

**Instituto de Matemática e Estatística - USP**  
**MAC0218 - Técnicas de Programação II**

# **MONOLITO**

Bruno Carneiro Cunha - NUSP

Daniel da Silva Nunes - NUSP 10297612

Eduardo Rocha Laurentino - NUSP 8988212

Pedro Henrique Barbosa de Almeida - NUSP 10258793

## Versão Final

Chegamos a versão final para entrega da prova de conceito do SaaS no qual trabalhamos neste semestre. Na última fase, entregamos o aplicativo já com a estrutura do banco de dados completo, views bem encaminhadas e recursos fundamentais para a nossa proposta, como a possibilidade de realizar upload de arquivos, devidamente implementados.

No relatório da primeira entrega, no qual especificamos nossos objetivos, escrevemos: “Pretendemos levantar toda a estrutura para registo de usuários, criação de novos projetos, edição de contas e projetos, exibição de conteúdo para interação dos usuários com as imagens, e levantamento de estatísticas sobre as respostas a ser enviada para os autores dos projetos junto com as rotulações requeridas por ele”.

Felizmente, chegamos a versão final com esses objetivos conquistados, ainda que, talvez, não da melhor maneira possível. A ideia, porém, sempre foi mostrar que a nossa ideia, o nosso projeto de SaaS, é factível, e provar isso desenvolvendo uma versão minimal que pudesse elucidar o conceito o suficiente para termos essa prova de conceito feita. E acreditamos termos conseguido, conforme explicamos ao longo deste relatório.

## O que foi feito desde a última entrega

### 1. Criação da interface para rotulação:

Uma vez que já era possível realizar o upload de arquivos e, assim, criar projetos com um conjunto de imagens associado, o passo seguinte foi criar uma interface que permitisse o funcionamento do mecanismo central da nossa proposta, que é a interface de rotulação das imagens de um projeto. Isto é, agora ao cadastrar um projeto é necessário indicar a pergunta de rotulação que deve ser passada para os usuários, bem como as devidas opções de respostas. Com isso, todo projeto tem necessariamente um autor, descrição, o conjunto de imagens, sua pergunta de rotulação e suas opções de resposta.

Do ponto de vista do usuário que contribuirá com a rotulação, as imagens do projeto são exibidas de maneira aleatória junto com a pergunta e opções de resposta, bastando selecionar a resposta que julgar correta ou pular para a próxima imagem.

### 2. Criação do mecanismo de obtenção estatística à partir das rotulações:

Com a interface de rotulação descrita anteriormente em mãos, o próximo passo foi agregar valor ao mecanismo de acordo com o conceito de obtenção de estatísticas que definimos na nossa proposta. Isto é, permitir que pra cada imagem de um determinado projeto, que portanto tem uma pergunta e as devidas opções de resposta

associadas, fique registrado não só a quantidade de respostas mas que respostas foram essas (dentre as possíveis, determinadas pelo criador do projeto, conforme já explicado).

Para tanto, observe que o nosso banco de dados possui uma entidade “Answers” associada a projetos (Projects) e imagens (Images). Com isso, podemos gerar instâncias dessas entidades a cada resposta obtida pela interface de rotulação, e associar essas informações ao projeto como um todo por meio de uma planilha que pode ser obtida pelo autor do projeto quando este clicar em “obter dados”. Não só, mas o autor pode ainda fazer o download com as informações dessa planilha em formato excel ou csv, que é uma funcionalidade que implementamos e está devidamente funcional.

### 3. Padronização e ajustes no design:

Temos trabalhado com um design funcional desde a primeira fase, de modo a possibilitar um desenvolvimento mais integrado com nossas ideias. Agora, contudo, trabalhamos em ajustar problemas de interface que ainda tínhamos, e isto passou principalmente por padronizar essas interfaces todas nos diferentes ambientes do site, das dimensões de exibição às cores, por exemplo.

