|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  |  |  | | NissanWordmark_RGB_300dpi | |
| Project Specification |  |  | |  | |
|  | TCS/FQI Analytics | | | |
|  | EQUIP Reporting / Montly Quality Report | | | |
|  |  | | Version 0.0 | | Revision Date: 1/6/2014 |

Table of Contents

[Table of Contents i](#_Toc510020434)

[About This Document iii](#_Toc510020435)

[Notes to Authors: iii](#_Toc510020436)

[Version Control iii](#_Toc510020437)

[1 High-Level Design and Requirements/Epics (Plan Phase) 1](#_Toc510020438)

[1.1 Requirements/Epics 1](#_Toc510020439)

[1.1.1 Application 1](#_Toc510020440)

[1.1.2 Infrastructure 1](#_Toc510020441)

[1.2 High-Level Functional Design 1](#_Toc510020442)

[1.2.1 Screen/Form High-Level Design 1](#_Toc510020443)

[1.2.2 Screen Flow High-Level Design 1](#_Toc510020444)

[1.2.3 Report High-Level Design 1](#_Toc510020445)

[1.2.4 I/O Interface High-Level Design 2](#_Toc510020446)

[1.3 User Roles 2](#_Toc510020447)

[1.4 High-Level Data Architecture Strategy 2](#_Toc510020448)

[1.5 High-Level Application Architecture Strategy 3](#_Toc510020449)

[1.6 High-Level Training & User Support Strategy 4](#_Toc510020450)

[1.7 High-Level Design & Requirements Approval 4](#_Toc510020451)

[2 Go/No Go (Execute Phase) R 1.0 5](#_Toc510020452)

[2.1 Detailed Data Architecture 5](#_Toc510020453)

[2.2 Detailed Application Architecture 5](#_Toc510020454)

[3 EQUIP REporting Detailed desgin 5](#_Toc510020455)

[3.1 EQUIP Reporting High Level Requirement 5](#_Toc510020456)

[3.2 Vehicle Extracts 5](#_Toc510020457)

[3.3 Warranty Extracts 5](#_Toc510020458)

[3.4 Incident Rate Reporting 5](#_Toc510020459)

[3.5 AVES 6](#_Toc510020460)

[3.6 INSPECT 6](#_Toc510020461)

[3.7 Consumer Affairs 6](#_Toc510020462)

[3.8 Techline 6](#_Toc510020463)

[4 Go/No Go (Execute Phase) R 1.1 6](#_Toc510020464)

[4.1 Detailed Data Architecture 6](#_Toc510020465)

[4.2 Detailed Application Architecture 6](#_Toc510020466)

[4.3 Detailed Test Plan 6](#_Toc510020467)

[4.4 Detailed Training & User Support 6](#_Toc510020468)

[4.5 Test Results 6](#_Toc510020469)

[4.6 Deployment Plan (Use on New Application/System only) 7](#_Toc510020470)

[4.6.1 Requested CMT (RFC for NMEX projects) 7](#_Toc510020471)

[4.7 Production Support Check list 7](#_Toc510020472)

[4.8 Vendor Support Information 8](#_Toc510020473)

[4.9 Go/No-Go Approval 8](#_Toc510020474)

About This Document

This document is a collaboration between project sponsor, business analyst / project manager, and major stakeholders after the Initiate phase has been completed. The Project Specification document may get refined and updated in the subsequent phases but should be baselined at the end of Planning.

* Guidance text and information to complete the document is hidden within each section. Click on the Show/Hide (Ctrl+Shift+\*) button to toggle display.
* In order to print the hidden text check the Print Hidden Text box found in Office Button / Word Options / Display / Printing Options

Notes to Authors:

<Note: text in pointed brackets indicate a field that should be replaced with data>

Version Control

The following table contains the version control information for this publication.

| Version | Date | Authors/Contributors | Description |
| --- | --- | --- | --- |
| 1.0 | 2/1/2017 | X321761 Andria Wilson | Initial Draft |
| 1.1 | 3/23/2018 | X321761 Andria Wilson | Revision |
| 1.2 |  |  |  |

# 

# High-Level Design and Requirements/Epics (Plan Phase)

## Requirements/Epics

[This section provides a brief description of project requirements. All application related requirements are stored in Quality Center. Embed the Quality Center Requirements Report or provide a hyperlink to the Quality Center Requirements Report stored in Clarity.

Infrastructure requirements must be documented here, and supporting documents, such as DPS, must be referenced here and stored in the Clarity 04. Requirements folder.]

The TCS/FQI Analytics project is using the Nissan SDLC Agile framework. The user stories from Service Now are included in the export below.

### Application

[This section provides a brief description of project requirements. All application related requirements are stored in Quality Center. Embed the Quality Center Requirements Report or provide a hyperlink to the Quality Center Requirements Report stored in Clarity.]

