

SiGFlux: Sistema de Gerenciamento de Fluxos.

Luiz Antônio de Faria Júnior¹

Rafael Souza Paixão¹

Daniel Facciolo Pires²

RESUMO

Este projeto tem como objetivo analisar, projetar, prototipar e implementar um software que seja capaz de gerenciar o fluxo de uma Associação Atlética Acadêmica, no qual há a necessidade de ter informações seguras e centralizadas, além do acesso por todos os membros ser imprescindível. Através de Reuniões com o atual Presidente João Pedro Souza, atual Vice-Presidente Pedro Paulo Rodrigues, ex-Presidente Leonardo Taveira e fazendo parte de equipe vôlei e futsal que representa a faculdade em jogos esportivos universitários, os autores tiveram conhecimento acerca da carência em relação a um sistema que atendesse às necessidades da Associação Atlética Acadêmica. Sendo assim, o projeto foi nomeado de SiGFlux (Sistema de Gerenciamento de Fluxo), implementado em um ambiente web, utilizamos Python como Linguagem de Programação, Framework Django, modelo MTV (Model, Template, View), Banco de Dados PostgreSQL, além de linguagem HTML e CSS. Outrossim, tem como função oferecer uma interface amigável para que o usuário possa ter acesso a relatórios, fazer pesquisas e inserir informações de maneira fácil, rápida, estável e confiável. O projeto se inicia com os módulos de controle de estoque, controle de membros e financeiro com Compras e Vendas. Foi implementado também registros de logs de qualquer informação alterada, inserida e excluída no sistema. Como conclusão, as técnicas e as metodologias da engenharia de software mostraram-se adequadas para o desenvolvimento da solução, que passa a ser uma opção de solução para as atléticas melhor gerirem seus dados para melhor tomar decisões.

Palavras Chave: Associação Atlética Acadêmica, Gerenciamento de Fluxo, Django, Python, SiGFlux.

ABSTRACT

The goal of this project is manage all the flow of an Academic Athletic Association, that need to have secure and centralized information and access to all members is fundamental. Through Meetings with current President João Pedro Souza, current Vice President Pedro Paulo, ex President Leonardo Taveira, and also being part of the team that represents the college in university sports games, We realize the lack of a system that satisfies the needs of the Academic Athletic Association. Our project was named SiGFlux (Flow Management System), We will use python as a programming language, Django Framework, PostgreSQL Database, following the methodology MTV (model, template, view) and, HTML and CSS language. Its function will be to offer an easy interface to user views a report does research and inserts informations, in an agile, easy, fast, and stable and trust way. The project will

¹ Graduado em Sistemas de Informação – Uni-FACEF

² Docente do curso de Sistemas de Informação – Uni-FACEF

start with the control of stock, members and financial with purchase and sales modules. Logs of any data inserted, altered or deleted in the system will also be implemented.

Key Words: Academic Athletic Association, Flow Management, Django, Python.

1. Introdução

De acordo com os registros, as primeiras federações esportivas surgiram no eixo Rio-São Paulo na década de 30, sendo oficializada, em 1933, a *Federação Atlética dos Estudantes* (FAE). Essa tinha o intuito de organizar e estabelecer diretrizes para o esporte universitário, tendo como objetivo principal montar equipes de modalidades esportivas para representar a universidade em campeonatos (GLOBO UNIVERSIDADE, 2012).

A Associação Atlética Acadêmica é responsável pela seleção, montagem e manutenção dos times, agendamento de treinos com treinadores especialistas para cada modalidade, inscrição em campeonatos universitários, ações sociais como visitas em creches, arrecadação de alimentos não perecíveis, arrecadação de livros entre outros. Ela é formada apenas por alunos da universidade e para conseguir arrecadar dinheiro para custear todos os gastos com viagens, materiais, aluguéis de quadras, treinadores e inscrições em campeonatos, periodicamente promove eventos, rifas e vendas de itens personalizados da atlética, para alunos, atletas e qualquer outra pessoa que se interessar. A venda destes itens também ajuda na divulgação da Atlética e da universidade.

Em uma associação atlética que não possui um sistema informatizado com suporte de software, todas as informações de atletas, fluxos de dinheiro e materiais são gerenciadas por meio de planilhas eletrônicas, tornando bem precária a forma de gerenciamento, pois não é centralizada e confiável, pode haver falta e ou redundância de informações e ainda mais grave, a perda de dados. Todos esses problemas se agravam com a saída de algum membro, que leva consigo toda sua rotina e dados que lhe era empregado sobre o departamento em que era responsável. Isso causa um impacto muito grande para a associação, uma vez que raramente tem o controle exato da quantidade de dinheiro em caixa, da quantidade de materiais disponíveis para cada modalidade, cadastro de atletas e suas informações como digitalização de sua documentação, telefone, código de matrícula entre outros.

Com essa situação, visto que também que os autores deste trabalho fazem parte de uma atlética que não possui apoio de software, nota-se a carência de um sistema para gerenciamento e centralização de dados e processos. Os autores fizeram diversas reuniões com membros da Atlética para entender como eram feitos os processos internos, quais eram as necessidades reais e que gostariam que fossem atendidas. Decidiu-se então fazer um sistema web, possibilitando assim o acesso dos usuários e membros da atlética de qualquer dispositivo e de qualquer lugar em que o usuário tenha acesso a internet.

O objetivo geral desse trabalho é apresentar uma solução de software que tem como propósito eliminar o uso de interfaces manuais e padrões pessoais para gerenciar cada tipo de informação de uma associação atlética. Além de que, visa-se como objetivos específicos otimizar o fluxo, e aumentar a qualidade e eficiência do mesmo dentro da Atlética, eliminando redundância, melhorando o controle dos processos e otimizando as tomadas decisões.

