

t



Documento de Visão

Histórico de Revisão

Data	Versão	Descrição	Autor
16/08	0.1	Criação do documento de visão	Lucas Fernandes
18/08	0.2	Adição da história dos requisitos	Karina Hagiwara
22/08	0.3	Alteração da história dos requisitos	Karina Hagiwara
23/08	0.4	Organização e Completude do Documento	Guilherme Bornia
03/09	0.5	Cartões de Atributos	Guilherme Bornia
06/09	0.6	Visão Geral do Produto	Guilherme Bornia
18/10	0.7	Revisão do documento e atualização do diagrama de caso de uso	Lucas e Luís
25/10	0.8	atualização de dados	Lucas

Sumário

Histórico de Revisão	1
Introdução	3
Escopo	3
Oportunidade de Negócio	4
Glossário	4
Principais Necessidades da Parte Interessada ou do Usuário	5
Visão Geral do Produto	6
Resumo das Capacidades	6
Custo e Precificação	7
Recursos do Produto	7
Requisitos Conscientes	7
Requisitos Subconscientes	8
Requisitos de Restrição	8
Requisitos de Qualidade	9
Requisitos Inconscientes	9
Cartões de Atributos	9
Restrições	13
Apêndice	14
Identificação dos Riscos	14
Diagrama de Casos de Uso	17
Cronograma	18

1. Introdução

Este documento tem como propósito especificar o que será desenvolvido, para quem e quando será entregue, além de delimitar as funções do sistema.

As partes interessadas no software estão aqui descritas e, para elas, há um glossário com as palavras comumente usadas e essenciais para compreender o objetivo do sistema. Também é feita uma análise de mercado para verificar a demanda pelo software assim como também foi feita a listagem dos requisitos extraídos do cliente por meio de uma entrevista não-estruturada e, posteriormente, de uma entrevista estruturada.

Também são estabelecidas as restrições do projeto, os riscos existentes e um diagrama de casos de uso. Dessa forma, espera-se que as partes envolvidas compreendam o projeto do sistema “RateMyCampus”.

2. Escopo

O objetivo do software “RateMyCampus” é oferecer uma visão crítica da situação dos campus das universidades brasileiras através da avaliação dos próprios alunos que possuem uma vivência em determinado(s) campus.

Será desenvolvido um sistema no qual haverá as universidades e seus campus pré-cadastrados e, para o usuário conseguir avaliá-lo(s), será necessário que o mesmo se cadastre (aluno, professor ou funcionário) no sistema informando seus dados pessoais e o(s) campus em que estuda/leciona.

Quando cadastrado, o usuário conseguirá avaliar apenas o(s) campus que foram cadastrados em sua conta. Os objetos a serem avaliados incluem salas, professores, bibliotecas, restaurantes universitários e os cursos oferecidos por aquela unidade. Sendo que, os professores, assim como os funcionários, apenas não poderão avaliar os professores e os cursos. Quanto aos campus dos quais o usuário não faz parte, a avaliação poderá ser apenas visualizada.

É importante ressaltar que a avaliação será feita por meio de votos e que, se o usuário desejar, poderá comentar sobre aquele campus. Todos os votos e comentários serão anônimos, cabendo a equipe desenvolvedora assegurar tal sigilo.

3. Oportunidade de Negócio

RateMyCampus é destinado a todos que vivenciam seu dia a dia em uma universidade, seja aluno, seja funcionário, seja professor. Atualmente muitas universidades não estão agradando seus clientes e não há um meio tão eficaz no mercado brasileiro para que se possa mostrar, com indicadores, o quanto precisam de atenção.

Com esse software, ao criar uma conta, o usuário poderá mensurar sua opinião sobre sua universidade. Pois ao classificar seu curso, restaurante e professores gerará uma nota no qual é destinada uma média à sua universidade. Assim será possível analisar se uma universidade é um bom ambiente para conviver, se um curso é bem ministrado, se alguns professores possuem uma boa didática e até comparar universidades e cursos.

