|  |
| --- |
| **1. Getting started** |

*What is the* ***purpose*** *of the program? Who are your* ***end users****?*

|  |
| --- |
|  |

*What are the program* ***requirements****? What are the requirements and needs of your end users? Read the brief, then summarise the key points. What will your MVP (minimum viable product) look like based on the problem and the context you were given?*

|  |
| --- |
| * *Must have a* * *Must have* * *Must have* * *Must have* * *Must present* |

***Decomposition****:*

You are encouraged to break down your program into a list of components and work through each component before assembling them together into a final program. Remember that a component is a sub-task in your program with a specific goal. It can be one function or multiple functions. At this level, between 3 and 6 components are enough.

|  |
| --- |
| **Component One:** Input Variables and Constants  -  **Component Two:** Output Variables and Constants  -  **Component Three:** Libraries/Modules needed   * e.g. Time   **Component Four:** Functions   * Introduction/Welcome function   **Component Five**: Collections (lists, dictionaries, etc.) and/or External Files needed  **Component Six:** Display Final |

***What project management and version control tools will you be using****? This is encouraged, not assessed for this project:*

|  |
| --- |
| Link to your Trello/Notion board: |

|  |
| --- |
| **2. Developing the components and the program iteratively:**  You must keep a **daily log** to provide evidence of how the components have been developed, trialled, tested and debugged. Evidence could include notes of user feedback, before and after screenshots, testing tables with annotated screenshots and/or screencastify videos.  **For each component:**   * briefly **describe** the component, e.g. what does it do, what do you want to achieve   **test and debug** your code, including what you are testing, sample inputs, expected outcome, actual outcome, issues to be fixed, and how you fixed it.   * **trial** with end users, if needed, and collect their **feedback** * **update** the Trello board and **paste** a screenshot |
|  |
|  |
|  |
|  |
|  |
|  |
| Final Testing Video – demonstrate that your program is working as intended. – insert link here |

|  |
| --- |
| **3. Final Testing** |

*This is testing for the* ***entire program*** *(once all components are developed and assembled together). Your testing at this point should focus on how the entire program functions, rather than individual inputs. Make improvements and test again if needed. Include testing for* ***expected****,* ***boundary*** *and* ***invalid*** *inputs (if your program can handle them).*

|  |  |  |  |
| --- | --- | --- | --- |
| **What input are you testing?** *Examples in green* | **Sample Inputs** | **Expected Outputs** | Works/Fixed  ✔️ |
| *Name - Expected* | *Name = “Pika”* | *Accepted* | *✔️* |
| *Name - Boundary* | *Name = “Pika143!”* | *Error Message, Retry* | *✔️* |
| *Movie Pick - Invalid* | *Movie = “black panther”* | *Retry* | *✔️* |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

*Add extra rows as needed.*

*Paste screenshots or links to testing videos below that show your program works with the sample inputs above.*

**4. Version Control**

***Version Control Evidence:*** *This could be in the form of annotated screenshots which show how you managed this process, or you could make a brief screencast explaining how you implemented version control. This can also be done automatically through GitHub – explain how you have achieved this.*

|  |
| --- |
| Version one was  Version two focus was  Version three focus was  Version four purpose was  Snapshot of FOLDER |