The TCS/FQI Analytics project includes the creation of a single solution to replace many existing systems and tools in the TCS/FQI group. The group uses these tools in order to make essential decisions and peform day to day operations. The new solution will conform to IS technology standards, increase supportability, decrease existing RTB costs for proprietary platforms, and increase user efficiency through performance improvements.

### Infrastructure

[If Infrastructure is required and the Infrastructure Requirements Document (IRD) is required, provide a hyperlink to the IRD and the Design Proposal Summary (DPS) documents stored in the Clarity 03 Infrastructure folder.

If Infrastructure is required and the IRD is NOT required, list any non-functional requirements/impacts including but not limited to the following:

* Capacity and performance
* Availability
* Security
* System management
* Back-up and or Disaster Recovery needs
* Other required properties of the system, such as:
* Portability
* Maintainability]
* Location of Services (where is system/infrastructure hosted? i.e. NDC, Tulsa, NTCNA, Smyrna, etc.
* Server requirements / expectations
* SAN requirements
* Testing environments
* Security requirements (firewalls, etc.)
* Facilities Readiness (Power/Cabling) (describe any facilities readiness work required, such as power work, cabling, etc.)]

Infrastructure is not required as part of the TCS/FQI Analytics project. The solution will be going on to existing infrastructure.

## High-Level Functional Design

[This section provides is the “TO BE” High-Level functional design. It should be in business process terms (business activities employees perform). This section provides a basis for communicating with end users and developers, estimating the application development effort, identifying common functionality/objects early in development, providing a basis for test case definition, and providing a basis for producing user support materials. See the Guide for PSD Section 2.3 document for more guidance.]

Details about the existing high level functional design exist as attachments to the user stories in Service Now.

### Screen/Form High-Level Design

[Describe enhancements to application screens or forms. Describe modified and new screens that will support the application enhancement requirements.

**HINT**: If the document is too large or impractical to include here, then save the design in Clarity, write a summary here and provide a hyperlink.]

The screens in the new system will replicate the screens and visualizations in the existing systems, but following Nissan standards. Screenshots of the existing systems and visualizations are attached to the relevant user stories in Service Now

### Screen Flow High-Level Design

[Describe enhancements to existing application screen flow. Describes the screen flows that have had major changes and newly defined screen flows that will support the existing application enhancement requirements. HINT: if the document is too large or impractical to include here, then save the design in Clarity, write a summary here and provide a hyperlink.]

The screen flow in the new system will replicate the screen flow in the existing system as make sense to the team with the agreement of the product owner. The agreement and documentation of the new screen flow will be documented in Service Now user stories

### Report High-Level Design

[Describe enhancements to existing application reports. Describe the modified and new defined reports that will support the existing application enhancement requirements. HINT: if the document is too large or impractical to include here, then save the design in Clarity, write a summary here and provide a hyperlink.]

The reports in the new system will replicate the reports in the existing system as make sense to the team with the agreement of the product owner. The agreement and documentation of the reports will be documented in Service Now user stories

### I/O Interface High-Level Design

[Describes enhancements to existing application I/O interface enhancements. Includes requirements for modifications to other I/O interfaces and additional I/O interface definitions. HINT: if the document is too large or impractical to include here, then save the design in Clarity, write a summary here and provide a hyperlink.]

The solution is replacing several existing applications and thus will not have any known interfaces with existing systems.

## User Roles

[Understanding who needs to access new system components or what changes need to be made to give current users access to changed components should be included in this section. Understanding who will use a system and knowledge of their characteristics is the first step in designing a user interface that meets users’ needs. User roles help identify and describe the intended users of the system so that developers can build meaningful ease-of-use features into their design. User roles serve as the basis for selecting users to participate in usability evaluations and anticipating training needs to relay new or changed functionality. In addition, these are used to identify potential locations to place various hardware and software components of the system. This section replaces the SDLC-Standard Information Security Controls.]

User roles will replicate roles in the existing applications and be defined as the team builds out the backlog further and stored in the Service Now user stories.

## High-Level Data Architecture Strategy

[Data Architecture documents provide a comprehensive architectural overview of the project. The intention behind these documents is to capture and convey significant decisions regarding data, data stores and databases that have been made on the project. This includes, but is not limited to Business Objects changes or ETL changes.

