MS IS 8300: Software Quality Management

1. Catalog Description
The current university catalog description of this course can be obtained in the University's Catalog: [http://www.gsu.edu/es/catalogs_courses.htm](http://www.gsu.edu/es/catalogs_courses.htm)

A recent university catalog description follows:

The course will cover methods and tools for achieving software quality assurance at various levels of a software system including at the module, subsystem, and system levels. State of the art tools and techniques including inspections, version control, and configuration management will be covered. Also, the role of standards, policies, and procedures will be discussed. The course will prepare students to develop a software quality assurance program in structured, organized ways. This course should provide practical knowledge of a variety of quality assurance techniques, and an understanding of some of the tradeoffs between techniques.

1.1 Prerequisites
**Required:** CSP 1-8, CIS 8030 (see instructor if waivers apply)

2.2 Course web site
Web sites for our course is on [www.onedrive.com](http://www.onedrive.com). See your email for details.

3. Overview
This class covers the principles of software development emphasizing processes and activities of quality assurance.

3.1 Intended audience
Anyone with a keen interest in software quality will do well in this course. It's mainly geared to produced IS Project Managers, Systems Analysts and Designers. However, Programmers, Database Administrators, and CIO’s may benefit from this course.

3.2 Learning objectives
Upon successful completion of this course, you will accomplish the following objectives and outcomes. In particular, students who complete this course will gain “Ready for work” skills, including:

1. Defining quality assurance plans
2. Applying quality assurance tools & techniques
More specific objectives, include the following:

1. Understand the quality assurance context
   a. Describe the challenges of software quality
   b. Define software quality

2. Understand software quality models
   a. Define a software quality model
   b. Describe components of a software quality assurance system
   c. Describe software quality plans

3. Understand SQA activities of the SDLC
   a. Relate software quality to the software development life-cycle
   b. Describe quality tools & techniques

4. Understand SDLC’s aimed at quality
   a. Describe a common software process model, and tailor it for increased quality.

5. Understand the role of testing
   a. Testing
   b. Runtime assurances & monitoring

6. Understand SQA management
   a. Describe process controls
   b. Describe quality metrics
   c. Describe quality costs

7. Understand SQA standards
   a. Describe common standards
   b. Describe IEEE standards

8. Understand SQA management organization
   a. Describe management roles, organization, and activities

9. Know software quality management, comprehensively
   a. Be able to analyze current issues in software quality assurance

10. Know a software-quality management topic, in detail

11. Demonstrate critical thinking, integrative reasoning, & communication skills

4 Schedule

The following table defines the schedule. However, the topics and readings may change according to the interests and abilities of the class. See the Academic Calendar.

