

# Analizando a Morte de Ecossistemas de Software para plataformas Web a partir da abordagem de Redes Sociais e Reciclagem de Recursos

Pedro Arantes<sup>1</sup>, Awdren Fontão<sup>1</sup>

<sup>1</sup>Faculdade de Computação – Universidade Federal do Mato Grosso do Sul (UFMS)  
Caixa Postal 79.002 – 79.070-900 – Campo Grande – MS – Brazil

{pedro.arantes, awdren.fontao}@ufms.br

**Abstract.** *Software Ecosystems (SECO) depend on platforms that serve as environments for developers' interaction, as the StackOverflow and the GitHub. If the keystone does not support the synergy between organizational goals and developers' expectations, the SECO may die. The death results from suspending vital activities, such as software evolution and maintenance. When a SECO dies, the corporation loses resources invested, and the developers lose work, learning, and experience gained. This paper reports a GitHub exploratory case study focusing on three web SECOs: AngularJS, PhantomJS, and MomentJS. We analyze metrics based on developer community engagement, collaboration, and resource recycling to understand what happens in these SECOs before, during, and after the platform's death.*

**Resumo.** *Ecossistemas de Software (ECOS) dependem de plataformas que apoiam a interação entre a comunidade de desenvolvedores, como o StackOverflow e o GitHub. Se a organização responsável não suportar a sinergia entre seus objetivos organizacionais e as expectativas dos desenvolvedores, o ECOS pode morrer. A morte resulta da suspensão das atividades vitais, como a evolução e manutenção do software. Quando o ECOS morre: (1) a organização perde recursos investidos, e; (2) os desenvolvedores perdem investimento de trabalho, aprendizado e experiência ganhos. Nesse artigo é relatado um estudo de caso exploratório realizado no GitHub focando em três ECOS web: AngularJS, PhantomJS, e MomentJS. Foram analisadas métricas baseadas no engajamento, colaboração da comunidade e na reciclagem de recursos para entender o que acontece nesses ECOS antes, durante e após a plataforma do ECOS morrer.*

## 1. Introdução

O ecossistema de software (ECOS) é composto por ambientes que permitem a colaboração de desenvolvedores para o desenvolvimento de uma plataforma de software e as comunidades que interagem e utilizam esses ambientes. A relevância e contínua evolução dos projetos é primordial para atrair colaboradores e engajar a comunidade de desenvolvimento para colaborar com o sistema [Manikas 2016], [Massanori et al. 2020].

Como ferramentas sociais e cooperativas, o GitHub e StackOverflow que compõem o ECOS, são úteis para a troca de recursos e informações entre seus membros [Hyrnsalmi et al. 2015]. Com o trabalho conjunto para criar, manter e evoluir os artefatos em torno do ECOS, os desenvolvedores criam relacionamentos, resultando em uma

rede de usuários interconectados. É por meio dessas redes que novos usuários conseguirão apoio para colaborar com o desenvolvimento e manutenção do sistema [Bird et al. 2008].

Se a organização detentora do ECOS não aplicar estratégias sustentáveis que visem engajar e suportar a sinergia entre os objetivos organizacionais e a comunidade de desenvolvedores, o ECOS pode morrer devido a falta de estímulo dos mantenedores [Evertse et al. 2021], [Massanori et al. 2020]. Desta forma, sem uma força de trabalho ativa, os usuários e plataformas que dependiam do ECOS estático precisarão migrar para outros ECOS, ou seja, encontrar novos ambientes que consigam suprir suas necessidades.

De acordo com [Businge et al. 2022], outro ponto a ser abordado no planejamento da migração entre ECOS é a reciclagem dos artefatos e conhecimentos adquiridos pela comunidade de desenvolvedores. Tendo em vista o alto custo relacionado ao desenvolvimento e manutenção dos ECOS, esses artefatos e conhecimentos podem ser reaproveitados em novos projetos.