Com a implantação deste projeto, busca-se um ganho significativo na manutenção e eficácia dos processos, do controle interno das rotinas e processos da Atlético, permitindo que as informações fiquem centralizadas e seguras, e que todos os membros tenham seus determinados acessos, impossibilitando que quando um membro se desligar, estes dados pertinentes ao mesmo não sejam perdidos, auxiliando para que o presidente tenha conhecimento de quem realmente está se dedicando, tenha controle das rotinas, e a informação do caixa, do patrimônio, dos times e dos técnicos contratados.

Como procedimento metodológico para o desenvolvimento deste projeto na plataforma web foram estudados e apresentados como referencial teórico o conceito e as vantagens da utilização de um sistema ERP (*Enterprise Resource Planning*) e o conceito e arquitetura da linguagem Python juntamente com o *framework* Django. Todo o desenvolvimento seguiu os padrões especificados pela engenharia de software. Foi realizado levantamento de requisitos funcionais e não funcionais, modelagem do banco de dados, documentação de regras de negócios, documentação do software, diagramas da UML, diagrama com BPMN, casos de uso e prototipação das telas do software. Todos os artefatos gerados estão disponíveis no GitHub em <https://github.com/dfpires/sigflux>.

Sendo assim, o trabalho está estruturado em cinco capítulos. O segundo capítulo abordará alguns conceitos e fundamentos de engenharia de software. No quarto capítulo serão apresentados os artefatos construídos nas fases de análise e projeto de engenharia de software do SigFlux, como diagramas e requisitos. O quinto capítulo abordará detalhes da implementação e prototipação do software. E finalmente, no sexto capítulo serão apresentadas as considerações finais do projeto.

2. Engenharia de Software

Atualmente os profissionais da área de software vivenciam um cenário de extrema competitividade e a busca por novos segmentos de mercado. Para se destacarem, as empresas buscam mostrar pontos que realmente possam enaltecer seu aplicativo ou solução, para que este possua um diferencial no grande mercado atual, junto com um grande esforço constante de evolução. Neste cenário, tem-se que grande parte das decisões importantes são tomadas sem que haja o diálogo da gestão comercial com os profissionais de software, conseqüentemente gerando conflitos, desgastes e frustração entre as áreas administrativas e as áreas responsáveis pela execução das tarefas. (SLACK,2003; PORTER, 1986; HILL, 1993).

A engenharia de software tem o objetivo de estudar as melhores formas possíveis de desempenhar uma tarefa ou projeto, para que esta venha contribuir com o sucesso da organização onde foi estruturada, por meio do desenvolvimento da área tecnológica da empresa em questão. Usando abordagens metodológicas desenvolvidas com este intuito, que são divididas entre dois modelos: rigorosos e ágeis, estes que se diferenciam entre si pelo fato de que metodologias ágeis se baseiam em um conjunto mínimo de funcionalidades que podem ser entregues mais rapidamente para o cliente final.

Para conceituar estes modelos percebe-se que há certo senso comum entre (KRUCHTEN, 2001),(SUTHERLAND, 2001), (GLASS, 2001), (PAULK, 2001) e (SOMMERVILLE. 2003):

- As necessidades e os problemas que foram levantados, devem estar extremamente claros e estabelecidos, inclusive com a condição que o atendimento é considerado satisfatório ao escopo.
- Toda a descrição das funcionalidades do software que irá solucionar os problemas e o atendimento das necessidades, deve ser extremamente detalhado, a ponto que se possa ser feito um desenho e construção do sistema e seu respectivo teste.
- Prazos e custos do desenho, construção e testes do sistema são resultados do tamanho e da complexidade do escopo do projeto a ser implementado.

Baseando-se nessas premissas, os modelos representam a série de atividades que os participantes do desenvolvimento ou manutenção do sistema, bem como o formato de execução e ferramentas de apoio. Os participantes dialogam e trocam informações sempre que for necessário, estes meios poderão ser registrados por diversas ferramentas, variando do modelo em que foi implementado. Esta cooperação entre os participantes se torna indispensável no trabalho diário do TI quanto no contexto de negócio. Os participantes devem definir o escopo e a solução dos problemas de negócio que será implementado no sistema, levando em consideração as necessidades do negócio em questão (GLASS 2001, SOMMERVILLE 2003).

Dado estes fatos, a comunicação e cooperação efetiva se torna imprescindível entre os profissionais de negócio e os de software, independente do modelo que foi adotado. Porém, a realidade atual é completamente diferente em grande parte dos casos. As áreas de negócio decidem com base em discussões estratégicas, da qual os profissionais de software ou mesmo outras áreas raramente tem participação, por acharem que a participação dos mesmos não irá contribuir no conteúdo dessas discussões, levando que seu conhecimento é restrito a área de tecnologia. Esta prática é extremamente prejudicial na parte posterior, a implementação da estratégia, seja na área de TI ou nas áreas relacionadas a produção de bens ou serviços de manufatura. (HILL, 1993).

Os principais conflitos dessas duas áreas estão relacionados a estipulação de prazos de meta e custos de um determinado escopo que são, de certa maneira, inatingíveis pelas equipes de desenvolvimento ou produção. Mesmo que tenha ocorrido uma boa comunicação durante a aplicação dos modelos de desenvolvimento de sistemas, os parâmetros definidos pela decisão estratégica podem não ser alcançados pela equipe de desenvolvimento de software, mesmo com esforço de profissionais dedicados.

Segundo Porter (1986) e Hill(1993), um engenheiro de software pode contribuir para que uma estratégia de negócio seja elaborada considerando a visão de um profissional de TI em sua formulação. Qualquer que seja a organização e seu modelo de discussão e decisão estratégica, ela deverá chegar as seguintes definições:

- Quais são os objetivos (prazos e resultados) a serem alcançados ao final do projeto para que o mesmo alcance um diferencial no mercado;
- Quais soluções ou obstáculos a serem superados para alcançar os objetivos finais, e os riscos que podem ser associados. Elementos estes que definem em alto nível o escopo da estratégia.
- As ações a serem realizadas e os recursos que serão gastos para resolver os problemas de riscos, para que os prazos sejam cumpridos conforme estabelecidos.

