

[white paper]

Diamond Open Access

[awaiting peer review]

Sistema Digital Universitário

Colaboração Aberta¹

10 de Setembro de 2022

Resumo

Apresentamos uma proposta de plataforma digital para gestão de processos universitários.

palavras-chave: gestão universitária, plataforma digital

A versão mais atualizada deste artigo está disponível em

<https://osf.io/3h8wz/download>
<https://zenodo.org/record/7066838>

Introdução

1. É uma questão estratégica investir em desenvolvimento de software.
2. O símbolo $:=$ significa que o que está à esquerda está sendo definido pelo que aparece à direita.
3. SIDU $:=$ Sistema Digital Universitário
4. Este white paper é *atualizado* em **fluxo contínuo**.
5. Inicialmente, discutiremos apenas os serviços e processos relacionados à matrícula dos estudantes na universidade e nas disciplinas.
6. Posteriormente, outros processos e serviços serão incluídos de forma modular.

¹Todos os autores com suas afiliações aparecem no final deste artigo.

Estrutura

7. O SIDU está dividido em módulos.
8. Cada módulo contempla uma categoria de serviços e/ou processos.

Módulos

9. M_i := i -ésimo módulo referente à matrícula
10. M_1 := cadastro de novos alunos na universidade
11. M_2 := cadastro e oferta de disciplinas
12. M_3 := matrícula dos alunos nas disciplinas de cada período letivo

Layout da plataforma

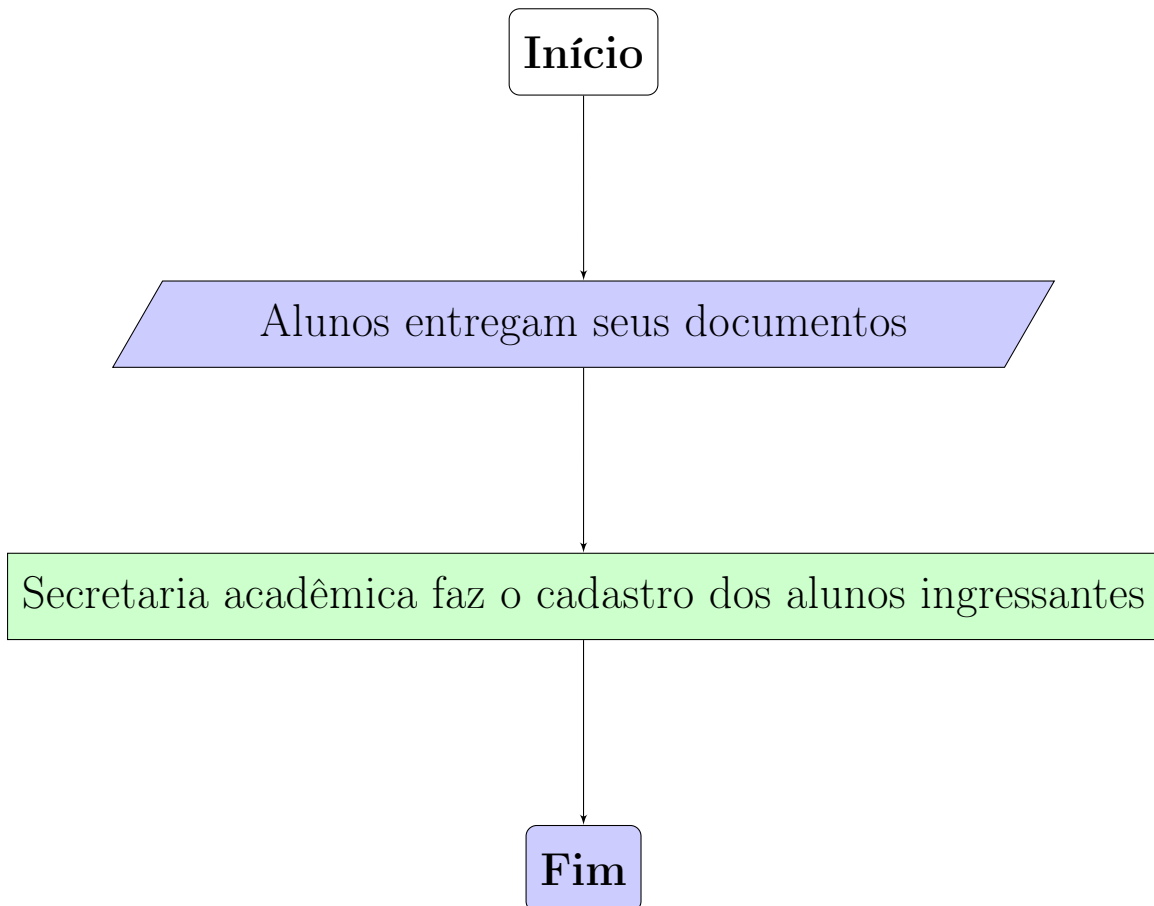
13. Seguiremos o princípio minimalista, onde *menos é mais* [1,2].

