**UNIVERSIDAD TECNOLÓGICA DE QUERÉTARO**

**CESEQ**



**Diplomado en Software Embebido**

Proyecto <Titulo>

<File Name: Software Development Plan/SWRA\_YYYYMMDD.docx/etc>

DOCUMENT: Software Development

Document No. #CESEQ\_SDP\_001

Scrum Master: Omar, Bermudez

Developer: Hector, Bojorquez

Developer: Paul, Tapia

Date (YYYYMMDD): 20190405

Version: 1.0.0.

Project Version: x.x.x.

# Log

Document Version

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date (yyyymmdd) | Description | Reviewer |
| 1.0.0. | 20190405 | First release | Pérez, Adbeel |
|  |  |  |  |

Project Document Version

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date (yyyymmdd) | Description | Reviewer |
| x.x.x. | yyyymmdd | Anything | Author and reviewer |
|  |  |  |  |

# Index

Table of Contents

[1. Log 2](#_Toc8215461)

[2. Index 3](#_Toc8215462)

[3. Project Scope 4](#_Toc8215463)

[4. Deliverables 4](#_Toc8215464)

[5. Development methodology 4](#_Toc8215465)

[6. Estimates 4](#_Toc8215466)

[7. Planning 5](#_Toc8215467)

[8. Solving Problem Strategy 6](#_Toc8215468)

[9. Design 6](#_Toc8215469)

[9.1. Standards 6](#_Toc8215470)

[9.2. Naming conventions 6](#_Toc8215471)

[10. Testing 7](#_Toc8215472)

[10.1. Verification strategy (black box test) 7](#_Toc8215473)

[10.2. White box strategy 7](#_Toc8215474)

[10.3. Cyclomatic Complexity Redundance index 7](#_Toc8215475)

[11. Release 8](#_Toc8215476)

[11.1. Software Development Folder 8](#_Toc8215477)

[11.2. Integration Tests Strategy 8](#_Toc8215478)

[11.3. Validation Testing / Functional Testing 8](#_Toc8215479)

[11.4. Throughput and Flash and RAM measurement 9](#_Toc8215480)

[12. Results 9](#_Toc8215481)

[13. Lessons Learned 9](#_Toc8215482)

# Project Scope

Description of the full Project, in case the scope was not reached then it **SHALL** be resized and reflected in this section.

Define the requirement document. Every requirement **SHALL** be enumerated.

This section **SHALL** indicate the stakeholders documents which shall be contained at:

<PATH DEL PROYECTO>\1) Requirements\stakeholder

This section **MUST** be contained in this document or in a different document indicating the path in this section, in case a new document needs to be created then it **SHALL** be contained at:

<PROJECT\_PATH>\ 1) Requirements\3. SWRA\_20190405.xlsx

All the requirements **SHALL** be enumerated.

# Deliverables

Work products like: Code (hex), Documents (Software Requirement Document, Estimates file, Planning file, Design file, verification file, Functional testing file), or hardware if apply (schematic files, PCB file and Gerber file, general draft) **SHALL** be described here.

Include in this section the delivering process and dates if it applies

# Development methodology

In case the team select SCRUM Methodology, it **SHALL** specify the controls like:

* Scrum board,
* Length of the sprint.
* Schedule of the Meetings.
* Positions: Scrum masters, product owner and developers.
* Planning board.

# Estimates

* Estimates **SHALL** contain all the Inputs, like:
  + Hw Facts:
    - Board availabilty
    - Plant availability
    - PC availability
    - Osciloscope
    - Signal generator
    - Multimeter
    - Power supply
  + Activities Facts
    - Human resources
      * Product owner should be the tutor
      * Master scrum and developer (better called as leader)
      * Number of Developers
  + SW Facts
    - Operative system form Renesas works.
  + Hw Assumptions:
    - Hardware damaged.
    - Laboratory time availability.
  + Activtities Assumptions
    - Team time availability.
    - Hardware in good conditions.
  + SW Assumptions
    - Programming language, SW IDE or Hw platform unkown.
    - SW Module unknown.
* It **SHALL** be defined all risks, remember that this is an input for the FMEA:
  + UTEQ holidays.
  + Team is not complete due other projects or trips.
  + New hardware and microcontroller
  + Error in the OS Configuration.
  + Error hardware connection.
* It **SHALL** have a breakdown of all task and activities that are needed and analyzed their dependency between them, some good examples to estimate are:
  + **Activities etimated**
    - **Create and update documents** (design planning verification and so on). Consider the time to create and update documents (SDP, schedule, control code, meetings and peer reviews).
    - **Create, update and execute Verification** **Plan** (white and black test, cyclomatic complexity index calculation, Integration testing, throughput, RAM and FLASH measurement, C99, C11 or other standard evaluation).
  + **SW modules estimated**
    - **Software** **Modules** (RAM, ROM and throughput). Time estimated for each Modules development, it means, they need to reflect the time for every task needed to implement each module like: (UART, I2C or SPI, ADC, PWM, HMI, PID Algorithm implementation, Operative system implementation, etc).
  + **Hw Modules estimated**
    - **Hardware modules** (devices like pc, debugger, board, plant, etc).

# Planning

* It **SHALL** contain the roll definitions of the team members and their responsibilities.
* All the tasks from estimates section **SHALL** be reflected into the Schedule and assigned to the team. Every task **SHALL** contain the definition of done.
* Remember that any document created, updated, White/Black test execution, Integration testing execution, meetings etc, **SHALL** be contained in this section as part of the activities of the plan.

