

Yaşar University
Faculty of Engineering
Software Engineering Department

Senior Project Guidelines

The purpose of this document is to provide guidelines for students on their work in courses SE 4910 Senior Project I and SE 4920 Senior Project II. Senior Project is a sequence of two required courses and takes two semesters to complete. This capstone sequence of courses culminates in the development and presentation of a significant project which puts the finishing touch on students' education in Software Engineering by allowing them to design and build a comprehensive solution to a real-world problem. Software Engineering students usually take it in their last two semesters of study.

A Senior Project is typically completed by a team of two students; however, when appropriate due to the specifics of the project topic and scope the project may be assigned to a team of three or more students or an individual student. Teams may collaborate with students from other disciplines to develop larger interdisciplinary design projects.

SENIOR PROJECT IN THE SOFTWARE ENGINEERING CURRICULUM

The ability to identify, define, research, and solve an issue, problem, or task in Software Engineering is a necessity for an individual engaged in professional practice. To develop this, students are required to prepare a scholarly report that examines, in detail, the issue or task and to provide a solution to the problem through the application of design principles or learned ideas and concepts. The subject of the report should be relevant to the theory, ideas, and practice of Software Engineering. In addition, the designed solution must be implemented as a working and tested device and/or software system. Finally, students are required to present their report and implementation to a group of Software Engineering faculty.

The Senior Project is the capstone of computing and engineering education in Software Engineering at Yasar University. It concludes the series of required core computing, design, and software engineering courses. The Senior Project builds upon previous courses in mathematics, basic sciences, humanities, social sciences, professional issues, and communication skills. Therefore, it has been structured to be the major meaningful engineering experience for students. Students develop knowledge, skills, and attitudes in the sequence of courses that help them to integrate previous course work and to successfully plan, control, and implement design projects.

Specific learning objectives include:

- identify the problem
- conduct background research
- analyze feasibility
- consider business and marketing issues
- develop requirements specifications
- recognize & articulate design constraints
- perform systems analysis
- generate and evaluate alternative concepts that meet specified functions and constraints

- perform and refine system and components design
- create efficient programming code
- address successfully issues of usability and user interface
- write professional system and program documentation
- complete testing, simulation, and verification activities as appropriate
- evaluate the performance of the developed solution and identify issues for further development and improvement
- identify & articulate issues with patents, legal liability, safety, and ethics
- effectively select and where appropriate procure equipment and software from suppliers
- verbally & graphically present and communicate information
- develop understanding and skills in project management
- learn & use effective teamwork skills
- utilize concepts, topics & resources developed in previous coursework
- utilize the library, laboratory, and computer hardware and software resources
- get experience in work with and develop professional relationships with an advisor and faculty members

WHAT KIND OF PROJECTS ARE SUITABLE?

Each project intends to provide the students with valuable research and practical development experience. A significant portion of each project should include a development component related to Software Engineering. This should include the design and implementation of systems, devices, algorithms, software, processes, or other relevant elements.

A Software Engineering senior project is normally a small-scale system and/or device involving both software design and implementation, and hardware design and construction. In certain cases, the aim of the project may be the development of an information system, a website, and/or a mobile application. Software in this context will require using concepts learned in the undergraduate curriculum. Examples of these concepts include object-oriented programming, data abstraction, modularity, portability, web design, e-commerce, networking, user interface, software engineering issues, logic programming, heuristics, algorithm efficiency. Original code written with languages such as Assembly, C, C++, C#, Java, Prolog, Python, SQL is more highly valued as it requires a greater attentiveness to detail and hence a more thorough understanding. Conversely, precompiled, off-the-shelf libraries are of less value than the original software. However, existing software packages may also be used after a careful analysis of the advantages their utilization will bring. Hardware in this context means electronic components and devices of the sort studied and used in the undergraduate curriculum. Examples would include microprocessors, microcontrollers, logic elements, integrated circuit devices, various electronic devices, interfacing modules such as sensors, actuators, etc.

While some pre-built modules may be used, a significant part of the system must be *designed*.

Students are normally expected to carry a project from problem statement to implementation of solutions. As an exception, where implementation is not practical within the available time frame or resources, a strong set of feasible recommendations are expected as an outcome of the project.