Conditions where this section must be completed:

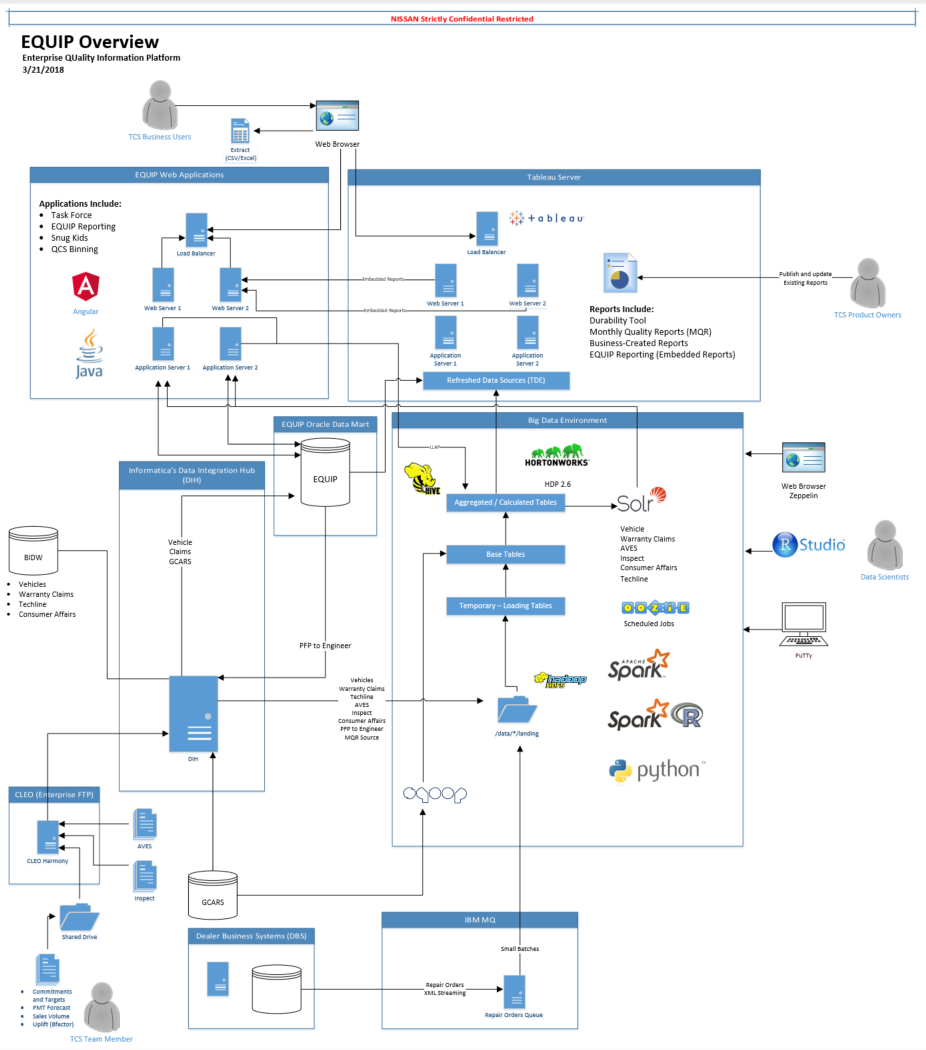
* The following Data Architecture deliverables are required for a SDLC Maintenance Project that adds or modifies database structures in an existing database supporting one or more applications:

1. Logical Data Model/Checklist
2. Physical Data Model/Checklist
3. Database change script provided by the Domain Data Architect ONLY
4. Data Architecture Document contents required up to and including phase 4.

* For Maintenance projects that DO NOT add or modify database structures in an existing database supporting one or more applications -- follow current EA Assessment process to ensure no database structures will be modified
* For new/upgrade COTS projects:
  + If NNA is responsible for installing and/or maintaining the database structures then all the maintenance project data architecture deliverables are required.
  + If the database structures are maintained by a 3rd party then no Data Architecture deliverables are required
* For Maintenance projects that will introduce new database technology to an existing application (Database technology must exist in the IT Roadmap) -- all Maintenance Project data architecture deliverables are required.
* For new development projects that replace an existing application or add a new application, the following deliverables are required:
  + Conceptual Data Model/Checklist
  + Entity Relationship Data Model/Checklist
  + All Maintenance Project data architecture deliverables]

Data models and other data-related deliverables will be stored in the Service Now user stories as the product team uncovers the requirements. Copies of those deliverables will be gathered and stored in Clarity as available.

## High-Level Application Architecture Strategy



[Briefly describe the high level strategy possibly including

* general application architecture strategy for the business domain
* code reusability
* elimination of obsolete of soon-to-be obsolete software
* upgrade to new software
* new approaches to be implement
* security, ongoing directions
* transaction response time
* logging strategies
* backup strategies
* archiving strategies
* code written using a standard language.]

## High-Level Training & User Support Strategy

[Identify if training will be required for the project.

* Tool training require for project team
* Vendor ABC will provide technical and functional training to Nissan IS
* Vendor or Nissan will provide user training on the application

Identify user support needs such as

* Vendor or Nissan will provide user manuals
* Vendor XYZ is primary support for application ABC
* An incident resolver queue will be created during the Execution phase.]

