SAINT LOUIS UNIVERSITY.

Program-Level Assessment: Annual Report

Program Name (no acronyms): Computer Science	Department: Computer Science
Degree or Certificate Level: Bachelor of Arts	College/School: Arts and Sciences
Date (Month/Year): August 30, 2021	Assessment Contact: Outgoing: Jacob Sukhodolsky Incoming: David Ferry (david.ferry@slu.edu)

In what year was the data upon which this report is based collected? AY 2020-2021

In what year was the program's assessment plan most recently reviewed/updated? 2018

1. Student Learning Outcomes

Which of the program's student learning outcomes were assessed in this annual assessment cycle? (Please list the full, complete learning outcome statements and not just numbers, e.g., Outcomes 1 and 2.)

PLO 2 – Design and Implementation – Design, implement, evaluate and test a software system that meets a given set of computing requirements.

2. Assessment Methods: Artifacts of Student Learning

Which artifacts of student learning were used to determine if students achieved the outcome(s)? Please describe and identify the course(s) in which these artifacts were collected. Clarify if any such courses were offered a) online, b) at the Madrid campus, or c) at any other off-campus location.

In early Fall 2020 the department assessment committee met to initiate our first trial of the assessment procedure and formalize a rubric for PLO 2 . We selected courses that involve substantial student software development effort as trial candidates for the new assessment rubric.

CSCI 2300 – Object Oriented Software Design – This course is typically taken by sophomores and represents an evolution in the curriculum from small-scale programming to medium-scale software systems. This course reinforces mastery of basic programming skills, while software architecture strategies and software design principles are introduced. This course is normally taught in-person, but was entirely online in response to COVID-19 in FL20.

The instructor for 2300 selected a late-semester, long-term team programming project for assessment. In this assignment teams of 2-3 students were asked to create the game of Black Jack in Java using the SWING API for a graphic interface. The project was split into four phases. Students were asked to generate a list of functional requirements, a set of design documents, a rough implementation with a set of associated unit tests, and then a final application for submission.

CSCI 4961 – Capstone Project 1 – This course is typically taken as a first-semester senior and is a culminating software development experience for the CS degree. This course is normally taught in-person, but was entirely online in response to COVID-19 in FL20.

Teams of 2-3 students are given a client who has come to the department with a software project. Students are expected to meet with their client, collect software requirements, propose a software solution, and implement said software over a two-semester sequence. Typically, students are expected to have prototyped substantial portions of their software system by the end of Capstone 1, though specific expectations vary significantly between groups.

Individual team expectations are set at the start of Capstone 1 in collaboration with their faculty capstone supervisor and the capstone instructor.

The assessed artifact for CSCI 4961 is the project state at the end of the course (midway through the two semester sequence). This artifact can be considered to be all of the code and documentation the student teams have produced by the end of the semester, as well as the content of a summative presentation given before finals. The specific content of this artifact varies considerably between groups.

3. Assessment Methods: Evaluation Process

What process was used to evaluate the artifacts of student learning, and by whom? Please identify the tools(s) (e.g., a rubric) used in the process and **include them in/with this report document** (do not just refer to the assessment plan).

The instructors of the two courses were notified of the desire to collect assessment data early in the semester and a copy of the assessment rubric (submitted with this report) was provided. The instructors were asked to identify and assess all criterion (see rubric) they thought were relevant to their course and assessment artifact. The instructors were also provided two standard forms for documenting their assessment efforts and how they collected assessment data (also submitted with this report).

Students in CSCI 4961 were assessed at the end of the semester on five of the rubric criteria through a review of submitted project materials, based on which criteria the instructor thought were relevant.

Students in CSCI 2300 were assessed at the end of the project on four of the rubric criteria through a review of submitted project materials, based on which criteria the instructor thought were relevant..

4. Data/Results

What were the results of the assessment of the learning outcome(s)? Please be specific. Does achievement differ by teaching modality (e.g., online vs. face-to-face) or on-ground location (e.g., STL campus, Madrid campus, other off-campus site)?