4. Glossário

Termo	Descrição
Milestone	É uma técnica de gerência de projetos que permite o teste da funcionalidade de um novo produto ao longo do projeto. Não é uma atividade e não possui duração.
Software	É uma sequência de instruções escritas para serem interpretadas por um computador com o objetivo de executar tarefas específicas. Também pode ser definido como os programas que comandam o funcionamento de um computador.
Backup	Termo em inglês que tem o significado de cópia de segurança. É frequentemente utilizado em informática para indicar a existência de cópia de um ou mais arquivos guardados em diferentes dispositivos de armazenamento. Se, por qualquer motivo, houver perda dos arquivos originais, a cópia de segurança armazenada pode ser restaurada para repor os dados perdidos.

5. Principais Necessidades da Parte Interessada ou do Usuário

As partes envolvidas no desenvolvimento do software “RateMyCampus” são clientes universitários os quais procuraram uma equipe para construir um sistema que pudesse cumprir com os seus requisitos.

Os clientes são dois estudantes: Victor Hugo e André Gonçalves - enquanto a equipe é composta por oito membros cujos nomes são Guilherme Borna, Luis Ricardo, Ana Carolina Pedroso, Lucas Fernandes, Karina Hagiwara, Vitor Artoni, Mirelle Bueno e João Victor Ignácio.

Foram levantadas as seguintes perguntas para extrair as necessidades das partes interessadas e através delas, foi elaborada uma tabela para facilitar a leitura:

- Quais são os motivos para o problema?
- Qual é a prioridade do problema?
- Quais soluções a parte interessada deseja?

Necessidade	Prioridade	Solução Encontrada
Saber o que precisa ser melhorado dentro do campus.	Alta	Elaborar um sistema que avalie toda a estrutura do campus e forneça um feedback por meio de uma nota.
Saber, se a avaliação do campus for ruim, ao que se o usuário está referindo e o que ele acha que pode ser melhorado.	Média/Alta	Permitir que o usuário realize comentários de forma opcional e anonimamente.

6. Visão Geral do Produto

6.1. Resumo das Capacidades

Benefício para o Cliente	Recursos de Suporte
Visão geral da infraestrutura e do corpo docente da universidade.	É possível obter uma visão geral das avaliações acerca das salas, cursos, etc.
Avaliação por meio de notas e comentários.	Se as notas não forem suficientes para o usuário expressar sua opinião, ele pode expressá-la por meio de comentários, os quais oferecem uma maior liberdade para a crítica.
Auxilia na tomada de decisão.	Por meio do feedback das notas e comentários extrai-se o que precisa ser investido para ter uma melhoria.

6.2. Custo e Precificação

Nome	Função	Custo/hora
Guilherme Bornia	Gerente	R\$ 51,63
Karina Sayuri	Programador	R\$ 15,94
Lucas Fernandes	Analista	R\$ 28,61
Luis Ricardo	Analista	R\$ 28,61
Ana Pedroso	Programador	R\$ 15,94
Mirelle Bueno	Programador	R\$ 18,12
Vitor Artoni	Programador	R\$ 15,94
João Victor	Programador	R\$ 15,94

Total =	R\$ 10.679,90
----------------	----------------------

Fonte: <http://www.catho.com.br/profissoes/>

7. Recursos do Produto

7.1. Requisitos Conscientes

- 7.1.1. A votação deve contemplar o restaurante universitário, a biblioteca, as salas de aula, os professores e o curso para fornecer indicadores de satisfação em relação a infraestrutura do local, bem como suas particularidades.
- 7.1.2. A votação deve ser feita através de um questionário, ou seja, com opções pré-estabelecidas, para ser possível gerar uma média de cada opção votada, assim como gerar uma média geral.
- 7.1.3. O resultado dos votos deve ser exibido por meio de símbolos (estrelas, por exemplo) para facilitar a “visualização crítica” do usuário acerca daquela unidade de ensino.
- 7.1.4. A votação deve conter um campo opcional para comentários para que o usuário consiga expressar sua opinião caso o questionário não a contemple.

