**INDEX Page**

[1. Introduction 1](#_Toc53048142)

[2. Scope 2](#_Toc53048143)

[2.1. Out of scope 2](#_Toc53048144)

[2.2. Software Safety Classification 3](#_Toc53048145)

[3. Definitions 3](#_Toc53048146)

[4. Roles and responsibilities 3](#_Toc53048147)

[5. Reference Documents 4](#_Toc53048148)

[5.1. System references 4](#_Toc53048149)

[5.2. DevelopmentStandards, Methods and Tools and Regulatory references 5](#_Toc53048150)

[6. Processes used in the development 7](#_Toc53048151)

[7. Software Deliverables 7](#_Toc53048152)

[8. Traceability 7](#_Toc53048153)

[9. Software Configuration List 8](#_Toc53048154)

[10. SOUP Configuration Items 8](#_Toc53048155)

[11. Software Integration and Testing Plan 8](#_Toc53048156)

[12. Software risk management 9](#_Toc53048157)

[13. Project deliverables 10](#_Toc53048158)

[14. Common Software Defects 10](#_Toc53048159)

[15. Appendix 10](#_Toc53048160)

[16. References and related documents 11](#_Toc53048161)

[17. Signatures and approval 12](#_Toc53048162)

[18. History 12](#_Toc53048163)

**General note:**

This document is based on the template for a Software Development Planning, as proposed in the IEC 62304:2006, Section 5.1.

In order to prevent deviations from this template all chapters and paragraphs are displayed. When a chapter or paragraph is not applicable to the subject described in this Software Development Planning, this will be recorded as NA.

# Introduction

The purpose of this software development plan is to establish the safety and effectiveness of using the R PACKAGE DAS-COMBAT for the data analysis diagnostic measurements. This plan, and related documents, will document the intended use of the software and demonstrate that the use of the software fulfils those intentions without causing any unacceptable risks.

DAS-COMBAT allows users to adjust for batch effects in datasets where the batch covariate is known, using methodology described in Johnson et al. 2007. It uses either parametric or non-parametric empirical Bayes frameworks for adjusting data for batch effects. Users are returned an expression matrix that has been corrected for batch effects. The input data are assumed to be cleaned and normalized before batch effect removal.

The module DAS-COMBAT is an R package and is used in the analysis of routine experiments performed in the DAS lab according to to SOP7600 – Diagnostic Assay Service. If applicable, it may also be used as an analysis step for related studies performed according to SOP7620 – Performing trials and studies in the DAS lab.

The R-package DAS-COMBAT is used as a resource in the module DAS-NORMALIZE. DAS-NORMALIZE is an additional module specific for the METHOD-CC as described in FRM6313 DAS Legacy Software Plan (LSP). I DAS-NORMALIZE will be interfaced to the DAS platform and fully documented elsewhere.

The R-package DAS-COMBAT will be used for the purpose of calculating a batch correction between known and indicated batches using the ComBat methodology.

* More specifically, the DAS-COMBAT R-package performs in the R environment and implements the following:Receives kinase activity measurement profiles that are suitably cleaned and normalized and a batch indicator variable .
* Calculates a ComBat correction model for batch effects in the measurement profiles based on the batch indicator variable.
* Returns the ComBat correction model.
* Applies the ComBat correction model to data and returns the corrected data. This can be the same data that was used to calculate the model, new data from the same batches as the data that was used to calculate the model (i.e REF and DAS samples), or a combination thereof. DAS-COMBAT implements the option to use an indicated *reference batch* in the correction. The reference batch will be unchanged and the other batches cahnegd towards the reference batch.
* DAS-COMBAT implements the option to calculate a Location-only (mean-only) model or a Location-Scale model.
* In the original ComBat methodology the option exist to use additional covariates besides the batch indicator variable. This option will not be implemented.

# Scope

Scope of this procedure is the software development plan for the R-package DAS-COMBAT.

## Out of scope

The DAS platform legacy software (which is covered in FRM-6313) is out of scope.

DAS-COMBAT will be implemented in the DAS routine process by the module DAS-NORMALIZE. DAS-NORMALIZE will be defined separately and is out of the scope of this plan.

## Software Safety Classification

[Define the overall classification of the software (Class A, B or C). Also, document the justification for this classification]

The module is a critical module for preprocessing DAS results prior to calculating a diagnostic result.

Incorrect functioning of the DAS-COMBAT module can result in an incorrect diagnostic result. Therefore, the initial risk classification of the software is class C.

Subsequently, risks will be further assessed using the company-wide Risk Management Process According to SOP4500. This includes a FMEA on DAS-COMBAT.

# Definitions

SOUP Software of unknown provenance

SP Software Product, i.e. DAS-COMBAT

DAS PLATFORM The software platform to which modules plug-in

METHOD The name of the medical device

FMEA Failure Mode and Effect Analysis

DAS Diagnostic Assay Services

REF Reference sample, used to indicate reference sample or Internal Assay Control sample.