On the web, the underlined items link to supporting information. Materials may be updated 24 hours prior to class; please check before attending class.
<table>
<thead>
<tr>
<th>#</th>
<th>Readings</th>
<th>In class</th>
<th>Due</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td><strong>Introducing SQA</strong>: §1, 2, 4, Epilogue (Galin 2004)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>SQA planning</strong>: §6, 23.4, 23.5, 24 (Galin 2004)</td>
<td>Review example SQA plans</td>
<td>M0: Email of group members</td>
</tr>
<tr>
<td>2.</td>
<td><strong>Introducing methodology &amp; cost</strong>: §7 (Galin 2004)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>SQA planning</strong>: §6, 23.4, 23.5, 24 (Galin 2004)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3.</td>
<td><strong>SQA management</strong>: §25, 26 (Galin 2004)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>SQA costs</strong>: iDAVE (Boehm et al. 2004)</td>
<td>Apply iDAVE</td>
<td>Student News presentation</td>
</tr>
<tr>
<td></td>
<td><strong>Identifying risks</strong>: §1-3,6 of (Carr et al. 1993); (DeMarco et al. 2003; Murthi 2002; Tiwana et al. 2004)</td>
<td>Specify project risks</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Draft an SQA plan</td>
<td></td>
</tr>
<tr>
<td>4.</td>
<td><strong>Measuring quality</strong>: §3(Galin 2004); (Schneidewind 2002); scan (IEEE 1998);</td>
<td>Specify quality goals and their measurement</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Student News presentation</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1, 2</td>
</tr>
<tr>
<td>5.</td>
<td><strong>Exam 1</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Social software practices</strong>: (Dabbish et al. 2012; Singer et al.; Storey et al. 2010; Vasilescu et al. 2013)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>FLOSS methodology &amp; tools</strong>: Agile §1 (Ambler et al. 2012); Continuous delivery(Wilson 2013); The</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>**Eclipse Way(Frost 2007; Velzen 2008); scrum (Li et al. 2010); Agile, MSF, &amp; CMMI (Anderson 2005); OSS Quality scan findings (Zhao et al. 2003); (Aberdour 2007); scan conclusions of (Boulanger 2005); scan conclusions of (Mockus et al. 2002); tools §13 (Galin 2004); (Murphy et al. 2012)</td>
<td>Gain cost removal</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Specify a process model &amp; tools</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Student News presentation</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>5, 6</td>
</tr>
<tr>
<td>6.</td>
<td><strong>SQA assessment</strong>: §16(Persse 2001); §16 (Deprez et al. 2008); §8(Schulmeyer 2008); §23 (Galin 2004)</td>
<td>Draft an assessment</td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Reviews &amp; templates</strong>: §8, 14, 15 (Galin 2004); RUP checklists</td>
<td>Review checklists &amp; reviews</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>OSS Requirements Assessment</td>
</tr>
<tr>
<td></td>
<td><strong>Testing</strong>: §9, 10 (Galin 2004); (Talby et al. 2006)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td><strong>Maintenance</strong>: §11, 12, 20 (Galin 2004); continuous quality (Cannizzo et al. 2008; Talby et al. 2006); (Robinson 2009; Robinson 2010)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7.</td>
<td><strong>Exam 2</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
5 Readings by Session

Readings provide content for class discussions. Thus, readings must be read prior to the class. For example, week 1 readings must be read prior to class on week 1. The readings are in order of importance. Thus, where there are a lot of readings, you may need to scan the last articles.

Don't get more than 1 week ahead of the class in the readings. Sometimes (mostly rarely) readings may be changed 1 week prior to their presentation in class.

1. Introducing SQA: §1, 2, 4, Epilogue (Galin 2004); (Sobejana 2014)
2. Introducing methodology & cost: §7 (Galin 2004)
3. SQA planning: §6, 23.4, 23.5, 24 (Galin 2004)
4. SQA management: §25, 26 (Galin 2004)
5. SQA costs: iDAVE (Boehm et al. 2004)
6. Identifying risks: §1-3, 6 of (Carr et al. 1993); (DeMarco et al. 2003; Murthi 2002; Tiwana et al. 2004)
7. Measuring quality: §3 (Galin 2004); (Schneidewind 2002); scan (IEEE 1998)
8. Exam 1
10. SQA assessment: §16 (Persse 2001); §16 (Deprez et al. 2008); §8 (Schulmeyer 2008); §23 (Galin 2004)
11. FLOSS methodology & tools: Agile §1 (Ambler et al. 2012); Continuous delivery (Wilson 2013); The Eclipse Way (Frost 2007; Velzen 2008); scrum (Li et al. 2010); Agile, MSF, & CMMI (Anderson 2005); OSS Quality scan findings (Zhao et al. 2003); (Aberdour 2007); scan conclusions of (Boulanger 2005); tools §13 (Galin 2004); (Murphy et al. 2012)
12. Reviews & templates: §8, 14, 15 (Galin 2004); RUP checklists
13. Testing: §9, 10 (Galin 2004); (Talby et al. 2006); (Murphy et al. 2013)
14. Maintenance: §11, 12, 20 (Galin 2004); continuous quality (Cannizzo et al. 2008; Talby et al. 2006); (Robinson 2009; Robinson 2010)
15. Exam 2

5.1 References

Students must have access to the primary textbook, (Galin 2004):

- **Primary Textbook:** Software Quality Assurance: From Theory to Implementation, Daniel Galin, Pearson 2004.

