

Promovendo a Participação de Mulheres e Grupos Minorizados em Comunidades de Software Livre: Um Relato de Experiência com a Metodologia BOSS

Maria Eduarda Andrade¹ , Daniela de Oliveira¹ ,
Suzane Alves Duarte¹ , Giovanna Barbosa² , Carla Rocha¹ 

¹Faculdade de Ciências e Tecnologias em Engenharia
Universidade de Brasília (FCTE/UnB)
Brasília – DF – Brasil

²Instituto de Informática
Universidade Federal de Goiás (INF/UFG)
Goiânia – GO – Brasil

222037577@aluno.unb.br, dannysoaresdeoliveira@gmail.com,

ssuzane9@hotmail.com, waleskatheodoro@discente.ufg.br, caguiar@unb.br

Abstract. *This report investigates the promotion of women’s retention in Free and Open Source Software (FOSS) communities through the BOSS methodology, structured into four stages with technical workshops, 1:1 mentoring, pair programming, and contributions to FOSS projects. The results indicate reduced initial technical barriers, increased participants’ self-confidence, and the development of the technical and collaborative skills, suggesting that BOSS contributes to the creation of more inclusive learning and development environments. We present BOSS as a structured methodology for promoting diversity in FOSS, providing insights for educational initiatives and training programs.*

Resumo. *Este relato investiga a promoção da permanência feminina em comunidades de software livre (FOSS) por meio da metodologia BOSS, estruturada em quatro etapas com workshops técnicos, mentoria 1:1, programação em pares e contribuições em projetos FOSS. Os resultados indicam a redução de barreiras técnicas iniciais, o fortalecimento da autoconfiança das participantes e o desenvolvimento de competências técnicas e colaborativas, sugerindo que a BOSS contribui para a construção de ambientes formativos e de desenvolvimento mais inclusivos. Apresentamos a BOSS como metodologia estruturada para a promoção da diversidade em projetos FOSS, oferecendo lições para iniciativas educacionais e programas de formação.*

1. Introdução

A diversidade na área de computação tem sido amplamente discutida na literatura devido à sub-representação de diferentes grupos sociais no desenvolvimento de tecnologias. Estudos indicam que equipes de engenharia de software frequentemente não refletem a diversidade da população usuária, apresentando predominância de perfis demográficos específicos e barreiras estruturais à participação de mulheres e grupos minorizados [Prana *et al.* 2022]. Esse cenário evidencia a necessidade de iniciativas que

promovam maior inclusão na área. A homogeneidade das equipes pode influenciar decisões de projeto, processos de desenvolvimento e a construção de soluções tecnológicas. Consequentemente, a diversidade impacta diretamente a qualidade e a abrangência das tecnologias produzidas.

A sub-representação na computação é um fenômeno amplamente documentado em diversos contextos educacionais e profissionais. Segundo relatório da UNESCO, mulheres representam uma parcela reduzida da força de trabalho em áreas tecnológicas, sendo observado que apenas 2% dos participantes de projetos de software livre, eram mulheres no estudo conduzido pela organização [UNESCO 2012]. Além da desigualdade de gênero, estudos apontam que disparidades relacionadas à fatores socioculturais como estereótipos, percepção de não pertencimento e ausência de representatividade, contribuem para menor ingresso e maior evasão de mulheres em cursos e carreiras tecnológicas [Cheryan *et al.* 2017]. O impacto negativo à construção da identidade profissional e o engajamento nas comunidades técnicas evidenciam que a evasão é influenciada tanto por fatores técnicos quanto por aspectos sociais e psicológicos [Beyer 2014, Dasgupta and Stout 2014].

Projetos de software livre (*FOSS – Free and Open Source Software*) são frequentemente utilizados como ambiente de aprendizado para estudantes [Pinto *et al.* 2017]. Em vez de hierarquias rígidas, o desenvolvimento flui por meio de contribuições voluntárias, em que o código nasce da revisão contínua e de debates técnicos intensos entre pessoas do mundo todo [Feller *et al.* 2002]. A presença de mulheres e grupos minorizados em projetos de código aberto ainda é baixa, evidenciando que o ‘aberto’ do FOSS nem sempre significa ‘acessível’. Pesquisas recentes demonstram que menos de 10% de contribuidoras ativas e menos de 5% de projetos abertos, reportam ter mantenedoras mulheres [Trinkenreich *et al.* 2022, Hechtel *et al.* 2025].