The project concept itself does not have to be unique or original. Of course, original ideas and solutions will attract higher evaluation. However, the system developed must have some

identifiable aspect that is innovative. It does not mean that the project must be something that has never been done before, but it does mean that some aspects of the project must be innovative as compared to standard uses and functions as taught in the program. That is, the project must be more than a couple of lab exercises that have been combined.

The project topic must allow the achievement of the course goal and objectives specified above. Therefore, the Senior Project must address a complex task that involves all the phases and components of completing real-life projects in the prospective working place where the students are likely to land after graduation. The project should involve the systematic investigation of a topic or design task that is significant and timely and that demonstrates the student's ability to work individually with self-motivation or as part of a team using newly acquired knowledge or building on previously learned experience in the field to solve the relevant engineering problems.

Research projects that have implementation parts are also acceptable, however, the execution and grading of these projects have some differences from software development projects described in this document. See appendix 4 for detailed information about research projects.

Unacceptable as total projects are:

- Pure research work without any implementation part
- System integration using off-the-shelf hardware and software
- Library research
- Market surveys
- Software evaluation
- Comparative analysis of methods, packages, etc.
- Simulation or implementation of known single simple algorithms

The above tasks can be the only components assumed to be completed during the work on the Senior Project.

To earn an A or B grade in SE 4910 and SE 4920 your project will need to include a challenging piece of design work. Such work will require development beyond what you have studied in class and encountered in lab exercises and term projects.

SELECTION OF TOPIC AND ADVISOR AND REGISTERING FOR THE COURSE

For each senior project, there will be an advisor assigned by the Department, and a team working on the specified topic. The students from the team are registered and the advisor is assigned during the project selection process described in this section. The Head of Department appoints a faculty member responsible for the sequence of the two courses; this person will be referred to as the course coordinator.

Project topics may be suggested by:

- A faculty member from the Software Engineering department
- A faculty member from another department
- A student or a group of students
- A company or organization

An approximately even distribution of projects among advisors will be generally sought. Interested faculty members from other academic departments may also suggest project topics if they fulfill the general project requirements described above.

External companies or organizations may suggest topics of interest to them. Such projects may have either an advisor from the faculty or two co-advisors – one from the company or organization and one from the faculty.

Students are also strongly encouraged to suggest their project topics or topics of interest to an external company or organization.

Every student is responsible for registering for a Senior project not later than the end of the fourth week of the Fall semester. To register, students must:

- Register for SE 4910 in the Student Affairs Office.
- Contact the faculty member who has proposed a topic in the posted list or approach a faculty member and discuss with them a topic proposed by the student.
- Discuss the prospective project topic and obtain the agreement of the faculty member to have the student or the group of students work on the topic under the supervision of the faculty member.

The project advisor informs the project coordinator of the projects (and corresponding teams) (s)he has agreed to supervise.

A student should select a project from the list that best matches his/her vision and plans as well as the student's resources and skills.

If the student wants to come up with a project topic of their own, it is helpful for the student to become familiar with the research interests of the faculty and to discuss tentative topics with them.

Sometimes a student might be already employed and suggest a project topic related to their ongoing work as an employee. When proposing such a topic, the student must explicitly state the fact that the topic concerns ongoing work. Such requests will be considered by the potential advisor and the course coordinator on a case-by-case basis. It is recommended that the employing company consider the possibility of defining a suitable project from their ongoing work that could involve a Yasar University team and the senior student–employee in that company.

PREPARATION, REVIEW, AND APPROVAL OF PROJECT PROPOSAL

After registering, choosing a topic and advisor, and forming the project team (explained in the previous section), the team must start to work on the project under the supervision of the project advisor. The team must first research the background of the task and clearly understand the goals of the project and determine and specify its requirements and outcomes.

The team must prepare and submit by the end of the 2nd week of the Fall semester a formal Project Proposal.

The Project Proposal serves as a contract between the team, the advisor, and the Department. It helps you clearly understand and specify in a written form what you are supposed to do to complete the project. The Project Proposal is accessed by your advisor and two other faculty

members for acceptability; they will also advise you on what you should focus at and what adjustments you should make to succeed in your project work.

The Project Proposal must be a document of 2 to 4 pages length and include the sections described below.

- Title of the Project.

