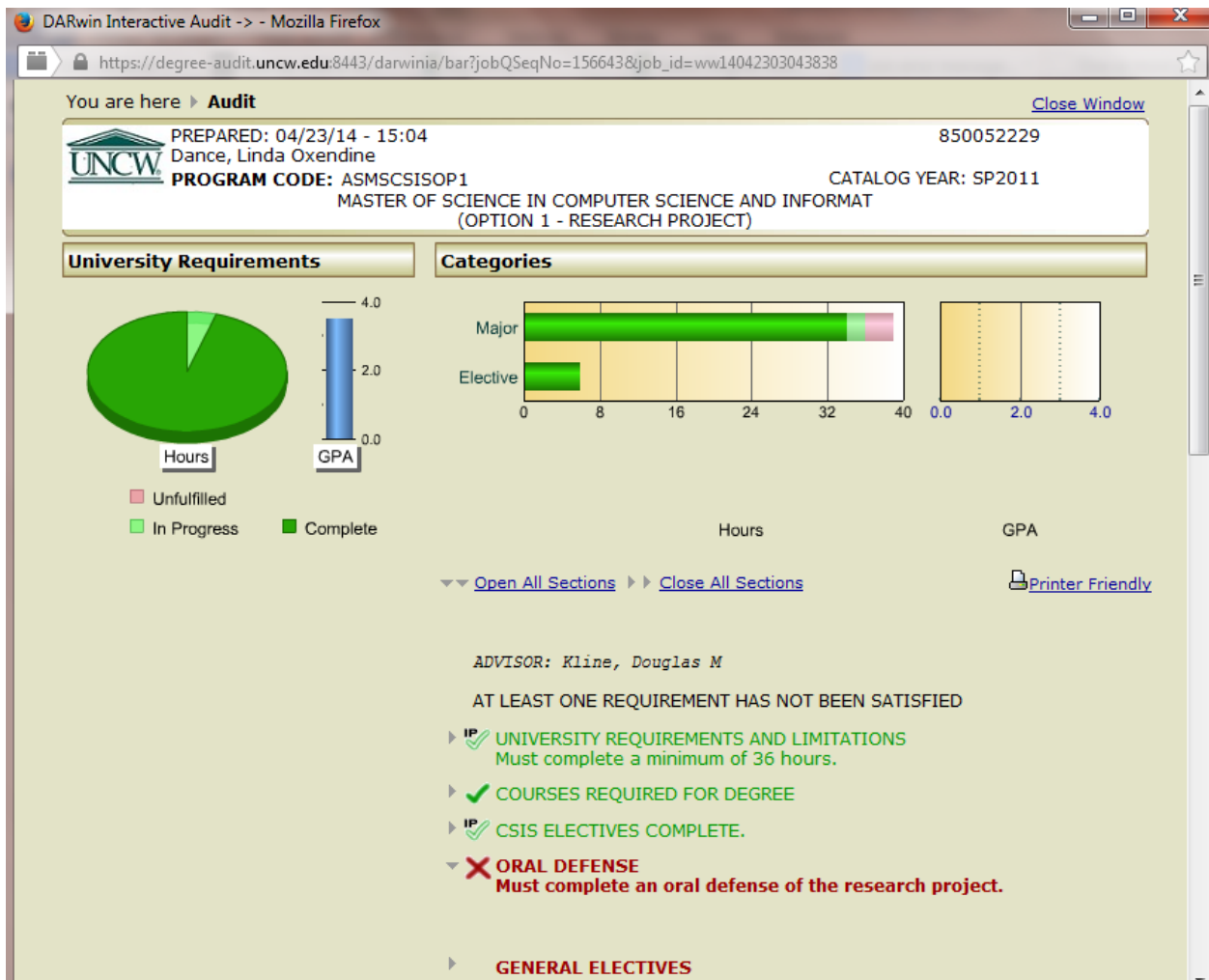
## APPENDIX U: Query Designer for Potential Graduation Report



```

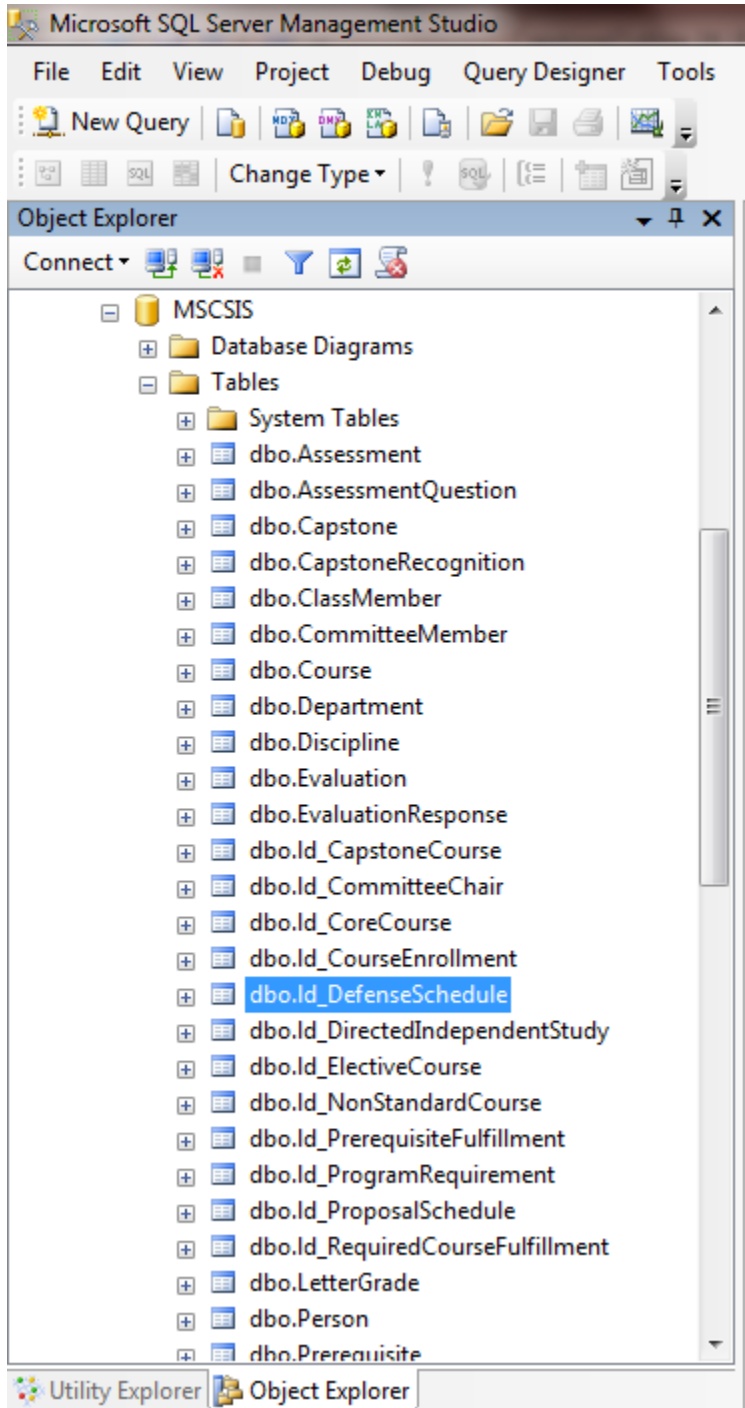
SELECT DISTINCT
    Person.lastName, Person.firstName, IdCourseGPA.GPA, IdStudentPrereqCounts.PrereqsDone, IdStudentPrereqCounts.TotalPrereqsRequired,
    IdCoreCoursesCompleted.[Core Course Completed], IdCoreCoursesCompleted.[Core Courses Required],
    IdCapstoneThesisHoursCompleted.[Capstone/Thesis Hours Completed], Person.ID, IdGradHoursCompleted.[Graduate Hours Completed]
FROM
    IdCapstoneThesisHoursCompleted LEFT OUTER JOIN
    IdGradHoursCompleted INNER JOIN
    IdStudentPrereqCounts INNER JOIN
    Student ON IdStudentPrereqCounts.personID = Student.personID INNER JOIN
    IdCoreCoursesCompleted ON Student.personID = IdCoreCoursesCompleted.studentID ON IdGradHoursCompleted.personID = Student.personID ON
    IdCapstoneThesisHoursCompleted.studentID = Student.personID RIGHT OUTER JOIN
    Person ON Student.personID = Person.ID RIGHT OUTER JOIN
    IdCourseGPA ON Student.personID = IdCourseGPA.personID
    
```