Estes pontos são inter-relacionados de modo que devem ser avaliados juntos, e a organização deve escolher realidade é mais cabível ao seu ambiente atual, seja na atuação de concorrentes, fornecedores, clientes, governo, seja com relação a área financeira, gerencial etc.

A contribuição do engenheiro de software no processo de discussão e planejamento estratégico deve ter o foco em aspectos em que a área de TI afeta nas operações do negócio, tanto de processos atuais quanto processos requeridos para suportar uma dada opção estratégica sendo considerada. O foco tem que ser totalmente nos processos, para que possam ser realizadas análises e troca de ideias efetivas com as outras áreas de negócio da empresa, pois são através desses processos que serão implementadas qualquer estratégia que foi formulada. Como o resultado final saem a partir de interações com as áreas de negócio, deve-se atentar a dois objetivos, segundo Sommerville (2011):

- Decisões finais serão tomadas pelas áreas de negócio, porém deverá ser considerado os aspectos de TI (capacidade e limites) que estarão estabelecidas no escopo, prazos e custos do que será implantado, para cumprir uma dada estratégia.
- Deverá haver um acordo entre as áreas de negócio em relação ao escopo e as prioridades na construção dos elementos do projeto

Caso os engenheiros de software consigam cumprir com estes dois objetivos, conseguirão evitar os problemas mais críticos no desenvolvimento e manutenção de sistemas, que são: metas impossíveis de serem atingidas e mudanças relevantes no escopo dos sistemas.

Conforme Sommerville (2011), a engenharia de software é essencial para o funcionamento das sociedades nacionais e internacionais atuais devido ao avanço da tecnologia, principalmente, gerado pela globalização. Nesse sentido, a engenharia de software visa apoiar o desenvolvimento profissional do software, não referindo apenas ao programa em si, mas sim toda a documentação e configuração necessárias para o programa execute todas suas funções corretamente.

Com isso, os softwares a serem vendidos para um cliente podem se classificar em produtos genéricos e produtos sob encomenda, nos quais o primeiro refere-se a produtos que são produzidos por uma organização de desenvolvimento e pode ser vendido para qualquer cliente, como softwares que são ferramentas de banco de dados, enquanto os produtos sob encomenda são sistemas encomendados e desenvolvidos para um cliente em particular.

A qualidade do software, por sua vez, inclui o comportamento dele, assim como a estrutura e organização dos programas do sistema e sua documentação. Os atributos de um bom software, segundo Sommerville (2011), são apresentados na Figura 1 a seguir.

Nesse sentido, Sommerville (2011) aponta que a abordagem sistêmica utilizada na engenharia de software é denominada como processo de software, isto é, uma sequência de atividades que resulta na produção de um produto de software. As principais atividades nesse processo são:

- Especificação do software, em que clientes e engenheiros definem o software a ser produzido e as restrições de sua operação.
- Desenvolvimento de software, em que o software é projetado e programado.

- Validação de software, em que ele o software é verificado para garantir que é o que o cliente quer.
- Evolução de software, em que o software é modificado para refletir a mudança de requisitos do cliente e do mercado.

| Características do produto | Descrição |
|----------------------------|---|
| Manutenibilidade | O software deve ser escrito de forma que possa evoluir para atender às necessidades dos clientes. Esse é um atributo crítico, porque a mudança de software é um requisito inevitável de um ambiente de negócio em mudança. |
| Confiança e proteção | A confiança do software inclui uma série de características como confiabilidade, proteção e segurança. Um software confiável não deve causar prejuízos físicos ou econômicos no caso de falha de sistema. Usuários maliciosos não devem ser capazes de acessar ou prejudicar o sistema. |
| Eficiência | O software não deve desperdiçar os recursos do sistema, como memória e ciclos do processador. Portanto, eficiência inclui capacidade de resposta, tempo de processamento, uso de memória etc. |
| Aceitabilidade | O software deve ser aceitável para o tipo de usuário para o qual foi projetado. Isso significa que deve ser compreensível, usável e compatível com outros sistemas usados por ele. |

Figura 1. Característica de um bom software

Fonte: Sommerville (2011)

Sendo assim, esse processo deve incluir os produtos, ou seja, os resultados esperados em uma das atividades do processo, os papéis de cada um, refletindo as responsabilidades dos indivíduos envolvidos nos processos e as pré e pós-condições (declarações das atividades do processo antes e depois de realizado). Na prática, essas são atividades complexas, que incluem subatividades como validação de requisitos, projeto de arquitetura e testes unitários. Ademais, há atividades que são responsáveis por apoiar o processo, como documentação e gerenciamento de configuração de software (SOMMERVILLE, 2011).

Esses processos podem ser categorizados como processos dirigidos a planos, em que todas as atividades são planejadas com antecedência, o progresso é avaliado por comparação com o planejamento inicial, ou como processos ágeis, sendo esse um planejamento gradativo, incremental, que facilita alterações recorrentes de necessidade de mudança dos clientes.

Além disso, Sommerville (2011) também apresenta os modelos de processos de software, no qual primeiramente tem-se o modelo cascata, responsável por considerar as atividades fundamentais do processo de especificação, desenvolvimento, validação e evolução, representando-os como fases distintas, como: especificação de requisitos, projeto de software, implementação, teste e assim por diante, como ilustra a Figura 2 a seguir.

