S&SE – Software and Systems Engineering

Assignment #1: Systems Engineering

Group 4

Farhaj Kazmi

Muhammad Mustafa

Sitara Khurram

9/15

Task 1:

Requirement	Functional/ Non-Functional	Mandatory/ nice to have	Errors
The laptop is better if it is smaller.	Non-Functional ✓	Nice to have ✓	Requirements are not specific for what is considered smaller and better. ✗
The energy consumption during office use has to be smaller than 50 Watts.	Non-Functional ✓	Mandatory ✓	The condition for the requirement is clearly defined. ✓
The laptop has to have a means for wireless communication.	Functional ✓	Mandatory ✓	More precise details regarding the required wireless communication can be defined. ✗
The laptop should not get too warm when in use.	Non-Functional ✓	Nice to have ✗	No specific temperature or the standards by which to measure the warmth are defined. ✓
The color of the laptop is not important.	Non-Functional ✗	Nice to have ✗	The requirement does not provide any standard for evaluation. ✗

Task 2:

5/5

By just looking at individual parts of the system, it is nearly impossible to predict the overall behavior of the system. The reason is that emergent properties are not only dependent on the individual parts of the system but also on how well the components fits together and interact with each other. Therefore, even if individual components inside a system are reliable, if these components do not work well together, then the overall system is not reliable. In conclusion, having in-depth knowledge about each component is not enough to estimate the overall reliability of the system.

Task 3:

3/5

2.1

System integration is one of the major parts of the systems development as it allows the gathering of all the components that are developed separately to work and perform together to see the functionality of the whole system. Many factors that make integration critical are.

1. **Checking Operability:** System integration allows to check if all the components of the system are connected and communicated effectively or not.
2. **Verifying System Functionality:** With the help of integration, it allows the whole system to be tested. So, using this verifies if all the requirements and objectives are met that are desired from the system or not.
3. **To identify any Inconsistencies:** All the defects, issues, and inconsistencies during the development of individual components can be identified using system integration before they are deployed.

4. **Scalability and Adaptability:** This is one of the most important things as the system should be capable of adapting to new features and functionality in the future without any disruption.

Moreover, problems do arise at the stage of integration due to many reasons.

1. One of the reasons is that the development of different components is done by different teams so coordination between the teams becomes challenging. Moreover, every team has its own coding style and development methodologies.
2. Secondly, requirements do change during the development process which leads to discrepancies between the actual requirement and the modified requirement. Which is then catered during the integration phase.
3. Thirdly, incomplete documentation, which can severely impact the components of the interface and can become a blocker for testing as well. As documents consist of complete process of how each functionality will work.
4. Lastly, limitation of testing environment is a big issue as environments often don't fully replicate the production environment due to which functionalities cannot be testing completely with keeping real world setting in mind.

Task 4:

a.

We have these categories (connection, microcontrollers, and cable) from which we can chose, so ***the dimensions of design space are 3***.

b. Variants:

Connection = WIFI or Ethernet or CAN = 3

Microcontrollers = Cortex-MO, Cortex-M3 or Cortex-R = 3

Cable = 1 or 2 or 3 or 5 meters = 4

So, 3 x 3 x 4 variants in this case.

However, if CAN or Ethernet is not chosen from the connection option, then the cable option is not required or irrelevant.

In that case,

3.1

Connection = WIFI = 1

Microcontrollers = Cortex-MO, Cortex-M3 or Cortex-R = 3

Cable = 0

So, 1 x 3 are the possible variants.