### 4. Correção de bugs

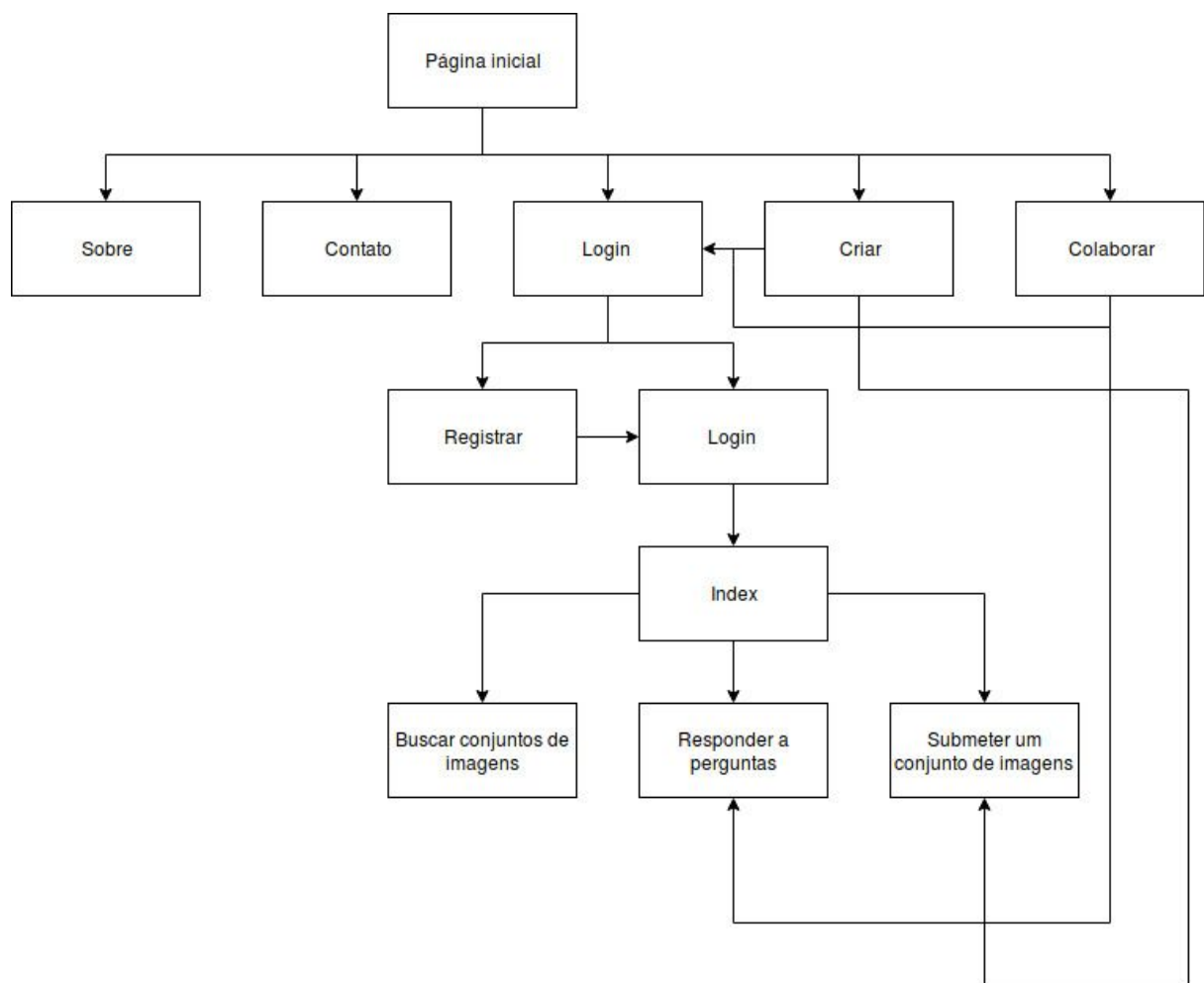
Obviamente, boa parte do trabalho em todas as fases e, portanto, também nesta, esteve relacionado a corrigir bugs. Todos os bugs centrais que nós identificamos foram corrigidos de alguma maneira, permitindo o funcionamento do sistema o suficiente para pelo menos termos a prova de conceito do nosso projeto devidamente executável. Contudo, o projeto não está 100% polido. Há alguns engargalos aqui ou ali cujos respectivos consertos, no entanto, são factíveis, bastando mais tempo de trabalho para que sejam solucionados.

## Documentação da Arquitetura

Temos reportado a documentação da arquitetura do nosso sistema desde a primeira entrega e o fazemos aqui mais uma vez por acreditarmos ser central no entendimento do nosso trabalho.

