CS411 Course Syllabus (Dr. Sarah Hosni)

1 Basic Course Information

1.1 Catalog Description

Laboratory work required. Students write professional and non-technical documents and continue the development of the project defined in CS 410. Written work is reviewed and returned for corrective rewriting. Students will design and develop a project prototype, and demonstrate the prototype to a formal panel along with delivering the formal product specifications and a draft formal grant proposal. This is a writing intensive course.

1.2 Overall Description

The Professional Workforce Development course sequence (CS 410 and CS 411W) expands upon the experimental and design approach of typical computer science curricula by addressing the creativity and productivity required for business and industrial applications. Students in this course sequence engage in projects that investigate each stage of transforming a creative idea into an innovative product.

Writing can comprise as much as 60% of a computer scientist's professional activities. The evolution of software engineering into a team-centric process (possibly across geographic distances) requires the sharing of a wide variety of documentation with attention to version control and version management.

This course is the writing intensive (W) course for the major, which requires 51% or more of the computed grade be based upon writing assignments. You must demonstrate the ability to write clearly and accurately for a technically oriented audience. You will complete graded and ungraded writing exercises.

You will write professional and technical documents, and continue the development of the project defined in CS 410. Written work is reviewed and returned for corrective rewriting.

Students will design and develop a project prototype, and formally demonstrate the prototype.

1.3 Prerequisites

Students must have earned a grade of C or better in one (i.e., any) of the following

- ENGL 211C
- ENGL 221C
- ENGL 231C

and a grade of C or better in all three of the following

- CS 330
- CS 350
- CS 410

1.3.1 General Programming Knowledge

Students should be familiar with certain basic programming techniques that are largely independent of any specific programming language:

- using editors, compilers and other basic software development tools.
- basic software design (i.e., stepwise refinement and top-down design)
- software testing, including the use of scaffolding code (stubs and drivers), selection of test cases for black-box testing, and head to head testing.
- debugging, including the use of debugging output, the use of automatic debuggers to set breakpoints and trace program execution, and the general process of reasoning backwards from failure locations to the faulty code responsible for the failure.

1.3.2 C++ & Java Knowledge

I will assume that you are familiar with the basics of C++ or Java. This includes:

- the various C++ statements and control-flow constructs,
- the built-in data types,
- the use of arrays, pointers, pointers to arrays, and linked lists,
- the use and writing of functions, and
- the basic use of structs and/or classes for implementing abstract data types.

1.3.3 Python 3 & Rust Knowledge

Prior knowledge of Python and Rust is neither expected nor assumed.

In general, CS students at the 400 level should be able to pick up new programming languages with only moderate effort. If you need resources to get started with Python… take some time to work through the CS 263 materials

1.3.4 Unix/Linux

All students in the course will receive accounts on the CS Dept. network, and knowledge of how to work with the Linux servers is part of the course prerequisites. This course does not require familiarity with shell scripting. All other topics in CS 252 are required.

Some assignments will require the use of software available only on the Linux servers.

Others may require (or, at least, be simplified by) use of the X windowing system.

1.3.5 General Computer Literacy

You will be studying techniques in this course for preparing professional-quality software documentation. The key embedded word in “software documentation” is “document”. Students taking this course should be able to use word processors and other common tools to produce good quality documents, including mixing text and graphics in a natural and professional manner.

1.4 When and Where

Scheduled Recitation Meetings will be held via Zoom. Meeting information can be found in Canvas under *Course Collaboration Tool.* Recordings of these meetings will be available under *Media Gallery.*

1.5 Instructor

General Information

Instructor	Sarah Hosni
Office	Dragas 1100G
Email	shosni@odu.edu

Office Hours	Tuesdays and Thursdays 11:30-12:30 PM My office hours are typically 15-30 minutes Zoom meetings. *Please email me (shosni@odu.edu) to schedule a time slot when needed.*

Dr. Sarah Hosni is a lecturer in the Computer Science Department at Old dominion University. She earned a master's degree in computer sciences from Ain Shams University, Cairo, Egypt in 2010, and a Ph.D. in Electrical Engineering (Biomedical Engineering Program) from University of Rhode Island in 2021. She has more than 10 years' experience in teaching Computer Science courses spanning many knowledge areas including Software Development Fundamentals, Algorithms and Complexity, Discrete Structures, Computational Neuroscience, and Professional Workforce Development.