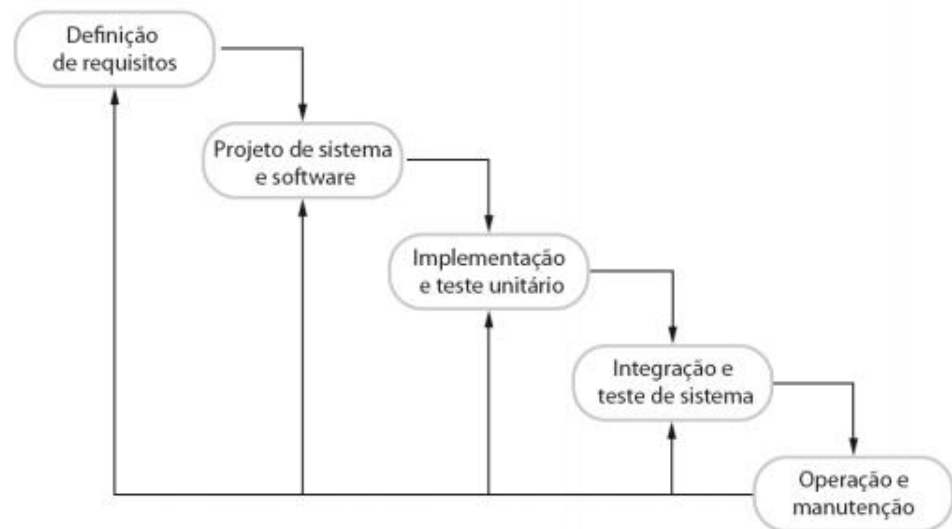


Figura 1: Modelo cascata.

Fonte: Sommerville (2011).

Há também o desenvolvimento incremental, segundo Sommerville (2011), em que nessa abordagem as atividades de especificação, desenvolvimento e validação são intercaladas, de forma que exista o desenvolvimento de diversas versões (incrementos). Cabe ressaltar que em cada versão deve ser adicionado uma funcionalidade em relação a versão anterior, como ilustra a Figura 3.

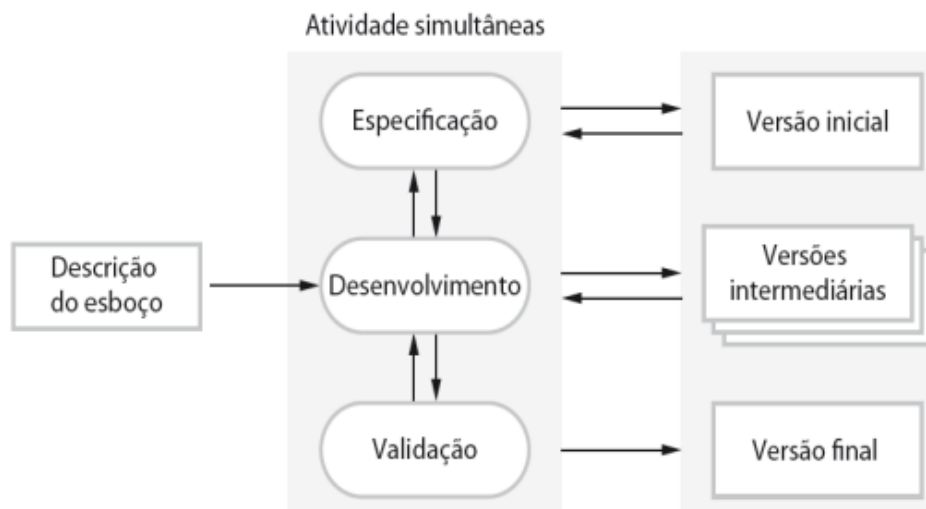


Figura 2: Desenvolvimento incremental.

Fonte: Sommerville (2011).

Por sua vez, a engenharia de software orientada a reuso refere-se a abordagem pautada em componentes reusáveis, em que o processo de desenvolvimento do sistema visa integrar esses componentes em um sistema já existente em vez de desenvolver um sistema a partir do zero. Posto isto, é importante salientar que esses modelos são não mutuamente exclusivos, sendo muitos utilizados em conjunto, principalmente, para o desenvolvimento de sistemas de grande porte (SOMMERVILLE, 2011), como ilustrado na Figura 4.

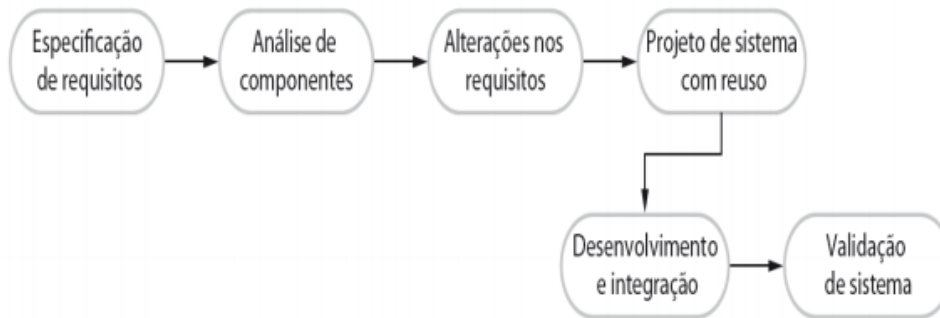


Figura 3: Engenharia de software orientada a reuso.

Fonte: Sommerville (2011).

Segundo o mesmo autor, existem vários tipos de softwares, no entanto, não há um método universal que aplica a todos, uma vez que há três aspectos principais que afetam diversos tipos de softwares:

- **Heterogeneidade:** desafio é desenvolver técnicas para construir um software confiável que seja flexível o suficiente para lidar com a heterogeneidade das operações dos sistemas, já que cada vez mais, os sistemas operem distribuídos através das redes que incluem diferentes tipos de computadores e dispositivos móveis.
- **Mudança de negócio e social:** negócio e sociedade estão mudando rapidamente, sendo preciso evoluir para que o tempo requerido para o software dar retorno a seus clientes seja reduzido.
- **Segurança e confiança:** é essencial a confiança nos softwares, uma vez que eles estão na vida cotidiana dos indivíduos, de forma que a proteção das informações seja mantida e que usuários maliciosos não consigam atacá-lo.

