## Final Year Project Proposal

**(Fall 2023)**

1. Project Title

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Names and IDs of Students

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Project Advisor (Name, Email Address)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Project Co-Advisor Name (Optional)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Nomination of 5 Evaluation Committee members by the advisor (FYP Committee will select two of them)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. External Collaboration/Funding (if any) + Paid or Unpaid (Confirmed or Expected)

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Project Description (Brief Introduction)

***Overview of topic/domain***

*This project aims to leverage the capabilities of the Generative Pre-trained Transformer (GPT) model for automating the generation of Embedded C code.*

***Description of project/problem***

*The manual creation of Embedded C code is a time-consuming and error-prone task. This project addresses this challenge by exploring the use of GPT to automatically generate Embedded C code snippets based on a given context or requirement.*

***Solution to explore***

*The proposed solution involves training the GPT model on a dataset of clean Embedded C code obtained from GitHub. The model will then be fine-tuned to generate code snippets relevant to specific requirements.*

***Expected results/product***

*The expected outcome is an automated code generation system powered by GPT, capable of producing accurate and contextually relevant Embedded C code snippets. This will significantly reduce development time and enhance code quality.*

1. Major Features/Requirements/Objectives (Tentative)

***Objectives***

*Train GPT model on a dataset of 70GB clean Embedded C code.*

*Fine-tune the model for code generation specific to embedded systems.*

*Develop a system to accept input context and generate corresponding Embedded C code snippets.*

***System Requirements (Tentative)***

***Functional Requirements:***

*Accept user input for code generation context.*

*Generate Embedded C code snippets based on the input.*

*Provide options for code customization and parameterization.*

***Non-functional Requirements:***

*Ensure generated code is syntactically correct.*

*Optimize code snippets for efficiency.*

*Support integration with common Embedded C development environments.*

1. Scope of the Project/Proposed System

***The project scope includes:***

*Training the GPT model on a 70GB dataset of clean Embedded C code.*

*Fine-tuning the model for specific embedded systems requirements.*

*Developing a user interface to input context and receive generated code snippets.*

*Testing the system on various embedded systems scenarios.*

1. Target Users/Beneficiaries of the Proposed System

***The target users include:***

*Embedded system developers seeking to expedite code development.*

*Organizations involved in embedded systems development.*

*Educational institutions teaching Embedded C programming.*

*The project is important for enhancing productivity, reducing development time, and promoting best practices in Embedded C programming.*

1. Tools/Technologies (Tentative Listing)

***Data Collection:***

*GitHub API for obtaining code links.*

*Kaggle for accessing the 70GB clean Embedded C code dataset.*

***Model Training:***

***GPT model with the following parameters:***

*n\_layer: 6*

*n\_head: 6*

*n\_embd: 384*

*dropout: 0.2*

*learning\_rate: 1e-3*

*max\_iters: 5000*

***Code Generation System:***

*Programming language: Python*

*User interface: TBD (e.g., web-based or command-line interface)*

*Integration with Embedded C development environments.*