**System implementation, testing and validation report for <project name>**

*(Formatting is important: Please use Times new Roman, Font 12, spacing should be 1.5, justified alignment, All chapters Must start on a new page, you should have a list of tables, list of figures if any, Table of contents. Table captions must appear at the top of the table, figure captions must be placed at the bottom of the figure, page sections before chapter 1 must be numbers as ii,iii etc, Do not show the page number on the first page )*

|  |  |
| --- | --- |
| Document No: |  |
| Prepared by: | <Authors> |
| Date: | DD-MMM-YYYY |
| Version: | 1.0 |

Document Approval

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Role** | **Date** | **Signature** |
| <Author> | Author(s) |  |  |
|  | Validation |  |  |
|  | Client |  |  |

**Table of Contents**

**List of Figures**

**List of Tables**

**Chapter 1: Introduction**

**1.1 Background and scope of the project**

**1.2 Overview of the document**

This document describes the implementation, testing and validation findings for the xxx system. It is divided into the following sections:

Section 1: This section gives an overview of the document

**2. System Specifications**

*The section describes and specifies the system completely and is the basis for the validation process.*

**2.1 Version of requirements and Version Control**

*Version of, and changes applied to, the requirements specification. Specify the version of your requirements document. Remember you used to make changes. If you made changes between versions, please specify which changes were made to get which versions. Also, explain why the changes were made*

*How to identify different versions of the computer system and to distinguish output from the individual versions.if you have different versions of your software, how do you know which version is 1.0, or 1.2?? Did you use tools like GIT, TOTROISE ETC to manage versioning of your code?*

**2.2 Input**

*All inputs the computer system will receive. Include ranges, limits, defaults, response to illegal inputs, etc. Please explain in detail what your inputs are for example, if one has a security system GPS coordinates:*

*Input 1: The coordinates are received from the GPS system. When one is attached, the system picks the coordinates and extracts the corresponding locations name and …. which is sent to all phone numbers in the contact list.*

*Input 2: (please don’t simply state the inputs, go ahead and explain them so that the reader fully understands your system)*

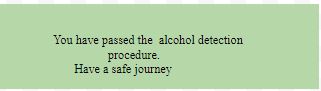
Input 1: Amount of light going through the skin. When this is recorded, the system determines whether the alcohol content in a person’s blood is above acceptable level or not.

**2.3 Output**

*All outputs the computer system will produce. Includes data formats(eg. Images, text, video, etc.), screen presentations, data storage media, printouts, automated generation of documents, etc. do a thorough explanation of all the outputs here*

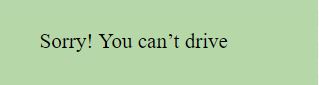
Output 1: User Alcohol Level Acceptable

On successful detection and acceptable level detected, the above message is displayed.

****

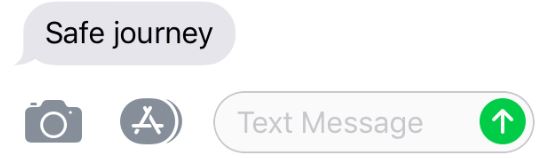
Output 2: User Alcohol Level Unacceptable

When the alcohol level detected is beyond that accepted, the user will be view the message below.

****

Output 3:SMS Notification

The user will receive such an SMS regarding the test results.

****

**2.4 Functionality**

*All functions the computer system will provide. Includes performance requirements, such as data throughput, reliability, timing, user interface features, etc. Explain all the functionalities your system in detail.*

**2.5 Limitations and safety**

*All acceptable and stated limitations in the computer system. What limits the use of the system. Please identify the limitations of the system* ***during its use*** *and NOT during development. Things like The system requires an active Internet connection. If there is no Internet connection, messages sent from users cannot be received. Etc. For embedded systems, they have energy/power limitations, web systems have …*

*All precautions taken to prevent overflow and malfunction due to incorrect input or use. When the system is in use, such and such must be done to prevent incorrect input or use of the system. Eg The administrators shall enter all eligible districts. This is to ensure that only eligible villages are entered into the system. (Give others)*

