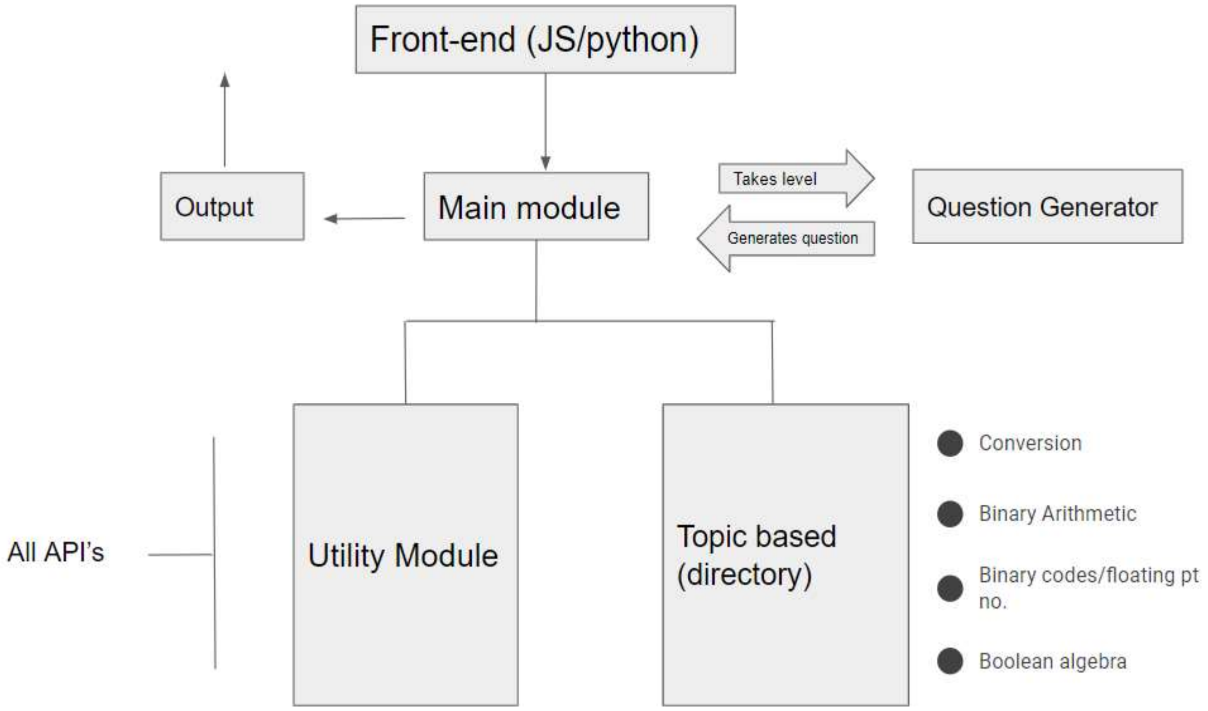
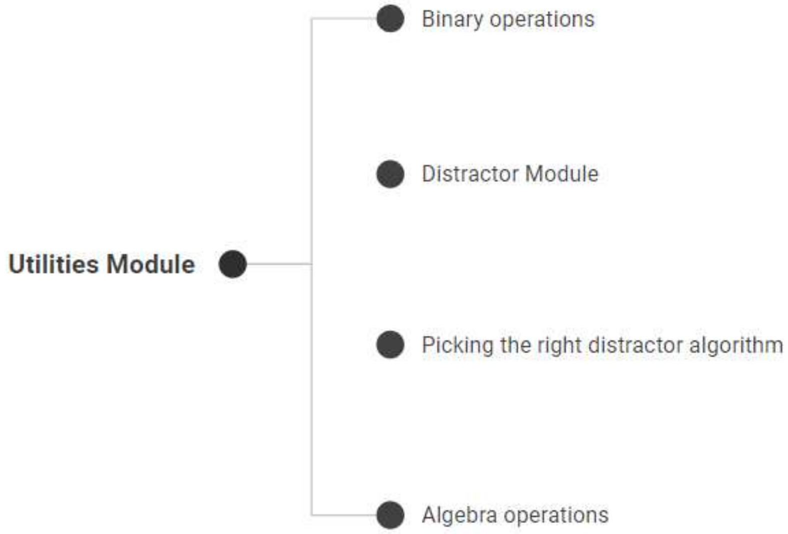
Design Document

Components

- User Interface: A command-line or graphical user interface that allows users to select difficulty levels and view questions.
- Question Generator: A Python module that generates random questions and correct answers based on the chosen difficulty level.
- Database (optional): An optional component for storing and retrieving questions and answers.
- Settings Module: Manages settings, including difficulty levels, topics, and other configurable parameters.
- Distractor Module: Contains various Distractor algorithms based on the type of question and its level of difficulty





High-Level Architecture

The project will follow a modular architecture with the following components:

- **User Interface:** Handles user interactions and displays questions.
- **Question Generator:** Responsible for generating random questions and answers.
- **Database (optional):** Stores questions and answers for future use.
- **Settings Module:** Manages difficulty levels and settings.
- **Distractor Module**: Contains various Distractor algorithms based on the type of question and its level of difficulty

Implementation

Initially command Line user interface in python.

Technology Stack

- Programming Language: Python(for API's)
- User Interface: JS/ReactJS, Tailwind CSS
- Database (optional): SQLite or other lightweight database.

Algorithms

- Randomization algorithms for generating questions.
- Algorithms for selecting questions based on difficulty level.
- Various Distractor algorithms for generating options based on varying difficulty levels.