




TITAN Turnaround Integration in Trajectory And Network Project Number: 233690		
Turnaround Tool Verification Document		
CLASSIFICATION: PU	ISSUE: 1.0	DATE: 16/11/2012

DOCUMENT REFERENCE				
Project	Work Package	Partner	Nature	Number
TITAN	WP 4	RWT	DEL	06

	Turnaround Tool Verification Document	Issue: 1.0 Date: 16/11/2012
--	--	------------------------------------

DOCUMENT CONTROL			
Responsible	Company	Name	Date
Author	RWT	Athanasios Katsaros	16/07/2012
Partners involved	JEP	John Butcher, Alicia Grech	
	SLO	Balázs Kerülő, Noémi Král	
Reviewer	JEP	John Butcher, Alicia Grech	08/10/2012
	INE	Luis Chocano, Ana Sáez	
Approval	INE	Luis Chocano	12/11/2012

DOCUMENT CHANGE LOG			
Issue	Date	Author	Affected Sections / Comments
0.1	16/07/2012	Athanasios Katsaros	Creation
0.2	03/08/2012	Athanasios Katsaros, Noémi Král	Integration of test results for iteration 04
0.3	28/09/2012	Athanasios Katsaros	Integration of test results for iterations 05-06.1 Integration of comments from 2 nd WP4 PM Integration of test results for final iteration 07 (final verification)
0.4	05/10/2012	Athanasios Katsaros	Integration of comments from WP4 Consolidation Meeting WP4 internal review
0.5	12/10/2012	Athanasios Katsaros	Integration of comments from WP4 internal review
0.6	12/11/2012	Athanasios Katsaros	Approved issue
1.0	16/11/2012	Ana Sáez	Final version after EC approval



Turnaround Tool Verification Document

Issue: 1.0

Date: 16/11/2012

DOCUMENT DISTRIBUTION

To/Cc	Organisation	Name
To	EC	Remy Denos
To	INECO	Luis Chocano
To	INECO	Ana Sáez
To	Jeppesen	Alicia Grech
To	Jeppesen	John Butcher
To	CRIDA	Nicolás Suarez
To	CRIDA	Eva Puntero
To	ISDEFE	Vicente Bordón
To	ISDEFE	Sara Peces
To	ISDEFE	Rosana Casar
To	Blusky	Steve Zerkowitz
To	Blusky	Daniel Zerkowitz
To	SLOT	Roland Gurály
To	SLOT	Noemi Kral
To	SLOT	Balázs Kerülő
To	RWTH Aachen	Sebastian Kellner
To	RWTH Aachen	Athanasios Katsaros



Turnaround Tool Verification Document

Issue: 1.0

Date: 16/11/2012

TABLE OF CONTENTS

1. INTRODUCTION	7
1.1 DOCUMENT SCOPE	7
1.2 DOCUMENT STRUCTURE.....	7
1.3 INTENDED AUDIENCE	7
1.4 ASSOCIATED DOCUMENTATION	7
1.5 ABBREVIATIONS AND ACRONYMS	8
2. VERIFICATION ACTIVITIES	10
2.1 VERIFICATION METHODOLOGY.....	10
2.2 VERIFICATION SCENARIO	10
2.3 NON-COMMERCIAL DEMONSTRATOR REQUIREMENTS TO BE VERIFIED.....	10
2.4 DEFECT/ERROR SEVERITY AND ACTION/REPAIR PRIORITY	11
3. VERIFICATION RESULTS	13
4. CONCLUSIONS	23
APPENDIX - VERIFICATION RESULTS PRIOR TO FINAL RELEASE	24



Turnaround Tool Verification Document

Issue: 1.0

Date: 16/11/2012

LIST OF TABLES

Table 1. Description of defect/error severity levels	11
Table 2. Description of action/repair priority levels	12
Table 3. TITAN non-commercial demonstrator non-functional requirements verification matrix - Final.....	15
Table 4. TITAN non-commercial demonstrator general functional requirements verification matrix - Final.....	18
Table 5. TITAN non-commercial demonstrator specific UI functional requirements verification matrix - Final.....	22
Table 7. TITAN non-commercial demonstrator non-functional requirements verification matrix - Prior to final	26
Table 8. TITAN non-commercial demonstrator general functional requirements verification matrix - Prior to final	29
Table 9. TITAN non-commercial demonstrator specific UI functional requirements verification matrix - Prior to final.....	34




Turnaround Tool Verification Document

Issue: 1.0

Date: 16/11/2012

EXECUTIVE SUMMARY

The current document, referred to as “Turnaround Tool Verification Document”, builds together with the “Verification Plan” the verification activities that aim at the high-level TITAN objective of developing a decision-support tool for airlines to achieve a more efficient turnaround process by implementing the TITAN concept. The delivered non-commercial demonstrator serves as a subset of the final commercial TITAN tool, where enough of the requirements to effectively execute the selected/restricted operational scenario are implemented. The correct behaviour of its modules/components in an A-CDM environment was verified through testing against pre-defined requirements set in the verification plan. Into this document the results of the conducted verification activities are consolidated.

	Turnaround Tool Verification Document	Issue: 1.0 Date: 16/11/2012
--	--	------------------------------------

1. INTRODUCTION

1.1 Document Scope

According to [6] the objective of WP4 is to develop a decision-support tool for airlines to better evaluate and negotiate any changes in their schedule due to modifications affecting the turnaround process of one or several of their aircraft (to achieve a more efficient turnaround process).

To ensure, that the TITAN component is implemented properly, the delivered non-commercial demonstrator is verified on the basis of pre-defined requirements [5]. The tasks to be fulfilled in the context of the verification activities are the following:

- creation of a detailed verification plan based on the specification scope of a future production strength commercial tool, and specialised to verify features of the non-commercial demonstrator delivered within the TITAN project;
- verification of the correct behaviour of the demonstrator in an A-CDM environment;
- elaboration of the verification report, aim of the current deliverable.

1.2 Document Structure

The document consists of the following chapters:

- Chapter 1 is the current introductory section.
- Chapter 2 gives a brief description of the verification activities concerning verification method, operational environment, requirements to be verified and result classification.
- Chapter 3 summarizes the results of final verification in matrices.
- Chapter 4 consolidates the conclusions drawn after completion of the verification process.
- In Appendix the preliminary verification results for all iterations preceding final release are summarized.

1.3 Intended Audience

This public document (PU) may be distributed freely both within and outside the TITAN consortium. It is primarily a technical document intended to present to all TITAN partners as well as those outside TITAN who are engaged in verification processes, especially at the airport side, the results of testing the TITAN non-commercial demonstrator against specified requirements.

1.4 Associated Documentation

- [1] Butcher, J., 2011. *TITAN WP4 Scoping Document*. 3rd Version. Melbourne: JEPPESEN.
- [2] TITAN_WP4_JEP_DEL_02_v0.4_Turnaround Tool Design.doc
- [3] TITAN_WP4_SLO_DEL_01_v0.7_Turnaround Tool Specification.doc
- [4] IEEE Standards Board, 1990. *IEEE Standard 610.12-1990 - Glossary of Software Engineering Terminology*. New York: IEEE Standards Board.
- [5] TITAN_WP4_RWT_DEL_05_v0.7_Verification Plan.doc
- [6] TITAN Collaborative Project, 2011. *TITAN Annex I - "Description of Work"*. 5th Version. Madrid: INECO.



1.5 Abbreviations and Acronyms

A	Availability
A/C	Aircraft
A-CDM	Airport Collaborative Decision Making
AD	Administration/Administrator
AIRS	Airport Information Report Service
ALDT	Actual Landing Time
ASRF	Aircraft Status Reporting and Forecasting
ASRS	Aircraft Status Reporting Service
BFIS	Baggage Flow Information Service
C	Common
D	Detail Screen View
DEL	Deliverable
EH	Event Handling
EIBT	Estimated In-Block Time
ELDT	Estimated Landing Time
EOBT	Estimated Off-Block Time
ETOT	Estimated Take-Off Time
F	Functional
HMI	Human Machine Interface
IC	Information Classification
ID	Identity
NA	Network Architecture
NF	Non-Functional
N/V	Non-Verifiable
PBS	Passenger and Baggage Search
PD	Platform Dependency
PFIS	Passenger Flow Information Service
PM	Progress Meeting
PNL	Passenger Name List
PNR	Passenger Name Record
PU	Public Document
R	Reliability
S	System or Summary Screen View



Turnaround Tool Verification Document

Issue: 1.0

Date: 16/11/2012

SAU	System Access and Use
TIS	TITAN Information Sharing / System
TITAN	Turnaround Integration in Trajectory and Network
U	Usability
UI	User Interface
WP	Work Package



2. VERIFICATION ACTIVITIES

2.1 Verification Methodology

Testing¹ and demonstration were used for the verification of the TITAN non-commercial demonstrator. While software code reviews/inspections were conducted by the development team, the verification team tested executable non-commercial demonstrator iterative pre-releases preceding the final verification, whose results are consolidated in this document. Through these means, premature inefficiencies - if any - were identified already during the release iterations, reducing the possibility of delivering a final release with inefficiencies not to be easily mitigated at this last stage.

Requirements, which were difficult to scope in a demonstrator and therefore not (easily) testable such as reliability were verified by design giving full reference to the corresponding sections of [2], where meeting of such requirements is described.

2.2 Verification Scenario

As the TITAN non-commercial demonstrator does not aim to cover all possible TITAN concept use-cases, it focuses on a particular operational scenario referred to as the “missing passenger scenario”; the end of a normal aircraft turnaround process is disrupted by some passengers, who fail to board, and appropriate steps must be taken to locate them. However, finding the missing passenger(s) is beyond the scope of the demonstrator. Therefore, verification focuses on all processes taking place between a/c in-block and off-block times taking all through TITAN complemented A-CDM milestones in between into account. An overview of the operational scenario is given in full detail in section 3 of [5].

