Aarhus University

Systems Engineering

Company H

SEP-01-00-ALL-V01-20210224

Systems Engineering Management Plan

BEUMER Group

Group members:
Martin Jespersen (201706221)
Tristan Møller (201706862)
Rikke Christensen (201704464)
Jesper Jakobsen (201708777)
Mikkel Jensen (201708684)
Jens Bendtsen (201708413)
Mads Dahl (201705285)
Marie Bærentzen (201608667)

Superviser:
Stefan Hallerstede, Lektor

May 3, 2021

1 Version History

Ver.	**Date**	**Initials**	**Description**
1.0	18-02-2021	ALL	First draft of document.
1.1	19-02-2021	JJ, MD, MB	Time Plan Section
1.2	20-02-2021	TM, MJ	Updated Configuration Management
1.3	21-02-2021	RC	Added Identification & Status Accounting in CM
1.4	22-02-2021	MKJ, NG, JB	Risk Assesment Section

Contents

2 Introduction

This document contains the systems engineering management plan (SEMP). The SEMP is an important outcome of planning that identifies activites, key events, work packages and resources [1]. The first draft of the SEMP contains: **3 Risk assessment**, **4 Configuration Management Strategy** and **5 Time Plan**.

3 Risk assessment

This section serves to identify possible risks associated with the development of the "Baggage Handling Upgrade". Each risk should include a title, description, risk categories, potential root causes and responsible individuals/teams. Each risk will also get a likelihood and consequence factor assigned that will give the risk factor. The likelihood and consequence factors will be assigned between 1 and 5, where 1 is least likely/damaging value and 5 is the most likely/damaging value.

3.1 Risk analysis

In order to determine a meaningful risk probability for a technical risk, one could perform simulations or use risk scales with corresponding risk matrices. Simulations may also be used when assessing risk probability for cost risks, however for cost risks it might also be useful to consider decision trees and payoff matrices. For the schedule risks, simulations would be the main technique. Historical data from the company could be used to estimate appropriate probability values, possibly in conjunction with aforementioned techniques, such as decision trees.

Each risk should also be assigned a value representing its overall consequence in regards to the impact on the project development. The consequences should be determined from the techniques mentioned above. After assessing both the probability and the consequence of each risk, one can determine which risks need to be prioritized, based upon these values. Additional parameters can be used to more accurately prioritize the risks, such as frequency of occurrence and the concerned risks impact on other risks.

Both the risk probability and consequence is determined by intuition at the time of writing.

3.2 Risk handling

In order to handle risks for the system-of-interest in a productive manner a risk handling process is used that identifies different options, which can be used to reduce a risk to an acceptable level. Risk handling can be done on two levels. The first level is about prioritising the risks at the overall system level. The second level is about handling the individual risks and preparing for handling them individually. The second level is made by using Risk Element Plans (REP). These plans give an instruction on who is responsible and each handling option should include it's assumption, avoidance, mitigation and transfer.

The following tables describes the risks in the project. In the table Like is for Likelihood, Cons is for Consequence.

3.3 Table of Technical Risks

This subsection contains a table that gives an overview of the technical risks and their overall risk-value, which is calculated from the likelihood and consequence of each risk.

Risk Description	**Responsibility**	**Like**	**Cons**	**Overall Risk**
1. Conveyor Belt Precision and Stability	Technical Developing Team	1	5	5
2. Success Rate on X-Ray and Additional Screenings	SecureScreen RX 5001 Supplier	3	4	12
3. Insufficient Space to Fulfill Requirements	Technical Planning Team	2	5	10
4. Slow Sensor Hardware	Technical Planning Team	2	3	6

3.4 Table of Schedule Risks

This subsection contains a table that gives an overview of the schedule risks and their overall risk-value, which is calculated from the likelihood and consequence of each risk.

Risk Description	**Responsibility**	**Like**	**Cons**	**Overall Risk**
5. Delay of outsourced items	Project or Product Planning Team	4	3	12
6. Unforeseen delay in tasks	Project or Product Planning Team	3	2	6
7. Inaccurate time plan	Project or Product Planning Team	4	3	12
8. Late Deployment of System	Project or Product Planning Team	2	5	10
9. Staffing not available	Project or Product Planning Team	2	5	10

3.5 Table of Cost Risks

This subsection contains a table that gives an overview of the cost risks and their overall risk-value, which is calculated from the likelihood and consequence of each risk.

