

# ISAT 340--Software Development--Syllabus

Fall 2016

## Course Description

Software is a tool for solving scientific and technological problems. This course provides an overview of what constitutes “software” and gives students the opportunity to learn about and demonstrate their capacity to follow the steps and techniques required to develop modern software solutions to scientific and technological problems. Heavy emphasis will be placed on analysis and design, particularly with respect to the social context of the problems targeted by potential software solutions.

## Course Outcomes

As a result of taking this course, learners may:

- Be able to describe what is meant by the word “software” with particular emphasis on modern contexts which are more likely to result in web and/or mobile applications as opposed to desktop applications
- Have developed concrete artifacts to demonstrate that they have experience with, and capacity to perform tasks necessary to develop software, among them:
  - **Problem Statement:** Articulating a problem to be addressed by software, specifically focusing on why software is the right tool to address this problem, as opposed to other non-software approaches to problem-solving
  - **Social Context Analysis and Scenarios:** Describing the social context in which the problem is relevant, including an analysis of:
    - the needs/challenges of the specific people, or groups of people, to whom this problem applies
    - the people/groups who may provide resistance to this particular approach
    - a more or less comprehensive history of what other kinds of solutions have been applied to this problem, their outcomes, and how the lessons from those experiences are incorporated into the current design
  - **Mockups:** Thorough paper-and-pencil mockups of an application design, including rounds of revisions
  - **Architecture Diagram (Generic):** Analysis and selection of an appropriate software architecture, e.g. client-server, standalone, etc., for the proposed application
  - **Entity-Relationship Diagram and Database Schema:** Analysis and abstraction of a data model from the screen designs that results in a concrete database design

- **Architecture Diagram (Specific):** Analysis and selection of an appropriate set of technologies that will be used to implement software, including:
  - A programming language or languages
  - A database technology
  - A server technology
  - An app development framework
  - A web hosting provider
  - Any other services, e.g. notifications, pub/sub, etc., necessary to implement the solution
  - Unit and Integration testing framework(s)
- **Unit Tests and Integration Tests:** Automated software tests that specify at a granular level what the software is supposed to be able to do
- **Source Code:** The actual code that implements the software solution, stored in a public repository, e.g. GitHub
- **Database:** A concrete implementation of the data model articulated previously
- Have developed ancillary skills required to be effective at software development, and artifacts to demonstrate such skills, including, but not limited to:
  - Setting up a development environment including elements like:
    - Code editor
    - Necessary compilers, debuggers, unit testing frameworks, etc.
    - Proficiency/comfort using CLI tools (command line/terminal/console)
    - Virtual machines
  - Comfort/proficiency using revision control systems, particularly Git/GitHub
  - Ability to ask questions effectively in online forums such as Stack Overflow
  - Ability to identify and select among resources useful to learning how to write code, such as online documentation, tutorials, forums
  - A proactive attitude toward seeking outside help when necessary without undue procrastination
  - A collaborative and supportive attitude towards making oneself a valuable member of the community of developers
  - Choosing the “right” technological foundation for a software project

## Choosing the “Right” Technology

Software can be developed using a dizzyingly vast array of different programming languages, platforms, frameworks, technologies, and techniques. Not only is there an overwhelming array of technological choices, but also these options are constantly evolving, changing, merging, diverging and becoming more refined. While there absolutely are trends as to which technologies are more popular, more stable, and better supported, that does not mean that there will ever be only one “right” choice when it comes to implementation. However, that does not mean that which technology you choose is not a very important decision.

The choice of which technologies to use can have a critical impact on the success of a project. You will be asked to develop your ability to make this kind of decision. This will involve:

- Doing research into the available options to solve a specific problem
- Being able to evaluate the strengths and weaknesses of a particular option
- Not getting bogged down in “analysis paralysis” and making a timely decision
- Being able to defend your choices
- Embracing your choices which will be demonstrated by quickly teaching yourself how to use them and creating artifacts based upon them

In other words, your instructor will not tell you to use JavaScript over Python or PHP, or to use MySQL over SQLite or MongoDB. Any of these choices will likely be a satisfactory tool to complete your project. Your instructor will expect you to be able to explain and defend your decisions, and to take the initiative to move forward on your project once you’ve made a choice.