2.3 Non-commercial Demonstrator Requirements to be verified

Based on [5] a detailed verification plan was developed containing all functional and non-functional requirements that describe the expected behaviour of the final fully-functional commercial tool. These requirements are logically grouped in so called test (use) cases; each actor (user) should be able to use particular functionalities of TITAN services in a specified operational environment for providing/getting any information he needs. The outcome of the test (use) cases verifies that the tool is able to assist/support all involved actors in decision-making by making them aware of the situation tested/simulated and its impact and by enabling them to minimize or even eliminate it.

However, verification within the TITAN project was conducted against expectations for the non-commercial demonstrator that serves as a subset of the final TITAN tool. For that reason, some requirements set in [5] were either too demanding and so verification success criteria should be loosened or requirements should be verified by other means (such as design); or they were considered to be out of scope for the demonstrator. In both cases the scope of the development and the expectations of the subset of testable requirements were reconciled with the development team as consolidated in [1]. In the following, a full list of the requirements considered to be either out of scope for the demonstrator or not testable (and so verifiable by other means such as design) is provided²:

- full administration capabilities (focusing on service administration);

¹ Dynamic specification-based black-box system testing was used.

² Further details are given in Table 3, Table 4 and Table 5, where the final verification results are consolidated.



Turnaround Tool Verification Document

Issue: 1.0

Date: 16/11/2012

- user-client interaction (write access);
- reliability (i.e. notification on service availability);
- information classification;
- service-specific functionalities such as near-future a/c status or knock-on delays forecast and passenger/baggage “search & find” (data mining and intelligent retrieval of trend data in the TIS);
- service subscription;
- complete/detailed colour coding;
- provision of active key process-specific information (tabs of detail view).

Furthermore, testing of the use cases is beyond the scope of the non-commercial demonstrator. The reason for that lies in the decision for skipping the user-client interaction methodology as it is mostly a security-driven feature, and therefore not priority for the demonstrator. Not having any user-client interaction implemented in it makes use-case based verification infeasible, therefore postponing it for a future commercial tool. This, however, does not downgrade the quality level of the verification activities for the demonstrator, where the requirements set in [5] are tested in a non use-case-based verification environment.


2.4 Defect/Error Severity and Action/Repair Priority

The purpose of verifying the TITAN non-commercial demonstrator against the pre-defined requirements is to detect well in advance potential defects/errors and determine their criticality as well as the need to mitigate them so that a robust software that meets the objectives it is planned for in the defined operational environment is delivered.

Defect/error severity and action/repair priority scales - easily adaptable to any automated test management tool - were defined in alignment with the approved guidelines [4]; a five (5) level scale of (decreasing) defect/error severity and a five (5) level scale of (decreasing) action/repair priority. Both are summarized in Table 1 and Table 2. Any defect/error identified during the verification of the TITAN non-commercial demonstrator was classified according to its severity and repair priority in order for the development team to take any necessary action in a timely manner.

Severity level	Description
Critical	The defect/error results in the failure of the complete software system or subsystem/unit (program or module) within the system.
Major	The defect/error results in the failure of the complete software system or subsystem/unit (program or module) within the system. There is no way to use the failed component(s); however, there are acceptable processing alternatives which will yield the desired result.
Average	The defect/error does not result in a failure, but causes the system to produce incorrect, incomplete, or inconsistent results, or the defect/error impairs the systems usability.
Minor	The defect/error does not cause any failure, does not impair usability, and the desired processing results are easily obtained by working around the defect/error.
Exception	The defect/error is the result of non-conformance to a standard, is related to the aesthetics of the system, or is a request for an enhancement. Defects/errors at this level may be deferred or even ignored.

Table 1. Description of defect/error severity levels

	Turnaround Tool Verification Document	Issue: 1.0 Date: 16/11/2012
--	--	------------------------------------

Priority level	Description
Urgent	Further development and/or testing cannot occur until the defect/error has been repaired. The system cannot be used until the repair has been completed.
High	The defect/error must be resolved as soon as possible because it is impairing development/and or testing activities. System use will be severely affected until the defect is fixed.
Normal	The defect/error should be resolved in the normal course of development activities. It can wait until a new version is created.
Low	The defect/error is an irritant which should be repaired but which can be repaired after more serious defect/error has been fixed.
Defer	The defect/error repair can be put off indefinitely. It can be resolved in a future major system revision or not resolved at all.

Table 2. Description of action/repair priority levels



3. VERIFICATION RESULTS

In this chapter the results of the verification of the TITAN non-commercial demonstrator are summarized. For that purpose, the verification matrices developed in [5] are used here (Table 3 to Table 5) to provide information about fulfilment or non-fulfilment of the requirements considered to be in scope for the demonstrator. Therefore, they are supplemented with following information:

- *Information about completion level of the functionalities each requirement refers to*

As verification began when the first testable non-commercial demonstrator release was made available (iteration 03) - ending with the verification of the final release (iteration 07) - the different tool functionalities were implemented stepwise. This is depicted in the verification matrices, where three (3) completion levels are used (as indicated by the development team each time a new release was distributed for testing³):

- Completion level 1 indicates that the relevant functionalities the requirement refers to are “not implemented”;
- Completion level 2 means that the relevant functionalities are “partially implemented”; and
- Completion level 3 indicates that the relevant functionalities are “(fully) implemented”.

An appropriate colour coding is introduced (in Appendix) to distinguish each completion level from the rest; red for level 1, orange for level 2 and green for level 3. As development progressed completion levels did too, from 1 to 3 (red to green) except for requirements that remained beyond the scope of the demonstrator until the end of development.

- *Information about the outcome of testing against each requirement*

The outcome of testing each requirement may be positive or negative:

- Positive outcome means that the requirement is met;
- Negative outcome means that the requirement is not met.

In the second case, a full description of the identified inefficiencies that prevented the requirement from being met is given.

- *Information about defect severity and action necessity*

In this last case, the defect, error or the inconsistency identified in demonstrator’s behaviour is shortly described and classified with respect to its severity and the necessary repair action according to the specified classification scales.

Once the results of the verification process were obtained and potential inefficiencies were identified and classified, it was up to the development team to decide how to mitigate them.

Each testable non-commercial demonstrator release (beginning with iteration 03) was tested before verification of the final release (iteration 07), as a means of identifying and limiting potential inefficiencies well in advance and so minimizing the risk of delivering a highly inefficient final release at the very end. These preliminary results are consolidated in the Appendix of this report.

³ Information about “level of completion” was provided each time by the development team. Whether the corresponding requirement is finally met or not (fully, partially or not at all) was proven through the verification results.



Turnaround Tool Verification Document

Issue: 1.0

Date: 16/11/2012

Requirement ID-Number	Document / Section	Requirement Description	"Shall" Statement	Verification Success Criteria	Verification Method	Verification Level	Completion Level	Final Verification Result		
								Positive / Negative (Defect?)	Defect Severity	Action Priority
NF-NA-00	[3] / Section 2.1	Network architecture	The tool shall be built as semi-centralized according to system-oriented architecture principles.	1. TITAN information system (TIS) plays the central role. 2. Services are integrated into the system interacting successfully and being administrated with/by TIS.	Design / Development	System / Service	Partially Implemented	[2] / Sections 4.2 and 4.4, Figures 1, 3 and 6 Services are more logical than physical, so administration is fairly much a non issue. They are implemented as Java servlets and so can be managed by the servlet container (i.e. Tomcat)	-	-
NF-PD-00	[3] / Section 2.1	Platform dependency	Human-machine-interfaces (HMI) shall be specialized to users' needs.	HMI is: 1. adapted to each end user's needs; 2. formed in accordance with each end user's information needs.	Testing / Demonstration	End-user	Partially Implemented	Positive HMI customization is provided for the role of the gate allocator, where the client has a "proof of concept" display relating to this role's tasks (for future use). <i>See also [2] / Section 4.4.1.2 and Figure 4.</i>	-	-
NF-R-01	[3] / Section 2.1	Reliability	Authorized users shall exchange data successfully.	All UIs are capable of: 1. transferring required data; 2. verifying successful data receiving.	Testing	End-user	Partially Implemented	Positive regarding data exchange status A coloured box near the user name indicates the state of server communications. When it is green, communication with the server (therefore services) is stable and when it is red the server (and services) is non-responsive. Dots in the box indicate client side health. <i>* User-client interaction (user provides data in real time) is mostly a security-driven feature to be tested in a future production strength commercial tool, and therefore not priority for the demonstrator.</i> <i>See also [2] / Sections 6.3 and 6.3.4.</i>	-	-
NF-R-02	[3] / Section 2.1		End-user interfaces (UI) shall clearly indicate information exchange status.	UI can indicate receiving or pending status of user data sent to the system.	Testing	End-user	Partially Implemented	Positive Demonstrator's heartbeat is the clock that displays the server time; if the clock is ticking information exchange is working. <i>See also NF-R-01.</i> <i>See also [2] / Sections 6.3 and 6.3.4.</i>	-	-
NF-R-03	[3] / Section 2.1		Data concurrency shall be filtered on the database level.	All system users get exactly the same information as soon as it is made available.	Testing	System / End-user	Implemented	Positive a) All users get exactly the same information when made available. b) Some of the times corresponding to flight status changes are shown in the process-dedicated tabs of the detail view.	-	-
NF-R-04	[3] / Section 2.1		System shall get notified about service unavailability.	TIS, services and users get corresponding (warning) messages for service unavailability.	Design / Development	System / Service / End-user	Partially Implemented	[2] / Sections 4.4.1.2, 4.5 and 6.3.4 Such notifications are not of great importance for a user. Though not important for a demonstrator, it is important for a commercial tool.	-	-
NF-A-00	[3] / Section 2.1	Availability	System shall meet minimum operation requirements in case of service downtimes.	Upon the unavailability of a particular service overall system downtime is optimally reduced by implementing specific procedures.	Testing	System	Partially Implemented	See NF-R-04.	-	-
NF-U-01	[3] / Section	HMI usability	Information shall be correctly placed on	Information shown in the HMI screen should	Testing /	End-user	Implemented	Positive regarding positioning and grouping	Minor	Low