- 7.1.5. Uma média geral das notas dadas deve estar presente no sistema para que a avaliação contemple as opiniões de todos os eleitores e, assim, mostrar um resultado justo para todos.
- 7.1.6. O aluno deve poder apenas votar no(s) campus onde estuda para não influenciar nas notas de outras unidades já que o mesmo pode não conhecer outro local além do seu próprio campus. Isso traria mais precisão nos dados.
- 7.1.7. Não-alunos não devem poder votar, apenas visualizar, já que desconhecem a unidade de ensino a ser votada. Com isso, seria possível gerar dados mais precisos dos campus.
- 7.1.8. Cadastrar se faz necessário tanto para votar quanto para apenas visualizar para que se tenha um controle de quantos votos terão que ser processados e para se ter uma noção de quantos usuários existem no sistema.
- 7.1.9. O cadastro do usuário deve exigir apenas nome, e-mail e senha daqueles que não forem alunos e exigir daqueles que são alunos, além dessas informações, o nome da faculdade em que estuda, bem como o nome do campus e o curso. Dessa forma, é possível restringir a que cada usuário tem acesso.
- 7.1.10. Os votos e comentários dos usuários devem ser anônimos para que não haja medo por parte deles em votar ou expressarem suas opiniões acerca da unidade de ensino que frequentam.

7.2. Requisitos Subconscientes

7.2.1. Requisitos de Restrição

- 7.2.1.1. O sistema deve ser implementado para desktops (Windows 8.0 ou posterior) pois é a plataforma utilizada pelo cliente e, portanto, se faz necessário para poder obter os benefícios oferecidos pelo software.
- 7.2.1.2. O sistema exige conexão com a Internet para que os usuários consigam se cadastrar e visualizar o seu conteúdo de qualquer lugar.

- 7.2.1.3. Um banco de dados deve ser criado para armazenar as informações de cadastro dos usuários para que, dessa forma, seja possível identificar o perfil de cada um.
- 7.2.1.4. Talvez seja necessário fazer uma conexão com o servidor da UNICAMP para validar as informações do cadastro de alunos por meio do RA e, assim, garantir com segurança quem são os usuários que estão votando nas unidades de ensino.
- 7.2.1.5. O banco de dados do sistema deve ser hospedado em um servidor para que as informações dos usuários sejam armazenadas de forma mais organizada e com maior segurança.

7.2.2. Requisitos de Qualidade

- 7.2.2.1. Segurança da informação dos usuários é extremamente importante para que as opiniões e comentários do sistema permaneçam anônimas.
- 7.2.2.2. O sistema deve ter flexibilidade e desempenho para exercer suas funcionalidades com uma boa eficiência em diversos ambientes e, com isso, agradar ao usuário.

7.3. Requisitos Inconscientes

- 7.3.1. Um mecanismo de busca pelas unidades de ensino se faz necessário para que o usuário não encontre dificuldades em poder realizar o seu voto e/ou comentário, ou seja, para que ele encontre o que procura facilmente.

8. Cartões de Atributos

Identificador:	01.
Nome:	Cadastrar Usuário.
Descrição:	O usuário, ao acessar o sistema pela primeira vez, insere seus dados e faz o próprio cadastro.

Categorização de Kano:	Requisito Consciente.
Responsabilidade:	João Victor.
Fonte:	Cliente.
Autor:	Lucas e Luís.

Identificador:	02.
Nome:	Fazer Login.
Descrição:	O usuário insere seu e-mail e senha para acessar o sistema.
Categorização de Kano:	Requisito Consciente.
Responsabilidade:	Vitor e João.
Fonte:	Cliente.
Autor:	Lucas e Luís.

Identificador:	03.
Nome:	Consultar Campus.
Descrição:	O usuário pode pesquisar outros campus além do seu e visualizar os dados e avaliações de cada um.
Categorização de Kano:	Requisito Consciente.
Responsabilidade:	João.
Fonte:	Cliente.
Autor:	Lucas e Luís.

Identificador:	04.
Nome:	Consultar Professor.
Descrição:	O usuário pode clicar no menu “Professor” e visualizar os dados e as avaliações de cada um.
Categorização de Kano:	Requisito Consciente.
Responsabilidade:	João e Karina.
Fonte:	Cliente.
Autor:	Lucas e Luís.

