

FERRAMENTA DE APOIO AO ENSINO INTERATIVO DE EXPRESSÕES REGULARES

Aluno: Gabriel Takashi Katakura

Orientadora: Joyce Martins

Roteiro

- Introdução
- Objetivos
- Fundamentação Teórica
- Trabalhos correlatos
- Requisitos
- Especificação
- Implementação
- Demonstração
- Resultados e Discussões
- Conclusões
- Sugestões

Introdução

- Autômatos finitos determinísticos
 - Aplicação web
- Aprendizado sobre expressões regulares
 - Linguagens Formais e Autômatos

Objetivos

- disponibilizar um canal interativo para o desenvolvimento e compartilhamento de atividades;
- ter um mecanismo de notificações em tempo real para o acompanhamento das atividades;
- validar as expressões regulares especificadas, determinando a equivalência entre duas expressões regulares - a especificada pelo professor e a especificada pelo estudante.

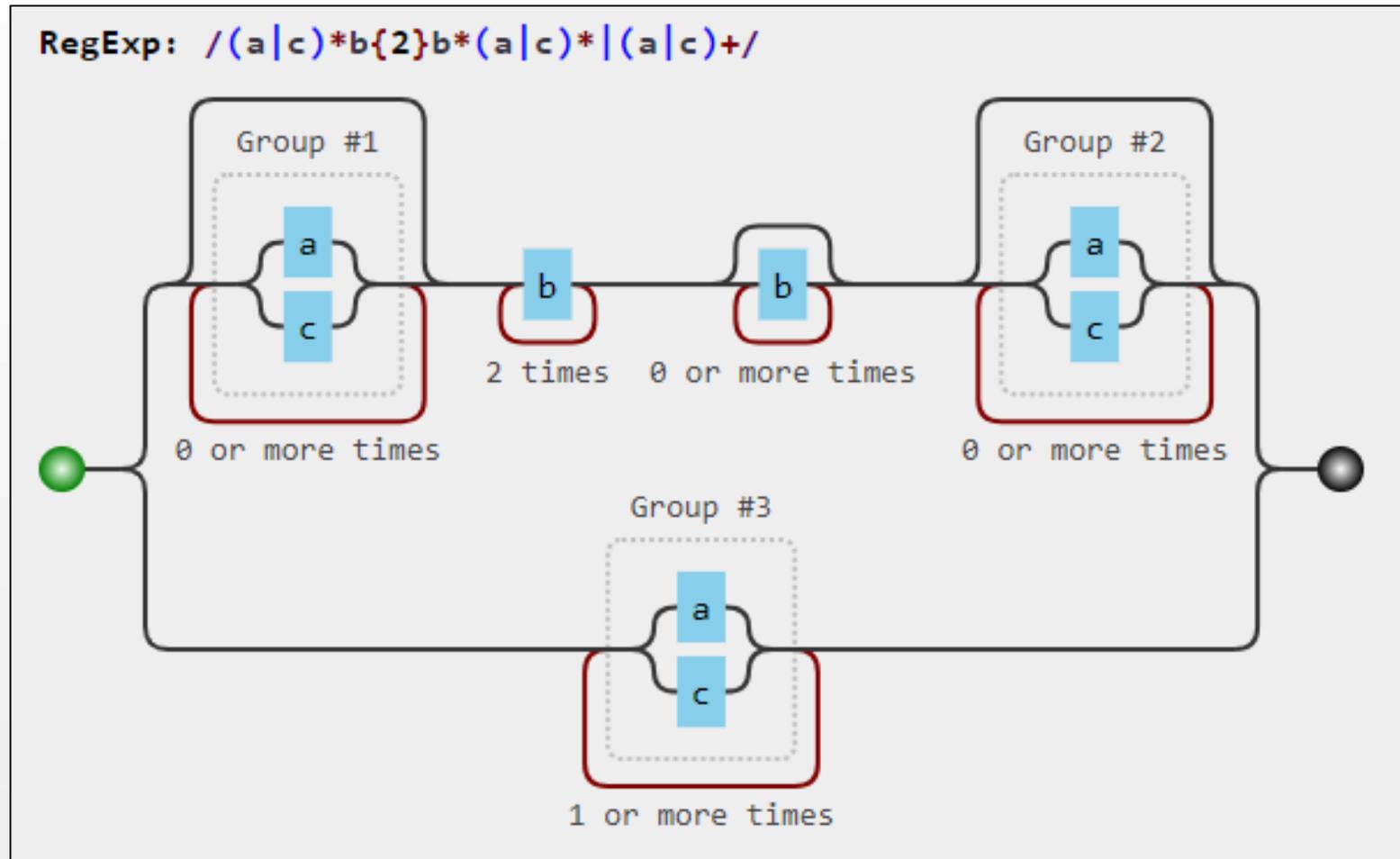
Fundamentação Teórica

- Expressões regulares
 - Padrões de texto
 - Metacaracteres

METACARACTERE	CONHECIDO COMO	PALAVRAS QUE ACEITA
{n}	exatamente n ocorrências	a definição precedente ocorre exatamente n vezes
+	uma ou mais ocorrências	a definição precedente ocorre uma ou mais vezes
*	nenhuma ou mais ocorrências	a definição precedente ocorre zero ou mais vezes
?	nenhuma ou uma ocorrência	a definição precedente ocorre zero ou uma vez
	alternativa	uma ou outra definição
(...)	grupo	definição de grupo
[...]	conjunto	definição de conjunto de caracteres

Expressões Regulares

$L = \{ w \mid w \in \{a, b, c\}^+ \text{ e todos os bs de } w, \text{ se existirem, são consecutivos} \}$



Trabalhos Correlatos

- Regulex
- JFLAP
- Regex101

Regulex

Regulex

JavaScript Regular Expression Visualizer.