All other books can be accessed from E-book from Books24x7. Most articles have a URL, which can be used to download the article. (This assumes that you are on the university network directly or VPN. You may be prompted for your campus ID and password.) Gartner articles can be access through GSU’s Gartner Access link—search for the article name or ID.

Some articles may be only available from our web site. To find other articles, use the method described in section 10, How to scan CIS literature.)

Readings

5. Boulanger, A. "Open-source versus proprietary software: Is one more reliable and secure than the other?" IBM Systems Journal (44:2) 2005, pp 239-248. [http://dx.doi.org/10.1147/sj.442.0239](http://dx.doi.org/10.1147/sj.442.0239)

5.2 E-book from Books24x7

Consider the E-books as good resource; they are free to our students. See this note: http://www2.cis.gsu.edu/cis/news/newandnoteworthy2.asp Access from the GSU online library:
Software Quality Management Syllabus

http://homer.gsu.edu/search/databases/proxy/GLL25038; select the link Books24x7. You can also scroll down to Books 24x7 in the list of “databases”: http://homer.gsu.edu/search/databases/alphabetical#B

Search Books24x7 for books with the title “software quality” to see related books.

See for example:


Implementing the Capability Maturity Model by James K. Pinto John Wiley & Sons 2001


Measuring Information Systems Delivery Quality by Brian W. Duggan and Han Randolph (eds.) 1st edition 2003


Practical Insight into CMMI by Tim Kezer 1st edition 2000

Process Improvement with CMMI v1.2 and ISO Standards by Doris Hutauff and Harvey Stromberg 1st edition 2005


Software Process Improvement with CMM by Joseph Raynou 1st edition 1999

Software Testing as a Service by Ashfaq Ahmed 1st edition 2010


5.3 Software
Additionally, much of the software is available for download, either from the instructor, or from the CIS agreements with MSDNAA and the IBM Academic Initiative.

6 Evaluation
Students are evaluated by the deliverables summarized in Table 1. The course credits are earned according to the following Table 1.

Table 1 Relative weights assigned to course deliverables.

<table>
<thead>
<tr>
<th>Assignment</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exam 1</td>
<td>20</td>
</tr>
<tr>
<td>Exam 2</td>
<td>30</td>
</tr>
<tr>
<td>In class exercises</td>
<td>5</td>
</tr>
<tr>
<td>SQA Headlines</td>
<td>5</td>
</tr>
<tr>
<td>SQAP draft</td>
<td>10</td>
</tr>
<tr>
<td>SQAP final</td>
<td>15</td>
</tr>
<tr>
<td>OSS Requirements Assessment</td>
<td>15</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>100</strong></td>
</tr>
</tbody>
</table>

The following table overviews how credit will be assigned. Note that all group work includes a peer review, which can distinguish an individual's assigned points from the group's assigned points. (See Self-Managed Teams in the Workload Expectations section.)

Table 2 Grading standards.

<table>
<thead>
<tr>
<th>Work quality</th>
<th>Percent</th>
</tr>
</thead>
<tbody>
<tr>
<td>Absolutely fantastic, walk on water, overflow grade</td>
<td>110</td>
</tr>
<tr>
<td>Excellent answer on all counts</td>
<td>100</td>
</tr>
<tr>
<td>Excellent answer on most counts</td>
<td>90</td>
</tr>
<tr>
<td>Very good answer, but not excellent</td>
<td>80</td>
</tr>
<tr>
<td>Professionally done and adequate</td>
<td>70</td>
</tr>
<tr>
<td>Inadequate, needs work</td>
<td>60</td>
</tr>
<tr>
<td>Varying degrees of inadequacy</td>
<td>0 - 50</td>
</tr>
</tbody>
</table>

The following breakout depicts how grades will be assigned under this system.
### Grade Percentage