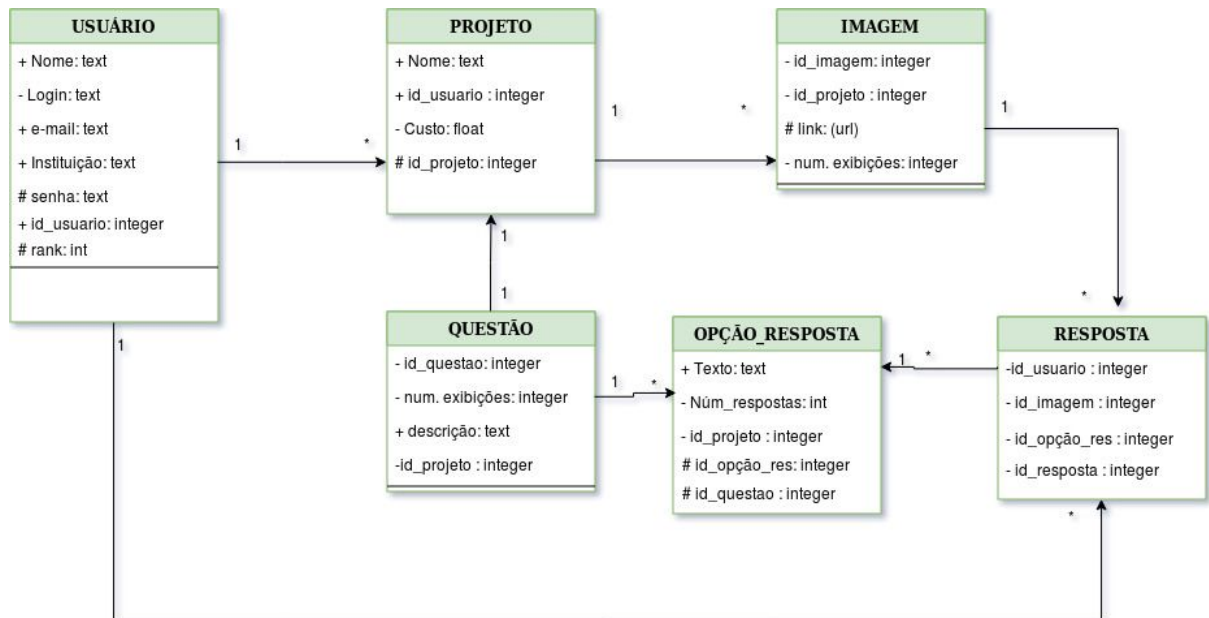
### Fluxo do site

Sem alterações desde a última entrega.



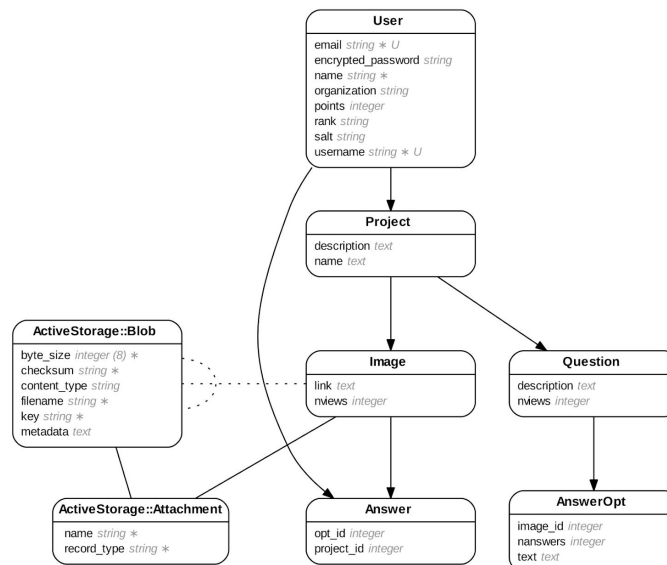
## Estrutura do banco de dados

A estrutura do banco de dados foi implementada com as devidas associações entre entidades. Chaves estrangeiras foram criadas para estabelecer relações entre as entidade e seus atributos.



A arquitetura acima implementada, somada ao uso de queries nas classes de controle, no fim das contas foram de fato suficientes para atender as funcionalidades no sistema.

