New York City College of Technology, CUNY

CURRICULUM MODIFICATION PROPOSAL FORM

This form is used for all curriculum modification proposals. See the [Proposal Classification Chart](http://openlab.citytech.cuny.edu/collegecouncil/files/2014/08/2013-10-09-Proposal_Classification_Chart.pdf) for information about what types of modifications are major or minor. Completed proposals should be emailed to the Curriculum Committee chair.

|  |  |
| --- | --- |
| **Title of Proposal** | Proposal for new course:  CET4915: Agile Testing of Embedded Software |
| **Date** | Sep 10, 2018 |
| **Major or Minor** | Major |
| **Proposer’s Name** | Yu Wang, Sunghoon Jang, Farrukh Zia |
| **Department** |  |
| **Date of Departmental Meeting in which proposal was approved** | Sep 27, 2018 |
| **Department Chair Name** | Sunghoon Jang |
| **Department Chair Signature and Date** |  |
| **Academic Dean Name** | Kevin Hom |
| **Academic Dean Signature and Date** |  |
| **Brief Description of Proposal**  (Describe the modifications contained within this proposal in a succinct summary. More detailed content will be provided in the proposal body. | Computer engineering technology department proposes a new technical elective course, CET4915 Agile Testing of Embedded Software, in CEB BTech program curriculum. CET4915 is designed for advanced level CEB students. |
| **Brief Rationale for Proposal**  (Provide a concise summary of why this proposed change is important to the department. More detailed content will be provided in the proposal body). | Testing has evolved from afterthought to a central activity in certain development methods, particularly agile methods.  This Agile Testing of Embedded Software course is designed to meet the needs of CET students for the technical electives.  This course allows students to develop their knowledge further to design tests in the phases of embedded software development, including hardware, electronics, and software. An understanding of Agile allows our CET students to work on any project in any sector in the tech field. |
| **Proposal History**  (Please provide history of this proposal: is this a resubmission? An updated version? This may most easily be expressed as a list). | Initial submission. |

Please include all appropriate documentation as indicated in the Curriculum Modification Checklist.

For each new course, please also complete the New Course Proposal and submit in this document.

Please submit this document as a single .doc or .rtf format. If some documents are unable to be converted to .doc, then please provide all documents archived into a single .zip file.

**ALL PROPOSAL CHECK LIST**

|  |  |
| --- | --- |
| Completed CURRICULUM MODIFICATION FORM including: |  |
| * Brief description of proposal | y |
| * Rationale for proposal | y |
| * Date of department meeting approving the modification | y |
| * Chair’s Signature | y |
| * Dean’s Signature |  |
| Evidence of consultation with affected departments  List of the programs that use this course as required or elective, and courses that use this as a prerequisite. |  |
| Documentation of Advisory Commission views (if applicable). |  |
| Completed [Chancellor’s Report Form](http://openlab.citytech.cuny.edu/collegecouncil/files/2014/08/2013-10-09-Chancellor_Report_Quick_Reference_Guide1.doc). | y |

**EXISTING PROGRAM MODIFICATION PROPOSALS**

|  |  |
| --- | --- |
| Documentation indicating core curriculum requirements have been met for new programs/options or program changes. |  |
| Detailed rationale for each modification (this includes minor modifications) |  |

Table of Contents

[NEW COURSE PROPOSAL FORM 4](#_Toc526955853)

[NEW COURSE PROPOSAL CHECK LIST 5](#_Toc526955854)

[Rationale: 7](#_Toc526955855)

[Course *Outline* 8](#_Toc526955856)

[Recommended Instructional Materials 9](#_Toc526955857)

[Textbook and Reference Book 9](#_Toc526955858)

[Suggested References 9](#_Toc526955859)

[Weekly Topic (lecture and lab) 11](#_Toc526955860)

[Course Policies 12](#_Toc526955861)

[Course Design and Assessment 14](#_Toc526955862)

[Assessment Criteria 14](#_Toc526955863)

[General Education Outcomes and Assessment: 15](#_Toc526955864)

[LIBRARY RESOURCES & INFORMATION LITERACY: MAJOR CURRICULUM MODIFICATION 16](#_Toc526955865)

[Chancellor’s Report 18](#_Toc526955866)

[Appendix A: Sample lecture and lab 19](#_Toc526955867)

[Sample Lecture: Model-driven test design 19](#_Toc526955868)

[Sample Lab: Extract the testable components from an embedded system 21](#_Toc526955869)

[Appendix B: Department Meeting Minutes 22](#_Toc526955870)

# NEW COURSE PROPOSAL FORM

New York City College of Technology, CUNY

This form is used for all new course proposals. Attach this to the [Curriculum Modification Proposal Form](http://openlab.citytech.cuny.edu/collegecouncil/files/2014/08/2013-10-10-Curriculum_Modification_Proposal_Form.docx) and submit as one package as per instructions. Use one New Course Proposal Form for each new course.