<table>
<thead>
<tr>
<th>Grade</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>A+</td>
<td>≥ 97</td>
</tr>
<tr>
<td>A</td>
<td>≥ 90</td>
</tr>
<tr>
<td>A-</td>
<td>≥ 87</td>
</tr>
<tr>
<td>B+</td>
<td>≥ 83</td>
</tr>
<tr>
<td>B</td>
<td>≥ 80</td>
</tr>
<tr>
<td>B-</td>
<td>≥ 77</td>
</tr>
<tr>
<td>C+</td>
<td>≥ 73</td>
</tr>
<tr>
<td>C</td>
<td>≥ 70</td>
</tr>
<tr>
<td>C-</td>
<td>≥ 67</td>
</tr>
<tr>
<td>D</td>
<td>≥ 60</td>
</tr>
<tr>
<td>F</td>
<td>&lt; 60</td>
</tr>
</tbody>
</table>

### In class exercises

Each exercise is intended as a group effort, which illustrates important concepts introduced in the associated readings. More detailed description and associated materials shall be found on the course web site.

- **Deliver** your results to the course web site during class (only).
  - Authors shall receive credit for each in-class exercise.
    - Prominently (at the top) of the delivered document, place the names of authors.
    - Do not include the name of anyone who is absent or did not contribute. Doing so will result in zero credit for all ‘authors’.
    - Late deliverables (after class) shall receive zero credit.

### 7.1 Galin cost removal

### 7.2 Specify a process model & tools

Given a problem statement requiring a software-intensive solution, define a software-development process model. Within the time provided, ensure that the most significant activities are described and placed in their proper order. Be prepared to describe the goals of the process (e.g., fast delivery with moderate quality) and defend how the process can achieve those goals.

### 7.3 Specify quality goals and their measurement

Given a requirements specification for a software-intensive solution, (1) characterize requirements using a quality model—some requirements may be associated with more than one quality factor, (2) rewrite some of the given requirements or write new requirements so that most factors of the model are illustrated by at least one requirement.

### 7.4 Specify project risks

Given a problem statement requiring a software-intensive solution, generate a list of project risks, some annotated with severity and/or priority. Consider some means to mitigate some risks, and select some means based on perceived ROI.
7.5 Review example SQA plans

7.6 Draft an SQA plan

7.7 Specify SQA organization

7.8 Review checklists & reviews

7.9 Draft an assessment

7.10 Draft a test plan

7.11 Apply iDAVE

Given a problem statement requiring a software-intensive solution, estimated costs, a variety of goals concerning levels of reliability, calculate ROI using the iDAVE method. Consider the usefulness of iDAVE as well as the general usefulness of trading-off costs against reliability.

7.12 Assessment project

7.13 SQA Headlines

Your group will present a software quality issue that has been in the headlines within the last 10 years (or 20 years if it is still considered a significant case). The goals of the assignment are to:

- Show the relevance of SQA for everyone
- Present SQA course materials in the context of real, ongoing, problems
- Generate discussion about SQA—in particular, tradeoffs, decision-making, and consequences of SQA for organizations and people

In your presentation:

- Show the news article(s), blogs, etc.
- Present a few PowerPoint slides summarizing the articles, the SQA issues, and provide issues and questions for subsequent discussion
  - Consider issues such as why the error was not detected in their SQA, how their SQA has changed as a result, and your opinion on the problem (e.g., is it an isolated incident or a reflection of SQA in general?)
- Moderate a brief discussion

Deliver to our web site:

- Your PowerPoint slides
- Any notes that might be relevant to aid further study

As illustrative examples, consider the following (you may choose among them or find another issue):

1. **Knight Capital Group Inc. (KCG)**'s $440 million trading loss stemmed from old computer software that was inadvertently reactivated when a new program was installed, according to two people briefed on the matter. [Summer, 2012]

2. The Royal Bank of Scotland is set to pay out up to £125m in compensation to its customers after a disastrous computer glitch back in June [2012].