Identificador:	05.
Nome:	Consultar Biblioteca.
Descrição:	O usuário pode clicar no menu “Biblioteca” e visualizar os dados e as avaliações dela.
Categorização de Kano:	Requisito Consciente.
Responsabilidade:	João e Karina.
Fonte:	Cliente.
Autor:	Lucas e Luís.

Identificador:	06.
Nome:	Consultar Restaurante.
Descrição:	O usuário pode clicar no menu “Restaurante” e visualizar os dados e as avaliações dele.
Categorização de Kano:	Requisito Consciente.
Responsabilidade:	João e Karina.
Fonte:	Cliente.
Autor:	Lucas e Luís.

Identificador:	07.
Nome:	Consultar Sala de Aula.
Descrição:	O usuário pode clicar no menu “Sala de aula” e visualizar os dados e as avaliações das salas de cada do campus.
Categorização de Kano:	Requisito Consciente.
Responsabilidade:	João e Karina.
Fonte:	Cliente.
Autor:	Lucas e Luís.

Identificador:	08.
Nome:	Consultar Curso.
Descrição:	O usuário pode clicar no menu “Curso” e visualizar os dados e as avaliações de cada um.
Categorização de Kano:	Requisito Consciente.
Responsabilidade:	João e Karina.

Fonte:	Cliente.
Autor:	Lucas e Luís.

Identificador:	09.
Nome:	Avaliar Biblioteca.
Descrição:	O frequentador do campus, ao consultar as informações da biblioteca, pode dar notas aos seus diferentes tópicos.
Categorização de Kano:	Requisito Consciente.
Responsabilidade:	João e Karina.
Fonte:	Cliente.
Autor:	Lucas e Luís.

Identificador:	10.
Nome:	Avaliar Restaurante.
Descrição:	O frequentador do campus, ao consultar as informações do restaurante, pode dar notas aos seus diferentes tópicos.
Categorização de Kano:	Requisito Consciente.
Responsabilidade:	João e Karina.
Fonte:	Cliente.
Autor:	Lucas e Luís.

Identificador:	11.
Nome:	Avaliar Sala de Aula.
Descrição:	O frequentador do campus, ao consultar as informações das salas de aula, pode dar notas aos seus diferentes tópicos.
Categorização de Kano:	Requisito Consciente.
Responsabilidade:	João e Karina.
Fonte:	Cliente.
Autor:	Lucas e Luís.

Identificador:	12.
-----------------------	-----

Nome:	Avaliar curso.
Descrição:	O aluno, ao consultar as informações do curso, pode dar notas aos seus diferentes tópicos.
Categorização de Kano:	Requisito Consciente.
Responsabilidade:	João e Karina.
Fonte:	Cliente.
Autor:	Lucas e Luís.

Identificador:	13.
Nome:	Avaliar professor.
Descrição:	O aluno, ao consultar as informações do professor, pode dar notas aos seus diferentes tópicos.
Categorização de Kano:	Requisito Consciente.
Responsabilidade:	João e Karina.
Fonte:	Cliente.
Autor:	Lucas e Luís.

Identificador:	14.
Nome:	Enviar Comentário.
Descrição:	O frequentador do campus pode deixar seus comentários para reforçar suas opiniões sobre os quesitos do campus em que pode votar.
Categorização de Kano:	Requisito Consciente.
Responsabilidade:	João.
Fonte:	Cliente.
Autor:	Lucas e Luís.

9. Restrições

- 9.1. Desenvolver o projeto apenas nas terças-feiras, durante 4 horas;
- 9.2. Não pode dar problemas no repositório como corromper um arquivo, por exemplo;
- 9.3. O tempo total de desenvolvimento é curto;

- 9.4. A falta de conhecimento prático por parte dos membros pode impactar no projeto;
- 9.5. A conexão com a Internet é fundamental;

10. Apêndice

10.1. Identificação dos Riscos

Id: 1	Data: 16/08/16	Probabilidade: 50%	Impacto: 8 (0 a 10)
Descrição: Programador se ausentar do trabalho devido à problema de saúde.			
Mitigação: Cuidar da saúde, não se expor a situações de risco.			
Plano de contingência: Escalar um programador secundário que esteja ciente do projeto.			
Status: Parado		Autor: Mirelle Bueno	