No entanto, é importante ressaltar que há fundamentos de engenharia de software, em que deve ser aplicado a todos os sistemas (SOMMERVILLE, 2011):

- Devem ser desenvolvidos em um processo gerenciado e compreendido: deve-se planejar o processo de desenvolvimento e ter ideias claras do que será produzido e quando será finalizado.
- Confiança e desempenho são importantes para todos os sistemas, de forma que se comporte conforme o esperado, sem falhas e deve estar disponível quando o uso for requerido.
- É importante entender e gerenciar a especificação e os requisitos de software: deve saber o que clientes e usuários esperam dele.
- Deve fazer o melhor uso possível dos recursos existentes.

Por fim, o autor aponta que é necessário compreender e definir os serviços requisitados pelo sistema, identificando suas restrições. Dessa forma, faz-se necessário a elaboração de um documento de requisitos, responsável por especificar os requisitos que os *stakeholders* apresentam. Esse processo se divide em quatro atividades principais, ilustrados na Figura 5 (SOMMERVILLE, 2011):

- **Estudo de viabilidade:** realização de uma estimativa acerca da possibilidade de se satisfazer as necessidades do usuário. Deve ser barato e rápido, sendo que seu resultado deve determinar se avança ou não com o projeto através de uma análise detalhada.
- **Elicitação e análise de requisitos:** processo de derivação dos requisitos do sistema por meio da observação dos sistemas já existentes e discussões com os potenciais usuários e compradores, análise de tarefas, ou seja, é entender o que os *stakeholders* do sistema precisam ou esperam dele.
- **Especificação de requisitos:** apresenta as informações obtidas durante a análise, definindo os requisitos com detalhes.
- **Validação dos requisitos:** verifica os requisitos quanto o realismo, consistência e completude. É importante, principalmente, devido ao fato de que os erros no documento de requisitos podem ser descobertos nessa atividade.

A figura 5 a seguir ilustra as atividades para elaboração do documento de requisitos.

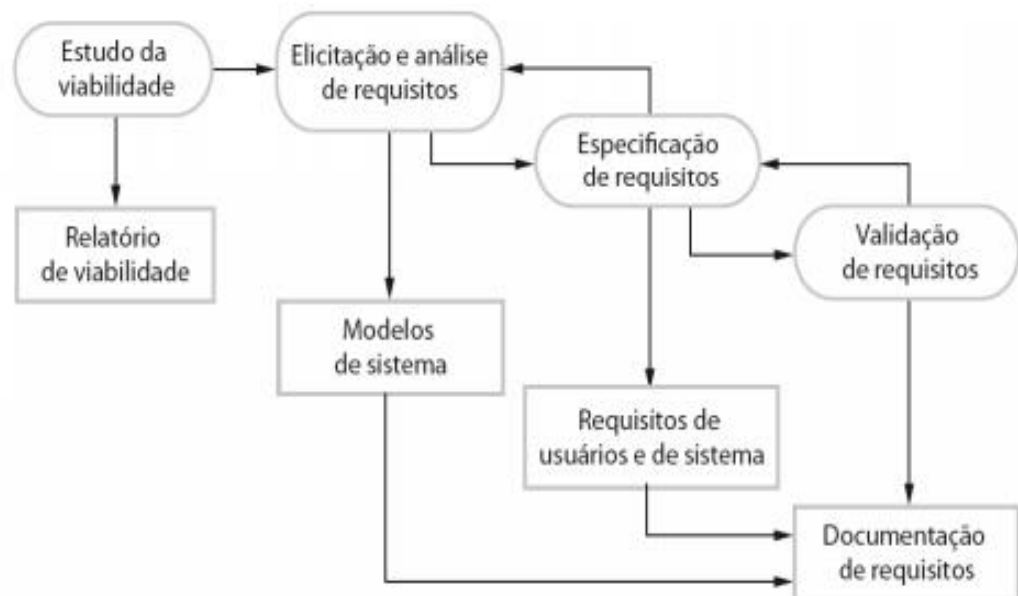


Figura 4: Atividades para elaboração do documento de requisitos.

Fonte: Sommerville (2011).

3. Empreendendo a Solução Proposta

Nesta etapa, os autores abordaram os motivos pelos quais escolheram o determinado cliente, e os meios que foram realizados para a validação da proposta, seguindo padrões que são usados por empresas multinacionais. Estas validações foram feitas baseadas no modelo Canvas, seguindo à risca sua estruturação e a seu objetivo primário.

3.1. Conceito de Empreendedorismo

Segundo Dornelas (2008), empreendedorismo é a capacidade de pessoas e processos, que por meio de uma transformação, conseguem tirar do papel uma ideia que antes era somente conceitual e fazer que desta saia uma oportunidade. Dessa

maneira, pode-se criar negócios bem estruturados e que tem grande capacidade de sucesso.

“O empreendedor é aquele que destrói a ordem econômica existente pela introdução de novos produtos e serviços, pela criação de novas formas de organização ou pela exploração de novos recursos materiais.”
Schumpeter (1949)

A ideia de Kirzner (1973) sobre o assunto é que um bom empreendedor consegue encontrar o equilíbrio claro de um ambiente de caos e turbulência, conseguindo encontrar neste ambiente uma oportunidade revigorante. Porém, os dois afirmam em suas ideias que um bom empreendedor, é construído por sua capacidade de conseguir abstrair de diversas situações oportunidades de criação ou melhora em seu ambiente de trabalho ou mundo em que vive.

Schumpeter (1949) diz que o empreendedor mais conhecido é o que cria um novo negócio, uma nova ideia, mas isso não impede de inovar dentro de negócios já existentes. Portanto, para o autor é possível empreender dentro de empresas já constituídas e que anteriormente teriam outro pensamento sobre o assunto.

Um empreendedor tem que ser capaz de analisar com frieza uma determinada situação ou empresa, estar sempre atento em novas tecnologias e informações, para que, juntando todos esses meios, consiga apresentar dados que sejam capazes de proporcionar a criação de uma forma de negócio inovadora e disruptiva. Ressalta-se que é essencial a quebra dos preceitos até então criados sobre aquele assunto, para criar algo novo, algo extraordinário, bem estruturado e com grande probabilidade de crescimento.