The project team does not require training. Nissan will provide user training on the application. The primary support vendor for the application is TBD depending on the makeup of the final solution.

## High-Level Design & Requirements Approval

[APPROVAL TABLE DIRECTIONS:

below is the table to record approvals. To record approvals, save the email message to your local drive then embed the emails message using Insert Object directly into table below.

When you have a subsequent update to the document that requires another approval, expand the rows and indicate the version number to which these approvals apply]

| Approver | Title | Embedded Email approval | Approvals Apply to Version |
| --- | --- | --- | --- |
| Jason Snap | Product Owner |  | 1.0 |
| Danielle Beringer | Sr. Manager, IS |  | 1.0 |

# Go/No Go (Execute Phase) R 1.0

AGILE NOTE: For Agile projects, this section is repeated for each Release. The intent is to complete this PSD Go/No Go section just prior to release once all components (SDLC Documentation, Production Support Checklist, etc.) have been approved.

## Detailed Data Architecture

[Each sprint will have a sprint grooming exercise that reviews the stories to be developed during the sprint as well as produces specifications, screen layouts, rules, etc.  The Logical Data Model for the sprint should be completed during this sprint grooming exercise. The Logical Data Model will follow all data modeling standards. The following will also be required for the sprint as needed based on the sprint goals:

* Data Profiling
* Data Access and Security Requirements
* Platform and Capacity information

The Physical Data Model and DDL needed for the sprint will be generated from the Logical Data Model. The DDL will be provided by a Nissan appointed Domain Data Architect and executed by the appropriate DBA.

Detailed data architecture information is stored in the Data Architecture folder in Clarity

## Detailed Application Architecture

[The architecture overview diagram is a schematic diagram that represents the governing ideas and candidate building blocks of an IT system. It provides an overview of the main conceptual elements and relationships in IT architecture, which frequently include candidate subsystems, components, nodes, connections, users and external systems. The architecture overview diagram should be simple, brief, clear, and understandable. A System Context Diagram is also used to:

* Identify the processes the system must perform and the structuring or relationships between those processes.
* Identify the interfaces and data exchanges between the processes.

Conditions where this section must be completed:

* For Maintenance projects that use existing technology (mainframe, web, etc….) - System context diagram only
* For Maintenance projects that will introduce new technology - System context diagram and Software Architecture Document contents required up to and including phase 4 (which is a full Software Architecture Document)
* For new/upgrade COTS projects – System context diagram and Software Architecture Document contents required up to and including phase 2 only
* For new development projects - System context diagram and Software Architecture Document contents required up to and including phase 4 (which is a full Software Architecture Document)]

Application architecture information is stored in the System Architecture folder in Clarity

# EQUIP REporting Detailed desgin

## EQUIP Reporting High Level Requirement

[Each sprint will have a sprint grooming exercise that reviews the stories to be developed during the sprint as well as produces specifications, screen layouts, rules, etc.  The Logical Data Model for the sprint should be completed during this sprint grooming exercise. The Logical Data Model will follow all data modeling standards. The following will also be required for the sprint as needed based on the sprint goals:

* Data Profiling
* Data Access and Security Requirements
* Platform and Capacity information

The Physical Data Model and DDL needed for the sprint will be generated from the Logical Data Model. The DDL will be provided by a Nissan appointed Domain Data Architect and executed by the appropriate DBA.

The EQUIP Reporting Product will replace the ALERT Endecca tool to facilitate the extraction and visualization of data for Vehicle, Warranty, PFP, AVES, INSPECT, Consumer Affairs and Techline. This will eliminate costly licensing and support for TCS.

## Vehicle Extracts

[Each sprint will have a sprint grooming exercise that reviews the stories to be developed during the sprint as well as produces specifications, screen layouts, rules, etc.  The Logical Data Model for the sprint should be completed during this sprint grooming exercise. The Logical Data Model will follow all data modeling standards. The following will also be required for the sprint as needed based on the sprint goals:

* Data Profiling
* Data Access and Security Requirements
* Platform and Capacity information

The Physical Data Model and DDL needed for the sprint will be generated from the Logical Data Model. The DDL will be provided by a Nissan appointed Domain Data Architect and executed by the appropriate DBA.

The TCS user needs the ability to extract Vehicle Information in the same format they had previously used ALERT to accomplish. The embedded attachment shows the order and the renaming of the column headers. Rows through “order 51” should be a part of MVP. The remaining extract fields can be considered part of Equip 2.0.