Turnaround Tool Verification Document

Issue: 1.0

Date: 16/11/2012

Requirement ID-Number	Document / Section	Requirement Description	"Shall" Statement	Verification Success Criteria	Verification Method	Verification Level	Completion Level	Final Verification Result		
								Positive / Negative (Defect?)	Defect Severity	Action Priority
	2.1		HMI screen.	be: 1. positioned correctly; 2. introduced in the right order; 3. in appropriate grouping.	Demonstration			Negative (mitigation beyond demonstrator scope) In detail view information contained in load/board tabs could be ordered or grouped (interpretation issue).		
NF-U-02	[3] / Section 2.1		Adequate colouring shall be implemented.	Under consideration of colour blindness, immediate identification of screen groups and display areas differentiation: 1. Appropriate background colouring; 2. Appropriate screen subsection colouring; 3. Appropriate colour code for particular information status/alarms is used: a) Red - Turnaround is surely delayed (Information Level 3) b) Yellow - Immediate action is required in order to keep the departure on time (Information Level 2) c) Blue - At least one process is delayed, but buffer times ensure on-time operations (Information Level 1) d) Green - Operation normal (Information Level 0)	Testing / Demonstration	End-user	Partially Implemented	Positive regarding background colour (neutral) Negative (mitigation beyond demonstrator scope) Instead of using a neutral colour (white) for all subsections, they might be distinguished from each other through colour differentiation. <i>*The demonstrator has all levels of information implemented in the back-end; however only two colour coding levels are implemented (red delay, green normal). Further colour coding is beyond the scope of the demonstrator.</i>	Minor	Low
NF-U-03	[3] / Section 2.1		Appropriate fonts shall be implemented.	The implemented fonts are: 1. common; 2. of appropriate size to enable easy key information reading.	Testing / Demonstration	End-user	Implemented	Positive Common fonts are used. Zoom in/out can be used to adapt font size for easy key information reading.	-	-
NF-U-04	[3] / Section 2.1		Appropriate unit systems shall be implemented.	The implemented unit systems are uniform.	Testing / Demonstration	End-user	Partially Implemented	Positive	-	-

Table 3. TITAN non-commercial demonstrator non-functional requirements verification matrix - Final



Turnaround Tool Verification Document

Issue: 1.0

Date: 16/11/2012

Requirement ID-Number	Document / Section	Requirement Description	"Shall" Statement	Verification Success Criteria	Verification Method	Verification Level	Completion Level	Final Verification Results		
								Positive / Negative (Defect?)	Defect Severity	Action Priority
F-S-AD-01	[3] / Section 2.1	Administration ⁴	Administrator shall manage users and their sessions.	Through a simple client administrator adds/deletes users and edits their profiles and roles. Administrator manages user sessions by checking credentials and issuing unique session IDs. After initial authentication, session IDs are used for maintaining message authenticity, until session timeout.	Testing	System (Admin)	Implemented	Positive User session management is out of scope for the demonstrator.	-	-
F-S-AD-02	[3] / Section 2.1		Administrator shall manage services and their sessions.	Through a simple client administrator starts up/shuts down services and manages their sessions.	Design / Development	System (Admin)	Implemented	[2] / Sections 4.4.2, 4.5 and 7.3, Figures 1, 3 and 6 Service start/stop can be implemented by 3 rd party servlet container - services are logical. See also NF-NA-00.	-	-
F-S-SAU-01	[3] / Section 2.1	System access and use	Users shall log in/out of TIS successfully (from the user's point of view).	Users log in TIS that subscribes them to services based on their roles.	Testing	End-user	Implemented	Positive User does not need to log into the system through service-dedicated web clients.	-	-
F-S-SAU-02	[3] / Section 2.1		Services shall subscribe to TIS for getting necessary information.	Services tell TIS/administrator that they need to get notified about specific event types they subscribe for (continuously or temporarily).	Design / Development	Service	Implemented	[2] / Sections 4.4 and 4.5, Figures 3 and 6 TITAN clients do not need to subscribe to specific services. So, if the services work we can assume that service work with TIS. See also F-S-AD-02.	-	-
F-S-EH-01	[3] / Section 2.1	Event handling (submission and distribution)	Events triggered by end users shall be processed via services to TIS (first submission or updating).	End users submit newly available information regarding events (incl. milestones and warnings) they are responsible for to services.	Testing	End-user	Not Implemented	User-client interaction is beyond the scope of the demonstrator.	-	-
F-S-EH-02	[3] / Section 2.1		TIS notifies subscribed services and users accessing them about new/updated events.	On the basis of the subscribers' list notifications are sent to services subscribed for specific event types by TIS.	Testing	System (TIS)	Implemented	Positive a) Flight status in summary view is automatically updated. b) Flight list of summary view can be updated, though either manually or after redirecting to it from detail view. - No problem when processing to next day; - Flight ordering is not confusing. Negative (mitigation beyond demonstrator scope) a) The user must periodically refresh the summary view or redirect to it from detail view for getting an updated view of the flight list. b) Completed flights are filtered out only once the list is full and next day comes (00:00). c) When processing to next day (00:00) completed milestones turn to incomplete again. d) The user must reload/refresh the detail view himself to	Minor	Low

⁴ Verification of the administration requirements depends on the level of detail of the administration interface to be finally implemented by the development team.



Turnaround Tool Verification Document

Issue: 1.0

Date: 16/11/2012

Requirement ID-Number	Document / Section	Requirement Description	"Shall" Statement	Verification Success Criteria	Verification Method	Verification Level	Completion Level	Final Verification Results		
								Positive / Negative (Defect?)	Defect Severity	Action Priority
								get information updates (i.e. flight status) with the exception of the Gantt diagram that is updated in real time. <i>* Long turnarounds (i.e. arriving early today and departing tomorrow) are filtered out and shown again on next day.</i>		
F-S-IC-00	[3] / Section 2.1	Information classification ⁵	TIS classifies information to be distributed to services and users.	The system assesses the levels of information upon the data available.	Design / Development	System (TIS)	Not Implemented	[2] / Section 4.4.1.1 This is not a key feature for a demonstrator; however it is important for a future commercial tool.	-	-
F-S-ASRF-00	[3] / Section 2.1	A/c status reporting / forecasting	Upon users' requests ASRS shall report and forecast a/c status back to them.	ASRS gathers a/c status relevant information from all users involved in its turnaround and uses it for: 1. reporting the actual status and 2. forecasting a near-future status of a specified a/c including milestone information. Upon stepping a Milestone forward, ASRS broadcasts a notification event to the TIS.	Testing	Service (ASRS)	Partially Implemented	Positive regarding reporting a) The following flight status items were received; approach, landed, taxi, in, unloading, unloaded, taxi out, en-route. b) Flight status updates inconsistencies for short turnarounds (missing milestones) are effectively solved by reducing the number of the flight status items. Negative (mitigation beyond demonstrator scope) On ETOT flight status "taxi out". <i>* Forecasting functionality is beyond demonstrator scope.</i>	Minor	Low
F-S-PBS-01	[3] / Section 2.1	Passenger / Baggage search	ASRS shall provide deadline information for finding missing passenger regarding on-time performance.	ASRS combines TIS data with output data of a/c status forecaster to provide so called "maximum time to find" type information when a passenger is missing.	Testing	Service (ASRS)	Partially Implemented	Positive This information is provided when clicking on the missing passenger notification bar, which enables display of an alert page. Negative (mitigation beyond demonstrator scope) The new alert page arising has no scroll bar in case that more than one passenger is missing. <i>* Finding the missing passenger is beyond demonstrator scope.</i>	Minor	Low
F-S-PBS-02	[3] / Section 2.1		PFIS shall search for last trace of a missing passenger (i.e. by concessionaires) in order to give a safe prediction about his position.	PFIS mines data in the TIS log to: 1. find "last seen" type records related to specified passengers; 2. guess their location within the terminal building.	Testing	Service (PFIS)	Partially Implemented	Positive Notification appears when a passenger is missing indicating the flight number. Negative (mitigation beyond demonstrator scope) a) If more than one alerts must be raised at the same time, only one notification is shown or none at all; however red colour code is applied. b) Notification lasts only for a second and disappears after manual refreshing. If the view is not manually refreshed, the notification does not disappear.	Minor	Low

⁵ Verification of the information classification requirement depends on the level of detail of the information classification logic to be finally implemented by the development team. This requirement is indirectly considered in the formulation of requirements F-UI-D-02/03 of Table 4.