|  |  |
| --- | --- |
| **Course Title** | Agile Testing of Embedded Software |
| **Proposal Date** | September 10, 2018 |
| **Proposer’s Name** | Yu Wang, Sunghoon Jang, Farrukh Zia |
| **Course Number** | CET 4915 |
| **Course Credits, Hours** | 3 Class hour, 3 Lab Hours, 4 Credits |
| **Course Pre / Co-Requisites** | Pre: CET3510; Pre/Co: CET3640 |
| **Catalog Course Description** | Introduction to the theory and principles of testing embedded software. The topics cover the basics of test design, test automation, and test-driven development, agile principles, values, techniques, practices, procedures, and tools. The lab and cases provide students with the opportunity to apply the theory in practical ways to design tests during the phases of embedded software development. |
| **Brief Rationale**  Provide a concise summary of why this course is important to the department, school or college. | This Agile Testing of Embedded Software course is designed to meet the needs of CET students for the technical electives.  This course allows students to develop their knowledge further to design tests in the phases of embedded software development, including hardware, electronics, and software. An understanding of Agile allows our CET students to work on any project in any sector in the tech field. |
| **CUNY – Course Equivalencies**  Provide information about equivalent courses within CUNY, if any. | N.A |
| **Intent to Submit as Common Core**  If this course is intended to fulfill one of the requirements in the common core, then indicate which area. | N.A |
| **For Interdisciplinary Courses:**   * Date submitted to ID Committee for review * Date ID recommendation received   - Will all sections be offered as ID? Y/N | N.A |
|  |
|  |
| **Intent to Submit as a Writing Intensive Course** | N.A |

Please include all appropriate documentation as indicated in the NEW COURSE PROPOSAL Combine all information into a single document that is included in the Curriculum Modification Form.

# NEW COURSE PROPOSAL CHECK LIST

Use this checklist to ensure that all required documentation has been included. You may wish to use this checklist as a table of contents within the new course proposal.

|  |  |
| --- | --- |
| **Completed NEW COURSE PROPOSAL FORM** |  |
| * Title, Number, Credits, Hours, Catalog course description | y |
| * Brief Rationale | y |
| * CUNY – Course Equivalencies | y |
| Completed [Library Resources and Information Literacy Form](https://openlab.citytech.cuny.edu/collegecouncil/files/2014/08/curriculum_modification_library_form-rev3F16.doc) |  |
| **Course Outline**  Include within the outline the following. | **y** |
| Hours and Credits for Lecture and Labs  If hours exceed mandated Carnegie Hours, then rationale for this | y |
| Prerequisites/Co- requisites | y |
| Detailed Course Description | y |
| Course Specific Learning Outcome and Assessment Tables   * Discipline Specific * General Education Specific Learning Outcome and Assessment Tables | y |
| Example Weekly Course outline | y |
| Grade Policy and Procedure |  |
| Recommended Instructional Materials (Textbooks, lab supplies, etc) | y |
| Library resources and bibliography |  |
| **Course Need Assessment.**  Describe the need for this course. Include in your statement the following information. | y |
| Target Students who will take this course. Which programs or departments, and how many anticipated?  Documentation of student views (if applicable, e.g. non-required elective). | y |
| Projected headcounts (fall/spring and day/evening) for each new or modified course. | y |
| If additional physical resources are required (new space, modifications, equipment), description of these requirements. If applicable, Memo or email from the VP for Finance and Administration with written comments regarding additional and/or new facilities, renovations or construction. | y |
| Where does this course overlap with other courses, both within and outside of the department? |  |
| Does the Department currently have full time faculty qualified to teach this course? If not, then what plans are there to cover this? | y |
| If needs assessment states that this course is required by an accrediting body, then provide documentation indicating that need. |  |
| **Course Design**  Describe how this course is designed. | Y |
| Course Context (e.g. required, elective, capstone) | y |
| Course Structure: how the course will be offered (e.g. lecture, seminar, tutorial, fieldtrip)? | y |
| Anticipated pedagogical strategies and instructional design (e.g. Group Work, Case Study, Team Project, Lecture) | y |
| How does this course support Programmatic Learning Outcomes? | y |
| Is this course designed to be partially or fully online? If so, describe how this benefits students and/or program. |  |
| **Additional Forms for Specific Course Categories** |  |
| [Interdisciplinary Form](http://openlab.citytech.cuny.edu/collegecouncil/files/2014/08/Application-for-Interdisciplinary-Course-Designation.docx) (if applicable) |  |
| Interdisciplinary Committee Recommendation (if applicable and if received)\*  \*Recommendation must be received before consideration by full Curriculum Committee |  |
| [Common Core (Liberal Arts) Intent to Submit](http://openlab.citytech.cuny.edu/collegecouncil/files/2014/08/CommonCoreCourseSubmissionForm_4.2.12.doc) (if applicable) |  |
| Writing Intensive Form if course is intended to be a WIC (under development) |  |
| If course originated as an experimental course, then results of evaluation plan as developed with director of assessment. |  |
| **(Additional materials for** [**Curricular Experiments**](http://www.300jaystreet.com/college-council/curriculum_proposals/curricular-experiments)**)** |  |
| Plan and process for evaluation developed in consultation with the director of assessment. (Contact Director of Assessment for more information). |  |
| Established Timeline for Curricular Experiment |  |