A literatura foca no comportamento mortal de ECOS [Evertse et al. 2021], na morte do Windows Phone no Stack Overflow [Massanori et al. 2020], em detecção de ECOS não-mantidos [Coelho et al. 2020] e na migração de *advocates* entre ECOS [Michael Mu Sun and Kuttal 2018]. Para aprimorar o estado da arte, o objetivo deste estudo é investigar a atividade de desenvolvedores e a reciclagem de recursos para entender o fenômeno da morte de ECOS. Como resultado, aprimora-se a detecção e elaboração de estratégias de restauração ou migração de ECOS. Abaixo, são apresentadas as quatro questões de pesquisa e o resumo dos principais resultados encontrados.

- *Que tipos de artefatos fazem parte da **reciclagem de recursos** durante e após a morte do ECOS?* Arquivos de código-fonte e texto são os mais modificados antes da morte do ECOS. Entretanto, após a morte não há atividade voltada para a reciclagem de recursos.
- *O ECOS consegue manter a **atividade dos contribuidores** antes, durante e após a sua morte?* Nos primeiros anos de vida, os projetos atraem desenvolvedores mas não conseguem manter a comunidade ativa após a morte do projeto.
- *O ECOS consegue manter o **nível dos relacionamentos dos desenvolvedores** antes, durante e após a sua morte?* Usuários em média se conectam com apenas um desenvolvedor e o nível dos relacionamentos decai após a morte do projeto.
- *Os desenvolvedores do ECOS conseguem **construir relacionamentos fortes**?* Desenvolvedores constroem relacionamentos fortes com outros usuários, mas não mantêm suas conexões por um longo período. Resulta-se em uma queda constante na força dos relacionamentos após alguns anos de vida do projeto.

## 2. Fundamentação Teórica

### 2.1. Morte de ECOS

Um Ecossistema de Software (ECOS) consiste em um ambiente em torno de uma plataforma de software, que permite a interação entre atores e artefatos [Manikas 2016]. É nesse ambiente que informações e recursos podem ser trocados entre os desenvolvedores, além do desenvolvimento e manutenção dos artefatos em torno da plataforma do ECOS.

O GitHub (GH) é um exemplar de ambiente que compõe o ECOS e que permite o controle de versão, gerenciamento de mudanças e integração/implantação contínua de

projetos de software. Além disso, possui métricas relacionadas a atividade dos projetos, como a quantidade de *commits* feitos por usuário e a quantidade de arquivos modificados por *commit*, permitindo o monitoramento da atividade do ECOS [Coelho et al. 2020].

A morte do ECOS acontece por diversos fatores que levam a inércia da sua atividade. Segundo a Ecologia, um ecossistema estático é um ecossistema morto. Portanto, é definido que a morte do ECOS ocorre quando (i) há um anúncio oficial pelos mantenedores ou comunidade [Massanori et al. 2020]; ou (ii) quando o software não apresenta atividade de revisão ou aperfeiçoamento pelo período de um ano [Coelho et al. 2020].

Os ECOS possuem um alto custo de produção, pois envolvem uma comunidade de desenvolvedores, plataformas tecnológicas, conhecimentos e recursos técnicos. Vale ressaltar que alguns desses recursos podem ser reaproveitados após a morte do ECOS, sendo a atividade de reaproveitamento referida como reciclagem de recursos.

## 2.2. Redes Sociais

Uma rede social pode ser definida como um conjunto de atores e interações em um ambiente comum [Garton et al. 1997]. Pode-se utilizar como exemplo a interação de usuários em mídias digitais, relações de trabalho em uma organização e relações familiares.

O foco principal deste estudo é entender como uma comunidade de desenvolvedores trabalha em conjunto para manter e evoluir a plataforma, construindo relacionamentos e conexões entre si. A rede será apresentada como um conjunto de vértices, representando usuários, e arestas, representando as interações entre eles. A direção das interações e a força dos relacionamentos também são trabalhadas no estudo.

A partir da definição das redes, alguns comportamentos da comunidade no ECOS poderão ser verificados. São eles; (i) a criação e fortalecimento das relações entre usuários, (ii) o movimento de migração de desenvolvedores e (iii) a quantidade de usuários socialmente ativos no ECOS.

