Requisitos Técnicos Mínimos para a Aquisição de Soluções Fornecidas por Terceiros
Versão 1 (18/11/2022)

1 APRESENTAÇÃO

O presente documento elenca os requisitos técnicos mínimos para soluções de software que venham a ser adquiridos por órgãos da Administração Direta ou Indireta, fornecidos por terceiros, isto é, por empresas ou outro tipo de entidade produtora de sistemas que não possuam relação com a CODATA – Companhia de Processamento de Dados da Paraíba, visando salvaguardar um funcionamento saudável e duradouro das referidas soluções, inclusive, após o encerramento de garantias contratadas, possibilitando a extensão e integração de tais sistemas com outras aplicações corporativas do Estado da Paraíba, bem como a sustentação destas pela CODATA.

2 DEFINIÇÕES

Esta sessão elenca conceitos utilizados ao longo deste documento, buscando minimizar possíveis dúvidas acerca do texto.

2.1 REQUISITO MANDATÓRIO

Trata-se de uma característica, funcionalidade ou condição essencial para a aquisição de uma solução que trata o presente documento.

2.2 REQUISITO RECOMENDADO

Característica, funcionalidade ou condição muito desejável, ou seja, que pode impactar positivamente a solução, de modo complementar, não comprometendo o funcionamento básico, caso não seja atendido.

2.3 REPOSITÓRIO

Espaço lógico ofertado pela CODATA, semelhante a um diretório, com características de segurança, controles de versão e de acesso, destinado a acomodação apropriada de artefatos de software e ou sua documentação (manuais, etc).

2.4 CÓDIGO FONTE

Conjunto de arquivos contendo as instruções completas, para a construção/compilação e consequente execução de um programa de computador.

2.5 AMBIENTE DE PRODUÇÃO

Recursos de hardware e software que correspondem a instâncias do sistema destinadas aos usuários finais, ou seja, trata-se do ambiente onde a operação de fato ocorre, consumindo e ou produzindo dados válidos.

2.6 REGRAS DE NEGÓCIO

Definições de funcionalidades que não são relacionadas aos requisitos técnicos de um sistema, mas sim ao modelo negocial ou *modus operandi* de uma organização ou divisão de uma organização.

Tais regras podem ser diagramadas em notação BPMN (*Business Process Model and Notation)*.

2.7 DOCUMENTO DE VISÃO

Documento que reúne todas as características de um sistema. Dentre as quais, escopo, atores, regras de negócio, requisitos funcionais e não-funcionais e integrações. Tais informações podem estar representadas graficamente, utilizando notações UML (*Unified Modeling Language*), BPMN (*Business Process Model and Notation*), MER (Modelo Entidade-Relacionamento), etc.

2.8 SUSTENTAÇÃO DE SISTEMA

Processos e ou processamentos para garantir o funcionamento de um sistema ao longo do tempo. Incluindo correção de *bugs*, gestão corretiva e adaptativa.

2.9 PLANO DE IMPLANTAÇÃO

Documento que define cronogramas, responsáveis, tecnologias e recursos (a exemplo de scripts de inicialização e configuração de serviços de autenticação e segurança) para colocar o sistema em funcionamento, abrangendo desde a instalação dos ambientes até o treinamento dos usuários.

3 REQUISITOS TÉCNICOS MÍNIMOS

Este documento busca orientar quanto aos requisitos técnicos mínimos. Para melhor entendimento, classificamos os requisitos abaixo em dois tipos:

- Recomendado
- Mandatório

3.1 AQUISIÇÃO DE SOFTWARE COM A CESSÃO DOS CÓDIGOS FONTES (RECOMENDADO)

A CODATA recomenda que aquisições de soluções baseadas em programas de computador, por órgãos da Administração Direta ou Indireta, incluam a cessão dos códigos fontes, notadamente quando se tratarem de soluções verticais, isto é, soluções especialistas destinadas aos processos finalísticos do referido órgão ADQUIRENTE.

3.2 RECEBIMENTO E ACONDICIONAMENTO DOS CÓDIGOS FONTES DA APLICAÇÃO (MANDATÓRIO)

A entrega dos códigos fontes necessariamente deverá ocorrer através do repositório oficial do Estado da Paraíba (https://gitcodata.pb.gov.br), visando garantir a integridade e integralidade do objeto da aquisição, ou seja, do software fornecido.

O repositório supracitado é capaz de identificar em detalhes os arquivos de programas entregues, registrando o autor ou responsável, a data e hora da entrega e posteriores modificações em cada um dos arquivos que compõem os códigos fontes da aplicação e demais artefatos que componham a solução.