## Learning Strategy

Focus on building a team to get a project done. I would say that everyone should be able to accomplish all of the less technical project steps: problem description, social context analysis, mockups, generic architecture, data model. After that, achieving technical depth in all of these areas would be difficult, and it would probably be smarter to focus on one or two areas. Join a team of people working on complementary areas on which each person can be the “expert” over one or more of the tasks that need to be completed.

## The Semester Project

The semester project should be a website, mobile application, or a self-contained component of such an application that will require significant effort and require you to integrate the skills and knowledge gained from the rest of the semester.

- Doing a semester project is NOT optional.
- I strongly recommend that you have teams of no more than three people.
- I also urge you to start ASAP and use the project to guide your choice of other activities throughout the semester.

Here are the deliverables:

- Step 1: Develop the Problem Statement
- Step 2: Develop a System Analysis that includes:
  - Social Context Analysis
  - Scenarios
  - Complete screen mockups (paper and pencil)

- A high-level description of the project architecture
- A functional specification
- A unit test specification
- Step 3: Develop a project skeleton that includes:
  - Development, staging, and test environments
  - The minimal set of files necessary for testing
  - A complete set of failing tests
- Step 4: Develop the project, i.e. write the code
  - Demonstrate passing tests
- Step 5: Deploy the project to a live environment (web server, mobile device)
- Step 6: Present your project to the class and other interested parties

Here's a rough suggestion of the schedule you should follow ("Week" means by the end of that week):

Step	Week
1	2
2	4
3	6
4	13
5	14
6	15

I can't stress enough that you don't procrastinate on this. Software almost ALWAYS takes longer than you expect. It's just part of the process.

## Grades

I have a unique approach to grading called "choose your own grade" that means basically this:

***Almost nothing*** is required in this course. You don't have to show up, crack a book, take a test, or turn in any homework. The only requirement is that you come meet with me face-to-face at the end of the semester and tell me what grade you would like me to report to the registrar.

That being said, you spend A LOT of money to be here at JMU. Learning should be fun. Learning should be risky. Learning should challenge you and make you take a hard look at yourself, and at the world. You should be doing it because just being here means you are ***extraordinarily privileged***. I want you to work hard because you mean it. Take it seriously. Stretch yourself.

## Why?

Although I think it deserves at least a book-length explication, I'll try to explain briefly the philosophy behind the choose-your-own-grade approach. I'll include links to relevant research and writings that you can follow-up on if you're curious.

I've made a truly ISAT-ty study of grades—looking at their history, psychometric and statistical properties, political, economic, cultural and sociological expression, and their cognitive, motivational, and affective impacts. As it turns out, none of these perspectives provides evidence or analysis that supports using grades to evaluate student performance. Except for one, and I'll start with that.

## The argument FOR using grades

There's only one, and it's simple: it's an efficient way to provide feedback to a large number of students (and other stakeholders—parents, legislatures, colleges, employers) in a short period of time. While early proponents of this argument in the 19th century, when use of grades really became prevalent, were aware of grades' flaws and shortcomings, they wrung their hands and argued (probably justifiably) that there really was no other way to do it given the student-teacher ratios and the need to make a case for the value of schools and schooling at all. Keep in mind that the US was still predominantly agrarian at this point in history, and many people did not see a need for anyone to acquire more than basic reading skills, if that. It is also important to note that this is the time of the Industrial Revolution, and the standardized, one-size-fits all, assembly line approach to evaluation seemed extraordinarily appropriate given the times.

In short, the primary argument in favor of using grades is economic. I mean, think of the alternative to letter and number grades. Do we really expect teachers to write full essays describing each student's strengths and weaknesses in detail? Who has such time? How could we afford to pay the number of people it would take to accomplish such a task?

As such, it should be no surprise that the loudest debates about how we grade (not if we grade—that's taken for granted) tend to be couched in economic terms, e.g. grade "inflation." (BTW, [grade inflation is a myth](#).) People also try to correlate grades and GPA to likelihood to get a (good) job, lifetime income potential, etc. All of them are primarily economic arguments and do not deal in a truly substantive way with the negative impacts that grades have had on nearly everyone in our society.

