|  |
| --- |
| **Name:**  Multipleinput issue |
| **Description:**  The system must support several types of input simulations as well as being able to add new types if the costumer requires it.  **Factors:** |
| **Solution:**  Making the interface between the module handling input and the sub modules for different input types work the same no matter what input type it is. |
| **Strategies/Tactics: "***Generalize the module" "Software Architecture in Practice Second Edition"* Chapter 5.3 Len Bass, Paul Clements, Rick Kazman 2003 |

|  |
| --- |
| **Name:**  Multiplehardware emulations issue |
| **Description:**  The system must support several types of hardware emulations as well as being able to add new types if the costumer requires it.  **Factors:** |
| **Solution:**  Explore standards for hardware communications currently used or in development to support most hardware emulations without impacting the system. |
| **Strategies/Tactics:**  **"***Anticipate expected changes" "Software Architecture in Practice Second Edition"* Chapter 5.3 Len Bass, Paul Clements, Rick Kazman 2003 |

|  |
| --- |
| **Name:**  Multiplesoftware emulations issue |
| **Description:**  The system must support several types of software emulations as well as being able to add new types if the costumer requires it.  **Factors:** |
| **Solution:**  Explore standards for software communications currently used or in development to support most hardware emulations without impacting the system. |
| **Strategies/Tactics:**  **"***Anticipate expected changes" "Software Architecture in Practice Second Edition"* Chapter 5.3 Len Bass, Paul Clements, Rick Kazman 2003 |

|  |
| --- |
| **Name:**  Multipletesting techniques issue |
| **Description:**  The system must support several types of testing techniques as well as being able to add new types if the costumer requires it.  **Factors:** |
| **Solution:**  Scripting  Keeping the semantics of the testing modules coherent so that further testing techniques can be added with minimal changes to the current structure |
| **Strategies/Tactics:**  "Maintain semantic coherence" and **"***Anticipate expected changes" "Software Architecture in Practice Second Edition"* Chapter 5.3 Len Bass, Paul Clements, Rick Kazman 2003 |

|  |
| --- |
| **Name:**  Not crashing with tested system issue |
| **Description:**  The MIB needs to be able to standby further testing if the tested system crashes until it has been restarted from a earlier point. This involves putting the input feed on hold.  **Factors:** |
| **Solution:** |
| **Strategies/Tactics:** |

|  |
| --- |
| **Name:**  Report creation issue |
| **Description:**  The MIB needs to be able to create a report once the testing is done. This report must contain data from both the output component as well as the log for all the test data.  **Factors:** |
| **Solution:** |
| **Strategies/Tactics:** |

|  |
| --- |
| **Name:**  Keeping the system running through a test crash |
| **Description:**  The MIB needs to be able to standby further testing if the tested system crashes until it has been restarted from a earlier point. This involves putting the input feed on hold.  **Factors:** |
| **Solution:** |
| **Strategies/Tactics:** |

|  |
| --- |
| **Name:**  Running system on all standard computers with required performance |
| **Description:**  The MIB needs to be able to run on all standard computers on the market which have enough performance to run both the MIB itself and the tested system.  **Factors:** |
| **Solution:**  Making the MIB cross-platform compliant |
| **Strategies/Tactics:** |