View on GitHub

```
/ (a|c)*b{2}b*(a|c)*|(a|c)+ /
```

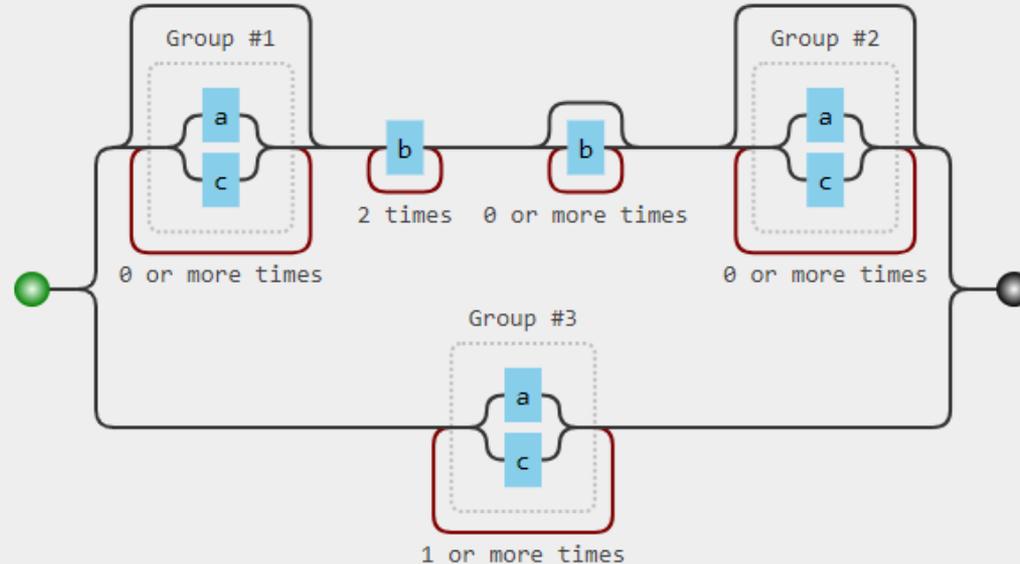
Visualize

Export Image

Embed On My Site!

IgnoreCase Multiline GlobalMatch

RegExp: `/(a|c)*b{2}b*(a|c)*|(a|c)+/`



JFLAP

JFLAP v8.0 (Beta)

File Edit Input Convert Help

Regular Expression Editor

Expression: $a + b^*$

Regular Expression = (Σ, E)

$\Sigma = \{ a b \}$

Table Text Size

JFLAP v8.0 (Beta)

File Edit Input Test View Convert Help

Automaton Editor

```
graph LR; q0((q0)) -- ε --> q2((q2)); q0 -- ε --> q4((q4)); q2 -- a --> q3((q3)); q3 -- ε --> q1(((q1))); q4 -- b --> q5((q5)); q5 -- ε --> q4; q5 -- ε --> q1;
```

Finite State Automaton (FSA) = $(Q, \Sigma, \delta, S, F)$

$\Sigma = \{ a b \}$

Table Text Size

Regex101

REGULAR EXPRESSION 108 matches, 3037 steps (~10ms)

`/ (a|c)*b+(a|c)* / mg`

TEST STRING SWITCH TO UNIT TESTS ▶

a b c

aa ab ac ba bb bc ca cb cc

aaa aab aac aba abb abc aca acb acc baa bab bac bba bbb bbc bca bcb bcc

caa cab cac cba cbb cbc cca ccb ccc

aaaa aaab aaac aaba aabb aabc aaca aacb aacc abaa abab abac abba abbb

abbc abca abcb abcc acaa acab acac acba acbb acbc acca accb accc baaa

baab baac baba babb babc baca bacb bacc bbab bbab bbac bbba bbbb bbbc

bbca bbcb bbcc bcaa bcab bcac bcba bccb bcbc bcca bccb bccc caaa caab

caac caba cabb cabc caca cacb cacc cbaa cbab cbac cbba cbbb cbbc cbca

cacb cbcc ccaa ccab ccac ccba ccbb ccbc ccca cccb cccc

Correlatos - Comparativo

características	trabalhos	Regex101	JFLAP	Regulex
testar valores a partir das expressões regulares especificadas		X	*	
destacar valores validados pela expressão regular		X		
representação gráfica de uma expressão regular			X	X
utilização de perfis				
acompanhamento/registo de atividades				
validar a equivalência entre duas expressões regulares				
ambientes suportados		web	desktop	web