Dr. Sarah is also an interdisciplinary researcher, a neurotechnologist, neural engineer, and an entrepreneur. Her research interests are in brain-computer interfaces, computational neuroscience, and the application of machine learning in neurotechnology for healthcare. Her research has been published in some of the most renowned journals like Neuroinformatics, Computers in Biology and Medicine, and the Journal of Neural Engineering.

1.5.1 Contact Policy

All email related to this course should have the phrase **CS 410** in the subject line. This flags your message in my mailbox for faster attention.

Email is my favorite mode of communication with students. Don’t hesitate to email me your questions, I would be happy to provide guidance when needed.

I will respond to emails within 24 hours on weekdays, 48 hours on Weekends and holidays. For any course-related communication, please use your ODU email, as I will not respond to non-ODU emails.

1.5.2 Office Hours

I plan to use Zoom for my office hours twice a week.

Tuesdays: 12:00 -01:00 PM

Thursdays: 11:30 AM-12:30 PM

My office hours are typically 15-30 minutes Zoom meetings. *Please email me (shosni@odu.edu) to schedule a time slot when needed.*

2 Required Materials

2.1 Textbooks

There is no required textbook. All readings will be provided through Canvas and the course site.

2.2 Supplemental Materials

All supplemental resources and materials will be available in Canvas.

2.3 Technology Requirements

2.3.1 Computer Accounts

Students will need an account on the CS Dept. Linux network to participate in this class. This account is unrelated to any University-wide account you may have from ODU's Information Technology Services (ITS).

If you have had a CS Unix account in the recent past, it should still be active with your login name, password, and files unchanged. If you have had an account but forgot your password, or think your password is no longer active, go to Sign in to CS Account Management (odu.edu) . If you have tried all that and it has not been restored, contact the CS Dept systems staff at root@cs.odu.edu requesting that it be restored.

If you do not yet have such an account, go to Unix Account - Systems Group (odu.edu) and follow the instructions under "**Creating Account**". All students in this course are responsible for making sure they have a working CS Unix account prior to the first assignment.

3 Course Objectives

At the end of this course, students will be able to:

1. Recognize the value of collaboration in the field of computer science.
2. Plan and execute the development of a major software project.
3. Identify the order in which tasks should be completed either in parallel or in sequence in completing large projects.
4. Establish communication strategies to stay on task, complete team-based deliverables, and conduct effective team meetings asynchronously.
5. Establish effective asynchronous and synchronous communications with faculty and industry mentors.
6. Prepare collaborative outlines of the required written Assignments.
7. Prepare quality written papers based upon the collaborative outlines.
8. Document individual progress, challenges, concerns, and successes.
9. Recognize the need for the four types of professional writing addressed in the course as professionals and academics.
10. Utilize the assignments to develop high quality group projects.
11. Recognize the value of prototyping in the development of software projects.

4 Course Schedule

The course schedule is available in Canvas under *Modules* and *Calendar*.

5 Grading

Graded Element	Description	Weight	Individual/Group
Writing & Discussion Exercises	Discussion Activities	5%	(Individual)
Lab 1	Descriptive/Persuasive Paper	15%	(Individual)
Lab 2	Prototype Product Specification	15%	(Individual/Group)
Lab 3 & Lab 4	Prototype Test Plan & User Manual	15%	(Individual/Group)
Prototype Setup & Evaluations	Product Laboratory Prototype	5%	(Individual/Group)
Prototype Development	Product Laboratory Prototype	35%	(Individual/Group)

Website	Product and Course Documents	5% (Group)
Project Poster	Project Synopsis	5% (Group)

Percent	Letter Grade	4pt Value
≥ 94	A	4.0
≥ 90	A-	3.7
≥ 87	B+	3.3
≥ 84	B	3.0
≥ 80	B-	2.7
≥ 77	C+	2.3
≥ 74	C	2.0
≥ 70	C-	1.7
≥ 67	D+	1.3
≥ 64	D	1.0
≥ 60	D-	0.7
< 59	F	0.0

5.1 Presentations and Discussion Activities

Throughout the semester, several **discussion and presentation activities** will be organized to ensure continuous engagement and progress on your capstone projects. These will include **monthly informal presentations** where you'll showcase your development progress to me. These sessions are crucial as they provide opportunities for feedback, help you refine your project incrementally, and prepare you thoroughly for the **formal prototype presentation** at the end of the course.