This section **MUST** be contained in this document or in a different document linked to this section, the new document SHALL be contained at:

<PROJECT\_PATH>\2) Planning\7. Planning\_20190405.xlsx

# Solving Problem Strategy

* This section SHALL contain an FMEA for the full Project considering the sw functionalities defined in the risk analysis from the estimates section.
* In case an error be detected during the development stage, this section SHALL contain a mitigation plan including the 5 whys methodology for hw, sw and document issues.

This section MUST be contained in this document or in a different document linked to this section, the new document SHALL be contained at:

<PROJECT\_PATH>\2) Planning\8. DFMEA\_20190405.xlsx

# Design

This section **SHALL** contain Static and dynamic modeling diagrams like: block diagram flow diagram, call tree diagram, state machine diagram, sequence diagram and others depending on the programming paradigm.

Additionally, this section SHALL contain control diagram where is defined the: inputs, outputs, noise, and its feedback (if apply).

This section MUST be contained in this document or in a different document linked to this section, the new document SHALL be contained at:

<PROJECT\_PATH>\3) Design\9. SoftwareDesignDocument\_20190405.docx

Sections 9.1. and 9.2. MUST be contained in this document or MUST be divided into different documents. With the naming defined in every section.

## Standards

In case C89-C90, C11 or other standard be used, it SHALL be specified in this section and additionally add the link to the standard used.

Additionally, the tool used to evaluate the standard SHALL be defined here if apply.

This section MUST be contained in this document or in a different document linked to this section, the new document SHALL be contained at:

<PROJECT\_PATH>\3) Design\ 9.1. SoftwareStandards\_20190405.docx

## Naming conventions

The tags SHALL be defined for: local and global variables, local and global functions, macros, enumerations and structures.

It is **SUGGESTED** to use capital letter for global variables and macros.

In case prefix be used, it is SUGGESTED to consider for variable type, module or file, for example:

uint8\_var1

adc\_variable1

etc.

File names SHALL have a convention defined in this section, for instance: first letter SHALL be capital.

For folder in code, it SHALL be defined the names or conventions used.

This section MUST be contained in this document or in a different document linked to this section, the new document SHALL be contained at:

<PROJECT\_PATH>\3) Design\9.2. NamingConventions\_20190405.docx

In code comments, It SHALL contain the requirements which is implemented with the code described.

# Testing

## Verification strategy (black box test)

This section SHALL be contained at:

<PROJECT\_PATH>\4) Verification\10.1. BlackboxTest\_baseline.docx

…and its results SHALL be located with the date as suffix, as following is indicated:

<PROJECT\_PATH>\4) Verification\Results\10.1. BlackboxTest\_20190405.docx

Every time a module or feature is implemented, it SHALL contain their tests section and SHALL be contained with the reference to the requirement number in order to have traceability.

## White box strategy

It SHALL define the software which is going to be used, for instance: gtest, junit, sunit, etc.

A document baseline SHALL be created as a reference for all the project implementation. This document SHALL be located at:

<PROJECT\_PATH>\4) Verification\10.2. WhiteboxTest\_baseline.docx

…and its result SHALL be located at:

<PROJECT\_PATH>\4) Verification\Results\10.2. WhiteboxTest\_20190405.docx

Every time a module or feature is implemented, every test case SHALL contain a reference to the requirement number in order to have traceability.

## Cyclomatic Complexity Redundance index

<This section is optional>

This section MUST be contained in this document or in a different document linked to this section, the new document SHALL be contained at:

<PROJECT\_PATH>\4) Verification\ 10.3. CCRI\_20190405.docx

…in case this section is implemented, then its result SHALL be located at:

<PROJECT\_PATH>\4) Verification\Results\10.3. CCRI\_20190405.docx

# Release

Firmware version number SHALL be defined in this section, and the strategy used for that, an example MUST be:

Naming convention for delivered work products like: code and documents shall be defined in this section, the name shall be kept for those documents that SDP describes.

Date/Hw version/Sw version

20190405/001/ 001

The code shall be controlled in GITHUB and path shall be defined here.

## Software Development Folder

The path for software development folder shall be defined in this section and be contained and controlled at GITHUB previous to the final release.

## Integration Tests Strategy

This section SHALL be contained in the planning and reflected in the schedule.

IT **SHALL** be defined a document baseline as a reference for all the project implementation. This document **SHALL** be located at:

<PROJECT\_PATH>\4) Verification\11.1. IntegrationTesting\_baseline.docx

…and its RESULT SHALL be located at:

<PROJECT\_PATH>\4) Verification\Results\11.1. IntegrationTesting\_20190405.docx

Every time a module or feature is implemented, every test case SHALL contain a reference to the requirement number in order to have traceability.

This test MUST contain the plant connected or not.

## Validation Testing / Functional Testing

This section SHALL be contained in the planning and reflected in the schedule.

IT **SHALL** be defined a document baseline as a reference for all the project implementation. This document **SHALL** be located at:

<PROJECT\_PATH>\4) Verification\11.2. ValidationTesting\_baseline.docx

…and its RESULT SHALL be located at:

<PROJECT\_PATH>\4) Verification\Results\11.2. ValidationTesting\_20190405.docx

Every time a module or feature is implemented, every test case SHALL contain a reference to the requirement number in order to have traceability.

This test SHALL contain the plant connected.

## Throughput and Flash and RAM measurement

This section SHALL be contained in the planning and reflected in the schedule.

It SHALL define the RAM, Flash and Throughtput measurements strategy at:

<PROJECT\_PATH>\4) Verification\ 11.3. ThroughputRAMFlash\_procedure

# Results

All pictures, videos or miscellaneous SHALL be posted at:

<PROJECT\_PATH>\5) Results

# Lessons Learned

All comments, feedback or others SHALL be documented in this section.