# Roles and responsibilities

[Define Roles and Responsibilities for the Software Development Plan and the overall project]

PM (Project Manager Faris Naji

PO (Product Owner) Faris Naji

ST (Software Team) Faris Naji, Alexandre Maurel, Rik de Wijn

U (User) Tineke van Doorn

The responsibilities are outlined in the SOP 6300 Software Planning Process.

# Reference Documents

## System references

### *System Requirements*

[Include a reference to the system requirements of the medical device. These will be used as input for the software requirements.]

The DAS-COMBAT is an R-package (or library) that operates in the R environment. It supplies functions that can be called from the R environment.

The measurement or phosphorylation profiles are represented as a matrix of measurements where each row represents an observation (e.g. array, Sample name, Patient ID) and each row represents a particular peptide in the phosphorylation measurement. The batches are represented by a batch indicator variable that contains a batch identifier for each observation in the measurement data.

A correction model according to the ComBat methodology decribed in literature by Johnson et al. [1] and Zhang et al. [2] and implemented in its original form in teh R-package SVA (Leek JT, Johnson WE, Parker HS, Fertig EJ, Jaffe AE, Zhang Y, Storey JD, Torres LC, 2020 sva). The ComBat method is described in literature by Johnson et al. [1] and Zhang et al. [2] and implemented in its original form in the R-package SVA Leek JT, Johnson WE, Parker HS, Fertig EJ, Jaffe AE, Zhang Y, Storey JD, Torres LC (2020). sva: Surrogate Variable Analysis. R package version 3.36.0 ). However, the SVA package cannot be used “as is” because a modification is necessary to allow a correction calculated on a set of REF samples to be applied to a set of DAS samples, as described in [3]. Hence, for use in DAS\_NORMALIZE the Combat code has to be programmed under life cycle control. An R-Package (“dascombat”) can be programmed for this purpose, using the information in Appendix A (methodology) and appendix B of [3] as supporting documentation.

The full specification process of the DAS-COMBAT module is outlined by the SOP 6300 software planning process. However, The DAS-COMBAT module is a low complexity module performing a single task. The software is intended to run on a dedicated generic Windows PC. Therefore, the module DAS-COMBAT and the supporting package DAS-COMBAT are fully specified in the requirements document:

* dasnormalize-6304-Software Requirements.docx
* dascombat\_6305-Software Requirements.docx

Hence, a dedicated architecture document and detailed design document are deemed unnecessary.

### System Development procedures

[Include references to all procedures used in development of the medical device to coordinate the software development and system development]

* SOP6300 software planning
* FRM6300 software development plan
* FRM6301 tractability matrix
* FRM6302 configuration list
* FRM6303 soup items
* SOP6301 software specs
  + FRM6304 requirements
  + FRM6305 architecture
  + FRM6306 unit design
* SOP6302 software coding
  + WI6302 coding guidelines
  + FRM6307 test report
* SOP6303 software testing
  + FRM6308 integration testing
  + FRM6309 system testing
* SOP6304 software release
  + WI6302 coding guidelines
* SOP6305 software deployment
  + WI6302 coding guidelines

See table below for version/approval dates of the document.

## DevelopmentStandards, Methods and Tools and Regulatory references

### Standards

[List here or include references to standards used in development of the software]

* IEC 62304:2006

This standard refers to the Medical device software -- Software life cycle processes.

### Methods

[List here or include references to methods used in development of the software]

Concepts and techniques of the SCRUM Agile software development method is used internally.

### Tools

[List here or include references to tools used in development of the software. Development tools can also be included in the SOUP Configuration Items List]

The development tools are outlined in the soup configuration list (see sections below):

181102-6303 Soup Configuration.xls

[Refer to the SOUP items list with the way how to identify the development tools in that list]

### Regulatory references

[List here or include references to relevant regulations used in development of the software]

* [Regulation (EU) 2017/745](https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32017R0745&locale=en) repealing Council Directives 90/385/EEC and 93/42/EEC (MDR) (May 2017)
* [Council Directive 93/42/EEC](https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:01993L0042-20071011&locale=en) on Medical Devices (MDD) (1993)

# Processes used in the development

The following process will be used during the development of the medical device software:

|  |  |  |
| --- | --- | --- |
| **Document ID** | **Title** | **Version** |
| SOP 6300 | Software Development Planning | 1.0 |
| SOP 6301 | Software Specifications | 1.0 |
| SOP 6302 | Programming and Coding Guidelines | 1.0 |
| SOP 6303 | Software Verification and Testing | 1.0 |
| SOP 6304 | Software Release | 1.0 |
| SOP 6305 | Software Deployment | 1.0 |
| SOP 4500 | Risk management procedure | 1.0 |
| SOP 4300 | Change control procedure | 1.0 |
| SOP4600 | SOP 4600 Validation of Computer Software used for production and QMS | 1.0 |

# Software Deliverables

[Provide a list of the outcome of this development process. These should be all the software, it’s components and other items that are created to operate directly or indirectly with the medical device. This can also be provided in the Software Configuration List. In that case this section can be removed]

* DAS-COMBAT (R-package)

The R-package is the software product. A software git repository idelivered and located at:

https://github.com/pamgene/dascombat

The software repository contains all the code, tests and documentation, unless otherwise specified.

