

Padrões de Projeto em Java: Um Estudo Prático sobre a Utilização e Benefícios

Marina Gabriela do Amaral Santos, Maurício R. de A. Souza, Eduardo Figueiredo

Laboratório de Engenharia de Software (LabSoft) – Universidade Federal de Minas Gerais (UFMG) – Belo Horizonte – Brasil

{marina.santos, mrasouza, figueiredo}@dcc.ufmg.br

***Abstract.** Design patterns are defined as reusable solutions to recurring problems. These solutions have many expected benefits for the project, such as ease of communication, maintainability, and organization. Design patterns also bring a common vocabulary to the development team. However, it is necessary for the project to be designed for the correct application of design patterns and organizations do not always provide time for such. This paper presents the results of a survey conducted with Java software developers in order to investigate their knowledge, incentives, and difficulties in the adoption of design patterns. As a result, we observed great influence of the company's culture and adopted process practices.*

***Resumo.** Os padrões de projeto são definidos como soluções reusáveis para problemas recorrentes. Essas soluções visam diversos benefícios para o projeto, tais como: facilidade de comunicação, uma vez que os padrões de projeto trazem um vocabulário comum; facilidade de manutenção e organização do projeto. No entanto, é necessário que o projeto seja modelado para a aplicação correta dos padrões de projeto e nem sempre as organizações disponibilizam tempo para tal modelagem. Este artigo apresenta os resultados de um questionário aplicado à desenvolvedores de software Java com o objetivo de investigar seus conhecimentos, incentivos e dificuldades no uso de padrões de projeto. Como resultado, observou-se grande influência da cultura da empresa e seus processos adotados.*

1. Introdução

De acordo com Gamma e colegas (1995), os padrões de projeto são descrições de objetos e classes comunicantes que são customizadas para resolver um problema geral de projeto num contexto particular. Um padrão possui quatro elementos essenciais (Gamma et al., 1995): nome do padrão, problema, solução e conseqüências. O nome do padrão é um identificador que podemos usar para descrever o problema do projeto, suas soluções e as conseqüências em uma ou duas palavras. Isso torna mais fácil para pensar em projetos, comunicá-los e as suas vantagens e desvantagens para outros. O problema, por outro lado, descreve quando aplicar o padrão. Ou seja, ele explica o problema e seu contexto. O problema pode descrever os problemas de design específicos, tais como a forma de representar algoritmos como objetos. Algumas vezes o problema inclui uma lista de condições que precisam ser atendidas antes para que se faça sentido aplicar o padrão.

Além do nome e problema, outros dois elementos essenciais são a solução e as consequências. A solução descreve os elementos que fazem parte do desenho, seus relacionamentos, responsabilidades e colaborações. A solução não descreve um projeto concreto particular ou implementação. Isso porque o padrão é como um esboço que pode ser aplicado em diferentes situações. Finalmente, as consequências são os compromissos e resultados de aplicar o padrão. Embora as consequências sejam muitas vezes mudas quando descrevemos decisões de projeto, elas são críticas para avaliar alternativas de projeto e para a compreensão dos custos e benefícios da aplicação do padrão (Cardoso e Figueiredo, 2015) (Garcia et al., 2005).

Este artigo apresenta os resultados de um questionário aplicado a 34 profissionais de desenvolvimento de software atuantes na indústria, visando investigar o nível de conhecimento destes sobre padrões de projeto e sua utilização no desenvolvimento de sistemas. A ferramenta para coleta de informações utilizada neste estudo foi o questionário [Wohlin et al., 2000]. Os questionários se destacam pela popularidade e abrangência de utilização, sendo eles usados desde votações a definição de requisitos em software. O questionário é um método para coletar informação de pessoas acerca de suas idéias, sentimentos, planos, bem como origem social, educacional e outros fatores humanos. Além disso, uma pesquisa por meio de questionários é utilizada para identificar características de uma ampla população de indivíduos, comumente associada ao uso de questionários para a coleta dos dados (Easterbrook et al., 2008).

