

Privacy by designers: software developers' privacy mindset

Hadar, I., Hasson, T., Ayalon, O., Toch, E., Birnhack, M., Sherman, S., & Balissa, A. (2017). Privacy by designers: software developers' privacy mindset. *Empirical Software Engineering*, 23(1), 259–289. doi:10.1007/s10664-017-9517-1 (<https://doi.org/10.1007/s10664-017-9517-1>)

1 Fichamento de Conteúdo

Privacy by Design (PbD) é uma abordagem à Engenharia de Sistemas que leva em conta a privacidade em todas as etapas de desenvolvimento. A proteção de dados pessoais é um tópico relevante na comunidade de software, por isso, este artigo busca entender a interpretação de desenvolvedores a respeito de privacidade. Para isso, foi feita uma pesquisa qualitativa com programadores classificados de acordo com sua experiência e área que passaram por uma entrevista semi-estruturada, sendo perguntados a respeito da definição de privacidade de informação, ambiente de privacidade organizacional e práticas adotadas pelos profissionais referentes à privacidade. Foi concluído que os desenvolvedores possuíam uma percepção limitada à ameaças de terceiros. Além disso, eles são constantemente desencorajados a priorizá-la e, muitas vezes, não possuem o conhecimento suficiente de seu conceito. Para que *PbD* se torne um conceito viável, é necessário que haja uma mudança para que sua implementação seja bem-sucedida.

2 Fichamento Bibliográfico

Privacy by design: Abordagem à Engenharia de Sistemas que leva em conta a privacidade em todas as etapas de desenvolvimento;

Social Cognitive Theory: Teoria que defende que o aprendizado de um indivíduo pode estar diretamente relacionado a observar outros através de interações sociais;

Fair Information Practice Principles: Princípios que buscam trazer diretrizes mais concretas para o conceito de privacidade.

3 Fichamento de Citações

1. “We took a qualitative research approach, which has advantages when aiming to explore and understand complex socio-technical processes”

2. “Participants tended to discuss privacy as a social concern, based on norms of morality and ethics, rather than a technological, engineering concern.”
3. “A recurring phenomenon was participants’ interpretation of privacy as a theoretical, abstract, and impractical – perhaps even naïve – concept.”

Enhancing the Student Learning Experience in Software Engineering Project Courses

Marques, M., Ochoa, S. F., Bastarrica, M. C., & Gutierrez, F. J. (2018). Enhancing the Student Learning Experience in Software Engineering Project Courses. *IEEE Transactions on Education*, 61(1), 63-73. doi:10.1109/te.2017.2742989

1 Fichamento de Conteúdo

As faculdades de computação buscam preparar os estudantes para trabalhar na indústria, ensinando-os conceitos técnicos fundamentais. No entanto, muitas vezes as habilidades interpessoais, como liderança, tomada de decisão e negociação deixam de ser aprendidas pelos alunos. Constantemente, instituições buscam novos métodos de ensino, como o uso de processos ágeis. Este artigo é um estudo de caso durante nove semestres consecutivos em cursos de desenvolvimento de projetos que utilizam processos de software para ensinar. Para isso, foi utilizado o *Reflexive Weekly Monitoring*. O projeto concluiu que as sessões de monitoramento auxiliaram os estudantes a melhorar sua coordenação, efetividade e senso de pertencimento.

2 Fichamento Bibliográfico

Reflexive Weekly Monitoring: Método concebido para monitorar times de desenvolvimento, ensinado na *Universidad de Chile*. Consiste em sessões semanais de monitoramento onde os colaboradores do projeto procuram entender a respeito de sua performance e do status do projeto;

Agile: Métodos ágeis buscam resolver problemas do desenvolvimento de software através do esforço colaborativo entre os times de desenvolvimento e seus clientes;

Scrum: Processo ágil de desenvolvimento de software feito para times pequenos, onde os colaboradores dividem suas tarefas em objetivos que podem ser completados em iterações chamadas *sprints*.