# Traceability

The traceability matrix provides the overview of relationships between the system requirements, software requirements, test results and risk control measures within the software. This matrix shows all the specified requirements have been addressed, tested. It includes the list of implemented risk control measures.

The documents outlining traceability are

TBD

# Software Configuration List

To keep an overview of the software configuration of all DAS modules a separate common repo called DASCOMMON, this will also highlight any potential conflicts between SOUP or DAS modules.

The document outlining configuration is:

181001-6303 Configuration.xlsx

Located at:

<https://github.com/pamgene/dascommon/tree/master/configuration>

# SOUP Configuration Items

To keep an overview of the software configuration of all DAS modules a separate common repo called DASCOMMON, this will also highlight any potential conflicts between SOUP or DAS modules.

The document outlining SOUP is:

181001-6301 SOUP.xlsx

Located at:

<https://github.com/pamgene/dascommon/tree/master/configuration>

# Software Integration and Testing Plan

The software integration testing will be performed as outlined in:

SOP6303 Software Verification and Testing

Software integration testing will be documented in:

TBD

Software system testing will be performesd as outlied in

SOP6303 Software Verification and Testing

Software system testing will be documented in:

TBD

# Software risk management

The software risk management conducts the activities and tasks of the software risk management process, including the management risks related to SOUP*.*

Risk management is performed according to SOP4500 as part of the risk management activities for METHOD DAS as specified in:

* 210200KS18001 Risk Management Plan METHOD DAS
* 210227RW18038 Risk Management DAS software.

The specific implementation of the R-package DASCOMBAT will be done by implementing it in the DAS module DASNORMALIZE. Therefore, the risks associated with using it in the DAS process (e.g. risks associated with correctness of input data, correctness of settings etc.) are best handled in the risk analysis to be performed for DASNORMALIZE.

For DASCOMBAT a FMEA was performed focussing on correct implementation of the COMBAT methodology:

* DASCOMBAT-FRM4503-FMEA sheet.xlsx

14 risks were identified which are addressed by software testing.

# Project deliverables

| **Document ID** | **Document name** |
| --- | --- |
| 210227RW20010 | Software Development plan |
| 181001-6302 | Software Configuration |
| 181001-6303 | SOUP Configuration |
|  | Software requirements |
|  | Traceability Matrix |
|  | FMEA sheet |
|  | Software integration testing |
|  | Software system testing |

# Common Software Defects

This section lists the (categories of) common software defects that have been identified in relation to the selected programming technology. For each of the items in the list it is described how these common defects can be avoided (or if not otherwise possible will be mitigated to the best possible extend).

|  |  |
| --- | --- |
| **Common Software Defect** | **Defect resolution or mitigation** |
| Interruption due to DAS session | Restart |
| Interruption due to Windows operating system | Restart |
| Interruption due to hardware | Restart |

# Appendix

NA

# References and related documents

[1] W. E. Johnson, C. Li, and A. Rabinovic, “Adjusting batch effects in microarray expression data using empirical Bayes methods,” *Biostatistics*, vol. 8, no. 1, pp. 118–127, Jan. 2007.

[2] Y. Zhang, D. F. Jenkins, S. Manimaran, and W. E. Johnson, “Alternative empirical Bayes models for adjusting for batch effects in genomic studies,” *BMC Bioinformatics*, vol. 19, no. 1, p. 262, Jul. 2018.

[3] PamGene International, “210228RW19030 Development and verification of the use of reference samples in teh DAS lab,” 2019.

[4] PamGene International, “210228RW19031 METHOD clinical calibration experimental report,” 2019.

# Signatures and approval

The signatures in the table below ensure that all items in this Software Development Planning are accepted by PamGene, Den Bosch, Netherlands. PamGene is responsible for the review and approval of this Software Development Planning.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Role** | **Function title** | **Name** | **Signature** | **Release date** |
| Author | Software Manager | Faris Naji |  |  |
| Reviewer | Program Manager IVD | Kristiane Schmidt |  |  |
| Authoriser | Head of DAS | Rik de Wijn |  |  |
| Quality Assurance | QARA manager | Theo van der Leij |  |  |

# History

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Remarks** | **Document owner** |
| 1.0 | 01-Jan-2020 | Initial version | Faris Naji |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |