Computer Science Department
cs.salemstate.edu

ITE 501 Information Technology Capstone Project Specification **1 cr.**

Instructor: TBA **Office:** location **Phone:** (978) 542-extension
email: TBA@salemstate.edu **Office Hours:** days and times

Section	Time	Room	Final Exam
nn	days and times	location	date and time

Catalog description:

The main purpose of this course is to develop a detailed proposal for ITE 505 Information Technology Capstone Project. The instructor will assist students in choosing appropriate project topics and refining their project proposals through all stages, from the initial outlines, the formal specification, and the final presentation. The completed proposal will serve as the contract for ITE 505. A presentation of the completed proposal is required. This course is graded on a Pass/Fail basis. Open only to Information Technology majors. One lecture hour per week, plus project proposal preparation work outside of class.

Prerequisites: ITE 350

Course Goals:

The purpose of this course is to develop students' ability to construct a proposal for a project in Information Technology. The goals of this course are:

- G1: to develop an appreciation for the process of formulating a project for implementation;
- G2: to develop the skills necessary to assess a project proposal for appropriateness and feasibility;
- G3: to further develop the skills and knowledge necessary to propose, analyze, design system or software projects;
- G4: to develop students' writing skills in the context of all aspects of the software engineering process;
- G5: to use written assignments and class discussion to teach students to write effectively for various purposes and audiences;
- G6: to have students experience writing as a process;
- G7: to give students experience in making and critiquing presentations.

Upon completion of the course, a student will have demonstrated the ability to perform the activities and techniques necessary to identify a potential development target, developed a formal project proposal, researched and selected a project design / architecture.

Course Outcomes (Objectives):

Upon successful completion of the course sequence, students will have:

- O1: demonstrated knowledge of the phases and workflows of the project development life cycle;
- O2: demonstrated knowledge of the major process models used in the development of large-scale systems;
- O3: demonstrated knowledge of the tools and techniques appropriate for implementation of the project, specifically including design/diagramming tools as appropriate for the project;
- O4: demonstrated knowledge of modern design paradigms;
- O5: developed a plan for project implementation;
- O6: presented and defended a project proposal to the department faculty and students;

O7: demonstrated the ability to critically analyze materials ranging from project proposals to technical specifications to scholarly research and to express this analysis clearly in both spoken and written form for a variety of appropriate audiences;

O8: demonstrated an understanding of writing as a process by giving and responding to feedback and reflecting on his/her own writing processes.

Course Narrative:

In ITE 501 Information Technology Capstone Project Specification students select a project focus/topic, and then develop a formal project proposal that specifies: intended functionality of the project; student objectives; technical aspects of designing and implementing the project; project schedule and evaluation criteria; and a list of deliverables that will be produced at the end of ITE 505 Information Technology Capstone Project.

The overarching goal of the ITE 501 / ITE 505 sequence is for students to experience all aspects of the development process from initial conception of intended functionality through to project completion. This experience ties together in one extended activity 809098the research, procedural, and technical aspects of the Information Technology major, simulating the environment that students will be expected to be able to function in upon graduation. The procedural and research aspects are the focus of ITE 501, wherein a project is proposed, defined, and planned for; the technical aspects of implementing a project are experienced in ITE 505. Additional research may be required in ITE 505 as a result of roadblocks detected during implementation, which may in turn require refinement of specified procedural aspects of the project.

Students are required to engage in writing activities throughout the course sequence. ITE 501 requires the development of project components intended to convey to potential *non-technically inclined* clients (ranging from owners to stakeholders to users) the proposed functionality of the project and to *technically inclined evaluators* a proposed solution, tools list, schedule, and evaluation criteria. The proposed solution must be accompanied by documentation of possible alternative strategies and justification of the selected solution.

Student activities relating to Written Communication - Level III criteria are found throughout the ITE501/ITE505 course sequence and are intimately integrated into the learning process. All project proposals are evaluated based on formal assessment rubrics; students are given the opportunity to make multiple submissions of all project components and are strongly encouraged to submit multiple drafts of proposed functionality documents, with each submission receiving feedback from the instructor. Instructor/student meetings provide multiple opportunities for students and the instructor to review work and to discuss the principles underlying their writing efforts. Proposal and project components include a wide assortment of activities designed to assist students in selecting a project process model and how the selected model will relate to project implementation.