3.2. Startup enxuta

Startup enxuta, para Ries (2012), é um conceito usados por empreendedores para criação de novos produtos e mercados. Destaca-se que esse conceito foi criado por Eric Ries (2012), no qual tem como objetivo a criação de soluções rápidas para validar suposições de mercado, usando o *feedback* do usuário como um método para envolvimento do cliente em sua aplicação.

O principal objetivo de uma *Startup* enxuta é reduzir os custos de processos, fazendo contatos frequentes com o cliente para conseguir abstrair a real visão de mercado, validando ou retirando suposições incorretas sobre o mercado. Além disso, melhora a abordagem empreendedora, uma vez que reduz o trabalho, o custo de validação de suposições e o tempo gasto para o negócio se engajar no mercado, entregando uma abordagem de MVP (*Minimum Viable Product*), que nada mais é que o mínimo que um software ou negócio precisaria ter para atender todas as expectativas do usuário.

Este novo método de criação de empresas se baseia no processo de *Customer Development*, metodologia essa criada principalmente por Blank (2012) em *“The Four Steps to the Epiphany”* que descreve o processo para uma *Startup* conseguir encontrar o alinhamento entre produto e mercado. Nesse sentido, o primeiro passo para uma *Startup* de sucesso é conseguir abstrair as necessidades dos consumidores e focar tempo e energia para garantir que todas essas demandas sejam construídas dentro de um software ou negócio que entregue com clareza, objetividade e com o mínimo de desperdício de tempo e esforço para o consumidor final. Portanto, é importante ter sempre pensamentos que visem evoluir cada dia

mais a empresa e processos, por meio de avaliações de mercado e usuários para criação de um mercado até então não explorado.

3.3. Conceito generalizado da ferramenta Canvas

Business Model Canvas (Canvas) é um documento estruturado em somente um painel, que descreve todos os pilares fundamentais de sua empresa. Foi desenvolvido para facilitar a montagem e documentação de seu negócio. Este modelo foi baseado no livro "*Business Model Generation*" de Osterwalder (2011) e vem sendo muito usado na criação de Startups pela fácil implementação e manutenção.

Nesse contexto, o conceito tratado no livro apresenta em nove componentes os pilares de uma empresa, e conseguindo preencher todos, sua empresa estará estruturada para gerar valor no mercado. Os componentes presentes nos pilares cobrem 4 áreas principais de uma empresa de negócio: clientes, oferta, infraestrutura e viabilidade financeira, como mostrado e discutido no Quadro do documento suplementar deste artigo.

3.4. Modelo Canvas SigFlux

A Figura do modelo Canvas desenvolvido para o projeto SigFlux contendo cada um dos 9 componentes é discutido em documento disponível no github do projeto.

4. Projeto Desenvolvido

Neste capítulo são apresentados os processos de levantamento de requisitos que foram utilizados para a modelagem do sistema de acordo com a necessidade do cliente. Estes processos foram desenhados seguindo os padrões da disciplina de engenharia de software. O processo de funcionamento atual da atlética será representado por meio do BPMN este que foi desenvolvido através de várias reuniões com membros e ex-membros da atlética. Eles foram responsáveis por apresentar aos autores o funcionamento e os principais pontos a serem focados no sistema.

Desta forma, os autores passaram a modelar o sistema, dando o primeiro passo ao desenvolver o caso de uso, que é responsável pela representação de uma maneira fácil e intuitiva, para que o processo do sistema seja compreendido pelos desenvolvedores e também o cliente final. Para verificar a integridade das operações, os autores modelaram alguns fluxos dos processos que são representados através de três tipos diversos de diagramas: diagrama de sequência, diagrama de máquina de estado e diagrama de atividade.

Outras metodologias também foram usadas para a representação destes processos e rotinas, que são: levantamento de requisitos funcionais e não funcionais, regras de negócio, diagrama de classe. Também foi realizada uma análise de portabilidade, que é responsável por levantar em quais computadores ou dispositivos o sistema será executado, para que não haja problemas nesta execução.

4.1. Atlética

A Associação Atlética Acadêmica XX de Março surgiu no ano de 2015 com intuito de juntar grupos de alunos que ficariam responsáveis por organizar eventos esportivos para o Centro Universitário Municipal de Franca. Logo que foi criada, teve

muita aderência de membros, nos quais seriam divididos em áreas administrativas estruturadas em diversos cargos, alguns destes membros seriam responsáveis pela parte financeira, outros pela área de marketing, alguns ficariam responsáveis pela administração dos bens da atlética (tais como bolas, redes, uniformes, entre outros), e também pela venda de itens exclusivos da atlética que ainda seriam desenvolvidos.

A princípio, a ideia era somente a organização de eventos esportivos, tais como interclasses, disputa de campeonatos com outras faculdades entre outros eventos, mas, como foi muito grande o envolvimento de todos os alunos do Uni-Facef, o atual presidente e fundador da atlética, viu uma oportunidade de ampliar os eventos que eram organizados pela atlética, fazendo uma festa para a comemoração da criação da nova atlética da Uni-Facef. Este evento foi muito divulgado pela atlética, causando um grande interesse na faculdade, muitas pessoas compareceram e a atlética começou a ganhar seu espaço entre as atléticas da cidade de Franca.

Do outro lado, os times que eram levados para os campeonatos tiveram um desempenho que surpreendeu a todos, inclusive os próprios membros da atlética, e já no primeiro ano, o Centro Universitário Municipal de Franca, conseguiu subir para a primeira divisão da Copa Inter Atlética, superando outras faculdades que ainda não havia conseguido tal feito.

Mesmo sendo criada recentemente a XX de Março, já fez sua história em campeonatos e em eventos de Franca/SP, nunca decepcionando os alunos e integrantes, sempre visando comprometimento e responsabilidade com seus atletas e integrantes, representando muito bem o nome do Uni-Facef por onde passa.