# Rationale:

Embedded software testing is a very specialist skill. Embedded testing can be performed manually or by using automated software testing tools. There are many software testing techniques. The tester needs to understand when and how to use them. The role of testing in software development has undergone radical changes in recent years. Testing has evolved from afterthought to a central activity in certain development methods, particularly agile methods. This Agile testing course is designed to meet the needs of CET students for the technical electives. This course teaches them how to identify requirements and execute on their ideas. Giving them the ability to determine the proper direction and lead their team.

Embedded systems exist as part of a bigger system. Nowadays, it has become more complex, Thousands and thousands of lines of codes now work with devices. The embedded skill sets become partitioned by discipline: hardware developer, firmware developer, software developer, and software and system tester.

Over the past decade or more, the software team has benefited from an Agile method. Agile testing brings many advantages to teams, from increasing overall product quality to providing greater scope for flexibility. According to VersionOne - State of Agile Survey 2017, 52% of respondents stated that more than half of the teams in their organizations are using Agile practices. There is 75% engineering practice in unit testing to use Agile method and 35% engineering practice to employ TDD (Test-driven development). Agile methodology has been widely adopted in software development and testing. Embedded software testing methodologies share much in common with software testing. Agile practices can apply well to embedded product development including hardware, electronics, and software.

The principles of Agile testing in software are applicable everywhere. An understanding of Agile allows our CET students to get entry point in tech fields and to work on any project in any sector. Indeed search for Software testing Tech job in New York shows that 40.2% job descriptions require Agile testing experience (2847 job requires Agile testing experience over 7079 jobs of software testing in New York). <https://www.indeed.com/jobs?q=test+software&l=New+York%2C+NY&radius=50> [results +7079 https://www.indeed.com/jobs?q=Agile+and+test+&l=New+York%2C+NY&radius=50](https://www.indeed.com/jobs?q=Agile+and+test+&l=New+York%2C+NY&radius=50) results +2847

Agile testing knowledge is essential for CET students who want to become better engineers or technicians. Agile testing makes students understand better at software debugging, which in turn leads to better projects. This understanding can enable students to their ideas from inception to release, create better systems, and communicate on a professional team. Students have an option to get the career booster of software testing –ISTQB Foundation Level software testing certification (CTFL) after taking this course. This certification gives students a common testing vocabulary and principles and opens a software and embedded testing career path for our CET students.

# Course *Outline*

**Course outline**

New York City College of Technology

Computer Engineering Technology Department

BTech Program in Computer Engineering Technology

**CET4915: Agile Testing of Embedded Software**

|  |  |  |  |
| --- | --- | --- | --- |
| **Prepared by** | Yu Wang, Sunghoon Jang, Farrukh Zia | **Revision date** |  |
| **Course No. & Title** | CET 4915 Agile Testing of Embedded Software | | |
| **Course Description** | Introduction to the theory and principles of testing embedded software. The topics cover the basics of test design, test automation, and test-driven development, agile principles, values, techniques, practices, procedures, and tools. The lab and cases provide students with the opportunity to apply the theory in practical ways to design tests during the phases of embedded software development. | | |
| **Hours / Credits** | 3 Class hour, 3 Lab Hours, 4 Credits | | |
| **Pre-requisites** | Pre-Requisites: CET3510 | | |
| **Pre / Co-requisites** | Co-Requisites: CET 3640 | | |
| **Gen-Ed Objectives** | 1. SKILLS/Inquiry/Analysis: Students will employ scientific reasoning and logical thinking. 2. SKILLS/Communication: Students will communicate in diverse settings and groups, using written (both reading and writing), oral (both speaking and listening), and visual means 3. INTEGRATION/Integrate learning: Students will resolve difficult issues creatively by employing multiple systems and tools. 4. VALUES, ETHICS, RELATIONSHIPS / Professional/Personal Development: Students will work with teams, including those of diverse composition. Build consensus. Respect and use creativity. | | |
| **Course Objectives** | Upon successful completion of this course, students shall be able to   1. Demonstrate knowledge of fundamental testing principles and processes associated with embedded systems. 2. Demonstrate understanding handling of the differences between testing in traditional and agile approaches 3. Demonstrate understanding Agile core principles and values 4. Design embedded system tests and test software in structured and organized ways 5. Apply Agile software principles into practice and procedure | | |