Novaes *et al.* apontam que a maior barreira de entrada não é técnica, mas socio-cultural [Novaes *et al.* 2025]. Sem se sentirem representadas ou acolhidas, a permanência dessas profissionais se torna uma batalha constante contra o ambiente em que estão inseridas. A literatura também pontua que mulheres sofrem com ‘micro-inequidades’ na área de desenvolvimento de software: contribuições ignoradas, interrupções constantes em debates técnicos e divisões injustas de trabalho [Guzman *et al.* 2023]. No mundo do software livre, essa dinâmica se intensifica por se tratar de um ambiente voluntário e disperso: estudos evidenciam que o trabalho de mulheres neste âmbito é inviabilizado em plataformas de desenvolvimento [Batista *et al.* 2023].

Programas de mentoria, iniciativas de formação técnica e comunidades de apoio têm sido apontados como estratégias relevantes para reduzir as barreiras de participação feminina. No entanto, muitas dessas ações se concentram na participação técnica, sem mecanismos contínuos de permanência, integração comunitária e fortalecimento do sentimento de pertencimento. Adicionalmente, programas notórios como Google Summer of Code e Outreachy são construídos em uma esfera anglófona, concentrada em organizações e estruturas do Norte Global. Barreiras culturais e linguísticas dificultam ou restringem a participação de pessoas do Sul Global nesses ambientes [Steinmacher *et al.* 2019, Ferreira 2020].

A metodologia BOSS (*Big Open Source Sibling*) foi desenvolvida para promover

a permanência de mulheres e de grupos minorizados em comunidades de software livre e código aberto no Brasil e é composta por quatro etapas: (i) introdução ao ecossistema de software livre; (ii) enfrentamento de desafios técnicos e socioculturais com suporte de mentoria; (iii) integração das participantes em uma comunidade local de software livre; e (iv) desenvolvimento de autonomia para participação em comunidades externas.

Conforme discutido na sistematização da metodologia BOSS [Oliveira 2025], a evasão de mulheres e grupos minorizados não decorre de incapacidade técnica, mas de barreiras estruturais, da ausência de redes de apoio e da fragilização da identidade profissional. Torna-se necessária, então, uma abordagem que articule formação técnica, suporte socioemocional e inserção progressiva em comunidades reais de prática.

Este relato de experiência sobre a aplicação da metodologia BOSS busca responder à seguinte questão de pesquisa: *Como uma metodologia estruturada pode contribuir para a permanência de mulheres e grupos minorizados em comunidades de software livre e código aberto?* Como contribuição, o artigo sistematiza a metodologia BOSS e discute suas etapas, a estrutura de mentoria e as estratégias empregadas para promover a participação de mulheres e de grupos minorizados em comunidades de software livre.

2. A Metodologia BOSS - Estrutura e Aplicação

2.1. Histórico

A metodologia BOSS foi concebida pela professora Carla Rocha através de sua experiência de uma década ministrando as disciplinas de “Metodologia de Desenvolvimento de Software” e “Engenharia de Produto de Software” no graduação em Engenharia de Software na Universidade de Brasília (UnB)¹. A disciplina foi inicialmente inspirada em aplicações de práticas de *Extreme Programming* (XP) e metodologias ágeis encabeçadas por Alfredo Goldman, Fabio Kon e Paulo J. S. Silva na Universidade de São Paulo (USP) através da disciplina “LabXP”. Em sua forma original, 20 estudantes são divididos em grupos de 6 a 10 integrantes. Professores de Ciência da Computação ou de outros departamentos assumiam os papéis de clientes, que definiam qual sistema seria produzido, e, às vezes, de mentores ao lado de outros estudantes que já passaram pela disciplina [Goldman *et al.* 2004].

Na versão brasiliense, estudantes de quarto e oitavo período da graduação em Engenharia de Software são agrupados com base nas duas disciplinas supracitadas. O agrupamento de estudantes em dois momentos distintos da graduação abre o caminho para mentorias entre pares. Os grupos devem, então, construir uma prova de conceito (temática, com variações de semestre em semestre) ou um novo módulo para um software livre existente. O desenvolvimento dos projetos ocorre principalmente na plataforma GitHub², seguindo princípios da cultura open source: disponibilização de código-fonte sob licenças MIT ou AGPL e de documentação pública, com instruções de uso e preparo para contribuições externas.

A expansão da metodologia para um programa de mentoria remota nasce da parceria entre Carla Rocha e egressas do curso de Engenharia de Software da UnB, a partir de suas experiências com comunidades como PyLadies e Django Girls, e com programas

¹<http://fga-eps-mds.github.io/A-Disciplina-MDS-EPS/>

²<https://github.com/fga-eps-mds>

similares como Outreachy e Google Summer of Code. Cada edição é construída a partir de um eixo temático diferente. A primeira iteração, focada em contribuições a projetos de software livre brasileiros, foi premiada pela Fundação GNOME no GNOME Community Engagement Challenge. Edições subsequentes exploraram também tópicos relacionados a desenvolvimento de jogos e maratonas de programação [D'Alessandro 2021].