Dontpanic domain model

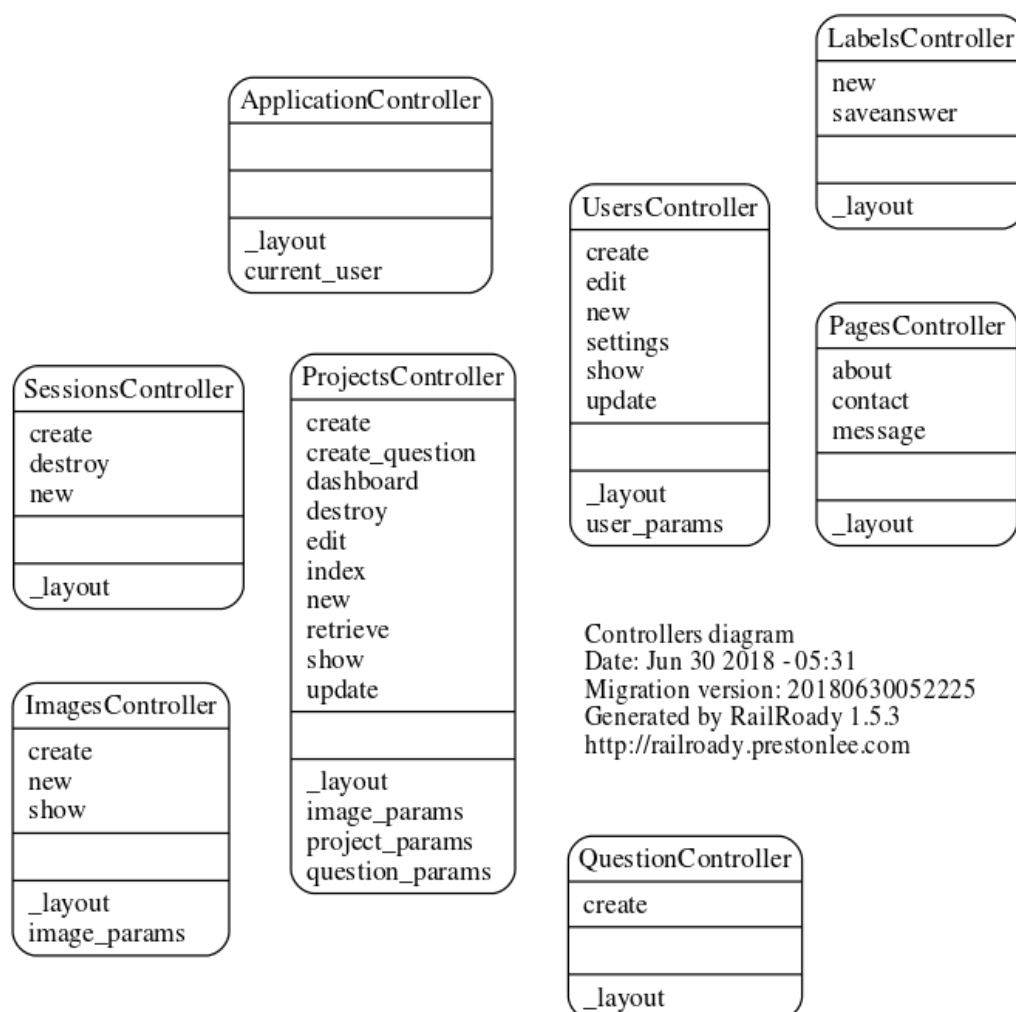


Para indicar como a arquitetura ficou em termo de classes, models e banco de dados, a cada fase, utilizamos duas ferramentas: **Rails ERD** ("is a gem that allows you to easily generate a diagram based on your application's Active Record models", ver <https://github.com/voormedia/rails-erd>) e **Rails Roady** ("generates Rails 3/4/5 model (ActiveRecord, Mongoid, Datamapper) and controller UML diagrams as cross-platform .svg files, as well as in the DOT language.", ver <https://github.com/preston/railroadly>).