Para garantir que a versão submetida ao repositório, dos códigos fontes, corresponda a mais atual homologada pelo órgão adquirente do sistema, um colaborador da CODATA, ou alguém sob a sua supervisão, deverá ser encarregado da atribuição de empacotar[1] os referidos códigos, bem como a consequente publicação no ambiente de produção.

As atualizações de versões de sistemas em ambiente de produção serão restritas aos profissionais citados no parágrafo anterior. Fornecedores de soluções não terão acesso ao ambiente de produção para este fim.

3.3 QUANTO AO SGBD – SISTEMA GERENCIADOR DE BANCO DE DADOS (RECOMENDADO)

Recomenda-se a utilização de SGBD de código aberto, preferencialmente o PostgreSQL, o qual a equipe de DBAs da CODATA possui maior expertise.

Caso a solução de software a ser adquirida utilize SGBD diferente do supracitado, deverá constar em TERMO DE REFERÊNCIA da aquisição:

- O responsável pela compra das licenças de uso, caso se aplique;
- Descrição das limitações das licenças;
- Quantidade de núcleos de processamento cobertas pela licença;
- Volume de dados ou memória suportada.

Em via de regra, a compra das licenças de SGBDs proprietários é de responsabilidade do ADQUIRENTE da solução.

1 Empacotar, neste contexto, refere-se as atividades necessárias para a construção de um programa executável, a partir de seus códigos fontes.

É fortemente desaconselhado a utilização de SGBDs que não sejam padrão de mercado, isto é, pouco conhecidos, logo, com baixa confiabilidade e documentação escassa.

A CODATA poderá se recusar a oferecer qualquer tipo de suporte a SGBDs diferentes daqueles recomendados pela própria CODATA.

Nos casos em que uma aplicação utilize um SGBD não recomendado, a CODATA ofertando algum suporte a pedido do cliente, estará isenta de quaisquer sinistros.

3.4 DOCUMENTAÇÃO DO BANCO DE DADOS (MANDATÓRIO)

É mandatório o fornecimento de documentação do banco de dados como artefato da solução, seja ela adquirida com a cessão dos códigos fontes ou não.

Entende-se como documentação do banco de dados os seguintes artefatos, mas não limitados a estes:

- Diagrama de Entidade-Relacionamento; e
- Dicionário de Dados e Metadados;

Recomenda-se que os artefatos acima sejam entregues em formato aberto, isto é, que não necessite de uma ferramenta proprietária para sua abertura ou utilização, a exemplo de arquivos PDF, DOCX, PNG, JPEG, etc.

3.5 LEI GERAL DE PROTEÇÃO DE DADOS – LGPD (MANDATÓRIO)

É mandatória a utilização de recursos de anonimização e ou criptografia de dados pessoais que sejam gravados no banco de dados, conforme define a legislação em vigor, notadamente a LGPD.

3.6 RESTRIÇÃO DE ACESSO AO SGBD DE PRODUÇÃO (MANDATÓRIO)

Aplicações que se conectem a bancos de dados em ambiente de produção deverão fazê-lo, necessariamente, utilizando credenciais com acesso restrito, isto é, sem permissões de *root* ou administrador, tampouco permissões para execução de comandos de *Data Definition Language* (DDL).

Credenciais de bancos de dados utilizados pelas aplicações, notadamente aquelas que executem em ambiente de produção, deverão possuir apenas as permissões estritamente necessárias ao funcionamento da respectiva aplicação.

3.7 GERENCIAMENTO E DISTRIBUIÇÃO DE SENHAS PARA ACESSO A DADOS (MANDATÓRIO)

A elaboração de senhas deve seguir os padrões estabelecidos pela Política Geral de Segurança da Informação – PGSI[2] da CODATA.

Armazenamento de senhas em código-fonte é considerada uma falha grave de segurança e não deve ser utilizado. Ao armazenar senhas utilizar um algoritmo de *hash* seguro e *salt*, ou outra estratégia que não envolva gravar a senha em texto plano.

É incentivado o uso de variáveis de ambiente para a acomodação das senhas ou outras credenciais que as aplicações utilizem para conectar a outros serviços, a exemplo de SGBDs.

Dados de usuários, tais como informações pessoais, notadamente aqueles que são utilizados em alguma estratégia para recuperação de senha, inclusive os sistemas que cada usuário acessa, devem ser armazenados de forma segura, visando bloquear algum tipo de ataque ou engenharia social.

Cada ambiente – desenvolvimento, teste, homologação e produção – deverá ser acessível com credenciais distintas, ou seja, não se deve utilizar a mesma senha em ambientes diversos.

3.8 QUANTO A VERSÃO DA LINGUAGEM DE PROGRAMAÇÃO UTILIZADA (RECOMENDADO)

É fortemente recomendado exigir que a construção do sistema faça uso da versão estável mais atual da linguagem de programação e *frameworks* empregados, para evitar falhas de segurança, *bugs*, vulnerabilidades e outros mal funcionamentos ou comportamentos inesperados.