A presentation of the data and then analysis follows. The assessment rubric defines five proficiency levels:

0 – Does not meet the basic standard or not attempted
1 – Beginning
2 – Developing
3 – Accomplished
4 – Exemplary

These proficiency levels are somewhat subject to interpretation. For rubric criteria primarily intended for undergraduate students (true of all the categories below) it is hoped that most graduating students would exhibit a score of 3, but reasonable that some weaker students would still be at a level of 2 and some exemplary students would be at a level of 4. This means that scores of 0-2 would be expected and normal for early classes in the curriculum, while scores of 2-4 would be expected and normal for later classes in the curriculum.

Data Summary, CSCI 2300:

	Design	Design	Design	Implementation	Implementation	Testing & Eval
	Software Arch	Requirements	Documentation	Code Quality	Code Correctness	Code Quality Assurance
AVERAGE:	2	*	*	2.4	2.7	1.5
MEDIAN:	2	*	*	2	3	2

*Not assessed

Data Summary, CSCI 4961

	Design Software Arch	Design Requirements	Design Documentation	Implementation Code Quality	Implementation Code Correctness	Testing & Eval Code Quality Assurance
AVERAGE:	1.2	2	0.9	1.4	*	1.3
MEDIAN:	1	2	1	2	*	1

* Not assessed

The scores attained in 2300 are reasonable and even promising given its place in the curriculum. The scores attained in 4961 are less good. Notably, among those criteria assessed in both courses (Design – Software Architecture, Implementation – Code Quality, and Testing & Eval – Code Quality Assurance) all three metrics dropped in score between 2300 (a typically sophomore course) and 4961 (a typically senior course).

However, there are several reasons why CSCI 4961 was found to be a poor assessment endpoint for PLO 2:

1. Capstone projects are open-ended, so the assessment rubric may not apply well to specific projects. For example, every project scored at least a 1 in Code Quality Assurance because every project was tested in at least an ad-hoc manner. However, only one project adopted systematic quality assurance mechanisms such as automatic unit testing. While other projects may have benefitted from such testing, only one project required it, so only one student group adopted it, and only one group scored higher than a 1 on this criteria. Similar arguments can be made for most of the other rubric criteria.
2. The assessment occurred at the end of CSCI 4961, which is halfway through the two-semester capstone sequence. The student expectation is to have an incomplete prototype project at this point, rather than a fleshed out and completed project.
3. Some of the work that goes into capstone is difficult to capture as a gradable artifact. For example, students scored poorly on Design – Documentation, and in most cases these groups had not submitted such documentation through the formal software tracking system. However, students use a variety of means to communicate these days (especially during remote learning due to COVID) such as Slack, Discord, email, Trello boards, and other such venues. There are several cases where a group scored a 1 due to lack of formal documentation in the tracking system, but the instructor knows through conversations and feedback that such design documentation does exist in an alternate venue, but is not available to the instructor.

Notably, the FL20 CSCI 4961 section scored a cumulative GPA of 3.68, and (qualitatively) 11 out of 13 groups entirely or mostly met their client's goals by the conclusion of this cohort's capstone projects in SP21 CSCI 4962. Thus, the assessment scores in CSCI 4961 are probably a reflection of a mismatch between the assessment process and the capstone course design.

5. Findings: Interpretations & Conclusions

What have you learned from these results? What does the data tell you?

Beyond the analysis of CSCI 4961 as an assessment endpoint given above, the basic finding of the assessment committee is that our assessment process needs revision. Several high-level criticisms of the current assessment process were identified:

1. The assessment scores, being based on individual assignments at the discretion of the individual instructors, are highly subjective. CSCI 2300 is a class that spends a lot of time talking about software design, software architecture, and software testing, so the artifact from CSCI 2300 is highly compatible with the assessment rubric. The artifacts from CSCI 4961 were not. It is not clear how this assessment data should be used to improve our curriculum.

2. The assessment scores may not be comparable over time. Allowing instructors to pick an their own artifacts for assessment may mean that different instructors will not assess the students in the same way from year to year, frustrating our desire to gather accurate longitudinal data.
3. The assessment artifacts used for AY20-21 were group projects. This does not allow us to really assess individual performance.
4. Ultimately it is not clear whether the data from AY20-21 is meaningful, or whether we should continue conducting assessment the same way in AY21-22.
5. The instructors involved reported that the time/effort burden of assessment was manageable, but it would be desirable to reduce this time burden.
6. The current assessment plan and set of PLOs adequately reflect desired academic outcomes but does not capture real-world performance of students after they graduate.