## 3. Trabalhos Relacionados

[Michael Mu Sun and Kuttal 2018] analisam o comportamento migratório de *advocates* engajados em plataformas do GitHub e StackOverflow. Os autores utilizam redes sociais para analisar a migração desse conjunto de contribuidores que desempenham um papel específico em ECOS, mas não aprofundam na criação de relacionamentos e interações entre os participantes para análise do trabalho cooperativo da comunidade.

[Coelho et al. 2020] analisam os riscos e vulnerabilidades de projetos *open source* que apresentam dificuldades para manter a atividade de manutenção e evolução da comunidade. Portanto, é proposto um modelo de aprendizado de máquina capaz de identificar repositórios sem uma comunidade ativa. Sendo assim, os autores mineraram repositórios de código para a detecção de projetos vulneráveis enquanto o estudo a seguir coleta os dados dos projetos para analisar a atividade do ECOS.

[Massanori et al. 2020] analisam a morte de ECOS a partir da perspectiva de Developer Relations (DevRel). O repositório de perguntas e respostas StackOverflow é utilizado para analisar a atividade da comunidade no ECOS Windows Phone. Os autores estudam o domínio de sistemas operacionais para aparelhos móveis enquanto o estudo a seguir investiga ECOS em torno de ferramentas de desenvolvimento Web.

[Evertse et al. 2021] analisam condutas, princípios e comportamentos adotados por ECOS que facilitaram ou resultaram na morte dos projetos. Aponta-se quais as causas resultam no desengajamento da comunidade e quais contramedidas podem ser aplicadas para restaurar o projeto. Portanto, os autores analisam as "atitudes mortais" para ECOS enquanto este estudo identifica a atividade da comunidade em torno do ECOS.

#### 4. Metodologia de Pesquisa

Para obter os dados necessários sobre as atividades da comunidade de desenvolvedores em torno do ECOS, o estudo utiliza as diretrizes de mineração de repositórios de software (MRS) de [Hemmati et al. 2013] para a coleta de dados no GitHub. Essas diretrizes apresentam as etapas adequadas para a extração e tratamento dos dados coletados.

Como um estudo subsequente a um estudo pioneiro [Soupinski and Fontao 2022], a análise foi definida utilizando conceitos de Engenharia de Software Experimental. Para diminuir a possibilidade de fatores de confusão, foram examinados projetos semelhantes entre si (contexto específico, ECOS Web) e com um único objetivo (ECOS Web mortos). Portanto a abordagem permitiu uma comparação mais significativa da morte de ECOS. Os conjuntos de dados e análises [Arantes et al. 2023] estão disponíveis para replicação e avanço na investigação de novos problemas em torno da morte de ECOS.

Outro ponto a ser levantado é o método de análise do comportamento dos ECOS durante a morte dos projetos. Para entender a discrepância da atividade da comunidade de desenvolvedores durante o período de morte, é necessário levantar qual era o seu comportamento anterior, permitindo a comparação dos períodos e a detecção da variação.

##### 4.1. Objetivo e Questões de Pesquisa

O objetivo deste estudo é analisar a morte de ECOS partindo das perspectivas de reciclagem de recursos e interações sociais. Portanto, são utilizados os indicadores: (1) Taxa de atualização de arquivos, (2) Recrutamento e permanência de desenvolvedores e (3) Colaboração entre desenvolvedores.

- Que tipos de artefatos fazem parte da reciclagem de recursos durante e após a morte do ECOS? **Motivo:** A reciclagem é importante para reduzir o impacto da morte do ECOS e preservar os seus recursos, podendo ser decompostos e reaproveitados em novos projetos. **Métrica:** Tipos de artefatos modificados no GH;
- O ECOS consegue manter a atividade dos contribuidores, antes, durante e após a sua morte? **Motivo:** A quantidade de desenvolvedores presentes no ECOS mostra a sua capacidade de engajar novos colaboradores e manter antigos usuários para colaborar e evoluir o software. **Métrica:** Quantidade de Colaboradores (Ordem);
- O ECOS consegue manter o relacionamento dos contribuidores, antes, durante e após a sua morte? **Motivo:** O número de relacionamentos presentes no ECOS mostra a capacidade da comunidade em trabalhar em conjunto para resolver problemas e criar novas features rapidamente. **Métrica:** Quantidade média de relacionamentos por colaborador (Grau Médio);
- Os desenvolvedores do ECOS conseguem construir relacionamentos fortes? **Motivo:** A força dos relacionamentos mostra a habilidade da comunidade de desenvolvimento em criar fortes canais de comunicação para desenvolver demandas complexas que requerem o trabalho conjunto. **Métrica:** Força média dos relacionamentos (Força Média).