2.2. Bem-Vindo à BOSS

A **Figura 1** ilustra o fluxo logístico, a progressão da trajetória e competências adquiridas na jornada de participantes, denominados *Little Siblings*. A primeira etapa da metodologia, *Welcome to BOSS*, corresponde ao momento de introdução das participantes ao ecossistema de software livre. Essa fase foi aplicada nas temporadas de 2020, 2021, 2023, 2024 e 2025, em formato totalmente remoto, e teve como objetivo proporcionar um ambiente seguro de acolhimento, considerando o contexto de sub-representação de mulheres e de grupos minoritários nas comunidades FOSS. Diferentemente das comunidades de código aberto convencionais, que frequentemente operam sob o mito da meritocracia e impõem barreiras de entrada por meio de comunicação hostil e da presunção de letramento técnico prévio [Lee *et al.* 2019], a etapa *Welcome to BOSS* atua ativamente na construção de segurança psicológica. Ela não apenas abriga, mas também estabelece, do zero, as bases técnicas e culturais necessárias à participação inclusiva.

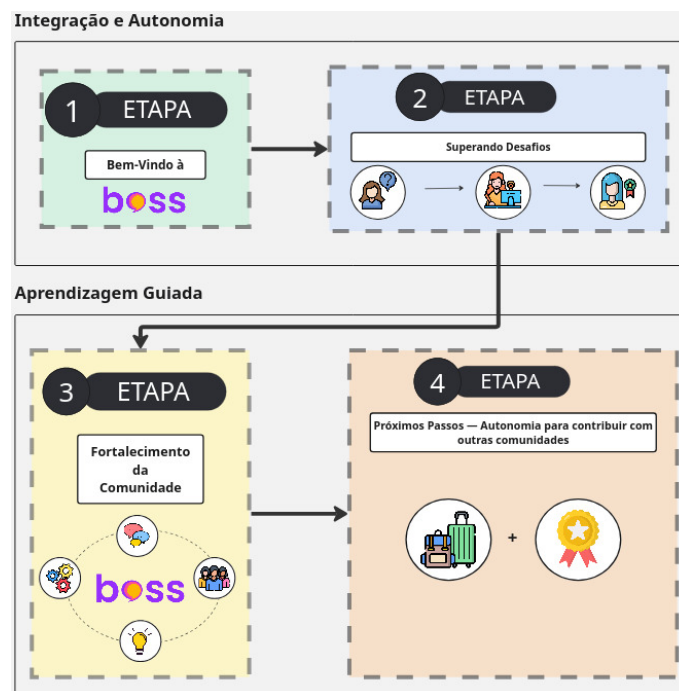


Figura 1. Fluxo metodológico da iniciativa BOSS, organizado em quatro etapas progressivas que conduzem da formação técnica inicial à autonomia para contribuir com as comunidades FOSS.

A metodologia BOSS se estrutura em dois momentos fundamentais: a seleção e o treinamento das mentoras (*Big Siblings*), e a mentoria de (*Little Siblings*). Anterior ao início de cada temporada, as mentoras voluntárias passam por um processo de treinamento e integração de quatro semanas. Esse período é dedicado à apresentação de conceitos como interseccionalidade, síndrome do impostor e os impactos do preconceito

sistêmico na carreira de mulheres e grupos sub-representados, preparando-as para conduzir as reuniões individuais com empatia e técnica.

Nesta etapa, foram realizados workshops semanais síncronos abordando temas como fundamentos de software livre, as quatro liberdades essenciais, introdução ao Git, configurações de ambiente local (instalação de sistema operacional Linux, IDEs e dependências), além do fluxo de contribuição em projetos *open source*. As atividades eram complementadas por tarefas práticas de escopo reduzido e repetitivo, estratégia adotada para fortalecer a autoconfiança das participantes e reduzir a ansiedade inicial frequentemente associada ao ingresso em comunidades técnicas.

Cada *Little Sibling* foi acompanhada por uma *Big Sibling* em encontros individuais semanais. Nestes encontros, mapearam-se interesses, experiências prévias, motivações, expectativas e possíveis inseguranças das participantes em relação à área de tecnologia e à contribuição em projetos de software livre. Esse acompanhamento individual constituiu um elemento estruturante da metodologia. Metodologias tradicionais dependem de uma mentoria informal e reativa, em que o iniciante deve buscar ajuda em ambientes frequentemente intimidantes [Steinmacher *et al.* 2015]. Aqui, encontros semanais sistematizam o acolhimento, alinhando o desenvolvimento técnico às metas pessoais e profissionais de cada participante, contornando a falha sistêmica da falta de direcionamento de carreira.