## The arguments against using grades

I began studying grades as a way to devise the "perfect" grading strategy and use the absolutely best practices in my classes. My studies have led me to abandon the notion of using

grades almost altogether. Here are what I believe to be some of the more compelling arguments.

## The Economic Arguments

Since we ended the last section with economic arguments for grades, let's look at some economic arguments against (or at least indifferent to) grading. I'll just give some basic ones, starting with one of the basics of economics, the law of supply and demand.

To restate the law, when supply is high and demand is low, prices fall. When supply is low and demand is high, prices rise. In the case of grades, the conventional argument would be that A's are (or should be) a scarce commodity, and therefore the price of acquiring one, i.e. student talent and/or effort should be high. In the case of grades, this is a bogus argument.

First of all, A's are not a scarce commodity. From the instructor's standpoint, they cost nothing to produce and there is a limitless supply. (If they were, lot of techers would be more creful bout wht they typed!) Not only can I assign as many A's as I choose, but I'm not charged \$50 for every A that I assign, \$25 for B's, \$10 for C's etc. If anything, it's what economists would refer to as an artificial scarcity, which is when a producer intentionally destroys, withholds or declines to produce goods in order to keep prices high. I guess it's not unheard of for instructors to intentionally limit the number of "available" A's in order to scare students in to working harder, but I'm not sure that a scare tactic is really a valid argument for using grades.

This begs the question of what is the value of an A? I'm sure someone has tried to quantify it in monetary terms, but a short thought experiment will show how ludicrous this is. Is an A in organic chemistry worth the same in 2014 dollars as an A in scuba diving? How about a 2.9 in the College of Business as opposed to a 3.8 in English? The artificiality of trying to make such value comparisons quickly devolves into the absurd.

Video: <http://bit.ly/pQSF4H>

But doesn't that point out the value of GPA for getting good jobs? At least [one study done at VA Tech](#) found that about 42% of the time, college recruiters did not even appear to consider GPA at all in deciding whether or not to award an on-campus interview. In 17% of the cases, it appeared that a higher GPA decreased students' chances of getting the interview. Of course, this still means that in over 40% of the cases, GPA was at least somewhat considered. Some companies do have GPA cutoffs for new hires. (After reading this, though, you may not want to work for those companies.) But it's important to keep in mind that your GPA will only possibly be important for your very first job upon graduating. After that, no employers will ask for it. They'll make their decision to hire you or not based on your previous job performance (possibly). So it's important to keep in mind that your GPA may not be as important in getting a job as you've been led to believe.

There is one important exception to this: graduate school. Graduate schools absolutely and nearly universally use GPA in their admissions criteria. So if you plan to go to grad school, an economist and I would agree in advising you to strive to get as high a GPA as possible. An unethical person would add “by any means necessary.” This overvaluing of grades is actually a primary source of academic dishonesty and friction between teachers and students, but we’ll get to that in the section on motivation.

## The Statistical Argument

Most grades are calculated as a sum or average of some smaller set of scores. Two relevant questions to ask are:

1. What is the source of those sums or averages?
2. Are sound statistical practices being used in the interpretation of those averages?

Let’s actually start with the second question first. Statistics is a field of analysis which seeks to use the properties of numbers and mathematics to make justifiable and accurate descriptions and predictions about phenomena in the world. We typically think of the Mediterranean as a sunny and warm place. However if you were thinking of escaping a cold Virginia winter, you may be surprised to learn that the [average January temperature in Malta](#) is only 50-60°F. Here we’ve used the arithmetic mean, i.e. the average, a statistical analytical procedure, to interpret weather phenomena. Your likely conclusion in this case is that if you do plan to go to Malta in January, the interpretation of the statistic is pretty clear: you’d better take a jacket.

Let’s try the same interpretive exercise with grades. Let’s say that a student, Otis, got an 87 on an assignment. Here are some plausible interpretations. In each, pay close attention to how you feel towards Otis.

