Aplicação da Ferramenta Google Colaboratory para o Ensino da Linguagem Python

Martony Demes da Silva [1]

[1] Departamento de Ciência da Computação – Universidade Federal do Piauí (UFPI)
Teresina – PI – Brasil

Abstract. *This work presents the application of the Google Collaboratory tool (or colab) in the teaching of Python programming language. Through this environment, it is possible for student interaction in the practice of programming activities in an easy and deductive way. To validate this ease of use, an assessment was made using the Perceived Utility and Ease of Use perceived in this tool. The results showed that the two analyzes had rates above 90%. In this context, the platform is easy to use and effortless*

Resumo. *Esse trabalho apresenta a aplicação da ferramenta Google Colaboratory (ou colab) no ensino de linguagem de programação Python. Por meio desse ambiente, é possível a interação do estudante na prática de atividades de programação de maneira fácil e dedutiva. Para validar essa facilidade de uso, fez-se uma avaliação por meio da Utilidade Percebida e Facilidade de Uso percebida nesta ferramenta. Os resultados apontaram que as duas análise tiveram índices superiores a 90%. Nesse contexto, a plataforma é de fácil utilização e sem necessidade de esforço*

1. Introdução

A ferramenta digital *Google Colaboratory* ou *Colab* é um ambiente digital de acesso aberto e disponibilizado pela *Google* para aplicação de conhecimentos de programação na linguagem *Python* [Carneiro et al. 2018]. Por meio dos recursos providos, a *Colab* facilita práticas de programação. De tal maneira, utiliza-se essa ferramenta em outras disciplinas: matemática [Access 2020] e medicina [Balaraman 2020].

O ensino de programação ainda é um desafio no contexto da computação [Neto and Schuvartz 2007], [Amaral et al. 2017] e [Blatt et al. 2017]. Entre os motivos tem-se a questão de estudantes iniciantes precisam adquirir habilidades como a abstração e o raciocínio lógico. Diante disso, o professor precisa motivar os alunos pelo interesse na disciplina.

Por outro lado, o professor não consegue realizar um atendimento individualizado. Isso se deve ao grande número de alunos e reduzida disponibilidade de tempo. Dado esse desafio, as disciplinas de programação ainda tem um alto índice de reprovação em programação [Amaral et al. 2017].

Diante do contexto de Engenharia de Software, verificou-se que o uso de ferramentas colaborativas e integradas propiciam novas oportunidades ao processo de ensino-aprendizagem na computação. E práticas com novas abordagens no âmbito educacional, como planejamento integrado e uso de funcionalidades específicas, permitem executar as tarefas na internet de modo colaborativo [Abegg et al. 2010].

Nesse cenário, esta pesquisa objetiva avaliar a ferramenta colaborativa e online *google colab* aplicada no ensino de programação em *Python*. O uso da ferramenta favorece a interação do aluno na prática por meio dos recursos facilitadores da plataforma *colab*. No contexto virtual, a ferramenta permite que vários alunos interajam ao mesmo tempo ou no mesmo arquivo. A validação desta pesquisa foi feita por meio de uma avaliação com estudantes do curso de *Python* na modalidade a distância (EaD).

Vale ressaltar que a motivação na escolha da *colab* se deve por ser uma ferramenta integrada da plataforma da google e possui maior disponibilidade de recursos comparado-se com outras tecnologias similares como a *jupyter*. Essas tais características são: bibliotecas pre-instaladas, uso de recursos da máquina virtual da *google*, acessado de qualquer lugar entre outras que serão especificadas na Seção 3.

Este trabalho está organizado da seguinte forma: na Seção 1 abordou sobre os trabalhos relacionados. A Seção 3 apresenta a ferramenta desta pesquisa. A Seção 4 detalha a metodologia da pesquisa. A Seção 5 ilustra os resultados da pesquisa e a Seção 6 finaliza com as considerações finais.

2. Referencial Teórico

Conforme já exposto, o ensino de programação é um grande desafio em cursos técnico e de graduação. Para tanto, alguns métodos e ferramentas tem sido propostos para facilitar o processo de ensino aprendizagem.

