

UNIVERSIDADE FEDERAL DE SÃO CARLOS
DEPARTAMENTO DE COMPUTAÇÃO

DISCIPLINA DE ENGENHARIA DE SOFTWARE 2

Relatório 1 do Projeto Prático

Amanda Basso de Oliveira - 727331
Jorge Vinicius Gonçalves - 758594
Lucas Cruz Reis - 754757
Lucas Martins de Almeida - 758983
Matheus Schmidt Figueiredo - 758616
Pabolo Vinicius da Rosa Pires - 760648

Outubro de 2020

List of Figures

1	Exemplo de caso válido e de caso não válido	2
2	Entrada com apenas um caracter	2
3	Saída considerada válida para ano inválido	2
4	Saída considerada válida para ano composto por letra	2
5	Tabela de testes funcionais para inserção em <i>Article</i>	3
6	Tabela de testes funcionais para inserção em <i>Book</i>	4
7	Importação válida	5
8	Importação com sucesso e sintaxe inválida	5
9	Resultado no <i>JabRef</i>	5
10	Importação com erro e sintaxe inválida	5
11	Tabela de testes funcionais para importação de itens bibliográficos	6

Contents

1	Metodologia ágil e Scrum	1
2	Tabelas de testes funcionais	2
2.1	Inserção de item bibliográfico - <i>Article</i>	2
2.2	Inserção de item bibliográfico - <i>Book</i>	3
2.3	Importação de itens bibliográficos	4
3	Conclusão	6

1 Metodologia ágil e Scrum

O projeto está sendo desenvolvido utilizando a metodologia ágil Scrum. Foi realizada uma reunião, com quatro semanas de antecedência da data de entrega final do projeto, na qual foram planejados as divisões de tarefas entre os membros do grupo, o *Product Backlog* e uma estimativa dos *Sprint Backlogs*.

A divisão de tarefas ficou determinada da seguinte forma:

- *Scrum Master*: Amanda. Além do papel de desenvolvedora, também ficou encarregada de puxar as reuniões de *Daily Scrum*.
- *Product Owner*: Lucas Reis. Responsável por criar o repositório e os *branches*, além de adicionar os colaboradores e atuar como desenvolvedor.
- Desenvolvedores: Jorge, Lucas Martins, Matheus e Pabolo. São responsáveis pelo desenvolvimento prático do projeto, atuando desde a criação das tabelas de testes manuais até a finalização do projeto.

O *Backlog*, por sua vez, ficou dividido da seguinte forma:

- Primeiro *Sprint* (duração: uma semana)
 - Testar a inserção de item bibliográfico - *Book*.
 - Testar a inserção de item bibliográfico - *Article*.
 - Testar a importação de itens bibliográficos.
- Segundo *Sprint* (duração: duas semanas)
 - Manutenção na inserção - validar ano.
 - Manutenção na inserção - validar Bibtexkey.
 - Adicionar funcionalidade - importar csv.
- Terceiro *Sprint* (duração: uma semana)
 - Testar a validação do ano.
 - Testar a validação do Bibtexkey.
 - Testar a importação de um arquivo csv.

A *Sprint Review Meeting* foi realizada uma semana após o início do primeiro *Sprint*. Nessa reunião, foram discutidos o andamento da primeira semana de projeto e os detalhes finais para elaboração do primeiro relatório.

Foi feito um esforço no sentido de manter as *Daily Scrums* diárias mas, em alguns dias, não foi possível devido a conflitos de horários entre as disciplinas realizadas pelos membros do grupo. As dificuldades iniciais do projeto estavam relacionadas a quebrar o projeto em partes menores e factíveis ao longo dos sprints, além da organização necessária para trabalhar seguindo uma metodologia ágil.

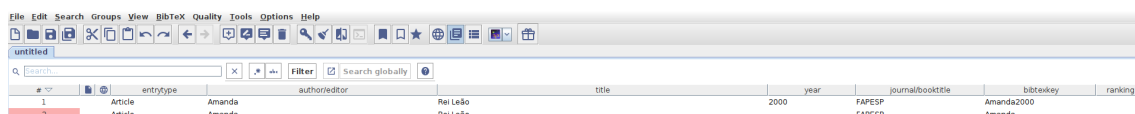
2 Tabelas de testes funcionais

Para o primeiro *Sprint* do projeto, testou-se a inserção de item bibliográfico tanto para *Book* quanto para *Article*, além da importação de itens bibliográficos.