3. Officials at stock exchange operator BATS Global Markets said that a software bug caused an outage that halted trading of its own shares Friday, the first day of its initial public offering (IPO), and also put a temporary stop to trading in shares of computer giant Apple. [March, 2012]

4. The Coventry-based black cab maker said that "Due to a combination of system and procedural errors, a number of transactions relating to 2010 and 2011 and some residual balances from the previous system
were not properly processed through the new IT system," Manganese Bronze said in the statement. "This problem led to the over-statement of stock and under-statement of liabilities in the financial statements of previous years." "The cumulative effect of these errors is an estimated £3.9m understatement of historical losses which go back over several years, although the work to apportion the loss between previous years is not yet complete," it added. [August, 2012]

5. Microsoft suspends new Windows Phone apps due to a bug. According to Microsoft, this particular error affected handsets that were upgraded to Windows Phone 7.5 Mango from an previous version of the mobile operating system, which is the main reason behind why they have decided to suspend all the publication of new Windows Phone apps.

6. In a letter on April 15, AXA Rosenberg told its clients — which include mutual funds, pension funds and holders of separately managed accounts — that it had made a “coding error” that affected returns in its various portfolios in ways that had yet to be determined. The problem came to light in the United States more than a year ago, when the Securities & Exchange Commission ordered the fund manager Axa Rosenberg to make good losses worth $217m (£139m) after a significant “coding error” with its risk-modelling program. The fund manager was also ordered to pay a $25m “civil penalty”, partly for being slow to report the problem, which was first spotted in 2009.

7. Toyota issued a recall for hybrid anti-lock brake software in February 2010.

8. The Northeast blackout of 2003 was a widespread power outage that occurred throughout parts of the Northeastern and Midwestern United States and Ontario, Canada, on Thursday, August 14, 2003, just before 4:10 p.m. EDT (UTC−04). While some power was restored by 11 p.m., many did not get power back until 8 a.m. the next day. At the time, it was the second most widespread blackout in history, after the 1999 Southern Brazil blackout. The blackout affected an estimated 10 million people in Ontario and 45 million people in eight U.S. states. A software bug known as a race condition existed in General Electric Energy's Unix-based XA/21 energy management system. Once triggered, the bug stalled FirstEnergy's control room alarm system for over an hour. System operators were unaware of the malfunction; the failure deprived them of both audio and visual alerts for important changes in system state.

9. On February 25, 1991, an Iraqi Scud hit the barracks in Dhahran, Saudi Arabia, killing 28 soldiers from the US Army's 14th Quartermaster Detachment. A government investigation revealed that the failed intercept at Dhahran had been caused by a software error in the system's clock.

10. Therac-25 was involved in at least six accidents between 1985 and 1987, in which patients were given massive overdoses of radiation, approximately 100 times the intended dose.

11. Medtronic - Heart device Vulnerable to remote attack. Theoretical Threat discovered by researchers from Univ. of Washington and Univ. of Massachusetts.


13. Facebook Nasdaq stock auction.

See this site for more ideas Google software error or fault in News, or check this site:

- [http://www5.in.tum.de/~huckle/bugse.html](http://www5.in.tum.de/~huckle/bugse.html)

# 8 Homework

See the web site for the most recent and detailed information on these assignments. The following is provided as an introduction to each assignment.

## 8.1 Group project

1. For each milestone, deliver via email:
   - A single file (zip if large) containing
     - Requirements artifacts as specified in the assignment description (see the web site)
     - Summarize group work:
       - Tasks completed by each member
       - Percentage of the total work completed by each member
   - Your Personal Statement
Each team member must a **confidential personal statement** (maximum of one page per member) highlighting:

- His or her contribution to the project
- Comments on the contribution of other group members
- Lessons learned from the project
- Comments on the tool and
- Any other issues or concerns
- Comments about the course

Please use the following SUBJECT LINE: CIS 8300-personal-statement

**Do NOT** use attachments. Submit it as text in the body of the message.

### 8.1.1 SQAP draft

Your group shall deliver a SQAP using a variation of the IEEE template. Your SQAP shall include:

- A SQAP based on the IEEE template
  - Download the software quality process model document template (CIS.GSU-SQA_plan_template.doc) from our web site. Edit the document to become your SQAP.
  - Focus on the following aspects of the project:
    - Section 1.0: specify the primary goals of the project with respect to quality; for example, time-to-market or reliability in the face of a mission or safety critical project.
    - Section 3.2: sketch the development methodology, including its major tasks, and specify (in more detail) those tasks that will address quality. (The implication is that process implies quality, even in light-weight methods like Agile.) Review CMMI, IEEE, and the online IBM RUP guide to practices and processes. Select those most relevant. Ideally, specify the methodology as hierarchical list of tasks (like MS Project Gantt chart).
    - Section 5: specify a quality model. Consider the factor models introduced by Galin and customize them to your project.
    - Section 7: specify a testing strategy. Consider possible testing techniques and select those most relevant considering a limited budget.
    - Section 14: specify a risk model. Consider the SEI risk taxonomy as well as those risks suggest in your readings. Customize the risks to your project.

