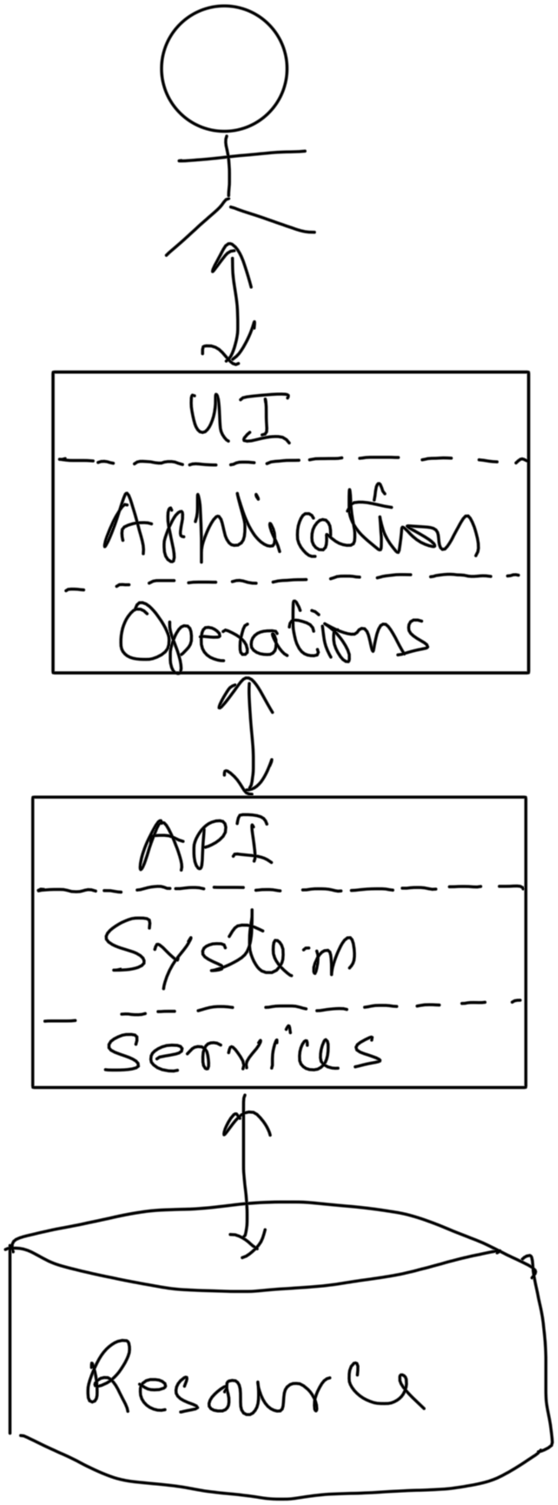
# DAC 0521 MET-M Writeup

Software - It is a sequence of instructions (code) known as a program that a machine can load (from storage into memory) and execute to perform a particular task involving processing of some data according to a predefined set of logical steps known as an algorithm. There are two main types of software

1. System - It implements services to manage some resource (source of data) and provides an application programming interface (API) allowing other softwares to consume those services.
2. Application - It implements operations to process data managed by a system and provides a user interface (UI) allowing humans to consume those operations.



Programming Language - It is a tool which provides a syntax consisting of a set of symbols and a set of rules for combining these symbols into a human readable representation of a program known as its source-code. There are two main types of languages

1. Assembly Language - The symbols are mnemonics (friendly short names) for machine

specific opcodes (binary machine instructions). A source file containing assembly language code is translated by an assembler into its binary equivalent file containing object code and all such object code files required by a program are combined by the linker to produce the loadable program file.

2. High Level Language - The symbols are machine independent keywords derived from a natural language such as English. High level languages are classified into following categories

(A) The entire source code is checked and translated by the language compiler into assembly language code of a particular machine which is then assembled and linked to produce the loadable program file for that type of machine. Example - C and C++

(B) The source code is executes statement by statement by the language interpreter on any machine on which this interpreter is preinstalled. Example - Python and JavaScript

(C) The entire source code is checked and translated by the language compiler into machine neutral binary instructions which are executed by the language runtime on any machine on which this runtime is preinstalled. Example - Java and C#

| Feature | Assembly Language | High Level Language |
|---|---|---|
| Power | High | Low |
| Portability | Low | High |
| Performance | High | Low |
| Productivity | Low | High |

| Support | Category A | Category B | Category C |
|---|---|---|---|
| Checked Typing | ✓ | ✕ | ✓ |
| Dynamic Typing | ✕ | ✓✓ | ✓ |
| Binary Distribution | ✓ | ✕ | ✓ |
| Portable Distribution | ✕ | ✓ | ✓ |
| Fast Execution | ✓✓ | ✕ | ✓ |
| Safe Execution | ✕ | ✓ | ✓ |