- Otis, normally an A student, was feeling sick and didn’t perform so well
- Otis, normally a C student, really studied hard and outperformed his usual scores
- Since the class average was 96.42, Otis fell well below his peers’ performance
- Since the class average was 42.96, Otis did extremely well on this assignment
- Otis’ paper was on top of the stack when the teacher began grading so evaluated it more strictly than later papers
- Otis’ paper was on the bottom of the stack, and the teacher, being tired by this point, graded leniently
- The teacher had a mistake on the key used to score the assignment
- Otis has undiagnosed test anxiety disorder, which means he routinely scores below his potential
- This particular teacher really likes/dislikes Otis
- This is a particularly rigorous/unchallenging course, school, etc.
- etc., etc., etc.

I'm sure that many more different scenarios could be developed. Regardless, based on the number alone, without understanding the context it is nearly impossible to interpret. "But wait!" you say, "Wouldn't such difficulties in interpretation be alleviated if we had an average of all of Otis' scores for a whole semester?" Answer the question for yourself by imagining that instead of one assignment, Otis got an 87 for the semester, or a 2.96 GPA for his entire time in college. Can you not come up with an equally large number of plausible interpretations of such scores? The problem lies in a misapplication of statistical procedure. Misapplication of statistics results in the inability to make a reliably meaningful interpretation.

What is different about the average temperature example and the average grade example is the source of the underlying data used to calculate the statistic. Average temperature is based on quantitative data whereas grades are based on qualitative data. "But," you ask, "isn't an 87, or a 2.96, a number? Don't numbers represent quantity by their very nature?" Nope. In the case of grades, an 87 is higher than an 86 and lower than an 88, but these are just indicators of relative quality.

I'll illustrate this more clearly with another example. Take the scale below that might be commonly be seen on a customer satisfaction survey in response to a question, such as, "How satisfied were you with your server, today?"

extremely dissatisfied	dissatisfied	neither satisfied nor dissatisfied	satisfied	extremely satisfied
---------------------------	--------------	---------------------------------------	-----------	------------------------

Now it's clear that "satisfied" is a higher ranking than "dissatisfied," and while it's clearly possible (and actually common) to assign each of these rankings a number, say from 0 to 4, does a ranking of 4 (extremely satisfied) really mean that a customer is "twice as satisfied" as one that gave the server a 2 (neither satisfied nor dissatisfied). Is a customer who gave the server a 3 three times as satisfied as a customer who gave the same server a 1? I don't think so. And even if this were the case, how would we make a meaningful interpretation of these values? Can you not come up with a nearly endless list of reasons as to why a particular server/customer interaction might have yielded those scores? Getting back to grades, is a score of 80 twice as good as a 40? Is 100 twice as good as 50? 60 twice as good as 30? A 4.0 twice as good as 2.0? Did the person who scored the higher score learn twice as much? Not likely. And it's because grades don't represent quantity but quality.

We could dive more deeply into the source of the 87, but hopefully the point is made. Consult a textbook on statistical analysis and it will tell you that the arithmetic mean is not applicable to analysis of qualitative data (except in very specific circumstances which are not really relevant and beyond the scope of this essay). Grades are clearly qualitative, not quantitative, and therefore should not be analyzed with averages lest they be susceptible to misinterpretation.



Incidentally, I find it painfully ironic that nearly all statistics class grades are determined using averages. You should note that in ISAT, Dr. Radziwill does NOT do this. She uses a points accumulation system, and in fact, the points accumulation system in this class is patterned after hers.

In case you're interested, there's a [really fascinating 2000 paper by Vickers](#) that shows that because of the inconsistencies in GPA calculations across different institutions, it is logically impossible to rank students using GPA. But let's move on.

## The Motivational Argument

On the surface, grades would appear to motivate people to study harder, and as such, be a good thing. Grades' power to motivate students is undeniable and real. However, the problem lies not in whether grades motivate or not, but how.

