CLIENT ORIENTATION DOCUMENT (C1)

0. About this Document

This document is intended to inform clients and potential clients about the Senior Assessment projects undertaken by student development teams in Computer Science senior project courses (CS425 and CS499) at SIUE. If you have any question or comment about the content of this document, please contact the instructor of the course (see last page). Suggestions for improving this document are most welcome.

Important Points:

- Projects are scheduled over **two consecutive full-semesters**
- The **client must be available to the team** for meetings
- There is **no promise to deliver any product**, partial product, or work product of any kind to the client
- Any product or material delivered is done so **without a warranty of any kind**, expressed or implied
- The client is responsible for **providing all hardware** resources required for the project

1. Overview of Senior Assessment (Senior Project) Projects

Computer Science (CS) Senior Assessment (senior project) is a two-semester course sequence in which teams of students undertake a project for an "outside" (i.e., not a CS faculty member) client. The person who proposes a project is the "client". The CS senior project course-sequence presents the students with the opportunity to practice and refine the skills, and use the knowledge, they have gained in Computer Science courses. The experience comes through the computing system development they undertake for their client. The experiences the Computer Science Department desires for the students include working with clients, team skills, and designing, building, and testing software systems.

Projects are scheduled for the two courses that comprise the Senior Assessment for Computer Science majors: CS 425 and 499. The same team will work with the same client during the two semesters. Projects may begin in August and complete in May (~9 months), or begin in January and complete in December (~12 months).

The project begins with Requirements Elicitation and ends with Acceptance Testing. In other words, teams will begin by working with the client to understand what the client needs, and finish by having the client determine if the team has delivered the needed product. Though the delivery of the required product is a focus of these courses, there are other course activities during the semesters the students must complete.

1.1. Limitations

Though this document uses terminology appropriate for software consulting firm, Computer Science Faculty and students are not a commercial entity. **Our sole focus is on the learning experience for the students**. Certain limitations arise from our situation and the focus on learning. The following list of limitations is not exhaustive.

1.1.1 No Guarantee of Delivery

Due to the limitations of the academic setting, the Computer Science Department, nor any individual student team does not guarantee that a team will be 100% successful. **There is no**

promise to deliver any product, partial product, or work product of any kind to the client. Unforeseen technical challenges or any number of other obstacles can hinder the progress of teams.

1.1.2 No Warranty / No Support following development

Any product or material delivered is done so without warranty of any kind, expressed or implied. No maintenance, bug fixes, enhancement(s), improvement(s), nor work of any kind, will be preformed following the project period. No computing service will be provided after the conclusion of the development period. Any hardware required for use of the product remains the responsibility of the user.

1.1.3 Limited Development Time

The academic setting limits the amount of time students should reasonably contribute to these projects. In addition, other events (e.g., exams, holidays, semester breaks, etc.) dictated by the academic calendar usually stop project progress. In particular, teams working on January-to-December projects are not expected to work during the summer term.

1.1.4 Intellectual Property

The project development team (i.e., students) and the client will share the intellectual property of the project and resultant product equally. Either party is capable of independently correcting, improving, or otherwise reusing any or all parts of the resultant intellectual property and product, unless otherwise agreed to in writing not later that the date the product specification document has been signed and accepted.

1.1.5 Computing Resources

The client will provide all hardware resources required for the project, including servers on which to host the developed software (if required). The client understands that neither the students nor the Computer Science Department have any responsibility to provide such hardware, software subscriptions, nor software licenses. In addition, if the project requires a software interface with an existing system, or requires the students to use software with license requirements, the client will bear the responsibility of ownership and cost for those licenses and systems. Furthermore, the client will assist the student team with contacting and organizing specific end-users of the developed software.

2. Project Function

2.1. Organizational Overview

Senior Assessment courses function largely like a small software-consulting firm. The instructor of the course oversees each of the teams and provides technical assistance as necessary. Ideally, the instructor of the course has nothing to do as the teams work with the clients to achieve the clients' goals. In reality, the instructor is frequently involved in technical aspects of the projects, as well as managing personnel of the teams.

2.2. Role of the Client

The client is the one person who knows what is needed in the product. Also, the client is the one person who will determine when the project is done (within reason). Thus, the client has an extremely important role in the project.

While the development team must respect the clients' schedule, the client must also be available to the team on as as-needed basis to allow the team to discuss requirements, review designs, and test prototypes. The client, or client's representative, must be available to the students and a regular team participant during the development. This includes regular face-to-face meetings with the student team (typically) once every-other week, end of semester presentations, and CS Department client surveys.

2.3. Client-Team Interactions

In most cases, the development team will contact and communicate with the client directly. The team needs direct communication for understanding the client's needs and demonstrating progress. When projects are progressing well, the client and the instructor of the course may have very little contact. The length and frequency of meetings can vary considerably from project to project and depends on many factors.