- **Deliver** your draft SQAP an email attachment and present to the class

It is difficult to assign points to section because different projects will (correctly) emphasize different sections. Nevertheless, here’s an outline.

<table>
<thead>
<tr>
<th>Section</th>
<th>Points</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
</tr>
<tr>
<td>3</td>
<td>15</td>
</tr>
<tr>
<td>5</td>
<td>15</td>
</tr>
<tr>
<td>6</td>
<td>10</td>
</tr>
<tr>
<td>7</td>
<td>15</td>
</tr>
<tr>
<td>9</td>
<td>10</td>
</tr>
<tr>
<td>14</td>
<td>15</td>
</tr>
<tr>
<td>Professionalism</td>
<td>10</td>
</tr>
</tbody>
</table>
8.1.2 SQAP presentation

You team represents the SQA unit of a large software development company. You are to present your SQA plan to the company executives for approval. A recommended outline for your presentation is as follows:

1. Team introduction
2. Problem introduction
   a. Describe the context of the software development effort, such as project domain, size (estimated effort), developers, required level of quality
3. SQA overview
   a. Quality goals, time to market vs. quality target
   b. Quality and project risks
   c. Unique aspects of plan
4. SQA details
   a. How SQA activities integrate into development process
   b. SQA management
   c. SQA assessments, reviews, and important (acceptance) tests
5. Conclusion
   a. Justify SQA goals
   b. Justify SQA plan meets goals

- Deliver a PowerPoint presentation of your topic on the scheduled date
  - You will have about 20 minutes

8.1.3 SQAP final

The main purpose of the final version is to address any issues that were identified in the first version of your plan. Additionally, introduce concepts presented in class since the last version. This includes the following concepts:

- Agile process concept—in particular, fewer development activities with unit testing and customer acceptance tests integrated within the many iterations.
- Economic concepts—in particular, consider the ROI your organization may obtain for an increased level of quality. Also, consider benefits not directly addressed in a classic ROI calculation, such as reputation.
- Runtime testing concepts—in particular, testing and requirements analysis after the product has been delivered to the customer.

Moreover, with increased understand the term provides, revisit the software development activities and management lines. Consider revising your software development process and management structure to increase quality.

- A separate 1-2 page executive summary document that justifies your process model, with special attention clarifying the ROI of the quality related activities. It’s important to note that your model may be much different from that of your peers, yet it may be well-suited for its purpose. Describe the purpose of the model and justify how your SQAP achieves the goals of its particular purpose.
  - Executive summary document is due in the final version.
- Deliver your final SQAP as an email attachment

8.1.4 OSS Requirements Assessment

Apply an assessment method to an open source project. (Several assessment methods will be accepted.) See the web site for a more complete description.

9 Examinations

Online review guides to be updated one-half week prior to the exam.

9.1 Exam 1

See the online exam review for a description.

9.2 Exam 2

Comprehensive! Similar in nature to a certification exam. See the online exam review for a description. Page 13/16
10 How to scan CIS literature

10.1 Software
Install EndNote:
1. Free EndNote @ GSU

10.2 Literature review
Search for peer reviewed articles using keywords:
2. Scan the web
   a. www.google.com
3. Scan the web using scholar search engines
      i. Set the Google Scholar Preferences to
         1. Show library access links for Georgia State University
         2. Show links to import citations into EndNote
   c. http://citeseer.ist.psu.edu/
4. Scan using library databases (@GSU)
   a. http://www.galileo.usg.edu
   b. In particular, the following databases
      i. ABI/INFORM Complete
      ii. ACM Digital Library
      iii. IEEE Xplore

11 Workload Expectations
Students should plan for 2 - 3 hours of work outside of class each week for each course credit hour. Thus, a 3-credit course averages between 6 and 9 hours of student work outside of the classroom, each week. See GSU sites for Academic Success:
- http://www2.gsu.edu/~wwwcam/incept/successtips.htm
- http://www2.gsu.edu/~wwwctr/sac/StudySkills.htm

Students must take responsibility for their learning. In contrast to high school, college has fewer opportunities for student teacher interactions. Consequently, students must prepare to gain the most from each interaction.