The final grade for ITE 501 is determined by the formal project proposal, which is based on writing as it is commonly practiced within the field of Information Technology in general and the subfield of software engineering in particular. The final grade for ITE 505 is determined by the evaluation schema defined in the ITE 501 proposal and always includes a significant percentage determined by the project journal, project documentation, and the materials produced in support of the completed project presentation.

Program Outcome vs. Course Objectives matrix

Program Objective	O1	O2	O3	O4	O5	O6	O7	O8
PO-A: An ability to apply knowledge of computing and mathematics appropriate to the program's student outcomes and to the discipline.	√	√	√	√	√	√	√	
PO-B: An ability to analyze a problem, and identify and define the computing requirements appropriate to its solution.	√	√	√	√	√	√	√	√

Program Objective	O1	O2	O3	O4	O5	O6	O7	O8
PO-C: An ability to design, implement, and evaluate a computer-based system, process, component, or program to meet desired needs.				√			√	√
PO-D: An ability to function effectively on teams to accomplish a common goal.								
PO-E: An understanding of professional, ethical, legal, security and social issues and responsibilities.								√
PO-F: An ability to communicate effectively with a range of audiences.						√		√
PO-G: An ability to analyze the local and global impact of computing on individuals, organizations, and society.								√
PO-H: Recognition of the need for and an ability to engage in continuing professional development.		√	√	√			√	
PO-I: An ability to use current techniques, skills, and tools necessary for computing practice.		√	√	√	√	√	√	√
PO-J: An ability to use and apply current technical concepts and practices in the core information technologies.		√	√	√		√		√
PO-K: An ability to identify and analyze user needs and take them into account in the selection, creation, evaluation and administration of computer-based systems.		√	√	√	√	√	√	√
PO-L: An ability to effectively integrate IT-based solutions into the user environment.	√	√	√	√	√	√	√	√
PO-M: An understanding of best practices and standards and their application.	√	√	√	√	√	√	√	√
PO-N: An ability to assist in the creation of an effective project plan.	√	√	√	√	√	√	√	√

Note: All projects are expected and required to relate to the specific Program Objectives as indicated above. In addition, a specific project may relate to Program Objective PO-D depending on whether the project involves multiple students.

Topics:

- requirements: **SIA1(2), SIA4(0.5), SP2(1)**
  - review of the initial phases of the development of a formal proposal
  - project planning
  - investigation of general needs
  - analysis of existing functionalities
  - proposal of a set of new/modified functionalities
- review of the systems integration process **SIA3(2)**

- basic principles
- components, interfaces, and integration
- the development life cycle

- review, as necessary, of systems analysis techniques **SP2(1), SP1(0.5), SP2(0.5)**
  - information gathering
  - team communication
  - feasibility studies
- project management techniques **SIA4(1.5)**
  - roles and responsibilities for key project personnel and stakeholders
- overview of systems architecture **SIA7(1)**
  - for software projects, primary focus on ADTs, object recognition and specification, and file/database design (if appropriate)
  - for hardware projects, primary focus on system block diagrams, system circuit diagrams and wiring diagrams
- review of general implementation, acquisition, testing, and quality assurance **SIA2(2), SIA5(1.5)**
  - reliability
  - testing
  - verification
  - evaluation and benchmarking
- design, analysis, and documentation of: **SIA4(0.5),SIA6(0.5),SP6(1),SP3(1), SP4(1), SP1(1.5), SP5(1)**
  - project requirements
  - selection of project planning and management strategies
  - intellectual property consideration
  - assessment of project quality: selection of and evaluation against appropriate style rules for code and documentation
  - test cases as use case diagrams and/or scenarios and/or stories and/or automated test cases

Student Experiences:

The primary goal of ITE 501 is to guide students through the process of designing a detailed proposal for a software- or hardware-system project and specifying its implementation requirements at a level appropriate to the proposed project. Students will choose an application arena of sufficient complexity so as to necessitate a non-trivial solution to the problem of designing and implementing a solution for the project. The selected topic area will then be studied through research and discussion. After a thorough analysis of the functionalities required by the proposed project, students will develop and present to the instructor and the ITE505 supervisor various data modeling and system architecture possibilities: the possibilities will be iteratively discussed with and evaluated by the ITE505 supervisor, leading to a final document that:

- describes the functionalities of the proposed system in clear, concise and non-technical terms;
- specifies the tools necessary to implement a solution;
- defines a high-level design architecture for a solution;
- specifies important developer-designed objects required to represent the application area;
- describes the implementation techniques that are appropriate for manipulating the objects;
- presents an implementation schedule;
- presents a mechanism for determination of the final grade for ITE 501.

The finished ITE 501 proposal will be presented to department faculty and to the department at large at the end of the semester (typically on Reading Day).

The (pass/fail) grade for ITE 501 will be based on the final proposal document (in particular on the analysis of the required functionalities, the scope of the project, and on the appropriateness of any proposed design(s)) and the quality of the presentation and defense of the proposal. The finalized document will act as the contract document for the project that is to be implemented in ITE 505.

Course Objective / Assessment Mechanism matrix

	Proposal

	Problem Specification	Proposed Solution Design	Proposed Implementation Techniques and Tools	Presentation
O1	✓	✓	✓	✓
O2	✓	✓	✓	✓
O3	✓	✓	✓	✓
O4	✓	✓	✓	✓
O5	✓	✓	✓	✓
O6	✓	✓	✓	✓
O7	✓	✓	✓	✓
O8	✓	✓	✓	✓
O9	✓	✓	✓	✓
O10	✓	✓	✓	✓

Bibliography: Highly variable, dependent upon application area selected by student.

Requirements:

First: the onus of picking a topic, developing a proposal and completing the proposal is on the student (or group of students). The instructor is available for consultation and suggestions, but the student(s) are responsible for "making things happen". Students should expect to meet the deadlines set by the instructor to move their proposals forward.

Proposed projects for ITE 501/505 must involve the design and implementation of a moderate-to-large system or software project. Proposals should adhere to the following general guidelines:

- Proposed projects should be primarily applications-oriented and non-trivial in nature; projects must exhibit algorithmic complexity and/or research into area(s) new to the student and may not be simply "output generators". Projects may be disallowed for insufficient technical content, duplication of current or previous projects, or insufficient background on the part of the student.
- The main focus of the project must draw upon one or more upper-level (above ITE 210) courses, utilizing and possibly extending information (algorithms, structures, methodologies, etc.) acquired in such courses, and will preferably involve integration of concepts and technologies presented in multiple courses;
- During ITE 501 the instructor will provide guidance for the students in choosing a topic, designing the proposal, choosing a ITE505 capstone project supervisor, determining appropriate components (with consultation and input from the project supervisor) for the final report and presentation, and creating a proposal presentation; Once a project supervisor has been selected and a topic / application area agreed upon, the student must prepare a formal proposal detailing the specific requirements and expectations of the project.
  The proposal must include the following components (explained in more detail below). Proposals lacking any of the following components will not be scheduled for presentation.

  ☐ Cover Page
  ☐ Student Objectives
  ☐ Problem Specification
  ☐ Solution Processes and/or Design
  ☐ Benchmark Specifications
  ☐ Tools List
  ☐ Time Schedule
  ☐ Grading Scheme
  ☐ List of Deliverables
  ☐ Presentation (must be in "presentation format", e.g., Microsoft Office PowerPoint, OpenOffice, Impress, Prezi, etc.)

Cover Page

Center the project title on the page. Place the name(s) of all student participants under the title. Place the name of the ITE505 project supervisor and the ITE501 presentation date in the bottom right corner.

Student Objectives

State what your personal goals and objectives for the project are, that is, state in general terms what you hope to accomplish by completing your proposal and project, and then state the specific new skills and/or skill enhancements you expect to demonstrate via your project. Examples include "experience with advanced database design concepts", "experience with the complete life cycle of a project, from initial fact-finding and problem specification all the way through to implementation, verification and documentation", "ability to install, configure and use MySQL".

Problem Specification

Describe in clear *non-technical* language what the project will attempt to do: focus on the project's ***functionality*** and ***not*** on the *technical* aspects of its implementation. Explain any terms that may be unknown to a reader unfamiliar with the specific subject area of the proposal. Any use of technical vocabulary and concepts (terminology unlikely to be familiar to an audience not trained in Information Technology) is very strongly discouraged. Focus on *what* the project will accomplish, that is, on the functionalities that it will support. The problem specification should be one to two pages in length; anything longer than two pages tends to be either too detailed or too ambitious for a single-semester implementation. Group projects may need to exceed the two-page limit.