## 4.2. Critérios de Seleção de ECOS

A seleção foi iniciada através da busca de ECOS utilizando as palavras chaves: "plataforma de software morta" e "morte de plataforma de software" no mecanismo de busca do Google. Para selecionar um projeto candidato, este deveria apresentar quatro características, sendo elas: (i) Ser *open source* e aceitar contribuições, (ii) Disponibilizar sites oficiais para verificar a morte do projeto, (iii) Apresentar o GitHub como ambiente de desenvolvimento e contribuição e (iv) Estar focado para soluções Web.

Após aplicar os critérios de seleção, foram escolhidos três ECOS, listados na Tabela 1 com as seguintes datas; (i) data de lançamento, (ii) data de descontinuação, representando a data da morte oficial do ECOS, (iii) data de início de suporte, sendo o início do período de suporte dado pela organização responsável para a comunidade encontrar e migrar para outros ECOS e (iv) data de fim de suporte.

**Tabela 1. Datas de lançamento, descontinuação e suporte**

ECOS	Lançamento	Descontinuação	Início do Suporte	Fim do Suporte
AngularJS	2009	2018/01	2018/07	2021/12
PhantomJS	2011/01	2016/01	2016/01	2018/03
Moment.js	2011	2020	2020	-

## 4.3. Extração de Dados e Modelagem de Dados

A extração dos dados do GitHub foi realizada por meio da biblioteca Pydriller<sup>1</sup>. Para analisar a reciclagem de recursos, foram coletados todos os *commits* e seus registros de modificação de arquivo. Já para analisar as interações sociais, foram coletadas as contribuições de desenvolvedores em todas *issues* e *pull requests*. Para uma melhor compreensão da reciclagem de recursos, os arquivos modificados foram classificados de acordo com [Ma et al. 2018]. As categorias e extensões são descritas na Figura 1.

Classe	Formatos
Aplicação	.bat .cmd .exe .ser .swf
Arquivo	.a .gz .jar .pack .zip
Áudio	.kt .mp3 .ogg .wav
Imagem de Disco	.scl
Fonte	.eot .otf .ttf .woff
Imagem	.blp .bmp .dds .gif .ico .jpeg .jpg .png .psd .rs .svg .tga .tif .xpm
Projeto	.csproj .pbxproj .vcproj .vcxproj
Código-fonte	.as .asm .c .cc .class .coffee .cpp .cs .cshtml .css .ctp .cxx .d .dll .ebuild .ejs .el .erb .erl .f .f90 .go .gradle .groovy .h .haml .hpp .hs .i .java .js .jsp .less .lua .m .mo .o .php .phpt .phtml .pl .pm .pp .py .pyc .r .rb .s .scala .scss .scssc .sh .smali .so .sql .swift .t .tcl .ts .vb .vim .rkt .html .json .yaml .launch
Texto	.template .ngdoc .gdoc .xml .md

**Figura 1. Atualização das classes de arquivos**

O primeiro passo para analisar as redes de interação das comunidades de desenvolvimento é definir a diferença entre relacionamentos e interações. Enviar uma mensagem para um usuário não estabelece automaticamente um relacionamento entre receptor e

<sup>1</sup><https://pydriller.readthedocs.io/>

transmissor. É necessário uma interação mútua, partindo mensagens de ambos, para que o relacionamento seja definido. Sendo assim, a interação de um usuário A para um usuário B é a participação de A em uma *issue* ou *pull request* após B. Se consequentemente houver o processo inverso, B para A, será definido o relacionamento entre eles.