A partir da aplicação do questionário, procurou-se entender de desenvolvedores de software atuantes na indústria sobre os principais fatores que incentivam e que dificultam o uso de padrões de projeto, e a influência da colaboração da equipe, da alta gestão e de processos na utilização dos padrões. Espera-se o entendimento de como estes aspectos humanos e organizacionais podem influenciar em práticas e decisões técnicas durante o desenvolvimento de software, enquanto atividade colaborativa. Os resultados deste estudo mostram que aproximadamente 65% dos desenvolvedores concordam que lideranças da empresa, tais como gerentes e coordenadores de equipe, incentivam a utilização de padrões de projeto. Por outro lado, os principais fatores apontados que dificultam a adoção de padrões de projeto são prazos curtos dos projetos e a falta de conhecimento a respeito dos padrões.

Além desta seção introdutória, este artigo está organizado da seguinte forma. Na seção 2 são analisados os benefícios obtidos com o uso de padrões de projeto. A seção 3 descreve a metodologia de execução do questionário. Na seção 4, são apresentados e discutidos os resultados obtidos neste estudo. A seção 5 discute as ameaças à validade do estudo. Por fim, na seção 6 são realizadas as considerações finais e apontadas direções para trabalhos futuros.

2. Padrões de projeto: benefícios esperados

O objetivo principal dos padrões de projeto é propor soluções reusáveis para problemas recorrentes. Assim, a adoção de padrões de projeto pode trazer diversos benefícios para o projeto ou organização (Cacho et al., 2014) (Cardoso e Figueiredo, 2015) (Gamma et al., 1995). Esta seção lista e descreve alguns dos principais benefícios esperados pela adoção de padrões de projeto, tais como aqueles associados à facilidade de manutenção e organização do projeto. Por exemplo, em relação a facilidade de manutenção, os

padrões de projeto prezam pelo baixo acoplamento entre as classes e a alta coesão. Com isso, extensões e alterações possam ser feitas sem grandes traumas. Além deste benefício esperado, padrões de projeto defendem a padronização da estrutura das classes. Desta forma, eles podem contribuir também para a organização da estrutura geral do sistema.

Gamma (1995) enumera alguns pontos sobre o que se esperar de um padrão de projeto: (i) fornecer um vocabulário comum do projeto, (ii) um auxílio para a documentação e o aprendizado, (iii) um acréscimo aos métodos existentes e (iv) um alvo para a refatoração. Em relação ao primeiro ponto, padrões de projeto fornecem um vocabulário comum, permitindo que os designers o usem para se comunicar, documentar e explorar alternativas. Os padrões de projeto fazem um sistema parecer menos complexo, permitindo que você fale sobre ele em um nível mais alto de abstração do que em uma notação de design ou linguagem de programação.

Conhecer os padrões de projeto também torna mais fácil o entendimento de sistemas existentes (Cacho et al., 2006) (Garcia et al., 2005) (Gamma et al., 1995). A maioria dos grandes sistemas orientados a objetos utiliza padrões de projeto. Pessoas aprendendo programação orientada a objetos muitas vezes se queixam de que os sistemas que estão trabalhando usam a herança de forma complicada e que é difícil seguir o fluxo de controle. Em grande parte, isso é porque eles não entendem os padrões de projeto no sistema. Aprender esses padrões de design vai ajudar a entender sistemas orientados a objetos existentes.

Os padrões de projeto mostram como usar técnicas essenciais, tais como objetos, herança e polimorfismo. Padrões de projeto fornecem uma maneira de descrever mais do "porquê" de um projeto e não apenas registrar os resultados de suas decisões. A aplicabilidade, consequências e implementação dos padrões de projeto ajudam a guiar nas decisões do que é necessário fazer. Finalmente, de acordo com Opdyke e Johnson (1990), um dos problemas no desenvolvimento de softwares reutilizáveis é que muitas vezes ele tem de ser reorganizado ou refatorado. Padrões de projeto ajudam a determinar como reorganizar um projeto e eles podem reduzir a quantidade de refatoração que é preciso fazer mais tarde.