Requisitos

- Professor
 - Controle de exercícios, atividades e turmas
 - Acompanhamento das atividades
 - Notificações na ferramenta em tempo real
 - Informações sincronizadas em tempo real
 - Por e-mail quando uma resposta é válida
- Estudante
 - Minhas atividades
 - Resumo das atividades
 - Resolução de exercícios

Casos de Uso

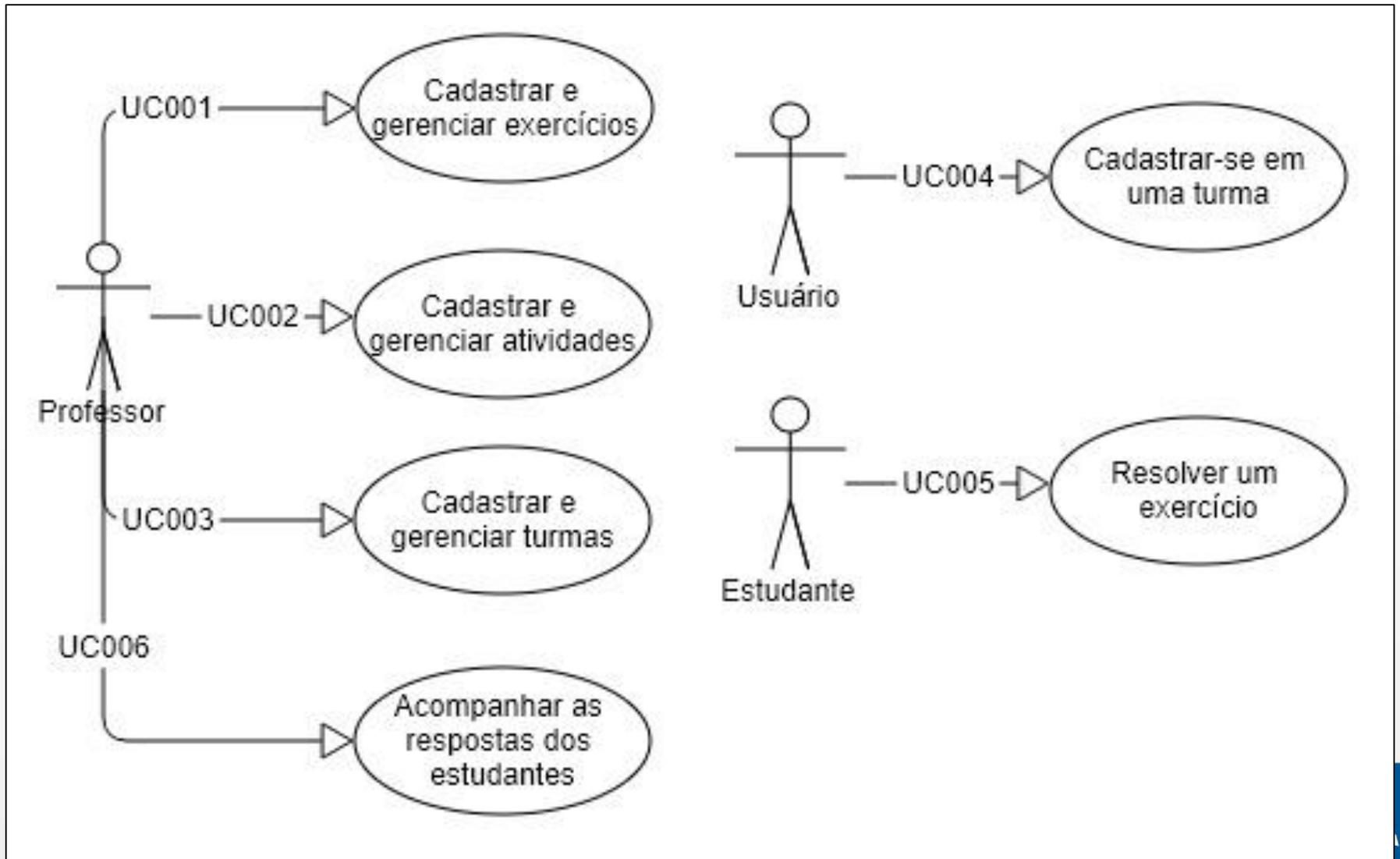
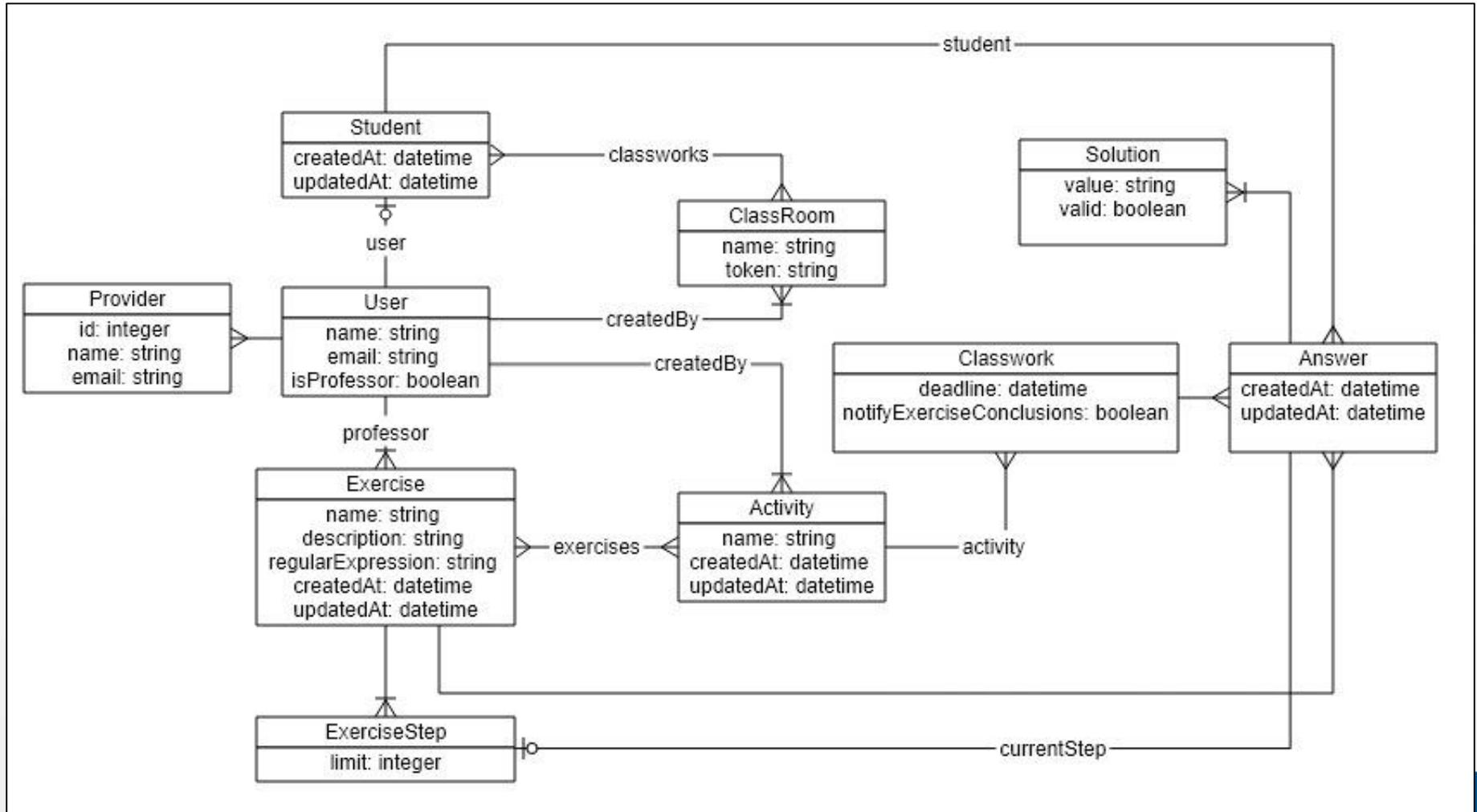