Risk Description	**Responsibility**	**Like**	**Cons**	**Overall Risk**
10. Outsourced Items Exceeding Initial Cost	Project or Product Planning Team	2	4	8

3.6 Table of Root Causes

This subsection contains a table that gives an overview of the Root Causes for each of the different risks in the project.

Risk Description	**Root Causes**
1. Conveyor Belt Precision and Stability	Insufficient hardware components or software capabilities.
2. Success Rate on X-Ray and Additional Screenings	Overload of baggage at manual inspection.
3. Insufficient Space to Fulfill Requirements	The facility room is smaller than required.
4. Slow Sensor Hardware	Limited or incomplete hardware component analysis.
5. Delay of outsourced items	Insufficient time plan or communication with outsourcing company.
6. Unforeseen delay in tasks	Insufficient man power or plan not taking risk into account.
7. Inaccurate time plan	Project planners missing technical knowledge.
8. Late Deployment of System	Insufficient planning, missing manpower or other schedule issues.
9. Staffing not available	Planning not being representative of project.
10. Outsourced Items Exceeding Initial Cost	Unexpected hardware costs or unexpected outsource hardware needs. Broken hardware.

3.7 Descriptions of Risks

This subsection contains a table that gives extra description to some of the risks.

Risk Description	**Description**
6. Unforeseen delay in tasks	Tasks taking longer than expected and not able to be completed in time according to plan.
7. Inaccurate time plan	Time plan not accurate for project or estimates not possible to be achieved.
8. Late Deployment of System	System is finished too late and therefore not being able to deployed and tested before deadline.
9. Staffing not available	Technical staff not available at the needed time.

4 Configuration Management Strategy

The purpose of the configuration management process is to control and monitor the system elements and configuration during the life cycle. This section describes the projects configuration management strategy that is used to identify how the project's products will be controlled and protected [2]. The strategy is strongly inspired by the configuration management procedure outlined by PRINCE2 Agile [3].

4.1 Planning

Configuration management is a collection of all the activities that maintain and control changes for each product throughout the life cycle of the project and after the project is completed [3].

CrisBag(r) is the product delivered to the customer, which consists of several critical functional components. Each of these components are listed below:

```
Main components:
    Tote
    Transport
    Curve
    90 Transfer
    Divert
    Merge
    Top-load
    Side-load
    Discharge
    Vertical Sort
    Walk-through
```

`Stacker`

The list of components above will be referred to as 'main components' and must all be versioned and labelled after a specific identification system, which is described in the next section. Sub-systems/sub-modules of each main component will not be forced to follow same versioning/labelling. Updating a baseline for a main component can only be done by following the change procedure described in section **4.3 Change Management**. If there is a change in a sub-module in a main component it will only be 'persisted / approved' if it follows previous mentioned procedure.

Most of the main components are delivered by Beumer or 3rd parties and will most likely not be up for revision during the systems life cycle. The products / solutions that are prone to change during the system's life cycle is the mechanical solution (and associated software solution). These solutions must also be versioned and labelled and will be explained in section **4.4 Status Accounting**. All major components/systems can be seen in figure 4.1