**2.6 Default settings**

*All settings applied after power-up such as default input values, default instrument or program control settings, and options selected by default. Includes information on how to manage and maintain the default settings. EG By default, when the system is first installed, it contains only one user (give the username and password). For security reasons, the password can be changed via (give the menu name). etc.*

**2.7 Special requirements**

*Requirements the laboratory is committed to, security, confi­dentiality, change control and back-up of records, protection of code and data, precautions, risks in case of errors in the computer system, etc.*

**2.8 Errors and alarms**

*Identify the errors that could arise from the use of the system. How would you handle the errors?*

*C***hapter 3: Design output**

**3.1 Implementation (coding and compilation)**

*Development tools used to implement the system, notes on anomalies, module and integration details, etc.*

*All device interfaces and equipment to be supported. Identify the devices and explain how they are used in your system. For those doing embedded systems, you have sensors, key pads etc, mobile applications, you have mobile phones, web, you have computers …*

*The hardware and software operating environment in which to use the computer system. E.g. laboratory or office computer, the actual operating system, network, third-party ex­ecuta­bles such as Microsoft® Ex­cel and Word, the actual version of the platform, etc.*

The devices used are shown and listed below.

Spark Fun Triad Spectroscopy Sensor - AS7265x (Qwiic)

GSM module

LCD display

Power Supply

LED Conductor

|  |
| --- |
| **3.4 Documentation**  *What Kind of Documentation has been provided as output from the Design and what is its role to the readers?* |
| ***Table No: Design details (check all that apply to your project. Make sure you can defend what you tick )*** |

| *Topics* | **Design output** | |
| --- | --- | --- |
| **Good programming practice**  *Efforts made to meet the recommendations for good programming practice...* | Source code is... | Source code contains... |
| **Windows programming**  *If implementing Windows applications... remove this row* | Comments: | |
| **Dynamic testing**  *Step-by-step testing made dynamically during the implementation...* | Comments: | |

## Chapter 4: Inspection and testing

**4.1 Introduction**

*The inspection and testing of the computer system is planned and documented in a test plan. The ex­tent of the testing is in compli­ance with the requirements, the system acceptance test specification, the approach, complexity, risks, and the in­tended and expected use of the computer system. (Check what applies)*

**Table No: Inspection plan and performance**

| *Topics* | **3.3.1 Inspection plan and performance** | *Date / Initials* |
| --- | --- | --- |
| **Design output**  *Results from the Design Output section inspected...* | Comments: | Add dates please and initials of members concerned |
| **Documentation**  *Documentation inspected...* | Comments: |  |
| **Software development environment**  *Environment elements inspected...* | Comments: |  |
| **Result of inspection**  *Approval ofinspection.* | Comments: |  |

**4.2 Test plan and performance**

*The test plan is created during the development or reverse engineering phase and identify all elements that are about to be tested. The test plan should explicitly describe what to test, what to expect, and how to do the testing. Subse­quently it should be confirmed what was done, what was the result, and if the result was approved.*

|  |
| --- |
| **4.2.1 Test objectives**  *Description of the test in terms of what, why, and how. Ie why do the test? Steps taken and What was tested* |
| **4.2.2 Scope and Relevancy of tests**  *In terms of coverage, volumes, and system complex­ity. Relative to objectives and required operational use.* |
|  |
| **4.2.3 Levels of tests**  *Module test, integration test, and system acceptance test.* |
| **4.2.4 Types of tests**  *E.g. input, functionality, boundaries, performance, and us­ability.* |
| **4.2.5 Sequence of tests**  *Test cases, test procedures, test data and expected results.* |
| **4.2.6 Configuration and calculation tests**  *Platform, network, and inte­gration with other systems.*  *Calculation tests confirm that known inputs lead to specified outputs.* |
| **4.3 Precautions** |

