Logo, company name

Description automatically generated

Solution Architecture Definition Template

<< Project Name >>

<< Author >>

# **Purpose of this document**

|  |  |
| --- | --- |
|  | * To identify any business processes that will need to change * To describe any new or modified processes proposed as part of the solution * To assess the impacts of the project outcome on the business that need to be managed. Considering for example new or changed: * Business process * Business organisation * Culture or behaviour * Resourcing * To describe the strategy for deploying the final solution and/or any increments of it, from a business perspective * To describe the strategy for educating and supporting those impacted by the solution and any associated business change. Consider: * Training for business and technical participants in the solution (e.g. those who will use the products created) and those who will support the solution in live use (operations and support staff) * What documentation may be required to support the operation of the solution and its on-going maintenance and enhancement * *To provide a common understanding of the technical architectures to be used during development and deployment of the solution including:* * *Hardware/Infrastructure* * *Software Architecture* * *To describe the target environment(s) for the solution and (if different) the development environments.* * *To provide an outline description of anticipated solution in areas such as:* * *Hardware (i.e. the infrastructure, processing, storage, networking etc.) for both development and deployment* * *Software (i.e. the major software objects or components - both process and data - and their interactions)* * *Information Security (e.g. access policy, access control etc.)* |

|  |  |  |  |
| --- | --- | --- | --- |
|  | **Project Role** | **Name** | **Signature & Date** |
| **Produced by:** | Business Analyst |  |  |
|  | Technical Coordinator |  |  |
| **Approved by:** | Business Visionary |  |  |
|  | Project Manager |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Revision History** | | | | |
| Name | Ver | Reason for change | Status | Date |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

# Table of Contents

Purpose of this document 1

Table of Contents 2

Business Impact Assessment 3

Business Process Impacts 3

Process Context 3

Technology context 3

Operational context 3

Process Scenarios 3

External Impacts 3

Internal Business Organisation & Resourcing Impacts 4

Business Culture Impacts 4

Business Implementation Strategy 5

Change to Business Organisation Structure 5

Change to External Business Interfaces 5

Staff / User Training 5

System Architecture 6

Architecture Overview 6

Hardware Components and Relationships 6

Service Components and Relationships 6

Non-Functional Requirements 6

System Life Expectance & Maintenance Strategy 8

Technical Environments 8

Development Platforms 8

Target Platform 8

# Business Impact Assessment

## Business Process Impacts

### Process Context

Using diagrams, describe at a high level all business processes impacted by this project. This should not be restricted to the processes covered by the proposed system (if any) but should describe the business process in its entirety. Consider defining the business information ‘inputs’ and ‘outputs’ of each process as well as identifying information that is stored. A very high level description of how each process step manipulates this information could also be considered.

### Technology context

Describe in more detail the parts of the process, identified above, that will be supported by technology or where current technology will need to change.

### Operational context

To help with decision making around target solution Service Level Agreements, describe who will operate the automated processes (the business role e.g. Customer Services Agent, Underwriter), from what geographic locations (e.g. Head Office, Mumbai call centre), and in what numbers. Categorise how they will use the automated process and when they will use it.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **User Class *(Business Role)*** | **Geographic Location(s) (or Mobile)** | **Approx no. of users** | **Use of the system  *(e.g. heavy user of basic functions, occasional user of management info etc.)*** | **Usage Window  *(e.g. 8am-6pm Mon-Fri) (state timezone where relevant)*** |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

### Process Scenarios

Provide a list of Business Events that will trigger the use of the system (e.g. telephone enquiry received, customer misses a payment, financial year-end). Where possible prioritise these. This does not need to be an exhaustive list but needs to cover around 80% of the events. Approximate prioritisation is fine at Feasibility with more certainty expected by the end of Foundations.

For each business event you should record :

|  |  |  |  |
| --- | --- | --- | --- |
| **Priority** | **Event Name** | **Description of Event** | **User Class(es) impacted** |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

### External Impacts

Describe the impacts on third parties. This should include customers, suppliers and service partners who will interact differently with the organisation as a result of changes to business processes. For large organisations with autonomous business functions also consider how other parts of the business will be impacted.