S0: Servidor e Sistema

14. Este módulo contempla duas etapas,
 - (a) Configuração do servidor (gerenciamento de portas e configuração do IP);
 - (b) Implantação do sistema por meio das tecnologias **Flask**, **NGINX**, **Contêiner Docker**, **Kubernetes**, entre outras que se fizerem necessárias.
15. $S0a$:= configuração do servidor
16. $S0b$:= implantação do sistema

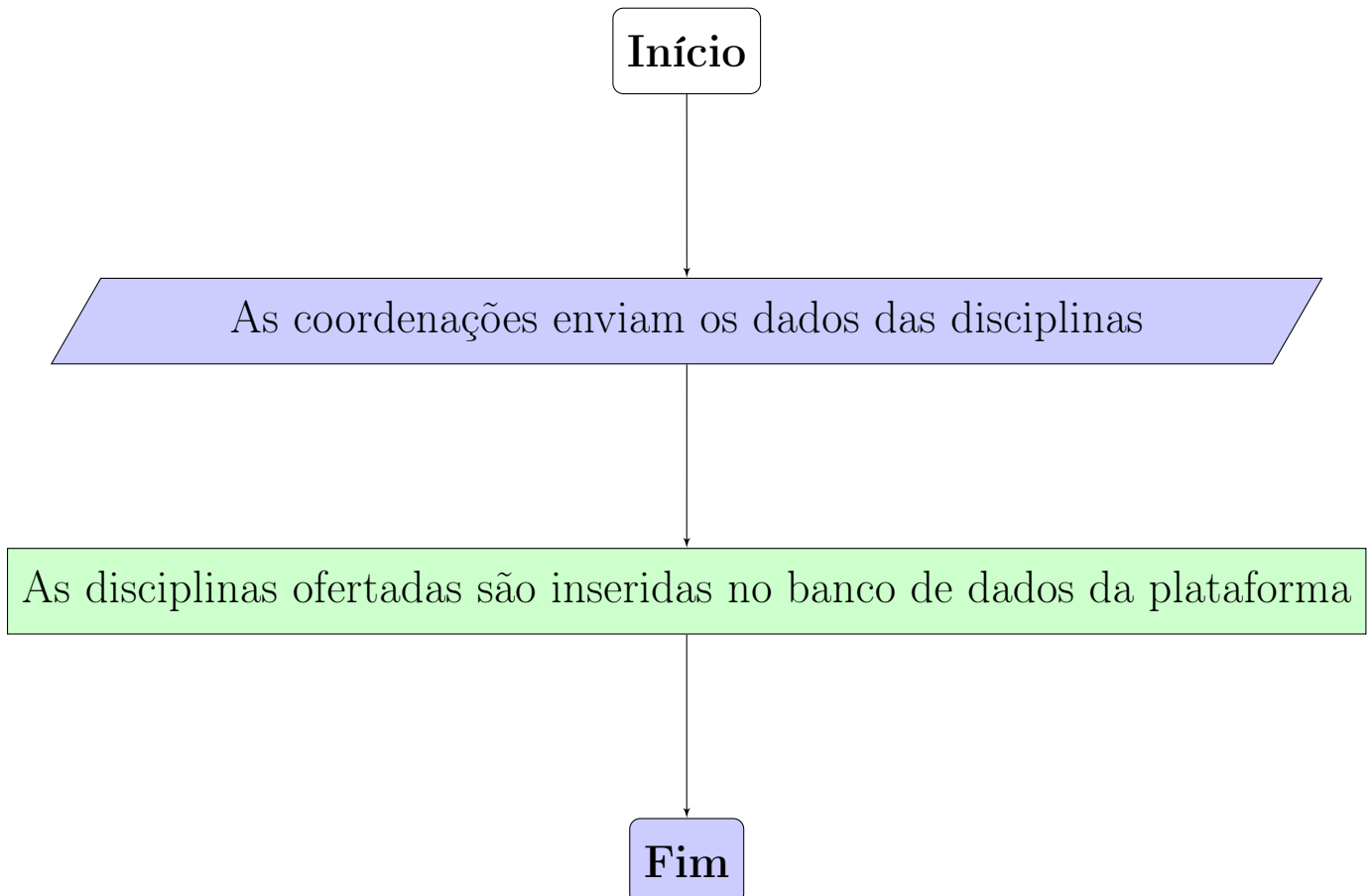
M1: Gerenciador de Estudantes

17. O módulo M1 realiza o cadastro dos alunos ingressantes na universidade.
18. Os parâmetros envolvidos na programação incluem d_1, d_2, d_3, \dots , sendo d_i o i -ésimo tipo de dado relevante do aluno.
19. Por exemplo,
 $d_1 :=$ nome do aluno,
 $d_2 :=$ data de entrada na universidade,
 $d_3 :=$ número de matrícula etc.
20. O fluxograma abaixo descreve, de forma resumida, alguns processos de M1.