# Recommended Instructional Materials

## Textbook and Reference Book

|  |  |  |
| --- | --- | --- |
| **Text book 1** | Introduction to Software Testing |  |
| **Author** | Paul Ammann (Author), Jeff Offutt (Author) |
| **Publisher** | Cambridge University Press 2 edition 2016 |
| **ISBN** | ISBN-10: 9781107172012  ISBN-13: 978-1107172012 |

|  |  |  |
| --- | --- | --- |
| **Text book 2** | More Agile Testing |  |
| **Author** | Janet Gregory, Lisa Crispin |
| **Publisher** | Addison-Wesley Professional, @2015 |
| **ISBN** | ISBN-13: 9780133749564 |

|  |  |  |
| --- | --- | --- |
| **Reference book 1** | Agile Testing Foundations: An ISTQB Foundation Level Agile Tester guide | C:\Users\YWang\Pictures\360h.jpg |
| **Author** | Rex Black |
| **Publisher** | BCS Learning & Development Limited 2017 |
| **ISBN** | ISBN: 9781780173368 |

|  |  |  |
| --- | --- | --- |
| **Reference book 2** | Test-Driven Development for Embedded C | 51W6W+UOptL__SX415_BO1,204,203,200_ |
| **Author** | James W. Grenning |
| **Publisher** | Pragmatic Bookshelf, 2011 |
| **ISBN** | ISBN-13: 978-1934356623  ISBN-10: 193435662X |

## Suggested References

* Foundations of Software Testing, by Aditya P. Mathur, Addison-Wesley Professional, @2014, ISBN ISBN-10: 8131794768, ISBN-13: 978-8131794760
* Test-Driven Java Development , by Alex Garcia, Packt Publishing, 2nd edition, 2018, ISBN-10: 1788836111, ISBN-13: 978-1788836111
* <https://www.astqb.org/istqb-foundation-level>
* IEEE Standard for Software and System Test Documentation. Institute of Electrical and Electronic Engineers, New York. IEEE Std 829-2008.
* Grindal, M., Offutt, J., and Andler, S. F. (2005). Combination testing strategies: A survey. Software Testing, Verification, and Reliability, 15(2):97–133, Wiley
* Mathur, A. P. (2014). Foundations of Software Testing. Addison-Wesley Professional, Indianapolis, IN, second edition.
* Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., Grenning, J., Highsmith, J., Hunt, A., Jeffries, R., Kern, J., Marick, B., Martin, R. C., Mellor, S., Schwaber, K., Sutherland, J., and Thomas, D. (2001). The agile manifesto. Online Report. <http://agilemanifesto.org>

# Weekly Topic (lecture and lab)

|  |  |
| --- | --- |
| **Week** | **Topic (lecture and lab)** |
| 1 | Course outline, classroom conduct, academic integrity, attendance, and grading policy. Intro to testing and overview of the test process between hardware and software.  Lab: Extract the testable components from an embedded system - Raspberry Pi, Arduino, Sensor, and embedded programming |
| 2 | Testing levels, fault, failure, error, validation, verification, and SDLC model  Lab: Design pulse sensor detection embedded software and write the test harness |
| 3 | Model-driven test design, steps, activities, examples, and technically challenging  Lab: Design different test cases for given faulty programs under a testing framework. |
| 4 | Test automation, unit testing frameworks, and components of a test case  Lab: Using an embedded device to mock the system for test – identifying components of the system, test plan and design, and test execution |
| 5 | Putting test first, TDD cycle, BDD cycle, dual target, and automated embedded system testing  Lab: Use test-driven design to add additional functionality and include a narrative describing each TDD test created |
| 6 | Coverage criteria, test requirements, test sets, and fault revealing  Lab: Testing notation, unit testing, and test programming skills (C++, Java, or FPGAs) |
| 7 | State transition and state machine model  Lab: Apply boundary value and state transition to generate test cases for given requirements |
| 8 | Midterm |
| 9 | Graph coverage - control flow criteria  Lab: Draw a control flow graph and design a test case |
| 10 | Input space partitioning and input domain model  Lab: Derive input space partitioning test inputs and selecting a finite subset of inputs |
| 11 | Test planning, phases, and roles in embedded systems teams  Lab: Test phases and roles in embedded systems teams |
| 12 | Evolution of Testing -testing in Agile vs. traditional. Agile core principles and values. Agile project inception  Lab: The Agile team approach, team formation, and user stories with Agile tools. |
| 13 | Agile testing for mobile and embedded systems, Agile techniques, exploratory testing and challenges  Final project: Agile testing in practice |
| 14 | Agile tools, testing methods, and continuous improvement  Final project work continued: Agile testing in practice |
| 15 | Final |

