

GAZI UNIVERSITY FACULTY OF ENGINEERING

COMPUTER ENGINEERING



SELİN CANSU AKBAŞ

191180005

**Takehome Exam**

DOÇ. DR. MURAT YILMAZ

Object Oriented Analysis and Design - CENG361

Midterm

April 2023

# CONTENTS

1. PROJECT PROPOSAL.....	2
1.1 Introduction.....	2
1.2 Project Objectives .....	2
1.3 ChatGPT Integration.....	3
1.4 System Architecture / Design .....	4
1.5 Implementation Plan .....	5
1.6 Evaluation / Metrics.....	7
1.7 Conclusion .....	7

# 1. PROJECT PROPOSAL

## 1.1. Introduction

The proposed project aims to develop a tutoring system that utilizes ChatGPT to answer students' questions related to software engineering concepts. Software engineering can sometimes seem complicated. Students studying in this engineering may also have difficulties from time to time. In addition, it is not always possible to reach an expert or someone who knows this field to answer the questions of these students.

This system will provide students with a user-friendly interface to interact with the ChatGPT-based tutor, allowing them to obtain quick and accurate answers to their software engineering-related queries. The motivation behind this project is to enhance the learning experience of software engineering students by providing them with an efficient and effective way of getting their questions answered. In this way, it is aimed that students adapt to software engineering more quickly, like this department more and hold on to it, and it is thought that these students have the opportunity to be more successful.

## 1.2. Project Objectives

The main objectives of the project are as follows:

- *Develop a tutoring system that utilizes ChatGPT to answer students' questions related to software engineering concepts:* The main purpose of this project is to create an education system that uses ChatGPT technology to produce accurate and original answers to students' questions about software engineering topics. This system must be able to provide high-quality answers to a wide variety of software engineering-related queries, including programming languages, software design patterns, tests, and any application imaginable about the project.

- *Create a user-friendly interface for students to interact with the ChatGPT-based tutor:* Another aim of the project is to design and develop a user-friendly interface that allows students to easily ask their questions and communicate with the teacher based on ChatGPT. The interface should have a simple design that makes it easy for students to ask questions and get answers. This can include features such as suggestions for auto-understanding and completion of questions and topics, topic recommendations, and the ability to ask follow-up

questions for clarification.

- *Enhance students' understanding of software engineering concepts by providing accurate and concise answers to their questions:* In short, it aims to create an interface that will enable students to learn software engineering in a way that they can learn and be interested in more easily.

- *Develop ChatGPT's understanding of software engineering and domain-specific information:* ChatGPT needs to be trained on a wide variety of software engineering data and knowledge in this field in order to provide accurate and quality answers to students' questions. This includes textbooks on software engineering concepts, may include articles and online resources. The goal is to build a robust and comprehensive knowledge base that ChatGPT can leverage when generating responses.

- *Continually enhance the ChatGPT-based tutor's performance:* The system should be built to continually learn and perform better as students engage with the ChatGPT-based tutor and ask questions. As the system develops further in this way, it will be a more successful and effective system for the benefit of students.

- *Analyze the tutoring system based on ChatGPT's performance:* The aim is to show that the system can efficiently assist software engineering education and offer students a different learning resource.

### **1.3. ChatGPT Integration**

- The first step in integrating ChatGPT into the course system is to collect massive amounts of data on software engineering topics. This may include textbooks, academic articles, online resources.

- After all information and data is collected, it needs to be pre-processed to eliminate unnecessary information.

- The preprocessed data is then used to train the ChatGPT model. This includes gathering a large amount of information on the model and giving high-quality answers to a variety of questions related to software engineering topics and tuning parameters.

- With the ChatGPT model trained and ready to use, the next step is to build a user interface that enables students to input their questions and receive answers. The interface should be user-friendly and easy to navigate.

- Deploy the tutoring system and continue to monitor its performance. You can also periodically retrain ChatGPT on new data to improve its accuracy.

Overall, this requires careful planning, implementation, and ongoing maintenance. By following these steps, we can develop an effective tutoring system that enhances students' learning experiences in the field of software engineering.

ChatGPT will be integrated into the project by using it to generate responses to students' questions related to software engineering concepts and specific ChatGPT prompts used to generate ideas and insights for the proposal include:

- "How can I design a tutoring system that utilizes ChatGPT to answer students' questions related to software engineering concepts?"

- "What are some best practices for creating a user-friendly interface for a tutoring system?"

- "How can I ensure that the ChatGPT-based tutor provides accurate and concise answers to students' questions?"

