|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Sr | task | Estimated Duration (weeks) | Dependencies | ES | EF | LS | LF | Slack |
| 1 | A | 2 | (A) can happen in the same time with (B) | 0 | 2 | 0 |  |  |
| 2 | B | 1 | (B) can happen in the same time of (A) |  |  |  |  |  |
| 3 | C | 7 | C()Can happen after finishing (A) |  |  |  |  |  |
| 4 | D | 5 | (D) can start after finishing (C) |  |  |  |  |  |
| 5 | E | 9 |  |  |  |  |  |  |
| 6 | F | 10 |  |  |  |  |  |  |
| 7 | G | 6 |  |  |  |  |  |  |
| 8 | H | 7 |  |  |  |  |  |  |
| 9 | I | 8 |  |  |  |  |  |  |
| 10 | J | 2 |  |  |  |  |  |  |
| 11 | K | 2 |  |  |  |  |  |  |
| 12 | L | 2 |  |  |  |  |  |  |

Question 1

Question 2

I chose 1. Vehicle manufacturing and industrial processes. And

2. Human Resources Management.

A) stakeholders and actors

Application 1: Vehicle manufacturing and industrial processes Stakeholders:

1. Vehicle manufacturers (stakeholder)
2. Workers in the manufacturing plant (stakeholder)
3. Customers of the manufactured vehicles (stakeholder)
4. Automated production line robots (actor)
5. Quality control inspector (actor)
6. Maintenance technician (actor)

Application 2: Human Resources Management Stakeholders:

1. Human Resources department (stakeholder)
2. Employees (stakeholder)
3. Management (stakeholder)
4. HR administrator (actor)
5. Recruiter (actor)
6. Employee (actor)

B) user story

Application 1: Vehicle manufacturing and industrial processes

1. An automated production line robot must be able to assemble vehicle parts accurately and efficiently according to instructions.
2. A quality control inspector must be able to easily access and review production data to ensure that vehicles are being manufactured to meet quality standards.
3. A maintenance technician must be able to identify and fix any issues with the production line robots quickly and easily.

Application 2: Human Resources Management

1. An HR administrator must be able to easily manage employee information and track employee performance.
2. A Recruiter must be able to easily access and manage candidate resumes and information for current open positions.
3. An Employee must be able to easily access and update their personal information and view their performance reviews.

C) use cases

D) identifying which one is the best between (MTTF, MTBF, ROCOF, POFOD or Availability)

Application 1: Vehicle manufacturing and industrial processes

1. An automated production line robot must be able to assemble vehicle parts accurately and efficiently according to instructions.

• The optimum reliability metric for this user narrative is MTTF, which calculates the average interval between robot breakdowns. This statistic is crucial since it enables producers to comprehend how frequently and how quickly a robot will break down.

2. A quality control inspector must be able to easily access and review production data to ensure that vehicles are being manufactured to meet quality standards.

• The most trustworthy dependability metric for this user story is MTBF. It is crucial to understand the average time between system failures, which is gauged by this statistic, in order to ascertain how frequently the system will be unavailable for use.

Application 2: Human Resources Management

1. An automated production line robot must be able to assemble vehicle parts accurately and efficiently according to instructions.

* Availability is the best reliability metric for this user story. It measures the percentage of time the system is available for use, which is important for understanding how often the system is down and unavailable to users.

1. A Recruiter must be able to easily access and manage candidate resumes and information for current open positions.

* The most appropriate dependability metric for this user experience is ROCOF, which gauges how quickly a system's frequency changes over time. This statistic is crucial to comprehending the stability of the system since abrupt variations in frequency might be an indication of instability or other issues.

E) usability requirements for the user stories

Application 1: Vehicle manufacturing and industrial processes

1. An automated production line robot must be able to assemble vehicle parts accurately and efficiently according to instructions.

• Usability requirement 1: For maintenance personnel, the robot's interface must be straightforward and uncomplicated.

• Usability requirement 2: For convenient maintenance, the robot must be remotely controllable.

• Usability requirement 3: The robot must be simple to program for various assembly procedures.