## APPENDIX V: Banner System Degree Audit

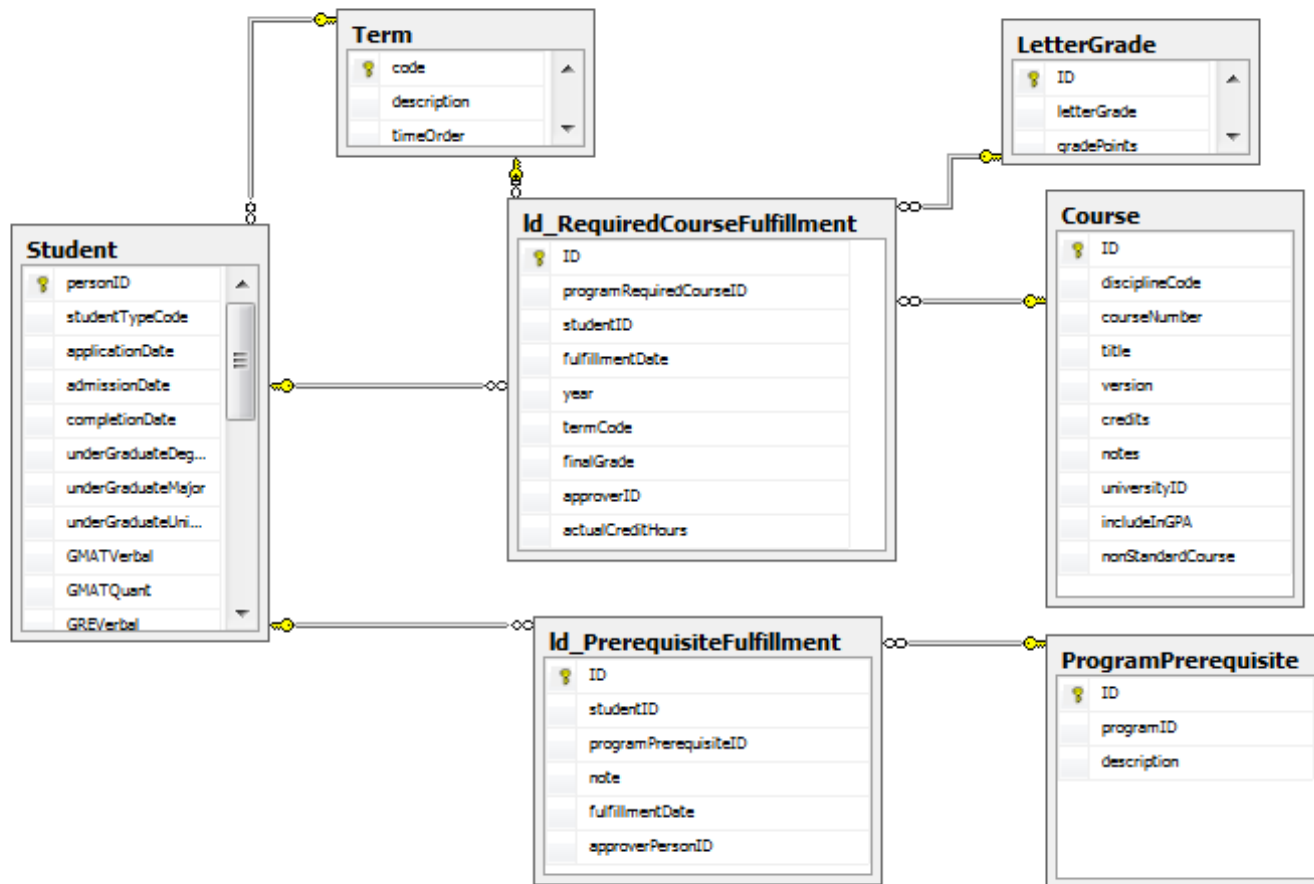


Each section in green is a drill down to the course details.

**APPENDIX W: Tables added during this project, with the naming convention of Id\_\*\*\***



## APPENDIX X: Data Model for Program Required Course Fulfillments



## APPENDIX Y: Prerequisite Courses already loaded prior to project

The screenshot displays the Microsoft SQL Server Management Studio interface. The Object Explorer on the left shows a tree view of the database schema, with the 'dbo' folder expanded to show various tables. The main window displays a table named 'CSBSQL.MSCSIS - ...ogramPrerequisite'. The table has three columns: 'ID', 'programID', and 'description'. The data is as follows:

ID	programID	description
1	1	second level programming course
2	1	database course
3	1	systems analysis or software engineering course
4	1	networking course
5	1	financial accounting course
6	1	introductory marketing course
7	1	introductory management course
8	1	introductory finance course
9	1	introductory programming course
10	1	data structures
* NULL	NULL	NULL

The status bar at the bottom indicates 'Ready' and 'Cell is Read Only.' The bottom of the window shows 'Utility Explorer' and 'Object Explorer' tabs.