**Architectural Review Checklist**

SPORE Project Relationship and Clinical Intervention Portal

1.2

Revision Document History

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Version | Description | Author |
| 06/17/08 | 1.0 | First draft | Eric Kascic, Makiko Duncan |
|  |  |  |  |
|  |  |  |  |

Table of Contents

1. Introduction 6

Purpose 6

Guidelines for Completing the ARC 6

2. Project Details (To be answered by development team) 6

General Information 6

Project Description 6

Contact Information 6

Major Deployment Milestones 6

Architectural Details 7

High level architectural description: 7

High Level Design Diagram (If available): 7

Implementation language(s) used? 7

Will connections/requests to the application be session based (i.e. statefull versus stateless)? If so, is there any reason why application would not support “sticky session” load balancing? 7

Will the application be caching data? If so, what is being cached and how much data will be cached? 7

What data files will be created (if any)? How much data will be saved on an on going basis? 7

Will this application need a database schema(s) created on the NCICB infrastructure? If so, what is the maximum number of objects to be stored? 7

Are there any external/non-NCICB data sources that will be accessed by the application? 7

If this is a web-based application, what are the preferred virtual hosts names to be registered? 7

Any additional architectural details that may be of significance to the needed deployment environment. 7

Performance Requirements 8

Total number of users for this application? 8

Peak number of concurrent users? 8

Peak number of requests/minute? 8

Up time requirements? 8

Acceptable down time when recovering from major systems disaster? 8

NCICB Project Dependencies 8

Configuration Management Details 8

Version Control 8

Change Control 8

Migration to CVS 8

Users 8

Build Process 8

Other CM Needs 8

Additional Notes 8

3. System Requirements (To be answered by both development & systems team) 9

Operating System 9

<<Solaris/LINUX/Windows 2000/…>> 9

Software (Technology Stack) 9

Web Server: <<Apache/Tomcat/ZOPE/…>> 9

App Server: <<Tomcat/JBoss/Oracle 9iAS/…>> 9

Database Server: <<Oracle/MySQL/PostgreSQL/…>> 9

Other software components: <<PERL x.x, Python x.x, …>> 9

Server Hardware 9

Server: <<type, model, …>> 9

Minimum processor speed: 9

Minimum memory: 9

Minimum local drive space: 9

Storage 9

Expected file server disk storage (in MB): 9

Expected database storage (in MB): 9

Expected ftp storage (in MB): 9

Expected media/image storage (in MB): 9

Load Balancing/Fault Tolerance 9

Does the application support load balancing? 9

Implement load balancing – YES/NO 9

Networking 9

Any application specific port assignments? 9

Any additional configuration? 9

Additional Notes 9

4. Proposed NCICB Deployment Environment (To be answered by systems team) 9

Hardware 9

Technology Stack 10

File Server 10

Database 10

Networking 10

Other Resources 10

5. Impact Assessment (To be answered by systems team) 10

Overview 10

Cost 10

Timeline to implement 10

Additional Notes 10

6. Acceptance 10

7. Appendix 1 - Future Systems Requirements 11

New Architecture Diagram 11

Notes on the design change 11

Hardware 11

Servers 11

Processor speed 11

Memory 11

Drive Space 11

Software (Technology Stack) 11

Web Server 11

App Server 11

Other components 11

Fault Tolerance (Redundancy) 11

Does the application support load balancing? 11

Implement load balancing – YES/NO 11

Load balancing requirements 11

Fault tolerance solution suggested 11

Additional requirements (if any) 11

Performance Enhancements 11

Processing Power 11

Memory 11

I/O 11

Other needs 11

# Introduction

## Purpose

The purpose of the Architectural Review Checklist (ARC) is to help identify the deployment environment (both hardware and software components) necessary for an application to execute optimally within the NCICB IT infrastructure. The ARC is considered to be a dynamic artifact for any NCICB project and should be maintained in the configuration library (i.e. CVS) for each project. For new projects, this document template will be automatically checked into the projects repository by the SCM team upon project initiation. The ARC should be reviewed on at least a quarterly basis to capture any application design changes that may effect the deployment environment necessary for optimal performance.

## Guidelines for Completing the ARC

The ARC is divided into 4 main sections. The first section (Project Details) is meant to capture general information about the project. The project development team should fill out the Project Details section and return it to the systems team. The second section (Systems Requirements) is meant to capture details of specific system requirements for the project. Both the development and system teams will preferably answer the Systems Requirement section jointly. However, depending on the development team’s familiarity with the NCICB infrastructure, they may choose to answer either all or parts of this section on their own. The third section (Planned Deployment Environment) specifies the proposed deployment environment for the application and is to be answered by the systems team based on the response to section 1 & 2. Finally, the last section (Impact Assessment) describes the impact (additional hardware/software costs, staffing resources…) to the existing NCICB IT infrastructure in order to support this application. Upon completion of the Deployment Environment and Impact Assessment sections, the systems team will return the ARC to the development team for final review and comments.

# Project Details *(To be answered by development team)*

## General Information

### Project Description

The mission of PRCIP is to provide the OSB staff with the ability to better manage P50 SPORE grant applications. This includes accessing the grant applications and searching them in a convenient way.