Tais ferramentas nos fornecem as seguintes informações sobre o estágio atual do sistema (todos os arquivos originais estão na pasta docs do projeto):

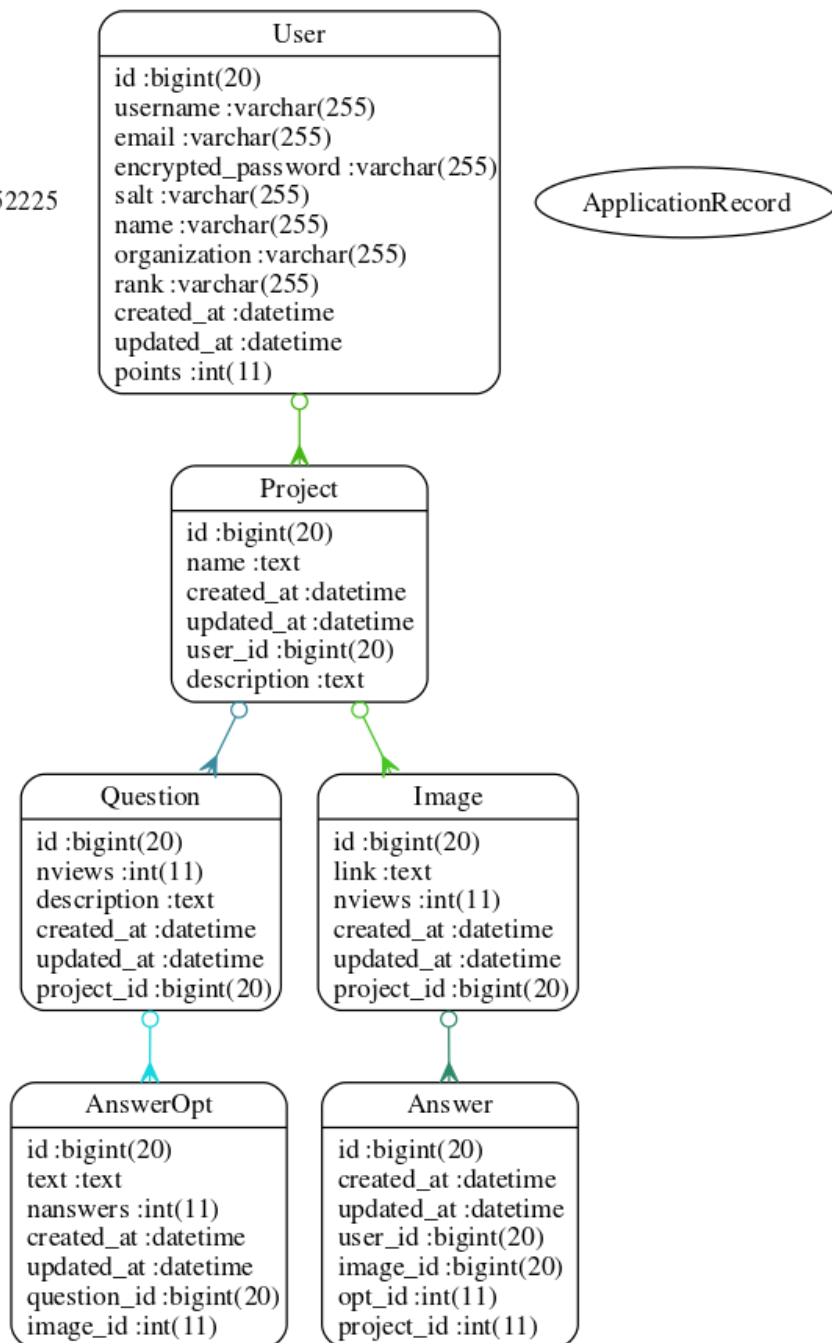
## Controllers

Observe que solucionamos a questão de controllers vazios, apontada no feedback da Entrega 1.



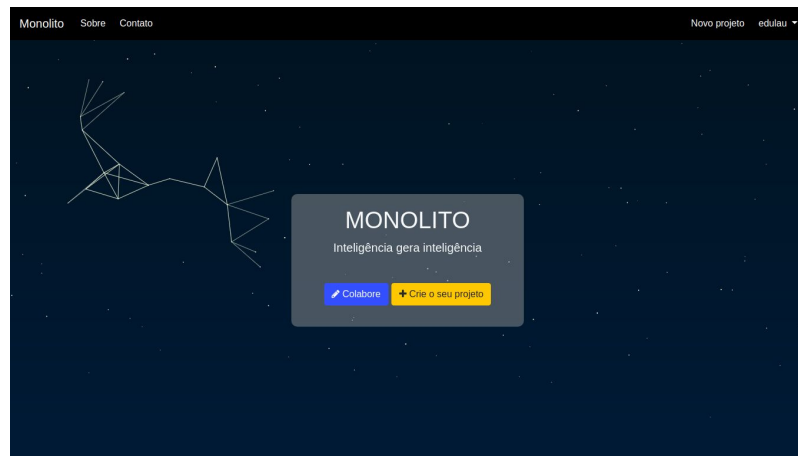
## Modelos

Models diagram  
Date: Jun 30 2018 - 05:31  
Migration version: 20180630052225  
Generated by RailRoady 1.5.3  
<http://railroady.prestonlee.com>