This should be clear and concise. Try to avoid acronyms if possible.

- List of the students forming the *project team*
- *Advisor's name*
- *Introduction / Background / Overview*

Include an introduction to the subject area. This will provide the reader with an understanding of the field in which your project lies and an idea of where and how your project fits into this field. This aspect will set your project into the overall context and will show where it is bound within a recognized field – not an idea that you have had that makes no sense and has no recognizable foundation.

Describe concisely the current research and development in the field. This will emphasize that your project is not based in a field that is out-of-date and that you are aware of current issues within the field of study.

You should be able to identify some aspect of the field that requires further study or development. This component emphasizes that the field is not exhausted and is worthy of further investigation.

Identify how your work fills the gap identified above. This will emphasize the contribution your project will make. Concisely but clearly, specify what you are producing in your project.

It is useful to highlight any major risks your project might face and ways you envisage dealing with them. If your assessors feel you have not accounted for potential risks to your project, your proposal may not be accepted as they may not appreciate that you have potential contingency plans in place.

You may also want to include reasons why you feel you are a suitable candidate for performing this project, what skills are required, and how you fulfill these requirements. You may write why the topic interests you specifically. The section may also include an introduction to the industry or organization being investigated or where the results of the project will be used.

- *Aims and objectives*

Aims identify at the highest level what it is you hope to achieve with your project. An aim is a broad statement of intent that determines your project's purpose (goal). *In most cases, a project has a single aim.*

Objectives, on the other hand, identify specific, measurable achievements that build towards the ultimate aim of your project. They are more precise than aims and are quantitative and qualitative measures by which completion of the project will be judged. They must be clear and unambiguous. They represent major components of your project that direct your work activity.

Examples of the project's aims and objectives are given in Appendix 1.

- *Expected outcomes/deliverables*

This section of your proposal will identify precisely what you intend to submit at the end of the project.

It can describe programs and user documentation you must produce, models and algorithms you will be developing, written reports that cover particular points and make certain recommendations, functional specifications for a piece of software or a device, prototypes, test plans, etc.

The sections of the project described above are compulsory. You *may* also choose to include additional sections such as the following.

- *Type of project*

You might wish to identify the type of project you are undertaking, for example, research-based, development, evaluation, etc. However, make sure these terms are recognized and provide more detail if appropriate.

- *Keywords*

The keywords are used to identify the topic area your project draws on. People use keywords to see at a glance what subjects your project relates to. Keywords are not necessarily single words but can be simple phrases, for example, artificial intelligence or software engineering.

- *Related research*

This section identifies other work, publications, or research related to your topic. It will demonstrate that your project does not exist in an academic vacuum but relates to other research topics and areas of current interest. Related research can also help demonstrate your understanding of your topic field, showing the reader that you are aware of what is currently happening in the field and are conversant with other topics that impinge upon it.

This section may be also combined with the Background section.

- *Research questions and hypotheses*

Your project may also include the research question you intend to investigate and, hopefully, answer to some extent within your project. Computing projects do not necessarily set out to answer particular questions, but for some projects (particularly

research degree projects) a statement of your research question is essential. Examples of research questions are:

- Does the size of an organization affect its commitment to software quality standards?
- What is the relationship, if any, between software maintainability and coding structure?

Examples of hypotheses related to the first example question above that might be investigated are:

- Hypothesis #1: Large organizations invariably employ recognized standards to maintain internal software quality
- Hypothesis #2: Large organizations generally have quality departments that oversee the implementation of procedures that ensure the quality of internal software

- *Methods*

This section describes the research and project methods you will use in performing your project. It might include development methods (e.g., agile software engineering), survey methods for a case study evaluation or a comparative study, etc.

- *Resource requirements*

You might need to identify any resource requirements for your project such as hardware, software, and access to particular computers. You might also include a list of literature resources you will need.

Students must upload an electronic copy of their Project Proposal to the SE 4910 course site by the end of the 2nd week of the Fall semester.

By the end of the 3rd week of the Fall semester, the Head of Department assigns two reviewers for each project proposal who are faculty members. The reviewers, as well as the project advisor, prepare written reviews (critiques) by the end of the 4th week of the Fall semester. Reviewers upload their reviews to the comment section of the corresponding project proposal submitted by the students.

