

MAC0218 - Técnicas de Programação II - Entrega II

Artur Magalhães	10297734
Bruna Thalenberg	8569211
Guilherme Yambanis	8041265
Gustavo Silva	9298260
Lucas Santos	9345064
Victor Colombo	8988657

1 Descrição do trabalho já feito

1.1 Entrega I

1.1.1 Setup do ambiente de desenvolvimento

O repositório no GitHub foi criado de forma que todos os participantes do grupo tenham acesso a ele. O Rails foi instalado dentro da pasta `USPAnalytics` por meio do comando `rails new`, criando a estrutura básica do aplicativo.

O grupo se organizou de maneira a determinar *milestones* para as entregas do projeto, que podem ser verificadas na Wiki do repositório. No `README`, foi escrito um pequeno manual de instalação e utilização.

1.1.2 Modelagem do banco de dados

Grande parte do trabalho para essa entrega consistiu em investigar e entender os dados fornecidos pela Universidade, para decidir quais seriam utilizados em nosso banco de dados e como ele seria modelado. Para tal, entramos no portal da transparência da Universidade e verificamos tanto as planilhas disponíveis quanto à documentação que diz respeito a elas, além de buscarmos suporte em pesquisas externas para entender termos técnicos pouco claros.

Feito isso, a base de dados foi gerada seguindo o procedimento descrito em `database.md`. Na wiki do projeto, em "glossário do orçamento", descrevemos melhor o que cada uma das variáveis significa em português simples. Essas explicações constarão posteriormente no site.

1.1.3 Definição da arquitetura a ser implementada

Discutimos a implementação desejável do aplicativo, chegando à seguinte arquitetura:

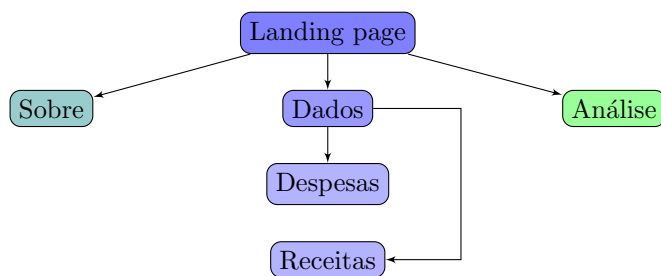
[A definição da arquitetura foi alterada na fase II. Ver nova descrição abaixo.]

A estrutura da arquitetura foi implementada ao criarmos páginas (ainda que em branco) ligadas dessa forma. Na próxima fase iremos preenchê-las com conteúdo por meio do desenvolvimento do *front-end* e sua ligação com o banco de dados.

1.2 Entrega II

1.2.1 Redefinição da arquitetura

Após maior contato com o banco de dados e definições de layout, decidimos que a arquitetura pensada inicialmente não era adequada para a aplicação. Ela foi redefinida da seguinte forma:



1.2.2 Implementação do front-end

Desenvolvemos, nesta etapa, a interface *front-end* do website para a visualização das informações. O trabalho inicial foi a integração Bootstrap<->Rails e o desenvolvimento dos *templates*. Os caminhos feitos na fase anterior foram ajustados para refletir a nova decisão de arquitetura. O conteúdo das páginas estáticas, **landing page**, **about** e **glossary** foi escrito em **html** e plugado no Rails.

Para o conteúdo principal, a página com os dados em si, utilizamos a API do Google Charts¹. Para a visualização do orçamento, utilizamos a estrutura de Treemap para as despesas, uma vez que há diversos níveis hierárquicos, e de gráfico pizza para as receitas, uma vez que há apenas uma camada de informações.

Buscamos também dar mais ênfase na programação pareada nesta fase, com os pares criando o código no mesmo computador, revezando-se no teclado. Isso se mostrou desafiador, pois muitos dos integrantes nunca haviam usado tal método, além da dificuldade inerente à realização de encontros físicos, devido a incompatibilidade de horários e deslocamento.

Entretanto, após uma fase de adaptação inicial, essa prática foi muito benéfica, principalmente no desenvolvimento do *front-end*, já que melhorou a comunicação

¹<https://developers.google.com/chart/>

efetiva de ideais, permitindo que conclusões mais duradouras pudessem ser alcançadas, principalmente em características subjetivas do projeto, como estilo e linguagem visual, diminuindo a necessidade de retrabalhar partes de código feitas por integrantes individualmente.

1.2.3 O banco de dados

Com a nova arquitetura, foi necessário fazer pequenos ajustes na modelagem do banco de dados. Mapeamos manualmente os níveis hierárquicos das despesas do orçamento e decidimos quais as variáveis de fato relevantes para o mapa que seria implementado. Adaptamos os modelos que já estavam feitos e populamos o banco de dados de acordo com o modelo.

Percebemos, no entanto, que o *treemap* possui muito mais níveis hierárquicos do que o esperado. Por isso, implementamos somente as duas primeiras camadas. As outras serão entregues na fase final do projeto.

2 Próximos passos

Para a terceira e última entrega, iremos realizar análises mais complexas dos dados, a serem integradas na seção **analysis** ou na própria página de visualização dos dados – a decidir. Buscaremos correlacionar alguns dos dados apresentados na fase dois, levantando informações relevantes e inconsistências nos dados.

Além disso, iremos finalizar a implementação do *treemap*, com os níveis hierárquicos mais baixos, que não implementamos nessa fase por conta da complexidade inesperada dos dados.

O resultado das três entregas consiste no produto final que propusemos ao início do curso.

3 Principais dificuldades

Durante esta fase, nossa maior dificuldade deu-se durante a utilização da API do Google Charts. Esse tutorial, no entanto, nos ajudou muito. Além disso, a modelagem do banco de dados, feita na fase anterior, teve que ser parcialmente refeita, por conta de novas decisões de projeto. O mapeamento das variáveis em níveis hierárquicos para a implementação do *treemap* também foi bastante trabalhoso. Em seguida, o banco de dados foi populado manualmente e, apesar de não ser algo inerentemente complicado, é um trabalho que toma tempo.

A criação das páginas estáticas fora do ambiente *Rails* não apresentou grandes dificuldades, devido ao conhecimento prévio dos participantes. As decisões de estilo, por sua subjetividade, por vezes atrapalharam o prosseguimento do projeto, sendo mais demorada a tomada de decisão do que em aspectos técnicos, como os do *back-end*. Algumas versões de *landing-page* foram criadas, até chegarmos na versão final.

A integração das páginas criadas com o Bootstrap com o *Rails* se mostrou um pouco trabalhosa, já que diversas adaptações no código, tanto do *HTML*,

quanto importações no *Rails*, tiveram de ser feitas, algum tempo foi consumido até que a integração estivesse completa.