Outro tópico discutido é a força dos relacionamentos. Quando dois usuários trabalham conjuntamente em uma demanda, seu relacionamento é fortificado. Pode-se então utilizá-lo como canal para a solução de novas tarefas. Sendo assim, a força do relacionamento entre dois usuários A e B é definida como o total de interações mútuas entre eles (valor mínimo de interações entre A para B e B para A).

#### 4.4. Síntese

As métricas aplicadas às redes sociais para analisar o comportamento da comunidade de uma perspectiva social serão apresentadas a seguir: (i) **Ordem** é responsável por quantificar o número de usuários ativos na rede social durante o ano analisado; (ii) **Grau médio** é responsável por quantificar o número de relacionamentos médio da rede por usuário ativo durante o ano analisado; (iii) **Força Média** é responsável por quantificar a força média dos relacionamentos da rede social durante o ano analisado.

### 5. Que tipos de artefatos fazem parte da reciclagem de recursos durante e após a morte do ECOS?

**Arquivos de código fonte recebem maior atenção dos desenvolvedores antes da morte do ECOS.** Ao analisar a Figura 2 sobre a dinâmica de trabalho das comunidades nos ECOS, nota-se a presença de comportamentos distintos. No software PhantomJS, os desenvolvedores trabalham periodicamente na manutenção e evolução do software, enquanto o AngularJS apresenta uma atividade contínua. O engajamento dos desenvolvedores é um fator decisivo nas atividades de desenvolvimento e reparação do ECOS.

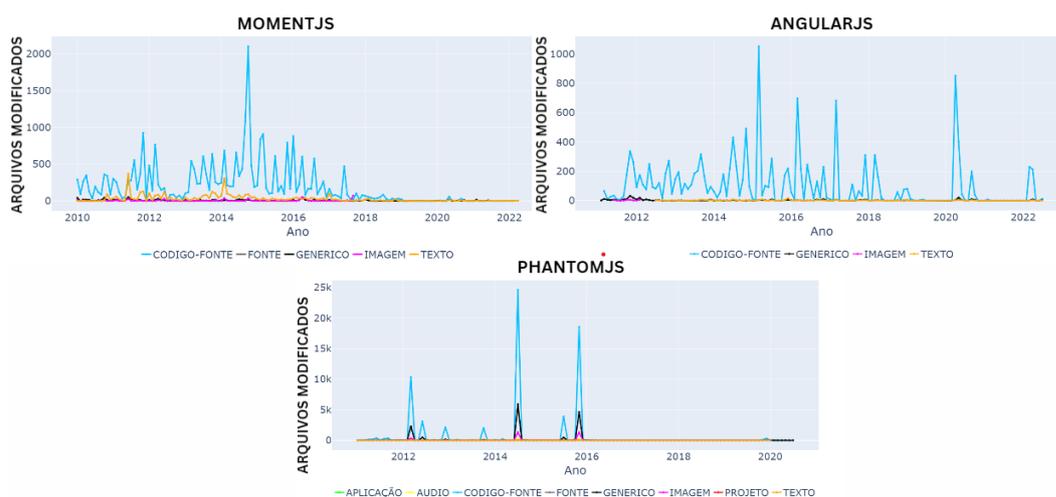


Figura 2. Arquivos modificados por projeto

**Após a morte dos ECOS, o código-fonte não demonstra atividade voltada para a reciclagem de recursos.** Apesar dos arquivos de código fonte serem os principais arquivos trabalhados pelas comunidades durante a vida do ECOS, a atividade de alteração

e manutenção desses arquivos perde intensidade após a sua morte. Todos os projetos apresentam quedas consideráveis na quantidade de arquivos modificados por mês.

**Comunidades mais ativas durante a vida do ECOS permanecem operantes por maior tempo após a morte do ECOS.** A dinâmica de trabalho construída durante a vida do projeto, aparentemente, influencia no comportamento dos desenvolvedores durante a morte do ECOS. O projeto PhantomJS, que enfrentava dificuldades para engajar a comunidade, apresenta os menores índices de atividade de desenvolvimento após a sua morte. Já o AngularJS, que apresentava um plano de migração definido e uma atividade da comunidade intensa, apresenta uma queda mais lenta no desenvolvimento.