The committee also found specific benefits to the assessment process as currently implemented:

1. The task of going through the assessment process forces the faculty to think about what it is they are trying to achieve and how their assignments and class work correspond to that. Both faculty reported making modifications to their courses in SP21 as a result of the assessment process.
2. A variety of specific rubric improvements were identified in terms of wording and scaling of achievement levels.

6. Closing the Loop: Dissemination and Use of Current Assessment Findings

A. When and how did your program faculty share and discuss these results and findings from this cycle of assessment?

The AY20-21 data was discussed at the CS Department faculty retreat on August 18, 2021. The incoming Assessment Committee chair (David Ferry) and the outgoing assessment committee members reviewed the previous year's data via Zoom meeting the previous May, and the incoming chair produced a report for the faculty at large.

B. How specifically have you decided to use these findings to improve teaching and learning in your program? For example, perhaps you've initiated one or more of the following:

Changes to the Curriculum or Pedagogies	• Course content • Teaching techniques • Improvements in technology • Prerequisites	• Course sequence • New courses • Deletion of courses • Changes in frequency or scheduling of course offerings
Changes to the Assessment Plan	• Student learning outcomes • Artifacts of student learning • Evaluation process	• Evaluation tools (e.g., rubrics) • Data collection methods • Frequency of data collection

Please describe the actions you are taking as a result of these findings.

Although not a product of specific assessment findings, the CS faculty ratified, submitted, and had approved a major revision to our curriculum in AY20-21. Our two required Systems courses – CSCI 2400 Computer Architecture and CSCI 3500 Operating Systems – have been replaced with two new courses – CSCI 2500 Computer Organization and Systems and CSCI 2510 Principles of Computing Systems. These will be taught for the first time in AY22-23. Operating Systems will continue to be offered as an upper-level elective. This demonstrates continued diligence in keeping our curriculum relevant and current in a fast-moving field.

Second, the incoming assessment committee chair has been granted a course release with the goal of enacting the existing assessment plan as well as implementing improvements.

With respect to the assessment plan, the current plan has not generated assessment data that is obviously robust and meaningful. The current assessment plan will be continued for AY21-22, but the assessment committee will seek to revise the existing plan. Several revisions were proposed during the faculty retreat, and the committee will consider these as well as others.

In an effort to make assessment scores less subjective and more comparable over time, it was suggested that we pursue an assessment strategy that occurs outside of regular coursework. This could take the form of a standardized test or other extracurricular assessment at regular intervals, such as at the end of sophomore and senior year.

To make scores less subjective and more comparable over time, and also reduce faculty workload in assessment, it was suggested that we could identify automatic tools to help gather assessment data from existing coursework. There are tools such as code quality analyzers, linters, call graph analyzers, etc. that are specific to our field and could automatically produce quantitative scores relevant to the rubric.

To make scores less subjective and more comparable over time, it was suggested that we fix just one or two courses for each PLO so that each PLO would always be assessed at approximately the same place in our students' programs.

It was suggested that we could identify nonacademic measures of student success to incorporate in our assessment. These would include measures such as student job placement rates, alumni surveys to find what skills they find useful in their employment, external sources of achievement data to compare our students against, etc.

The next courses to be assessed under the current plan are offered primarily in the spring. The incoming assessment committee will spend the Fall semester considering proposed revisions and writing rubrics for the assessment of PLO 3 (Apply Theory/Knowledge) and PLO 5 (Informed Judgement). In the Spring the committee will ensure that assessments are conducted, collect data, and analyze the data.

If no changes are being made, please explain why.

7. Closing the Loop: Review of Previous Assessment Findings and Changes

A. What is at least one change your program has implemented in recent years as a result of assessment data?

Due to unexpected medical leave, the response to the COVID pandemic, and the retirement of the assessment committee chair, assessment data has not yet been collected for the CS-BA program before this year. See the 2020 assessment report for more details. As a result, there have not yet been any program level changes due to assessment data.

B. How has this change/have these changes been assessed?

N/A

C. What were the findings of the assessment?

N/A

D. How do you plan to (continue to) use this information moving forward?

N/A

IMPORTANT: Please submit any assessment tools (e.g., rubrics) with this report as separate attachments or copied and pasted into this Word document. Please do not just refer to the assessment plan; the report should serve as a stand-alone document.