Em paralelo, a programação incluiu atividades práticas semanais com tarefas delimitadas e de baixa complexidade inicial, como atualização de repositórios pessoais, abertura de issues e submissão de pull requests em ambientes previamente preparados para fins pedagógicos. Diferentemente da curva de aprendizado abrupta imposta por repositórios de código aberto em produção, em que iniciantes são expostos precocemente a arquiteturas complexas que dificultam a identificação de tarefas iniciais adequadas [Steinmacher *et al.* 2018], o uso de ambientes previamente preparados atuou como um sandbox pedagógico. A adoção de tarefas repetitivas e incrementalmente desafiadoras configurou-se como estratégia metodológica para promover a familiarização com o fluxo de trabalho colaborativo em FOSS, reduzindo as barreiras técnicas iniciais e favorecendo a consolidação de competências operacionais básicas.

Além das atividades técnicas, foram estabelecidos canais formais de comunicação síncrona e assíncrona, incluindo grupos em plataformas de mensagens instantâneas e repositórios colaborativos, a fim de estimular a interação entre participantes, mentoras e equipe organizadora. O uso orientado desses canais buscou reproduzir dinâmicas próprias de comunidades FOSS, promovendo a aprendizagem não apenas de ferramentas técnicas, mas também de práticas comunicacionais do trabalho colaborativo distribuído. Do mesmo modo, ao simular essas interações em um ambiente controlado, a metodologia protege os participantes da ansiedade de exposição pública e do debate frequentemente combativo típico de fóruns abertos do ecossistema FOSS [Nafus 2012], permitindo que o letramento comunicacional ocorra de forma segura.

Como critérios de acompanhamento dessa etapa, adotaram-se: (i) a configuração correta do ambiente local de desenvolvimento; (ii) a realização dos exercícios propostos sobre controle de versão; (iii) a utilização adequada dos canais de comunicação; e (iv) o mapeamento inicial dos interesses e objetivos individuais das participantes, conforme previsto no framework metodológico. Entre os desafios relatados no próprio desenho da

iniciativa, destacam-se dificuldades técnicas iniciais relacionadas à configuração de ambiente e ao fluxo de versionamento distribuído, bem como inseguranças quanto à exposição pública do código. A estrutura de tarefas progressivas, aliada ao acompanhamento individualizado e ao suporte coletivo, foi concebida para mitigar essas barreiras. A consolidação desses indicadores sinalizava a conclusão da etapa *Welcome to BOSS* e a preparação das participantes para um aprofundamento técnico.

2.3. Superando Desafios

Nessa etapa, o objetivo é aprender tecnologias necessárias para contribuir em uma comunidade local de software livre, a metodologia BOSS avança do nível introdutório para um aprofundamento técnico orientado à contribuição efetiva em um projeto de software livre. Essa fase foi estruturada com o objetivo de superar barreiras técnicas e comportamentais que frequentemente dificultam a permanência de mulheres em comunidades FOSS, especialmente as relacionadas à compreensão da arquitetura de projetos reais, à leitura de código legado e à interação em ambientes colaborativos distribuídos.

A literatura evidencia a gravidade dessas barreiras: análises realizadas na plataforma GitHub, as mulheres apresentam uma discrepância significativa no tempo de participação (de 2 a 9 anos) em relação aos homens (de 4 a mais de 10 anos) [Batista *et al.* 2023]. Além da evasão precoce, os mesmos autores demonstraram que a representatividade global é criticamente baixa, com autoria feminina em apenas 16% das *issues* reportadas e em 22% dos comentários postados [Batista *et al.* 2023].

Diferentemente da fase anterior, focada na familiarização com ferramentas e fluxos básicos, esta etapa concentrou-se na exploração dos conceitos técnicos específicos do projeto adotado na temporada, incluindo as bibliotecas utilizadas, a organização do código, os padrões arquitetônicos e as dependências. Os workshops semanais passaram a incorporar sessões de análise de código, demonstrações práticas de *live coding* e atividades com escopo delimitado, desenvolvidas preferencialmente em pares, com o intuito de promover a aprendizagem colaborativa e fortalecer a confiança técnica.

Essa imersão técnica, em um espaço seguro, atua diretamente na retenção. Estudos demonstram que a participação em ambientes dedicados a mulheres promove o empoderamento e a segurança necessários para que elas se insiram posteriormente em comunidades abertas mais complexas, fenômeno observado na transição de desenvolvedoras de projetos exclusivos para repositórias gerais [Batista *et al.* 2023]. Dessa forma, o aprofundamento colaborativo da metodologia atua como uma ponte estruturada para a contribuição sustentável em ecossistemas globais.