Turnaround Tool Verification Document

Issue: 1.0

Date: 16/11/2012

Requirement ID-Number	Document / Section	Requirement Description	"Shall" Statement	Verification Success Criteria	Verification Method	Verification Level	Completion Level	Final Verification Results		
								Positive / Negative (Defect?)	Defect Severity	Action Priority
								c) Notification bar is not available in detail view. <i>* Finding the missing passenger is beyond demonstrator scope.</i>		
F-S-PBS-03	[3] / Section 2.1		PFIS shall generate a PNR formatted output for a missing passenger.	Upon request PFIS provides PNR-s of specified passengers.	Testing	Service (PFIS)	Not Implemented	Missing passenger ID is given in the corresponding alert page. However, no compliance with any formal guidelines is possible due to lacking requirement.	-	-
F-S-PBS-04	[3] / Section 2.1		BFIS shall search for last trace of a missing passenger's baggage (i.e. by baggage screening station) in order to give a safe prediction about its position.	BFIS mines data in the TIS log to: 1. find "last seen" type records related to specified bag-tags; 2. guess their location within the airport or name the containing pallet.	Testing	Service (BFIS)	Partially Implemented	Positive ID and location (last trace) of the baggage of the missing passenger is provided and an estimated unloading time too. <i>* Finding the missing passenger is beyond demonstrator scope.</i>	-	-

Table 4. TITAN non-commercial demonstrator general functional requirements verification matrix - Final



Turnaround Tool Verification Document

Issue: 1.0

Date: 16/11/2012

Requirement ID-Number	Document / Section	Requirement Description	"Shall" Statement	Verification Success Criteria	Verification Method	Verification Level	Completion Level	Final Verification Results			
								Positive / Negative (Defect?)	Defect Severity	Action Priority	
F-UI-C-01	[3] / Section 2.1	UI common	Login /Logout information	User information (name, profile) shall be visible after successful login and logout shall be easily executable.	After successful login, user name is displayed in a consistent UI location and user profile pops up i.e. by clicking on it. Logout/confirmation (i.e. pop up window) are possible through a logout button/link.	Testing / Demonstration	End-user	Implemented	Positive	-	-
F-UI-C-02	[3] / Section 2.1		Federated login	Login shall be centralized to a login screen.	Upon first http request to any TITAN domains, the login screen pops up, and then the user is redirected to the requested page.	Testing / Demonstration	End-user	Implemented	Positive	-	-
F-UI-C-03	[3] / Section 2.1		Time display	Time shall be visible to all users.	Common local time is displayed permanently on all UIs.	Testing / Demonstration	End-user	Implemented	Positive Time display issues are effectively solved; times given (running) in summary and detail view coincide with each other (incl. actual time line in Gantt diagram).	-	-
F-UI-C-04	[3] / Section 2.1		Touch optimization	The interacting UI shall be appropriately sized.	The views are size-optimized to enable convenient use of both touch-enabled and conventional displays on desktops and tablets (i.e. iPads).	Testing / Demonstration	End-user	Implemented	Positive regarding avoidance of right/double clicks (stationary PCs) <i>* Only a stationary-PC-compatible web application was used for testing purposes by the verification team; however, touch and view optimization for mobile devices was verified by the development team that was equipped with a tablet device (iPad), though not tested on all mobile devices.</i>	-	-
F-UI-C-05	[3] / Section 2.1		Data refreshment / update	Data on the current UI view shall be automatically refreshed as soon as their updates shall be made available.	The web UI is designed to enable refreshment of individual page portions containing updated data without needing to reload the whole page at once.	Testing / Demonstration	End-user	Implemented	Positive a) Flight status information in summary view is automatically updated. b) Following information is shown in the process-dedicated tabs: - GEN tab: All fields (except for actual departure time) - CHKIN tab: All fields with "planned" values - UNLOAD tab: All fields - DEBRD tab: "Gate allocated" field - LOAD tab: "Stand allocated, start/end of baggage load" fields - BOARD tab: "Gate allocated" field - STARTUP tab: "Stand allocated" field Negative (mitigation beyond demonstrator scope) a) The user must reload/refresh the detail view to get information updates with the exception of the Gantt diagram that is updated in real time (actual time, landing/departure milestones) b) Information in the process-dedicated tabs of the detail view is shown only in the following flight	Minor	Low



Turnaround Tool Verification Document

Issue: 1.0

Date: 16/11/2012

Requirement ID-Number	Document / Section	Requirement Description	"Shall" Statement	Verification Success Criteria	Verification Method	Verification Level	Completion Level	Final Verification Results			
								Positive / Negative (Defect?)	Defect Severity	Action Priority	
								status updates; "landed"; "taxi"; "in"; "unloading"; "unloaded". They disappear after manual refreshment of the detail view. Baggage loading and unloading start/end times wrongly coincide with each other. c) The user must periodically refresh the summary view or redirect to it from detail view for getting an updated view of the flight list. (See F-S-EH-02)			
F-UI-S-01	[3] / Section 2.1	UI summary (default) screen view	Layout	Specific (common) flight information shall be fixed presented on the UI default screen view.	Common flight information such as status, number, departure/destination airport and EIBT/EOBT is fixed, while other such as a/c type or number of passengers can be replaced.	Testing / Demonstration	End-user	Partially Implemented	Positive * Instead of EI/OBT, ELDT and ETOT are used as fixed information. * The demonstrator has all levels of information implemented in the back-end; however only two colour coding levels are implemented (red delay, green normal). Further colour coding is beyond the scope of the demonstrator.	-	-
F-UI-S-02	[3] / Section 2.1		Warnings	Flight status warnings shall be related to a specific colour code.	Colour coding (NF-U-02) is used in close relation to the fixed columns (i.e. via a frame) and the replaceable columns as well	Testing / Demonstration	End-user	Implemented	Positive * The demonstrator has all levels of information implemented in the back-end; however only two colour coding levels are implemented (red delay, green normal). Further colour coding is beyond the scope of the demonstrator. See also NF-U-02.	-	-
F-UI-S-03	[3] / Section 2.1		Customizability	The UI default screen view shall be user-customizable regarding non-fixed flight information enabling also save option of customized layout.	Unnecessary flight information columns can be replaced by the desired ones selected from a respective list* (that i.e. pops up when clicking on a columns header). Server registers the layout changes for the particular user, who gets his last saved UI layout when logging in from a different client. <small>* i.e. a/c type, tail number, passenger number, C/A/ETOT, E/ALDT, AI/OBT, status, requested services etc.</small>	Testing / Demonstration	End-user	Implemented	Positive	-	-
F-UI-S-04	[3] / Section 2.1		Navigation	Further flight specific information shall be accessed through the UI default screen view.	The UI detail screen view of a particular flight pair comes up through the UI default screen view (i.e. by clicking on fixed flight information columns).	Testing / Demonstration	End-user	Implemented	Negative (mitigation beyond demonstrator scope) The client does not remember the flight clicked on and cannot return to this place in the summary view list when redirecting from detail view.	Minor	Low
F-UI-D-01	[3] / Section 2.1	UI detail screen view	In line with UI summary (default) screen view requirements	UI summary screen view requirements regarding layout, warnings, customizability and navigation shall apply to the detail view too.	Fixed flight information is displayed similarly to the summary view. Colour coding similar to that of the summary view is used on tabs bar and Gantt views of different tab panels. Unnecessary flight data (of a customisable list) can be replaced by the desired ones and customized layout can be saved and	Testing / Demonstration	End-user	Implemented	Positive a) regarding customizability b) regarding retrieving summary view * For colour coding and fixed information see also F-UI-S-01/02.	-	-



Turnaround Tool Verification Document

Issue: 1.0

Date: 16/11/2012

Requirement ID-Number	Document / Section	Requirement Description	"Shall" Statement	Verification Success Criteria	Verification Method	Verification Level	Completion Level	Final Verification Results		
								Positive / Negative (Defect?)	Defect Severity	Action Priority
				retrieved just like in the summary view. Summary view can be retrieved with the closed flight still in focus (i.e. by clicking on the fixed flight data or a "back" button).						
F-UI-D-02	[3] / Section 2.1	Active key information on a/c turnaround processes	The actual progress of a/c turnaround processes related to milestones shall be displayed through tabs containing active (real time) textual key content in line with the applied colour coding.	In line with the applied colour coding, active textual key content provided through a clickable tabs bar informs about the level of completion of particular a/c turnaround processes related to critical milestones. Following tabs are available: general flight information, check-in, passenger security control, passenger (de)boarding, baggage (un)loading, other services (cleaning/catering), fuelling, start-up and de-icing. Following common messages is displayed: 1. Operations normal (Green – information level 0) 2. Not applicable (Gray) 3. Short status message introduced by the user 4. The highest information level or "n ISSUES" text, where "n" is the number of information of similar level.	Testing / Demonstration	End-user	Partially Implemented	Positive regarding the tabs provided * The tabs indicate some of the active key information specified. Colour correlation is beyond the scope of the demonstrator. (See also F-S-EH-01)	-	-
F-UI-D-03	[3] / Section 2.1	Active detailed information on a/c turnaround processes	All information capable of giving a detailed view over the conduction of a/c turnaround processes related to milestones shall be displayed in an active (real time) process detail section in line with the applied colour coding and the information overload prevention.	In line with the applied colour coding, an active process detail section can be accessed by selecting one of the clickable tabs for the processes listed in F-UID-02 informing about the conduction of particular a/c turnaround processes and granting particular users write access over particular information. The tabs listed in F-UID-02 apply also here. In compliance with information overload prevention, the process detail section contains following minimum information clusters ⁶ : 1. Process start/end time and milestone information; 2. Gantt diagram of affected flights/processes in case of expected delay in the current process; 3. Stand/gate number information; 4. Process-specific information.	Testing / Demonstration	End-user	Implemented	Positive a) Process detail section is accessed through clicking on tabs. b) All necessary tabs are provided (see also F-UI-D-02). c) A detailed turnaround progression chart (turnaround viewer) can be accessed by clicking on the Gantt chart providing information on process progression. Negative (mitigation beyond demonstrator scope) a) Some sub-process-specific information is missing – Gate open/close times are provided in de-boarding instead of boarding tab. – Gantt diagram is not provided in boarding and loading tabs. * No PNL information in check-in tab as no compliance with any formal guidelines is	Minor	Low

⁶ For further details on minimum content of active detailed information of each tab please see Appendix!



