MODEL RESEARCH: AI-Driven Code Reviewer

- **Introduction**

An AI-Driven Code Reviewer is an intelligent system designed to automatically analyse source code and identify potential bugs, code quality issues, and areas for improvement. The objective of this model is to enhance the efficiency, accuracy, and consistency of the software code review process.

- **Model Selection**

This project utilizes a **Transformer-based pre-trained language model** specialized for programming languages. Such models are capable of understanding both the syntactic structure and semantic meaning of source code.

Selected Model:

- CodeBERT / GraphCodeBERT

These models are pre-trained on large-scale code repositories and natural language data, making them suitable for code analysis tasks.

- **Working of the Model**

1. Source code or code changes (diffs) are provided as input
2. The code is tokenized and structurally represented
3. The Transformer model analyzes code patterns and context
4. The model predicts issues and generates review suggestions

- **Dataset Used**

The model is trained and evaluated using publicly available datasets, including:

1. Open-source GitHub repositories
2. Code review and commit history data
3. Bug-fix datasets such as Defects4J

- **Model Output**

The AI-driven code reviewer provides the following outputs:

- Detection of potential bugs
- Identification of code quality and style issues
- Suggestions for code improvement

- **Evaluation Metrics**

The performance of the model is evaluated using standard metrics such as:

1. Accuracy
2. Precision
3. Recall
4. F1-Score

- **Conclusion**

  The AI-Driven Code Reviewer model supports automated and reliable code analysis, reducing manual effort and improving software quality. It serves as an effective tool to assist developers during the code review process.