A programação em pares constitui uma das principais estratégias metodológicas dessa fase, empregada tanto no desenvolvimento das atividades propostas quanto na resolução de problemas técnicos emergentes. A divisão das participantes em duplas buscou favorecer a troca de conhecimentos, estimular a verbalização do raciocínio técnico e reduzir a insegurança associada à exposição individual a tarefas mais complexas. Pesquisas sobre a percepção de gênero na computação evidenciam que mulheres relatam sistematicamente que o trabalho em duplas atua como um mecanismo crucial para atenuar a frustração inicial diante de códigos complexos e mitigar a sensação de isolamento [Ying *et al.* 2019]. Dessa forma, as atividades foram planejadas com escopo delimitado e progressivamente desafiador, permitindo que conceitos densos, como a arquitetura do

projeto, a organização de módulos, o tratamento de dependências e as boas práticas de documentação, fossem assimilados de forma contextualizada e colaborativa, contornando as barreiras à evasão típicas de ambientes de desenvolvimento solitários.

Nessa etapa, a metodologia lida com a complexidade técnica de um projeto real de software livre, envolvendo desafios relacionados à leitura e à compreensão de código já existente, à realização de modificações compatíveis com a arquitetura do projeto e à participação em discussões técnicas mais estruturadas. Para lidar com esse cenário, foram adotadas estratégias como a programação em pares, a fragmentação das tarefas em unidades menores e a manutenção do acompanhamento individual semanal, oferecendo suporte progressivo às exigências técnicas e colaborativas da fase.

Com base em critérios de acompanhamento, considerou-se a capacidade de executar o projeto em ambiente local, compreender sua organização interna e realizar contribuições de escopo delimitado, conforme os conteúdos abordados nos workshops. Também se consideraram como referência o engajamento nas atividades colaborativas propostas e o desenvolvimento das tarefas em pares, o que sinaliza a progressão necessária para a etapa subsequente, voltada à interação direta com comunidades FOSS externas ao ambiente formativo do programa.

2.4. Fortalecimento da Comunidade

Na etapa subsequente, a metodologia BOSS introduz o *onboarding* na comunidade local do ecossistema, correspondente ao momento de inserção das participantes em comunidades reais de software livre e de código aberto. Após a consolidação das competências técnicas básicas e o aprofundamento progressivo na estrutura de um projeto FOSS, essa fase foi concebida para promover a transição do ambiente formativo para o ecossistema aberto, no qual as dinâmicas de colaboração extrapolam o contexto pedagógico. O objetivo central consistiu em possibilitar que as *Little Siblings* contribuíssem para projetos ativos, interagissem com mantenedores e demais colaboradores, experimentando o fluxo autêntico de desenvolvimento distribuído característico dessas comunidades.

Para viabilizar essa inserção, foram selecionados projetos compatíveis com o nível de experiência das participantes e alinhados aos interesses previamente mapeados nos encontros individuais. As contribuições inicialmente incentivadas incluíram atividades de menor complexidade, como a melhoria da documentação, a correção de pequenos bugs ou ajustes incrementais, respeitando o ritmo de aprendizagem previsto pela metodologia. Essa curadoria de tarefas conta com forte respaldo na literatura sobre *onboarding* em comunidades de código aberto. A dificuldade em encontrar uma tarefa inicial adequada é uma das barreiras recorrentes enfrentadas por iniciantes [Steinmacher *et al.* 2018]. Os autores evidenciam que direcionar recém-chegadas a tarefas de baixa complexidade, que oferecem recompensas rápidas, é um passo fundamental para que elas se tornem conhecidas na comunidade e mantenham o engajamento. O acompanhamento por parte das *Big Siblings* permaneceu ativo, oferecendo suporte na interpretação dos feedbacks da comunidade e na adequação das contribuições às normas e aos padrões específicos de cada projeto.

A participação em comunidades FOSS reais envolve a interação em ambientes colaborativos abertos, caracterizados por fluxos próprios de comunicação, revisão pública de código e diferentes níveis de experiência entre os colaboradores. Nesse contexto, a

necessidade de submeter contribuições à avaliação de mantenedores externos, interpretar feedbacks técnicos e adequar-se a padrões específicos de cada projeto configura um momento de maior exposição e responsabilidade para as participantes. A metodologia buscou oferecer suporte a esse processo por meio do acompanhamento contínuo das *Big Siblings* e da preparação prévia realizada nas etapas anteriores, atenuando as barreiras técnicas e simbólicas associadas ao ingresso em comunidades consolidadas. Estudos demonstram que a falta de mentoria e o choque cultural com a comunicação direta de desenvolvedores experientes, que frequentemente emitem críticas sem polidez, são fatores-chave para a evasão. Nesse sentido, a figura da mentora atua exatamente na prestação de suporte guiado e na mediação da comunicação, estratégias apontadas por [Steinmacher *et al.* 2018] como cruciais para ajudar a superar essas barreiras e interpretar corretamente as interações da comunidade.