## Vehicle Filters

[Each sprint will have a sprint grooming exercise that reviews the stories to be developed during the sprint as well as produces specifications, screen layouts, rules, etc.  The Logical Data Model for the sprint should be completed during this sprint grooming exercise. The Logical Data Model will follow all data modeling standards. The following will also be required for the sprint as needed based on the sprint goals:

* Data Profiling
* Data Access and Security Requirements
* Platform and Capacity information

The Physical Data Model and DDL needed for the sprint will be generated from the Logical Data Model. The DDL will be provided by a Nissan appointed Domain Data Architect and executed by the appropriate DBA.

A variety of filters is needed to allow the user to slice / dice the data.

These will be grouped into a few categories that will collapse via the User Interface.

|  |  |  |
| --- | --- | --- |
| CATEGORY | ATTRIBUTE NAME | EQUIP FIELDNAME |
| ENGINE / TRANSMISSION ATTRIBUTES | Drivetrain Code | DRV\_TRN\_CD |
|  | Engine Model Code | VHCL\_ENGN\_MDL\_CD |
|  | Engine Serial Number | VHCL\_ENGN\_SRL\_NB |
|  | Transmission Type Code | TRNMSN\_TYP\_CD |
|  | Transmission Type Name | TRNMSN\_TYP\_NM |
|  | Vin\_Digit\_4 | VHCL\_IDFCTN\_4\_DGT\_NB |
| GENERAL ATTRIBUTES | Associate Distributor Code | ASCT\_DSTRBR\_CD |
|  | Company Vehicle Indicator | CV\_IN |
|  | Crushed Indicator | VHCL\_CRSHD\_IN |
|  | Emission Certificate Number | EMSN\_CRTFCT\_NB |
|  | Emission Type Code | EMSN\_TYP\_CD |
|  | Fleet Indicator | VHCL\_FLT\_IN |
|  | Fleet Sales Type Code | VHCL\_FLT\_SL\_TYP\_CD |
|  | Inventory Status Code | IVNTRY\_STS\_CD |
|  | Model Year | VHCL\_YR\_NB |
|  | NCI Inventory Status Code | NCI\_IVNTRY\_STS\_CD |
|  | NCI Location Status Code | NCI\_LCTN\_STS\_CD |
|  | NCI Original Retail Type Code | NCI\_ORGNL\_RTL\_TYP\_CD |
|  | Plant Code | MNFCTG\_VHCL\_PLNT\_CD |
|  | Retail Sales Type Code | RTL\_SL\_TYP\_CD |
|  | Sold Flag | SOLD FLAG |
|  | Vehicle Country Code | CNTRY\_CD |
|  | Vehicle Global Market Code | GLBL\_MRKT\_CD |
|  | Vehicle Location Code | VHCL\_LCTN\_CD |
|  | Vehicle Production Source Code | VHCL\_PRDCTN\_SRC\_CD |
|  | Wholesale Action Type Code | WHLSL\_ACTN\_TYP\_CD |
| MODEL ATTRIBUTES | Body Style Name | BDY\_STYL\_NM |
|  | End Item Model Code | EIM\_CD |
|  | Exterior Color Code | EXTR\_CLR\_CD |
|  | Exterior Color Name | EXTR\_CLR\_NM |
|  | Factory Option Code | CRYPTC\_OPTN\_CD |
|  | Generic Option Code | GNRC\_OPTN\_CD |
|  | Model Line Code | VHCL\_LN\_CD |
|  | Model Line Name | VHCL\_LN\_NM |
|  | NCI Option Group Code | NCI\_OPTN\_GRP\_CD |
|  | NCI Sales Model Code | VHCL\_NCI\_MDL\_CD |
|  | NCI Vehicle Series Code | NCI\_VHCL\_SRS\_CD |
|  | NHTSA Vehicle Type Code | NHTSA\_VHCL\_TYP\_CD |
|  | NML Production Code | NML\_PRDCTN\_MDL\_CD |
|  | NNA Sales Model Code | VHCL\_NMC\_MDL\_CD |
|  | Telematics Indicator | TELEMATICS INDICATOR |
|  | Trim Level Description | TRM\_LVL\_DS |
|  | VEHICLE CHANNEL CODE |  |
|  | Vehicle Make Code | VHCL\_MK\_CD |
|  | Vehicle Model Description | VEH\_MDL\_DS |
|  | Vehicle Type Code | VHCL\_TYP\_CD |
| RANGE FILTERS | Factory Ship Date | FCTRY\_SHP\_DT |
|  | Lot Days | LOT DAYS |
|  | NCI Inventory Add Date | NCI\_IVNTRY\_ADDED\_DT |
|  | NMC Receipt Date | NMC\_RCPT\_DT |
|  | Original Retail Date | ORGNL\_RTL\_DT |
|  | Original Vehicle In-Service Date | ORGNL\_IN\_SVC\_DT |
|  | Original Wholesale Date | ORGNL\_WHLSL\_DT |
|  | Retail Date (Current) | RTL\_SL\_LSE\_DT |
|  | Vehicle In-Service Date (Current) | IN\_SVC\_DT |
|  | Vehicle Manufacture Date | MNFCTG\_DT |
|  | Vehicle MIS | VEHICLE MIS |
|  | Vehicle Wholesale Date (Current) | INVC\_DRFT\_DT |
| RETAIL DEALER INFORMATION | Original Retail City | TBD |
|  | Original Retail Dealer Country Code | ORGNL\_RTL\_DLR\_CNTRY\_CD (TBD) |
|  | Original Retail Dealer Number | ORGNL\_RTL\_DLR\_NB |
|  | Original Retail Phone Number | TBD |
|  | Original Retail State Code | ORGNL\_RTL\_ST\_CD |
|  | Original Retail Zip Code | TBD |
|  | Retail City | TBD |
|  | Retail Dealer Number | RTL\_DLR\_NB |
|  | Retail Phone Number | TBD |
|  | Retail State Code | TBD |
|  | Retail Zip Code | TBD |
| WHOLESALE DEALER INFORMATION | Original Whlsl Dealer Country Code | ORGNL\_WHSL\_DLR\_CNTRY\_CD (TBD) |
|  | Original Whlsl Dealer Number | ORGNL\_WHLSL\_DLR\_NB |
|  | Original Wholesale City | TBD |
|  | Original Wholesale Phone Number | TBD |
|  | Original Wholesale State Code | ORGNL\_RTL\_ST\_CD |
|  | Original Wholesale Zip Code | TBD |
|  | Wholesale City | TBD |
|  | Wholesale Dealer Number | WHLSL\_DLR\_NB |
|  | Wholesale Phone Number | TBD |
|  | Wholesale State Code | TBD |
|  | Wholesale Zip Code | TBD |