Diagrama de Classes



Implementação

- Validação de respostas na resolução de exercícios
 - Cliente
 - Erros léxicos ou sintáticos
 - Símbolos inexistentes no alfabeto do enunciado
 - Palavras válidas e inválidas
 - Servidor
 - Comparação literal
 - Equivalência entre expressões regulares através de autômatos finitos determinísticos mínimos

Implementação

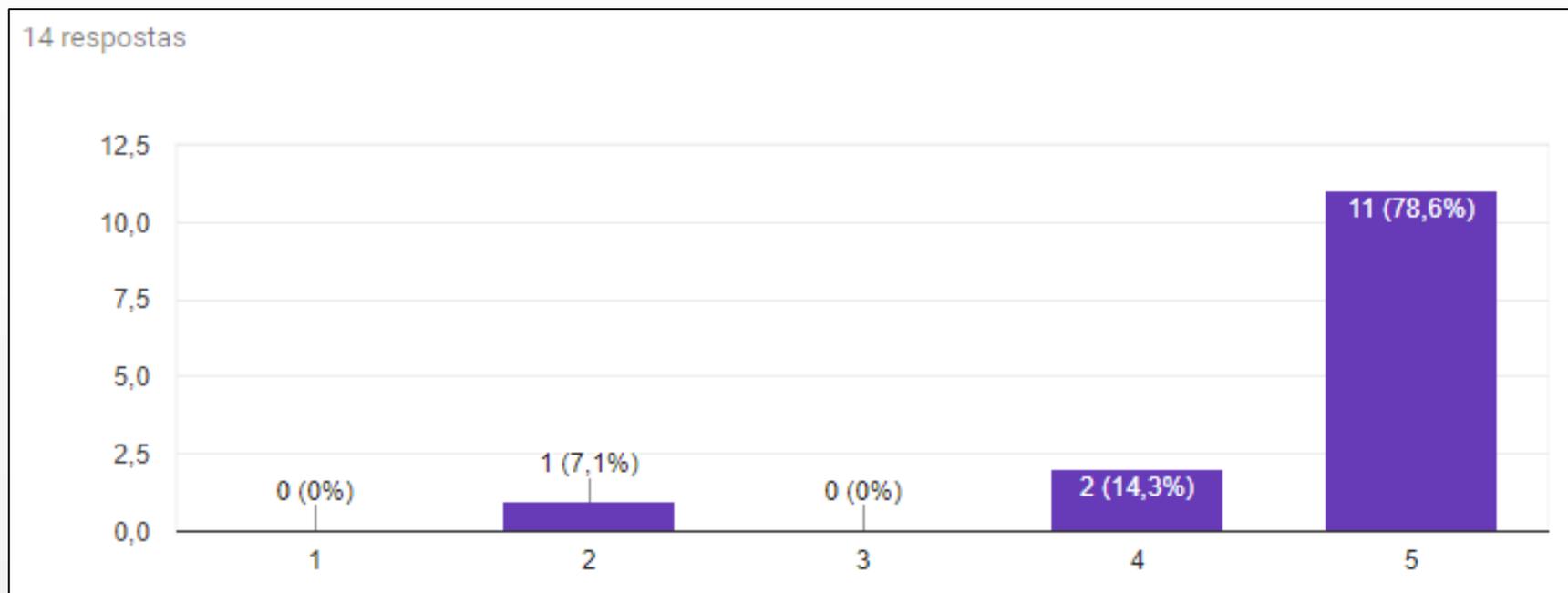
- Notificação e atualização de dados em tempo real
 - Cliente
 - Redux
 - Single Source of Truth
 - Sistema baseado em eventos
 - Observer Pattern
 - Servidor
 - WebSockets