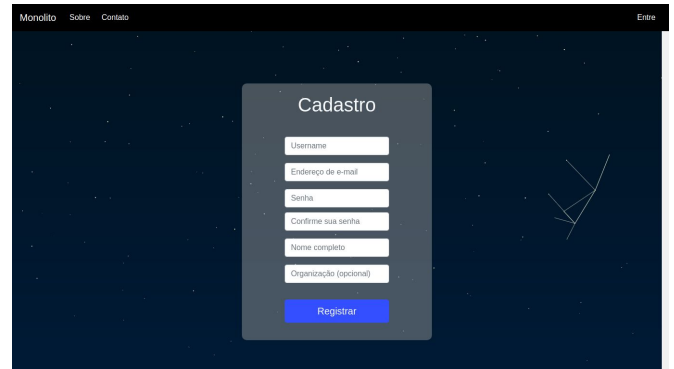
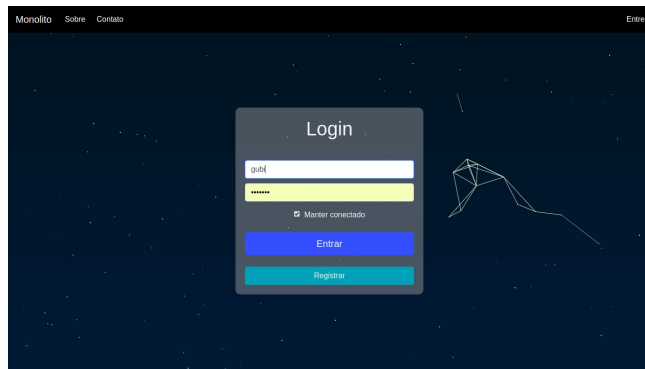


## Como utilizar

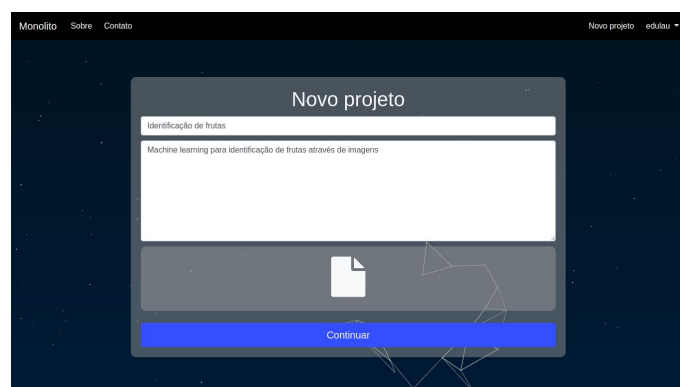
Para o bom funcionamento do sistema dentro do que implementamos, isto é, dentro das limitações de uma versão minimal, as funcionalidades operam bem desde que o banco de dados esteja populado com usuários e projetos, cada um com as respectivas informações necessárias, de descrição a imagens.



Portanto, um primeiro passo é criar um usuário pela página de cadastro e logar este usuário pela página de login.



Além disso, é importante cadastrar novos projetos com uma quantidade de imagens. De preferência criar mais de um projeto, a partir de contas de usuários diferentes.






Se você clicar em “Colabore”, será direcionado aleatoriamente para algum dos projetos existentes na interface de rotulação referente a ele para alguma de suas imagens. Se houver poucas imagens (que não é o caso da prova de conceito, onde na verdade o conjunto de imagens são bastante grandes) haverá, obviamente, repetição.

[Monolito](#) [Sobre](#) [Contato](#)

[Novo projeto](#) [edulau](#) ▼



Qual a fruta que você vê nessa imagem?

☐ Banana

☐ Abacaxi


☒ Maça

☐ Morango

☐ Não vejo fruta

[Monolito](#) [Sobre](#) [Contato](#)

[Novo projeto](#) [edulau](#) ▼



Qual a fruta que você vê nessa imagem?