Essa etapa representa o ponto de convergência da trajetória formativa proposta pela BOSS ao articular o desenvolvimento técnico, o fortalecimento da autoconfiança e a inserção efetiva no ecossistema de software livre. O encerramento do ciclo metodológico não se configura como conclusão da participação das *Little Siblings*, mas sim como transição para uma atuação mais autônoma, alinhada aos objetivos profissionais mapeados ao longo do programa e sustentada pelas competências técnicas e colaborativas construídas nas fases anteriores.

2.5. Próximos Passos — Autonomia para contribuir com outras comunidades

A última etapa, intitulada *What's Next?* corresponde ao momento final da metodologia BOSS e tem como objetivo ampliar o horizonte de atuação das participantes para além do projeto trabalhado ao longo do programa. Essa fase marca a transição do acompanhamento estruturado para o estímulo à autonomia, promovendo o contato com diferentes comunidades FOSS por meio de workshops de integração. Mantenedores e colaboradores convidados apresentam suas respectivas comunidades, abordando aspectos como o propósito do projeto, as principais funcionalidades, o fluxo de contribuição, os canais de comunicação e as possibilidades de atuação para iniciantes. Workshops incluem demonstrações práticas e orientação para configuração local do ambiente do projeto, favorecendo uma aproximação inicial com novas comunidades. Ao final da etapa, esperava-se que cada *Little Sibling* participasse de ao menos uma dessas atividades e identificasse potenciais áreas de interesse para a continuidade de sua trajetória em software livre.

Como fechamento do percurso metodológico, essa etapa sinaliza a transição de um processo formativo estruturado para uma atuação mais autônoma e em rede. A literatura da área indica que a retenção a longo prazo de iniciantes em projetos Open Source depende fortemente de fatores de socialização, como a intensidade da interação e a criação de laços de suporte mútuo [Carillo *et al.* 2014]. Ao promover o contato direto com mantenedores convidados, os *workshops* de integração reduzem o atrito de uma nova inserção, facilitando o que Steinmacher *et al.* descreveram como a expansão orgânica do movimento FOSS e de sua diversidade [Steinmacher *et al.* 2018]. Dessa forma, a BOSS consolida o capital social necessário para fomentar a permanência e a continuidade da participação feminina na tecnologia, mais especificamente em comunidades FOSS, ampliando as possibilidades de inserção profissional e de construção de redes de colaboração no longo prazo.

3. Resultados e Lições Aprendidas

A primeira temporada do programa BOSS, realizada em 2020, contou com a participação de 10 *Little Siblings*. No decorrer de 77 dias, *Little Siblings* trabalharam em um *bot* baseado no projeto Tais do laboratório de pesquisa aplicada Lab Livre da Universidade de Brasília (UnB)³, produzindo 45 novos commits e 12 forks (bases de código derivadas do repositório original). 10 issues foram criadas, e 4 delas foram finalizadas. Em formulários de acompanhamento, a principal dificuldade relatada por *Little Siblings* foi a de acompanhamento do conteúdo apresentado ao mesmo tempo que cumpriam obrigações de seus semestres letivos.

A temporada de 2021 recepcionou 26 *Little Siblings* e teve como base o mesmo repositório no GitHub. Em 88 dias de programa, 48 novos commits foram incorporados ao repositório, e 16 novos forks foram criados. Nesta temporada, *Little Siblings* citaram a busca por aprimoramento técnico como principais motivações para participar do programa; 76,5% relataram conseguir programar com confiança e de forma independente ao final do programa. Ao menos duas *Little Siblings* concluíram o programa selecionadas para vagas de estágio.

Temporadas subsequentes adotaram focos e durações diferentes – a temporada de 2024, por exemplo, aconteceu no decorrer de dois meses e se concentrou em formar novas contribuidoras para o projeto aberto Plone⁴, um CMS escrito em Python. Adicionalmente, a documentação da jornada de *Little Siblings* se tornou mais escassa. Sugerimos que futuros trabalhos para o rastreamento de impacto realizem análises de repositórios correlatos e um estudo longitudinal com participantes de todas as temporadas.

Apesar disso, observamos através da análise de documentos internos que aplicação da metodologia BOSS ao longo de suas edições proporcionou aprendizados significativos sobre a estruturação e manutenção de programas de diversidade em ecossistemas de software livre. Uma das principais lições extraídas foi a evolução conceitual da iniciativa: a percepção de que a BOSS transcende a execução de um programa isolado de mentoria, consolidando-se como um *framework* metodológico adaptável. A refatoração do roteiro do programa evidenciou que sua estrutura base pode ser replicada e ajustada para atender às necessidades de qualquer outra comunidade de código aberto, potencializando seu alcance para além do contexto em que foi concebido.