Id: 2	Data: 16/08/16	Probabilidade: 50%	Impacto: 9 (0 a 10)
Descrição: Servidor local apresentar defeitos ou dificuldades de acesso.			
Mitigação: Fazer backup diários.			
Plano de contingência: Utilizar servidores alternativos para acesso.			
Status: Parado		Autor: Mirelle Bueno	

Id: 3	Data: 16/08/16	Probabilidade: 60%	Impacto: 8 (0 a 10)
Descrição: O ambiente de trabalho ficar sem acesso a internet.			
Mitigação: Usar acesso de internet secundário.			
Plano de contingência: Utilizar os backups e adaptar o projeto.			
Status: Parado		Autor: Mirelle Bueno	

Id: 4	Data: 16/08/16	Probabilidade: 80%	Impacto: 08 (0 a 10)
Descrição: Computadores apresentarem defeitos			
Mitigação: Disponibilizar computadores alternativos para uso de desenvolvimento.			
Plano de contingência: Utilizar um computador alternativo.			
Status: Parado		Autor: Mirelle Bueno	

Id: 5	Data: 16/08/16	Probabilidade: 60%	Impacto: 04 (0 a 10)
Descrição: Problemas de comunicação com o cliente principal			
Mitigação: Estabelecer contatos alternativos (Ex:Skype)			
Plano de contingência: Entrar em contato com os usuários secundários.			
Status: Parado		Autor: Mirelle Bueno	

Id: 6	Data: 23/08/16	Probabilidade: 5%	Impacto: 10 (0 a 10)
Descrição: Crise econômica suspender atividades da universidade e congelar o projeto.			
Mitigação: Procurar por recursos externos e independentes.			
Plano de contingência: Analisar a situação em conjunto com o cliente			
Status: Parado		Autores: Mirelle Bueno e Vitor Artoni	

Id: 7	Data: 23/08/16	Probabilidade: 50%	Impacto: 9 (0 a 10)
Descrição: O software não conseguir uma boa aceitação por parte dos usuários.			
Mitigação: Requisitos levantados se aproximarem o máximo do que os usuários realmente querem.			
Plano de contingência: Rever o projeto e propor melhorias ao software			
Status: Parado		Autores: Mirelle Bueno e Vitor Artoni	

Id: 8	Data: 23/08/16	Probabilidade: 40%	Impacto: 9 (0 a 10)
Descrição: Os dados anônimos das avaliações vazarem			
Mitigação: Elaborar mecanismo de segurança dos dados			
Plano de contingência: Encontrar o responsável e reforçar a segurança			
Status: Parado		Autores: Mirelle Bueno e Vitor Artoni	

