

Gradle para Geração de Multi-Applicativos Móveis: Um Estudo de Caso em uma Universidade

Ramon Abilio¹, Cristiano M. Garcia¹, João A. Silva¹

¹ Diretoria de Gestão de Tecnologia da Informação (DGTI)
Universidade Federal de Lavras (UFLA)
Lavras – MG – Brasil

Resumo. *Com o crescimento no uso de Smartphones, diversos aplicativos tem sido desenvolvidos. Em 2015, a Universidade Federal de Lavras iniciou o desenvolvimento de aplicativos móveis institucionais e percebeu retrabalho da equipe no desenvolvimento e manutenção de aplicativos com características semelhantes. Então, conceitos de Linha de Produtos de Software foram estudados e foi encontrado um meio de aplicar tais conceitos no âmbito de aplicativos utilizando o Gradle. O objetivo deste trabalho é mostrar a utilização do Gradle para geração de multi-aplicativos, de forma a evitar retrabalho, viabilizando o desenvolvimento de uma família de produtos.*

1. Introdução

Smartphones estão, dia após dia, mais presentes na vida das pessoas. Diversas atividades, como acesso a *Internet Banking*, redes sociais, e-mail e apontar problemas na iluminação pública têm sido facilitadas por meio de aplicativos para dispositivos móveis como *smartphones* e *tablets*. Em ambientes acadêmicos, informações como consulta a locais de aula, calendários e notas de atividades acontecem com frequência alta. Dessa forma, um aplicativo móvel permitiria o acesso rápido e prático a tais informações e serviços acadêmicos.

Nesse contexto, a Diretoria de Gestão de Tecnologia da Informação (DGTI) da Universidade Federal de Lavras (UFLA) verificou a viabilidade técnica da disponibilização de informações e serviços acadêmicos, como os mencionados, em um aplicativo móvel, proporcionando integração dinâmica entre comunidade acadêmica e órgãos administrativos da UFLA. O aplicativo foi desenvolvido entre 2015 e 2016, sendo lançado em 2016 [Araujo et al. 2016].

A primeira ideia foi desenvolver um único aplicativo que atendesse três segmentos específicos dentro da Universidade: estudantes, técnicos e docentes. Com o desenvolvimento das primeiras funcionalidades de estudantes e docentes, percebeu-se que o tamanho do *Android Package Kit*¹ (APK) estava ultrapassando 20 MB. Com isso, discutiu-se as seguintes questões: i) "Manter um único aplicativo para todos os nichos, ou separar as funcionalidades em aplicativos específicos para cada nicho?"; e ii) Um aplicativo acadêmico com tamanho de 20 MB é grande? Foi tomada como premissa que 20 MB é pouco considerando-se um aplicativo popular como o do Facebook²; porém é muito para um aplicativo acadêmico.

¹Pacote usado pelo Android para distribuição e instalação de aplicativos móveis

²<http://www.facebook.com/>

Diante dessa discussão, decidiu-se pela divisão do aplicativo em três e percebeu-se que eles deveriam compartilhar características básicas e ter características específicas de cada segmento. No entanto, seria oneroso para uma mesma equipe manter três aplicativos com essas características sem utilizar recursos que permitissem a minimização de retrabalho. Optou-se, então, pela utilização de métodos, técnicas e ferramentas de Engenharia de Software para criar *softwares* semelhantes a partir de um conjunto de funcionalidades relacionados ao desenvolvimento de Linhas de Produto de Software (*Software Product Lines - SPL*) [Pohl et al. 2006].

O Android Studio – ambiente de desenvolvimento integrado utilizado pela DGTI – disponibiliza um mecanismo, chamado Gradle, que auxilia na criação de produtos com funcionalidades compartilhadas, e outras exclusivas, e podem ser gerados aplicativos específicos os ambientes de desenvolvimento e produção. O objetivo deste trabalho é demonstrar a utilização do Gradle no desenvolvimento de uma linha de produtos. A principal contribuição deste trabalho é apresentar uma forma de evitar retrabalho e cópias desnecessárias de códigos de um aplicativo para outro, o que dificulta, principalmente, manutenção, testes e desenvolvimento de funcionalidades compartilhadas.

