Read the text and answer the questions.

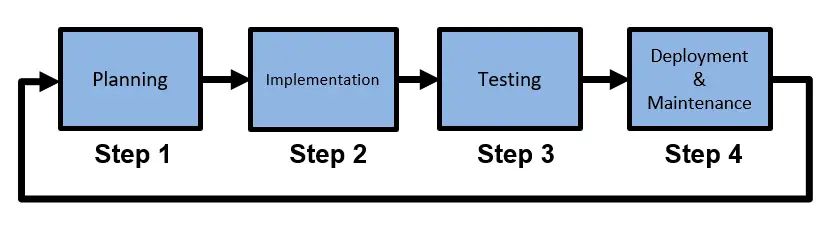
Purpose of a Software Development Process

A solid software development process ensures that high-quality software products are made quickly and well. A well-thought-out and well-executed method has several advantages:

1. Quality assurance: A solid method puts quality first at every step, from figuring out what needs to be done to testing and putting it into action. It includes things like code reviews, testing methods, and quality assurance techniques, which help find mistakes, bugs, and security holes early in the development process so they can be fixed.
2. Consistency and repeatability: A well-defined method gives software development a consistent framework. It spells out the steps, roles, and tasks, ensuring each project uses the same method. This consistency lets teams repeat good practices, reuse parts, and build on what they’ve already done, which increases output and reduces risks.
3. Collaboration and coordination: A robust process makes it easier for team members to work together by giving clear instructions and routines. It makes it easier for people with different jobs and responsibilities, like developers, designers, testers, and project managers, to talk to each other, share information, and work together. When people work together, there are fewer misunderstandings, and development cycles go more smoothly.
4. Risk management: A set method helps find and deal with risks throughout the software development lifecycle. It has ways to figure out the risks, how to deal with them, and what to do if something goes wrong. By dealing with possible problems early on, the process reduces the chances that the project will be late, cost more than expected, or fail in a major way.
5. Scalability and efficiency: A robust process makes scaling possible by letting teams work on bigger and more complicated projects. It helps with allocating resources, sharing work, and putting tasks in order of importance. With well-defined processes in place, organizations can get the most out of their development efforts, shorten the time it takes to get a product on the market, and adapt to changes in project needs.
6. Continuous improvement: One of the most important parts of a strong software development process is constantly focusing on improving things. It supports looking at past projects, lessons learned, and feedback loops to find places where things could be better. By reviewing and improving the process on a regular basis, companies can improve their software development skills, encourage new ideas, and stay competitive in a field that is changing quickly.

**Software Development Process Steps**

The software development process consists of four major steps. Each of these steps is detailed below.



**Taller**

1. Make a conclusion of the text (it may content 150 words)
2. Change the word in bold, according to the meaning in the text.

A solid software development process **ensures** that high-quality software products are made quickly and well. A **well-thought-out** and well-executed method has several **advantages**:

1. Write the definition (English definition) of the words in bold

Consistency and **repeatability**: A well-defined method gives software development a consistent **framework**. It spells out the steps, roles, and tasks, ensuring each project uses the same method. This consistency lets teams repeat good practices, reuse parts, and build on what they’ve already done, which increases **output** and reduces risks.

1. Look for the information about phrasal verbs and its use.
2. According to the text, find the meaning of the following phrasal verbs.

* Quality assurance: A solid method puts quality first at every step, from **figuring out** what needs to be done to testing and **putting** it **into** action.
* It **spells out** the steps, roles, and tasks, ensuring each project uses the same method.
* This consistency lets teams repeat good practices, reuse parts, and **build on** what they’ve already done, which increases output and reduces risks.
* It has ways to **figure out** the risks, how to deal with them, and what to do if something goes wrong.
* One of the most important parts of a strong software development process is constantly **focusing on** improving things.
* It supports looking at past projects, lessons learned, and feedback loops to find places where things could be better.

1. According to the Software Development Process Steps

Structure the process of your project, just the steps.

**Solution**

1. a robust software development process provides a foundation for efficiently creating high-quality products. Through prioritizing quality, consistency in practices, continuous improvement, and risk management, this approach offers significant benefits. It ensures early detection of errors and vulnerabilities, fosters collaboration among multidisciplinary teams, minimizes risks, and enhances operational efficiency. Moreover, the ability to scale and adapt to larger and changing projects ensures the capacity to remain competitive in a rapidly evolving technological landscape. Ultimately, a well-executed software development process not only leads to successful outcomes but also encourages innovation and continuous learning to thrive in a dynamic market.
2. D
3. **Repeatability:** is a measure of the ability of the method to generate similar results for multiple preparations of the same sample

**Framework:** a frame or structure composed of parts fitted and joined together.

**Output:** the process of transferring such information from computer memory to or by means of an output device.

1. A phrasal verb is a combination of a verb and one or more particles (prepositions or adverbs) that together convey a specific meaning. The meaning of a phrasal verb is often not immediately obvious from the meanings of its individual words. Phrasal verbs are a common feature of English language and are used extensively in both spoken and written communication.

Some examples: turn up, take off, give up, look after, run into

Using phrasal verbs effectively can make your English sound more natural and fluent. However, they can be challenging for non-native speakers to master due to their figurative meanings and varied usage patterns. The best way to learn phrasal verbs is to encounter them in context, practice using them, and gradually build your understanding of their meanings.

1. **figuring out:** to investigate or think something through in order to understand it.

**Putting into:** to use (a certain amount of energy or effort) when doing (something) He puts a lot of energy into his performances

**Spells out:** to make plain. spelled out the orders in detail

**Build on:** use something as a basis for further development.

**figure out:** solve a problem or discover the answer to a question

**focusing on:**

6.

Certainly, here are the typical steps in the software development process:

**Requirement Analysis:** Understand and document the project's objectives and user requirements.

**Planning**: Create a project plan, including scope, timeline, budget, and resource allocation.

**Design:** Create detailed technical specifications and system architecture.

**Implementation (Coding):** Write the actual code for the software.

**Testing**: Verify that the software functions correctly and meets the specified requirements.

**Deployment:** Release the software for use in the intended environment.

**Maintenance and Support:** Provide ongoing updates, bug fixes, and support as needed.