Figura 1 - Ciclo de vida de software orientado a objetos

O ciclo de vida do software orientado a objetos tem várias fases. Algumas destas fases são: prototipação, expansão e consolidação; conforme Figura 1 adaptada de Foote (1992). A necessidade contínua de satisfazer mais requisitos, juntamente com a necessidade de maior reutilização, faz o software orientado a objetos passar por repetidas fases de expansão e consolidação – de expansão à medida que novos

requisitos são atendidos, e de consolidação à medida que o software se torna mais genérico. Um ponto importante a se observar no uso de padrões de projeto é que o tempo gasto na identificação e estruturação dos padrões dentro do projeto é inicialmente maior que o desenvolvimento da solução diretamente sem uso destes padrões. No entanto, o ganho com a manutenção do código, um de seus principais benefícios, assim como com a clareza de entendimento para os próximos profissionais a manter ou evoluir o sistema é grande.

3. Metodologia

O método de pesquisa questionário é quantitativo, a pesquisa deve ser planejada pelo pesquisador e a aplicação deve estar ligada aos objetivos da pesquisa (Wohlin et al., 2000). A aplicação é desejada quando o pesquisador pretende investigar o que, porque, como ou quanto se dá determinada situação. Por outro lado, através de questionários não é possível determinar variáveis dependentes e independentes. A pesquisa por meio de questionários dá-se no momento presente ou recente e trata situações reais do ambiente. De maneira complementar, um estudo experimental completo poderia, por exemplo, adquirir a informação inicial utilizando um questionário, elaborar uma teoria através de um experimento controlado e verificar a teoria proposta em condições reais por meio de um estudo de caso (Wohlin et al., 2000).

Para este estudo, o questionário atua como base para levantar as informações necessárias que permitam analisar o conhecimento de desenvolvedores de software Java acerca de padrões de projeto e seu efetivo uso. Este estudo visa identificar quais fatores encorajam ou desencorajam o uso destes padrões no desenvolvimento diário de aplicações em organizações. Sendo assim, as questões de pesquisa que guiam este estudo podem ser enumeradas como a seguir.

- **QP1:** *Qual o grau de utilização de padrões de projetos, segundo a visão de profissionais em empresas de software?*

O objetivo desta questão é identificar a utilização dos padrões de projeto pelo desenvolvedor na organização.

- **QP2:** *A adoção de padrões de projeto é influenciada pelo processo de desenvolvimento de software?*

Partindo do princípio que o processo de desenvolvimento de uma empresa define as etapas que serão seguidas para execução de um projeto de software, visa-se identificar o quanto o processo de desenvolvimento influencia na utilização de padrões de projeto.

- **QP3:** *Quais os principais fatores que dificultam a adoção de padrões de projetos em organizações desenvolvedoras de software?*

Dados os principais fatores que usualmente dificultam o desenvolvimento de software, o objetivo é avaliar as dificuldades encontradas pelos desenvolvedores que os desencoraja a utilizar padrões de projeto.

- **QP4:** *Quais os principais benefícios esperados pela adoção de padrões de projeto, na perspectiva de profissionais de organizações desenvolvedoras de software?*

Identificados os fatores que influenciam a utilização ou não de padrões de projeto, visa-se esclarecer o entendimento dos desenvolvedores quanto aos benefícios que seriam alcançados com a utilização.

A partir destas questões de pesquisa, a elaboração do questionário foi realizada em seis etapas, ilustradas Figura 2. Em cada uma delas, o questionário foi alvo de sucessivas revisões e aplicações, garantindo que suas perguntas adequadas o atendimento dos objetivos da pesquisa e para o entendimento da população alvo de participantes. A Figura 2 mostra que, inicialmente, foi elaborado o questionário piloto após revisão com pesquisadores e profissionais. Este questionário foi revisado posteriormente para avaliar a coerência das perguntas com o estudo a ser realizado e uma versão final do questionário foi definida. O questionário final foi confeccionado e disponibilizado na plataforma *Google Form*¹, e dispõe de um total de treze questões, sendo sete questões para de caracterização do *background* dos participantes, e seis questões relacionadas às questões de pesquisa definidas.