A review should contain comments on the project proposal, suggestions for improvement, and warnings of possible difficulties. It must also contain a rating of the proposed project based upon the following grade scale, with suggestions of modifications or improvements that will allow the raising of the grade:

- *Excellent*: outstanding project
- *Good*: good result expected if the project is carried out well
- *Adequate*: will require excellence in development and presentation
- *Weak*: marginal chance of success regardless of development and presentation
- *Inadequate*: will not result in a passing grade, unsuitable

In using these ratings faculty are indicating their opinion as to the potential success of the project based on the Project Proposal presented.

The rounded average of the ratings of the three reviews establishes an upper bound for the possible final project grade for SE 4910 and SE 4920, as follows:

- Excellent and Good – max final grade A
- Adequate – max final grade B
- Weak – max final grade C
- Inadequate – will not result in a passing grade

In addition to the rating, the reviewers will also determine the technical complexity of the project using the use case points approach (See https://en.wikipedia.org/wiki/Use_case_points, also the table in Appendix 3). If at least 2 of the 13 factors is not determined as important, the project will be considered inadequate.

Students must carefully examine the feedback from the proposal reviews and, when needed, together with the project advisor, revise the project proposal to accommodate the suggestions for improvement of the proposal.

A revised project proposal must be uploaded to the course website by the end of the 5th week. It must be accompanied by a list of changes made to address reviewers' comments. The Project Proposal becomes an official document that must be attached to the Final Project Report.

In the case when two of the project proposal reviews indicate an Inadequate grade, students must immediately seek another project topic. They must contact the project coordinator for instructions. They should also contact the project coordinator in exceptional cases when they wish to change their project topic for any other reason.

GITHUB USAGE

All the teams must create a GitHub repository at the end of the 5th week, and add themselves, their advisors, and the two jurors to the repository. All of the reports, UML diagrams, source code, scripts to create the databases, etc., i.e. any resource needed to compile and build the project should be in the GitHub repository.

PROJECT REPORTS AND GRADING FOR SE 4910

Students should submit three reports in the fall semester and make an oral presentation to the Software Engineering faculty. A software project plan should be submitted by the end of the 6th week, a software requirements specification document should be submitted by the end of the 10th week, and a software design document should be submitted by the end of the 14th week. In addition, project teams will make oral presentations on the last week of the semester before the final exams.

Students should discuss the contents of their reports and their presentations with their advisors.

The grading policy for SE 4910 is as follows:

- Proposal 10%
- Software Project Plan 20%
- Software Requirements Specs 20%
- Software Design Specs 20%
- Oral Presentation 30%

The work on completing the project continues in the Spring semester as course SE 4920.

STUDENT - ADVISOR CONFERENCE

In both semesters, each team should meet *periodically* (typically, every week) with the project advisor to clarify issues and prevent problems arising during the study and project work.

Individual Weekly Activity Report (see below) must be submitted by each student each week (e.g., every Monday) to the website. Failing to submit weekly activity reports will result in decreasing the final grades SE 4910 and/or SE 4920.

GRADING OF SE 4920 – GENERAL

The final grade for SE 4920 Senior Project II will be assigned by a jury consisting of the project advisor and at least two more faculty members. It is possible to have one member from outside the faculty serving as an external examiner. The project advisor proposes the grade which is then discussed by the jury who makes the final decision. The Head of the Department assigns the jury for each team.

Each student will get a letter grade or an incomplete (I) grade. The student must be cautioned that even though an incomplete grade is possible if the project is not completed within 2 weeks, this will result in an F grade, and the need for registration for the next semester.

In grading a project, the following elements are taken into consideration:

- Testing document 30%
- Implementation Document 40%
- Presentation, Program Demo and Video 30%

Depending on the specific circumstances the jury may modify the relative importance and weights of these factors.

The project work culminates in the preparation of the Final Project Report and the public Project Presentation and Defense.

Project presentations take place during the last week of the Spring semester. The exact dates for all project presentations are posted by the project coordinator not later than two weeks before the end of classes for the Spring semester.

Each group should prepare a video of their project that demonstrates their project or a gameplay for games with short descriptions. The video length should not be exceed 60 seconds.