## 6. O ECOS consegue manter a atividade dos contribuidores antes, durante e após a sua morte?

**Os projetos atraem um grande número de desenvolvedores nos primeiros anos de vida.** Ao analisar a Figura 3, nota-se que nos primeiros anos de vida dos projetos, há a entrada de um grande número de desenvolvedores. Esse fator indica que os projetos conseguem engajar usuários e comunidades a descobrirem e colaborarem com projetos recentes. Entretanto, após a morte dos ECOS, as comunidades apresentam o menor nível de atividade histórico, demonstrando a falta de suporte aos usuários e outros sistemas na migração de seus requisitos e dependências para outras soluções e ECOS.



Figura 3. Quantidades de desenvolvedores (ordem) por projeto

## 7. O ECOS consegue manter o nível dos relacionamentos dos desenvolvedores antes, durante e após a sua morte?

**O número de relacionamentos da comunidade aumenta nos primeiros anos de vida do projeto.** O número de relacionamentos da comunidade apresenta um comportamento similar à métrica da atividade dos contribuidores como visto na Figura 4. O número de relacionamentos aumenta constantemente durante os primeiros anos de vida do projeto e após isso, decai constantemente até alcançar o menor nível do projeto.

**A comunidade não consegue criar um grande número de relacionamentos por usuário.** Outro padrão encontrado nos projetos é o baixo número de relacionamentos por usuário durante a vida do ECOS. Esse fenômeno serve como alerta para as organizações

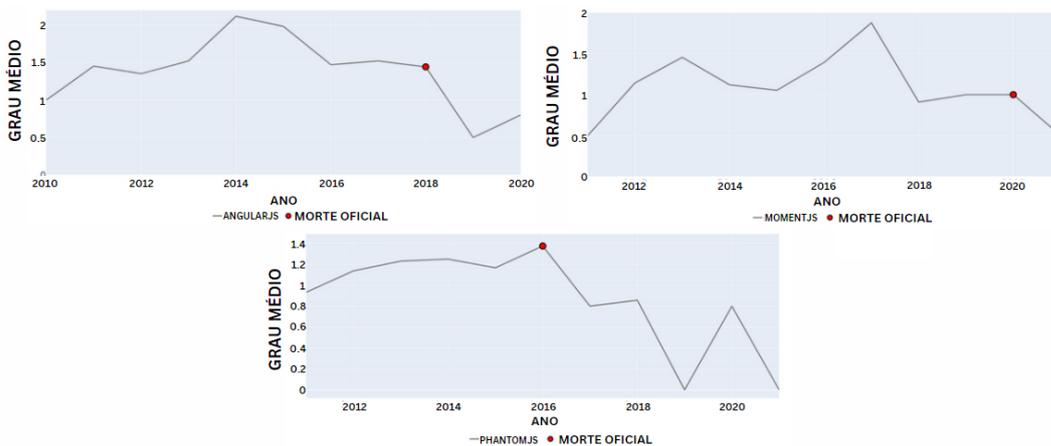


Figura 4. Média de relacionamentos por desenvolvedor (grau médio) por projeto

responsáveis pelos ECOS, principalmente em projetos *open source*, pois fere um dos fundamentos filosóficos de comunidades abertas, o trabalho em equipe.

**ECOS requerem comunidades engajadas e que trabalhem conjuntamente.** Demandas complexas e importantes, que requerem mais de um usuário para desenvolver, podem não ser finalizadas sem uma comunidade conectada, resultando em um projeto estático. Isso pode ocorrer quando *issues* não são investigadas, novos requisitos não são atendidos ou até *pull requests* não são verificadas e validadas.

## 8. Os desenvolvedores do ECOS conseguem construir relacionamentos fortes?

**O comportamento da força dos relacionamentos é único para cada ECOS.** Ao verificar a Figura 5, percebe-se que no AngularJS, a comunidade fortalece o seu relacionamento durante os primeiros anos do projeto, mas eles perdem força após esse período. Por outro lado, as comunidades do PhantomJS e MomentJS apresentam uma grande força inicial nos relacionamentos e em seguida uma constante perda de forças até a morte dos ECOS.

