

Lista de Regras de Negócio, Requisitos Funcionais e Não-Funcionais

Nome do projeto:	Sistema de Agendamento de Salas Corporativas (SASC)		
Nome dos participantes	Glenda Souza Fernandes dos Santos.	R.A	12723135022
	Marcus Vinicius Lameu Lima		12722128023
	Victor Oliveira Cerqueira		12723112575
	Jorge Henrique Ramos Gandolfi		1272313453
	Guilherme Ornellas Carvalho		1272319173
	Isaac Dias		12724130145

Identificador	Regras de Negócio
RNG-01	O sistema não deve permitir reserva duplicada
RNG-02	O cliente deve estar logado para acessar o sistema
RNG-03	O cliente poderá alocar a sala se ela estiver desocupada.
RNG-04	O cliente deverá reservar a sala com até uma hora de antecedência.
RNG-05	O cliente poderá cancelar a reserva com até 30 minutos de antecedência.
RNG-06	A reserva das salas só poderá ocorrer das 07:00 até às 18:00.
RNG-07	O limite de tempo das reservas é de no máximo 2 horas.
RNG-08	A reserva será feita por ordem de chegada.
RNG-09	O sistema deve emitir uma confirmação de agendamento automático.
RNG-010	O cadastro de usuários deve incluir e-mail e senha.

Lista de Requisitos Funcionais

Identificador	Requisitos Funcionais
RF-01	O sistema deve fazer uma validação de acesso dos usuários.
RF-02	Cadastro de salas inclui capacidade de pessoas, andar.
RF-03	Visualização de disponibilidade exibir salas livres em tempo real
RF-04	Permitir agendamento.
RF-05	Emitir um comprovante de validação
RF-06	O sistema deve permitir o cadastro de usuários.

Lista de Requisitos Não-Funcionais

Identificador	Requisitos Não-Funcionais
RNF-01	A taxa de ocorrência de falhas deverá ser a mínima possível.
RNF-02	O sistema deverá estar sempre disponível e acessível para os usuários.
RNF-03	O tempo de resposta do sistema deverá ser o mínimo possível.
RNF-04	O sistema deverá garantir que os dados estão protegidos de acesso não autorizado.

Descrição das Interações no Modelo Espiral

Fase 1 - RNG's; RF's e RNF's:

- **Determinação dos Objetivos:** Ao definir as regras de negócios e os requisitos funcionais e não-funcionais do nosso sistema temos como objetivos principais garantir um entendimento claro e compartilhado do que o software deve fazer e como deve se comportar. Trazendo com clareza a definição do escopo, das funcionalidades detalhadas e atributos do software, garantimos um desenvolvimento mais eficiente e criamos uma base sólida para a prototipação, o desenvolvimento e testes do sistema.
- **Avaliação de Riscos:** Podem surgir diversos riscos que comprometeriam o sucesso da fase 1, eles vão desde riscos relacionados à elicitación e documentação até riscos externos, algumas contramedidas para mitigar a maioria desses riscos são: Comunicação clara e eficaz; Processo de documentação e elicitación bem estruturados; Priorização dos requisitos; Definição de critérios de claros e mensuráveis; etc.
- **Desenvolvimento e Teste:** Através de uma leitura minuciosa do documento passado pelo cliente foi realizado um brainstorm entre as partes interessadas e foram definidos as regras de negócio e os requisitos funcionais e não-funcionais do software a ser desenvolvido, essas diretrizes serão apresentadas por meio de tabelas em conjunto ao código fonte e os protótipos.
- **Planejamento da próxima Interação:** Após uma construção rigorosa da Fase 1, alcançamos com sucesso todos os objetivos planejados. Com esse resultado excelente, decidimos avançar para a Fase 2: a prototipação do software.

Fase 2 - Prototipação:

- **Determinação dos Objetivos:** Os principais objetivos são a comunicação visual do produto para obtenção de aprovação das partes interessadas e para auxiliar junto com as regras de negócio a construção do back-end da aplicação.
- **Avaliação de Riscos:** Os principais riscos seriam relacionados ao escopo, como o desvio do que foi definido pela fase 1, a falhas no processo de criação do protótipo ou na obtenção de feedback, entre outros. As principais estratégias a serem adotadas nesses casos envolvem: A atenção aos cumprimentos dos requisitos do escopo; Planejar as sessões de feedback e documentar as decisões tomadas; etc.
- **Desenvolvimento e Teste:** Após um debate bem fundamentado sobre seus benefícios para o desenvolvimento do software, como interface intuitiva e recursos de compartilhamento, a equipe de prototipação optou pela construção de uma

interface visual produzida no Figma, considerando seu alinhamento e colaboração para a melhoria da aplicação.

- **Planejamento da próxima Interação:** Com o sucesso no desenvolvimento e conclusão das fases iniciais (1 e 2), a próxima etapa irá se concentrar no desenvolvimento completo da aplicação. Isso envolverá a construção do back-end e do front-end, momento em que serão definidas as linguagens de programação e tecnologias a serem utilizadas em todo o processo de desenvolvimento e implementação do software.

Fase 3 - RAD(Desenvolvimento da aplicação):

- **Determinação dos Objetivos:** O objetivo principal é a criação de um Produto Mínimo Viável (MVP) funcional, priorizando a implementação das funcionalidades essenciais.
- **Avaliação de Riscos:** Os principais riscos que poderiam surgir seriam: Problemas de integração entre frontend e backend; Comunicação ineficaz; Bugs e erros não detectados; Falta de ferramentas adequadas; Mudanças nas prioridades do negócio, entre outros. As soluções a serem adotadas podem variar mas em sua maioria são: A definição clara das linguagens utilizadas; o estabelecimento de canais de comunicação claros e eficientes, reuniões diárias curtas, ferramentas de comunicação online, etc; a definição de papéis e responsabilidades claramente; a implementação de testes unitários, de integração, funcionais e manuais, etc.
- **Desenvolvimento e Teste:** Para a produção da aplicação nesta etapa, foram designadas duas equipes distintas: uma focada no desenvolvimento do front-end utilizando HTML e CSS, e outra encarregada do back-end, construído com o framework Spring Boot. A escolha dessas tecnologias se deu em função do maior conhecimento e experiência da equipe com elas. Adicionalmente, priorizou-se uma comunicação segura e eficaz entre as equipes para assegurar o alinhamento e a conformidade da aplicação em sua totalidade.
- **Planejamento da próxima Interação:** Após a conclusão bem-sucedida de todas as fases, o sistema passará por manutenções preventivas para garantir sua estabilidade e evitar a ocorrência de bugs. Além disso, serão implementadas atualizações contínuas para assegurar sua competitividade no mercado e para identificar e mitigar prontamente quaisquer riscos à sua integridade.