**Name of student :** FAKIR Yasmine

**Name of your level 1**: IGUEME Merveille

**Title**: A data-centric service code generation using deep learning

**Key words:** Service-Oriented Architecture, Web Service APIs, Automatic Code Completion, Metadata Annotation, Data-Centric Service Code Generation (DCServCG), Open Source Software (OSS), Conditional Text Generation, Machine-Readable Documentation, Language Modeling Metrics, Binary Classifier, Sequence Overlap, Transformer, DistilGPT2, Code Auto-Complete, General-Purpose Code Generators, Neural Networks (NNs), BLEU Score, Perplexity, Word Embedding Techniques, GitHub.

**Summary:**
This study proposes a novel approach, the Data-Centric Service Code Generation (DCServCG) model, to enhance web service-based systems through automatic code completion. The primary focus is on leveraging open-source software (OSS) projects to capture and understand service usage, providing developers with accurate code suggestions and essential web service features during the programming process. The central challenge addressed is the lack of machine-readable documentation and structured metadata for web services, negatively impacting the performance of automated Service-Oriented Architecture (SOA) tasks.

Key contributions of the study include the introduction of the DCServCG model, designed to overcome limitations in existing general-purpose code generators. This model incorporates crucial service-based code characteristics, such as sequence overlap and bias issues, by adopting a data-centric concept, specifically conditional text generation. The study collects OSS project source code from public repositories, emphasizing projects utilizing web service APIs, to extract pertinent information for developers.

A notable strategy introduced is the use of a binary classifier for data de-noising, filtering out passive codes during the training phase. This not only reduces the computational load but also enhances the model's ability to suggest relevant codes related to web service usage. The study also employs conditional text generation to address bias and sequence-overlapping issues in web service source code, leading to improved model generalization.

Evaluation using language modeling metrics demonstrates a reduction in perplexity, indicating enhanced model performance. Comparisons with other models, including DistilGPT2 and ServCG, highlight the superior performance of the DCServCG model. In summary, the study provides a comprehensive approach to address challenges in web service development, offering an innovative solution for automatic code completion with a focus on data-centric modeling and leveraging OSS source code.

**AI Model Used:** The AI models used in the article encompass Transformers, with a focus on DistilGPT2, Binary Classifiers, Language Models, Conditional Text Generation, Neural Networks (NNs), and Machine Learning (ML).