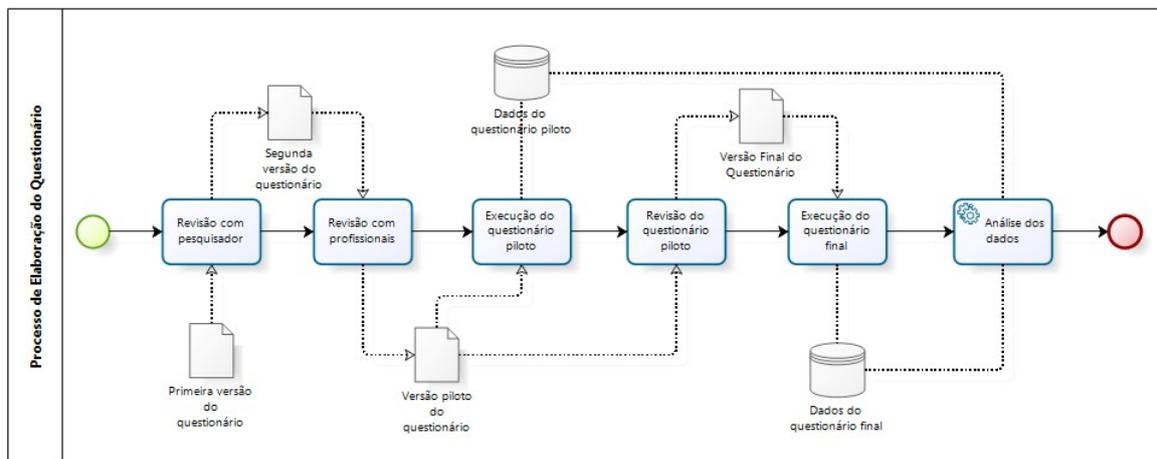


Figura 2 - Processo de Elaboração do Questionário

Uma característica fundamental do questionário é a seleção de uma amostra representativa de uma população bem definida e as técnicas de análise de dados usadas para generalizar a partir dessa amostra para a população (Easterbrook et al, 2008). Para este estudo, foi utilizada uma pequena amostra, visando não à generalização, mas a caracterização de dificuldades encontradas por um grupo de profissionais. No entanto, esta pesquisa pode ser estendida a um grupo maior de profissionais. Assim, poderá ser inferida a generalização dos resultados. Desta forma, o questionário contou com a participação de 34 profissionais de organizações desenvolvedoras de software.

4. Análise dos dados

Os dados obtidos a partir dos itens do questionário são analisados nesta seção. As ferramentas utilizadas para a análise dos dados foram *Google Form*¹, que provê suporte para análise e quantificação dos dados obtidos. Uma planilha eletrônica foi utilizada para sumarização dos dados e criação de gráficos.

¹ <http://www.google.com/Forms>

4.1. Caracterização dos participantes

Em relação aos participantes deste estudo, foram selecionados apenas desenvolvedores atuantes em organizações. Ou seja, o público alvo deste estudo é desenvolvedor de software em atividade. O tamanho da organização dos participantes variou bastante, como pode ser observado conforme Figura 3. Entretanto, a Figura 3 mostra que quase 60% dos participantes pertencem a organizações de grande porte, com mais de 100 funcionários. Quanto a sua formação acadêmica, 45% dos participantes são graduados e 55% são pós-graduados.