Generally, to pass SE 4920 your product must be *operational* and *meet the Project Proposal specification* by the day of the Project Presentation. The jurors will use the approved Project Proposal specification to compare the promised features to the actually delivered features and grade accordingly.

PLAGIARISM

Plagiarism is the act of using other people's words and work as if they were your own. It is an attempt to advantage and is, therefore, one form of cheating. It is a very serious academic offense since it is a violation of the objectives of university education.

If you make a point, without saying where you got that idea from then the reader will assume that you created this idea. If, however, this is not the case then you have plagiarized it; you have stolen it. This is cheating.

It is always obvious when a student has lifted words from a text without referencing, as there is a change in writing style. If you do not use references correctly, it would come across that if you had 'stolen' words or ideas from other sources. This is plagiarism and can have serious consequences.

Do not copy text from other references without identifying the source. You can paraphrase some sentences and one or two paragraphs at most. You should demonstrate that you read the references, understood their content, and can digest, summarize, and present them in your reports.

These rules also apply to source codes. If you use someone else's code, mark the part of the code, and write a comment about why you copied the code and from where. Also, inform your advisor to discuss the reason for copying the corresponding code.

PRESENTATION AND DEMONSTRATION

There will be a half-day or a full-day event for demonstrating all of the projects in a public area of the campus. Each team will prepare a poster promoting their project and will be given a table where they can place the computer(s), cell phone(s), tablet(s), or any other devices necessary to demonstrate their working products. The jury, other faculty members, and other people may be visiting the stands to get information about the projects. The project demonstration day is public – any interested person may attend.

Among their other duties, the juries have the responsibility to promote the application of common grading standards and criteria to all graduating students in the Department.

The evaluation of the quality of the presentation is based on factors such as:

- Is the project well defended and conveyed to the audience?
- Is the demonstration well prepared and organized?

The jury meets within two working days after the demonstration to discuss and determine the final grades of the team members. In making their decision, the jurors take into consideration their evaluation of the project, as well as the information provided by the project advisor on the achievements of the team and the individual work of each student.

If the jury cannot agree on the students' final grades, the issue will be discussed with the project coordinator, and, if necessary, the Head of the department. If these meetings cannot resolve the disagreement, the final authority to determine the grade lies with the Department Head who will carefully consider all presented arguments.

TIMELINE AND MILESTONES

Fall Semester

- According to the registration period set in the Academic Calendar - Students register with the Students Affairs Office for course SE 4910.
 - End of 2nd week - The project team submits a formal Project Proposal
 - End of 3rd week – Project proposals reviewed by project advisor and two faculty members
 - End of 4th week – Revised project proposal submitted
 - End of 5th week – Each team creates a GitHub repository, adds the members and the advisors, and the two jurors for their project
 - End of the 6th week - Software project plan submitted
 - End of the 10th week - Software requirements specification document submitted
 - End of the 14th week - Software design document submitted
 - End of the semester – Oral Presentation
-
- Each week starting with the 4th week – each student submits a Weekly Activity report

Spring Semester

- End of 10th week – Testing Document submitted
 - End of 14th week – Implementation Document submitted
 - End of the semester – Project Demonstration
-
- Each week starting with the 2nd week – each student submits a Weekly Activity report

Note: Above deadlines apply for the academic year 2021/2022 and may be updated in the future.

Appendix 1

Project Aims and Objectives: Examples

Example 1

Aim

- Evaluate artificial intelligence techniques for modeling weather patterns

Objectives

- Identify and evaluate existing weather pattern modeling techniques
- Identify AI approaches suitable for modeling weather patterns
- Design and develop at least three AI intelligent systems for modeling weather patterns
- Compare and contrast the developed systems with one another and existing approaches to modeling weather patterns

Example 2

Aim

- The project aims to develop an online digital advertising system that will allow users to make proposal requests for commercial purposes, offering ease of use for advertising in public areas.

Objectives

- Designing a user-friendly interface.
- Building a website in which users can upload videos or images to be displayed on selected outdoor led screens
- Building an Android-based mobile application
- Implementing a preview feature in which users can experiment on their advertisement plan
- Providing approval processes for proposal requests to notify the person in charge.
- Displaying information about currently broadcasting advertisements.
- Supporting payment services for invoicing.