3 Fichamento de Citações

1. “The RWM method is led by monitors who perform weekly monitoring sessions with a software development team to facilitate members’ understanding of their own performance and project status.”
2. “The team reflection is designed to make students reflect on possible causes of the issues identified in the team and project diagnosis conducted in the previous step.”
3. “RWM’ benefits are similar to those of other approaches, like the use of coaches, Scrum masters, and e-portfolios. However, RWM sums up the benefits of all of these using disciplined processes and without generating tension between students.”

Lean Learning Applying Lean Techniques to Improve Software Engineering Education

Chatley, R., & Field, T. (2017). Lean Learning - Applying Lean Techniques to Improve Software Engineering Education. 2017 IEEE/ACM 39th International Conference on Software Engineering: Software Engineering Education and Training Track (ICSE-SEET). doi:10.1109/icse-seet.2017.5 (<https://doi.org/10.1109/icse-seet.2017.5>)

1 Fichamento de Conteúdo

Construir um plano de ensino que reflete as práticas utilizadas no mercado é difícil. É bem comum ouvir a respeito da lacuna entre o que é aprendido em universidades e no “mundo real”. Este artigo é um estudo de caso que busca demonstrar como foi desenvolvido um programa que busca diminuir esta distância entre os mundos, centrado em ferramentas e técnicas presentes no dia-a-dia de um desenvolvedor profissional. O principal desafio foi ensinar engenharia de software de uma maneira que permita dar feedback e orientações de qualidade aos alunos. O programa busca, nos primeiros anos, passar por diversos paradigmas e linguagens de programação, além de vários tópicos da ciência da computação. Após o biênio inicial, são introduzidos assuntos a respeito de *software design pattern*.

2 Fichamento Bibliográfico

Design Pattern: Solução para um problema que ocorre frequentemente dentro de um contexto em projeto de software;

Kanban: Sistema ágil para gerir tarefas, formado por colunas de cartões que sinalizam o status dos afazeres;

Continuous Delivery: É conseguir enviar todo tipo de alteração em um projeto para produção, ou para os usuários, de maneira segura, rápida e sustentável.

3 Fichamento de Citações

1. “...Academics and postgraduate researchers often work on their own, and so often do not have experience of planning and managing the work of many different contributors to a software project...”
2. “It seems futile to teach software engineering before learning to program, so the first year of Imperial’s degree places a great deal of emphasis on programming ‘in the small’.”
3. “Team working is an essential component of any software engineering programme and is a key skill that many employers look for when hiring graduates.”

Research on the Influencing Factors of User Continuance Intention in Mobile Video Live Broadcast Platforms: Based on the Perspective of User Experience

Hong, J., & Tang, L. (2019). Research on the Influencing Factors of User Continuance Intention in Mobile Video Live Broadcast Platforms: Based on the Perspective of User Experience. 2019 International Joint Conference on Information, Media and Engineering (IJCIME). doi:10.1109/ijcime49369.2019.00075 (<https://doi.org/10.1109/ijcime49369.2019.00075>)

1 Fichamento de Conteúdo

Fazer o usuário consumir conteúdo continuamente é extremamente importante para a indústria de *streaming* de vídeos. Este artigo usa, da perspectiva de experiência de usuário, o *Technology Acceptance Model* para construir um modelo teórico dos fatores que influenciam a continuidade de usuários nestas plataformas. Para isso, foram selecionadas plataformas populares de *streaming* e foi feito um

questionário para coletar dados relevantes e, ao total, foram coletados 411 questionários válidos. Concluiu-se que os usuários não costumam ficar nestas plataformas por muito tempo e UX tem muita influencia na lealdade à plataforma,junto com variáveis externas como usuários e comunidade.

