

**DIEGO D'LEON NUNES
DIÓGENES APARECIDO REZENDE**

APLICATIVO PARA CONSULTA DE NOTAS

**UNIVERSIDADE DO VALE DO SAPUCAÍ
POUSO ALEGRE – MG**

2015

SUMÁRIO

1 DISCUSSÃO DE RESULTADOS 2

REFERÊNCIAS..... 6

1 DISCUSSÃO DE RESULTADOS

Neste capítulo serão discutidos os principais pontos em relação aos resultados obtidos com a execução desta pesquisa. Espera-se com isso elucidar algumas questões referentes ao modo como a aplicação das teorias descritas no quadro teórico desta, refletiram na prática.

O sistema operacional *Android* mostrou o porquê de ser tão utilizado nos dias atuais. Com uma gama enorme de recursos totalmente gratuitos e com a documentação excelente torna-se claro o que cada função realiza, com decorrer do desenvolvimento do aplicativo.

Como se constatou que os alunos, na maioria das vezes, acessam o portal do aluno para consultar notas, faltas e provas agendadas, o aplicativo tem como importância facilitar para que os discentes tenham suas informações de maneira fácil e rápida. É notório que é mais simples acessar esses dados pelos *smartphones* do que por *desktops*, dessa maneira, quando um professor lançar uma determinada nota, o aluno será notificado de que alguma informação nova está no portal, evitando que tenha que ficar entrando no portal várias vezes ao dia ansioso em saber sua média final.

O aplicativo é de fácil utilização, pois com a *activity* principal, do tipo *Navigation Drawer Layout*, faz com que ele fique mais atraente, uma vez que o menu fica escondido e apenas é visível se chamado pelo estudante, além de facilitar aos usuário encontrar as opções por ele desejada, pois as possibilidades de navegação encontram-se em uma lista. A seguir pode-se ver a Figura 1 do menu. Quando o usuário clicar no ícone em destaque aparecerá as opções de navegação.



Figura 1 – Menu do Aplicativo. **Fonte:**Elaborado pelos autores.

Quanto a lista de sites que aparecem na *Home*, o resultado não foi muito empolgante, pois os sites não tornaram-se responsivo, criando barras de rolagem e gerando desconforto ao usuário conforme mostra a Figura 2. Contudo há a possibilidade de abrir o site em um navegador, porém foi analisado e constatado que não seria uma boa prática fazer um usuário sair do aplicativo.



Figura 2 – Navegador Interno Do Aplicativo. **Fonte:**Elaborado pelos autores.

As informações referentes às notas, faltas e provas agendadas estão sendo apresentadas

em uma lista do tipo *ExpandableListView*, este, traz a vantagem que quando clicado em um de seus itens, é apresentado os seus itens filhos, por essa razão é desnecessário abrir uma outra *activity*, ficando mais agradável ao usuário e melhorando o desempenho do *software*. Abaixo, na Figura 3 é possível ver a *activity* de notas, listando algumas informações com *widget ExpandableListView*.

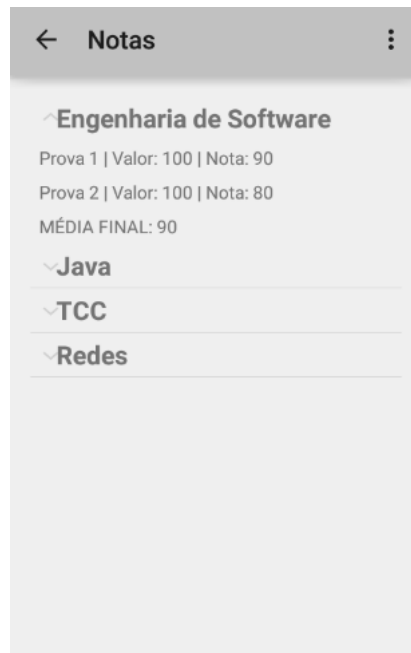


Figura 3 – Tela de Apresentação de notas, faltas e provas agendadas. **Fonte:**Elaborado pelos autores.