4.2. Análise de Portabilidade

Para desenvolver o sistema, os autores escolheram utilizar a linguagem Python juntamente com o framework Django para auxiliar no desenvolvimento das telas. É um sistema web totalmente responsivo que pode ser acessado de diversos dispositivos sem que ocorra algum tipo de quebra ou falha no layout. O sistema terá versões um pouco diferentes nas versões *mobile* e *desktop*, para que a usabilidade do usuário não seja afetada, independentemente do dispositivo em que ele escolha acessar o sistema.

Maiores detalhes sobre a análise de portabilidade encontram-se no github deste projeto.

4.3. BPMN – Business Process Modeling Notation

O BPMN, *Business Process Modeling Notation*, é uma notação para modelagem BPM (*Gestão de Processos de Negócios*), no qual atualmente pode ser considerada uma das mais atuais notações para modelar processos de negócios, de acordo com Turban e Volonino (2013). Portanto, é um meio que facilita a comunicação entre a área de tecnologia da informação e a área de negócios, possibilitando a automação de processos. Destaca-se que há, atualmente, mais de 60 ferramentas responsáveis por suportar BPMN, como o Bizagi, representado pela figura a seguir.

Nesse sentido, tem-se como vantajoso o uso dessa modelagem o fato de ser possível começar a utilizá-lo com elementos básicos de fluxograma, evoluindo para elementos complexos, assim como, permite que as organizações expandem seus modelos de processos sem prejudicar as especificações existentes. Além disso, a

modelagem acarreta na redução da distância que relaciona os consultores responsáveis por mapearem processos, os técnicos que automatizam e os usuários que utilizam tal ferramenta no cotidiano. Por fim, o BPMN também possibilita a interpretação do modelo de processo e geração de códigos do sistema. O Diagrama BPMN do SigFlux está presente no GitHub do projeto.

4.4. Diagrama de Caso de Uso

De acordo com Guedes (2014), diagramas de casos de uso faz referência à todas as funcionalidades que o sistema irá oferecer ao usuário. É utilizado para representar os comportamentos pretendidos das ferramentas do sistema. No github encontra-se os casos de uso para o sistema SigFlux.

A documentação dos casos de uso é de extrema importância para o esclarecimento dos processos em que o sistema será imerso. Estes casos de uso também tem uma maneira mais detalhada de representação, que é representada nos quadros disponíveis no github do projeto.

4.5. Regras de Negócio

O GitHub do projeto contém as regras de negócio do sistema SigFlux.

4.6. Análise de requisitos

Esta seção irá apresentar a modelagem para o aplicativo de gerenciamento de fluxo SigFlux, este aplicativo veio para que os usuários possam centralizar suas informações em um único lugar, evitando o uso de planilhas e aumentando a segurança das informações. Além de validar informações inseridas, evitar perda de informações, acesso fácil de qualquer lugar, dependendo apenas de conexão com a internet e um dispositivo móvel, desktop, smartTV entre outros similares.

O público alvo deste documento deverá incluir pessoas envolvidas com desenvolvimento de sistemas com boas práticas e leitura dos padrões UML, testes do sistema e avaliação do projeto. Deverá ser testado junto com profissionais da área para verificar e validar os resultados.

4.6.1. Escopo

O objetivo do projeto é que sua funcionalidade de controles de estoque, de pessoas, e informações da instituição de ensino superior que a atlética for utilizar.

4.6.2. Benefícios esperados

Com o aplicativo a centralização da informação para todos os membros da instituição, podendo ser acessado a qualquer momento, inclusive não será perdida nenhuma informação de membros que saírem, desde que estas estejam salvas no sistema.

4.6.3. Outros artefatos produzidos

A apresentação dos requisitos funcionais, requisitos não-funcionais, matriz de rastreabilidade, modelo entidade relacionamento, diagramas de classe, de

sequencia, de máquina de estado, e de atividades, bem como uma discussão, está disponível no github do projeto.

4.6.4. Métricas

As métricas são responsáveis por mensurar um comportamento, tendência ou variável do projeto, de forma que visa facilitar a avaliação e determinação do prazo, custo e valor do projeto. A discussão das métricas construídos para o sistema SigFlux está disponível no github do projeto.

5. Implementação Desenvolvida

Neste capítulo, os autores descreveram um pouco da linguagem que foi usada no desenvolvimento do sistema e os motivos que os levaram a essa decisão, e em seguida os protótipos das telas que foram desenvolvidos para a solução, descritas detalhadamente para o melhor entendimento de todas as partes envolvidas no processo, de modo que o usuário final possa entender perfeitamente as funções que o sistema que está sendo entregue irá atuar e quais problemas ele irá sanar.

Para o desenvolvimento desta solução, os autores escolheram utilizar a linguagem Python, por conta de sua baixa curva de aprendizado e pelo fato de ser uma linguagem que está em ascensão no mercado de trabalho atual.

5.1. Prototipação

A seguir, é apresentado o desenvolvimento das telas que foram criados pelos autores, junto com uma breve descrição da funcionalidade de cada tela.

5.1.1. Página institucional da Associação Atlética Acadêmica XX de Março

Primeiramente, os autores desenvolveram um protótipo da tela institucional da Associação Atlética Acadêmica XX de Março, como ilustra a figura 6, no qual há uma barra de menu superior, figura 7, contendo os botões home, equipes, campeonatos, SiGFlux e parceiros.

A função *home* é responsável por redirecionar o usuário para o estado inicial da página, enquanto o botão equipes apresenta as equipes esportivas presentes na atlética. Da mesma forma, o botão campeonatos exhibe os campeonatos que a atlética disputa com suas respectivas equipes. Ademais, se faz presente o botão *parceiros* que expõe uma página com os parceiros e patrocinadores da empresa. Por fim, o botão *SIGFlux* direciona o usuário para uma página de login, sendo possível ter acesso restrito ao sistema.