No âmbito operacional, a carga horária e o desgaste associados ao processo de mentoria emergiram como desafios críticos à sustentabilidade da iniciativa. Observou-se, sobretudo na primeira temporada, que o acúmulo de funções gerava sobrecarga para as mentoras voluntárias, comprometendo o bem-estar da equipe e a viabilidade do projeto a longo prazo. Para mitigar esse risco estrutural, a metodologia foi refinada por meio da dispersão de responsabilidade e da adoção de dedicação delimitada no tempo (*time-box*). A divisão clara de papéis, ao segmentar a atuação em mentoras de carreira, permitiu uma distribuição equilibrada do trabalho, resultando em um modelo operacional mais resiliente e sustentável.

Mentoras e organizadoras também pontuaram que participantes do programa realizavam suas atividades em momentos geralmente dedicados ao descanso, cumprindo

³<https://github.com/BOSS-BigOpenSourceSibling/bot-da-boss>

⁴<https://plone.org/>

assim um “terceiro turno”. Em programas como Outreachy, a bolsa-auxílio para pessoas mentoradas permite que elas se dediquem exclusivamente ao projeto a ser desenvolvido. Em programas como a BOSS, a ausência de uma estrutura de suporte financeiro, por meio de bolsas e/ou doações de computadores, dificulta a retenção de participantes.

A experiência também demonstrou que não basta atrair participantes de grupos sub-representados; é imperativo que a própria equipe organizadora e de mentoria reflita a diversidade almejada. A inclusão de mulheres em diferentes estágios de formação e atuação, desde estudantes universitárias até líderes técnicas na indústria e em grandes projetos *open source*, enriqueceu a metodologia, fornecendo perspectivas mais realistas sobre o mercado de software brasileiro e atualizando continuamente as práticas de desenvolvimento colaborativo repassadas às *Little Siblings*.

A efetividade da metodologia também foi validada empiricamente por meio de sua apropriação por atores externos. Constatou-se que o *framework* BOSS passou a ser adotado corporativamente por empresas de tecnologia no Brasil como instrumento formal de capacitação e preparação de grupos minoritários e de mulheres para o preenchimento de vagas no mercado de trabalho. Essa adoção atua como uma chancela prática da estrutura formativa desenhada, indicando que o desenvolvimento articulado de competências técnicas e a mitigação de barreiras psicológicas (como a síndrome do impostor) têm valor tático que transcende a retenção no ambiente da tecnologia e nas comunidades FOSS, impactando diretamente a empregabilidade.

Por fim, a necessidade de padronização e de documentação contínua dos processos revelou-se um requisito indispensável não apenas para a escalabilidade operacional da iniciativa, mas, fundamentalmente, para ampliar seu impacto social no combate à evasão feminina na tecnologia. A criação de modelos estruturados, roteiros para eventos síncronos e guias operacionais conferiu grande organização ao programa, permitindo replicar a metodologia como uma ferramenta concreta de intervenção contra a desigualdade de gênero no setor. Mais do que um *framework* técnico-pedagógico, a BOSS revelou-se um instrumento essencial para a desconstrução de barreiras sistêmicas que, historicamente, afastam as mulheres da área. Como desdobramento desse amadurecimento processual, a internacionalização da iniciativa desponta como um passo estratégico: ao viabilizar que comunidades globais adotem e adaptem o modelo aos seus próprios ecossistemas, o programa amplia sua capacidade de acolhimento e de fortalecimento da identidade profissional, atuando ativamente para reverter a sub-representação e garantir a permanência de mulheres e grupos minoritários na construção da tecnologia.

Uso de Inteligência Artificial

Utilizamos ferramentas de inteligência artificial, mais especificamente LLMs, para realizar a revisão de gramática, de coesão e coerência deste artigo.

Agradecimentos

Este estudo contou com o apoio financeiro da Secretaria-Geral da Presidência da República, no âmbito do Projeto Brasil Participativo.

Referências

Batista, E. M., Silva, G. B., and Silva, T. R. (2023). Diversidade de Gênero em Projetos Open Source no GitHub. In *Computer on the Beach*.

