# CSN-254
# Software Engineering
# Assignment-1

Group-4

| Name | Roll No | Contribution |
|---|---|---|
| Meet Sindhav | 22114053 | Common discussion and articulation of Q2 |
| Mohammed Haaziq | 22114055 | Common discussion and adding insights |
| Aditya Mundada | 22114058 | Common discussion and articulation of Q1 |
| Nayan Kakade | 22114060 | Common discussion and articulation of Q1 |
| Polasa Hariharan | 22114068 | Common discussion and suggesting changes |
| Roopam Taneja | 22125030 | Common discussion, reviewing and final document preparation |

## Q1. Discuss and compare the process for performing feasibility study, requirement gathering and analysis and software deployment and maintenance in the following scenarios:

**a. When the software is to be developed for a given customer. i.e. end users are known**

**b. When the software is to be developed for an open market i.e. to be launched in the product market.**

### Comparing Feasibility Study

**What is Feasibility Study?**

Feasibility Study is an analysis done to determine whether it is financially viable and technically feasible to develop the software.

Following are the processes undertaken during feasibility study stage:

1. Development of an overall understanding of problem statements and requirements of the customer.
2. Development of multiple high-level solutions to the problem statement.
3. Analyze the above developed solutions based on benefits and consequences.
4. Make estimates of available resources, required cost and development time.

**When the end user is known:**

When the end user is known we can perform a targeted analysis of the user's requirements and hence we can determine how costly the project will be for the consumer or which of the technical aspects of the to-be developed software should be kept at high priority. We can also ensure that the technology stack aligns with the preferences and infrastructure of the end users.

Making a timeline of the project also becomes easy as the user is continuously involved in the process. Continuous collaboration with the user helps us adjust to the changing requirements from time to time and hence makes project management easier. Scalability of the project in the future can also ensured as the customer is in continuous touch with us.

**When the end user is not known:**

Here we need to perform a very rigorous market analysis of the trends in the market as well as the market demand. Competitive analysis is also necessary. We must also keep in our mind that the project must be not replicating something already existing in the market as it will not be of much use to the people. An extensive cost analysis is also important.

**Comparing requirements gathering**

**What is requirements gathering?**

This phase involves collecting all necessary information related to the expectations, needs and requirements of the user.

It also involves surveying users, customer market to gather information.

Software Requirements Specification (SRS) documentation is prepared which includes use cases, functional and non-functional requirements.

*Requirements Analysis:* This process is carried out to remove the ambiguities, incompleteness and inconsistencies in the requirements to carve out a clear and crisp SRS document which is then forwarded to the development team.

**When the end user is known:**

In this scenario, we can continuously keep interviewing the user to understand what are the functionalities that are at top priority for the user and what outcomes the user expects from the product. Also, changing requirements with time can be easily captured. This makes the process more targeted and focused.

Handling different use cases becomes easy. The overall security of the system is also enhanced as the user's interaction with the system can be configured. It also makes the documentation of functional and non functional requirements easy.

**When end user is not known:**

When we have to develop software for the open market space and not a particular user, we need to conduct market surveys and understand the needs and expectations of the various stakeholders (investors, owners, common public) in the market. We need to develop use cases for each stakeholder having different actors. Analysis of non functional requirements such as scalability, features etc. is needed. We need to gather the changing requirements time to time from the market through feedback and beta testing.

**Comparing software deployment and maintenance:**

**What is software deployment and maintenance?**

*Software Deployment:* Software developed is made available to users for use. This can be done through various deployment platforms such as cloud servers. It involves the activities of release, installation, testing several times to check for bugs and correcting them.

*Software Maintenance:* After the release of software, bugs may arise, which need to be solved. Also with changing user requirements, it becomes necessary to cater their needs with updated software. So software needs to be modified and errors need to be rectified. It also involves release of newer versions of the software.

After the release of software, software maintenance is the most important step which need to be carried out time to time.

**When the end user is known:**

When the end user is known we can ensure that the user continuously gives us information and feedback about the software as the user puts it in use. This feedback loop helps us determine what are the improvements that we can make in the software from time to time. Knowing the end user ensures lesser number of phases in the process of development and the pilot testing of each additional feature is easy.