Um trabalho relevante foi feito por [Blatt et al. 2017]. Nessa pesquisa, fez-se um mapeamento sistemático da literatura sobre metodologias e ferramentas para apoio ao ensino de programação. Entre os resultados apresentados, tem-se a preferência dos professores em utilizar *Scratch* como ferramenta de apoio. O *Scratch* possibilita que estudantes (inclusive crianças) elaborem blocos visuais por meio de animações, histórias interativas ou jogos.

Outra alternativa para facilitar o ensino de programação é o auxilio de mecanismos de inteligencia artificial (IA) por meio de tutores inteligentes[Neto and Schuvartz 2007]. Essa pesquisa propôs um ambiente ativo, com suporte aos estudantes e auxiliando nas dificuldades encontradas.

No mesmo contexto teórico relacionado, pesquisas propuseram e validaram o uso de ferramenta de apoio ao processo de ensino e aprendizagem de lógica e programação para iniciantes como em [Amaral et al. 2017]. Com objetivo semelhante, a pesquisa de [Farias et al. 2015] propõe um ambiente que facilita a interação e comunicação de estudantes iniciantes e professor.

Por fim, outra proposta relevante é a apresentada por [Torezani and de Lira Tavares 2014]. Nesta pesquisa, é validado um editor para desenvolver atividades de aprendizagem de programação para crianças. Os resultados mostraram que foi possível elaborar atividades adequadas ao perfil dos estudantes.

No âmbito desse tema, à luz da engenharia de software, não encontrou-se trabalhos com uso de ferramentas aplicadas no ensino de Python e ferramentas colaborativas e onlines. As abordagens com uso da ferramenta *Google colab* encontradas são restritas a comparação com hardware real [Carneiro et al. 2018], a comparação de algoritmos de classificação [Balaraman 2020] e em aplicação em estudos de matemática

[Alves and Machado Vieira 2019] e [Access 2020].

3. Ambiente Google Colaboratory - Colab

O *Google Colab* ou "*Colaboratório*" é um ambiente digital disponível na nuvem, gratuito e hospedado pelo Google. O objetivo desta ferramenta é prover serviços de desenvolvimento em python. O ambiente é uma máquina virtual, denomidada de notebook, em que o usuário desenvolver práticas de programação em python. Ao mesmo tempo que programa, o usuário pode fazer comentário, compilar, criar relatório, texto, tudo junto no mesmo trabalho.

As vantagens do colab são:

1. Suporte para linguagem Python;
2. Aceleração de GPU grátis;
3. Bibliotecas pré-instaladas: Todas as principais bibliotecas Python, como o TensorFlow, o Scikit-learn, o Matplotlib, entre muitas outras, estão pré-instaladas e prontas para serem importadas [Vishakha Lall 2018];
4. Construído com base no Jupyter Notebook;
5. Recurso de colaboração (funciona com uma equipe igual ao Google Docs): permite que os desenvolvedores usem e compartilhem o Jupyter notebook entre si sem precisar baixar, instalar ou executar qualquer coisa que não seja um navegador;
6. Suporte a comandos bash;
7. Os notebooks do Google Colab são armazenados no drive.

Uma particularidade do *colab* é o mecanismo de dividir blocos de textos e de código no mesmo arquivo. Com isso, é possível facilitar os comentários do código, a compilação do código modularizada (separada), elaboração simultânea de relatório, clareza e organização dos estudos. Essas possibilidades favorecem o ensino de programação. A Figura 1, a seguir, ilustra a segregação de bloco de código e bloco de texto.