Alfie Kohn, in his book [Punished By Rewards](#), articulates five ways in which grades are harmful (paraphrased):

1. Grades punish
2. Grades ignore reasons
3. Grades rupture relationships
4. Grades discourage risk-taking
5. Grades undermine motivation

Grades punish because in order to have "winners" you have to have "losers." Teachers create tests and assignments intentionally at a level where they know that some portion of their students will be unsuccessful. Let me come to teachers' defense and argue that the reason they create tests that they know some students will fail is not because they are sadistic or enjoy seeing some people fail. Rather, they have to teach a lot of students with varying levels of ability, and different life circumstances. They pitch to the middle intentionally so that hopefully the higher performing students won't be completely bored, and the lower performing students won't be hopelessly lost. Teachers are trapped in an industrial revolution mentality educational system just like everyone else. Regardless of the reason, the students forced to endure these tests can't help but feel in many cases that they are being punished.

Grades ignore reasons. For example, many teachers have an attendance policy. If a student misses a class, or many classes, the reason becomes unimportant ("I've got too many students to review all of the individual reasons and excuses why you can't bother to show up!"). There's a policy in place. It was communicated clearly on the syllabus. If you don't show up and don't have a doctor's note, that's not my problem. How does this make students feel? It can't feel good to be treated as just another name on a roster. The grading policy in this case, allows us to shield ourselves from the effort it might otherwise require us to track a student down and find

out if they are okay. Grades enable teachers to ignore the idiosyncrasies and individuality of the people in our care. I know. I've been one of those teachers.

Grades rupture relationships. This one is mostly about power. It is very difficult for two people to have a genuine friendship when one has power over the other one. In the classroom, I have sole power to decide your grade. The US Supreme Court has actually ruled that a professor can be fired for disobeying the university if he or she refuses to change a grade. But even after being fired, the university can't change the grade. In the classroom, to the extent that you care about grades, I am your god. I can make you do anything I want to. While the vast majority of professors wield this power responsibly, tales of sexual or monetary exploitation are not unheard of, and even in the best of circumstances it's still difficult to be "friends" with your professor. How can you say what you really think to a person who might dock your grade for it? The power dynamics are the primary source of cheating and other academic dishonesty. Take away the grade and there's really no reason for teachers and students to lie to one another.

Grades discourage risk-taking. Have you ever heard anyone ask "Will this be on the test?" Have you ever hounded an instructor to find out exactly how many pages, what font size, how many inch margins the paper has to be to not get docked points? Have you ever failed to speak your mind for fear of angering the professor? Have you heard a teammate argue that your team should do take the easy and safe option for the group project? "Let's just get it done and out of the way." On the contrary, have you ever wanted to take on a more ambitious project, but not done it for fear you'll fail? Our grading system is inadvertently socializing our young people to be docile, compliant followers of orders. It's ironic given that many, if not most, of our society's heroes are people who took risks, faced big challenges and succeeded. We don't often acknowledge how many times they had to fail to get there. Classrooms should be the safe place to fail. The classroom is the environment where the impacts of failure can be contained. We should be encouraging students to swing for the fence, to shoot for the moon, to go big, and if you're going to fail, fail hard and know that we're here to catch you. Not gonna happen though. You might get a bad grade.

Video: [https://www.ted.com/talks/ken\\_robinson\\_says\\_schools\\_kill\\_creativity](https://www.ted.com/talks/ken_robinson_says_schools_kill_creativity)

Grades undermine motivation. This is the fifth and most insidious of the motivational arguments against grades. The basic idea here is that because we emphasize grades so much, learning becomes a secondary or nonexistent concern. How many tests have you crammed for, only to forget everything the next day? If you really cared about learning, wouldn't you have studied in a way that would more likely promote long-term retention? How many times have you finished a particularly grueling class and walked out of the final thinking, "I'm so glad I'll never have to study that again!" If the method of our teaching makes students LESS excited about learning, then we, as teachers, are failing utterly. As it turns out, this is exactly what we're doing. I'll let you read the paper for yourself, but a meta-analysis of hundreds of studies showed that even when they get the maximum reward, i.e. an A, student motivation to remain engaged with the content decreases! For students who got less than an A, the effect on motivation was

devastating. As a teacher, why would I want to engage in a practice that I know will actually make students LESS excited about what I teach???

