ITM 695
Application Development for Managers
Spring 2019 Syllabus

Class Time:	MTWRF– 5:00 – 10:00 PM
Location:	Computer Center, Jaeb, 106
Prerequisites:	Basic understanding of math and statistics (i.e. mean, median)
Professor:	**Dr. Tim Smith**
Office:	KBB 203
Office Hours:	MTWThF 4:00 – 5:00 PM
	Or by appointment
E-mail:	tcsmith@ut.edu
Contacting Me:	The best way to contact me is via email. If you send me an email before 4 PM, you should expect a response the same day.
Required Text:	There is no required textbook for the course
Required Software:	Python, Git, Visual Studio Code (NOTE: This course requires the use of your own personal laptop. All software is freely available without charge.)
Home Page:	GitHub (https://github.com/itm695x-m19) and Blackboard: http://ut.blackboard.com. All grades, and quizzes will be made available through Blackboard. Syllabus, and class materials will be made available on the classes GitHub page.

Course Description:
The course focuses on a managerial view of the software development and programming process with a focus on data processing, analytics and secure programming. In this course you will be exposed to a detailed study of the Python programming language and associated tools. Topics include Bash shell, working with an IDE, the role of programming languages, Python, writing secure code that is resistant to attacks, version control systems, command line interfaces, common constructs found in every modern programming language, data structures, and predictive modeling.

An interest in programming, willingness to put in time and effort to learn exciting and new options, and ability to work as a productive team member are essential skills for success in this course.

Specific Educational Objectives of the Course:
Upon completion of the course students will be able to:
1. Understand common programming constructs found in any modern computer programing lanugage.
2. Demonstrate fundamental programming skills concerning:
    i. variable assignment and scope
    ii. control structures, loops, and branches
    iii. methods and class interfaces
    iv. arithmetic and Boolean expressions
3. Demonstrate basic data import, processing, and modeling.
4. Understand the role API's and the emerging phenomena of social coding.

5. Introduce the application of Static Application Security Testing (SAST) Tools for secure code review and application gating (aka "secure" programming).

Academic Expectations: The course will challenge you to expand your critical thinking and analytical problem-solving skills in ways that will be beneficial throughout your career. Though you are not expected to have prior programming experience, aptitude in math/statistics and general computing knowledge is recommended.

Computer Resources: You will use your own laptop to perform work in this class. All software used in the course is freely available, and you will be required to install the necessary software on your personal computer. A portion of the first class will be allocated to assist you with software installation.

Classes: We will meet five times a week for two weeks from 5:30 pm to 10:50 pm. The last hour and a half of each class will be dedicated to practice problems and team challenges.

Class Requirements

Grade determination:

Component	Points	Description
Individual Assignments	260	Calculated from average mark from 8 assignments
Tests	450	2 worth 225 pts each
Class participation	90	Attendance, and positive contribution to class
Journal	100	Journal entry for each class (10 classes)
Repo Management	100	Overall neatness, good structure, and management of student individual repo, and contribution to one of the class repos (pull request). Included in this evaluation will be evidence that in-class exercises have been completed.
TOTAL POSSIBLE POINTS	**1000**	

Grading Scale:

92 -100%	A
88 – 91%	AB
82 – 87%	B
78 – 81%	BC
72 – 77%	C
68 – 71%	CD
60- 67%	D
Below 60%	F

Academic Integrity: The University of Tampa is committed to the development of each student to become a productive citizen who embraces the values of honesty, trust, fairness, respect, and responsibility. The scholarly community at The University of Tampa strives to instill values that uphold academic integrity and promotes an ethical standard that does not condone academic misconduct. Violation of academic integrity and academic misconduct tarnish the reputation of the University and discredit the accomplishments of past and present students. Sanctions for violation of academic integrity and academic misconduct include a failing grade in an assignment or the course or suspension or expulsion from the University. I take integrity very seriously, including academic integrity. I will monitor all submissions and exams for violations of the academic integrity policy. Students are held responsible for knowing and observing the University's Academic Integrity Policy posted at http://www.ut.edu/provost. If you have any questions about the policy, please feel free to talk with me.

Unless otherwise instructed, you must work alone on exams and other related coursework. **Any form of cheating may result in a grade of "F" in this course and an Academic Integrity violation.** Communication during exams

with anyone other than the professor (including via electronic means such as cell phone, email, chat, etc.) will be considered an academic integrity violation. **Keep in mind that helping another student who is supposed to be working alone is also cheating – it does not matter whether you are giving or receiving the information.** So if someone asks you to help them, they are asking you to cheat and endangering your future – real friends don't do this to each other.