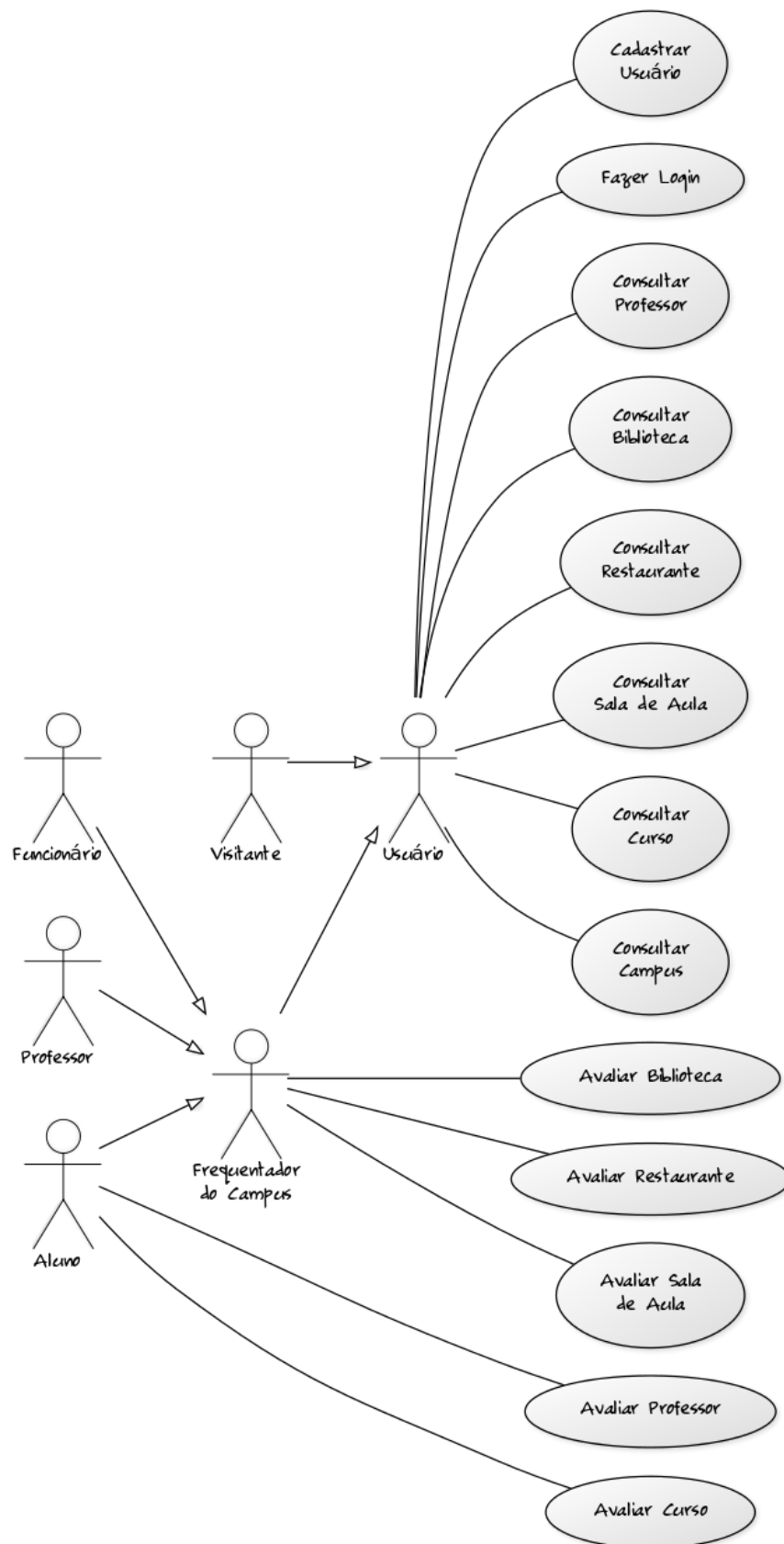
Id: 9	Data: 23/08/16	Probabilidade: 70%	Impacto: 8 (0 a 10)
Descrição: Erro no planejamento do projeto causa atraso na entrega			
Mitigação: Revisar constantemente o planejamento			
Plano de contingência: Adaptar o máximo possível o planejamento e encurtar os			

prazos	
Status: Parado	Autores: Mirelle Bueno e Vitor Artoni

Id: 10	Data: 23/08/16	Probabilidade: 30%	Impacto: 7 (0 a 10)
Descrição: Incompatibilidades do software com o sistema operacional			
Mitigação: Atentar-se a questões de compatibilidade e atualizações do sistema operacional			
Plano de contingência: Convocar os desenvolvedores e resolver o problema			
Status: Parado		Autores: Mirelle Bueno e Vitor Artoni	

Id: 11	Data: 23/08/16	Probabilidade: 35%	Impacto: 5 (0 a 10)
Descrição: Risco não ser encontrado nos testes e causar transtornos ao usuário			
Mitigação: Certificar-se de que os testes estão cobrindo a maioria dos erros			
Plano de contingência: Resolver o problema o mais rápido possível			
Status: Parado		Autores: Mirelle Bueno e Vitor Artoni	

10.2. Diagrama de Casos de Uso



10.3. Cronograma

Semana	Data	Tema
4	23/08	Milestone (M1): Definição dos objetivos do ciclo de vida
8	20/09	Milestone (M2): Ciclo de vida da arquitetura
14	01/11	Milestone (M3): Capacidade operacional inicial
15	08/11	Milestone (M4): Release do produto