## Warranty Extracts

[The architecture overview diagram is a schematic diagram that represents the governing ideas and candidate building blocks of an IT system. It provides an overview of the main conceptual elements and relationships in IT architecture, which frequently include candidate subsystems, components, nodes, connections, users and external systems. The architecture overview diagram should be simple, brief, clear, and understandable. A System Context Diagram is also used to:

* Identify the processes the system must perform and the structuring or relationships between those processes.
* Identify the interfaces and data exchanges between the processes.

Conditions where this section must be completed:

* For Maintenance projects that use existing technology (mainframe, web, etc….) - System context diagram only
* For Maintenance projects that will introduce new technology - System context diagram and Software Architecture Document contents required up to and including phase 4 (which is a full Software Architecture Document)
* For new/upgrade COTS projects – System context diagram and Software Architecture Document contents required up to and including phase 2 only
* For new development projects - System context diagram and Software Architecture Document contents required up to and including phase 4 (which is a full Software Architecture Document)]

Application architecture information is stored in the System Architecture folder in Clarity

## Incident Rate Reporting

[A paragraph with High-Level testing goals for this project should be in this section including statements pertaining to testing functional requirements, and performing different test types (unit/system, integration/regression, load/performance & user acceptance); an example is below. Also include a table of test deliverables and estimated timing for each, as input into your project plan. This should describe the approach that will be used for testing, and who will need to be involved, and who will be responsible for what

For Infrastructure only projects, this should describe what testing will be required before anything is put into production and how changes put into production be validated.

For Agile projects, coding and testing are part of one process during the iteration. Write detailed tests for a story as soon as coding begins. Focus on completing one story at a time. Collaborate closely with programmers so that testing and coding are integrated. Keep the business in the loop throughout the iteration; let them review early and often. You should build your test cases in QC as you go and execute every test case per iteration make sure you include testing time for each of your user stories.

EXAMPLES:

Project XYZ will write, execute tests, track defects and report results against the functional requirements as they highlighted above as they are being coded by the project team utilizing NNA’s enterprise test case management system, HP Quality Center. Unit and system testing will be performed by the development team with system testing documented in Quality Center. Integration and High-Level regression testing will also be documented and tracked in Quality Center. User Acceptance Testing should occur at a minimum at the end of each release

Project XYZ will write, execute tests, track defects and report results against the functional requirements highlighted above utilizing NNA’s enterprise test case management system, HP Quality Center. Unit and system testing will be performed by the development team with system testing documented in Quality Center. Integration and High-Level regression testing will also be documented and tracked in Quality Center. User Acceptance Testing is unnecessary for this project as it is a change to back end / batch processes unrelated to end user interface screens.]

