| DO NOT TYPE IN THIS BOX |
|---|
| Bulletin # : _____ |
| Academic Year : _____ |

# FLORIDA INTERNATIONAL UNIVERSITY
# UNIVERSITY CURRICULUM COMMITTEE
## *Proposal for a New Course*

1. **School/College** College of Engineering and Computing
   **Div./Dept. in Which Taught** Knight Foundation School of Computing and Information Sciences

2. | Alpha Prefix | 1st Digit | Last 3 Digits | "C"-lec-lab "L"-Lab | Cr. Hrs. |
   |---|---|---|---|---|
   | CIS | 3 | | | 0 |

   **CIP Code** ***(Leave this blank):*** _____

3. **Grading Method (select one):** ☐ Graded ☐ Pass/Fail

4a. **Course Title** Internship Ready Software Development

b. **Abbreviated course Title** ***(for computer class schedules, transcripts)*** Internship Ready SWDev

LIMITED TO 25 Characters *(including spaces)*

5. **Statewide Course Numbering Subject Matter Area** Computer Science and Information Systems

6. **Catalog Description/Major Topics** *(not to exceed 200 characters including spaces)*

*College of Medicine and College of Law: Attach description not exceeding 1,000 characters including spaces.*

Gain effective hard and soft skills for a successful software engineering internship. Topics covered include software version control, project management, open-source development, and AI.

7. **Attach detailed syllabus course outline and course justification on separate page(s).**
8. **Prerequisite(s):** COP 2210 or COP 2250 or equivalent
9. **Corequisite(s):** COP 3337 or COP 3804 or equivalent
10. **Objective(s) of Course:**

Gain effective hard and soft skills for a successful software engineering internship. Topics covered include software version control, project management, open-source development, and AI.

11. **Does this course duplicate/overlap other courses at FIU?** ■ No ☐ Yes
    **If yes, please explain:** _____
12. **What other closely related department(s) have been consulted about this course?**
    _____
13. **Is this course used for the assessment of a program or a certificate (if yes, then send a notification to assessment@fiu.edu)?** ☑ **No** ☐ **Yes**

**PROPOSAL REQUESTED BY:**

**Faculty Contact** Juan Mancilla-Caceres (Type name) _____ (Signature) 12 / 19 / 2023
jmancill@fiu.edu (Email address) 2177218610 (Phone number)

**Chairperson (Dept./Div.)** Jason Liu (Type name) _____ (Signature) 01 / 16 / 2024

**Chairperson (Curr. Comm.)** Alexander Afanasyev (Type name) _____ (Signature) ____ / ____ / 2024

**College/School Dean** Ines Triay (Type name) _____ (Signature) ____ / ____ / 2024

Submit one original form. Attach one copy of the course justification and a draft of the course syllabus for this New Course Proposal. *The complete syllabus should include all components listed on the New Course Checklist.*

# Internship Ready Software Development - Justification

This course will provide effective skill training for students to have a successful Software Engineering Internship. It will allow the student to gain experience and confidence using version control, agile project development, and developing open-source software in an active learning environment within nine weeks. The need for this comes from conversations with industry partners and a commitment from KFSCIS to provide tech career awareness, readiness, upskilling, and other wraparound services tailored to improve tech students' employment success and alumni & company engagement in career development activities. Feedback will be given to students in a managerial style, e.g., through code reviews and sprint retrospectives.

The course prepares students for their first internship and post-graduation employment by bridging knowledge gaps, building software development portfolios, strengthening soft skills, and increasing confidence. Specifically, the instructor and learning assistant will guide computing students in building open-source software portfolios, which are critical to impress recruiters and hiring managers and are considered an industry best practice for acquiring a software development job.

According to the Linux Foundation's 2022 Open Source Jobs Report, 93 percent of employers have difficulty hiring open-source talent. Growth in this area is projected to accelerate to at least 18.2 percent through 2026. Fig. 1 shows the growing importance of open source for employment in computer science.

Students will start and/or contribute to an Open Source software project as a team by (1) using Git and Github, (2) applying Scrum software project management techniques, (3) learning a core framework or platform such as artificial intelligence application programming interfaces as part of their project. Learning will be self-paced, peer group organized, learning assistant guided, and appropriate for their skill level —building a career development community support system so students can support each other. In Fig. 2, from the above-mentioned report, the hiring demand and perceived need for tech talent with open-source skills are highlighted. The survey conducted in March 2022 among 1,672 open-source professionals and 559 respondents responsible for hiring open-source professionals shows the significance of open-source skills in the tech industry.