As informações que vem do *web service* estão sendo salvas no banco de dados *Sqlite* do aplicativo que mostra-se muito eficiente, uma vez que é rápido e leve.

O aplicativo resultado desta pesquisa, tinha necessidade de consumir dados para posteriormente apresentá-los ao usuário. Era necessário que, os dados do sistema acadêmico da instituição de ensino que serviu como contexto para esta pesquisa, fossem transmitidos de alguma forma ao aplicativo. Era necessário também que os dados chegassem ao aplicativo respeitando as particularidades de cada usuário, trazendo somente informações relevantes aos mesmos. Com esse intuito de disponibilizar informações já citadas anteriormente, a quem quer que fosse necessário, inclusive aos usuários do aplicativo, foi criado um *Web Service* REST. Este foi um dos resultados alcançados através desta pesquisa.

A construção do *web service*, de início, mostrava-se um tanto quanto custosa, devido a restrições das tecnologias que foram escolhidas. Por se tratar de uma simples *web service* que seria disponibilizado para suprir a demanda de dados do aplicativo, os primeiros serviços foram construídos e disponibilizados fazendo uso de *servlets* simples e conexão JDBC¹. Este modo

¹ JDBC - Java Database Connectivity

como foi pensado inicialmente, era simples de ser contruído e de uma *performance* aceitável.

Porém de acordo com o crescimento da demanda do serviço, tornou-se inviável a construção do mesmo com estas tecnologias, devido a complexidade com que era necessário contruir os serviços, haja vista que, com estas tecnologias era necessário que se fosse configurado praticamente tudo de forma manual inclusive tratamento de erros da aplicação, respostas as requisições e tipos de dados. Esta etapa teve, portanto, um resultado não muito amigável do ponto de vista de sua construção. No entanto se for analisado do ponto de vista do conhecimento adquirido, obteve-se um resultado satisfatório, pois, foi na pesquisa e na busca melhores alternativas a estas tecnologias, que se chegou ao resultado final.

O *web service* foi desenvolvido com algumas técnicas que facilitaram a sua construção. Foi usado o *framework Jersey* para prover os serviços necessários. Este *framework* usa *servlets* para disponibilizar os serviços, porém com a facilidade de já ter embutido em si o tratamento para qualquer um dos tipos de dados que usado na implementação de um serviço REST, tanto para entrada ou saída de dados. Foi ainda usado o *framework* de persistência de dados *Hibernate*. Este por sua vez desempenhou papel notório, pois, ao mesmo tempo que facilitou o desenvolvimento do *web service* com relação ao foco nas regras de negócio da aplicação e não tanto na implementação, pode-se dizer que se, por algum motivo for necessário migrar de banco de dados este será um fator facilitador. O resultado final foi extremamente satisfatório, pois era notório que era fácil tanto contruir e disponibilizar um novo serviço, quanto consumir o mesmo.

Para que houvesse dados para que o *web service* transmitisse para o aplicativo, era necessário receber os dados do sistema acadêmico da referida instituição, haja vista que o *web service* é independente do sistema acadêmico desta instituição. Para esse propósito foi construído um módulo que era responsável por fazer a importação dos dados necessários para a base de dados do *web service*. Este por sua vez tinha a responsabilidade de fazer a importação dos dados periodicamente, e ainda tratar os tipos de dados recebidos para tipos aplicáveis ao banco de dados local. Além disso era preciso notificar o módulo responsável por invocar o serviço *Google Cloud Messaging* para que os dispositivos dos alunos aos quais houveram atualizações nos dados, fossem notificados e fizessem acesso ao *web service* para solicitar esses dados atualizados. Pode-se dizer então, que este módulo desempenhou perfeitamente seu papel, pois, com ele era fácil manter os discentes atualizados em relação aos seus dados e ao banco de dados do *web service* também atualizado e com uma estrutura que fosse condizente com seus propósitos preestabelecido.

Portanto, conclui-se que se for analisado estes resultados citados de forma conjunta,

pode-se dizer que o resultado geral foi satisfatório, pois, os discentes puderam obter as informações de que mais fazem uso de forma plática e rápida.