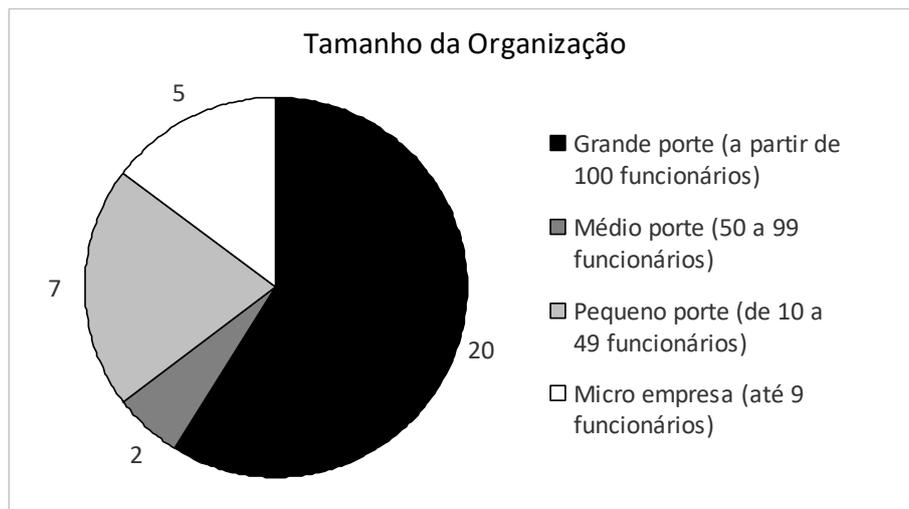


Figura 3 - Tamanho da organização dos participantes

Foi avaliado também o conhecimento dos participantes quanto a quatro aspectos relevantes ao estudo: programação orientada a objetos, modelagem de software (UML), reutilização de software e padrões de projeto. A distribuição dos conhecimentos pode ser observada no gráfico de barras da Figura 4. Este gráfico mostra que, na grande maioria das áreas de conhecimento questionadas, os participantes se classificaram como experientes ou com conhecimento moderado. Nenhum participante afirmou desconhecer qualquer dos quatro aspectos questionados.

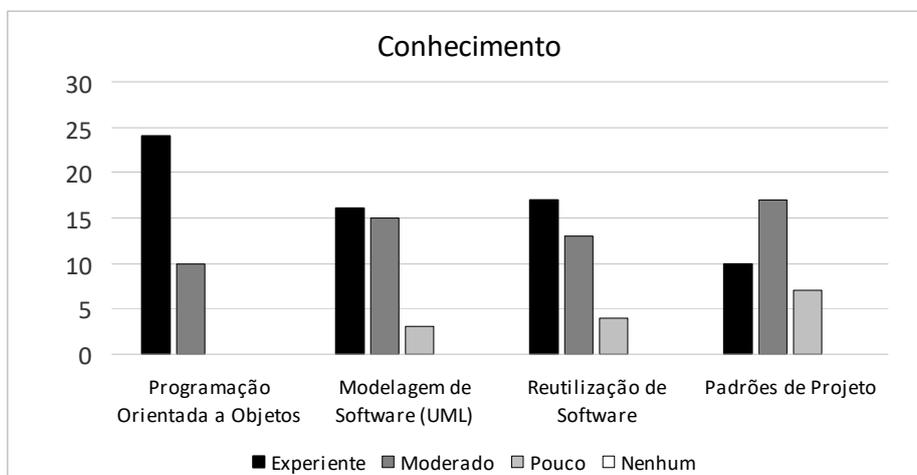


Figura 4 - Nível de conhecimento dos participantes

4.2. Influências na utilização de padrões de projeto

Para que fosse possível avaliar a utilização ou não dos padrões de projeto, foi solicitado aos participantes que avaliassem a seguinte afirmação: "Com que frequência você utiliza padrões de projeto no desenvolvimento de software na sua empresa?" Nessa questão, treze participantes afirmaram que utilizam frequentemente, dezesseis participantes afirmaram que utilizam às vezes, três afirmaram que raramente são utilizados e dois participantes afirmaram que nunca utilizam.

Em seguida, os participantes avaliaram a afirmação: "A liderança (alta gestão, gerente de projetos, líder técnico, etc.) incentiva a utilização de padrões de projeto." O resultado é apresentado na Figura 5. Como podemos observar por esta figura, vinte e dois participantes concordaram totalmente (7) ou parcialmente (15) com a afirmação. Em contrapartida, doze discordaram totalmente (6) ou parcialmente (6).