2 Fichamento de Bibliografia

User Experience: Abrange todos os aspectos da interação do usuário com um produto; Experiência deste cliente ao longo da interação;

Technology Acceptance Model: É uma teoria que procura modelar como usuários passam a aceitar e usar uma tecnologia;

3 Fichamento de Citações

1. “Cross table analysis results show that users with different education levels and different monthly incomes have significant differences in the user experience level of mobile video live broadcast platforms.”
2. “In terms of direct impact, the user experience has a significant positive impact on the continuance use intention of the mobile video live broadcast platform, indicating that the level of user experience will directly affect the users’ loyalty to the product.”
3. “At present, there are not many researches on the mobile video live broadcast and the user continuance intention. ”

The effect of UI, UX and GX on video games

Kristiadi, D. P., Udjaja, Y., Supangat, B., Prameswara, R. Y., Warnars, H. L. H. S., Heryadi, Y., & Kusakunniran, W. (2017). The effect of UI, UX and GX on video games. 2017 IEEE International Conference on Cybernetics and Computational Intelligence (CyberneticsCom). doi:10.1109/CYBERNETICSCOM.2017.8311702 (<https://doi.org/10.1109/CYBERNETICSCOM.2017.8311702>)

1 Fichamento de Conteúdo

Este *paper* busca entender os efeitos da interface e experiência de usuário em sua experiência de *Gameplay*. Foi feita uma pesquisa com 60 participantes de diferentes gêneros e idades, respondendo um questionário de 10 perguntas a respeito da interface, *character design* e *character customization*, utilizando o jogo *Tekken*

7. Concluiu-se que há uma importância da interface e experiência em relação ao *Gameplay*, em questões de customização de personagem, disposição de itens no menu e design visual dos personagens jogáveis.

2 Fichamento de Bibliografia

User Interface: O meio onde ocorre a interação humano-computador;

Gameplay Experience: É a maneira na qual jogadores interagem com um jogo;

User Experience: Abrange todos os aspectos da interação do usuário com um produto; Experiência deste cliente ao longo da interação.

3 Fichamento de Citações

1. “The interesting thing is from 52 respondents 69.2%(36 respondents) are male and 46.2% (24 respondents) between age 21-30, 34.6% (18 respondents) have been playing Tekken games, 71.2% (37 respondents) have ever played tekken 7 console version”
2. “Nowadays video games already at the pinnacle of human dreams, combining with new technology discovery, there’s always new innovation for video games.”
3. “In the first time of playing the new game, the player experiences unfamiliarity and challenge and starts to explore the game. After enough trying and replay the game to get the point where they’re used to the game, they’re not feel any longer to play that game anymore.”

The Dark (Patterns) Side of UX Design

Gray, C. M., Kou, Y., Battles, B., Hoggatt, J., & Toombs, A. L. (2018). The Dark (Patterns) Side of UX Design. Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems - CHI '18. doi:10.1145/3173574.3174108 (<https://doi.org/10.1145/3173574.3174108>)

1 Fichamento de Conteúdo

A experiência de usuário é um tópico cada vez mais relevante para o desenvolvimento de software. Regras e diretrizes são constantemente pesquisadas e propostas para a melhoria da interação humano computador. Este conhecimento, no entanto, pode também ser utilizado para induzir o usuário ao erro: conhecidas como *Dark*

Patterns. Este artigo é uma análise de problemas éticos frequentemente realizados por *UX Designers* que acabam manipulando a interação do cliente. Ao longo do *paper*, são mostrados itens re-ordenados propositalmente, como uma *anti-pattern* para confundir o usuário, inversão de hierarquias e forçando ações.

2 Fichamento de Bibliografia

Human-Computer Interaction: Interação humano-computador. Estuda o design e o uso da tecnologia focado em interfaces de interação entre pessoas e computadores;

User Experience: Abrange todos os aspectos da interação do usuário com um produto; Experiência deste cliente ao longo da interação.

3 Fichamento de Citações

1. “While not all interactions that take on these strategies are necessarily equally ”dark” in terms of design intent and motivation, they do have the potential to produce poor user outcomes, or force users to interact in ways that are out of alignment with their goals.”
2. “While there has been substantial interest in critical dimensions of HCI, often through the lenses of social justice, feminism, and care, these discussions have frequently remained at a highly abstracted level. What is lacking, perhaps, is a more explicit discussion of applied ethics in the pragmatist tradition.”
3. “Both strands are critical to stimulate a reflexive conversation regarding what kinds of ethical standards are necessary and appropriate for the nature of HCI and UX outputs, and the responsibility of the designer in motivating these outputs as technologies and systems are appropriated and modified by users and stakeholders alike as they define the future of social systems.”