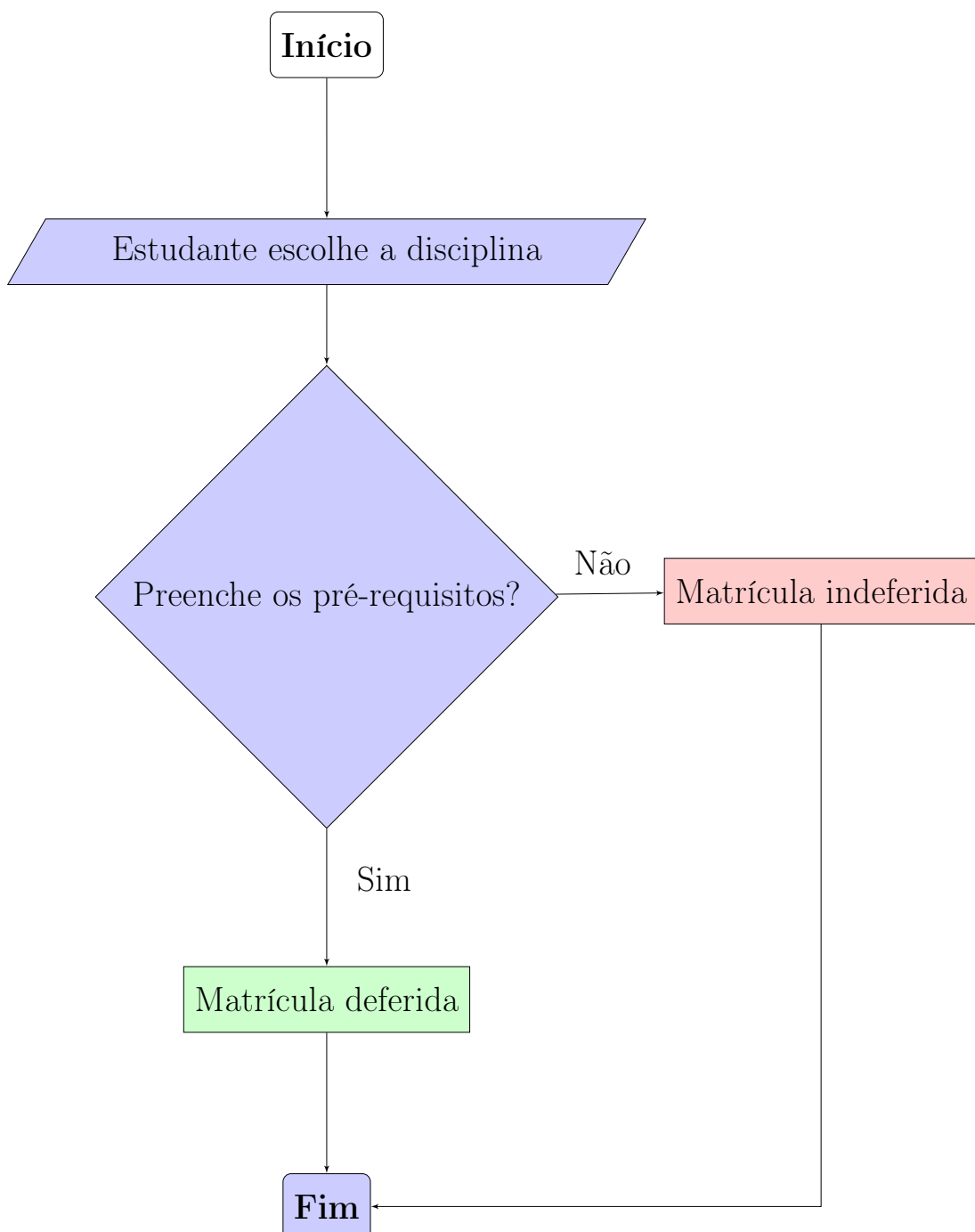
M2: Gerenciador de Disciplinas

21. O módulo M2 inclui cadastro, oferta e renovação das **disciplinas** em determinado *período letivo*.
22. A renovação da disciplina pode ser feita de forma automática pelo sistema, mas deve ser validada ou não por uma pessoa.
23. A seguir estão listados alguns parâmetros importantes na programação deste módulo.
24. `disciplina := {nome, id, dia da semana, horário da aula, pré-requisitos, ...}`
25. `id` := identificador único da disciplina
26. O fluxograma abaixo descreve alguns processos de M2.



M3: Gerenciador de Matrícula

27. A matrícula em uma disciplina envolve a verificação de alguns parâmetros (como dia e horário, pré-requisitos etc) que poderão validar ou não a inscrição do estudante na disciplina.
28. O fluxograma a seguir descreve um dos processos (recorrentes) de verificação de M3 para fins de matrícula dos estudantes nas disciplinas.



Linguagem de programação

29. Python [3–5]

Carga horária

30. Na tabela a seguir, há uma *estimativa* de horas e semanas que um programador gastaria para produzir cada um dos módulos na forma como foram apresentados neste documento.

M0	M1	M2	M3
<i>em estudo</i>	12 h = 0,6 semana	40 h = 2 semanas	360 h = 18 semanas

31. Os valores da tabela representam uma estimativa inicial, à medida que o projeto for avançando, esses números poderão variar para mais ou para menos.
32. Consideramos um(a) programador(a) trabalhando 20 horas semanais.

Próximas ações

33. Descrever o algoritmo dos módulos M1, M2 e M3.
34. Escrever o script em python (comentado) para cada um dos módulos apresentados.

Considerações Finais