# Course Policies

**Grading:**

* Project 20 %
* Final Examination 25 %
* Lab Exercises, Lab Reports, and Lab Project: 30 %
* Test and Midterm: 15 %
* Assignments, Class and Group Participation: 10%
* Total: 100 %

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Score % | < 60 | 60-69.9 | 70-76.9 | 77-79.9 | 80-82.9 | 83-86.9 | 87-89.9 | 90-92.9 | 93-100 |
| Grade | F | D | C | C+ | B- | B | B+ | A- | A |

**Blackboard and OpenLab:**

* Blackboard and OpenLab web sites will be used extensively to provide course material, collect assignments and reports and provide detailed grading information. Students must make sure their Blackboard login is working in the beginning of the course.

**Reading and Reference Material:**

* Reading and reference material will be provided in Blackboard and OpenLab.

**Software for Lab / Project:**

* Software applications will be used in the labs and course project. The links to download the software will be posted in Blackboard.

**Lab Reports:**

* Lab reports must be submitted
* Lab reports must be in Doc or PDF file format.
* All lab reports must be submitted through Blackboard – Assignments section.
* Each lab report is due one week after the lab work is performed.
* Any late-due lab report will have late penalty one week after due date.
* Additional requirements on lab report’s content and format will be posted on Blackboard or OpenLab. Please read them carefully.

**Library Usage:**

* Students are encouraged to use the library for supplementary resources in support of the lectures and labs.

**Classroom Conduct Policy:**

* Cell phone ringing and any other distracting and disruptive behavior such as talking loudly without permission are absolutely prohibited and may cause the student to be expelled from class.
* Any activity that threatens the college academic integrity will result in a disciplinary action.
* Please refer to the Student Handbook and the Catalog of New York City College of Technology for a full listing of Student Code of Conduct, Classroom Behavior Guidelines and Academic Integrity Rules.

**Academic Integrity Policy:**

Students and all others who work with information, ideas, texts, images, music, inventions, and other intellectual property owe their audience and sources accuracy and honesty in using, crediting, and citing sources. As a community of intellectual and professional workers, the College recognizes its responsibility for providing instruction in information literacy and academic integrity, offering models of good practice, and responding vigilantly and appropriately to infractions of academic integrity. Accordingly, academic dishonesty is prohibited in The City University of New York and at New York City College of Technology and is punishable by penalties, including failing grades, suspension, and expulsion.

**Attendance:**

* Under CityTech Policy, attendance in each class is required.
* At the beginning of each class, the instructor will make a roll call of all the student names to check the attendance.
* Any lateness must be reported to the instructor by the students before the class is dismissed.
* A name without on-time attendance nor reported lateness will be considered as an absence.
* Two lateness will be considered equal to one absence.
* Any absence due to emergencies (e.g., emergency medical condition or no-fault legal crisis) needs to be notified to the instructor by email or in-person. Excused absences can ONLY be considered with signed explanatory notes from proper party with a proper authority

# Course Design and Assessment

This Agile testing course is designed to meet the needs of CET students for the technical electives. The course is offered in lecture and lab combined. The lecture focus on the theory and principles of testing embedded software. The lab and cases provide students with the opportunity to apply the theory in practical ways to design and implement tests.

This proposed course will provide students with more options to make a choice to take 4 credits technical electives in CEB program. The course directly meets the CEB program educational objective. This course will be offered to CET B.Tech program. CEB junior or higher level students can take this 4 credits course as a technical elective. It is designed to implement basic knowledge of the major. The course is expected to offer in Fall 2019, once every academic year. Existing CET faculty are eligible to teach this course. The current lab facility is sufficient to run hands-on lab components and project. Students have an option to get the career booster of software testing –ISTQB Foundation Level software testing certification (CTFL) after taking this course. This certification gives students a common testing vocabulary and principles and opens a software and embedded testing career path for our CET students.