Turnaround Tool Verification Document

Issue: 1.0

Date: 16/11/2012

Requirement ID-Number	Document / Section	Requirement Description	"Shall" Statement	Verification Success Criteria	Verification Method	Verification Level	Completion Level	Final Verification Results		
								Positive / Negative (Defect?)	Defect Severity	Action Priority
								<p><i>possible due to lacking requirement.</i></p> <p>b) Gantt diagram still has some minor inefficiencies</p> <ul style="list-style-type: none"> - Only the progression of the selected flight and not that of the flights it affects is provided. - In case of delay the corresponding milestone time stamps in the Gantt diagram are not coloured accordingly (red). - In some cases no scroll bar is available (although necessary). - For long turnarounds status of completed milestones (i.e. arrival) turns into incomplete again (00:00). <p><i>* Colour correlation is beyond the scope of the demonstrator.</i></p> <p><i>* Write access is beyond the scope of the demonstrator.</i></p>		

Table 5. TITAN non-commercial demonstrator specific UI functional requirements verification matrix - Final



4. CONCLUSIONS

The verification of the TITAN non-commercial demonstrator against the specification requirements was conducted stepwise complying with the agile development process implemented. Having progressively verified 5 tool pre-releases of increasing maturity, the conclusions of the final verification are drawn here.

The delivered non-commercial demonstrator serves as a subset of the final commercial TITAN tool, where enough of the requirements are implemented to effectively execute the selected/restricted operational scenario. On the other side, the verification plan sets all the requirements that a future production strength commercial tool should meet in an A-CDM environment, therefore verifiable through well defined so called test (use) cases.

For the above reasons, verification was done against expectations for a demonstrator and not against demanding requirements for a commercial tool. Some of the requirements set in the verification plan were either beyond the scope of the demonstrator and therefore not testable or were verified by other means such as “verification by design”; however this does not imply that the demonstrator is not capable and has failed to meet expectations. This can be attributed, therefore, to the fact that although most of the anticipated functionalities already exist at the back-end, they cannot be accessed or the output cannot be seen. Most (if not all) of the service functionality for instance is actually available from the back-end; however there is no interface to access it (all services are implemented). The missing passenger scenario is also fully implemented; however no input in the client side is available (no user-client-interaction) to modify the server side. Although the conceptualised levels of information are mostly implemented in the back-end, the demonstrator has only two (colour) levels. Verification of such requirements goes beyond the scope of the demonstrator.

The results obtained from the final verification were satisfactory for the demonstrator, which met basic requirements or alternatively all requirements that were in its scope. Any inefficiency identified was of minor severity and low action/repair priority neither causing any failure nor impairing usability of the demonstrator. They are considered as improvement suggestions for developing a final robust tool being however beyond the scope of the demonstrator (i.e. enhancement of the colour-coding system, activation of user-client-interaction, access to further service functionalities implemented in the back-end). Examples of what would lead from a non-commercial demonstrator to a robust production strength commercial tool are listed below:

- improvement of HMI customisability to users’ needs and HMI delivery of system interaction status and the information colour-coding system;
- improvement of system reliability diagnostics (i.e. notification on service availability);
- full administration capabilities (i.e. service administration);
- greater automation of the data updating system (client summary/detail view);
- user-client-interaction (write access);
- implementation of further service-specific functionalities
 - near-future a/c status or knock-on delays forecast;
 - passenger/baggage “search & find” (data mining and intelligent retrieval of trend data in the TIS);
- provision of active key information on turnaround sub-process progression and improvement of the turnaround progression control functionality (Gantt diagram).



APPENDIX - VERIFICATION RESULTS PRIOR TO FINAL RELEASE

In the following matrices only most relevant information is retained (columns 1-4); for further information please see Table 3, Table 4 and Table 5. The results of testing each release against the specified requirements are summarized separately. The completion level of each requirement to be verified, the related verification result and the severity and action priority of eventual defects are provided for each release separately.

Following tests were performed:

- Iteration 03 - tests were exclusively related to modification of the available default dataset:
 - Modification of the default user setup (user access data);
 - Modification of the default a/c setup (a/c tail number, type and maximum passenger number);
 - Modification of the default flight setup (flight code, a/c, airport of departure/destination, scheduled/estimated/actual arrival and departures times, number of passenger);
 - Modification of the default setup for passenger and baggage generation (principally % of flights with x% of passengers for this iteration).
- Iteration 04 - tests were exclusively related to the use of the CDM emulator in connection with modification of the available default dataset:
 - Modification of the default flight setup in respect with the number of flights varying between small⁷ flight sample (1 to 14 flights), intermediate⁶ flight sample (up to 236 flights) and large⁶ flight sample (up to 529 flights);
 - Modification of the emulation speed with an increase varying between medium (2.0-20.0), high (20.0-40.0) and very high (200.0-1000.0);
 - Modification of the emulation start time set before or after the earliest time identified in the particular flight dataset;
 - Combination of the above modification alternatives.
- Iteration 05 - tests were exclusively related to the use of the CDM emulator in connection with modification of the available default dataset:
 - Integration of new flight datasets
- Iteration 06/06.1/07 - tests were exclusively related to the use of the CDM emulator in connection with modification of the available default dataset:
 - Running of the missing passenger scenario
 - Implementation of flight dataset from Palma de Mallorca airport

⁷ The size of the flight sample is characterized not only by the number of flights it contains but also by their distribution over time; i.e. larger flight samples may also contain significant time gaps between flights especially in the transition between consecutive days, when some flights may arrive late on previous day (when traffic is reduced) and depart on next day.



Turnaround Tool Verification Document

Issue: 1.0

Date: 16/11/2012

Requirement ID-Number	Requirement Description	Iteration 03				Iteration 04				Iteration 05				Iteration 06 (Summary up to 06)				Iteration 6.1					
		Completion Level	Verification Result			Completion Level	Verification Result			Completion Level	Verification Result			Completion Level	Verification Result			Completion Level	Verification Result				
			Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority		
NF-NA-00	Network architecture	Partially Implemented	N/V (until the system is fully operational)			Partially Implemented	No change to iteration 03			Partially Implemented	No change to iteration 04			Partially Implemented	N/V by the verification team JEP comment As will be made clear in the design docs, services are more logical than physical, so administration is fairly much a non issue. In addition, services are implemented as Java servlets and so can be managed by the servlet container (e.g. Tomcat) For that above reason, development team can assure that.			Partially Implemented	To be verified by design and development Developers' confirmation necessary Full reference to design document after its delivery Services are more logical than physical, so administration is fairly much a non issue. They are implemented as Java servlets and so can be managed by the servlet container (e.g. Tomcat)				
NF-PD-00	Platform dependency	Not Implemented				Not Implemented				Not Implemented				Not Implemented				Partially Implemented	Positive HMI customization is provided for the role of the gate allocator, where the client has a "proof of concept" display relating to this role's tasks (for future use).				
NF-R-01	Reliability	Partially Implemented	N/V (not available in this iteration as users cannot exchange data; instead, only the database can be modified manually and the changes shown in the UI.)			Partially Implemented	N/V (not available in this iteration as users can only get updates of a restricted data spectrum; they cannot send any data; updates shown on the UI are produced either after database modification or through CDM emulation.)			Partially Implemented	No change to iteration 04			Partially Implemented	N/V as users can get updates of a restricted data spectrum but cannot send any data. JEP comment Error messages are shown on system failure. If no error occurs, data transfer was successful. So, is it possible for the verification team to generate error messages, interfere with the data exchange and check whether error messages occur? If not, then the development team must assure that.			Partially Implemented	Positive regarding data exchange process status A coloured box near the user name indicates the state of server communications. When it is green, communication with the server (therefore services) is stable and when it is red the server (and services) is non-responsive. Dots in the box indicate client side health. * User-client interaction (user provides data in real time) is mostly a security-driven feature to be tested in a future production strength commercial tool, and therefore not too important for the demonstrator.				
NF-R-02		Not Implemented				Not Implemented				Not Implemented				Partially Implemented	Positive given the following JEP comment Demonstrator's heartbeat is the clock that displays the server time; if the clock is ticking information exchange is working.			Partially Implemented	Positive Demonstrator's heartbeat is the clock that displays the server time; if the clock is ticking information exchange is working. See also NF-R-01.				
NF-R-03		Partially Implemented	Positive regarding getting exactly the same information (for all users without exception) Negative regarding availability The user must reload/refresh the page of the web application himself to get information updates.	Major	High	Partially Implemented	Positive regarding getting exactly the same information (for all users without exception) Negative regarding availability The user must reload/refresh the page of the web application himself to get information updates coming either from the modification of the database or from the CDM emulator. (See also F-UI-C-05) * Only for the information provided up to this iteration; i.e. only landing milestone coming from emulator and S/ELDT and S/ETOT from database!	Major	High	Partially Implemented	No change to iteration 04			Implemented	Positive regarding getting exactly the same information (for all users without exception) and availability Negative The user must reload/refresh the detail view himself to get information updates with the exception of the Gantt diagram that is updated in real time. * Only for the information provided up to this iteration!	Minor	Low	Implemented	Positive regarding getting exactly the same information (for all users without exception) and availability Negative The user must reload/refresh the detail view himself to get information updates with the exception of the Gantt diagram that is updated in real time.	Minor	Low		
NF-R-04		Not Implemented				Not Implemented				Not Implemented				Partially Implemented	N/V by the verification team JEP comment As above, services are logical. If TITAN is working, services are available. So, is it possible for the verification team to test warning messages in case of service unavailability? If not, development team must assure that.			Average	Normal	Partially Implemented	To be verified by design Developers' confirmation necessary Full reference to design document after its delivery Such notifications are not of great importance for a user. Though not important for a demonstrator, it is important for a commercial tool.		
NF-A-00		Availability	Not Implemented				Not Implemented				Not Implemented				Partially Implemented	See above NF-R-04			Average	Normal	Partially Implemented	See NF-R-04.	



