|  |  |  |
| --- | --- | --- |
| Virtual university Logos | CS605- SOFTWEAR ENGEINRING-II  ASSIGNEMN-2  JUNAID MALIK | VULMS Help |
| [BC190202640@vu.edu.pk](mailto:BC190202640@vu.edu.pk)  [Junaidfazal08@gmail.com](mailto:Junaidfazal08@gmail.com) | **For More Visit:** **vulmshelp.com** | JUNAID MALIK (0304-1659294) |



**Question no. 1:**

A multi-national company has signed a contract with a renowned software house to scrap and re-design their software system. Although the current software system is already of high quality and is working absolutely fine.What could be the potential reasons for this decision? Justify your answer with at least **two valid reasons.**

**NOTE: WRITE ANY TWO VELID REASONS**

**SOLUTION:**

There could be several potential reasons for a company to scrap and re-design their current software system, even if it is already of high quality and working well.

1. Technological advancements

The software industry is constantly evolving, and new technologies and techniques may have emerged since the current system was developed. The company may want to take advantage of these advancements to improve efficiency, scalability, or security.

1. Business requirements:

The company's business requirements may have changed since the current software system was developed. For example, the company may have expanded into new markets or diversified its product offerings, which would require changes to the software system to support these new requirements.

1. Cost-effective:

The company may want to take the opportunity to re-architect their software system to be more cost-effective to maintain and operate. This could include reducing the number of third-party dependencies, optimizing performance, or simplifying the codebase.

1. Updating to new technology:

The company may want to take advantage of new technologies and advancements in software development that were not available when the current system was built. Updating the system to use newer technologies can improve performance, scalability, and security.

1. Need for scalability:

As the company's business grows, the current software system may not be able to handle the increased workload. Re-designing the system to be more scalable can ensure that it can handle the company's future growth.

1. Compliance:

The company may need to comply with new industry regulations or standards that the current system does not meet. Re-designing the system to meet these regulations can avoid potential penalties and ensure the company's continued compliance.

**Question no. 2: 9 Marks**

Based on the scenario (in the question#1) what could be the architectural problems with the software system which is being re-designed? Mention with at least **three of them.**

Solution:

Based on the scenario, there could be several architectural problems with the current software system that is being re-designed. Some possible problems include:

1. Lack of scalability:

The current software system may not be able to handle an increase in the workload as the company's business grows. This could result in poor performance and downtime.

1. Lack of modularity:

The system may not be modularly designed, making it difficult to add or remove features as the company's business requirements change.

1. Poor maintainability:

The current system may be difficult to maintain and update, making it costly and time-consuming to fix bugs or add new features.

1. Lack of support for new technology:

The current system may be built on outdated technologies that are no longer supported or are not efficient.

1. Lack of flexibility:

The current system may not be able to adapt to changes in the business environment or new market trends.

1. Lack of compliance:

The system may not be designed to meet the industry regulations or standards, which could make the company non-compliant and subject to penalties.

1. Scalability:

current software system may not be able to handle an increasing number of users or transactions, and may not be able to scale easily to meet the growing needs of the company.

1. Technological advances:

The company may want to take advantage of newer technologies or programming languages that can improve the performance or functionality of the software system.

1. Security:

The software system may not be secure enough to protect the company's data and assets from cyber threats.

1. Maintainability:

The current software system may be difficult to maintain and update, making it costly to fix bugs or add new features.

1. Flexibility:

The current system may not be adaptable to the changing business needs or new requirements of the company.

1. Technical debt:

The current software system may have accumulated technical debt over time, making it harder and more expensive to maintain and update.