Demonstração

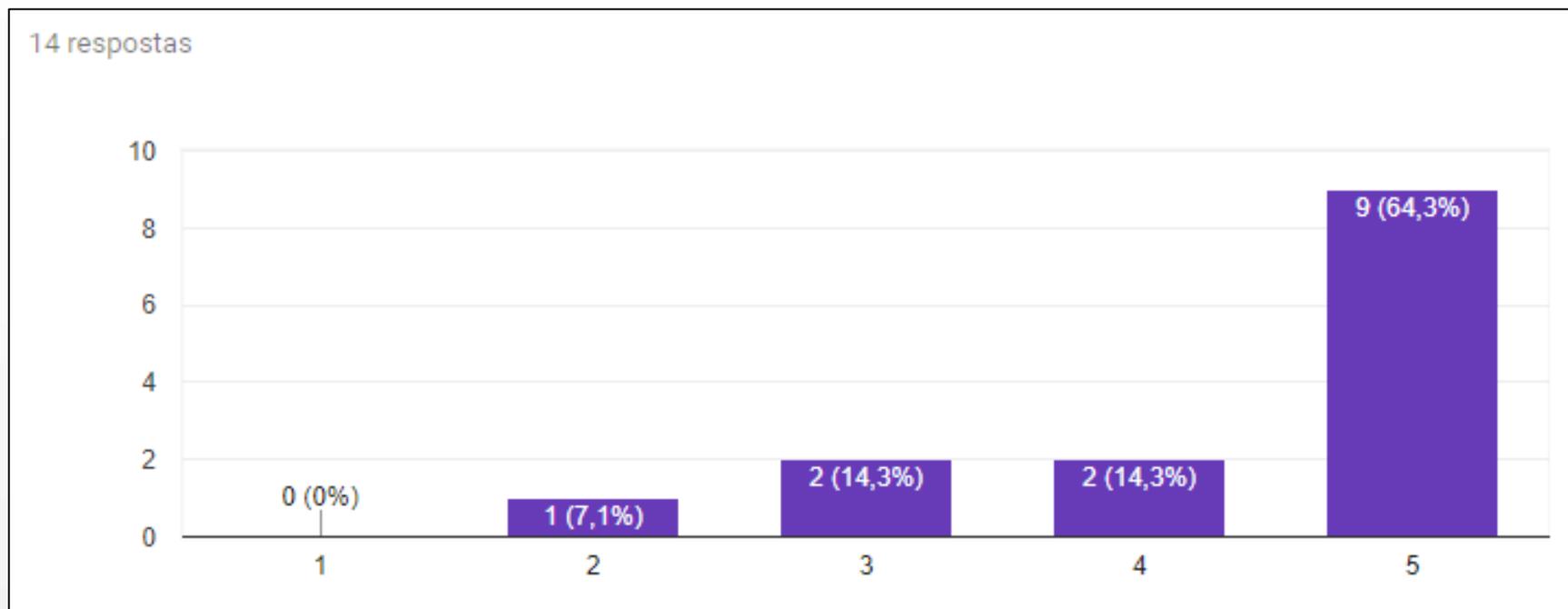
Resultados e Discussões

- Teste prático em sala de aula com 14 estudantes de Sistemas da Informação na FURB
- Formulário
 - Ambientes de testes
 - Sistemas operacionais: Windows, Mint
 - Navegadores: Chrome e Firefox
 - Usabilidade
 - Funcionalidade