Turnaround Tool Verification Document

Issue: 1.0

Date: 16/11/2012

Requirement ID-Number	Requirement Description	Iteration 03				Iteration 04			Iteration 05			Iteration 06 (Summary up to 06)				Iteration 6.1					
		Completion Level	Verification Result			Completion Level	Verification Result			Completion Level	Verification Result			Completion Level	Verification Result			Completion Level	Verification Result		
			Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority
NF-U-01	HMI usability	Partially Implemented	<p>Positive regarding positioning and grouping</p> <p>Negative regarding order</p> <p>a. In summary view the chronological ordering of the flight pairs can be misleading at first (although understood later on).</p> <p>b. In detail view load/board tabs contain too much information that should be ordered or grouped (if possible).</p>	Minor	Low	Partially Implemented	<p>Negative</p> <p>a. In summary view the chronological ordering of the flight pairs can be misleading at first (although understood later on). It would help though if the reason of ordering (landing or take-off time) was highlighted (Bold).</p> <p>b. In detail view load/board tabs contain too much information that should be ordered or grouped (if possible).</p>	Average	Normal	Partially Implemented	No change to iteration 04	Implemented	<p>Positive regarding positioning and grouping; flight ordering issue in summary view is solved</p> <p>Negative</p> <p>In detail view load/board tabs contain too much information that should be ordered or grouped (interpretation issue).</p>	Minor	Low	Implemented	<p>Positive regarding positioning and grouping; flight ordering issue in summary view is solved</p> <p>Negative</p> <p>In detail view load/board tabs contain too much information that should be ordered or grouped.</p>	Minor	Low		
NF-U-02		Partially Implemented	<p>Positive regarding background colour (neutral)</p> <p>Negative regarding screen subsection colour</p> <p>a. Instead of using a neutral colour (white) for all subsections, they might be distinguished from each other through colour differentiation.</p> <p>N/V regarding colour code</p> <p>Colour code is implemented partially.</p>	Minor	Low	Partially Implemented	No change to iteration 03	Partially Implemented	No change to iteration 04	Partially Implemented (Colour coding of alerts TBD)	<p>Positive regarding background colour (neutral)</p> <p>Negative regarding screen subsection colour (interpretation issue)</p> <p>a. Instead of using a neutral colour (white) for all subsections, they might be distinguished from each other through colour differentiation.</p> <p>b. In contrast to previous iterations colour code is not implemented anymore (no read colour frame when delay occurs or missing passenger alert is raised; furthermore only red (surely delayed) and green (normal) colour implemented).</p>	Major	High	Partially Implemented (Colour coding of alerts TBD)	<p>Positive regarding background colour (neutral)</p> <p>Negative</p> <p>Instead of using a neutral colour (white) for all subsections, they might be distinguished from each other through colour differentiation.</p> <p>* The demonstrator has all levels of information implemented in the back-end; however only two colour coding levels are implemented (red delay, green normal). Further colour coding is beyond the scope of the demonstrator.</p>	Minor	Low				
NF-U-03		Partially Implemented	<p>Positive</p> <p>a. Common fonts are used.</p> <p>b. Zoom in/out can be used to adapt font size for easy key information reading.</p>	-	-	Partially Implemented	No change to iteration 03	Partially Implemented	No change to iteration 04	Implemented	<p>Positive</p> <p>a. Common fonts are used.</p> <p>b. Zoom in/out can be used to adapt font size for easy key information reading.</p>	-	-	Implemented	<p>Positive</p> <p>a. Common fonts are used.</p> <p>b. Zoom in/out can be used to adapt font size for easy key information reading.</p>	-	-				
NF-U-04		Not Implemented				Not Implemented		Not Implemented		Partially Implemented	<p>Positive</p>	-	-	Partially Implemented	<p>Positive</p>	-	-				

Table 6. TITAN non-commercial demonstrator non-functional requirements verification matrix - Prior to final



Turnaround Tool Verification Document

Issue: 1.0

Date: 16/11/2012

Requirement ID-Number	Requirement Description	Iteration 03				Iteration 04				Iteration 05				Iteration 06 (Summary up to 06)				Iteration 6.1			
		Completion Level	Verification Result			Completion Level	Verification Result			Completion Level	Verification Result			Completion Level	Verification Result			Completion Level	Verification Result		
			Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority
F-S-AD-01	Administration	Partially Implemented	NV (no separate administrator interface in this iteration) Positive Users can be added, deleted and removed and their access (login) data can be changed. Session ID is issued automatically. <i>* This is done manually by changing the user setup in the database (only for this iteration)!</i>	Minor	Low	Implemented	Positive An administration-dedicated interface is provided, through which the administrator can successfully add and delete users as well as change their ID and access (login) data.	-	-	Implemented	Positive Bug fixes confirmed (no identical users, no self-destruction, username/password required)	-	-	Implemented	Positive	-	-	Implemented	Positive	-	-
F-S-AD-02		Not Implemented			Not Implemented				Not Implemented				Implemented	NV by the verification team JEP comment Service start/stop can be implemented by 3rd party servlet container - services are logical. So, development team must assure that.			Implemented	To be verified by design Developers' confirmation necessary Full reference to design document after its delivery Service start/stop can be implemented by 3rd party servlet container - services are logical. See also NF-NA-00			
F-S-SAU-01	System access and use	Partially Implemented	Positive regarding log in Negative regarding log out In Firefox 6.0.2 logout button does not function correctly or not at all; clear the browser history after logout and then reload the page or just wait some seconds and then reload/refresh. <i>* No service-dedicated web clients are provided in this iteration!</i>	Average	Normal	Implemented	Positive The logout-related issue is successfully mitigated. <i>* No service-dedicated web clients are provided in this iteration!</i>	-	-	Implemented	No changes to iteration 04			Implemented	Positive <i>* No service-dedicated web clients available!</i>			Implemented	Positive User does not need to log into the system through service-dedicated web clients.	-	-
F-S-SAU-02		Not Implemented			Not Implemented				Not Implemented				Implemented	NV by the verification team JEP comment TITAN clients do not need to subscribe to specific services. So, if the services work we can assume that service work with TIS. So, the development team must assure that.			Implemented	To be verified by design Developers' confirmation necessary Full reference to design document after its delivery TITAN clients do not need to subscribe to specific services. So, if the services work we can assume that service work with TIS. See also F-S-AD-02.			
F-S-EH-01		Not Implemented			Not Implemented				Not Implemented				Not Implemented	RWT comment Users cannot interact with the tool. Events can only be modified through manipulation of data in the CDM emulator. However, even if changes of the available (up to this iteration) data are introduced, this is not depicted in the client and has no effect. In this way, the high-level use cases cannot be tested as they pre-assume user-client-interaction. JEP comment Are TITAN clients supposed to modify data, or is this done by 3rd party systems and TITAN updates data and notifies observers as required?			Not Implemented	User-client interaction is beyond the scope of the demonstrator.			
F-S-EH-02	Event handling (submission and distribution)	Not Implemented			Partially Implemented	Negative Notifications about updates of information available up to this iteration are not obtained automatically; this is related to the UI refreshment issue. (See also F-UI-C-05) <i>* Only for the information provided up to this iteration; i.e. only landing milestone coming from emulator and S/ELDT and S/ETOT from database!</i>	Major	High	Partially Implemented	No changes to iteration 04			Implemented	Positive regarding automatic refreshing of the summary view (flight status). Negative a. The user must periodically refresh the summary view himself (after processing to next day 0:00) in order for completed/departed flights to be replaced by subsequent/arriving ones. The new flight ordering is confusing; when stepping into next day some flights having arrived on previous day and departing later on current day wrongly disappear, others already completed turn into incomplete again (manual refreshing necessary to finally see them disappear) and some others are wrongly shown to depart/arrive on current day although scheduled for previous/next day. b. The user must reload/refresh the detail	Average	Normal	Implemented	Positive regarding automatic refreshing of the summary view (flight status). Negative a. The user must periodically refresh the summary view himself (after processing to next day 0:00) in order for completed/departed flights to be replaced by subsequent/arriving ones. Even after that the new flight ordering is confusing/inconsistent - some flights having arrived on previous day and departing later on current (next) day wrongly disappear - some flights are shown to have arrived on previous day (status "unloaded") departing on the current (next) day although they were not included in the list of the previous day; arrival milestone wrongly turns into incomplete - some flights already	Average	Normal	



Turnaround Tool Verification Document

Issue: 1.0

Date: 16/11/2012

Requirement ID-Number	Requirement Description	Iteration 03				Iteration 04				Iteration 05				Iteration 06 (Summary up to 06)				Iteration 6.1			
		Completion Level	Verification Result			Completion Level	Verification Result			Completion Level	Verification Result			Completion Level	Verification Result			Completion Level	Verification Result		
			Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority
																				<p>completed turn into incomplete again</p> <ul style="list-style-type: none"> - some flights are wrongly shown to depart/arrive on current day although scheduled for previous/next day; no milestones shown in their Gantt chart. - some flights arriving/departing later on current day have the wrong status "en-route" <p>b. The user must reload/refresh the detail view himself to get information updates (i.e. flight status) with the exception of the Gantt diagram that is updated in real time.</p> <p>(See also F-UI-C-05)</p> <p>* Only for the information provided up to this iteration!</p>	
F-S-IC-00	Information classification	Not Implemented			Not Implemented				Not Implemented				Not Implemented				Not Implemented			<p><u>To be verified by design</u> <u>Developers' confirmation necessary</u> <u>Full reference to design document after its delivery</u></p> <p>This is not a key feature for a demonstrator, however it is important for a future commercial tool.</p>	
F-S-ASRF-00	A/c status reporting / forecasting	Not Implemented			Not Implemented				Partially Implemented	<p><u>Positive</u> The following flight status items were received; arriving, landed, taxi, in.</p> <p><u>Negative</u> The corresponding times are not updated.</p>	Major	High	Partially Implemented	<p><u>Positive</u> regarding reporting The following flight status items were received; approach, landed, taxi, in, unloading, unloaded, loading, loaded, doors closed, taxi out, departure, en-route.</p> <p><u>Negative</u> a. Flight status updating is inconsistent in case of short turnarounds (i.e. <40 min); jump "unloaded"-> "loaded"/"doors closed"/"departure" while milestones in-between are missing. b. Flight status changes to "departure" 5 minutes after ETOT; on ETOT flight status "taxi out" is shown instead. c. The times corresponding to flight status changes are not shown in detail view. (See also F-UI-C-05)</p> <p>* Forecasting functionality is beyond the scope of the demonstrator.</p>	Average	Normal	Partially Implemented	<p><u>Positive</u> regarding reporting The following flight status items were received; approach, landed, taxi, in, unloading, unloaded, loading, loaded, doors closed, taxi out, departure, en-route.</p> <p><u>Negative</u> a. Flight status updating is inconsistent in case of short turnarounds (i.e. <40 min); jump "unloaded"-> "loaded"/"doors closed"/"departure" while milestones in-between are missing. b. Flight status changes to "departure" 5 minutes after ETOT; on ETOT flight status "taxi out" is shown instead. c. The times corresponding to flight status changes are not shown in detail view. (See also F-UI-C-05)</p> <p>* Forecasting functionality is beyond the scope of the demonstrator.</p>	Average	Normal	
F-S-PBS-01	Passenger / Baggage search	Not Implemented			Not Implemented				Not Implemented				Not Implemented				Partially Implemented	<p><u>Positive</u> This information is provided when clicking on the missing passenger notification bar, which enables display of an alert page.</p> <p><u>Negative</u> The new alert page arising has no scroll bar and as a result if a list of missing passengers must be shown this is not feasible.</p> <p>* Finding the missing passenger is beyond the scope of the demonstrator.</p>	Minor	Low	
F-S-PBS-02		Not Implemented			Not Implemented				Not Implemented				Partially Implemented	<p><u>Positive</u> regarding missing passenger notification Red button labelled with flight number indicates a passenger which has not boarded yet (is missing).</p> <p><u>Negative</u> a. If two alerts must be raised at the same time,</p>	Average	Normal	Partially Implemented	<p><u>Positive</u> Notification appears when a passenger is missing indicating the flight number.</p> <p><u>Negative</u> a. If two alerts must be raised at the same time, only one notification is shown.</p>	Minor	Low	