Self-Managed Teams: Teams will be allowed for some activities during the term. Please note that unless the activity is explicitly identified as a "team activity", I expect everyone to perform their own work (your hands on the keyboard). For team activities, you will be allowed to work with partners (of your choosing).

- Initial teams must be established by the second week of classes. Established teams may continue working together on subsequent team activities. Team membership may change during the term, if problems arise. However, team members must be designated within one week of the due date for the team activity. Exception: you may withdraw from a team at any time and submit an assignment individually.
- Teams will submit one assignment for all team members. In most cases, each member of the team will get the same score. However, an individual's score may be reduced at the discretion of the instructor.
- Each team assignment must include the following:
  o Tasks completed by each member.
  o Percentage of the total work completed by each member.
- Any individual with a low team contribution will be removed from their team.

Arbitration: There will be a one-week arbitration period after graded activities are returned. Within that one-week period, you are encouraged to discuss any assumptions and/or misinterpretations that you made on the activity that may have influenced your grade.

Attendance: If you are unable to attend a class session, it is your responsibility to acquire the class notes, assignments, announcements, etc. from a classmate. The instructor will not give private lectures for those that miss class.
**Submission of Deliverables:** Unless specific, prior approval is obtained, no deliverable will be accepted after the specified due date.

If you have a legitimate personal emergency (e.g., health problem) that may impair your ability to submit a deliverable on time, you must take the initiative to contact the instructor before the due date/time (or as soon after your emergency as possible) to communicate the situation.

**Make-up exams will not be given:** However, if a student has a planned absence, he or she may take the exam earlier with the permission of the instructor.

## 12 Student Behavior

Behavior in class should be professional at all times. People must treat each other with dignity and respect in order for scholarship to thrive. Behaviors that are disruptive to learning will not be tolerated and may be referred to the Office of the Dean of Students for disciplinary action.

### 12.1 Discrimination and harassment

Discrimination and/or harassment will not be tolerated in the classroom. In most cases, discrimination and/or harassment violates Federal and State laws and/or University Policies and Regulations. Intentional discrimination and/or harassment will be referred to the Affirmative Action Office and dealt with in accordance with the appropriate rules and regulations.

Unintentional discrimination and/or harassment is just as damaging to the offended party. But, it usually results from people not understanding the impact of their remarks or actions on others, or insensitivity to the feelings of others. We must all strive to work together to create a positive learning environment. This means that each individual should be sensitive to the feelings of others, and tolerant of the remarks and actions of others. If you find the remarks and actions of another individual to be offensive, please bring it to their attention. If you believe those remarks and actions constitute intentional discrimination and/or harassment, please bring it to my attention.

### 12.2 Official CIS department class policies

1. Prerequisites are strictly enforced. Students failing to complete any of the prerequisites with a grade of “C” or higher will be administratively withdrawn from this course with loss of tuition fees. **There are no exceptions, except as granted by the instructor with the approval of the department.**

2. Students are expected to attend all classes and group meetings, except when precluded by emergencies, religious holidays, or bona fide extenuating circumstances.

3. Students who, for non-academic reasons beyond their control, are unable to meet the full requirements of the course should notify the instructor, by email, as soon as this is known and prior to the class meeting. Incompletes may be given if a student has ONE AND ONLY ONE outstanding assignment.

4. A “W” grade will be assigned if a student withdraws before mid-semester if (and only if) he/she has maintained a passing grade up to the point of withdrawal. Withdrawals after the mid-semester date will result in a grade of “WF”. See the GSU catalog or registrar’s office for details.