## Assessment Criteria

|  |  |
| --- | --- |
| **Upon successful completion of this course, students shall be able to** | **Evaluation methods and criteria** |
| Demonstrate knowledge of fundamental testing principles and processes associated with embedded systems. | Analysis of student performance on multiple choice questions, short answer questions on quizzes, homework, lab reports, project proposal and final project presentation. |
| Demonstrate understanding handling of the differences between testing in traditional and agile approaches | Analysis of student performance on multiple choice questions, short answer questions on quizzes, exams, and labs. |
| Demonstrate understanding Agile core principles and values | Analysis of student performance on multiple choice questions, short answer questions on quizzes, final exam, and the role in each lab within a group. |
| Design embedded system tests and test software in structured and organized ways | Students will demonstrate their ability to use test-driven design to write and implement additional functionality incrementally in labs and project. |
| Apply Agile software principles into practice and procedure | Agile is covered so students are able to form Agile teams in lab and understand each member’s role in the team. Students will complete project assignment in a groups. |

## General Education Outcomes and Assessment:

|  |  |
| --- | --- |
| **Learning Outcomes** | **Assessment Method** |
| SKILLS/Inquiry/Analysis: Students will employ scientific reasoning and logical thinking. | Students will identify embedded system fault and failure by using correct input harness and generating difference test cases for given system requirements. |
| SKILLS/Communication: Students will communicate in diverse settings and groups, using written (both reading and writing), oral (both speaking and listening), and visual means | In final project, students will present Agile testing practice project and how it was implemented using Agile testing principle. Student will demonstrate oral presentation skill through Power Point presentations. The formal lab report and formal final report in group will be used to assess communication skill in reading and writing and collaboration. |
| INTEGRATION/Integrate learning: Students will resolve difficult issues creatively by employing multiple systems and tools. | The agile community has developed several processes and tools to manage automating system tests and integrating the implementation as it is developed. Student works in a group on TDD repeating the cycle of small steps to resolve a technical issue. |
| VALUES, ETHICS, RELATIONSHIPS / Professional/Personal Development: Students will work with teams, including those of diverse composition. Build consensus. Respect and use creativity. | students will work in Agile team, build consensus and respect and use creativity in the final project |

# LIBRARY RESOURCES & INFORMATION LITERACY: MAJOR CURRICULUM MODIFICATION

Please complete for **all** major curriculum modifications. This information will assist the library in planning for new courses/programs.

Consult with your library faculty subject specialist (<http://cityte.ch/dir>) **3 weeks before the proposal deadline**.

**Course proposer:** please complete boxes 1-4. **Library faculty subject specialist:** please complete box 5.

|  |  |  |
| --- | --- | --- |
| **1** | **Title of proposal**  CET 4915 Agile Testing of Embedded Software | **Department/Program**  Computer Engineering Technology (CET) / BTech in Computer Engineering Technology |
|  | **Proposed by** (include email & phone)  Prof. Yu Wang  [ywang@citytech.cuny.edu](mailto:ywang@citytech.cuny.edu) / 718-260-5885  Prof. Sunghoon Jang  [SJang@citytech.cuny.edu](mailto:SJang@citytech.cuny.edu) /718-260-5885  Prof. Farrukh Zia  [fzia@citytech.cuny.edu](mailto:fzia@citytech.cuny.edu) / 718-260-5885 | **Expected date course(s) will be offered**  Fall 2019  **# of students**  22 |

|  |  |
| --- | --- |
| **2** | **The library cannot purchase reserve textbooks for every course at the college, nor copies for all students. Consult our website (**[**http://cityte.ch/curriculum**](http://cityte.ch/curriculum)**) for articles and ebooks for your courses, or our open educational resources (OER) guide (**[**http://cityte.ch/oer**](http://cityte.ch/oer)**). Have you considered using a freely-available OER or an open textbook in this course?**  Yes |

|  |  |
| --- | --- |
| **3** | **Beyond the required course materials, are City Tech library resources sufficient for course assignments? If additional resources are needed, please provide format details (e.g. ebook, journal, DVD, etc.), full citation (author, title, publisher, edition, date), price, and product link.**  No additional resources are required beyond the required course materials.  Sufficient number of papers, articles, journals on this subject are available in the library. |

|  |  |
| --- | --- |
| **4** | **Library faculty focus on strengthening students' information literacy skills in finding, critically evaluating, and ethically using information. We collaborate on developing assignments and customized instruction and research guides. When this course is offered, how do you plan to consult with the library faculty subject specialist for your area? Please elaborate.**  Once this course is offered, CET dept faculty will teach the course. We may consult with the library faculty specialist on the resources for the updated teaching reference materials, journal, or online resources. |

|  |  |
| --- | --- |
| **5** | **Library Faculty Subject Specialist:**  Junior Tidal (Email: jtidal@citytech.cuny.edu  **Comments and Recommendations**  After surveying the collection, I suggest not only acquiring texts outlined in the syllabus for this course but additional supplemental materials as well. The library has no print monographs related to agile testing. There are numerous eBooks related to conference proceedings related to the topic, however, these books may not be suitable as introductory texts for students. Software testing texts are also available in the library, but I also recommend updating the collection to more recent editions to fully support the course.  **Date 10.11.2018** |