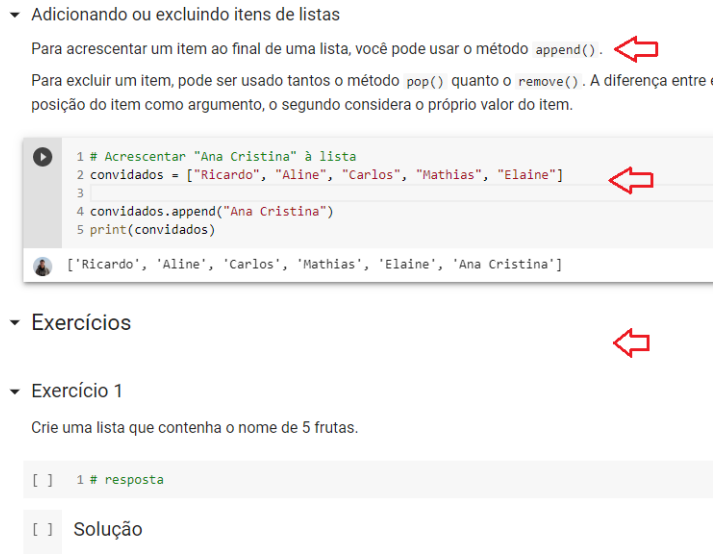
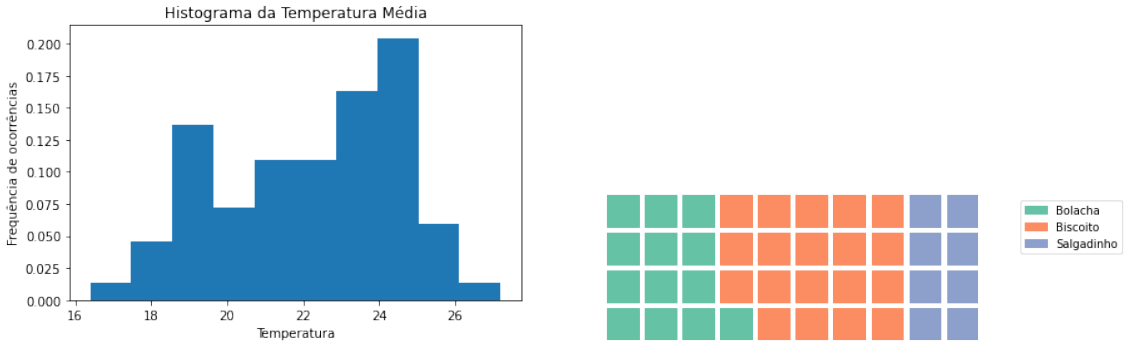


Figura 1. Separação de códigos, textos e atividades

Entre as funcionalidades do *colab*, existe a possibilidade de elaboração de gráficos e tabelas diretamente na ferramenta. A Figura 2 ilustra exemplos de gráficos criados no *colab*. A compilação é modular, sem afetar outras partes



(a) Gráfico tipo histograma

(b) Gráfico do tipo Waffle

Figura 2. Exemplos de gráficos no colab

Dada as diversas funcionadas, o *colab* apresenta ainda uma performance compatível com servidores reais[Carneiro et al. 2018], em um ambiente controlado. Como apresentado, *colab* pode ser utilizado por qualquer estudante mesmo que não tenha experiencia com programação. Na Seção 4 será validado essa facilidade por meio de uma avaliação realizada no *colab* com estudantes do curso EaD de *Python*. Importante frisar que alguns encontros do curso são online (em tempo real) do professor com os alunos.

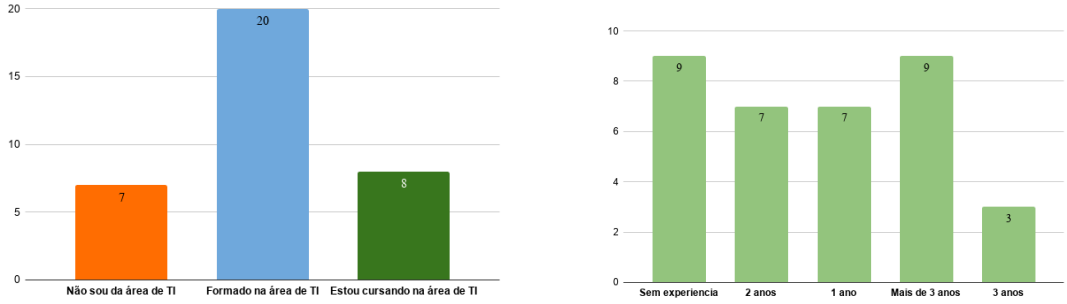
4. Metodologia

Para validar a flexibilidade e facilidade do ambiente em pesquisa, elaborou-se uma avaliação baseado em um experimento quantitativo com alunos do curso de online *Python*. A descrição do experimento será apresentado a seguir.

4.1. Participantes

Os participantes do experimento foram **35 estudantes** de um curso online de *Python*. Inicialmente, foram levantadas as seguintes informações dos envolvidos: a idade, experiência com programação e formação na área de TI.