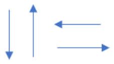
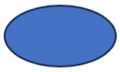
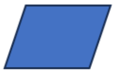
**When the end user is not known:**

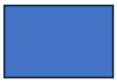
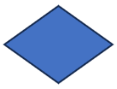
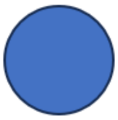
When developing product for an open market, continuous monitoring of the changing market trends becomes important. We must continuously update our software to meet the rising expectations and use cases of the large market. Thus, here we need more to deploy the software in phases and overall ensure that the requirement changes are met in all the updated versions. Here, we must also create user-friendly documentation that includes step-by-step guides, FAQs, and troubleshooting information for the consumers to be able to use the software properly.

## Q2. Draw various symbols for flow charts and describe them using any example.

### What is a flowchart?

A flowchart is a visual representation of a process or algorithm, using various shapes, symbols, and arrows to depict the steps involved and the flow of information or control within that process. It provides a clear and concise way to understand the sequence of activities and decision points in a system. Flowcharts are useful for designing, documenting, analyzing, and improving processes or programs in various fields and disciplines

| Symbol | Description |
|---|---|
|  | **Flow Lines**:<br>Flow lines are used to connect symbols and indicate the direction of flow. |
|  | **Terminal (START / STOP)**:<br>This is used to represent the start and end of the flowchart. |
|  | **Input / Output**:<br>Parallelograms represent program instructions that take input or show output in a flowchart. |

| Symbol | Description |
|---|---|
|  | **Processing:**<br>All the processes are represented by a rectangle which includes arithmetic operations like addition, subtraction, division, etc. |
|  | **Decision:**<br>This symbol is used to check any condition or take any decision for which there are two answers, Yes/True or No/False. |
|  | **Connector**:<br>It is used to connect or join flow lines. |

**Example:**

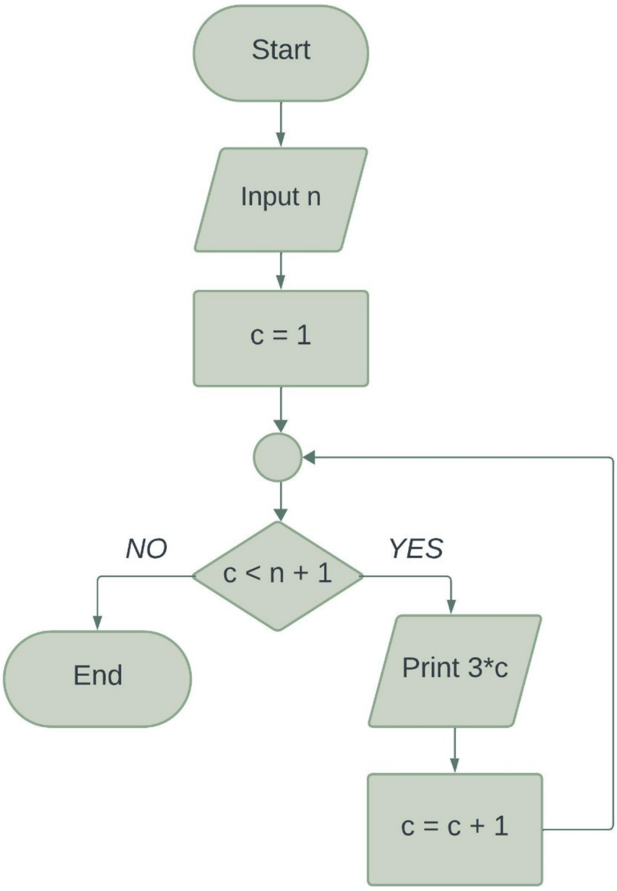


**Figure 1:** Program to print first n multiples of 3

**Explanation:**

1. *Initialization*: Start by initializing the counter variable to 1 and getting the value of ‘n’ from the user.
2. *Loop*: Enter a loop that continues as long as the counter is less than or equal to ‘n’.
3. *Print if Multiple*: If the condition is true, print the value of the 3 x counter.
4. *Increment Counter*: Increment the counter by 1 in each iteration of the loop.
5. *End of Loop*: Once the loop is completed, the flowchart ends.
6. *End*: The flowchart concludes.