**4.3.1 Anomalous conditions**

*When operating in a third-party software environment, such as Microsoft® Windows and Office, some undesirable, inappropriate, or anomalous operating conditions may exist. A dis­crepancy between the description of the way an instrument should operate, and the way it actually does, may be regarded as an anomaly as well. Minor errors in a computer system may sometimes be acceptable if they are documented and/or properly cir­cumvented*. *Anomalous operating conditions associated with the computer system itself, Anomalous operating conditions in the used instruments.*

**4.3.2** Precautionary **steps taken**

*The steps taken to workaround anomalous, inappropriate, or undesired operating conditions are verified and tested. Precautionary steps taken in e.g. Windows settings, Precautionary steps taken to workaround problems with the used instruments, Precautionary steps taken to workaround problems with the computer system itself.*

## Chapter 5: Installation and system acceptance test

*The validation of the installation process ensures that all system ele­ments are properly installed in the host system and that the user obtains a safe and complete installation, especially when installing software products.*

**5.1** **Input files**

*List of (relevant) files on the installation media and what they are used for*

|  |
| --- |
| **5.2 Supplementary files**  *Readme files, License agreements, examples, etc.* |
| **5.3 Installation qualification**  *Steps to ensure and document that each component is installed correctly.* |

**Table no: Checklist of the Installation and system acceptance test**

| *Topics* | **Installation summary** |
| --- | --- |
| **Installation method**  *Automatic or manual installation...* | Comments: |
| **Installation media**  *Media containing the in­stallation files...* | Comments: |
| **Installed files**  *List of (relevant) installed files, e.g. EXE- and DLL-files, spreadsheet Add-ins and Templates, On-line Help, etc.* | * DBK files * HEX files * PWI files * Plg files * BAK files * Opt files * C files |

**Table no: Installation Procedure Check**

| *Topics* | **Installation procedure** | *Date / Initials* |
| --- | --- | --- |
| **Authorization**  *Approval of installation in actual environment.* | Person responsible: |  |
| **Installation test**  *The following installations have been performed and approved...* | Comments: |  |

## Chapter 6: Performance, servicing, maintenance, and phase out

**6.1 Service and maintenance**

*Documentation of service and sup­port concerning maintenance, fu­ture updates, problem solutions, requested modifications, etc.*

**6.2 Performance and Maintenance**

*Which are the requirements for service, maintenance, performance ( Maximum time taken before giving output) , and support(what kind of support do you expect to give to the clients at this stage). This phase is where all activities during performance reside and where deci­sions about changes (What would be the causes of incorporating changes?), upgrades(How should the software be upgraded? ), revalidation, and phase out are made(How do you move from the old system to the new one you have just come up with?? Ie How do you move data from the old system to the new system?). [EXPLAIN THESE IN SOME PARAGRAPHS /PARAGRAPH AND LATER FILL THE TABLE BELOW]*

**Table no: Performance and maintenance details**

| *Topics* | **Performance and maintenance** | *Date / Initials* |
| --- | --- | --- |
| **Problem / solution** | *Detection of system problems causing operating troubles. A first step could be to suggest or set up a well-documented temporary solution or workaround.* | Dates must be filled in |
| **Functional maintenance** | *E.g. if the computer system is committed to in­ternational standards, and these standards are changed, the computer system, or the way it is used, should be updated ac­cordingly.* |  |
| **Functional expansion and performance im­provement** | *List of suggestions and requests, which can improve the performance of the computer system. eg*   * Inclusion of a PH sensor to determine the soil PH levels to allow for automatic neutralization of the soil. * Use of wireless technology instead of wired connections. |  |

# Chapter 7: Conclusion and Recommendations

*Make a conclusion of your whole report*

**Appendix A: User Manual**

*Give details on how the system can be used, where to go for help etc accompany your explanations with screenshots*

(*For those doing embedded systems, please add your assembled diagram, those doing other projects can add some screenshots to their manual. The screenshots must appear in a logical order)*

| **Final approval for use** | | |
| --- | --- | --- |
| Identification: | |  |
| Responsible for validation: | |  |
| Remarks: | | |
| Date: | Signature: | |