Turnaround Tool Verification Document

Issue: 1.0

Date: 16/11/2012

Requirement ID-Number	Requirement Description	Iteration 03				Iteration 04				Iteration 05				Iteration 06 (Summary up to 06)				Iteration 6.1					
		Completion Level	Verification Result			Completion Level	Verification Result			Completion Level	Verification Result			Completion Level	Verification Result			Completion Level	Verification Result				
			Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority		
														only one notification is shown. b. Notification lasts only for a second and clicking on it has no effect. After manual refreshing the notification bar disappears and so even if passenger is finally boarded with delay (before doors are closed) there is no way to check if notification is de-activated. If the view is not manually refreshed, the notification does not disappear (even after departure) preventing other notifications from being shown. c. Even if notification is shown no colour coding is applied to the particular flight meaning that any delay possibility is not taken into consideration. d. Notification bar is not available in detail view. * Finding the missing passenger is beyond the scope of the demonstrator.					b. Notification lasts only for a second. After manual refreshing the notification bar disappears and so even if passenger is finally boarded with delay (before doors are closed) there is no way to check if notification is de-activated. If the view is not manually refreshed, the notification does not disappear and colour remains red (even after flight departure). c. Notification bar is not available in detail view. * Finding the missing passenger is beyond the scope of the demonstrator.				
F-S-PBS-03		Not Implemented			Not Implemented				Not Implemented				Not Implemented				Not Implemented			Missing passenger ID is given in the corresponding alert page. However, no compliance with any formal guidelines is possible due to lacking requirement.			
F-S-PBS-04		Not Implemented			Not Implemented				Not Implemented				Not Implemented				Partially implemented			Positive ID and location (last trace) of the baggage of the missing passenger is provided and an estimated unloading time too. * Finding the missing passenger is beyond the scope of the demonstrator.			

Table 7. TITAN non-commercial demonstrator general functional requirements verification matrix - Prior to final



Turnaround Tool Verification Document

Issue: 1.0

Date: 16/11/2012

Requirement ID-Number	Requirement Description	Iteration 03				Iteration 04				Iteration 05				Iteration 06 (Summary up to 06)				Iteration 6.1			
		Completion Level	Verification Result			Completion Level	Verification Result			Completion Level	Verification Result			Completion Level	Verification Result			Completion Level	Verification Result		
			Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority
F-UI-C-01	Login /Logout information	Partially Implemented	Negative a. User profile does not appear. b. Logout is problematic (see also F-S-SAU-01).	Average	Normal	Partially Implemented	Negative User profile does not appear.	Average	Normal	Partially Implemented	No changes to iteration 04			Implemented	Negative User profile does not appear.	Average	Normal	Implemented	Negative User profile does not appear.	Average	Normal
F-UI-C-02	Federated login	Implemented	Positive	-	-	Implemented	No changes to iteration 03	-	-	Implemented	Positive Display of login page and redirect to /turnaround (summary view) works with these URLs - http://fraptitan.jeppesen.com/titan/turnaround/ - http://fraptitan.jeppesen.com/titan/login/ - http://fraptitan.jeppesen.com/titan/ - http://fraptitan.jeppesen.com/ Negative Display of login page works but redirect to requested page fails with these URLs - http://fraptitan.jeppesen.com/titan/detail/?Aircraft=AC29 (replace with any a/c) - http://fraptitan.jeppesen.com/titan/admin/	Minor	Low	Implemented	Negative Display of login page works but redirecting to requested page fails with the following URLs - http://fraptitan.jeppesen.com/titan/detail/?Aircraft=AC29 (replace with any a/c) - http://fraptitan.jeppesen.com/titan/admin/	Minor	Low	Implemented	Positive	-	-
F-UI-C-03	Time display	Partially Implemented	Negative The time given in the Gantt diagram (12:01) does not coincide with the time given on the top of the summary and detail view (02:01).	Major	High	Partially Implemented	No changes to iteration 03	-	-	Partially Implemented	Positive Time in Gantt diagram is the same as displayed in the header. Negative Time in the main and detail view is not updated automatically.	Average	Normal	Implemented	Positive Time display issues are effectively solved; times given (running) in summary and detail view coincide with each other (incl. actual time line in Gantt diagram).	-	-	Implemented	Positive Time display issues are effectively solved; times given (running) in summary and detail view coincide with each other (incl. actual time line in Gantt diagram).	-	-
F-UI-C-04	Touch optimization	Implemented	Positive regarding avoidance of right/double clicks (stationary PCs) * Only a stationary-PC-compatible web application was used for testing purposes by the verification team; however, touch and view optimization for mobile devices was verified by the development team that was equipped with a tablet device (iPad), though not tested on all mobile devices.	-	-	Implemented	No changes to iteration 03	-	-	Implemented	No changes to iteration 04			Implemented	Positive regarding avoidance of right/double clicks (stationary PCs) * Only a stationary-PC-compatible web application was used for testing purposes by the verification team; however, touch and view optimization for mobile devices was verified by the development team that was equipped with a tablet device (iPad), though not tested on all mobile devices.	-	-	Implemented	Positive regarding avoidance of right/double clicks (stationary PCs) * Only a stationary-PC-compatible web application was used for testing purposes by the verification team; however, touch and view optimization for mobile devices was verified by the development team that was equipped with a tablet device (iPad), though not tested on all mobile devices.	-	-



Turnaround Tool Verification Document

Issue: 1.0

Date: 16/11/2012

Requirement ID-Number	Requirement Description	Iteration 03				Iteration 04				Iteration 05				Iteration 06 (Summary up to 06)				Iteration 6.1			
		Completion Level	Verification Result			Completion Level	Verification Result			Completion Level	Verification Result			Completion Level	Verification Result			Completion Level	Verification Result		
			Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority
F-UI-C-05	Data refreshment / update	Partially Implemented	Negative The user must reload/refresh the whole page himself (not done automatically).	Major	High	Partially Implemented	Negative During testing particular problems are identified concerning successful information updating coming from CDM emulator. a. Only in some cases (small flight samples) information is updated automatically (without necessary user intervention). b. In most cases (large flight samples) only the status of the (chronologically) first flights is updated automatically, whereas updates on the status of the rest of the flights are obtained only after manual and periodical page refreshment by the user himself. Furthermore, even after manual refreshment the status of some flights cannot be updated (possible relation to the following error report: "com.google.gwt.user.client.rpc.StatusCodeException : 0"). Should the user be already logged in when emulator starts running, the above problem is even acuter. c. Although higher emulation speeds may be time efficient in case of large flight samples, the higher the speed (> 400.0) the higher the possibility for the user to be confronted with the above problems (especially that of unsuccessful flight status updating even after manual page refreshment). d. By setting emulation start time later than the earliest time identified in the sample, all milestones already completed when starting the emulation should be automatically updated at once. However, this is not the case for flights that arrive much earlier i.e. on previous day. * Flight status is updated only in respect with the landing milestone during CDM emulation up to this iteration!	Major	High	Partially Implemented	Negative in addition to iteration 04: - Updates are not submitted to the detailed view - Sometimes page refreshing in the detail view shows only the tail number in all rows of the left frame	Major	High	Implemented	Positive regarding automatic refreshing of flight status information in summary view Negative a. No automatic information refreshing in detail view (except for actual time running in Gantt diagram). The user must refresh by himself. The times corresponding to summary view information changes are not shown in detail view (in the corresponding tabs). b. The user must periodically refresh the summary view himself (after processing to next day 0:00) in order for completed/departed flights to be replaced by subsequent/arriving ones. The new flight ordering is confusing; when stepping into next day some flights having arrived on previous day and departing later on current day wrongly disappear, others already completed turn into incomplete again (manual refreshing necessary to finally see them disappear) and some others are wrongly shown to depart/arrive on current day although scheduled for previous/next day. c. Still some problems with when setting emulation start time later than the earliest time identified in the sample; i.e. status of flights already completed does not turn into "en-route" at once. (issue of minor importance)	Major	High	Implemented	Positive regarding automatic refreshing of flight status information in summary view Negative a. No automatic information refreshing in detail view (except for actual time running in Gantt diagram). The user must refresh by himself. The times corresponding to summary view information changes are not shown in detail view (in the corresponding tabs). b. The user must periodically refresh the summary view himself (after processing to next day 0:00) in order for completed/departed flights to be replaced by subsequent/arriving ones. The new flight ordering is confusing; when stepping into next day. (See F-S-EH-02)	Average	Normal