# Chancellor’s Report

**Section AIV: New Courses**

**AIV.1. Department:** Computer Engineering Technology

**New courses to be offered in the Computer Engineering Technology department**

|  |  |
| --- | --- |
| **Department(s)** | Computer Engineering Technology |
| **Academic Level** | **[ x  ] Regular  [   ] Compensatory  [   ] Developmental  [   ] Remedial** |
| **Subject Area** | Computer Engineering |
| **Course Prefix** | CET |
| **Course Number** | 4915 |
| **Course Title** | Agile Testing of Embedded Software |
| **Catalog Description** | Introduction to the theory and principles of testing embedded software. The topics cover the basics of test design, test automation, and test-driven development, agile principles, values, techniques, practices, procedures, and tools. The lab and cases provide students with the opportunity to apply the theory in practical ways to design tests during the phases of embedded software development. |
| **Prerequisite** | CET 3510 |
| **Corequisite** |  |
| **Pre- or corequisite** | CET3640 |
| **Credits** | 4 |
| **Contact Hours** | 3 Class Hours, 3 Lab Hours |
| **Liberal Arts** | **[ ] Yes  [ x  ] No** |
| **Course Attribute (e.g. Writing Intensive, etc)** | N.A |
| **Course Applicability** | |  |  |  | | --- | --- | --- | | **[x ] Major** |  | | | **[ ] Gen Ed Required** | **[ ] Gen Ed – Flexible** | **[ ] Gen Ed - College Option** | | **[ ] English Composition** | **[ ] World Cultures** | **[ ] Speech** | | **[ ] Mathematics** | **[ ] US Experience in its Diversity** | **[ ] Interdisciplinary** | | **[ ] Science** | **[ ] Creative Expression** | **[ ] Advanced Liberal Arts** | |  | **[ ] Individual and Society** |  | |  | **[ ] Scientific World** |  | |
| **Effective Term** | Fall 2019 |

**Rationale:** This course is designed to meet the needs of CET students for a technical elective in the Computer Engineering Technology BTech program. This course allows students to develop their knowledge further to design tests in the phases of embedded software development, including hardware, electronics, and software. An understanding of Agile allows our CET students to get entry point in tech fields.

# Appendix A: Sample lecture and lab

Course should focus on using testing to improve embedded systems development, then add agile to show how to further improve it.

## Sample Lecture: Model-driven test design

The role of testing in software development has undergone radical changes in recent years. Testing has evolved from afterthought to a central activity in certain development methods—particularly agile methods

Outline:

1. Fault and Failure model (RIPR model)
2. Test activities – “V” model
3. Agile testing methods and Incremental TDD (Test driven development)
4. Discussion
5. RIPR model

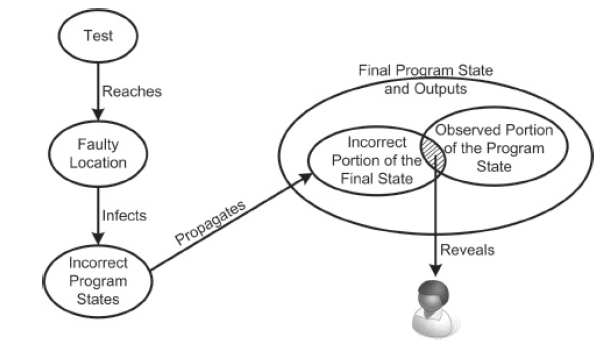
Reachability (R) -a test must reach the location or locations in the program that contain the fault.

Infection (I) - after the location is executed, the state of the program must be incorrect.

Propagation (P) - the infected state must propagate through the rest of the execution and cause some output or final state of the program to be incorrect

Revealability (R) - Finally, the tester must observe part of the incorrect portion of the final program state.

If the tester only observes parts of the correct portion of the final program state, the failure is not revealed.



Reachability, Infection, Propagation, Revealability (RIPR) model.

1. Activities of test engineers and “V” model

The most time-consuming parts of testing are actually the test design and construction, so test activities can and should be carried out throughout development.

Tests can be derived from requirements and specifications, design artifacts, or the source code. In traditional texts, a different level of testing accompanies each distinct software development activity:

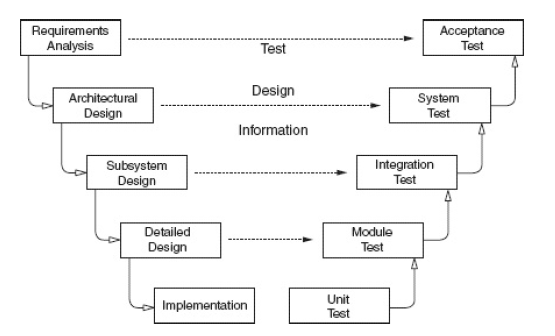
Acceptance Testing - assess software with respect to requirements or users’ needs.

System Testing - assess software with respect to architectural design and overall behavior.