Este trabalho está organizado da seguinte forma: na Seção 2, são apresentados o ambiente de desenvolvimento integrado Android Studio e o Gradle. Na Seção 3, são discutidos os procedimentos metodológicos do trabalho; na Seção 4, são apresentados e discutidos os resultados e, por último, na Seção 5 são apresentadas as conclusões.

2. Android Studio e Gradle

Android Studio³ é um ambiente de desenvolvimento integrado (*Integrated Development Environment - IDE*) para a plataforma Android disponibilizado e mantido pela Google. A IDE foi anunciada em maio de 2013 como substituto ao *Eclipse Android Development Tools* (ADT), iniciativa anterior do Google para desenvolvimento de aplicativos Android. O Android Studio pode ser utilizado nos principais sistemas operacionais (Windows, Linux e macOS), é disponibilizado gratuitamente sob a licença Apache 2.0, e sua versão atual é a 2.3.0 (março de 2017).

O Android Studio disponibiliza um conjunto de ferramentas que permitem a criação de aplicativos para qualquer dispositivo Android e facilitam o processo de criação, desde a codificação até o teste do aplicativo. Um desses recursos é o *Instant Run*, que acelera consideravelmente os ciclos implementação, compilação, execução. Esse recurso envia alterações tanto de implementação quanto dos recursos utilizados ao aplicativo sendo executado em um dispositivo ou emulador, de forma que as alterações possam ser visualizadas imediatamente. A IDE também conta com um editor de código inteligente, e sugestões para conclusão de código, fácil refatoração e análise da qualidade de código favorecem a escrita de códigos melhores e em menos tempo. A adição de código ao projeto ainda é facilitada por uma série de assistentes do Android Studio que permitem a utilização de modelos de códigos padrões e importação de códigos disponibilizados pela Google no GitHub⁴.

Ademais, o sistema de compilação da IDE é robusto e flexível, de forma que a inclusão de bibliotecas e geração de diversas versões do aplicativo a partir de um único

³<https://developer.android.com/studio>

⁴<https://github.com/>

projeto também é facilitada com a ajuda do Gradle. O Gradle é um *plugin* para gerenciamento de *build* lançado em 2009 e atualmente sua versão é a 3.4.1. Diferentemente de seus antecessores (Maven, Ant e Ivy) que usam XML para a definição do processo de *building*, o Gradle usa uma linguagem específica de domínio (*domain specific language - DSL*) baseada em Groovy. Ainda assim, a utilização de repositórios Maven, Ant e Ivy é possível por meio da recuperação dessas dependências.

No caso do Android Studio, a configuração do processo de *building* no Gradle é realizado em arquivos do Gradle e também em *scripts* em linguagem Groovy. A utilização do *plugin* em um projeto possibilita a utilização de recursos para a criação do aplicativo. Por exemplo, o Gradle fornece um conjunto de configurações pré definidas, de forma que a construção do aplicativo possa ser feita com o mínimo de esforço do desenvolvedor. Mudanças nos arquivos de configuração do Gradle só são necessárias se as configurações padrão não atenderem as necessidades do projeto. Além disso, o Gradle facilita o gerenciamento das dependências do projeto, o processo de assinatura dos APKs gerados e possibilita o controle de utilização ou não do ProGuard (ferramenta do Android Studio que otimiza, diminui e ofusca o *byte code* Java, com o objetivo de deixar o aplicativo mais eficiente e menos suscetível a engenharia reversa).

Além disso, o Gradle permite a criação de variantes customizadas de um aplicativo (*product flavors*), com a adição de poucas linhas de código. As diferenças entre variantes podem ser visuais (ícones, cores, textos...) e/ou funcionais. Um exemplo de uso desse mecanismo são os aplicativos que possuem uma variante paga e uma gratuita, mas com menos funcionalidades. Outro exemplo de uso do recurso é a possibilidade de gerar variantes do aplicativo direcionadas a diferentes dispositivos, de acordo com o nível da API (*Application Programming Interface*), que pode variar entre as versões do Android, ou em outras variações dos mesmos.