### Contact Information

|  |  |  |  |
| --- | --- | --- | --- |
| **Title** | **Name** | **Phone** | **Email** |
| Project Manager | Maki Duncan | 301.443.5307 | duncanmk@mail.nih.gov |
| Team Lead/Architect | Eric Kascic | 301.580.0891 | kascice@mail.nih.gov |
| SCM Coordinator |  |  |  |

### Major Deployment Milestones

|  |  |
| --- | --- |
| **Milestone** | **Date** |
| Planned Release to QA | N/A |
| Planned Release to Staging | N/A |
| Planned Release to Production | N/A |

## Architectural Details

### High level architectural description:

There is a collection of grant application PDFs on the file system that need to be served up. These PDFs are indexed by dtSearch for searching. The dtSearch search engine includes a front-end web application that integrates with IIS. The end users interact with the web application to do searches. The search results will refer to grant application PDF documents that IIS will serve up to the client browser.

### High Level Design Diagram (If available):

*N/A*

### Implementation language(s) used?

*N/A*

### Will connections/requests to the application be session based (i.e. statefull versus stateless)? If so, is there any reason why application would not support “sticky session” load balancing?

Stateless

### Will the application be caching data? If so, what is being cached and how much data will be cached?

*N/A*

### What data files will be created (if any)? How much data will be saved on an on going basis?

The size of grant applications can vary, but an average size to use is 60MB. There are roughly 75 grants per year. 75\*60MB = 4.5G

Per grant application there are other documents associated with it per year. These documents are usually not larger than 20MB each, and there are roughly 7 per year. (Progress Reports, etc). 75\*7\*20MB = 10.5G

A dtSearch index takes about 1/3 the size of the original documents.

Therefore, up-to/about 20G of data per year might be generated.

We also plan to download last five years data initially then, update the documents quarterly in the future.

### Will this application need a database schema(s) created on the NCICB infrastructure? If so, what is the maximum number of objects to be stored?

*N/A*

### Are there any external/non-NCICB data sources that will be accessed by the application?

*N/A*

### If this is a web-based application, what are the preferred virtual hosts names to be registered?

spore-dtsearch.nci.nih.gov

### Any additional architectural details that may be of significance to the needed deployment environment.

This web application is for internal use only by OSB staff. It should not be accessible to the “outside world”.

## Performance Requirements

### Total number of users for this application? 10

### Peak number of concurrent users? 10

### Peak number of requests/minute? 10

### Up time requirements? Not mission-critical

### Acceptable down time when recovering from major systems disaster? Not mission-critical

## NCICB Project Dependencies

N/A

## Configuration Management Details

Briefly describe your current configuration management practices here.

### Version Control

What version control software are you currently using, if any?

PRCIP has a project within CVS.

### Change Control

What are your current change control practices? What procedures are in place to determine whether to implement a change request?

Requirements are fluid for this system. As the customer has an idea, the PM and technical lead decide whether to do it and then we do it.

### Migration to CVS

Indicate whether you will require SCM support migrating your source code to the NCICB CVS repository.

N/A

### Users

Provide a list of all developers who require access to your repository modules.

N/A

### Build Process

Describe your build process here. For example, are you using Ant, Make, or something else?

The web application is an “off the shelf” component.

### Other CM Needs

Describe any other CM needs?

## Additional Notes

# System Requirements *(To be answered by both development & systems team)*

## Operating System

### Windows Server 2kX

## Software (Technology Stack)

### Web Server: IIS Server

### App Server: N/A

### Database Server: N/A

### Other software components: dtSearch Web

## Server Hardware

### Server: Doesn’t matter as long as it runs Windows Server 2kX and IIS

### Minimum processor speed:The faster the better, but no hard requirements

### Minimum memory:2GB

### Minimum local drive space: 5years of grants \* 20GB/yr = 100G

## Storage

### Expected file server disk storage (in MB):See [Architectural Details](#_What_data_files) and [Additional Notes](#_Additional_Notes)

### Expected database storage (in MB): N/A

### Expected ftp storage (in MB): N/A

### Expected media/image storage (in MB): N/A

## Load Balancing/Fault Tolerance

### Does the application support load balancing?

### Implement load balancing –NO

## Networking

### Any application specific port assignments? NO

### Any additional configuration? NO

## Additional Notes

For best performance, the drive should:

* NOT be an external drive such as Firewire, USB, or NAS.
* NOT be a compressed or encrypted NTFS folder.
* If possible, disable on-access virus scanning of the folder containing the index.
* Be defragmented fairly regularly

# Proposed NCICB Deployment Environment *(To be answered by systems team)*

## Hardware

*<<dev, qa, staging, production servers to be used>>*

## Technology Stack

*<<specific technology stack to be used>>*

## File Server

*<<space allocation on big IP, NFS mount points, initial size allocation…>>*

## Database

*<<database server to used, schema names to be created, initial size, maximum size…>>*

## Networking

*<<e.g. any BigIP configuration necessary, …>>*

## Other Resources

*<<e.g. ftp server access, media server access, …>>*

# Impact Assessment *(To be answered by systems team)*

## Overview

## Cost

## Timeline to implement

## Additional Notes

# Acceptance

|  |  |
| --- | --- |
| Project Lead ( ) | Project Coordinator – NCICB ( ) |
| Systems Team | SCM Administrator |

Appendix 1 - Future Systems Requirements

In this section, describe any projected future anticipated requirements for your system.

## N/A