Fig.1 Open Source Jobs Report

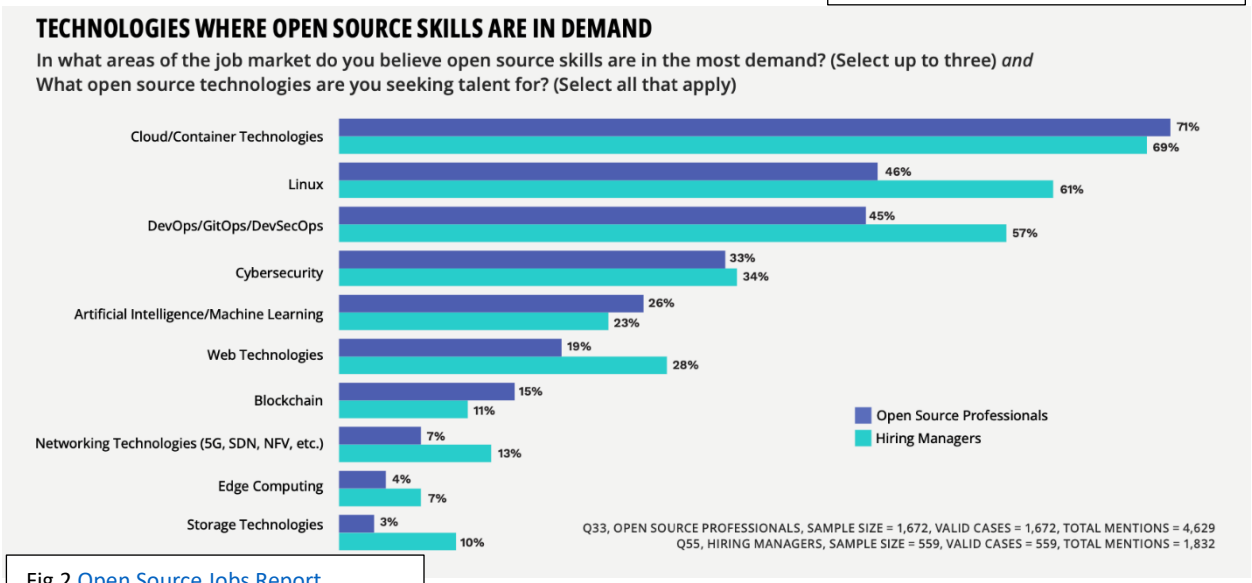


Fig.2 Open Source Jobs Report

## Knight Foundation School of Computing and Information Sciences

**Course Title:** Internship Ready Software Development **Date:** 1/1/2024

**Course Number:** CIS 3XXX

**Number of Credits:** 0

| | |
|---|---|
| **Subject Area:** XX | **Subject Area Coordinator:** XX<br>**email:** XX@fiu.edu |
| **Catalog Description:** Gain effective hard and soft skills for a successful software engineering internship. Topics covered include software version control, project management, open-source development, and AI. | |
| **Textbooks:** TBD | |
| **References:** TBD | |
| **Prerequisites Courses:** COP 2210 or COP 2250 or equivalent | |
| **Corequisite Courses:** COP 3337 or COP 3804 or equivalent | |

Type: Zero Credit Elective for CS, IT, Cybersecurity, and Data Science Majors

Prerequisites Topics:

1. Fundamental programming
    a. Objects, Classes, and fundamental Data Types
    b. Arrays, Strings, and Functions
2. Taking a course with Data Structures as co-requisite
3. Oral and written communication skills

Course Outcomes:

1. Apply the best practices for effective project management techniques (Applying).
2. Analyze various strategies and tools for developing a successful software development project (Analyzing).
3. Apply version control for software development (Applying).
4. Recognize the importance of version control in maintaining code (Remembering).
5. Evaluate the advantages and challenges of open-source software development (Evaluating).

**Knight Foundation School of Computing and Information Sciences**
**CIS 3XXX Internship Ready: Software Development**

## Relationship between Student Outcomes and Course Outcomes

| Student Outcomes | Course Outcomes |
|---|---|
| 1. Analyze a complex computing problem and apply principles of computing and other relevant disciplines to identify solutions. | 1,2,3 |
| 2. Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline. | 1,2,5 |
| 3. Communicate effectively in a variety of professional contexts. | 1,3,4 |
| 4. Recognize professional responsibilities and make informed judgments in computing practice based on legal and ethical principles. | 2,5 |
| 5. Function effectively as a member or leader of a team engaged in activities appropriate to the program's discipline. | 1,3,5 |

## Assessment Plan for the Course and how Data in the Course are used to assess Student Outcomes

| |
|---|
| Student and Instructor Course Outcome Surveys are administered at the conclusion of each offering, and are evaluated as described in the School's Assessment Plan: https://abet.cis.fiu.edu/. |