3. Metodologia

Este trabalho foi realizado na Coordenadoria de Sistemas de Informação (CSI), da Diretoria de Gestão de Tecnologia da Informação, da Universidade Federal de Lavras (UFLA). O desenvolvimento do aplicativo iniciou-se por meio de um trabalho de conclusão de curso de graduação e passou a ser mantido pela equipe da CSI desde o lançamento em setembro de 2016. Essa equipe é composta essencialmente pelos autores deste trabalho. O ambiente de desenvolvimento é composto pelas seguintes ferramentas:

- Android Studio com Gradle para desenvolvimento e testes;
- Git⁵ para versionamento de código;
- Redmine⁶ para gerência de tarefas;
- *Smartphones* com o sistema operacional Android.

O trabalho foi conduzido entre agosto/2016 e março/2017 da seguinte forma:

- Estudo sobre Linha de Produtos de Software (SPL) e seus conceitos;
- Estudo do Gradle, sua documentação e suas aplicações práticas, além de realização de testes com suas funcionalidades;
- Identificação das funcionalidades básicas e específicas esperadas de cada um dos aplicativos;

⁵<https://git-scm.com/>

⁶<http://www.redmine.org/>

- Refatoração do código;
- Testes com simuladores e dispositivos móveis pela equipe da CSI;
- Disponibilização dos aplicativos MinhaUFLA Estudante e MinhaUFLA Técnicos para testes pelos usuários.

4. Resultados

As variantes do aplicativo MinhaUFLA para estudantes, técnicos e docentes foram criadas com o auxílio do plugin do Gradle no Android Studio. O *plugin* também é utilizado no processo de *building* utilizado nos ambientes de desenvolvimento e produção. A Figura 1 apresenta exemplos da definição das variantes customizadas do aplicativo MinhaUFLA no arquivo de configuração do Gradle. Os *product flavors* para a variante de estudante e docente são definidas analogamente ao *flavor* tecnico, enquanto o processo de *building* para o ambiente de produção é configurado de forma semelhante ao *flavor* desenvolvimento.

```

tecnico {
    dimension = "grupo"
    applicationId "br.ufla.app.minhafla.tecnico"
    resValue "string", "app_name", "Minha UFLA TAE"
    resValue "string", "perfil", "Servidor TAE"
}
desenvolvimento {
    applicationIdSuffix = ".desenvolvimento"
    dimension = "versao"
    resValue "string", "app_name", "dev"
    resValue "string", "url_webservice", "url_web_service"
}

```

Figura 1. Definição dos *products flavors* no aplicativo MinhaUFLA.

As principais diferenças entre as variantes do aplicativo são as funcionalidades disponíveis para cada perfil de usuários, como pode ser visualizado na Figura 2. As funcionalidades compartilhadas pelas variantes e as específicas de cada variante são:

Funcionalidades compartilhadas: consultar informações sobre o restaurante universitário (cardápio diário, número de créditos e valor do crédito); consultar itinerário do ônibus interno; acessar mapa do Campus Universitário da UFLA; consultar informações para contato com a equipe de desenvolvimento MinhaUFLA.

Funcionalidades específicas: MinhaUFLA - Docente: consultar disciplinas sob sua responsabilidade; consultar horário das aulas, turmas e local onde as aulas serão lecionadas; consultar notas dos estudantes. **MinhaUFLA - Estudante:** consultar disciplinas; consultar horário das aulas; consultar local das aulas; consultar número de créditos e notas; consultar links para acesso externo ao Calendário Acadêmico, campus virtual, pontos de carona da UFLA, circular externo e e-mail institucional. **MinhaUFLA - TAE:** consultar registro diário do ponto eletrônico.

A integração dos aplicativos com os demais Sistemas Institucionais é realizado por meio de uma arquitetura de *Web Services* de Integração [Garcia et al. 2015], que foi desenvolvido pela própria Instituição. Esta arquitetura mantém todos os sistemas da Universidade com dados comuns sincronizados, possibilita a oferta de funcionalidades a novos sistemas, é facilmente monitorável, é segura e escalável.