Visualiations

## AVES

AVES

## INSPECT

INSPECT

## Consumer Affairs

Consumer Affairs

## Techline

Techline

## Modification Log

| **Revision Date** | **Modified By** | **Description** |
| --- | --- | --- |
| 6/7/2013 | PMO - Douglas Foster | Initial Creation. |
| 1/6/2014 | PMO - Douglas Foster | SDLC CR 12: Removed Project Close section. Project close information to be captured in Project Management repository. |
| 6/24/2014 | PMO - Douglas Foster | Updated for Agile. Moved App Architecture, Data Architecture, Test Plan, and Training to Section 2 |

# Go/No Go (Execute Phase) R 1.1

## Detailed Data Architecture

[Each sprint will have a sprint grooming exercise that reviews the stories to be developed during the sprint as well as produces specifications, screen layouts, rules, etc.  The Logical Data Model for the sprint should be completed during this sprint grooming exercise. The Logical Data Model will follow all data modeling standards. The following will also be required for the sprint as needed based on the sprint goals:

* Data Profiling
* Data Access and Security Requirements
* Platform and Capacity information

The Physical Data Model and DDL needed for the sprint will be generated from the Logical Data Model. The DDL will be provided by a Nissan appointed Domain Data Architect and executed by the appropriate DBA.

Detailed data architecture information is stored in the Data Architecture folder in Clarity

## Detailed Application Architecture

[The architecture overview diagram is a schematic diagram that represents the governing ideas and candidate building blocks of an IT system. It provides an overview of the main conceptual elements and relationships in IT architecture, which frequently include candidate subsystems, components, nodes, connections, users and external systems. The architecture overview diagram should be simple, brief, clear, and understandable. A System Context Diagram is also used to:

* Identify the processes the system must perform and the structuring or relationships between those processes.
* Identify the interfaces and data exchanges between the processes.

Conditions where this section must be completed:

* For Maintenance projects that use existing technology (mainframe, web, etc….) - System context diagram only
* For Maintenance projects that will introduce new technology - System context diagram and Software Architecture Document contents required up to and including phase 4 (which is a full Software Architecture Document)
* For new/upgrade COTS projects – System context diagram and Software Architecture Document contents required up to and including phase 2 only
* For new development projects - System context diagram and Software Architecture Document contents required up to and including phase 4 (which is a full Software Architecture Document)]

Application architecture information is stored in the System Architecture folder in Clarity

## Detailed Test Plan

[A paragraph with High-Level testing goals for this project should be in this section including statements pertaining to testing functional requirements, and performing different test types (unit/system, integration/regression, load/performance & user acceptance); an example is below. Also include a table of test deliverables and estimated timing for each, as input into your project plan. This should describe the approach that will be used for testing, and who will need to be involved, and who will be responsible for what

For Infrastructure only projects, this should describe what testing will be required before anything is put into production and how changes put into production be validated.

For Agile projects, coding and testing are part of one process during the iteration. Write detailed tests for a story as soon as coding begins. Focus on completing one story at a time. Collaborate closely with programmers so that testing and coding are integrated. Keep the business in the loop throughout the iteration; let them review early and often. You should build your test cases in QC as you go and execute every test case per iteration make sure you include testing time for each of your user stories.

EXAMPLES:

Project XYZ will write, execute tests, track defects and report results against the functional requirements as they highlighted above as they are being coded by the project team utilizing NNA’s enterprise test case management system, HP Quality Center. Unit and system testing will be performed by the development team with system testing documented in Quality Center. Integration and High-Level regression testing will also be documented and tracked in Quality Center. User Acceptance Testing should occur at a minimum at the end of each release

Project XYZ will write, execute tests, track defects and report results against the functional requirements highlighted above utilizing NNA’s enterprise test case management system, HP Quality Center. Unit and system testing will be performed by the development team with system testing documented in Quality Center. Integration and High-Level regression testing will also be documented and tracked in Quality Center. User Acceptance Testing is unnecessary for this project as it is a change to back end / batch processes unrelated to end user interface screens.]

Testing information is stored in the Testing folder in Clarity

## Detailed Training & User Support

The below document is the user manual for the system. The entire user base will receive ad hoc training as they are all part of UAT.



[The purpose of this activity is to capture details to create design specifications for help text, tutorials, and other training and user support documentation for this release cycle and to decide what kind of user documentation, help text, on-line tutorials, and other support materials will be provided. The design specification identifies the responsibilities of the business customer and the project team.

The level of detail required around training and support could vary with each project. For example, it may be necessary to provide more detailed specifications on a project where users require extensive training in multiple formats than for a project where a screen is to be changed.

Please note that if the project involves new Application Development, you will need to include new Configuration Item for use with the Service Now.

Document any training that the project will expect the vendor to provide, and to whom. Document any user support that will be required during and after deployment.]

[Identify if training will be required for the project.

* Tool training require for project team
* Vendor ABC will provide technical and functional training to Nissan IS
* Vendor or Nissan will provide user training on the application

Identify user support needs such as

* Vendor or Nissan will provide user manuals
* Vendor XYZ is primary support for application ABC
* An incident resolver queue will be created during the Execution phase.]