## Internal Business Organisation & Resourcing Impacts

Recommend a business organisation structure and required numbers of staff that will best suit the new business processes. Contrast this with the current structure and staff numbers being explicit about new, changed or redundant roles. Note: this should be focussed on roles (job functions) not individuals. Transitioning the organisation from current to future state will be dealt with as part of the Business Implementation Strategy section below.

## Business Culture Impacts

Where there is a need to introduce, change or re-enforce particular cultural aspects of the business describe current and future states clearly justifying the change. The business roles will need to ensure that the ground is prepared well in advance of solution delivery, if possible, to ensure its smooth deployment,

# Business Implementation Strategy

Under the heading below, describe the strategy for transitioning the organisation from its current state to the state encapsulated by the Business Vision section of the Business Case.

## 

## Change to Business Organisation Structure

Describe strategies for

* Hiring new staff
* Retraining existing staff
* Redeploying or making surplus staff redundant

This may include complex and/or sensitive negotiations with HR, individuals and/or trades unions and will typically be the responsibility of the Business Visionary (actively supported by the Business Sponsor, where necessary) to ensure this is handled effectively

## Change to External Business Interfaces

Describe strategies for transitioning to a new way of interacting with

* Customers of the organisation
* Suppliers to the organisation
* Service partners to the organisation
* Other parties such as regulators

This may involve complex commercial and legal negotiations

## Staff / User Training

Describe strategies for training staff/users in the operation of the new business processes

* Outline the anticipated need for and extent of training required
* Consider all impacted roles
* Consider all types of training from how to operate a new system to the wider cultural changes required
* Education of End-users of processes and systems who are not part of the business organisation e.g.
* Customers
* Suppliers
* Service Partners

# System Architecture

## Architecture Overview

Construct a high level model (diagram or series of diagrams) that clearly identifies all the hardware components of the architecture and illustrates how the software application is overlaid on the chosen hardware. Verify that it is complete in terms of both hardware (identifying all technical platforms and relationships between them) and software (identifying all interfacing systems, all application software components and appropriate middleware and database software etc.).

## Hardware Components and Relationships

By reference to the Architecture Overview above, describe in as much detail as is available:-

* Each of the hardware components of the system that are illustrated in the previous diagram (i.e. describe the ‘entities’ on the diagram – the boxes or other shapes)
* The physical interfaces between them under predicted implementation scenarios e.g. office access and home access (i.e .interfaces are the lines on the diagram)

Briefly explain how the proposed architecture fits with the organisation’s strategic architecture (if that exists).If there are several choices with recommended architectures (e.g. databases) state the rationale for the proposed choice. Identify risks and issues related to deviating from current technical architecture or policy.

**Note:** If the purchase of new hardware or the installation of new infrastructure is required to support the deployment of the proposed system, it needs to be described in sufficient detail to provide likely hardware specifications and associated costs

## Service Components and Relationships

By reference to the Architecture Overview above, describe at a high level:-

* The major software components of the system in terms of their purpose (e.g. application server, data server, web server, transaction manager etc)
* How these components relate to each other in terms of software. Describe the involvement of any middleware in the architecture. Describe the interfaces between the software components under predicted implementation scenarios e.g. office access and home access
* How the proposed system interfaces with existing applications and any others currently under development. Consider systems internal and external to the organisation.

Briefly explain how the proposed architecture fits with the organisation’s strategic architecture (if that exists). Identify risks and issues related to deviating from current technical policy with regard to middleware etc

## Non-Functional Requirements

Describe the key areas of non-functional requirements that are likely to be important to the project and how the architecture described above supports the non-functional requirements. If possible prioritise these on a scale of 1 to 3 as indicated in the table.