These prompts helped a clear structure and direction for my proposal, helping me to organize my thoughts and present a comprehensive plan for designing a tutoring system that utilizes ChatGPT to answer students' questions related to software engineering concepts.

## **1.4. System Architecture / Design**

The proposed tutoring system will consist of the following key components or modules:

- *User interface:* Students will be able to submit questions about software engineering concepts and receive responses from the ChatGPT-based tutor using this component. Students will be able to input questions, examine responses, and ask clarification questions via the interface. Students will find the ChatGPT-based tutoring system's user interface to be simple and easy to use.

- *ChatGPT-based tutor*: This component will utilize the ChatGPT model to generate responses to students' questions. An API will be used by the ChatGPT-based tutor component to receive questions from the user interface and provide answers produced by the ChatGPT model. The answer will be generated in natural language and could contain text, links to helpful websites, or visual aids like diagrams or bits of code.
- *Knowledge base*: This component will store information related to software engineering concepts and will be used by the ChatGPT-based tutor to generate accurate and concise responses.
- *Cloud-based deployment*: To guarantee high availability and scalability, the system will be set up in the cloud. Students will be able to use any internet-connected device and log into the system at anytime, anywhere.

## **1.5. Implementation Plan**

The following steps are required to implement the project:

### **Step 1. Data Collection and Preprocessing (1 month)**

- Gather a lot of information about software engineering principles from many sources.
- Using preprocessing methods such sentence tokenization, stop word removal, and stemming, the data is cleaned up of any unnecessary information.

### **Step 2. Training ChatGPT (2 months)**

- Train the ChatGPT model with a deep learning framework on the preprocessed data.
- To improve the model's performance on software engineering problems, change the weights and parameters.
- To increase the model's efficacy and accuracy, make necessary adjustments to certain software engineering subtopics.

### **Step 3. Building the User Interface (2 months)**

- Design a user-friendly interface that enables students to input their questions and receive answers.
- Include features such as autocomplete suggestions and the ability to ask follow-up questions for clarification.

### **Step 4. Integration and Deployment (1 month)**

- Connect the ChatGPT model's user interface, and then deploy the solution to a server or cloud environment.
- Test the system to make sure it can manage numerous simultaneous requests and offer prompt, precise answers to student inquiries.

### **Step 5. Continuous Improvement (ongoing)**

- Use usage statistics and user reviews to spot areas that could use improvement.
- To enhance the functionality and user experience of the system, update the ChatGPT model and user interface.
- Add new information and domain-specific knowledge to the system's knowledge base to increase the system's accuracy.

### **Step 6. Evaluation (1 month)**

- Create user studies or surveys to evaluate the effectiveness of the ChatGPT-based tutoring system in students learning software engineering topics.
- Analyze usage data and user feedback to identify areas for improvement and lead future development.

Overall, it is estimated that the ChatGPT-based system will take approximately 8-10 months to build for the software engineering field, and ongoing maintenance and improvements are planned after the application is released.

## 1.6. Evaluation / Metrics

The success of the project will be measured using the following metrics:

- *Accuracy of the ChatGPT-based tutor's responses to students' questions:* The correctness and comprehensiveness of the system-generated responses are a crucial criterion for assessing the efficiency of the ChatGPT-based tutoring system. This can be assessed by comparing the responses produced by ChatGPT to those given by subject-matter experts as well as by looking at the system's overall accuracy rate.
- *Learning results:* The main objective of the teaching system based on ChatGPT is to increase students' comprehension and memory of software engineering principles. This can be assessed by contrasting the performance of students who use the system with those who do not, as well as by looking at the long-term learning results of students who utilize the system.
- *Student satisfaction with the user interface and the ChatGPT-based tutor's responses:* User engagement and satisfaction are crucial metrics for measuring the effectiveness of the ChatGPT-based teaching system. This can be assessed through user surveys, feedback forms, and metrics for measuring user engagement including the quantity of queries each session, the length of sessions, and the proportion of users who return for numerous sessions.

## 1.7. Conclusion

In conclusion, the proposed ChatGPT-based software engineering tutoring system has the potential to fundamentally alter how students learn and comprehend difficult software engineering ideas. The system can give students prompt, precise responses to their inquiries and aid in their development of a deeper understanding of the subject matter by utilizing the capabilities of machine learning and natural language processing.

The system will be able to produce precise and succinct solutions to students' inquiries thanks to the integration of ChatGPT. Students will learn more and engage with the system more easily because to the user-friendly design. By giving students a quick and convenient means to get their questions answered, this project has the potential to have a significant impact on software engineering education.