CS4365/6365 Introduction to Enterprise Computing

Summer 2026, 4 credits

Instructor Information

Instructor	**Email**
Calton Pu	calton.pu@cc.gatech.edu

IMPORTANT: Please include [CS4365] or [CS6365] in the subject of course-related email. Professors are deluged with email. If we don't respond, don't take it personally. Just resend your email with a friendly reminder.

General Course Information

Course Description

All enterprises depend on information technology (IT) to conduct business and make profits. Successful enterprise IT platforms and applications provide high levels of quality of service (QoS), including performance, reliability, availability, and security, as well as scalability as enterprises grow. Modern enterprise IT achieves scalable QoS through a combination of foundational technologies such as databases, workflow, services and cloud computing. , as well as recent breakthroughs such as generative artificial intelligence (GenAI).

The course lectures will introduce these fundamental conceptual pillars of enterprise IT, complemented by an overview of techniques and platforms that support a scalable implementation of the QoS dimensions in the cloud. The practical overview is augmented by intuitive explanations of the theoretical foundations that guarantee the correctness (e.g., Principle of Conservation of Money) of these techniques despite system failures, as well as high scalability requirements.

In addition to enterprise IT technical concepts and techniques, students will apply these concepts and practice technical and general skills in a semester-long project.

Course Learning Outcomes

The course has two major learning objectives: (1) technical information on enterprise IT systems and (2) general skills that enable *autonomous learning* beyond the semester. The technical information emphasizes the foundations of information technology (IT) that enable and support

enterprise applications, including transaction processing, workflow and business process management, service computing, cloud computing, and AI/ML. The skills for autonomous learning include: fact vs. non-fact, science skills (e.g., applying theory in practice and checking assumptions), and recognition of trend patterns. The skills are illustrated through the explanation of technical topics and practiced in a semester-long project.

On technical information: Enterprise computing systems are distinguished from typical IT applications by their QoS (quality of service) guarantees of correctness, performance, and reliability properties. For example, web-facing mission-critical applications such as e-commerce often aim for near-zero latency and five-nines availability. Another example is the Principle of Conservation of Money through theoretical guarantees such as serializability of transactions. Useful techniques surveyed include: control systems for adaptation, queuing systems for performance, and optimization algorithms used in many applications including ML classifier training. The lectures are divided into modules:

- Introduction: course format, content, skills for autonomous learning.
- ML/AI: situational awareness, big data, cyber-physical gap, LLM and new AI tools.
- Transactions: concurrency control and crash recovery, ACID vs. BASE properties.
- Quality of Service: control and adaptation, optimization, performance and scalability of large-scale systems in cloud computing and edge computing.
- Services: workflow systems, service computing, microservices.
- Project: information and guidance on student-proposed or structured projects.

On skill learning: enterprise IT systems are characterized by fast technological evolution, with the GenAI as the most recent example. Students can benefit from acquiring skills for autonomous learning, so they can continue to grow (technically and professionally) on their own. Skills needed for autonomous learning include general skills such as fact vs. non-fact, and fact-based self-evaluation, as well as science skills of applying theory to practice and checking assumptions. Advanced skills include the recognition of trend patterns such as: application pull, technology push, software specialization/generalization cycles and management centralization/decentralization cycles.

Required Course Materials

Due to the rapid evolution of enterprise IT technologies, instead of textbooks we will adopt reading assignments, selected from classic and state-of-the-art research, survey, or review papers. They will be posted on Canvas. As appropriate, sample project checkpoint reports will be posted, in addition to the reproducibility exercise.

Grading Policy:

Due to continuous technological evolution in enterprise IT, e.g., the recent introduction of GenAI tools, the grading components and scaling have also evolved over time. The main graded components consist of: attendance, reading/writing exercises, and project-related activities. Currently, 10% is allocated to attendance, with 90% considered full attendance. Reading/writing exercises occupy 15%, and project-related activities the remaining 75%, divided among a reproducibility exercise, an initial project prototype, several checkpoint reports, and final deliverables. Project component grading consists of realistic feedback after the submission, with potential grade increases if improvements are demonstrated through the subsequent checkpoint reports.

Description of Graded Components

Reading/writing commentary assignments: Reading assignments (before each lecture) will be accompanied by written commentary assignments with components that include GenAI tool assistance.