In addition to these presentations, required discussion board activities will be required. Details for both presentations and discussions will be provided during recitation and through Canvas Announcements. These activities serve a dual purpose: they allow for comprehensive evaluation of each team's project as a whole and assess each student’s understanding of and contribution to the project. Participation in these discussions and presentations is key for adjusting individual grades.

The semester will conclude with a formal presentation of your Final Prototype, attended by external evaluators and mentors. This final assessment will not showcase the development effort in your prototype but also your ability to effectively communicate your work to an audience of professionals.

6 Course Policies

6.1 Attendance

Recitation attendance is not required. It is your responsibility to

1. watch the recording of any recitation that you are unable to attend
2. follow up with your team regarding any team feedback provided during recitation
3. contribute meaningfully to development of presentation, written, and prototype materials

Students must meet all course deadlines and be present for formal and unformal Prototype Demonstrations.

6.2 Due Dates and Late Submissions

Late papers, assignments, prototype contributions, and presentations/demonstrations will not normally be permitted.

Exceptions will be made only in situations of unusual and unforeseeable circumstances beyond the student's control, and such arrangements must be made prior to the due date in any situations where the conflict is foreseeable.

6.3 Group Membership (From CS 410 to CS 411W)

In CS 410 you were assigned to a team. Your team developed an idea and designed a Prototype. Barring specific conditions, you will remain with your team from CS 410.

You **will** be assigned to a new team if:

- It has been more than one semester since you completed CS 410.
- You are retaking CS 411W.
- You are completing this course (i.e., CS 411W) during the summer.

- Your original CS 410 team is no longer complete, either due to a reduced number of members or not having enough members to meet the course requirements.

You **may** be assigned to a new team, at the **instructor's discretion**, if you are unable to work with your CS 410 team. Note the use of *may*.

6.4 Group Contribution

You will recognize some prose from CS 350. Much of this section is based on the CS 350 and CS 410 syllabi.

You will work as part of a team for most of the semester (i.e., from week 2 onward).

Your grade will include (and be based on)…

1. **Active** participation within your team
2. **Active** contribution to your team
3. **Active** engagement with your team

Procrastination and just-before-the-deadline submissions are detrimental to any team dynamic, and will result in lowered grades.

6.5 Prototype Contribution

The Prototype (Software Solution) is a core deliverable of CS 411W. All students must contribute to development by

1. contributing to your group's efforts to learn selected tools and technologies.
2. contributing code **continuously** throughout the semester.
3. contributing **meaningful** code.
4. **communicating** with their group throughout the semester.

Any student who fails to effectively contribute to the Prototype will not pass the course.

6.6 Civility Among Team Members

You will recognize some prose CS 350. Much of this section is based on the CS 350 syllabus.

You will be working with your team for many weeks, and there will be a lot of communication expected among team members.

In accordance with the Monarch Creed and Code of Ethics, I expect all students to maintain *civility* in their dealings with one another.

Language that is abusive, harassing, or threatening to members of the class or that fosters high levels of personal and emotional anxiety may, at the instructor's discretion, result in expulsion from the team. Given the importance of the team project to this course, that is likely to result in a failing grade. Egregious or repeated violations will be referred to appropriate authorities for possible disciplinary action.

6.7 Unsatisfactory Performance

Your team will draft a *Team Contract* at the start of the semester (similar to CS 350 and CS 411W). Your team may elect to fire any member for failure to live up to the team contract.

6.7.1 Formal Warning

You will recognize the prose in this section from CS 350.

To issue a formal warning,

1. A majority of those team members who would remain, should the team member eventually be fired, must approve the warning.
2. A petition to warn the team member must be sent to the instructor and approved by the instructor.

```
This petition must document the reasons for the warning (e.g., what expectations from the Team Contract have been violated? What work has this person failed to do?) The petition need not be lengthy, but must document the reasons. The instructor will share these reasons with the person being warned (and it is only fair that they know why they are being warned).
```

Upon receipt of the petition, the instructor will make a decision on whether to issue the warning and will notify all persons involved of the warning.

6.7.2 Petition to Fire a Team Member

You will recognize the prose in this section from CS 350.