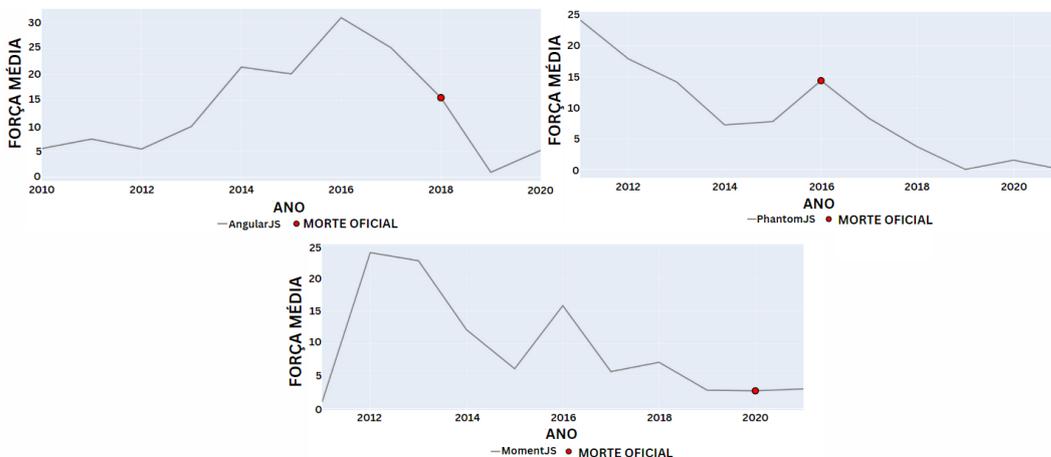


Figura 5. Força média dos relacionamentos por projeto

**Desenvolvedores conseguem desenvolver relacionamentos fortes, mas precisam de suporte para mantê-los.** É notório a habilidade das comunidades de desenvolvi-

mento para criar relacionamentos fortes entre seus membros, especialmente nos anos de ouro do projeto. Portanto, é válido investigar em estudos futuros o impacto de práticas e métodos nas relações dos desenvolvedores e nos aspectos sociais dos projetos.

## 9. Ameaças à Validade

Em relação à validade de *construto*, a base teórica deste estudo considerou as fragilidades apontadas nos trabalhos relacionados. A escolha do GH se deve à presença de desenvolvedores e usuários interagindo em torno do ECOS. Para auxiliar a validade interna, os conjuntos de dados foram relacionados ao ecossistema estudado. Para reduzir o efeito da expectativa dos experimentadores, as análises do estudo seguiram os procedimentos indicados para estudos de MRS. Generalização - este estudo está focado em ferramentas de soluções Web, o estudo precisa ser replicado em outros ECOS. Expectativa do experimentador - para contornar a expectativa do experimentador, foram realizadas reuniões de conciliação que envolveram a análise dos resultados.

## 10. Conclusão e Trabalhos Futuros

Nesse trabalho, avançou-se no estudo de reciclagem de recursos, redes sociais e no comportamento da comunidade de desenvolvimento em torno da morte de ECOS. Focou-se em projetos direcionados para ferramentas web, como AngularJS, PhantomJS e MomentJS. Das questões de pesquisa, foram encontradas algumas direções: **Reciclagem de Recursos:** arquivos de código-fonte e texto recebem mais manutenção, entretanto, após a morte do ECOS, não apresentam evolução em seu quadro. Nos ambientes que apresentam plano de migração, há uma queda (lenta) na manutenção e evolução do sistema após a morte do ECOS. **Recrutamento e permanência de desenvolvedores:** Sem engajamento, as comunidades emigram do ECOS após alguns anos de contribuição. A organização responsável precisa aplicar métodos eficazes para o controle da comunidade em torno da plataforma. **Relacionamentos da comunidade:** A comunidade necessita de engajamento para fortalecer e manter seus relacionamentos. Programas, *workshops* e eventos são úteis para incentivar usuários a colaborarem entre si.