Turnaround Tool Verification Document

Issue: 1.0

Date: 16/11/2012

Requirement ID-Number	Requirement Description	Iteration 03					Iteration 04				Iteration 05				Iteration 06 (Summary up to 06)				Iteration 6.1		
		Completion Level	Verification Result			Completion Level	Verification Result			Completion Level	Verification Result			Completion Level	Verification Result			Completion Level	Verification Result		
			Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority
F-UI-S-01	Layout <i>UI summary (default) screen view</i>	Partially Implemented	Negative a. Departure passengers are not indicated; instead, arrival passengers and maximum passengers in the rows of in-bound and out-bound flights are given, which may be misleading for the user. b. The flight status is inconsistent i. It is not actualized ii. "Normal" appears in delayed flights too. c. Should both arrival and departure flight be delayed only one of both delays is indicated in the colour frame (i.e. "Late Dep"). Not clear how this is decided. It would be better to indicate both. d. Should estimated time be greater than the scheduled one (delay) and the a/c not yet arrived then delay is indicated (red colour). But this is not the case when the actual time is equal to estimated and together greater than the scheduled one (a/c arrived with delay); the flight is not red framed (no delay), i.e. i. schedule 12:00, estimate 12:13 and not arrived, then delay ii. schedule 12:00, estimate and actual 12:13 and departure estimate 12:15, then no delay (inconsistent) This means that when an actual value exists (milestone completed) the milestone is not taken into account anymore (inconsistent). e. Only 50 flights pro page shown, but there is no way to see those following too (maybe in a next page). * Instead of EI/OBT, ELDT and ETOT are used as fixed information.	Critical	Urgent	Partially Implemented	Positive Flight status (only in respect with the landing milestone provided up to this iteration) is actualised either automatically or manually. Negative As in previous iteration.	Critical	Urgent	Partially Implemented	no changes to iteration 04	Partially Implemented	Negative a. Departing passengers are not indicated but only arriving ones (please ignore in case that these two numbers are thought to be the equal). b. No colour coding implemented (in contrast to previous iterations); no red-colour-coded delay is indicated when i.e. the estimated arrival time is greater than scheduled one. c. Completed flights are not filtered out from the list of the summary view (see also above F-UI-05). * Instead of EI/OBT, ELDT and ETOT are used as fixed information.	Major	High	Partially Implemented	Negative a. Departing passengers are not indicated but only arriving ones. b. Completed flights are not filtered out from the list of the summary view (see also above F-UI-05). * Instead of EI/OBT, ELDT and ETOT are used as fixed information. * The demonstrator has all levels of information implemented in the back-end; however only two colour coding levels are implemented (red delay, green normal). Further colour coding is beyond the scope of the demonstrator.	Average	Normal		
F-UI-S-02	Warnings	Implemented	Positive * Only for colour coding in this iteration!	-	-	Implemented	Negative In case of already colour-coded flight pairs (i.e. red coloured frame due to delay) the applied colour is altered after flight status updating (turns to green - normal operations) not corresponding to actual situation. In order to retain the right colour coding the user must either refresh the page or select detail view and go back to (updated) summary view.	Major	High	Implemented	No changes to iteration 04	Implemented	Negative Colour coding is not implemented (although done in all previous iterations).	Critical	Urgent	Implemented	Positive * The demonstrator has all levels of information implemented in the back-end; however only two colour coding levels are implemented (red delay, green normal). Further colour coding is beyond the scope of the demonstrator. (See also NF-U-02)	-	-		
F-UI-S-03	Customizability	Implemented	Positive * Tested through logging out and in again!	-	-	Implemented	No changes to iteration 03	-	-	Implemented	No changes to iteration 04	Implemented	Positive	-	-	Implemented	Positive	-	-		
F-UI-S-04	Navigation	Implemented	Positive	-	-	Implemented	No changes to iteration 03	-	-	Implemented	No changes to iteration 04	Implemented	Negative (interpretation issue) When retaining summary view (from detail view) then the top of the list is shown and the previously selected flight must be found again.	Minor	Low	Implemented	Negative The client does not remember the flight clicked on and cannot return to this place in the summary view list.	Minor	Low		



Turnaround Tool Verification Document

Issue: 1.0

Date: 16/11/2012

Requirement ID-Number	Requirement Description	Iteration 03				Iteration 04				Iteration 05				Iteration 06 (Summary up to 06)				Iteration 6.1			
		Completion Level	Verification Result			Completion Level	Verification Result			Completion Level	Verification Result			Completion Level	Verification Result			Completion Level	Verification Result		
			Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority
F-UI-D-01	In line with UI summary (default) screen view requirements	Partially Implemented	Positive - regarding colour coding (only for this iteration) - regarding customizability (only through log out and in again) - regarding retrieving summary view * For fixed information see F-UI-S-01.	Average	Normal	Partially Implemented	No changes to iteration 03			Partially Implemented	No changes to iteration 04			Implemented	Positive - regarding customizability - regarding retrieving summary view Negative - regarding colour coding (see F-UI-S-02) * For fixed information see F-UI-S-01.	Major	High	Implemented	Positive - regarding customizability - regarding retrieving summary view * For colour coding see F-UI-S-02. * For fixed information see F-UI-S-01.	-	-
F-UI-D-02	UI detail screen view Active key information on a/c turnaround processes	Partially Implemented	Positive regarding the tabs provided NV (not available in this iteration) a. The tabs do not indicate active key information according to the common messages to be implemented. b. No colour coding is used.	-	-	Partially Implemented	No changes to iteration 03			Partially Implemented	No changes to iteration 04			Partially Implemented	Positive regarding the tabs provided NV a. The tabs do not indicate active key information according to the common messages to be implemented. b. No colour coding is used.	-	-	Partially Implemented	Positive regarding the tabs provided NV a. The tabs do not indicate active key information according to the common messages to be implemented. b. No colour coding is used. (See also F-S-EH-01)	-	-



Turnaround Tool Verification Document

Issue: 1.0

Date: 16/11/2012

Requirement ID-Number	Requirement Description	Iteration 03				Iteration 04				Iteration 05				Iteration 06 (Summary up to 06)				Iteration 6.1			
		Completion Level	Verification Result			Completion Level	Verification Result			Completion Level	Verification Result			Completion Level	Verification Result			Completion Level	Verification Result		
			Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority		Positive / Negative (Defect?)	Defect Severity	Action Priority
F-UI-D-03	Active detailed information on a/c turnaround processes	Partially Implemented	Positive a. Process detail section is accessed through clicking on tabs. b. All necessary tabs are provided (see also F-UI-D-02). Negative a. Following sub-process-specific information is missing i. Gate open/close times are provided in de-boarding instead of boarding tab. ii. Gantt diagram is not provided in boarding and loading tabs. * No PNL information in check-in tab as no compliance with any formal guidelines is possible due to lacking requirement.	Average	High	Partially Implemented	Positive As in previous iteration. Negative Additional inefficiencies are identified in the Gantt diagram i. If ELDT and ETOT are not quite close to actual time (but instead well before it), these milestones are not depicted in the Gantt diagram and no time reference appears. ii. If a flight has arrived on previous day and departs on current day, only ELDT is depicted in the Gantt diagram therefore wrongly positioned on current day and later than ETOT. No time reference appears. iii. If a flight has arrived much earlier than actual time and departs later on, both milestones are not depicted in the diagram and no time reference appears. They appear only if estimated arrival is delayed (approaching the actual time). iv. If a flight arrives and departs later on in the day, no time reference appears in the Gantt diagram although both milestones are displayed. v. In all above cases, relation to actual time is inconsistently depicted in the Gantt diagram and the user cannot understand whether the diagram refers to actual or previous/next day. * Write access is beyond the scope of the demonstrator.	Major	High	Partially Implemented	No changes to iteration 04	Major	High	Implemented	Positive a. Process detail section is accessed through clicking on tabs. b. All necessary tabs are provided (see also F-UI-D-02). c. Gantt-related inefficiencies identified in testing iteration 04 were solved. Negative a. Some sub-process-specific information is missing i. Gate open/close times are provided in de-boarding instead of boarding tab. ii. Gantt diagram is not provided in boarding and loading tabs. * No PNL information in check-in tab as no compliance with any formal guidelines is possible due to lacking requirement. b. Information overload in loading and boarding tabs c. Gantt diagram still has some inefficiencies i. Only the progression of the selected flight and not that of the flights it affects is provided. ii. In case of delay (red framed flight) the corresponding milestone time stamps in the Gantt diagram are not coloured accordingly (red). iii. In some cases no scroll bar is available (although necessary) and no milestones are shown. iv. Problems when processing to next day (00:00) - for flights having arrived on previous day no actual time line is shown and status of completed milestones turns into "incomplete" again - for flights already departed on previous day status of completed milestones turns into "incomplete" again (see also F-UI-C-05) - flights arriving and departing on next day are wrongly shown to arrival/depart on current day - for flights arriving and departing later on milestone status in Gantt diagram is "incomplete" although flight status in summary view is changing NV a. No colour coding is implemented. b. No active information on process progression is provided. * Write access is beyond the scope of the demonstrator.	Average	Normal	Implemented	Implemented	Average	Normal

Table 8. TITAN non-commercial demonstrator specific UI functional requirements verification matrix - Prior to final



Turnaround Tool Verification Document

Issue: 1.0

Date: 16/11/2012

END OF DOCUMENT