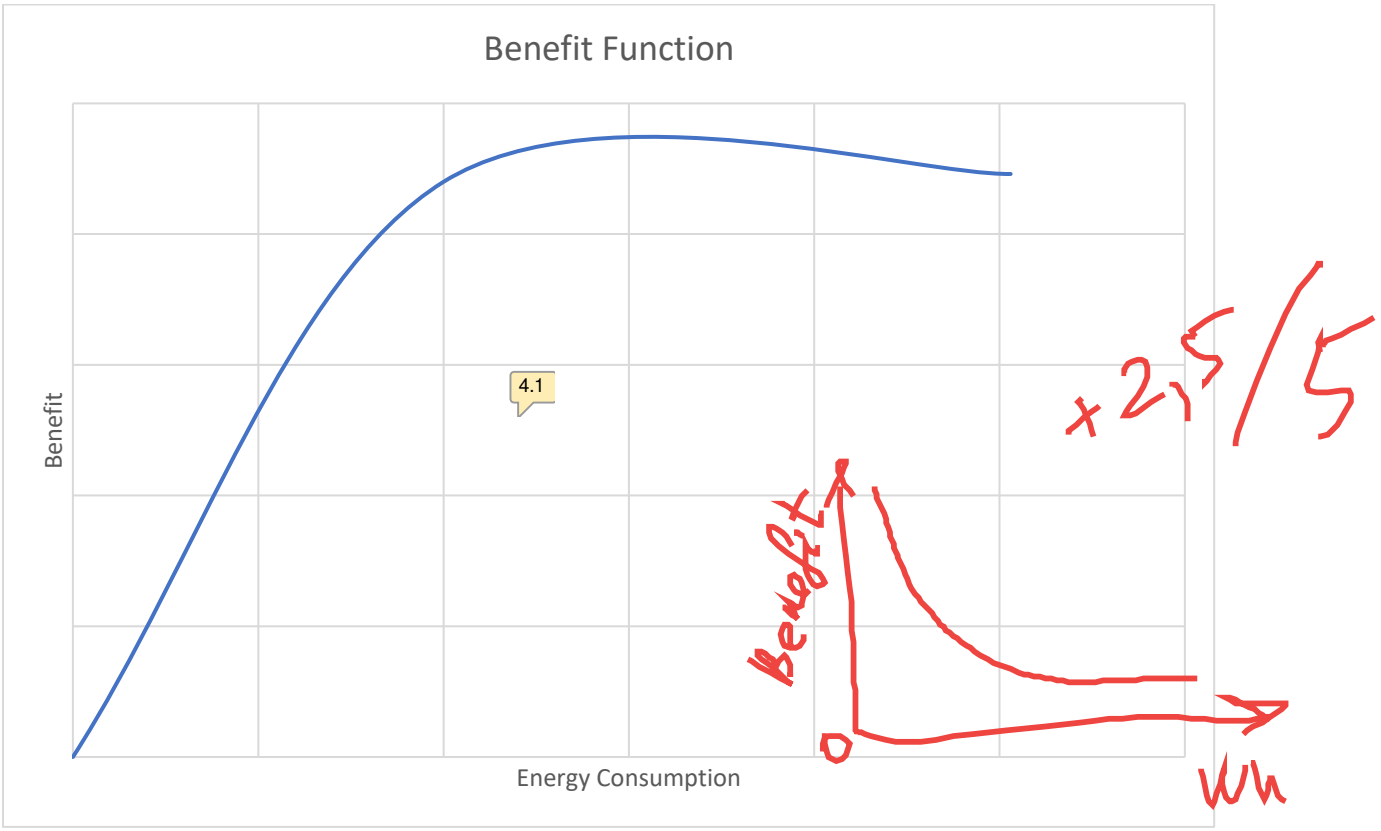
Total (3x3x4) + (1x3) = 39

+7.5/10

c.

If connection type is ‘WIFI’, Choosing the length of the cable is irrelevant.

Task 5:



a.

Consider x-axis as the energy consumption per hour and the y-axis as the benefit derived from the device's performance.

As energy consumption decreases, the benefits increase, but the rate of increase declines, and the form becomes a concave shape. This is to say that reducing energy consumption initially provides major benefits like longer battery life, cost savings but these advantages diminish when consumption approaches very low levels.

b.

Non-monotonic benefit functions are uncommon due to complex relationships between input changes and benefits. They are rare for non-functional aspects of systems because simpler, predictable functions are preferred as monotonic functions on the other hand show consistent improvement or decline with changes in parameters thus make it easier to understand, optimize, and make decisions about choices in system design. Additionally, they make testing and implementation easier with overall system reliability.

x5/5

Task 6:

Requirements	Details	Weight	Variant 1 Client-Server Solution (Server simulations)	Variant 2 Client-Server setup (Graphics)	Variant 3 Peer-to-peer solution (No central Server)
R1: Real-time player Interaction	The system should provide real-time and seamless interaction with the players.	Weight 30	High Score of each Variant From 1-5 R1: 4	High Score of each Variant From 1-5 R1: 4	High Score of each Variant From 1-5 R1: 4
R2: 3D Graphics	To provide high-quality visual graphics to the users.	Weight 50	High R2: 5	High R2: 5	Moderate R2: 3
R3: Bandwidth Usage	To efficiently use the network bandwidth to provide a smooth experience to the user.	Weight 20	Moderate R3: 2	High R3: 4	High R3: 4
Sum		**100**	**(30x5) + (50x5) + (20x2)** **= 440**	**(30x4) + (50x5) + (20x4)** **= 450**	**(30x4) + (50x3) + (20x4)** **= 350**

Benefit Analysis:

For the Client-Server (Server Simulation) variant, the real-time interaction is high due to centralized simulations and graphic quality should be high too as it works with 3D graphics. However, bandwidth usage is moderate as data is sent over the internet. The overall benefit is 440.

For the Client-Server (Graphic) variant, both real-time interaction and graphic quality are high as computation is to be done on the client's PC. Moreover, Bandwidth usage is high due to the need for frequent updates on the server. The overall benefit is 450.

For the Peer-to-Peer variant, real-time interaction is high, and bandwidth is also high because it requires synchronization between user PCs. The overall Benefit is 350.

+10/20

Conclusion:

The Client-Server (Server Simulation) variant has the highest overall benefit, emphasizing on the importance of real-time interaction and graphics.

5.1

Index of comments