Benefícios para a indústria e academia: analisar métricas sobre a atividade da comunidade em séries temporais para identificar a saúde do ECOS. Esse método auxilia que organizações elaborem estratégias de migração dos ECOS. Avanço no conhecimento sobre condições que podem resultar na morte de ECOS.

Como trabalho futuro, busca-se identificar ECOS que restauraram sua saúde após um período crítico e quais os métodos aplicados para controlar e recuperar a atividade da comunidade. Além disso, verificar projetos com um longo período de comunidade ativa e entender quais métodos foram aplicados para manter a saúde do ECOS estável.

## 11. Desdobramentos da Iniciação Científica

Como resultado deste trabalho, foram publicados dois artigos. O primeiro no Simpósio Brasileiro de Engenharia de Software (SBES) 2022, com título *"We are dying!" On Death Signals of Software Ecosystems*. Já o segundo no 2023 IEEE/ACM 11th International Workshop on Software Engineering for Systems-of-Systems and Software Ecosystems (SESoS), com título *Social Networks during Software Ecosystems' Death* e premiado como melhor artigo da categoria "Short Paper".

O projeto implicou em mais duas iniciações científicas que cobrem a análise da migração do ECOS AngularJS para Angular e a metrificação da saúde de ECOS.

## Referências

- Arantes, P., Fontão, A., and Soupinski, F. (2023). Paper replication. available at <https://encurtador.com.br/nHIM5> (2023/02/06).
- Bird, C., Pattison, D., D’Souza, R., Filkov, V., and Devanbu, P. (2008). Latent social structure in open source projects. In *Proceedings of the 16th ACM SIGSOFT International Symposium on Foundations of software engineering*, pages 24–35.
- Businge, J., Openja, M., Nadi, S., and Berger, T. (2022). Reuse and maintenance practices among divergent forks in three software ecosystems. *Empirical Software Engineering*, 27(2):54.
- Coelho, J., Valente, M. T., Milen, L., and Silva, L. L. (2020). Is this github project maintained? measuring the level of maintenance activity of open-source projects. *Information and Software Technology*, 122:106274.
- Evertse, R., Lencz, A., Šinik, T., Jansen, S., and Soussi, L. (2021). Is your software ecosystem in danger? preventing ecosystem death through lessons in ecosystem health. In *International Conference on Agile Software Development*, pages 96–105. Springer.
- Garton, L., Haythornthwaite, C., and Wellman, B. (1997). Studying online social networks. *Journal of computer-mediated communication*, 3(1):JCMC313.
- Hemmati, H., Nadi, S., Baysal, O., Kononenko, O., Wang, W., Holmes, R., and Godfrey, M. W. (2013). The msr cookbook: Mining a decade of research. In *2013 10th Working Conference on Mining Software Repositories (MSR)*, pages 343–352. IEEE.
- Hyrnsalmi, S., Seppänen, M., Nokkala, T., Suominen, A., and Järvi, A. (2015). Wealthy, healthy and/or happy—what does ‘ecosystem health’ stand for? In *International conference of software business*, pages 272–287. Springer.
- Ma, Y., Fakhoury, S., Christensen, M., Arnaoudova, V., Zogaan, W., and Mirakhorli, M. (2018). Automatic classification of software artifacts in open-source applications.
- Manikas, K. (2016). Revisiting software ecosystems research: A longitudinal literature study. *Journal of Systems and Software*, 117:84–103.
- Massanori, D., Cafeo, B. B., Wiese, I., and Fontão, A. (2020). Death of a software ecosystem: a developer relations (devrel) perspective. In *Proceedings of the 34th Brazilian Symposium on Software Engineering*, pages 399–404.
- Michael Mu Sun, Akash Ghosh, R. S. and Kuttal, S. K. (2018). Birds of a feather flock together? a study of developers’ flocking and migration behavior in github and stack overflow. *IJCSIS Computer Science Journal*.
- Soupinski, F. and Fontao, A. (2022). Ecossistemas de software “mortos”: uma visão a partir de devrel (developer relations). In *Anais Estendidos do XIII Congresso Brasileiro de Software: Teoria e Prática*, pages 99–108. SBC.