To fire a team member…

1. The team must have issued a formal warning to the team member at least one week before.
2. A majority of those team members who would remain must approve the firing.
3. A petition to fire the team member must be sent to the instructor and approved by the instructor.

```
This petition must document the reasons for the firing (e.g., what expectations from the Team Contract have been violated? What work has this person failed to do?)
```

```
The petition need not be lengthy, but must document the reasons. The instructor will share these reasons with the person being fired (and it is only fair that they know why they are being fired).
```

Upon receipt of the petition, the instructor will make a decision on whether to fire the team member and the date on which the firing will become effective.

6.7.3 Firing Aftermath

You will recognize most prose in this section from CS 350.

If a team member is fired, the team may remove their name from the access list to the team repository, tracking board, and shared documents. If the fired team member is the owner of the repository or board and therefore cannot be removed, then it will be the team's responsibility to make a fork or copy of the material in order to move forward.

The fired team member will retain their own copy of the team's repository and their knowledge of what was in any shared documents. Any disputes over access to formerly shared materials will be resolved by the instructor.

Due to the group nature of the course… the fired student will receive zeroes for any remaining group-based grades (e.g., Prototype Development).

6.8 Academic Honesty

Everything turned in for grading in this course must be your own work. If an assignment is **explicitly** described as a team assignment, it must be the work of the team members only.

The instructor reserves the right to question a student orally or in writing and to use his evaluation of the student's understanding of the assignment and of the submitted solution as evidence of cheating. Violations will be reported to the Office of Student Conduct & Academic Integrity for consideration for possible punitive action.

Students who contribute to violations by sharing their code/designs with others may be subject to the same penalties.

- Students are expected to use standard Unix protection mechanisms (`chmod`) to keep their assignments from being read by their classmates. Failure to do so will result in grade penalties, at the very least.

This policy is intended to prevent students from providing legitimate assistance to one another. Students are encouraged to seek/provide one another aid in learning to use the operating system, in issues pertaining to the programming language, or to general issues relating to the course subject matter.

Students should avoid, however, explicit discussion of approaches to solving a particular programming assignment, and under no circumstances should students show one another their code for an ongoing assignment, nor discuss such code in detail.

Policy on the Use of AI Tools in Course Assignments

1. General Usage of AI Tools:

- Students are permitted to use AI tools such as ChatGPT for enhancing productivity in coding and writing tasks. However, the fundamental expectation is that all submitted work—whether code or written text—must predominantly reflect the student's own understanding and originality.

2. Specific Guidelines for Coding:

- If AI tools are used to generate code, students must cite the specific tool and any external code sources directly within the code comments.

Just as when writing a paper, if you use someone else's ideas, you must cite your sources appropriately, if AI tools are used to generate code, students must cite the specific tool and any external code sources directly within the code comments.

- Example:

```
 /*...*/
  double x = 23.0;
  double xsqrt = sqrt(x);
  // Search algorithm based upon code by S Zeil at
  //
https://www.cs.odu.edu/~zeil/cs361/latest/Public/functionAnalysis/index.html#
orderedsequentialsearch
  int loc = 0;
  while (loc < arraySize && numbers[loc] < xsqrt)
  /*...*/Example of citing code generated by an AI tool:
```

Example:

```
  /*...*/
  double x = 23.0;
  double xsqrt = sqrt(x);
  // Search algorithm based upon code by S Zeil at
  //
https://www.cs.odu.edu/~zeil/cs361/latest/Public/functionAnalysis/index.html#
orderedsequentialsearch
  int loc = 0;
  while (loc < arraySize && numbers[loc] < xsqrt)
  /*...*/Example of citing code generated by an AI tool:
```

3. Specific Guidelines for Writing:

- The use of AI tools like ChatGPT in writing assignments is allowed for tasks such as drafting, revising, and style enhancement. However, all material generated by AI must be critically reviewed, edited, and integrated by the student to ensure it reflects their personal insight and academic rigor.
- Students must retain a precise copy of the prompt material supplied to the AI engine and be prepared to supply that prompt upon request by the instructor.
- Any direct use of text generated by AI tools must be cited appropriately in the references. Students may use the following citation format:
    - "This text was generated with the assistance of ChatGPT"