2.1 Inserção de item bibliográfico - *Article*

Para construir a tabela de testes manual da inserção de item bibliográfico de *Article*, utilizou-se como base a saída do *JabRef* para verificar se as condições de entradas eram ou não válidas. Na figura 1, verifica-se que todos os campos requeridos devem ser preenchidos para que a saída seja válida. Caso algum dos campos esteja vazio - como o campo *Year* do segundo exemplo - o programa exibirá um símbolo vermelho na coluna #.

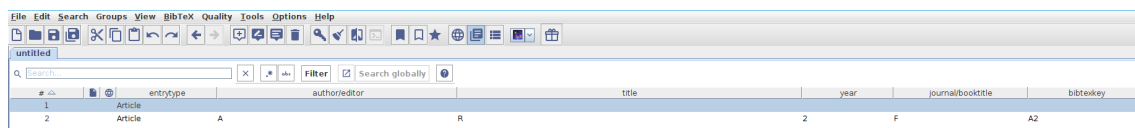
Figure 1: Exemplo de caso válido e de caso não válido



#	entrytype	author/editor	title	year	journal/booktitle	bibtexkey	ranking
1	Article	Amanda	Rei Leão	2000	FAPESP	Amanda2000	
2	Article	Amanda	Rei Leão		FAPESP	Amanda	

No entanto, vale ressaltar que o *JabRef* considera como válidas quaisquer entradas com ao menos um caracter, como observado na figura 2:

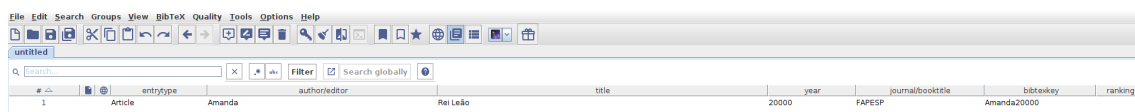
Figure 2: Entrada com apenas um caracter



#	entrytype	author/editor	title	year	journal/booktitle	bibtexkey	ranking
1							
2	Article	A	R	2	F	A2	

Verifica-se, então, que é possível adicionar entradas com anos menores que o ano atual. Outra saída válida, embora não seja esperado do programa, é a saída para entradas como anos posteriores ao ano atual, como o ano 20000. A figura 3 exibe a saída no *JabRef*:

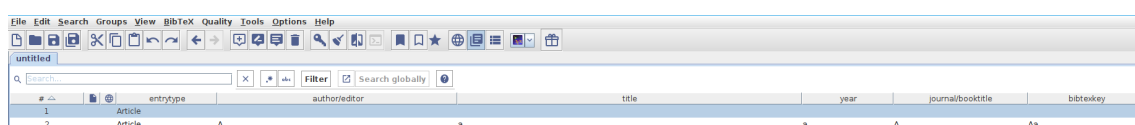
Figure 3: Saída considerada válida para ano inválido



#	entrytype	author/editor	title	year	journal/booktitle	bibtexkey	ranking
1	Article	Amanda	Rei Leão	20000	FAPESP	Amanda20000	

Um último exemplo de saída válida não esperada é a inserção de letras no campo *Year*, como verificado na imagem 4:

Figure 4: Saída considerada válida para ano composto por letra



#	entrytype	author/editor	title	year	journal/booktitle	bibtexkey	ranking
1							
2	Article	A	a	a	A	Aa	

Desse modo, foi possível criar a tabela de testes representada pela figura 5, que funciona de forma equivalente a uma tabela-verdade, visto ser necessário apenas preencher todos os campos

requeridos - *Author*, *Title*, *Journal* e *Year* - com ao menos um caracter para que a saída seja considerada válida pelo *JabRef*.

Figure 5: Tabela de testes funcionais para inserção em *Article*

ID	Entrada				Saída
	Author	Title	Journal	Year	É Válido?
1					Não
2				2000	Não
3			FAPESP		Não
4			FAPESP	2000	Não
5		Rei Leão			Não
6		Rei Leão		2000	Não
7		Rei Leão	FAPESP		Não
8		Rei Leão	FAPESP	2000	Não
9	Amanda				Não
10	Amanda			2000	Não
11	Amanda		FAPESP		Não
12	Amanda		FAPESP	2000	Não
13	Amanda	Rei Leão			Não
14	Amanda	Rei Leão		2000	Não
15	Amanda	Rei Leão	FAPESP		Não
16	Amanda	Rei Leão	FAPESP	2000	Sim

2.2 Inserção de item bibliográfico - *Book*

De forma análoga à inserção de item bibliográfico para *Article*, foi construída uma tabela de testes para *Book* que resultou na figura 6:

As diferenças entre *Book* e *Article* são a quantidade e os campos de entrada. Ainda assim, para que o *JabRef* resulte em saída válida, basta que todos os cinco campos de entrada - *Title*, *Publisher*, *Year*, *Author* e *Editor* sejam preenchidos com ao menos um caracter.

Figure 6: Tabela de testes funcionais para inserção em *Book*

ID	Entrada					Saída
	Title	Publisher	Year	Author	Editor	É Válido?
1						Não
2					Matheus	Não
3				Joe		Não
4				Joe	Matheus	Não
5			1997			Não
6			1997		Matheus	Não
7			1997	Joe		Não
8			1997	Joe	Matheus	Não
9		Globo				Não
10		Globo			Matheus	Não
11		Globo		Joe		Não
12		Globo		Joe	Matheus	Não
13		Globo	1997			Não
14		Globo	1997		Matheus	Não
15		Globo	1997	Joe		Não
16		Globo	1997	Joe	Matheus	Não
17	Walter Mity					Não
18	Walter Mity				Matheus	Não
19	Walter Mity			Joe		Não
20	Walter Mity			Joe	Matheus	Não
21	Walter Mity		1997			Não
22	Walter Mity		1997		Matheus	Não
23	Walter Mity		1997	Joe		Não
24	Walter Mity		1997	Joe	Matheus	Não
25	Walter Mity	Globo				Não
26	Walter Mity	Globo			Matheus	Não
27	Walter Mity	Globo		Joe		Não
28	Walter Mity	Globo		Joe	Matheus	Não

29	Walter Mity	Globo	1997			Não
30	Walter Mity	Globo	1997		Matheus	Não
31	Walter Mity	Globo	1997	Joe		Não
32	Walter Mity	Globo	1997	Joe	Matheus	Sim

2.3 Importação de itens bibliográficos

Os testes manuais realizados a respeito da importação de itens bibliográficos incluíram tanto a modificação do formato do arquivo a ser importado quanto a sintaxe deste, válida ou não.

Para arquivos *.bibtex* com sintaxe válida, como mostrado na figura 7, a importação ocorre com sucesso.

Figure 7: Importação válida

```

1 @Article{Amanda2000,
2   author   = {Amanda},
3   title    = {Rei Leão},
4   journal  = {FAPESP},
5   year     = {2000},
6 }

```

Ao testar a importação de arquivo *.bibtex* com sintaxe inválida, mais especificamente com excesso de }, obteve-se sucesso ao importar o arquivo. O próprio *JabRef* desconsiderou o } a mais.

Figure 8: Importação com sucesso e sintaxe inválida

```

1 @Article{Amanda2000,
2   author   = {Amanda},
3   title    = {Rei Leão},
4   journal  = {FAPESP},
5   year     = {2000}},
6 }

```

Figure 9: Resultado no *JabRef*

#	entrytype	author/editor	title	year	journal/booktitle	bibtexkey
1	Article	Amanda	Rei Leão	2000	FAPESP	Amanda2000

No entanto, a importação de arquivo *.bibtex* com falta de }, como mostrado na figura 8, resulta em falha.

Figure 10: Importação com erro e sintaxe inválida

```

1 @Article{Amanda2000,
2   author   = {Amanda},
3   title    = {Rei Leão},
4   journal  = {FAPESP},
5   year     = {2000,
6 }

```

Não é possível importar arquivos com extensão inválida, como é o caso dos arquivos *.csv*.

Portanto, obteve-se a tabela representada pela figura 11:

Figure 11: Tabela de testes funcionais para importação de itens bibliográficos

ID	Entrada			Saída	
	Nome do arquivo	Formato do arquivo	Tem sintaxe válida?	Observações	Importável?
1	1.csv	csv	Sim	-	Não
2	2.csv	csv	Não	-	Não
3	3.bibtex	BibTeX	Sim	-	Sim
4	4.bibtex	BibTeX	Não	Excesso de {}	Sim
5	5.bibtex	BibTeX	Não	Falta de {}	Não

3 Conclusão

No momento de conclusão deste relatório, apenas o primeiro *Sprint* foi finalizado. Foram criadas as tabelas de teste manuais para as funcionalidades de inserção de item bibliográfico para *Article*, de inserção de item bibliográfico para *Book* e de importação de itens bibliográficos.

As próximas etapas do projeto envolvem a manutenção de inserção de modo a validar ano e *Bibtexkey*, além de adicionar a funcionalidade para importação de *csv*. Por último, serão efetuados os testes das mesmas funcionalidades. O grupo estudará o código para entender como melhor realizar as manutenções necessárias. Também será estudada a possibilidade de automatizar os testes para o relatório final.

Vale ressaltar que as tabelas de testes funcionais não são definitivas e podem sofrer alterações ao longo das próximas etapas do projeto.