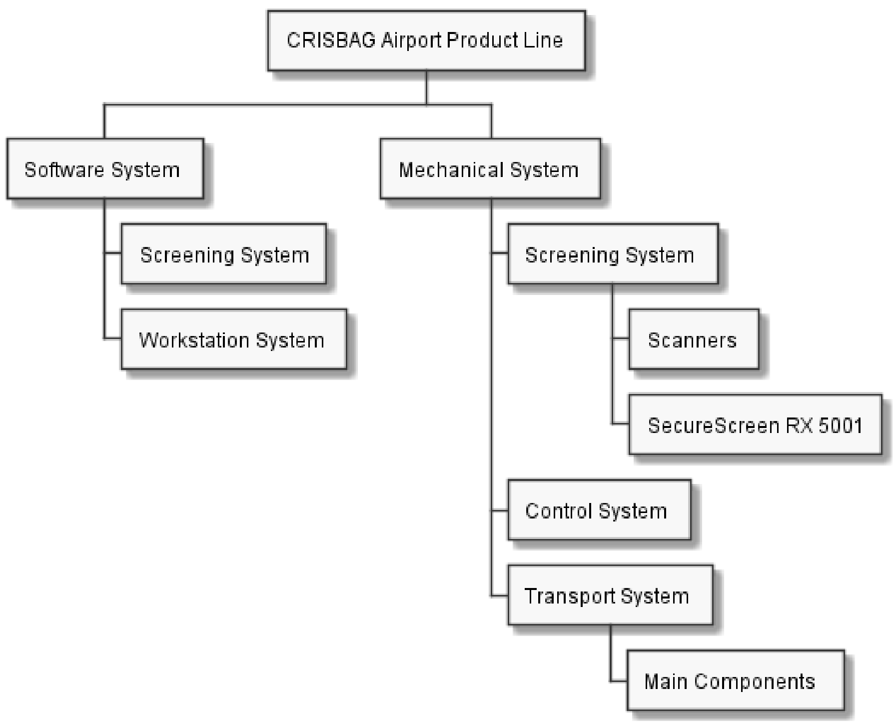

Figure 4.1: Components and subsystems that must follow the configuration management procedure. Components are also referenced as configuration items.

4.2 Identification

This activity gives the proper identification of configuration items (CIs) to be maintained under configuration control. The activity covers the identification of such items as well as defining unique identifiers for the CIs. This also includes establishing baselines for the CIs as well as agreements of the baselines by the acquirer and supplier.

Identification of CIs
The CIs of this project are mechanical modules (including hardware items), software modules, communication interfaces and project documentation. The main components of this system mentioned in **4.1 Planning** are the CIs to be tracked. The main components can be divided into additional sub products for which the identification can be specified if needed.

Unique identifiers
The purpose of the unique identifiers for each CI is to maintain version control to support and track the baselines of each product.

Identification of a product is done by naming the product by the following unique identifier:

```
<Product Code>
<Sub Product Code>
<Initials Of Owner>
<Version Number>
<Latest Modified Date>
```

The documentation of this product would be named:

```
01-00-SE-v01-20211802.pdf
```

The version number is significant for project deliverables and is assigned and changed when needed (e.g. for different iterations). The version number should not be changed for every product modification however the <Latest Modified Date> must always be updated.

The <Product Code> must indicate the relationship between the products (e.g. naming the adjacent products with adjacent or nearby numbers).

Establishing baselines
The baselines of the products are to be held as flexible as possible for as long as possible, but are to be approved by the acquirer and the supplier according to both the requirements and the associated designs.

A design baseline is described in terms of physical constraints and additional requirements (R1-R12, see case description [4]) including security and outsourcing of several system blocks. The design of the outsourcing is incomplete and is to be defined but are limited to 5 cabinets to hold electronics only.

4.3 Change Management

During the project it is unavoidable that multiple changes might be needed along the way. For this reason it is important that changes happen in a controlled and systematic way to avoid the change to effect cost, schedule etc. In this section a process will be described for changes to the individual components that are described in section 4.2. The change process begins with a ECP (Engineering Change Proposal) that is the baseline for a change. It is important for this ECP to have been classified for the impact it might have. The change should also be approved by a CCB (Configutration Control Board) that includes different stakeholders, which also includes the customer. When the ECP is approved by the CCB the change can be implemented. When the change has been implemented a change request must be submitted to merge it into the previous baseline. The steps of the process is illustrated on figure 4.2.