1. A quality control inspector must be able to easily access and review production data to ensure that vehicles are being manufactured to meet quality standards.
   1. Usability requirement 1: The user interface of the system must be straightforward to use and enable rapid searches for relevant information.
   2. Usability requirement 2: The system must be able to deliver easily comprehensible reports on production data.
   3. Usability requirement 3 states that the system must be simple to use across a range of devices.
2. An HR administrator must be able to easily manage employee information and track employee performance.

* Usability requirement 1: The system's interface must be easy to navigate and search through for relevant data.
* Usability requirement 2: The system must be able to generate clear and easy-to-understand reports on employee performance and attendance.
* Usability requirement 3: The system must be accessible from multiple devices for easy access.

Question 3

1. Black box testing table

| **Test Case** | **num1** | **num2** | **Expected Output** |
| --- | --- | --- | --- |
| 1 | 50 | 100 | 50 |
| 2 | 75 | 100 | 75 |
| 3 | -25 | 100 | -1 |
| 4 | 50 | -100 | -1 |
| 5 | 0 | 100 | -1 |
| 6 | 50 | 0 | -1 |

The code is in a (python file) called q3PartA.py

1. (BVA)

| **Category** | **Test Data** | **Expected Result** |
| --- | --- | --- |
| Temperature < 300 | 299 | Danger: temperature is below 300 degrees. |
| Temperature >= 300 and temperature < 650 | 300 | Warning: temperature is between 300 and 650. |
| Temperature > 300 and temperature <= 650 | 650 | Warning: temperature is between 300 and 650. |
| Temperature > 650 and temperature < 800 | 700 | The reactor is operating within the standard range of 650 to 800. |
| Temperature >= 650 and temperature <= 800 | 800 | The reactor is operating within the standard range of 650 to 800. |
| Temperature > 800 and temperature < 950 | 900 | The reactor is operating within the standard range of 800 to 950. |
| Temperature >= 800 and temperature <= 950 | 950 | The reactor is operating within the standard range of 800 to 950. |
| Temperature < 1100 and Temperature > 950 | 1000 | Warning: temperature is above 950 but below 1100. |
| Temperature <= 1100 and Temperature >= 950 | 1100 | Warning: temperature is above 950 but below 1100. |
| Temperature > 1100 | 1101 | Danger: temperature is above 1100. |

The code in a (python file) called q3PartB

1. Consider the following python code to find the median of 3 numbers, develop a test design for white box testing.

| **Test Case** | **Input** | **Expected Output** |
| --- | --- | --- |
| Test Case 1 | 42 | "You got it!" |
| Test Case 2 | 0 | "Too low" |
| Test Case 3 | 100 | "Too high" |
| Test Case 4 | 50 | "Too low" |

The code is in (python file) called Q3PartC

Question 4

* 1. The table of the issues of the functions

| **Function Name** | **Identified Issue** | **Category of Issue** | **Comments on how it is an issue** | **Proposed Solution** |
| --- | --- | --- | --- | --- |
| create\_beasts | Global Variables | Global State | Global variables make the code hard to understand and maintain, and can introduce bugs when used improperly | Refactor the code to pass data between functions as arguments and return values |
| attack | Function Signature | Code Duplication | The function's arguments are repeated across multiple functions which can lead to inconsistencies and duplicated code | Use a class to store the relevant information and pass an instance of the class to each function that needs it |
| play\_game | Input | User Input Validation | The input() function does not validate the user input, which can lead to errors | Validate the user input before processing it and provide a user-friendly error message |
| All functions | Function Names | Function Naming | The function names are not descriptive enough, making the code hard to understand | Rename the functions to have descriptive names that accurately describe their purpose |

أعلى النموذج

أسفل النموذج

* + 1. Demonstrate how the flag(s) used in the code provided are affecting the modularity, what specific kind issues are introduced using flag(s) in the code.

The global variables in this code affects the modularity.

The functions in the code are closely connected because of the use of global variables. The functions are not independent since they depend on certain variables. The code is challenging to test since it uses global variables. To properly test the code, it is vital to keep track of the status of the global variables since their value impacts the output of the functions. The use of global variables makes code maintenance more difficult. If a global variable is changed in one section of the code, it could influence other sections of the code that rely on that variable. This makes it more challenging to find and solve faults.

* + 1. It’s in (python file) called Q4
    2. Git is done