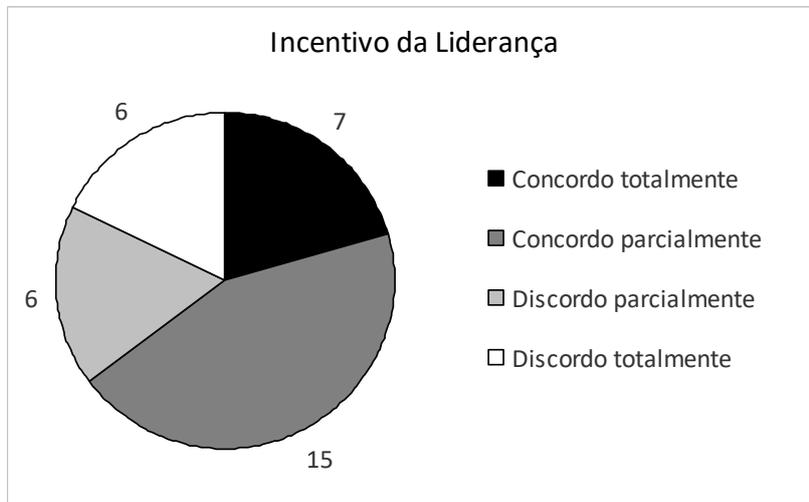


Figura 5 - Opinião dos participantes sobre o incentivo da liderança no uso de padrões de projeto

A partir da resposta da questão anterior, podemos observar a grande influência da liderança na utilização de padrões de projeto. Assim, foi solicitado aos participantes que enumerassem dentre algumas opções pré-definidas, quais os principais fatores que dificultam a utilização de padrões de projeto nos projetos em que eles trabalham. O resultado pode ser observado na Figura 6.

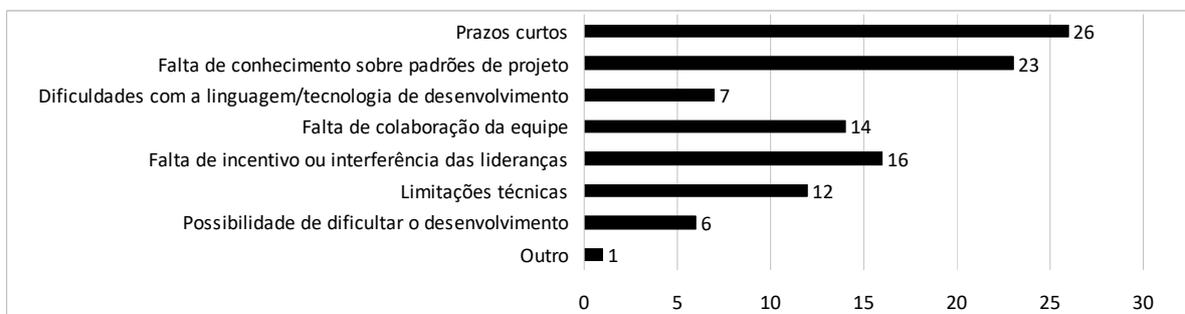


Figura 6 - Principais fatores que dificultam a utilização de padrões de projeto

Como mostra a Figura 6, na opinião dos participantes, os prazos curtos e a falta de conhecimento sobre padrões de projeto são os principais fatores que dificultam a utilização de padrões de projeto. A cultura da empresa conforme discutido anteriormente engloba os processos de software adotados, a dinâmica de desenvolvimento de software e solução de problemas. Se o foco da empresa é entregar um software com o menor tempo de desenvolvimento sem considerar manutenções e evoluções futuras, muito provavelmente os padrões de projeto serão descartados pelo líder técnico do projeto. Como foi mencionado anteriormente, o prazo inicial de desenvolvimento de software utilizando padrões de projeto é maior que o prazo onde a solução é desenvolvida diretamente. Com isso, os desenvolvedores destacaram que o curto prazo para desenvolvimento também é um dos principais fatores que dificultam a utilização dos padrões. Além destes dois fatores principais, foi citada também a falta de incentivo ou interferência das lideranças da empresa. Novamente, este resultado reforça a importância da liderança na adoção de padrões de projeto.