Example 3

Aim

- This project aims to bring people together with applications who need support about education and people who want to help. With this application, people who have financial inadequacy or want to learn something about a topic can find course materials or people who teach a specified subject.

Objectives

- Construct an easy-to-use user interface.
- Develop a mobile application that runs on both iOS and Android operating systems
- Develop a website for the application.
- Develop web services that will allow all three platforms will use.

Appendix 2

Weekly Activity Report: Example

Weekly Activity Report

Due: Mondays 1 p.m.

Your Name: Friendly Student

Student ID: 123456789

All weeks start on Monday and end on the next Sunday

Activity for week starting 25 March and ending 1 April
date of Monday date of next Sunday

Day	Describe Activity	Estimate Time Spent
Monday	Worked with Mabel to make the problem report	3.5 hours
Tuesday	I and Marie-Anne interviewed the chief accountant to get user requirements	1 hour
Wednesday	Team meeting – everyone was there	1 hour
Thursday	Team party	Can't remember
Friday	Read about project management in the library	2 hours
Saturday	No activity	
Sunday	Marie-Anne and I did the "balance the books" activity diagram	30 min

Total:

8 hours

Appendix 3

Project Proposal Review Form

Date:

Reviewer's name:

Project title:

Project advisor:

Team members:

Comments (Is the topic important and of current interest, novelty and difficulty of the project task, clarity/vagueness of project task, risks, and expected difficulties, etc. Is the project topic appropriate for a senior project?):

Suggestions regarding task, objectives, and deliverables:

Rating of project proposal in its current form:

Do you suggest a modified version of the project proposal to be submitted?:

Assigned Use Case Points:

<i>Factor Number</i>	<i>Description</i>	<i>Weight</i>	<i>Assigned Value</i>	<i>Weight * Assigned Value</i>
T1	Distributed system	2.0		
T2	Response time/performance objectives	1.0		
T3	End-user efficiency	1.0		
T4	Internal processing complexity	1.0		
T5	Code reusability	1.0		
T6	Easy to install	0.5		
T7	Easy to use	0.5		
T8	Portability to other platforms	2.0		
T9	System maintenance	1.0		
T10	Concurrent/parallel processing	1.0		
T11	Security features	1.0		
T12	Access for third parties	1.0		
T13	End user training	1.0		
TOTAL				

0 - No Influence

- 1 - Incidental
- 2 - Moderate
- 3 - Average
- 4 - Significant
- 5 - Essential

Total complexity should be greater than or equal **20**
https://en.wikipedia.org/wiki/Use_case_points

Appendix 4

Details about and grading of Research Senior Projects

1. Research Senior Projects:

These projects usually involve designing a solution to an academic research problem. The main output of the project is a research article that presents the study conducted throughout the thesis study. At the end of the 2nd semester usually, the article is expected to be submitted to a journal or conference. The students undertaking this type of project are supposed to submit the following documents to present their work during their thesis study:

Introduction: In this document, the focused problem with detailed context should be presented. The components of the problem (objective, limitations, assumptions, etc.) should be explained in detail. Any background information to facilitate the understanding of the problem should also be included in the document. The motivation of the study should be briefly mentioned.

Literature Review: A detailed review of the literature on the problem should be presented in the document. The difference between the thesis study from the presented studies (if any) should be stressed explicitly.

Methodology: The detailed designed solution should be explained clearly using diagrams and other visual elements if necessary. Some justification about the selected methodology should be made.

Experimental Study: The experimental work should be presented including the experimental configuration, how experiments are conducted, the results of the experiments, and conclusions from the results. The results should be presented in both tables and proper charts to ensure the clarity of the presentation.

Final Article: The summary of the whole work is presented in this document in the form of a scientific article preferably ready to submit at a conference or a scientific journal.

Documents for Research Projects:

Project Type / Semester	Research Project		Software Development Project	
1st Semester	Proposal	10	Proposal	10
	Introduction (Problem Definition, Motivation, Background Knowledge, etc.)	20	Software Project Plan	20
	Literature Review	20	Software Requirements Specs	20
	Methodology	20	Software Design Specs	20
	Oral Presentation	30	Oral Presentation	30
2nd Semester	Experimental Study	30	Testing document	30
	Final Article (Submission Ready)	40	Implementation Document	40
	Presentation and Poster	30	Presentation and Demo	30