2 https://codata.pb.gov.br/midias/politica-geral-de-seguranca-da-informacao-2013-pgsi

3.9 CANAL DE COMUNICAÇÃO CRIPTOGRAFADO (MANDATÓRIO)

Aplicações acessíveis pela Internet, sejam elas WEB ou mesmo APIs (Rest, SOAP, GraphQL, etc) deverão estar disponibilizadas por meio de um canal seguro, isto é, protegido por camada SSL (Secure Sockets Layer), padrão global de segurança que permite a comunicação criptografada entre um navegador da Internet e um servidor WEB, ou outra tecnologia homologada pela Diretoria de Tecnologia da Informação e Comunicação da CODATA.

3.10 CAMADA DE ACESSO AO BANCO DE DADOS (MANDATÓRIO)

É fortemente recomendado que a solução de software a ser adquirida, em sua camada de acesso ao banco de dados, faça uso de ferramentas que previnam ataques do tipo *SQL Injection*, tais como *frameworks* de mapeamento objeto relacional (ORM).

Em hipótese alguma deve-se utilizar estratégia de “concatenação de strings” para produzir sentenças SQL, isto é, justaposição de parâmetros textuais de origem não-segura, como parâmetros preenchidos pelo usuário em formulários HTML ou mesmo armazenados no banco de dados com fragmentos de comandos SQL e posterior execução no SGBD.

3.11 PRODUÇÃO DE EVIDÊNCIAS DE ACESSO – LOGS (MANDATÓRIO)

Para efeito de auditoria, bem como o monitoramento da saúde da aplicação, é exigido que qualquer aplicação a ser adquirida registre eventos – logs – do tipo:

- Acessos a determinadas telas ou seções do sistema;
- Acesso a informações com alguma restrição, como documentos sigilosos, dados pessoais e outros que sejam classificados (conforme a LAI – Lei de Acesso à Informação);
- Operações de inclusão, alteração ou exclusão de registros no banco de dados;
- Alteração de perfil de acesso;

- Execução de jobs e tarefas automatizadas, inclusive o resultado do processamento.

É recomendado que os registros que tratam esta sessão contenham os seguintes dados:

a) Data e hora;
b) Usuário que efetuou a operação;
c) Endereço IP;
d) Identificador da sessão do usuário, quando aplicável (cookie, por exemplo);
e) Página do sistema de onde a operação foi realizada;
f) Identificador da instância (para sistemas clusterizados); e o
g) Resultado da operação – falha, sucesso, cancelada, etc.

3.12 TESTES DE ACEITAÇÃO OU PROVA DE CONCEITO (MANDATÓRIO)

É fundamental que o órgão ADQUIRENTE proceda com atividades de validação e homologação da regra de negócio implementada pelo sistema em aquisição, a exemplo da Prova de Conceito (PoC), para verificar sua aderência ao problema que se deseja solucionar.

Tais testes de aceitação deverão considerar tanto os requisitos funcionais, isto é, as funcionalidades referentes às regras de negócio, quanto os requisitos não-funcionais, principalmente, segurança e tempo de resposta da aplicação.

3.13 INFRAESTRUTURA (RECOMENDADO)

É fortemente recomendado que o órgão ADQUIRENTE opte por soluções preparadas para executar sob ambiente de nuvem – *cloud ready* – contendo artefatos como **DockerFile** ou **docker-compose.yml**, a fim de garantir uma maior compatibilidade da respectiva solução com a estrutura computacional da CODATA.

3.14 SINGLE SIGN-ON (RECOMENDADO)

É recomendado que a aplicação se integre, de forma nativa, com a plataforma de Login Único oficial do Estado da Paraíba, viabilizando o controle centralizado de acesso aos diversos sistemas e plataformas do governo.

3.15 CAMADA DE INTEGRAÇÃO (MANDATÓRIO)

Tão importante quanto o banco de dados da aplicação é a capacidade desta se integrar com outros sistemas. Para isso, faz-se necessário a existência de APIs, ou seja, de uma camada de integração.

Tal camada possibilita inclusive a extensão de funcionalidades de um sistema legado, ampliando a vida útil de uma aplicação.

Aplicações que venham a ser adquiridas, notadamente aquelas direcionadas ao suporte de serviços públicos digitais, devem ser dotadas de camada de integração, principalmente para que os referidos serviços possam ser disponibilizados através do Portal de Serviços do Governo Digital.

3.16 DOCUMENTAÇÃO DA CAMADA DE INTEGRAÇÃO (RECOMENDADO)

A camada de integração precisa estar suficientemente documentada, visando o processo de transferência e assimilação tecnológica do sistema.