## Other Arguments

I'm far from exhausting the arguments against grading—like I said, I could write a book—but I think you get the point. I'll just briefly touch a couple more. For example, work in positive psychology shows that the stress induced by grades actually interferes with your cognitive abilities and makes you less able to grasp concepts globally and to see patterns and connections between related ideas. From a historical perspective, there were great debates about grades in the 19th century which just kind of died out as people gave up and decided that grades were a just a necessary evil. Now it's hard to believe there was a time when they didn't exist. Studying the field of psychometrics will destroy your hope that anyone could ever construct a valid multiple choice question. There's a lot more that could be said about everything that I've said so far, and more that I haven't even covered yet.

## Enter Choose-Your-Own-Grade

So this is why I developed the choose-your-own-grade approach:

- Grades have no meaning to me, and I have no clue how to assign one (and feel good about it).
- I don't want to create a reason for students to be dishonest with me, especially when I'm in control of the environment, and I don't have to set it up that way.
- I don't want to create an environment that hampers creativity and risk-taking.
- I don't want to make anyone feel inferior just because programming doesn't happen to be one of their gifts, especially since most people in my class don't really have much choice about whether to be there or not.
- I'm not your boss. You're not my employees. (If anything, your tuition pays my salary, so you're my boss.) If you trust that I have your best interests at heart, and are willing to follow my guidance, I'm humbled and grateful.
- I don't want artificial elements like grades to stand in the way of you falling in love with something new.

I like what Ben Zander has to say:

Video: <https://youtu.be/qTKEBygQic0>

Here's what I do want:

- To make friends and touch lives.
- To share the things that I'm passionate about.
- To create a safe, lively, compassionate space for people to try new things.

- To give people a chance to fail spectacularly and exclaim, “That’s fantastic!”

And the subversive part of me wants you to have that experience in my class, and for it to change the way you see EVERY class, and for you to go out and challenge others to really grab hold of the opportunity you have here at JMU and make the most of it.

Don’t take yourself so seriously. I don’t. Life is a lot of fun that way.

## The One Rule

There is only one requirement with choose-your-own-grade. You have to come meet with me face-to-face at the end of the semester and tell me what grade you’d like me to report to the registrar. Maybe some of you will take Ben Zander’s advice and sit down right now and write me the letter: “Morgan, I earned my A because....”

## Schedule & Flow

If you are ever sitting around wondering what you should be doing at any point in time, please follow these steps, in order, until your confusion abates:

1. Read the What-You-Should-Be-Doing-Now blog post for this week
2. Ask a new friend in the class
3. Ask your instructor

Don’t sit around! Suck the marrow out of this class!!!

From your instructor’s viewpoint, class time belongs to the class, and we should make decisions together about how it should be spent. In general the use of class time will be fluid. Any given day may be used in the following ways:

- Teams or individuals working on labs independently
- Mini or full lectures delivered by anyone in the class
- Code review sessions, where the class offers feedback on other teams’ code
- Status updates or reports by teams or individuals

The schedule for the course will roughly follow the timeline of the semester project.

Week	What the Class is Doing	What You should be Doing
------	-------------------------	--------------------------

1	Forming Teams; <a href="#">Meeting People</a> ; <a href="#">Studying the Syllabus</a>	Dreaming about projects; Setting semester goals
2	Presenting Software Ideas; Shuffling teams; Assigning roles	Picking your role; Writing Problem Statement/Social Context Analysis/Scenarios
3	Social Context Analysis; Writing Scenarios; Drawing Mockups	Beefing up basic skills; Learning programming/database skills
4	Deciding architecture; Functional specs; Test specs	Continue skills development; Figuring out your niche on your team
5	Building out app environments; Installing core files	Continued skills development e.g. PHP/JS/DB/etc.
6	Creating failing tests	Getting familiar with your MVC framework
7-11	Coding testing and documenting the project	Doing labs to serve the team coding process
12	Finalizing code; Getting all tests passing	Wrapping up project coding
13	Deploying the project	Working on project deployment
14	Practice Presentations	Working on the Presentation
15	Presenting the project	Working on the Presentation

## Important Dates & Deadlines

At the behest of the registrar, a list of dates you may wish to take note of:

- Tuesday, September 6th: Last day of add/drop
- Thursday, September 15th: Last day to add a class with Department Head signature
- Friday, September 16th: Last day to withdraw from JMU with charges canceled

So if I scare you off, get out early. Or conversely, if I turn you on, join us soon!