Reproducibility exercise: Reproduce software deliverables from previous projects to practice the skills of project execution, evaluation and documentation with supporting evidence.

Intermediate project checkpoint reports: At regular intervals (currently every 3 weeks), intermediate project checkpoint reports will provide an update on the project technical progress and concurrent skills learning. A report consists of three main technical components and a self-evaluation. The technical components are: (1) Adaptive Project Plan with starting point A, ending point B, and the steps from A to B; (2) execution status report of the Project Plan; (3) factual supporting evidence of execution status (github). The self-evaluation consists of an estimation of #Scope(plan), #Match(execution), and #Factual(supporting evidence).

Project proposal: The project checkpoint report #1 should contain the initial project plan, previously called “project proposal”. We expect the project plan to evolve during execution, with updates described in the sequence of checkpoint reports.

Final deliverables: The last checkpoint report with the same technical components, plus an overall project self-evaluation and final presentation video.

Project teams: Students choose between proposing their own project or participating in a structured project managed by TAs. They work in a team of 1, 2, or more persons (special requirements for teams of 3 or more), with guidance from instructors. Projects should be relevant to some aspect of the course (defined broadly). The design and implementation of projects will be discussed in class.

Grading components (current plan):

- Attendance: 10%. For courses with virtual delivery (Summer), the attendance credits will be re-allocated to Canvas online time (corresponding to lecture watching), timely

submission of assignments and reports, and participation in TA and instructor office hours.

- Reading papers and writing commentaries with the aid of GenAI tools: 15%.
- Project checkpoint reports: 75%. These start from the reproducibility exercise (checkpoint #0), continue with the initial project plan (checkpoint #1), through the intermediate checkpoints, and the final deliverables (checkpoint #k).
- Realistic feedback: the TAs provide a gold evaluation of each checkpoint report in terms of #Scope, #Match, and #Factual, as reference points for skill learning and training. The realistic feedback can be considered a floor function of each grade component.
- The weight of report grades will improve, depending on the evolution of report quality throughout the semester (from #1 to #k), their responsiveness to instructor/TA feedback, as well as the agreement of the gold evaluation with the self-evaluation.

Attendance and/or Participation

For in-residence courses (Fall and Spring), attendance will conform to institute policy. Currently, 90% physical attendance will be considered as having achieved full attendance credits. For courses with virtual delivery (Summer), the attendance credits will be re-allocated to Canvas online time (corresponding to lecture watching), timely submission of assignments and reports, and participation in TA and instructor office hours.

Academic Integrity

Georgia Tech aims to cultivate a community based on trust, academic integrity, and honor. Students are expected to act according to the highest ethical standards. Review Georgia Tech's Honor Code and the student Code of Conduct.

Any student suspected of cheating or plagiarism on a quiz, exam, or assignment will be reported to the Office of Student Integrity, who will investigate the incident and identify the appropriate penalty for violations.

Accommodations for Students with Disabilities

If you are a student with learning needs that require special accommodation, contact the Office of Disability Services (404-894-2563) as soon as possible to make an appointment to discuss your special needs and to obtain an accommodations letter. Please also e-mail me as soon as possible in order to set up a time to discuss your learning needs.

Student-Faculty Expectations Agreement

At Georgia Tech, we believe that it is important to strive for an atmosphere of mutual respect, acknowledgement, and responsibility between faculty members and the student body. The Student-Faculty Expectations articulate some basic expectations that you can have of me and that I have of you. In the end, simple respect for knowledge, hard work, and cordial interactions will help build the

environment we seek. Therefore, I encourage you to remain committed to the ideals of Georgia Tech while in this class.

Pre-requisites:

Completion of one database course or equivalent work experience.

Collaboration, Group Work, and Adoption of GenAI Tools

Project teams: Students choose between proposing their own project or participating in a structured project managed by TAs. They work in a team of 1, 2, or more persons (special requirements for teams of 3 or more), with guidance from instructors. Projects should be relevant to some aspect of the course (defined broadly). The design and implementation of projects will be discussed in class.

GenAI Tools: The written commentary assignments include training for using GenAI to aid writing. The project execution also encourages the adoption of GenAI tools as performance improvement aids. However, deliverables that are heavily (e.g., 100%) generated by such tools fall outside the definition of writing and/or programming aid. In these cases, the grades will be given to GenAI instead of the student.