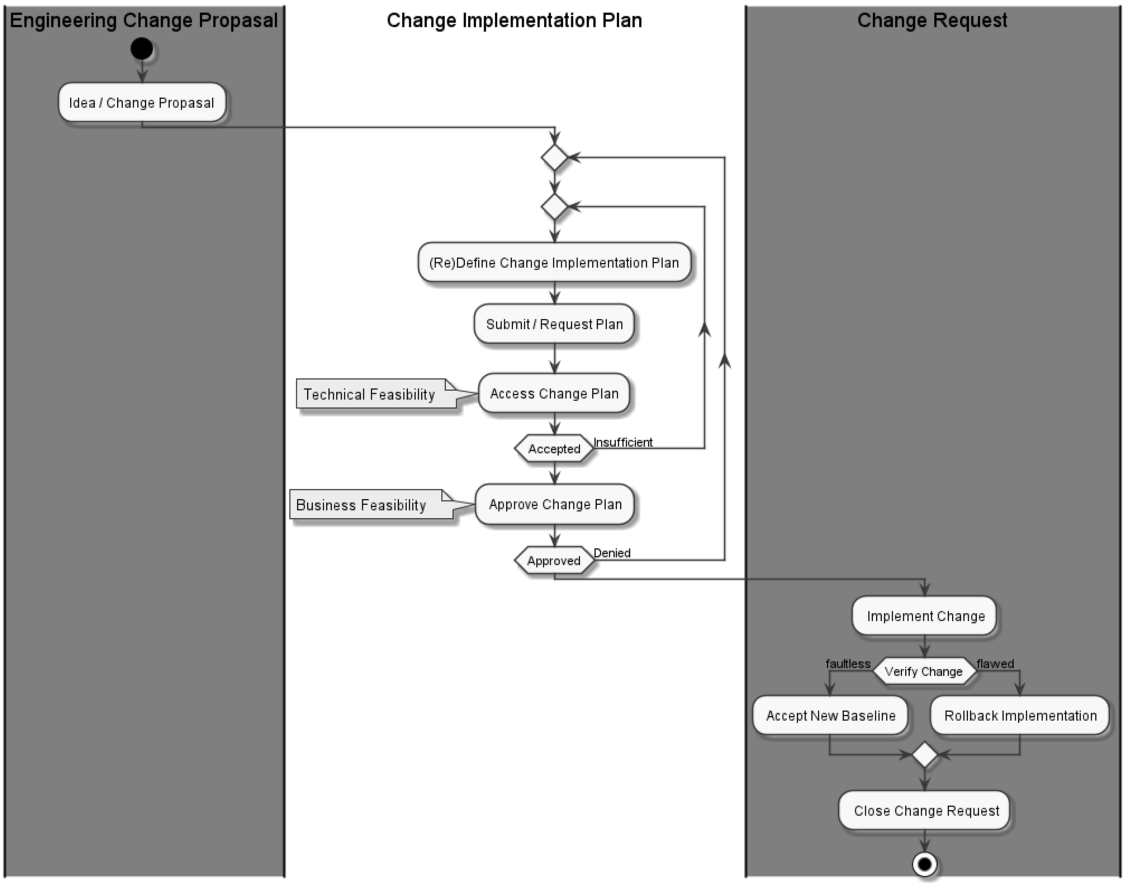

Figure 4.2: Change Management Plan

The first step after implementation is to create a 'Change Request' that is the new baseline for a product. Figure 4.3 illustrates what a change request must contain.

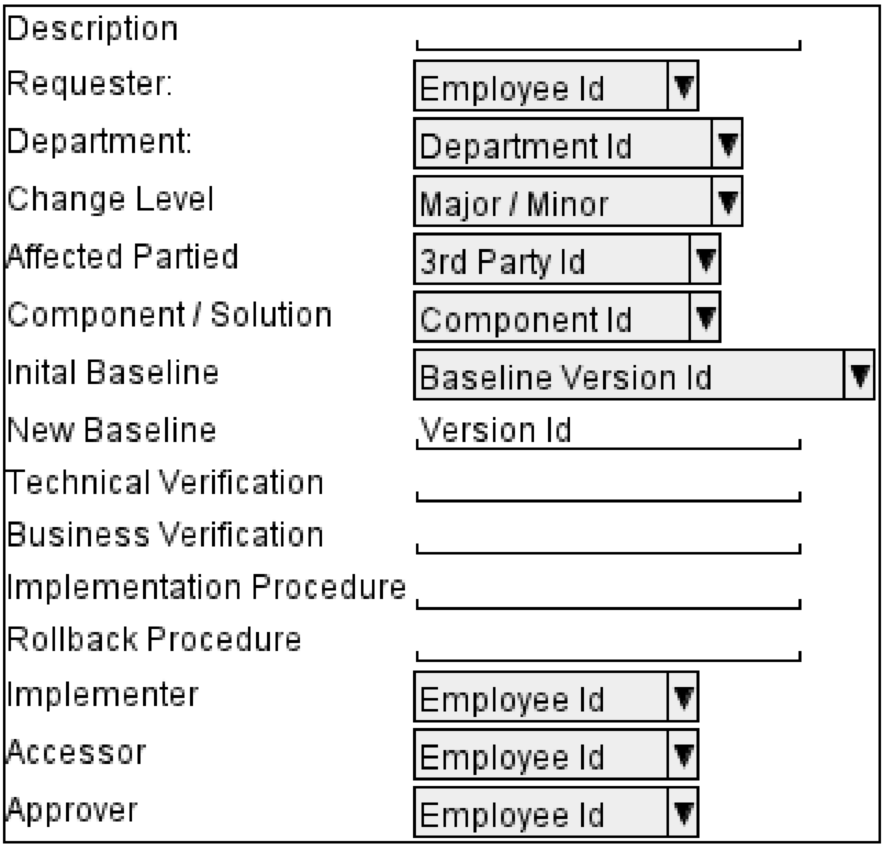

Figure 4.3: Graphical representation of change request. Non-dropdown elements should contain a detailed description of the specific task, e.g. 'Technical Verification' should describe how the change is verified after implementation.