Em uma análise preliminar dos participantes, a partir do resultado exposto na Figura 3, tem-se as seguintes inferências: dos 35 alunos, 7 não são da área de Tecnologia da Informação. Além disso, 57% dos estudantes já são formados na área de TI. Outro dado relevante é que cerca de 66% dos participantes não têm nenhuma experiencia ou somente até 2 anos de experiência com programação.



(a) Formação na área de TI

(b) Experiência com programação

Figura 3. Informações sobre os participantes

4.2. Procedimentos

O experimento foi realizado por meio de um questionário online com 12 questões, ilustradas na Figura 2. Cada questão tem como opções de resposta escores de 1 a 7, conforme o grau de resposta dada por cada participante. O ambiente digital de referência é o *colab*, utilizado no curso de online de programação em *Python*. Os participantes responderam o questionário com base em suas experiências de práticas transcorridas no curso.

Vale frisar que o questionário foi aplicado no período em que os alunos estavam com dois meses de curso. O quantitativo de participantes foi limitado pelo total de estudantes à época no curso

Como citado, a investigação foi guiada por meio do questionário online. Cada questão segue o método da escala Likert [Wainerman 1976] e a metodologia *Technology Acceptance Model (TAM)* [Davis et al. 1989]. TAM apresenta as intenções de uso de um sistema. Essas intenções são fundamentadas em dois aspectos: **Utilidade Percebida (UP)**, que mede o nível em que o indivíduo acredita que o uso do sistema pode melhorar suas atividades; e **Facilidade de Uso Percebida (FUP)**, que mede o grau em que o usuário acredita que o uso do sistema de informação livre de esforço.

Como citado, a escala Likert verifica o nível de concordância do sujeito com várias afirmações que expressam algo favorável ou desfavorável em relação a um objeto psicológico. Com base na abordagem de [Neto et al. 2018], os participantes da pesquisa escolhem entre opções e marcam a resposta conforme sua atitude ou opinião. As opções de respostas disponíveis para cada questão, nesta pesquisa, são: Discordância Total (DT); Discordância Parcial (DP); Discordância Leve (DL); Neutro (N); Concordância Leve (CL); Concordância Parcial (CP); Concordância Total (CT).

Apos aplicação do questionário aos 35 participantes, as respostas foram avaliadas levando em consideração dois pontos [Macnaughton 1996]: quantidade de concordantes (ConP) e Discordantes (DisP). ConP é o calculo das soma entre a quantidade de respostas totalmente concordantes e parcialmente concordantes (Equação 1). Já o DisP é calculado pela soma entre a quantidade de respostas totalmente discordantes e parcialmente discordantes (Equação 2)

$$ConP = CT + CP + CL + \frac{N}{2} \quad (1) \qquad DisP = DT + DP + DL + \frac{N}{2} \quad (2)$$

Com base no resultado de ConP e DisP, calcula-se o grau de concordância de cada proposição (GCP), pela Equação 3, com base na abordagem de [Wilder 1978].

$$GCP = 100 - \left(\frac{100}{\frac{ConP}{Disp} + 1} \right) \tag{3}$$

Diante dos resultados do GCP, utiliza-se a Tabela 1 de referência de valores de GCP a qual indica o quanto o participante concorda ou discorda sobre cada questão [Davis et al. 1989].

Tabela 1. Referência de valores do GCP

Valor de GCP	Frase adequada
90 ou mais	Uma concordância muito forte
80 a + 89,99	Uma concordância substancial
70 a + 79,99	Uma concordância moderada
60 a + 69,99	Uma concordância baixa
50 a + 59,99	Uma concordância desprezível
40 a + 49,99	Uma discordância desprezível
30 a + 39,99	Uma discordância baixa
20 a + 29,99	Uma discordância moderada
10 a + 19,99	Uma discordância substancial
9,99 ou menos	Uma discordância muito forte

5. Resultados

O produto resultante do questionário apresenta dois aspectos: a **UP**, que constitui 6 proposições e a **FUP**, compondo também 6 proposições. A Tabela 2 ilustra as questões definidas para esta pesquisa.