Solution Design

Provide a high-level (architectural, abstract) design of the proposed solution. Begin with a graphic showing the relationship(s) amongst the major solution components. For each of the components, describe the design of the solution in a format appropriate to the subject area (e.g., ER diagrams for database-centric projects, UML diagrams (class, activity, interaction and/or use-case) for large-scale software projects, etc.). Each component of the design must be accompanied by a brief paragraph describing the responsibility (intended functionality) of the component.

Benchmark Specifications

Benchmarks must be defined which will allow progress in the project to be monitored and documented. The benchmarks must be objective, readily measurable, and must clearly relate to one or more components of the solution design.

Tools List

List any and all tools that may be used in developing a solution to the problem. Tools include (but are not limited to):

- any software or hardware that will be used at any stage of the process, including (but not limited to) program language(s), IDEs, APIs. CASE environments, operating system(s), communication protocols, general productivity tools, FPG kits, hardware controller, cameras, etc.;
- algorithms and/or data structures, *if beyond those implemented as part of previous coursework*.

If evaluation and selection of tools is part of project implementation, state so explicitly *as part of the problem specification*; include a list of potential candidates and specify the criteria to be used in selecting specific tools.

Time Schedule

Establishing a timetable and agreeing on a reasonable rate of progress on the project is the joint responsibility of the student(s) and the faculty supervisor. List the major components/benchmarks from the previous two steps in the order in which it is anticipated they will be completed. Indicate which (if any) are dependent on earlier steps, and which (if any) can be worked on simultaneously (Gant or PERT charts may be appropriate). Include approximately how much time each component should take (in days or weeks): the total amount of time allocated should be approximately 14 weeks.

Grading Scheme

Possibilities include allocating a percentage of the grade to each of the components / benchmarks of the project or specifying the set of benchmarks representing progress of the project and awarding a final grade based on how many of the benchmarks have been reached *and documented*. Use the **Time Schedule** list as a reference for the components / benchmarks. Your ITE505 supervisor must approve the final grading scheme. Note that the presentation of the completed project must be allocated 10% of the final grade for ITE 505.

Deliverables

The specific list of deliverables will vary from project to project. Typical deliverable components include, *but are not limited to*, the following. **Note that all projects must include the components presented in boldface.**

Not all components will be included in all proposals; additional components may be required at the discretion of your project supervisor, based on the nature of the proposed project. ***Note that the following list is of components that are to be delivered upon the completion of ITE 505, not ITE 501.*** *In ITE 501, you are* ***listing (specifying)*** *what* ***will be*** *included in the completed package.*

- **original proposal and presentation file(s) (from ITE 501)**
- **amendments to the proposal (approved by the project supervisor**)
- system architecture diagram(s) (UML, DFD context, etc.), enhanced with details determined during implementation
- **appropriately commented source code**
- **documentation of project functionality (test results, screenshots, video capture of project execution, etc.)**
- **sample output (screen shots and/or reports)**
- **executables and/or projects**
- **presentation documents (used to support the presentation of the completed ITE 505 project), including any presentation file(s)**
- **project journal: a narrative of the progress of the project, in clear, concise English, including any problems encountered and how said problems were addressed**
- **project *post mortem*: a summary of what was learned from the project and (based on that experience) discussion of how various aspects of the project might have been approached differently**
- **a list of what areas of the proposal (if any) were not completed, including reasons why**
- **presentation of the completed project (PowerPoint format), including screenshots of the functioning project**
- user's manual

Additional components may be required at the discretion of the project supervisor, based on the nature of the proposed project. ***Note that the preceding list is of components that are to be delivered upon the completion of ITE 505, not ITE 501.*** *In ITE 501, you are listing what* ***will be*** *included in the completed package.*

Presentation

A slide show-based presentation of the proposal must be created, reviewed by the faculty supervisor, and presented to the Computer Science Department. The presentation must be a *summary* of the proposal, *not* a duplicate of it. In particular, the Project Specification and Solution Design components of the formal project proposal will always need to be summarized / condensed; the remaining components can sometimes be copied out of the proposal and pasted into the presentation but will usually also require summarization. There are no set limits on the number of slides - however, note that you will have no more than 15 minutes in which to make your presentation and respond to questions and comments, so the presentation itself should be limited to 10-12 minutes.