The next step of the process is to send the 'Change Request' to a verification (if it is interactive component, e.g. a part of the mechanical solution, it should be verified in its production environment). This could be done by one or multiple people from the same team or department and do not have to include the management or stakeholders. The next step is to have the change verified by manager or stakeholders (e.g. customer). Only after these steps the changes gets a new version number and replaces an older version. The new version should be verified by managers and stakeholders to make sure that the change is correct. It is possible to roll back the change after this. In this case a new 'Change Request' should be created and the steps should be followed from the beginning with a new version being the result of the change.

4.4 Status Accounting

This activity is the process of reporting all activity related to the CIs of this system in all life cycle stages. The report shows the current and historical data for each product including all changes and the cause of change when renewing the version of a product.

Status accounting for this project includes the following status information for each product in the system:

```
Identifier
Version
Latest modification
Cause of Change
Current status
Initials of Owner
Date of next baseline
```

The products included in the status accounting system is the components identified in **4.2 Identification**. This covers all changes of versions which are to be recorded in the status accounting system.

The reports should be documents available to any stakeholder during all life cycle stages and should be in a format such as .docx or .pdf depending on who is requesting the report.

All additions must be approved by the configuration manager, to be sure that they have reviewed the correct protocol, cf. change management.

4.5 Verification and Audit

Verification and audit is the act of ensuring that the configuration management procedure outlined in the change management section is followed. There is dedicated two steps in the change management to facilitate this: 'Access' and 'Approve'. The 'assessor' is usually a group of engineers from the same team or department that ensures that the change implementation plan is technical feasible and 'approver' should be a group of people with extensive business domain knowledge and experienced in change management verification and auditing (CCB).

Verification and audit management also includes the alignment of the project with stakeholders, consumers and customers. Below are some examples of elements that must be considered:

- Do the customer / stakeholder have access to the correct product versions?
- Change request aligned with end goal, e.g. is new baseline aligned with business goals?

5 Time Plan

The time plan, seen below, is a first draft and should be taken as such.

2020:	2020:	2020:	2020:	2020:
January	Feburary		March	April
Conceptual design	Preliminary design	**Decision gate**: Review	Detail design and development	
Problem definition	Functional analysis		Subsystem design	Verification of processes
Need Identification	Trade-off studies		Evaluation of alternatives	Development of prototypes
Requirement Analysis	Early prototyping		Production planning	Test and evaluation
System/program planning	Contracting			
	Program implementation			
2020:	2020:	2020:	2020:	2020:
May	June		July	August (First half)
Detail design and development		**Decision gate**: Review	Production/construction	
Order equipment	Production/constructon			System assesment
for construction	Construction of physical			
	system begins.			
	Support from SW Dev			
	during installation.			
2020:	2020:	2020:	2020:	2020:
	August (Second half)	September (first half)		September (second half)
Milestone: Construction done			**Decision gate**: System should	Operational use and system support
	SW testing on client site	Test and evaluation	be complete. (Should production	System operation in user environment
	Acceptance testing		and construction be extended?)	Operational testing
2020:	2020:	2020:	2021:	
October	November	December	January	
Operational use and system support				
Field data collection	System modification			
and analysis.	for improvement.			

Figure 5.1: First draft of a time plan.

The time dedicated to each phase has been calculated using the information in the case description [4]. For example the production and construction time was calculated using the time to install, number of units and the preprocessing time. This resulted in approximately 2.5 months of work, when working 8 hours a day for 20 days a month. The other phases were calculated in a similar fashion but this will not be described. The duration of the last phase, the operation and system support phase, has been shorten due to the other phases requiring more time, it is possible to shorten this phase even further in the decision gate before the phase starts. This is done so the developers can extend their testing phase by a month if needed.

Each phase, along with decision gates and milestones have been colour coded to gain a better overview.

References

[1] INCOSE, "Systems Engineering Handbook," no. 4, 2015.

[2] Axelos, "CONFIGURATION MANAGEMENT STRATEGY," 2016. [Online]. Available: https://publications.axelos.com/PRINCE2Agile2016/content.aspx?page=pra_185&showNav=true&expandNav=true.

[3] D. Litten, "The Configuration Management Procedure," [Online]. Available: https://www.prince2primer.com/the-configuration-management-procedure/.

[4] Beumer, "BAGGAGE HANDLING UPGRADE," vol. 2017, pp. 1–11, 2018.