Tabela 2. Proposições da Pesquisa

Utilidade Percebida (UP)	
nº	**Questão**
1	A utilização do *Colab* é importante para o meu aprendizado
2	As funcionalidades do *Colab* oferecem condições semelhantes a um ambiente de IDE (Ambiente de desenvolvimento Integrado)
3	O *Colab* facilita meu entendimento de assuntos passados em aula
4	Usar o *Colab* agrega valor ao meu aprendizado em *Python*
5	As universidades deveriam utilizar o *Colab* para treinamento de alunos de curso de computação
6	Estou motivado a continuar usando o *Colab*
Facilidade de Uso Percebida (FUP)	
7	No *Colab* eu sempre sei onde estou e como chegar onde quero chegar
8	O *Colab* tem uma interação compreensível e clara
9	Os recursos de programação e navegação são fáceis de encontrar
10	O *Colab* possui uma boa amigabilidade (fácil de aprender a usar)
11	Consigo utilizar o *Colab* sem auxílio de um instrutor
12	Utilizar o *Colab* é agradável

Com base nas respostas das questões da pesquisa dispostas na Tabela 2, calculou-se os valores das respostas de 35 participantes da pesquisa. O resultado é apresentado na Tabela 3.

Ao examinar o quadro de **Utilidade Percebida**, da Tabela 3, verifica-se que todas as seis questões tem *Uma concordância muito forte*, com índice GCP acima de 90%. Isso, tomando como parâmetro a Tabela 1. A questão 2 apresenta um valor relativamente abaixo. Isso é considerável pois a pergunta faz um comparativo com IDE. Diante desses resultados, pode-se sustentar que o *Colab* apresenta uma boa utilidade percebida pelos estudantes.

Os resultados para a **Facilidade de Uso Percebida** são equivalente ao UP. A FUP avalia o grau em que o usuário acredita que o uso da ferramenta é livre de esforço.Na Tabela 3 é apresentado, em todas as questões, valores GCP superior a 90%. O resultado reproduz que o *Colab* é de fácil utilização e sem necessidade de esforço - ”Uma concordância muito forte”, descrito na Tabela 1.

Tabela 3. Resultados da Pesquisa

Utilidade percebida											
Questão	DT	DP	DL	N	CL	CP	CT	QTR	DisP	ConP	GCP
1	0	0	0	1	2	5	27	35	0,5	34,5	98,57
2	0	0	2	3	6	10	14	35	3,5	31,5	90,0
3	0	0	0	2	4	5	24	35	1	34	97,14
4	0	0	0	1	0	6	28	35	0,5	34,5	98,57
5	0	1	0	0	1	4	29	35	1	34	97,14
6	0	0	0	1	2	8	24	35	0,5	34,5	98,57
Facilidade de uso percebida											
Questão	DT	DP	DL	N	CL	CP	CT	QTR	DisP	ConP	GCP
7	0	0	0	2	2	7	24	35	1	34	97,14
8	0	0	0	1	2	5	27	35	0,5	34,5	98,57
9	0	0	0	3	2	9	21	35	1,5	33,5	95,71
10	0	0	0	2	7	6	20	35	1	34	97,14
11	0	0	0	3	1	10	21	35	1,5	33,5	95,71
12	0	0	0	1	3	9	22	35	0,5	34,5	98,57

Conforme exposto, não encontrou-se estudos com uso de ferramentas computacionais colaborativas no ensino de linguagem de programação *Python*. Diante dessa lacuna, este trabalho apresentou resultados significativo no uso da ferramenta *Colab*, destacando a facilidade de uso.

6. Considerações Finais

Este trabalho apresentou a o ambiente colaborativo *Google Colab* como ferramenta de apoio ao ensino de programação Python. No âmbito da engenharia de software e com base nos resultados apresentados, na Seção 5, o uso desta ferramenta propicia a prática de programação e reduz a complexidade que surgem durante aprendizagem de programação.

O diferencial ou ganho computacional apresentado nesta pesquisa é a alternativa de uso de uma ferramenta que facilita o ensino de programação em *Python* aos estudantes de computação.

Nesse âmbito, o estudante de programação pode concentra-se dividir seus estudos em módulos organizados e acompanhado pelo professor colaborativamente. Para trabalhos futuros, pretende-se realizar uma avaliação comparativa de performance de processamento do *colab* com outras ferramentas IDE.