Suggested Course Plan

Week 1 topics	Week 1 activities
• Present and discuss syllabus • Picking a project – student objectives, topic selection, domain selection • Capstone project proposal and implementation as potential employment resources • Writing expectations and writing grading criteria • Submission formatting requirements (typeface, font size, margins, line spacing, no typographical errors, etc.)	• (A1.1) Students develop a list of their objectives for their project (due Week 2) • (A1.2) Student develop three half-page outlines of potential projects – assignment includes specific of what is expected (due Week 2)

Week 2 topics	Week 2 activities
• Revisit picking a project • Identifying potential project supervisors	• Students exchange and discuss objectives and potential project outlines

	• (A2) Visit with potential project supervisors, obtain proof of visit (form signature), half-page summary of each visit (due Week 3)

Week 3 topics	Week 3 activities
• Finalize project selection, project supervisor selection	• Students exchange and discuss (A2) submissions; Instructor leads discussion • (A3.1) Initial *partial* draft of proposal – student objectives + one-page outline of functional requirements (due Week 4) • (A3.2) Project supervisor signs form indicating that they agree to supervise project and that they have received a copy of the initial partial draft (due Week 4)

(A3.1) and (A3.2) must be submitted to the instructor prior to Week 4 class meeting – failure to submit results will be counted as a class absence.
If (A3.1) and (A3.2) not submitted prior to Week 5 class meeting – class failure, student(s) will start ITE501 again next semester.

Weeks 4 & 5 topics	Weeks 4 & 5 activities
• Discuss and guide towards finalization of student objectives • Discuss and guide towards elaboration of project's functional requirements • Overview of all remaining project proposal components: architectural design; abstract solution design; tools list; time schedule; milestone list; grading scheme; deliverable list; proposal presentation	• Students exchange and comment on each other's initial draft of proposals • Finalized student objectives (due Week 6) • Elaborated functional requirements (minimum 1 ½ pages, maximum 2 pages) (due Week 6) • Rough draft of architectural design; abstract solution design; tools list (due Week 6)

Weeks 6 & 7 topics	Weeks 6 & 7 activities
• Architectural design, abstract solution design, and tools list discussed in detail, with instructional examples	• Students exchange and comment on each other's proposals • Draft extended to include elaboration of architectural design, abstract solution design, and tools list components of proposal – focus on thoroughness (due Week 8)

Week 8 topics	Week 8 activities
• Architectural design, abstract solution design, and tools list reviewed, and questions answered • Discussion of time schedule, milestone list, and grading scheme; organization options for each; separate components vs. single integrated component	• Students exchange and comment on each other's proposals • Draft revised and enhanced (due Week 9)

Week 9 topics	Week 9 activities
• Time schedule, milestone list, and grading schemes reviewed; questions answered • "Proof of concept" and "prototyping"	• Students exchange and comment on each other's proposals • Draft revised and enhanced; *resulting draft shared with project supervisor.* • Meet with project supervisor to discuss *within one week of sharing draft with supervisor*; project supervisor signs form indicating that they received the draft in a timely fashion, and that they met with the supervisee and provided feedback; student documents feedback. • Proposal will be revised based on project supervisor's feedback (due Week 10)

Week 10 topics	Week 10 activities
• Discussion of deliverables list; project proposal presentation expectations • More on "proof of concept" and "prototyping"	• Students exchange and comment on each other's proposals • Draft of time schedule, milestone list, and grading scheme created as a separate document • Draft of time schedule, milestone list, and grading scheme inserted into proposal *before submission to ITE 501 Instructor for Week 11* (due Week 11)

Week 11 topics	Week 11 activities
• Finalizing project proposal	• Students exchange and comment on each other's proposals • Finalize the proposal (insertion of deliverables list), delivered to Instructor *and project supervisor* • Meet with project supervisor to discuss the finalized proposal. Project supervisor signs form indicating that they received the proposal in a timely fashion, and that they met with the supervisee and provided feedback; student documents feedback. • Revised proposal based on project supervisor's feedback to finalize the proposal to its final version. (due Week 12)

Week 12 topics	Week 12 activities
• Review of the project proposal process	• Students exchange and comment on each other's proposals • Final version of project proposal submitted to both Instructor and project supervisor.

Weeks 13 & 14 topics	Weeks 13 & 14 activities
• Project proposal presentation details – format, length (physical and presentation time), style considerations, etc.	• Produce presentation files • Perform mock up presentations (outside class time scheduled when needed)