5. Spirited class participation is encouraged and informed discussion in class is expected. This requires completing readings and assignments before class.

6. All exams and individual assignments are to be completed by the student alone with no help from any other person.

7. Collaboration within groups is encouraged for project work. However, collaboration between project groups will be considered cheating.

8. Copying work from the Internet without a proper reference is considered plagiarism and subject to disciplinary action as delineated in the GSU Student Handbook.

9. Any non-authorized collaboration will be considered cheating and the student(s) involved will have an Academic Dishonesty charge completed by the instructor and placed on file in the Dean’s office and the CIS Department. All instructors regardless of the type of assignment will apply this Academic Dishonesty policy equally to all students. Abstracted from GSU’s Student Handbook Student Code of Conduct “Policy on Academic Honesty and Procedures for Resolving Matters of Academic Honesty”

   a. [http://www2.gsu.edu/7Ewwdos/codeofconduct_conpol.html](http://www2.gsu.edu/7Ewwdos/codeofconduct_conpol.html)
As members of the academic community, students are expected to recognize and uphold standards of intellectual and academic integrity. The University assumes as a basic and minimum standard of conduct in academic matters that students be honest and that they submit for credit only the products of their own efforts. Both the ideals of scholarship and the need for fairness require that all dishonest work be rejected as a basis for academic credit. They also require that students refrain from any and all forms of dishonorable or unethical conduct related to their academic work.

Students are expected to discuss with faculty the expectations regarding course assignments and standards of conduct. Here are some examples and definitions that clarify the standards by which academic honesty and academically honorable conduct are judged at GSU.

**Plagiarism.** Plagiarism is presenting another person's work as one's own. Plagiarism includes any paraphrasing or summarizing of the works of another person without acknowledgment, including the submitting of another student's work as one's own. Plagiarism frequently involves a failure to acknowledge in the text, notes, or footnotes the quotation of the paragraphs, sentences, or even a few phrases written or spoken by someone else. The submission of research or completed papers or projects by someone else is plagiarism, as is the unacknowledged use of research sources gathered by someone else when that use is specifically forbidden by the faculty member. Failure to indicate the extent and nature of one's reliance on other sources is also a form of plagiarism. Any work, in whole or part, taken from the Internet or other computer-based resource without properly referencing the source (for example, the URL) is considered plagiarism. A complete reference is required in order that all parties may locate and view the original source. Finally, there may be forms of plagiarism that are unique to an individual discipline or course, examples of which should be provided in advance by the faculty member. The student is responsible for understanding the legitimate use of sources, the appropriate ways of acknowledging academic, scholarly or creative indebtedness, and the consequences of violating this responsibility.

**Cheating on Examinations.** Cheating on examinations involves giving or receiving unauthorized help before, during, or after an examination. Examples of unauthorized help include the use of notes, texts, or “crib sheets” during an examination (unless specifically approved by the faculty member), or sharing information with another student during an examination (unless specifically approved by the faculty member). Other examples include intentionally allowing another student to view one's own examination and collaboration before or after an examination if such collaboration is specifically forbidden by the faculty member.

**Unauthorized Collaboration.** Submission for academic credit of a work product, or a part thereof, represented as its being one's own effort, which has been developed in substantial collaboration with another person or source or with a computer-based resource is a violation of academic honesty. It is also a violation of academic honesty knowingly to provide such assistance. Collaborative work specifically authorized by a faculty member is allowed.

**Falsification.** It is a violation of academic honesty to misrepresent material or fabricate information in an academic exercise, assignment or proceeding (e.g., false or misleading citation of sources, the falsification of the results of experiments or of computer data, false or misleading information in an academic context in order to gain an unfair advantage).

**Multiple Submissions.** It is a violation of academic honesty to submit substantial portions of the same work for credit more than once without the explicit consent of the faculty member(s) to whom the material is submitted for additional credit. In cases in which there is a natural development of research or knowledge in a sequence of courses, use of prior work may be desirable, even required; however the student is responsible for indicating in writing, as a part of such use, that the current work submitted for credit is cumulative in nature.