Referências

[Abegg et al. 2010] Abegg, I., Bastos, F. d. P. d., and Müller, F. M. (2010). Ensino-aprendizagem colaborativo mediado pelo wiki do moodle. *Educar em Revista*, 1(38):205–218.

[Access 2020] Access, O. (2020). Visualizing the Newtons Fractal from the Recurring Linear Sequence with Google Colab : An Example of Brazil X Portugal Research. *International Electronic Journal of Mathematics Education*, 15(3).

[Alves and Machado Vieira 2019] Alves, F. R. V. and Machado Vieira, R. P. (2019). The Newton Fractal's Leonardo Sequence Study with the Google Colab. *International Electronic Journal of Mathematics Education*, 15(2).

[Amaral et al. 2017] Amaral, E., Camargo, A., Gomes, M., Richa, C. H., and Becker, L. (2017). ALGO+ Uma ferramenta para o apoio ao ensino de Algoritmos e Programação para alunos iniciantes. *Anais do XXVIII Simpósio Brasileiro de Informática na Educação (SBIE 2017)*, 1(Cbie):1677.

[Balaraman 2020] Balaraman, S. (2020). Comparison of Classification Models for Breast Cancer Identification using Google Colab. *Preprints*, pages 1–12.

[Blatt et al. 2017] Blatt, L., Becker, V., and Ferreira, A. (2017). Mapeamento Sistemático sobre Metodologias e Ferramentas de apoio para o Ensino de Programação. *Anais do XXIII Workshop de Informática na Escola (WIE 2017)*, 1(Cbie):815.

[Carneiro et al. 2018] Carneiro, T., Da Nobrega, R. V. M., Nepomuceno, T., Bian, G. B., De Albuquerque, V. H. C., and Filho, P. P. R. (2018). Performance Analysis of Google Colaboratory as a Tool for Accelerating Deep Learning Applications. *IEEE Access*, 6:61677–61685.

[Carneiro et al. 2018] Carneiro, T., Medeiros Da NóBrega, R. V., Nepomuceno, T., Bian, G., De Albuquerque, V. H. C., and Filho, P. P. R. (2018). Performance analysis of google colaboratory as a tool for accelerating deep learning applications. *IEEE Access*, 6:61677–61685.

[Davis et al. 1989] Davis, F. D., Bagozzi, R. P., and Warshaw, P. R. (1989). User acceptance of computer technology: a comparison of two theoretical models. *Management science*, 35(8):982–1003.

[Farias et al. 2015] Farias, H., Bonifácio, B., and Ferreira, R. (2015). Avaliando o Uso da Ferramenta Scratch para Ensino de Programação através de Análise Quantitativa e Qualitativa. *Anais do XXVI Simpósio Brasileiro de Informática na Educação (SBIE 2015)*, 1(Sbie):947.

[Macnaughton 1996] Macnaughton, R. J. (1996). Numbers, scales, and qualitative research. *The Lancet*, 347(9008):1099–1100.

[Neto et al. 2018] Neto, A. S., Neto, F. M., Lima, R., Silva, S., and de Oliveira, E. J. (2018). Avaliação de um ambiente virtual gamificado para auxiliar o ensino-aprendizagem de estudantes de medicina. In *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*, volume 29, page 496.

[Neto and Schuvartz 2007] Neto, W. C. B. and Schuvartz, A. A. (2007). Ferramenta Computacional de Apoio ao Processo de Ensino-Aprendizagem dos Fundamentos de Programação de Computadores. *SBIE - Simpósio Brasileiro de Informática na Educação*, 17:520–528.

[Torezani and de Lira Tavares 2014] Torezani, C. and de Lira Tavares, O. (2014). Eanewprog-um editor de atividades para o ambiente online newprog. In *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*, volume 25, page 422.

[Vishakha Lall 2018] Vishakha Lall (2018). Google colab — the beginner's guide. Access date: 1 jun. 2020.

[Wainerman 1976] Wainerman, C. (1976). *Escalas de medición en ciencias sociales*. Ediciones Nueva Visión.

[Wilder 1978] Wilder, J. W. (1978). *New concepts in technical trading systems*. Trend Research.