My academic integrity policy is different from JMU's standard policy, but I will adhere to JMU's standard policies listed on the [JMU Syllabus Information for Students page](#) with respect to add/drop, disability accommodations, inclement weather and religious accommodations.

# Personal Integrity

First:

**If I catch you cheating, or doing anything else dishonest, you will fail the course. Period.**

Second, that being said, I strongly encourage sharing and collaboration in most every aspect of the course. That means that I think it's a smart idea for you to:

- Download code you find on the web (include the URL of where you found it and some notes about how you got there)
- Download your classmates' code and use it, even before an assignment is due
- Pay someone to help you write code
- Get code from upperclassmen or people in previous semesters
- Ask your neighbor to give you a hint on a question on a test that you're stumped on
- Use whatever notes, websites, books, or other materials you need to complete most any assignment or test

You'll note that many of the above behaviors would be considered "cheating" in many or most other courses. Here are some guidelines I'd like you to follow:

- **Never EVER copy without attribution**  
Even on tests, if someone or something helped you out, acknowledge it. Make notes in your code if you got it from someone or somewhere else. Copying without attribution is stealing and is a breach of integrity. If you got the code off of the web, there should be a URL and some notes about how you found it. If you paid someone to help you write it, say so.
- **Never copy without understanding**  
The point of the class is to learn and understand stuff. Since you don't get any grades on individual tests or assignments, it's pretty stupid to copy something that you don't understand. Think about it. What point could it possibly serve?
- **Be very hesitant to copy an ENTIRE project**  
While there's a lot to be gained by incorporating parts of your classmates' code in your own project, copying someone else's entire project doesn't really provide you much of a learning experience and wastes people's time.
- **Try to figure it out yourself first**  
90% of writing programs is learning how to write them, and this will stay the same throughout your entire programming career. Being a self-sufficient learner is one of the primary goals of the course.

*Code re-use is a HUGE part of hacker culture. What hackers hate more than anything is not understanding stuff. I want you to get a sense for what it's like to be a part of the fun world of professional hackers.*

Okay, so what do I consider a breach of integrity worthy of failure?

- Lying about anything to anyone in the class
- It could be as trivial as the reason why you didn't show up for class or do your part of a group assignment. Everybody screws up sometimes. Don't compound the mistake by lying about it. We can forgive mistakes but it's VERY difficult to regain trust once it's broken. Swallow your embarrassment or fear and fess up.
- Stealing anything—this includes copying without attribution
- Stealing is just wrong, and since you have a blanket license to copy most any code you can find, there's no reason not to give people credit for the work they did. Passing someone else's original work off as your own is frankly disgusting.
- Threatening, antagonizing, or intimidating anyone in our learning community
- This is unacceptable behavior and will get you at least fired, if not sued in most every company you'd ever work for.

If you are in doubt about something, please ask your prof. Please feel free to come speak to your prof in confidence about anything in this course that troubles you. So far at JMU I've never had a problem with anyone's integrity (that I know about). Don't be the first group to ruin my perfect record. Thanks!

## The Prof

My research mainly involves coming up with pedagogical alternatives that maximize student motivation and learning. Being a tech geek, web-based technology plays a pretty heavy role in what I came up with.

My favorite part of my job is getting to hang out with students and play with technology. Feel free to come see me any time. My info:

Office	ISAT/CS 124
Office Hours	<a href="#">Make an appointment</a>



<b>Mobile</b>	973-495-7736 (calls and texts are ok within reason)
<b>Email</b>	bentonmc@jmu.edu
<b>Facebook</b>	<a href="http://www.facebook.com/morgan.benton">http://www.facebook.com/morgan.benton</a>
<b>Twitter</b>	<a href="http://twitter.com/morphatic">http://twitter.com/morphatic</a>
<b>Blog</b>	<a href="http://www.burningmindproject.org">http://www.burningmindproject.org</a>