

UNIVERSIDADE FEDERAL DE GOIÁS
INSTITUTO DE INFORMÁTICA
Bacharelado em Engenharia de Software
Arquitetura de Software
Discentes: Gustavo Batista, Murillo Nunes, Saulo Calixto

CHECKLIST DE AVALIAÇÃO DA IMPLEMENTAÇÃO DA ARQUITETURA

1. Introdução

O presente documento tem como objetivo apresentar os resultados da aplicação de checklist de avaliação da implementação da arquitetura com base nas decisões arquiteturais e RAS.

2. Requisitos Arquiteturalmente Significativos Atendidos

- RNF-SegTrans: Atendido pelo item: 9;
- RNF-SegInfo: Atendido pelo item: 9;
- RNF-AutentUsu: Atendido pelo item: 9;
- RNF-ContrAcesPapel: Atendido por nenhum item da avaliação.;
- RNF-IntgrCercom: Atendido por nenhum item da avaliação.;
- RNF-AmbOpServ: Atendido pelos itens: 1 e 3;
- RNF-AplicWeb: Atendido pelos itens: 7, 11 e 14;
- RNF-FuncIndep: Atendido pelos itens: 4, 5 e 12;

3. Cobertura dos Atributos de Qualidade Priorizados

De acordo com os RAS associados a cada atributo de qualidade priorizado, foi determinado um valor para avaliar o atendimento daquele atributo. O número de RAS atendidos dividido pelo total de RAS resulta no valor que varia de 0, atributo não é atendido, ou 1, atende completamente o atributo.

- Segurança - 1;
- Portabilidade - 1;
- Confiabilidade - 1;
- Manutenibilidade - N/A. Não existe RAS para esse atributo.;

4. Quantidade de Decisões que Atendem Atributos de Qualidade

De acordo com os RAS associados a cada atributo de qualidade priorizado, foi contabilizado o total de vezes que algum RAS de um determinado atributo de qualidade foi atendido por uma decisão arquitetural.

- Segurança - 6;
- Portabilidade - 4;
- Confiabilidade - 4;
- Manutenibilidade - N/A. Não existe RAS para esse atributo.;

5. Checklist de Avaliação

Produto:			Arquitetura SempreUFG	Data:	02/07/2018
Decisão Arquitetural Relacionado	RAS Atendido	#	Item	Atende	Não atende
DSUFG01	RNF-AmbOpServ	1	A arquitetura do SempreUFG é dividida em quatro camadas lógicas: cliente, servidor, APIGateway e banco de dados.	X	
	RNF-AmbOpServ	2	A camada cliente se comunica com o servidor através de requisições APIs REST.	X	
	RNF-AmbOpServ	3	A camada servidor não conhece a camada cliente.	X	
DSUFG02	RNF-FuncIndep	4	A camada servidor é composta por vários micro serviços.	X	
	RNF-FuncIndep, RNF-IntgrCercom	5	Cada microserviço representa alguma funcionalidade crítica do SempreUFG.	X	
		6	Os microserviços se comunicam através de endpoints REST API.	X	
DSUFG03	RNF-AplicWeb	7	A camada cliente é feita utilizando React.	X	
		8	Utiliza-se Redux para o gerenciamento de estado.	X	
DSUFG04	RNF-SegTrans, RNF-SegInfo, RNF-AutentUsu, RNF-ContrAcesPapel	9	É utilizado o protocolo de autorização OAuth2 para fazer a autenticação dos usuários no sistema.	X	
DSUFG05	RNF-SegTrans, RNF-SegInfo	10	Os microserviços irão expor seus serviços através de APIs restfull.	X	
	RNF-AplicWeb	11	Utiliza para conexão o protocolo HTTP e os dados são trafegados utilizando JSON ou XML.	X	

	RNF-FuncIndep	12	Cada comunicação REST é independente, ou seja, nela é preciso ter toda informação necessária.	X	
DSUFG06	RNF-FuncIndep	13	A orquestração dos microserviços é feito pela ApiGateway.	X	
DSUFG07	RNF-AplicWeb	14	Os microserviços são construídos em Java.	X	
		15	É utilizado o paradigma OO.	X	
DSUFG08		16	A gerência do banco de dados é feita pelo PostgreSQL.	X	
DSUFG09	RNF-FuncIndep	17	Cada microserviço tem seu próprio banco de dados.	X	
DSUFG10		18	Todos as configurações dos microserviços são centralizados no GitHub.	X	
DSUFG11		19	É utilizado o Eureka como serviço de descoberta de microserviços.	X	