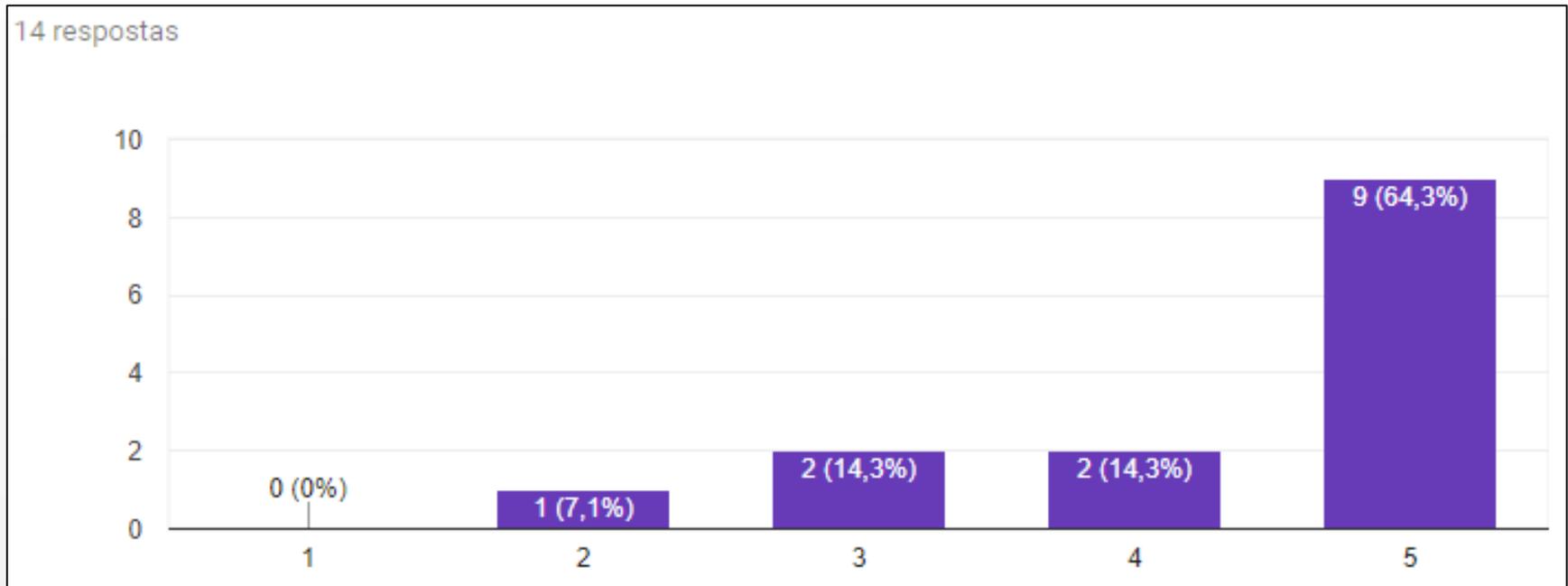
Usabilidade - Usar a ferramenta foi complicado?



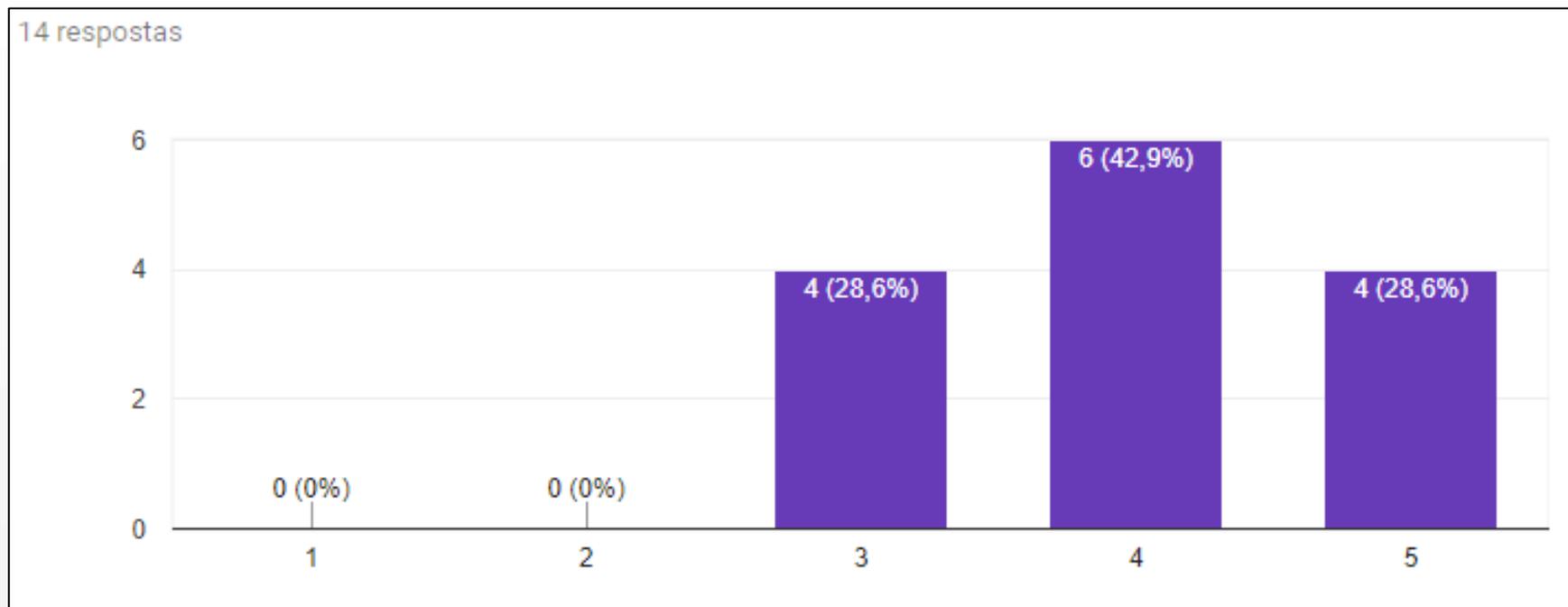
Usabilidade - A forma como as informações são apresentadas é clara e compreensível?