- Beyer, S. (2014). Why Are Women Underrepresented in Computer Science? Gender Differences in Stereotypes, Self-Efficacy, Values, and Interests and Predictors of Future CS Course-Taking and Grades. *Computer Science Education*, 24(2-3):153–192.
- Carillo, K., Huff, S., and Chawner, B. (2014). Understanding Contributor Behavior within Large Free/Open Source Software Projects: A Socialization Perspective. In *International Conference on Information Systems (ICIS)*. Association for Information Systems.
- Cheryan, S., Ziegler, S. A., Montoya, A. K., and Jiang, L. (2017). Why Are Some STEM Fields More Gender Balanced Than Others? *Psychological Bulletin*, 143(1):1–35.
- D’Alessandro, M. (2021). UnB Notícias - Programa de tecnologia voltado para a inclusão de grupos sub-representados ganha prêmio internacional. Disponível em: <https://noticias.unb.br/112-extensao-e-comunidade/4949-programa-de-tecnologia-voltado-para-a-inclusao-de-grupos-sub-representados-ganha-premio-internacional>. Acesso em: 14 mar. 2026.
- Dasgupta, N. and Stout, J. G. (2014). Girls and Women in Science, Technology, Engineering, and Mathematics: STEMing the Tide and Broadening Participation in STEM Careers. *Policy Insights from the Behavioral and Brain Sciences*, 1(1):21–29.
- Feller, J., Fitzgerald, B., Hissam, S., and Lakhani, K. (2002). *Perspectives on Free and Open Source Software*. MIT Press.
- Ferreira, I. V. (2020). Language barriers for open source contributors — CHAOSScon EU 2020. Disponível em: <https://www.youtube.com/watch?v=iRfqALpVsyE>. Acesso em: 08 mar. 2026.
- Goldman, A., Kon, F., Silva, P. J. S., and Yoder, J. W. (2004). Being Extreme in the Classroom: Experiences Teaching XP. 10:5–21.
- Guzman, E., Fischer, R., and Kok, J. (2023). Mind the gap: gender, micro-inequities and barriers in software development. *Empirical Software Engineering*.
- Hechtl, C., Joblin, M., and Apel, S. (2025). Is perceived gender related to contributions and standing in open-source software projects? *Empirical Software Engineering*.
- Lee, A. et al. (2019). FLOSS Participants’ Perceptions about Gender and Inclusiveness: A Survey. In *Proceedings of the 41st International Conference on Software Engineering (ICSE)*. IEEE/ACM.
- Nafus, D. (2012). ‘Patches don’t have gender’: What is not open in open source software. *New Media & Society*, 14(4):669–683.
- Novaes, T., Soares, L., Neto, A., Setti, M., and Emer, M. (2025). Desafios e Motivações de Mulheres na Computação - Análise de Entrevistas de um Podcast. In *Anais do XIX Women in Information Technology*, pages 207–217, Porto Alegre, RS, Brasil. SBC.
- Oliveira, D. S. (2025). *Metodologia BOSS: Formação, Permanência e Autonomia de Grupos Minorizados em Software Livre*. Edição Independente, Brasil.
- Pinto, G. H. L., Filho, F. F., Steinmacher, I., and Gerosa, M. A. (2017). Training Software Engineers Using Open-Source Software: The Professors’ Perspective. In *2017 IEEE 30th Conference on Software Engineering Education and Training (CSEE&T)*, pages 117–121.

- Prana, G. A. A., Sharma, A., Bhat, T., Zimmermann, T., et al. (2022). Including Everyone, Everywhere: Understanding Opportunities and Challenges of Geographic Gender-Inclusion in OSS. *IEEE Transactions on Software Engineering*, 48(10):3867–3884.
- Steinmacher, I., Gerosa, M., Conte, T. U., and Redmiles, D. F. (2019). Overcoming Social Barriers When Contributing to Open Source Software Projects. *Computer Supported Cooperative Work (CSCW)*, 28(1):247–290.
- Steinmacher, I., Silva, M. A. G., Gerosa, M. A., and Redmiles, D. F. (2015). A systematic literature review on the barriers faced by newcomers to open source software projects. *Information and Software Technology*, 59:67–85.
- Steinmacher, I., Treude, C., and Gerosa, M. A. (2018). Let Me In: Guidelines for the Successful Onboarding of Newcomers to Open Source Projects. *IEEE Software*, 36(4):41–49.
- Trinkenreich, B., Wiese, I., Sarma, A., Gerosa, M., and Steinmacher, I. (2022). Women’s Participation in Open Source Software: A Survey of the Literature. *ACM Transactions on Software Engineering and Methodology*, 31(4):81:1–81:37.
- UNESCO (2012). Women in Science. Technical report, UNESCO Institute for Statistics.
- Ying, K. M., Pezzullo, L. G., Ahmed, M., Crompton, K., Blanchard, J., and Boyer, K. E. (2019). In Their Own Words: Gender Differences in Student Perceptions of Pair Programming. In *Proceedings of the 50th ACM Technical Symposium on Computer Science Education (SIGCSE ’19)*, pages 1053–1059, New York, NY, USA. ACM.