☐ Banana

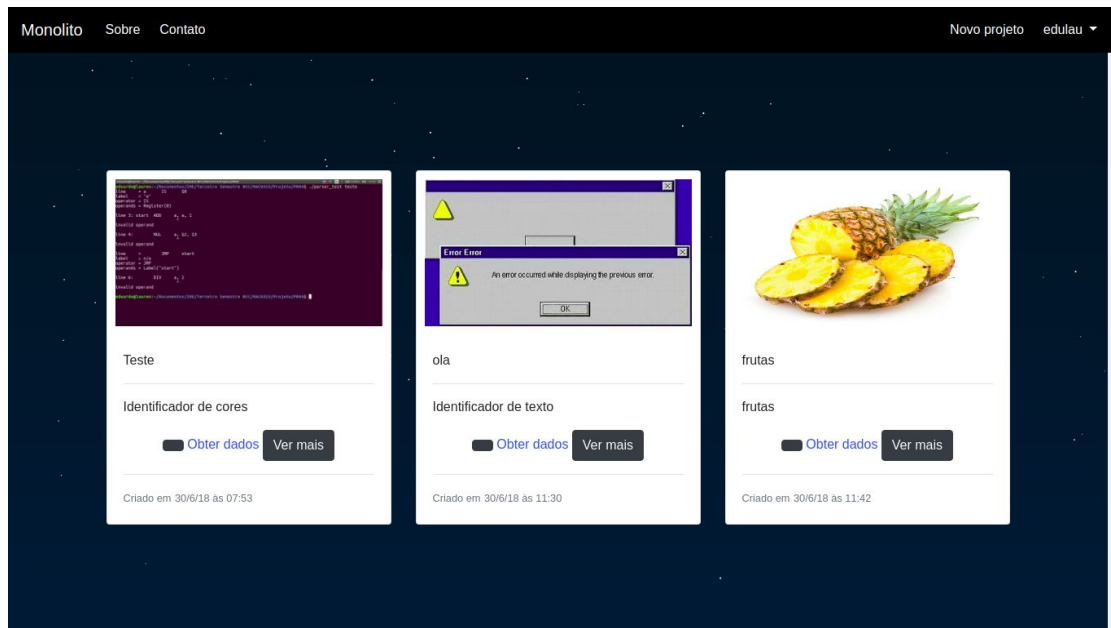
☐ Abacaxi

☒ Maça

☐ Morango




☐ Não vejo fruta

Se você é autor de projeto, existe uma dashboard em “meus projetos” na qual você pode visualizá-los, com opções para editá-los, rotulá-los ou obter dados sobre as respostas já conquistadas.



Finalmente, se você é autor de um projeto, consegue acessá-lo e obter a planilha com a relação das respostas que foram fornecidas para a pergunta cadastrada diretamente no site, mas também é possível exportá-la tanto em formato excel quanto em .csv.

The screenshot shows a table titled 'Resultado das respostas dos usuários'. The table has columns for 'Imagem', 'Opção 1', 'Opção 2', 'Opção 3', 'Opção 4', and 'Opção 5'. The data is as follows:

Imagem	Opção 1	Opção 2	Opção 3	Opção 4	Opção 5
 17	0	0	0	0	0
 18	0	0	0	0	0
 19	0	0	0	0	0

At the bottom of the table, there are two buttons: 'Exportar para o Excel' and 'Exportar para CSV'. Below the table, there is a small text: 'Algoritmo "excellentexport" para exportar os dados por jmaister, licenciado sob a Licença do MIT.'

(aqui o exemplo não de projeto não foi rotulado, por isso as estatísticas são nulas)

## Dificuldades encontradas

As dificuldades para essa entrega final foram uma continuação das dificuldades reportadas em entregas anteriores. Isto é, aqui o desafio foi basicamente se debruçar sobre os problemas que havíamos encontrado até então, buscando solucioná-los de alguma maneira para manter a prova de conceito funcional.

## Relatório de testes

Os testes foram feitos utilizando RSpec. Ao todo, 27 exemplos são testados e todos os testes têm resultado válido, abrangendo grande parte das funcionalidades implementadas.

Para executar o testador e verificar os resultados, basta executar no diretório do projeto o seguinte comando:

**bundle exec rspec**

Os testes verificam se os formulários realizam as validações corretamente, evitando dados duplicados, incoerentes ou ausentes.