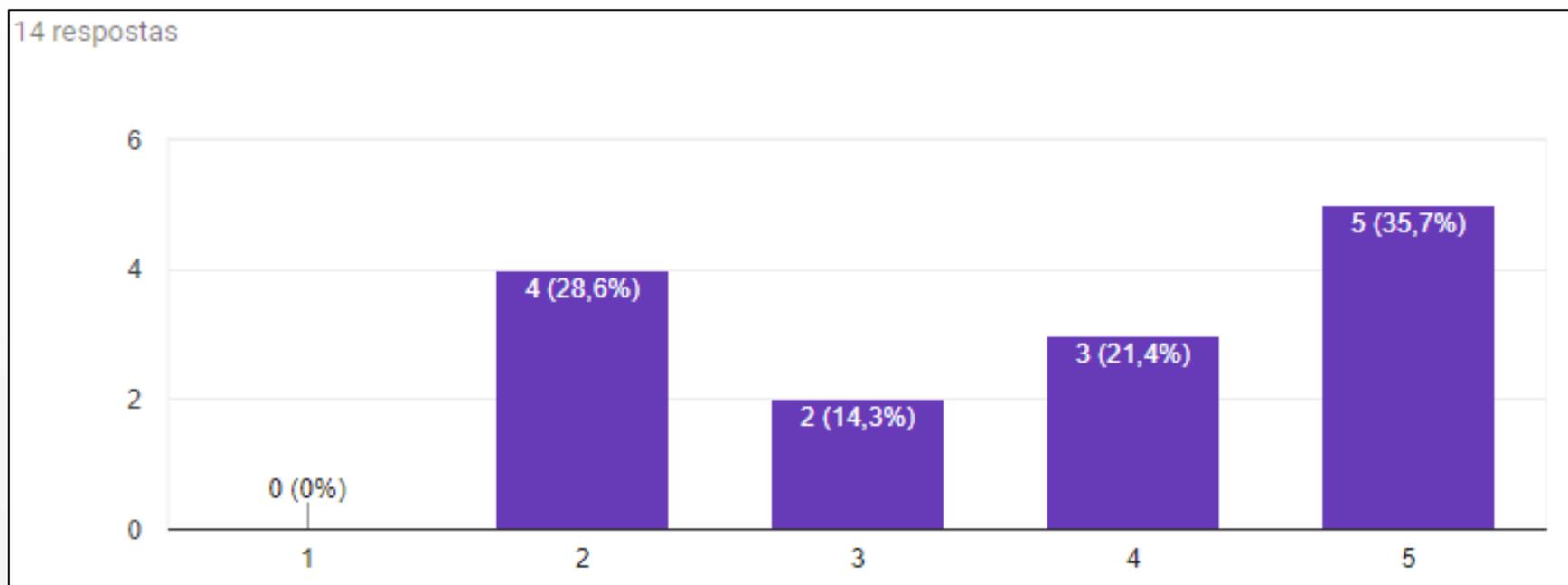
Usabilidade - É fácil navegar entre as telas da ferramenta?



Funcionalidade - A ferramenta facilitou a resolução dos exercícios?

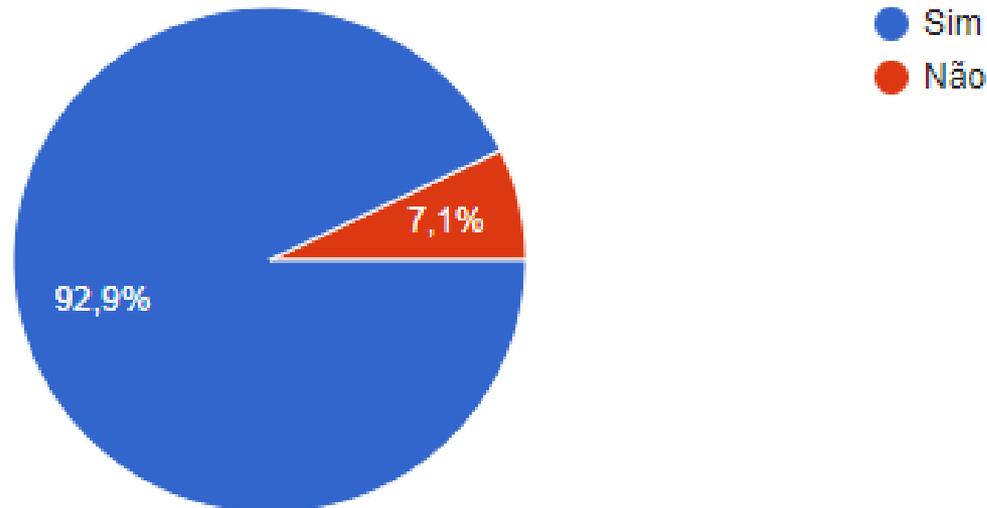


Funcionalidade - As mensagens de erros na resolução do exercício auxiliaram a encontrar a solução?