## Test Results

[Identify who approved the system, integration and if applicable user acceptance test results. This section may include a High-Level assessment of details outlined in the NNA IS QA Report, which should be found in the document repository of NNA’s project tracking system (CA Clarity) for this project.

**For Infrastructure only projects**, testing/validation is typically done during the deployment phase. This section should indicate the test cases to be executed during and after deployment. Create and store the results in the 08 Testing folder in Clarity. During Phase 5/6 update the testing results using Clarity version control.]

Test results are stored in the Testing folder in Clarity

## Deployment Plan (Use on New Application/System only)

The team is using the below embedded spreadsheet to track Communication, Rollback, Deployment Verification, and Deployment Activities.



[See the Deployment Plan Guide for details.]

### Requested CMT (RFC for NMEX projects)

[CH0044745](https://nnanissan.service-now.com/nav_to.do?uri=change_request.do?sys_id=674e04f00f7da600b0ef37f692050e4b%26sysparm_view=emergencyChange)

## Production Support Check list

| Component | Comments / Description |  |
| --- | --- | --- |
| Production Support Team | Organize the support team for the first days of productive operation. This team comprises implementation project team members. This planning step starts the transition of the implementation team to a support role. There should be at least one support person for each major business process area. |  |
| Issue Escalation Procedure | Ensure an updated contact, owner and escalation process is outlined. |  |
| Disaster Recovery Plan & Procedures | Define and document procedures for disaster recovery, including roles and responsibilities and proper escalation procedures. This includes but is not limited to (1) Company-wide procedures for continuing business operations, (2) data procedures, (3) technical procedures for hardware, networks, etc. PRIORITY 2 Incident |  |
| Update System Context Diagram in MEGA | If changes are done to the System Context Diagram, update MEGA with the new System Context Diagram. |  |
| APO Audit | If an application is added, rewritten, or replaced, work with Pryor Manning to determine if any changes are required to the Application Portfolio mapping in MEGA. |  |
| Setup Help Desk | | |
| Organization makeup, numbers and types of resources required | <Where is this information located? (files and paths)>  Release 1.0 EQUIP TaskForce will be supported by the Project Team for first level triage of issues. Those issues will be logged into the AGILE JIRA platform. For RTB items like DIH Failures pending Raj sending the members of RTB team to have SNOW group created to support after go-live. |  |
| Roles and responsibilities | <List roles required and a briefly explain responsibilities. If this information is documented in a separate file, indicate location>  RTB teams will support their subject areas and the AGILE Project Team will support planned enhancement and bug / fixes. |  |
| Telephone scripts | <Mention where these scripts are documented> |  |
| Guidelines for prioritizing and categorizing issues | <Location of guidelines>  AGILE Project team will triage issues and determine path of resolution. |  |
| Documentation and communication to the company on how to use the Help Desk | <Who/when/how this information was>  Release 1.0 EQUIP TaskForce will be supported by the Project Team for first level triage of issues. Those issues will be logged into the AGILE JIRA platform. For RTB items like DIH Failures pending Raj sending the members of RTB team to have SNOW group created to support after go-live. |  |
| Training sessions for Help Desk staff | <Schedule of training sessions>  After Release 1.0 we will engage Help Desk formally and document the process at that time. |  |
| Procedures Documentation  (If existing Help Desk, have the procedures been updated?) | <The Help Desk must have procedures for processing the priorities of problems. The procedures must identify the type of resource to which the problem is assigned.>  Future Release |  |

## Vendor Support Information

[In support of JSOX requirements for a Go/No Go decision, complete this section to indicate how the change will be supported when it is put into production]

| Vendor | Contract Change Request | Purchase Order Number |
| --- | --- | --- |
| Tech Mahindra | CCR 2016-015 | 7400097741 |

## Go/No-Go Approval

[APPROVAL TABLE DIRECTIONS:

below is the table to record approvals. To record approvals, save the email message to your local drive then embed the emails message using Insert Object directly into table below.

When you have a subsequent update to the document that requires another approval, expand the rows and indicate the version number to which these approvals apply]

| Approver | Title | Embedded Email approval | Approvals Apply to Version |
| --- | --- | --- | --- |
| Jason Snap | Manager, Product Owner |  | 1.1 |
| Greg Horton | Manager, IS |  | 1.1 |

## Modification Log

| **Revision Date** | **Modified By** | **Description** |
| --- | --- | --- |
| 6/7/2013 | PMO - Douglas Foster | Initial Creation. |
| 1/6/2014 | PMO - Douglas Foster | SDLC CR 12: Removed Project Close section. Project close information to be captured in Project Management repository. |
| 6/24/2014 | PMO - Douglas Foster | Updated for Agile. Moved App Architecture, Data Architecture, Test Plan, and Training to Section 2 |