35. Esta é uma proposta preliminar de uma plataforma digital para oferta e matrícula dos estudantes nas disciplinas de seus respectivos cursos do ensino superior.

Arquivos Suplementares

O **arquivo latex** para este artigo, juntamente com outros *arquivos suplementares*, estão disponíveis em [6, 7]. Seja coautor(a) deste artigo, envie sua contribuição para `mplobo@uft.edu.br`.

Consentimento

O autor concorda com [8].

Como citar este artigo?

<https://doi.org/10.31219/osf.io/3h8wz>

<https://zenodo.org/record/7066838>

Licença

CC-By Attribution 4.0 International [9]

Referências

- [1] McKEOWN, Greg. *Essencialismo: a disciplinada busca por menos*. Sextante, 2015.
- [2] Jay, Francine. *Menos é mais: um guia minimalista para organizar e simplificar sua vida*. Editora Schwarcz-Companhia das Letras, 2016.
- [3] Python. <https://www.python.org>
- [4] SWEIGART, Al. *Automatize tarefas maçantes com Python*. São Paulo: Novatec (2015).
- [5] Lobo, Matheus P. “Caderno Didático Em Python.” *OSF Preprints*, 22 Dec. 2020. <https://doi.org/10.31219/osf.io/t6x8p>

- [6] Lobo, Matheus P. “Open Journal of Mathematics and Physics (OJMP).” *OSF*, 21 Apr. 2020.
<https://doi.org/10.17605/osf.io/6hzyp>
- [7] <https://zenodo.org/record/7066838>
- [8] Lobo, Matheus P. “Simple Guidelines for Authors: Open Journal of Mathematics and Physics.” *OSF Preprints*, 15 Nov. 2019.
<https://doi.org/10.31219/osf.io/fk836>
- [9] CC. Creative Commons. *Attribution 4.0 International* (CC BY 4.0)
<https://creativecommons.org/licenses/by/4.0>

Colaboração Aberta

Matheus Pereira Lobo¹ (autor principal, mplobo@uft.edu.br)
<https://orcid.org/0000-0003-4554-1372>

¹Universidade Federal do Norte do Tocantins