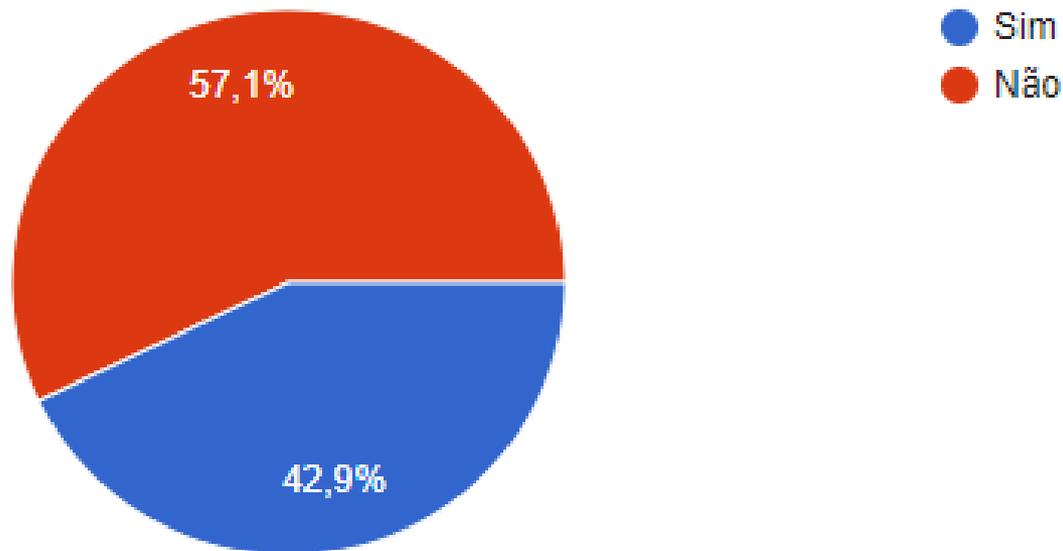
Funcionalidade - Você gostaria de ter usado essa ferramenta no aprendizado sobre expressões regulares?

14 respostas



Geral - Você teve alguma dificuldade ou dúvida para resolver as questões?

14 respostas



Comentários

No geral o Learn REGEX ajuda bastante a aprender. Ao invés do aluno ter que abrir o enunciado, abrir [a ferramenta], etc., o "Learn Regex" compacta tudo em uma só página contendo todas as funcionalidades e com um visual agradável.

O Learn REGEX possui uma interface muito mais amigável e mais prática. A ausência de um monte de menus facilita muito o aprendizado, ao contrário do RegExr.

A questão de mostrar possibilidades válidas e inválidas quando você erra a expressão regular ajuda MUITO. [...] Minha crítica é [...] Apoio totalmente o uso dessa ferramenta nas aulas de Linguagens Formais e Autômatos :)

A ferramenta Learn REGEX é mais focada para a aprendizagem, enquanto as outras que citei, para o "debug" e teste de Regex. Por isso, considero a ferramenta Learn Regex muito boa.

O Regexr se apresenta superior por pequenos detalhes que considero importantes para a usabilidade. Elenco aqui dois deles que mais senti falta no Learn REGEX [...]

Conclusões

- Todos os objetivos propostos foram alcançados
- Pelos resultados obtidos, conclui-se que:
 - É uma ferramenta simples e prática para o estudante
 - Serve como auxílio ao ensino de expressões regulares

Conclusões

- O regex2dfa possui problemas com expressões regulares específicas, exemplo:

$(a | b)^* (aa | bb) (a | b) \{10\}$

Sugestões

- trocar o regex2dfa por outra ferramenta capaz de validar a equivalência entre duas expressões regulares sem causar estouros de memória em casos específicos;
- desenvolver uma solução mais adequada para auxiliar o estudante a resolver um exercício quando ele atingiu a última etapa e não conseguiu resolver ainda o exercício;
- desenvolver telas que permitam o estudante verificar todo o histórico de soluções enviadas por exercício;

Sugestões

- destacar visualmente na expressão regular especificada por um estudante quando um problema for encontrado, apontando onde o problema foi encontrado na expressão regular;
- melhorar a exibição das palavras válidas e inválidas na resolução de exercícios, para quando possuir muitas palavras o usuário não precisar ficar utilizando a barra de rolagem para encontrar qual palavra ele precisa validar adequadamente;
- melhorar as telas de acompanhamento em tempo real, para destacar visualmente alterações efetuadas na tela, em vez de somente alterar os dados da tela;
- colocar uma opção de cadastro de usuário na própria ferramenta, bem como abranger outras opções por redes sociais como o Github e o Twitter.