**Note**: At Feasibility or even Foundations it may not be possible to describe the non-functional requirements in the detail indicated in the comment text within the table. Don't worry too much about this but be sure to make a note in the table that further investigation of this requirement is needed during the Foundatons phase. Where investigation work associated with non-functional requirements is required during Evolutionary Development, ensure this is appropriately defined and scheduled.

|  |  |  |
| --- | --- | --- |
| Non-functional requirement | Priority 1=critical 2=important 3=unimportant | How this is supported by the architecture |
| Usability |  | If there are any specific usability drivers for the system describe the required design interventions here. The need for minimal user training, use by the visually impaired, operation by inexpert or ‘computer illiterate’ users etc. are example drivers for specific usability requirements |
| Performance |  | What are the expected user response times for on-line/real time systems, run time constraints for background/supporting processes (e.g. must complete between 1 am & 3am) etc. and how does the architecture support these requirements especially at peak usage times? |
| Capacity |  | What are the likely number of simultaneous users (by channel where appropriate) and how much data is likely to be flowing between hardware components. How does the architecture deal with this especially at peak usage times? |
| Scalability |  | Over time, it may be the case that numbers of users, data volumes etc. will increase. There is more than one way deal with this. The first is to build the solution to exceed **current** requirements and meet possible future requirements. The alternative is to create a smaller system now that can be **scaled up** to deal with increased capacity and performance demands. If a scalable solution is most sensible, describe how the architecture will accommodate the increases as and when required. |
| Security |  | Describe any security requirements and how these are accommodated by the architecture |
| Availability & peak usage |  | Describe normal hours of operation including anticipated peaks and troughs in system usage over explicit periods of a day, a week, a month or a year. E.g. peak usage 8-9am, 12-2pm or Sunday night or last working day of the month or tax year-end. If appropriate, describe any appropriate provision within the architecture to deal with peak usage |
| Resilience & recovery |  | Based around the criticality of availability (i.e. how long the business can be without the system, or parts of it, without suffering loss or damage) describe how the architecture meets this need. Describe how recovery after failure will be handled and how this is accommodated by the architecture. |
| Disaster Recovery |  | Describe how the architecture provides for disaster recovery. E.g. Does the solution require cross-site mirroring of data? If so, does the mirror copy need to be up to the minute, hour, 4 hour etc. or is last night’s off-site backup sufficient in a disaster recovery scenario? |

## System Life Expectance & Maintenance Strategy

There is little point in rigorously engineering a system that will be used for three months and then thrown away but it is also very unwise to risk providing a poorly engineered system that is expected to be part of the business, long term. As the development time and cost can vary greatly depending on the rigour of the engineering, it is important to understand whether the system is a long term strategic solution or a quick and temporary fix before baselining these project parameters.

There are three basic options to consider when developing the solution that will be driven by the needs of the business:-

1. To build a relatively quick and potentially poor quality solution and throw it away after limited short term use. This will be cheap to build but very expensive to maintain during its lifespan
2. To build a relatively quick and potentially poor quality solution and then engineer it for quality immediately after initial implementation. This will provide a solution more rapidly if that is what is needed but will be expensive to support if the re-engineering part of the project is delayed or cancelled. In this instance, approval for the project MUST cover both the initial build and re-engineering costs
3. To build a fully engineered solution from the outset. If time is available this is the most efficient way of building the system and will almost certainly be cheaper than option 2 (above)

Describe the system life expectancy and maintenance strategy below and select the most appropriate development approach from the list above. If options 1 or 2 have been chosen, make a case for this recommendation and clearly identify any risks associated with choice

# Technical Environments

## Development Platforms

Identify the software tools that will be used during development beyond the standard desktop environment (Software tools include compilers, modelling tools, configuration management tools, testing tools, etc.)

Define the different environments that will be used by or created for the project e.g. development, testing, pre-production environments

Note: The description of the development platforms should provide the project with sufficient information to identify:

* the technical skills needed by the developers
* purchases needed to be made for development activity to take place

## Target Platform

Identify the hardware and software that will need to be in place for the system to be delivered

Identify the tools needed for staff performing support and maintenance activities

Note: The description of the target platform should enable planning later within the project for migration and cutover