Sugere-se a utilização do padrão Swagger, tendo uma descrição clara de todos os seus endpoints, para fim de documentação das APIs.

3.17 DOCUMENTO DE VISÃO (MANDATÓRIO)

Toda aplicação a ser adquirida precisa possuir um documento de visão detalhando as regras de negócio do sistema e as principais funcionalidades.

O documento de visão não é o manual de usuário da aplicação, mas o manual para a sustentação do sistema.

Recomenda-se que o documento de visão seja único, em formato aberto, isto é, não-proprietário, escrito em português do Brasil e deve referenciar a exata versão do produto a ser adquirido.

3.18 CRONOGRAMA E PLANO PARA IMPLANTAÇÃO (MANDATÓRIO)

O órgão ADQUIRENTE de uma solução deve ter clareza quanto ao plano de implantação da solução. Existem modalidades de comercialização de softwares que não abarcam processos de implantação.

É recomendado que o órgão ADQUIRENTE exija um plano de implantação como parte da solução fornecida, incluindo um cronograma com a relação dos atores responsáveis bem como a listagem de todas as dependências, informando as respectivas versões e documentação das variáveis de ambiente.

3.19 ENTREGAS INCREMENTAIS (RECOMENDADO)

Ao adquirir uma solução personalizada, ou seja, uma aplicação construída sob demanda, a ser desenvolvida a partir de uma concepção original ou baseada no código fonte de alguma solução preexistente, exigir que as entregas sejam incrementais, tal qual prescrito em manuais de gestão ágil de projetos de software, a exemplo do SCRUM.

A cada iteração, o software produzido precisa ser homologado, isto é, precisa ser verificado se as funcionalidades entregues correspondem àquelas que foram demandadas, resultando num documento que formalize a entrega/recebimento do software produzido e, de preferência, já seja colocado em produção uma vez que efetuada a homologação.

3.20 ANÁLISE DE PONTO DE FUNÇÃO (MANDATÓRIO)

Para fins de mensuração e faturamento dos projetos de desenvolvimento e melhoria de sistemas, a CODATA utiliza a metodologia de “contagem estimativa” de pontos de função, conforme as especificações e instruções contidas no documento

“Análise de Pontos de Função Inicial” - Versão 2015 ou superior, de autoria da NESMA – Netherlands Software Metrics Users Association.

Os serviços de sustentação de sistemas medidos utilizando-se a técnica de Análise em Pontos de Função FPA, conforme as especificações contidas no Manual de Práticas e Contagens (CPM) Versão 4.3, ou superior, publicado pelo IFPUG. Os cenários não contemplados pelo CPM serão avaliados conforme o Roteiro de Métricas de Software do SISP (RMS-SISP) Versão 2.3, ou superior, ou do Guia de Contagem de Pontos de Função do SISP para Projetos Data Warehouse, versão 1.0, disponíveis em https://www.gov.br/governodigital/pt-br/sisp.

É recomendado que projetos que envolvam sustentação de sistemas ou desenvolvimento e melhoria de sistemas contratados com outras empresas sigam as mesmas metodologias. Inclusive, os artefatos produzidos nos processos de mensuração, tais como análises de postos de função, componham as respectivas entregas.

3.21 LICENCIAMENTO DE COMPONENTES DA SOLUÇÃO (MANDATÓRIO)

Caso a solução de software a ser adquirida seja composta por elementos proprietários ou o ambiente para execução da aludida solução necessite de algum recurso, como: sistema operacional, *runtime* de linguagem de programação, *framework* ou alguma biblioteca, cujo licenciamento resulte em algum dispêndio, deverão constar em TERMO DE REFERÊNCIA da aquisição os itens a seguir:

- O responsável pela compra das referidas licenças de uso, caso se aplique;
- Descrição das limitações das licenças;
- Quantidade de núcleos de processamento cobertas pela licença;
- Volume de dados ou memória suportada.

Em via de regra, a compra de tais licenças é de responsabilidade do ADQUIRENTE da solução.

O mesmo se aplica para soluções que façam uso de APIs de acesso restrito.

4 CONSIDERAÇÕES FINAIS

A CODATA se reserva ao direito de impôr condições à admissão ou recusar quaisquer demandas referentes a soluções adquiridas por órgãos da Administração Direta ou Indireta, fornecidos por terceiros, que não sigam ao menos uma das recomendações contidas neste documento.

Angelo Guissepe Guido de Araújo Rodrigues
Diretor-Presidente

Helder Vieira da Silva
Diretor de Desenvolvimento de Sistemas

Eduardo Paiva Varandas
Diretor de Tecnologia da Informação e Comunicação

Renato Mendes Oliveira Filho
Diretor Administrativo e Financeiro