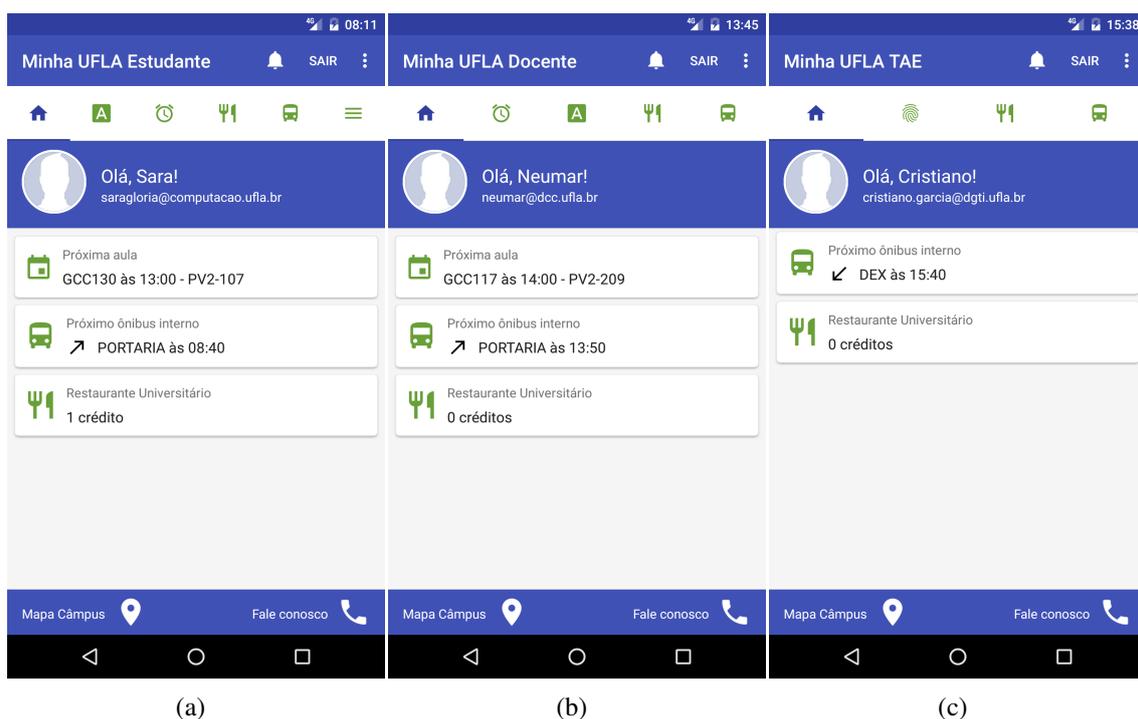


Figura 2. Versões customizadas do aplicativo MinhaUFLA.

5. Conclusão

O desenvolvimento de uma família de produtos de *software*, na qual existem características comuns e características específicas de cada variante, é desafiador do ponto de vista da manutenção desses produtos. Até o momento, a utilização do Gradle permitiu o desenvolvimento e a publicação, para testes, das variantes MinhaUFLA Estudante e MinhaUFLA TAEs. Problemas relatados pelos usuários em características comuns às duas variantes puderam ser corrigidos de maneira mais ágil, pois o código não precisou ser replicado entre projetos diferentes. A variante MinhaUFLA Docentes ainda não foi disponibilizada para testes por usuários docentes, mas encontra-se em fase de testes pela equipe da CSI.

Referências

- Araujo, S., Garcia, C. M., Abilio, R., and Malheiros, N. (2016). Acesso a Serviços e Informações Acadêmicas da Universidade Federal de Lavras em Dispositivos Móveis com a Plataforma Android. X Workshop de Tecnologia de Informação e Comunicação das Instituições Federais de Ensino Superior - WTICIFES.
- Garcia, C., Abilio, R., and Malheiros, N. (2015). Abordagens e tecnologias para integração de sistemas: Um estudo de caso na universidade federal de lavras. *Revista de Sistemas de Informação da FSMA*, (15):11–22.
- Pohl, K., van der Linden, F., and Metzger, A. (2006). Software product line variability management. In *Software Product Line Conference, 2006 10th International*, pages 219–219. IEEE.