Após a avaliação dos principais fatores que dificultam a utilização, foi questionado aos participantes sobre os benefícios que seriam alcançados com a utilização de padrões de projeto. Os benefícios mencionados pelos participantes são apresentados na Figura 7. De acordo com a maioria dos participantes, os três maiores benefícios obtidos a partir da utilização de padrões de projeto são: (i) facilitar na manutenção e evolução do código, (ii) dar apoio na solução de problemas e (iii) favorecer a reutilização de soluções. Em seguida, é destacada a facilitação no entendimento do código e diminuir o tempo de desenvolvimento.

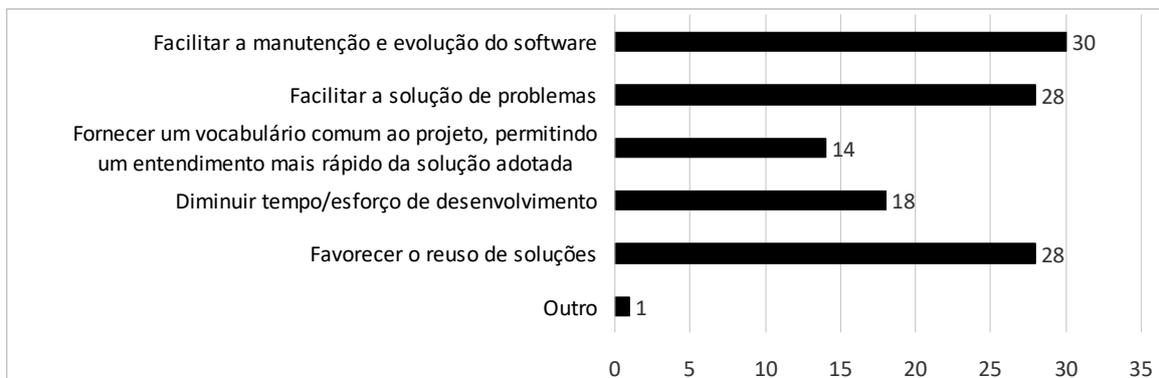


Figura 7 - Percepção dos participantes acerca dos benefícios obtidos com a utilização de padrões de projeto

5. Ameaças à Validade

Conforme apresentado na Seção 3 deste artigo, foi analisada uma pequena amostra de 34 desenvolvedores de software. Esta amostra não pode ser considerada para a generalização da situação em que os padrões de projeto não são aplicados, em geral, devido à cultura e dos processos de desenvolvimento de software das organizações. No entanto, a aplicação deste questionário a um grupo maior de participantes, com organizações diversas, bem como seus processos de desenvolvimento, permitirá com que seja inferida uma generalização acerca do uso de padrões de projeto na prática das organizações.

6. Considerações Finais

Neste artigo foram apresentados os resultados de um questionário realizado com 34 profissionais com experiência em desenvolvimento de software. O questionário teve o propósito de identificar, segundo a percepção dos participantes, quais os principais fatores que influenciam na utilização de padrões de projeto nas organizações e seus benefícios.

Os resultados do questionário fornecem indícios de que a cultura das organizações e seus processos de desenvolvimento são os grandes influenciadores na utilização de padrões de projeto. Os benefícios que seriam obtidos são bem claros para os participantes e os mais destacados são associados à manutenção e evolução do código, solução de problemas e a reutilização de soluções. Apesar de ter sido observado o entendimento dos profissionais sobre os benefícios que seriam obtidos pela adoção de padrões de projeto, os resultados indicam os padrões podem não estar sendo de fato aplicados devido ao curto prazo e a falta de incentivo das organizações.