Integration Testing - assess software with respect to subsystem design.

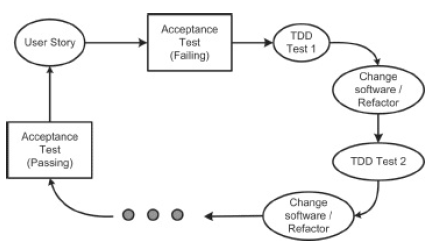
Module Testing - assess software with respect to detailed design.

Unit Testing - assess software with respect to implementation.



The “V Model”

1. The agile community has developed several processes and tools to manage automating system tests and integrating the implementation as it is developed. The core of TDD (test driven development) is a repeating cycle of small steps:
2. Add a small test
3. Run all the test and see the new one fail
4. Make small changes needed to pass the test
5. Run all the tests and see the new one pass
6. Refactor for remove duplication and improve expresiveness



The role of user stories and TDD cycle

1. Discussion
2. How are faults and failures related to testing and debugging
3. If you have never worked for a software development company, which of the four test activities do you think you are best qualified for? (test design, automation, execution, evaluation)

## **Sample Lab**: Extract the testable components from an embedded system

This mediocre diagram here shows how someone may use software testing in an embedded system:



Students need to be able to extract these testable components from an embedded system. Lab focus on getting a basic understanding of that. This is how programming should be approached. For example, in this situation, lab activity needs to:

1. write a function to generate random temperature
2. write a function to read temperature, and output string in format “%D degrees Fahrenheit”
3. write function to test that temperature is valid (between -89.2c and 56.7c)
4. write function to test that temperature is converted correctly
5. write function to test that string is formatted correctly

# Appendix B: Department Meeting Minutes

**2018-09-27 Faculty Meeting Minutes**

Attendance (excused are in **bold**): Sunghoon Jang, Robert Armstrong, Seymour Blank, Aparicio Carranza, Ohbong Kwon, Xiaohai Li, Lili Ma, Benito Mendoza, Edward Morton, John Razukas, José M. Reyes Álamo, Yu Wang, Chen Xu, Farrukh Zia, Natalia Maldonado.

Started at 12:49 PM

1. Approval of the minutes of the previous departmental meetings.
   1. Minutes approved.
2. Chair’s statement and report of the current developments in the department and college.
   1. More than 10 adjuncts has been hired
   2. Department has courses 7 days a week
   3. Report on students financial aid by college and finances
   4. Email update about the teaching load reductions in the coming years and how the policy may be implemented
   5. Tech Fee update on the items approved and the one rejected. The Department got most of the items requested.
   6. AIR – Annual Report Summary was discussed. It show the number of students in each programs, the trends, and some demographic information
   7. Middle States visit – They made several recommendations, but no major issues were found and the College in general is in good shape.
   8. Due to our full week schedule, the Chair requested a COA for the weekend. Email of the discussion with the administration were shown.
   9. Some candidates are having problems getting or changing their visas due to long wait times these days.
   10. Welcome to Prof. Maldonado who was recently hired and will be teaching in our Department. Prof. Maldonado introduced.
   11. College is accepting applications for promotions, please notify the Chair.
3. Meeting with Dr. Aiman Erbad – Qatar University
   1. Professors representing the government of Qatar and in town for the United Nations General Assembly visited the Department and shared remarks and their background.
4. New Faculty Search Committee Report and new full-time subs for Fall 2018
   1. Full-time Candidates are being considered
5. CET Department Reappointment Committee’s Meeting Schedule.
   1. The Committee will set the meeting time and date soon based on the schedule of all the members
6. Peer class observation for Fall 2018 and MCF for Spring 2019.
   1. Lists and schedules will be distributed later as the Department has several new hires
7. Curriculum and academic/research grant proposal update
   1. Lab manual for CET 3510 was showcased. Faculty are encouraged to provide similar material to students.
   2. Prof Wang showed the proposal for the course Agile Testing of Embedded Systems
   3. Prof. Wang also reported on the new forms being used to keep track of the courses
8. Report scholarly/creative works during summer
   1. Faculty are advised to update their profiles on the College website and contact Prof. Mendoza to do so.
9. ABET assessment of CET program by Profs. Zia and Mendoza
   1. Prof. Zia reported some of the changes in the evaluation report.
   2. The Department should be aware that the 2020 report should use the new framework and the past report has to be remapped into the new one.
10. Open House on Oct. 28, Sunday
    1. Prof. Li will provide more information about the event.
11. Updated list of the IAC members by Prof. Razukas and Kwon
    1. The final list of members will be submitted soon.
12. Uniform finals
    1. The courses that require uniform finals were discussed

Meeting Adjourned at 2:03 PM

Scribe: José M. Reyes Álamo