Classroom Conduct: Students are expected to behave courteously and professionally in the classroom. This means arriving for class early (arriving right as class starts means you are late), contributing to the class in a positive and constructive manner, staying until the class is over, minimizing restroom breaks, side-bar conversations, etc. You also need to come to class having read the required readings for that class session. I do not think it is valuable to recite the textbook material – that is the purpose of the textbook. My goal is to expand upon the foundation provided in the book and have us practice computer programming in a fun and inclusive atmosphere. I am planning to have a lot of in-class activities and expect that the classroom atmosphere will be at times noisy and slightly chaotic. However, please refrain from multitasking, meaning emailing, IM, web surfing (including Facebook, Twitter, etc.), working on assignments for other classes, and other activities not directly related to class. Students who disrupt class for any reason will be warned once and then referred to the Dean of Students for possible disciplinary action. If you have a health condition or some other issue that requires possible deviations from the above, please let me know so we can come up with a plan that meets your needs as well as the needs of your classmates.

Class Attendance: There is a strong relationship between attendance, advanced reading, home practice, and the final course grade. You can learn more by being present in class, listening to the explanation of the material, following along with in-class demonstrations, and asking questions than if you are absent. Students are expected to participate in class activities, volunteer for in-class demonstrations, and fully engage in every class session.

Individual Assignments: In each class you will have an associated assignment. These are to be submitted by midnight on the day of the class.

Tests: Two tests each worth 225 points each. These will be take-home tests that given on Friday and are due by the following Monday before 4:00pm.

Final Exam: There is no final exam for this course.

Journal: Write a journal documenting your learning progress through the course. This will be written in markdown and posted to your private repos. Journal entries must be made after each class and before the next class.

Repository Management: Each student will be assigned their own private GitHub repository. The professor will review the student's management of their repository at the end of the semester. Students must demonstrate good repository activity and management. Students will also be expected to conduct at least one pull request to add and or edit one of the classroom repositories.

Class Participation: This mark will be assessed by the professor based on your positive contributions to classroom discussion and work. Positive contributions may include being called upon to answer questions, present solutions to the class, participate in classroom discussion, and present an overall positive and helpful demeanor.

Grading Appeals: If you have questions about your grade for a aspect of the course stop by my office or e-mail me. **I will not discuss grades during class time**. You can appeal your grade within 48 hours from the time the grade has been assigned. After that period the grade becomes final.

Incomplete Grades: Incomplete grades will be assigned only in the event of an extraordinary circumstance that prevents you from completing the class this semester. Students must have completed the first two tests with a passing grade before being considered for an incomplete. All incomplete grades must be completed by the date specified on the incomplete form submitted to the Registrar.

Special Needs: If there is a student who has special needs because of any disability, please go to the Academic Success Center in North Walker Hall for information regarding registering as a student with a disability. You may

also call (813) 257-5757 or email disability.services@ut.edu. Please feel free to discuss this issue with me, in private, if you need more information.

Student's Professional Responsibilities:

1. Arrive on time to class. Entering the room late disturbs everyone's concentration. If you do arrive late, do not interrupt the class.
2. Side conversations and ringing cell phones during class make it difficult for your classmates to actively listen and learn. If you must have a cell phone with you, be sure it is in a silent mode, and only respond to true emergencies _after_ you have left the classroom.
3. It is your responsibility to monitor Blackboard and your UT e-mail account for class communications.
4. If you are having problems with the course materials, please let your professor know as soon as possible while there is still time for him to help you!

Session	Date	Topic(s)	Quizzes and Tests
Class01	Monday 05/13	Course Introduction Introduction to programing and computer architecture Introduction to the system command line/Bash shell Introduction to Git and Github Installation of software Creation of GitHub accounts	Assignment #1
Class02	Tuesday 05/14	Installation of Python and VSCode Introduction to Python programming Introduction to VSCode Introduction to Markdown Writing your first program More on the version control system Git and GitHub	Assignment #2
Class03	Wednesday 05/15	Writing 'code' using an IDE – debugging and running your programs Variables Expressions Statements	Assignment #3
Class04	Thursday 05/16	Structured programming (sequence, iteration, selection) Conditionals Functions	Assignment #4
Class05	Friday 05/17	Tuples, Lists, and Dictionaries List comprehensions	Take home Test#1
Class06	Monday 05/20	Strings and String Formatting Reading and writing files Searching files The os library (searching folders/directories)	Assignment #5
Class07	Tuesday 05/21	Organizing and documenting code (Modules & Docstrings) Downloading custom Modules/Libraries (PiPy, pip, conda, and pycharm) Exception handling	Assignment #6
Class08	Wednesday 05/22	Using WebAPI's Introduction to Jupyter Notebooks	Assignment #7
Class09	Thursday 05/23	PANDAS (data cleaning/munging) Predictive Modeling using SciPy	Assignment #8
Clas10	Friday 05/24	Secure/Robust Programming Exception Handling Unit testing/Test Driven Design Static Analysis	Take home Test#2

ITM 695 Course Schedule
(This is a proposed outline of course topics. Schedule variations will be announced in class and posted to Blackboard)