In addition to face-to-face meetings, the teams may want to communicate via phone and/or email. The team should show deference to the client's preferred modes of communications and respect the client's schedule.

2.4. Documentation

Teams should be documenting all of their work. The documentation takes many different forms including informal sketches of designs, UML diagrams, E-R diagrams, workflow diagrams, paper prototypes, internal and external code documentation, records of interviews, and formal written documents. (The list of required formal documents is given to the student teams.) Many of these documents will be of little value to the typical client. However, the client and team must find a way to document communication about the design of the user interface.

The students should show the client representations of user interface designs (i.e., windows, menus, buttons, dialog boxes, etc.) for client approval. The representation may be on paper or in the form of a (non-)functional prototype. The point in discussing the design of the user interface is to gain the client's feedback before investing more effort in the development. The user interface is more than just a part of the software; it represents the functionality of the product. In other words, the user interface is the medium through which the user performs tasks. The obvious consequence of this is that a task not present in the user interface is impossible to for users to perform.

3. Schedule

3.1. Project Schedule

As mentioned above, the projects are scheduled over two consecutive full-semesters. Generally speaking, teams that begin projects in Spring-semester and end in the Fall-semester do not work

on the project during the Summer term. More schedule details are given for each of the possible sequences below.

3.1.1. Fall-Spring Schedule

In Fall-Spring projects (Aug to May) the development teams should begin meeting with clients not later than early September. The goals of the first semester (Aug to Dec) are to fully understand the project scope, have a significant portion of the product design completed, and have built and tested some portion of the software system.

The goal of the second semester is to complete the product development to client satisfaction. Client satisfaction is determined by 'acceptance testing' in which the client (or client's designee) uses the product to determine if it meets all needs and is suitable for use. Teams strive to be ready for acceptance testing in early April, with final delivery of the product in late April or early May.

3.1.2. ~~Spring-Fall Schedule~~ (doesn't apply to this term)

~~In Spring-Fall projects (Jan to Dec) the development teams should begin meeting with clients not later than early February. The goals of the first semester are to fully understand the project scope, have a significant portion of the product design completed, and have built and tested some portion of the software system.~~

~~The goal of the second semester is to complete the product development to client satisfaction. Client satisfaction is determined by 'acceptance testing' in which the client (or client's designee) uses the product to determine if it meets all needs and is suitable for use. Teams strive to be ready for acceptance testing in November, with final delivery of the product in early December.~~

3.1.3 Development Team Presentations

Near the end of each semester (at the end of CS 425, and at the end of CS 499) development teams make presentations to the CS faculty and students. Clients are encouraged to attend. The teams should make their respective clients aware of the dates for presentation. Planned presentation dates are listed below.

4. Dealing with changes and the unexpected

Please contact the course instructor. (see below)

Fall 2022 / Spring 2023 specific information

Fall presentation date: November 18, 2022
Spring presentation date: to be determined

Instructor
Dr. Dennis J Bouvier
dbouvie@siue.edu

5. With all these risks, why propose?

After having read this document, and seeing **the client is expected to invest time and is guaranteed nothing in return**, why would you propose a project?

Here are the **possible benefits**:

- Proof of concept – you have an idea, but don't know if it will work
- Refinement of design – you have an idea for something, but don't have all the details worked out
- Exploration of technology – you don't have a specific project in mind, but you have a specific technology you want to explore … we can do that by creating a project to exercise the technology
- Working prototype – often, the project team delivers a partially working product
- (maybe, just maybe) a complete project – approximately 10% of the project teams in recent semesters have delivered complete working projects

6. Writing a proposal

The basic information needed to accept a project includes:

- **The client** (name, e-mail address, phone number, etc.) – can be multiple people
- **Brief background** or history (what's behind the need for the software?)
- **Project description** (what should the software do?)
- **Specific technical requirements**, if any (e.g., "must be in Java" or "must interface with this software we use")
- **Intellectual Property** rights - who will own intellectual property of the project at the completion of the students' work?

"ideas" that generally don't make good projects:

- Simple websites that contain only static information
- Games (especially ones in which sounds and/or art needs to be created)
- A commercial project (that is, the intention is to sell the resulting software)

Additionally:

- Frequently more projects are proposed than we can take – making a submission does not guarantee acceptance
- If you were a client of a failed project in the past, please resubmit and let a new team try the project
- If you proposed a project in the past that was not selected, feel free to resubmit
- If you were a client of a successful project in the past and want updates or changes, please be aware that if the changes are minor, we may not be able to take the project on. However, we might be able to create a new product from scratch with all the risks mentioned above
- Senior projects are not internships. Please do not propose a project that is an internship.
- It is unlikely we can accommodate projects in which you wish to retain all intellectual property rights

end of document