**Knight Foundation School of Computing and Information Sciences**
**CIS 3XXX Internship Ready: Software Development**

# Outline

| Topic | No. of Lecture Hours | Course Outcomes |
|---|---|---|
| **Version control with Git and GitHub** | | |
| • Describe why version control is a fundamental tool for coding and collaboration<br>• Install and run Git on your local machine<br>• Perform common tasks in GitHub.<br>• Collaborate with others through remote repositories<br>• Command-line interface / Integrations with various IDEs<br>Assessment:<br>• Managerial-style review of policies and feedback:<br>o Was a repository created?<br>o Does it follow guidelines?<br>o Is it secure?<br>o Is it properly described?<br>o Are the commits properly described? | 5 | 1,2,3 |
| **Open-Source Development** | | |
| • Define what open-source software is.<br>• Identify the benefits OSS has provided to the world's technology infrastructure.<br>• Describe collaboration best practices.<br>• Encourage skill diversity in teams working on open-source projects.<br>• Describe common methods, such as Continuous Integration and using GitHub and other hosting providers.<br>Assessment:<br>• Managerial-style review and progress report:<br>o Were relevant OSS properly identified and researched?<br>o Was a proper licensing selected?<br>o Were the policies of the selected OSS reviewed?<br>o What are the challenges and opportunities? | 5 | 1,2,4 |
| **Project Management** | | |
| • Adopt the 5 practices of Agile, a subset of DevOps: small batches, minimum viable product, pair programming, behavior- and test-driven development.<br>• Write good user stories, estimate, and assign story points and track stories using a Kanban board. Incorporate Scrum artifacts, events, and benefits.<br>• Create and refine a product backlog using the sprint planning process. Produce potentially shippable product increments with every iteration.<br>• Create burndown charts to forecast the ability to meet a sprint goal. Use metrics to enhance performance, productivity, and client satisfaction. | 10 | 1,2 |

| | | |
|---|---|---|
| • Describe the five important Scrum events and how to set up each event for a Scrum team.<br>Assessment:<br>• Managerial-style product review:<br>  ○ Review of User Stories<br>  ○ Were the sprints and daily stand ups set appropriately?<br>  ○ Are they using appropriate tools? | | |
| **Special Topics Reviews** | | |
| • Learn the general concepts of data visualization along with basic methodologies and applications.<br>• Implement Dashboards using open-source and proprietary tools.<br>• Identify the best patterns for data visualization of numerical and non-numerical data.<br>• Practice best strategies for communication of results.<br>• Give a pitch<br>Assessment:<br>• Managerial-style review:<br>  ○ Feedback on presentation<br>  ○ Was the audience properly identified and addressed?<br>  ○ Was there any area for improvement? | 2 | 1,2 |
| • Recognize the opportunities and challenges of the applications of Artificial Intelligence in software development.<br>• Evaluate Machine Learning Models.<br>• Using application programming interfaces (APIs) to transform data.<br>• Identify the best practices for the safe use of AI.<br>Assessment:<br>• Technology review:<br>  ○ Are all the risks and opportunities identified?<br>  ○ Are APIs used effectively?<br>  ○ Is there a plan for testing and deployment? | 3 | 1,2 |
| **Guided Development** | | |
| • Execute the proposed plan of a project.<br>• Implement the best practices of project management in a real-case scenario.<br>• Perform version control of a project using GitHub.<br>• Manage the scope of your project based on the current timeline.<br>Assessment:<br>• Code review:<br>  ○ Does the code follow guidelines?<br>  ○ Is it properly documented?<br>  ○ Has it been properly tested? | 15 | 1,2 |
| • Perform a product presentation to a general audience.<br>• Implement the best practices of technical communication | 5 | 1 |

| | | |
|---|---|---|
| Assessment:<br>• Start-up-style pitch review:<br>o Feedback on presentation<br>o Was the audience properly identified and addressed?<br>o Did the solution address all the requirements? | | |

## Performance Measures for Evaluation

| Assignment | Total Points | Percentage of Final Grade |
|---|---|---|
| Use Scrum / Agile to Manage your Project | 30 | 26% |
| Use an API and write code to implement features described in User Stories | 30 | 26% |
| Apply Version Control Effectively | 20 | 17% |
| Join an Open Source Project and Contribute to it | 20 | 17% |
| Final Reflection | 15 | 14% |
| | Total | 100% |

All assignments must be completed at least 80% score to gain the micro-credential badge.

## Letter Grade Distribution Table

| Letter | Range% | Letter | Range% | Letter | Range% |
|---|---|---|---|---|---|
| A | 93 or above | B | 82 – 85.9 | C | 70 – 73.9 |
| A- | 90 – 92.9 | B- | 78 – 81.9 | D | 60 – 69.9 |
| B+ | 86 – 89.9 | C+ | 74 – 77.9 | F | Less than 60 |

**Oral and Written Communication:**

The final presentation may include a poster, presentation, and / or written report.

**Social and Ethical Implications of Computing Topics:**

Proper use of Open-Source Licenses

**Theoretical Contents:**

No Significant Coverage

**Problem Analysis Experiences:**

No Significant Coverage