CEAS-14-CS-082

REQUEST TO COLLEGE CURRICULUM COMMITTEE FOR CURRICULAR IMPROVEMENTS

DEPARTMENT: CS **PROPOSED EFFECTIVE SEMESTER:** Spring 2015 **COLLEGE:** **CEAS**

PROPOSED IMPROVEMENTS

Academic Program

□ New degree*
□ New major*
□ New curriculum*
□ New concentration*
□ New certificate
□ New minor
□ Revised major
□ Revised minor
□ Admission requirements
□ Graduation requirements
□ Deletion □ Transfer
□ Other (explain**)

Substantive Course Changes

□ New course
☒ Pre or Co-requisites
□ Deletion (required by others)
□ Course #, different level
□ Credit hours
□ Enrollment restriction
□ Course-level restriction
□ Prefix □ Title and description (attach current & proposed)
□ General education (select one) Not Applicable
□ Other (explain**)

Misc. Course Changes

☒ Title
☒ Description (attach current & proposed)
□ Deletion (not required by others)
□ Course #, same level
□ Variable credit
□ Credit/no credit
□ Cross-listing
□ COGE reapproval
☒ Other (explain**)

** **Other:** A minimum grade of B in the prerequisite courses

Title of degree, curriculum, major, minor, concentration, or certificate:

Existing course prefix and #: CS6610 **Proposed course prefix and #:** **Credit hours:**

Existing course title: Software Engineering II: Verification and Validation of Software Systems
Proposed course title: Software Testing

Existing course prerequisite & co-requisite(s): CS6600
Proposed course prerequisite(s) : CS3310
If there are multiple prerequisites, connect with "and" or "or". To remove prerequisites, enter "none."
Proposed course co-requisite(s)
If there are multiple corequisites, they are always joined by "and."
Proposed course prerequisite(s) that can also be taken concurrently:
Is there a minimum grade for the prerequisites or corequisites?
A minimum grade of B in the prerequisite course.
Major/minor or classification restrictions:
List the Banner 4 character codes and whether they should be included or excluded.
For 5000 level prerequisites & corequisites: Do these apply to: (circle one) undergraduates graduates both

Specifications for University Schedule of Classes:
a. Course title (maximum of 30 spaces): Software Testing
b. Multi-topic course: ☒ No □ Yes
c. Repeatable for credit: ☒ No □ Yes
d. Mandatory credit/no credit: ☒ No □ Yes
e. Type of class and contact hours per week (check type and indicate hours as appropriate)
1. □ Lecture
2. □ Lab or discussion
3. ☒ Lecture/lab/discussion
4. □ Seminar or □ studio
5. □ Independent study
6. □ Supervision or practicum

CIP Code (Registrar's use only):

Chair/Director Date 2/27/15

Chair, College Curriculum Committee Date

Dean Date: Graduate Dean: Date

Curriculum Manager: Return to dean □ Date Forward to: Date

Chair, COGE/ PEB / FS President Date

FOR PROPOSALS REQUIRING GSC/USC REVIEW:

* □ Approve □ Disapprove | Chair, GSC/USC Date

* □ Approve □ Disapprove | Provost Date

1. Explain briefly and clearly the proposed improvement.

   This proposed improvement is to change the title of CS6610 from "Software Engineering II: Verification and Validation of Software Systems" to "Software Testing", and to change its prerequisite from CS6600 to CS3310. The proposed description "Students will learn practical ways to design high quality tests during various phases of software development. Students learn the theory behind criteria-based test design and to apply that theory in practice. Topics include test design, test automation, test coverage criteria, and how to test software in state-of-the-art software development environments" reflects this change.

   A minimum grade of B in the prerequisite course.

2. Rationale. Give your reason(s) for the proposed improvement. (If your proposal includes prerequisites, justify those, too.)
   - Title. Verification and validation can be considered part of software testing. With the rapid development in the field of software engineering testing, it is more appropriate to use a more general term so more recent development can be covered in the course. In addition, CS6610 no longer requires CS6600 as prerequisite. Sequential numbers of I and II associated with CS6600 and CS6610, respectively, will unnecessarily discourage those students who did not take CS6600 to register CS6610.
   - Prerequisite. Software testing is a relatively self-contained. The current teaching of CS6610 does not depends on the knowledge in CS6600 (Software Engineering I: Formal Specifications of Software Systems). The topics covered in CS3310 are sufficient for teaching CS6610.

3. Effect on other colleges, departments or programs. If consultation with others is required, attach evidence of consultation and support. If objections have been raised, document the resolution. Demonstrate that the program you propose is not a duplication of an existing one.

   None.

4. Effect on your department's programs. Show how the proposed change fits with other departmental offerings.

   The perspective students will have a better understanding of what they expect to learn from the course.

5. Effects on enrolled students: Are program conflicts avoided? Will your proposal make it easier or harder for students to meet graduation requirements? Can students complete the program in a reasonable time? Show that you have considered scheduling needs and demands on students' time. If a required course will be offered during summer only, provide a rationale.

   With the change of prerequisite form CS6600 to CS3310, it becomes easier for students to register CS6610.

6. Student or external market demand. What is your anticipated student audience? What evidence of student or market demand or need exists? What is the estimated enrollment? What other factors make your proposal beneficial to students?

   No change.

7. Effects on resources. Explain how your proposal would affect department and University resources, including faculty, equipment, space, technology, and library holdings. Tell how you will staff additions to the program. If more advising will be needed, how will you provide for it? How often will course(s) be offered? What will be the initial one-time costs and the ongoing base-funding costs for the proposed program? (Attach additional pages, as necessary.)

   None.

8. General education criteria. For a general education course, indicate how this course will meet the criteria for the area or proficiency. (See the General Education Policy for descriptions of each area and proficiency and the criteria. Attach additional pages as necessary. Attach a syllabus if (a) proposing a new course, (b) requesting certification for baccalaureate-level writing, or (c) requesting reapproval of an existing course.) N/A

9. List the learning outcomes for the proposed course or the revised or proposed major, minor, or concentration. These are the outcomes that the department will use for future assessments of the course or program. N/A

10. Describe how this curriculum change is a response to assessment outcomes that are part of a departmental or college assessment plan or informal assessment activities. N/A

11. (Undergraduate proposals only) Describe, in detail, how this curriculum change affects transfer articulation for Michigan community colleges. For course changes, include detail on necessary changes to transfer articulation from Michigan community college courses. For new majors or minors, describe transfer guidelines to be developed with Michigan community colleges. For revisions to majors or minors, describe necessary revisions to Michigan community college guidelines. Department chairs should seek assistance from college advising directors or from the admissions office in completing this section. N/A

Current Catalog Description:

CS 6610 - Software Engineering II: Verification and Validation of Software Systems

The terminology and limitations of verification and validation (V and V) approaches. Five approaches will be presented: technical reviews, testing, proofs of correctness, simulation and prototyping, and requirements tracing. Students will define a V and V plan and carry it out for several stages in the development cycle of a project.

Prerequisites/Corequisites: Prerequisite: CS 6600.

Proposed Catalog Description

CS 6610 - Software Testing

Students will learn practical ways to design high quality tests during various phases of software development. Students learn the theory behind criteria-based test design and to apply that theory in practice. Topics include test design, test automation, test coverage criteria, and how to test software in state-of-the-art software development environments.

Prerequisites/Corequisites: Prerequisite: CS 3310.

Credits: 3 hrs.

Notes: Open to Graduate Students Only.