How gamification motivates: An experimental study of the effects of specific game design elements on psychological need satisfaction

Sailer, M., Hense, J. U., Mayr, S. K., & Mandl, H. (2017). How gamification motivates: An experimental study of the effects of specific game design elements on psychological need satisfaction. *Computers in Human Behavior*, 69, 371–380. doi:10.1016/j.chb.2016.12.033 (<https://doi.org/10.1016/j.chb.2016.12.033>)

1 Fichamento de Conteúdo

A gamificação é a implementação de elementos de jogos digitais em contextos reais. Com o propósito de recompensar os usuários ao longo da sua interação com o sistema, esta técnica aparece de diversas maneiras em diversos *softwares*. Para saber o quanto *Gamification* motiva os usuários, foi feito um estudo clínico randomizado controlado utilizando um ambiente de simulação *on-line* e analisado a efetividade dos elementos nas necessidades psicológicas. Os resultados mostram que rankings, medalhas e gráficos de performance afetam a satisfação, enquanto avatares, histórias e companheiros de equipe afetam experiências de relações sociais.

2 Fichamento de Bibliografia

Gamificação: Aplicação de elementos de *game-design* em contextos que não sejam de jogos digitais;

Self-determination theory: É uma teoria que sustenta a ideia que pessoas são guiadas por necessidades de autonomia e também buscam maneiras de adquirir competências e relacionamentos positivos nas relações sociais;

Game Design: Arte de aplicar design e técnicas estéticas para criar um jogo.

3 Fichamento de Citações

1. “Points provide the player with granular feedback, which can be directly connected to the actions of the player. Performance graphs visually indicate the player’s progress over time, thereby providing sustained feedback. Badges and leaderboards assess a series of player actions and in doing so provide cumulative feedback.”
2. “All gamification elements that provide players with specific feedback on their performance should evoke feelings of competence. Thus, we expected that badges, leaderboards, and performance graphs would foster experiences of competence.”
3. “One way of generating meaning within a game is to use a story that embeds the game environment within a larger, fictional narrative. Thus, a story can evoke volitional engagement and subsequent experiences of autonomy in regard to task meaningfulness.”

Learner experiences of a blended course incorporating a MOOC on Haskell functional programming

Dale, V. H., & Singer, J. (2019). Learner experiences of a blended course incorporating a MOOC on Haskell functional programming. *Research in Learning Technology*, 27. <https://doi.org/10.25304/rlt.v27.2248>

1 Fichamento de Conteúdo

As universidades estão, cada vez mais, buscando incluir um curso online aberto massivo (*MOOC*). Este artigo monitorou as experiências de alunos em um curso deste tipo, buscando ensinar programação funcional através da linguagem *Haskell*. Foi feito também um questionário composto por 6 perguntas, além de acompanhamento de um grupo de foco. Percebeu-se que os alunos elogiaram a flexibilidade do curso, permitindo que aprendessem cada um em seu ritmo. No entanto, a maior frustração foi a falta de contato ao vivo com professores e companheiros de turma, além da falta de aulas práticas.

2 Fichamento de Bibliografia

MOOC: Curso *online* com acesso aberto e ilimitado via *web*;

Programação funcional: Paradigma de programação onde programas são construídos através da aplicação e junção de funções. É um paradigma *declarativo*, em contraste com os paradigmas *imperativos*;

Haskell: Linguagem de programação puramente funcional.

3 Fichamento de Citações

1. “One of the issues of incorporating a MOOC into a higher education course is the need to balance the needs of local learners with those of a massive cohort with varying experience of the subject.”
2. “The major benefit to learners in this study was the flexibility offered by the blended format, in terms of the first part of the course being delivered online.”

3. “We considered that interacting with a global community of learners was a required skill of programming graduates, who would – in practice – be interacting professionally on online networks to help source solutions to complex problems.”