Em questão a falta de incentivo das organizações, por exemplo, acredita-se que melhores resultados no projeto seriam obtidos se houvesse antes do início do desenvolvimento, uma fase de especificação técnica. Esta fase, que aconteceria logo após a finalização da especificação de casos de uso, avaliaria tecnicamente a estrutura do projeto e quais os padrões mais adequados a serem utilizados. A especificação técnica seria realizada por um arquiteto de software que produziria para os desenvolvedores, a estrutura-modelo a ser seguida no desenvolvimento com o esboço dos padrões a serem seguidos. Com isso, no início do desenvolvimento, o profissional teria de forma clara, a estrutura a ser seguida para a solução do problema. Assim, o desenvolvedor se preocupa com a implementação das regras de negócio do sistema e não em estruturar as classes da melhor forma para resolver o problema.

Como trabalhos futuros, pretende-se investigar mais a fundo como as organizações incentivam a adoção dos padrões de projeto e como os times são afetados por este fator. Assim, planejamos a replicação desta pesquisa em outros contextos e com desenvolvedores de variados perfis.

Agradecimentos

Este trabalho recebeu apoio financeiro das agências de fomento CAPES, CNPq e FAPEMIG.

Referências

- Fatimah Mohammed Alghamdi and M. Rizwan Jameel Qureshi (2014). "Impact of Design Patterns on Software Maintainability." *International Journal of Intelligent Systems and Applications*, 6.10: 41.
- Brian Foote (1992). "A Fractal Model of the Lifecycles of Reusable Objects". *OOPSLA Workshop on Reuse*. Vancouver, Canada.
- David Budgen (2013). "Design Patterns: Magic or Myth?" *IEEE Software*, vol.30, no. 2, pp. 87-90.

- Nelio Cacho, Claudio Sant'Anna, Eduardo Figueiredo, Francisco Dantas, Alessandro Garcia, and Thais Batista (2014). "Blending design patterns with aspects: A quantitative study". *Journal of Systems and Software (JSS)*, vol. 98, pp. 117-139.
- Nelio Cacho, Claudio Sant'Anna, Eduardo Figueiredo, Alessandro Garcia, Thais Batista, and Carlos Lucena (2006). "Composing Design Patterns: A Scalability Study of Aspect-Oriented Programming". In proceedings of the 5th International Conference on Aspect Oriented Software Development (AOSD), pp. 109-121. Bonn, Germany.
- Bruno Cardoso and Eduardo Figueiredo (2015). "Co-Occurrence of Design Patterns and Bad Smells in Software Systems: An Exploratory Study". In proceedings of the Brazilian Symposium on Information Systems (SBSI). Goiania, GO.
- Bruno Cardoso and Eduardo Figueiredo (2014). "Co-Occurrence of Design Patterns and Bad Smells in Software Systems: A Systematic Literature Review". In proceedings of the Workshop on Software Modularity (WMod), co-allocated with CBSOft. Maceio, Brazil.
- S. Easterbrook et al. (2008). "Selecting Empirical Methods for Software Engineering Research". Guide to advanced empirical software engineering. Springer-Verlag. p. 285-311, 2008.
- Erich Gamma, Ralph Johnson, Richard Helm, John Vlissides (1995). "Design Patterns: Elements of Reusable Object-Oriented Software". Pearson Education India.
- Alessandro Garcia, Cláudio Sant'Anna, Eduardo Figueiredo, Uirá Kulesza, Carlos Lucena, and Arndt von Staa (2005). "Modularizing Design Patterns with Aspects: A Quantitative Study". In proceedings of the 4th International Conference on Aspect Oriented Software Development (AOSD), pp. 3-14.
- William F. Opdyke and Ralph E. Johnson (1990). "Refactoring: An Aid in Designing Application Frameworks and Evolving Object-Oriented Systems". In SOOPPA Conference Proceedings, pages 145–161.
- Wohlin, C., Runeson, P., Höst, M., Ohlsson, M. C., Regnell, B., and Wesslén, A. (2000) "Experimentation in Software Engineering: an Introduction." Kluwer Academic Publishers.