4. Integrity and Originality:

- Direct copying of AI-generated content without critical engagement or significant personal input will be scrutinized. Submissions perceived to lack original thought, or that heavily rely on AI without substantial student input, will not be accepted.
- Example of unacceptable use:
    - Submitting a lab1 report containing unedited, AI-generated text, even if cited, as it does not demonstrate the student's original work and understanding -- I developed an exceptional ability to identify those

5. Prohibitions:

- Posting details of course assignments, projects, or tests on online forums, bulletin boards, homework sites, etc., soliciting help is strictly prohibited.
- All work turned in for grading must be demonstrably your own. Failure to meet these standards, even with proper citation, may result in a grade of zero for the assignment if the instructor determines the work does not sufficiently demonstrate the student's knowledge and effort.
- The overall principle stated in the first sentence of this section remains in effect. “Everything turned in for grading in this course must be your own work.” If the bulk of your assignment, code, paper, etc., are copied, even with appropriate citation, to the degree that, in the judgment of the instructor, you have not demonstrated your own knowledge of the course material, you will receive a zero for that exercise.

7 University Policies

7.1 Code of Student Conduct and Academic Integrity

The Office of Student Conduct & Academic Integrity (OSCAI) oversees the administration of the student conduct system, as outlined in the Code of Student Conduct. Old Dominion University is committed to fostering an environment that is: safe and secure, inclusive, and conducive to academic integrity, student engagement, and student success. The University expects students and student organizations/groups to uphold and abide by standards included in the Code of Student Conduct. These standards are embodied within a set of core values that include personal and academic integrity, fairness, respect, community, and responsibility.

7.2 Honor Pledge

By attending Old Dominion University, you have accepted the responsibility to abide by the Honor Pledge:

I pledge to support the Honor System of Old Dominion University. I will refrain from any form of academic dishonesty or deception, such as cheating or plagiarism. I am aware that as a member of the academic community it is my responsibility to turn in all suspected violations of the Honor Code. I will report to a hearing if summoned.

7.3 University Email Policy

Reformatted to follow https://ww1.odu.edu/about/policiesandprocedures/computing/standards/11/02.

7.3.1 Student Email

With the increasing reliance and acceptance of electronic communication, email is considered an official means for University communication. Old Dominion University provides each student an email account for the purposes of teaching and learning, research, administration, and service. It is important that all students are aware of the expectations associated with email use as outlined in the Student Email Standard.

7.3.2 Email Account Activation

It is the responsibility of every eligible student to activate MIDAS, the Monarch Identification and Authorization System, in order to obtain email access.

7.3.3 Expectations Regarding Use of Email

The email account provided by the University is considered to be an official point of contact for correspondence. Students are expected to check their official e-mail account on a frequent and consistent basis in order to stay current with University communications. Mail sent to the ODU email address may include notification of University-related actions, including academic, financial, and disciplinary actions. For more information about student email, please visit Student Computing.

7.3.4 Educational Uses of Email

University offices and instructors cannot validate that a communication coming by email is from an ODU student unless it comes from a valid ODU email address. If students send mail from non-ODU email accounts (e.g., Hotmail or Yahoo), faculty and staff are not obligated to respond and may request that official e-mail accounts be used.

7.4 Withdrawal

Enrollment in this course indicates your acceptance of its teaching focus, requirements, and policies. Please review the syllabus and the course requirements as soon as possible. If you believe that the nature of this course does not meet your interests, needs or expectations, if you are not prepared for the amount of work involved – or if you anticipate assignment deadlines or adherence to course policies will constitute an unacceptable hardship for you – you should drop the course by the drop/add deadline, which is listed in the ODU Schedule of Classes. Visit the Office of the University Registrar for more information.

7.5 Educational Accessibility

Old Dominion University is committed to ensuring equal access to all qualified students with disabilities in accordance with the Americans with Disabilities Act. The Office of Educational Accessibility (OEA) is the campus office that works with students who have disabilities to provide and/or arrange reasonable accommodations.

- If you experience a disability which will impact your ability to access any aspect of my class, please present me with an accommodation letter from OEA so that we can work together to ensure that appropriate accommodations are available to you.
- If you feel that you will experience barriers to your ability to learn and/or testing in my class but do not have an accommodation letter, please consider scheduling an appointment with OEA to determine if academic accommodations are necessary.

The Office of Educational Accessibility is located at 1021 Student Success Center and their phone number is (757) 683-4655. Additional information is available at the OEA website