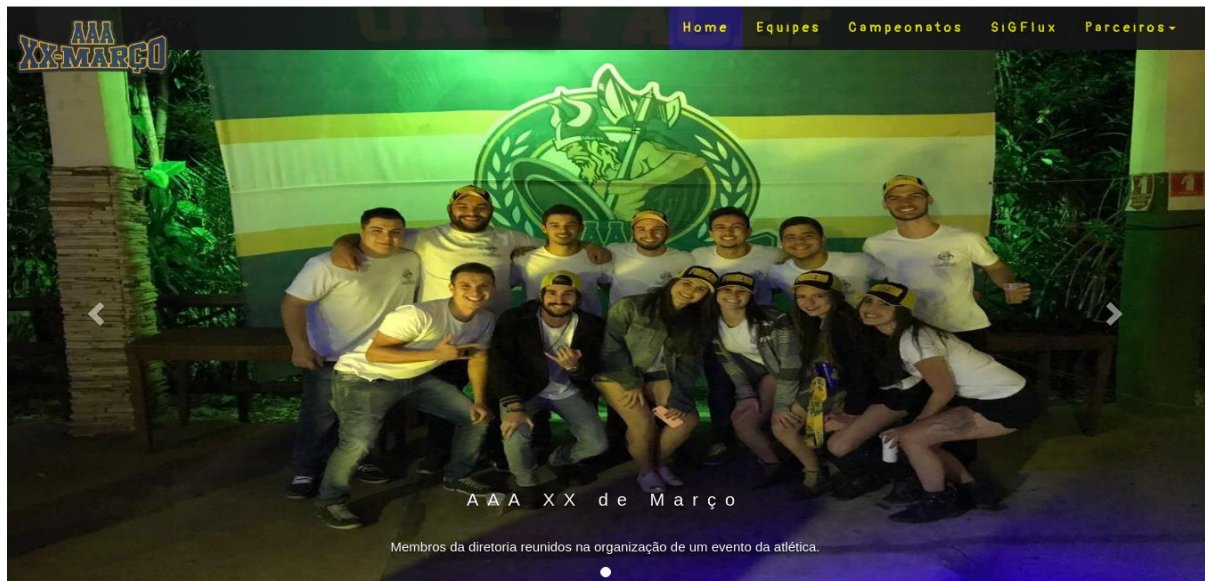


Figura 6: Página institucional da Associação Atlética Acadêmica XX de Março.

Fonte: Os autores

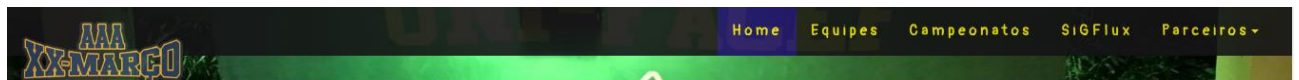


Figura 7: Menu da barra superior da página inicial.

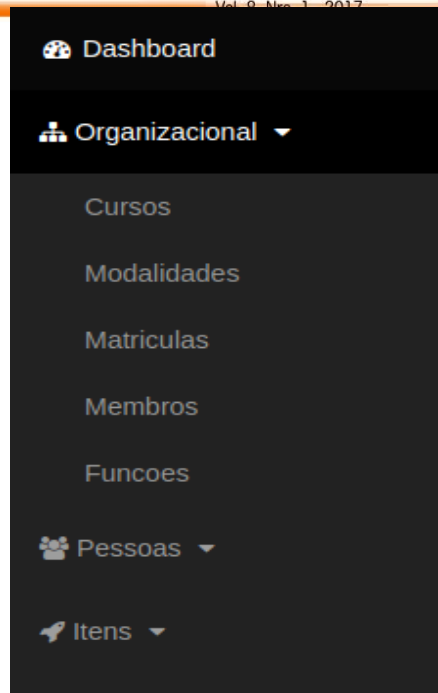
Fonte: Os autores

5.1.2. Menu do sistema

O menu do sistema SiGFlux, apresentado na figura 8, expõe ao usuário as funcionalidades do sistema, que será detalhado abaixo, da mesma forma que redireciona o mesmo as telas que ele deseja utilizar, ou seja, nessa aba o usuário poderá escolher qual sua necessidade dentro do sistema, podendo optar separadamente entre os módulos.

Portanto, o sistema possui vários menus e a partir deles submenus, por exemplo o menu organizacional apresenta os submenus: cursos, modalidades, matriculas, membros e funções, como ilustra a figura 9. Ressalta-se que esses menus e submenus estão separados nos módulos, *organizacional*, *pessoas* e *itens*, sendo que cada módulo tem seu submenu com todas as suas específicas funcionalidades.

Figura 8: Menu do sistema SiGFlux.



Fonte: Os autores

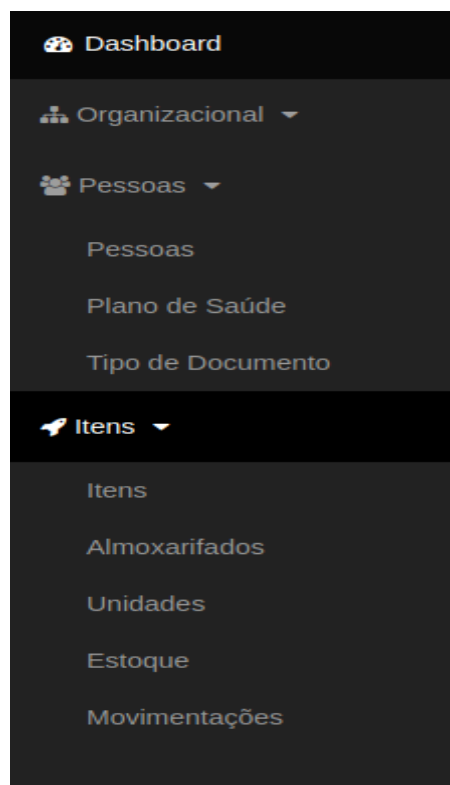


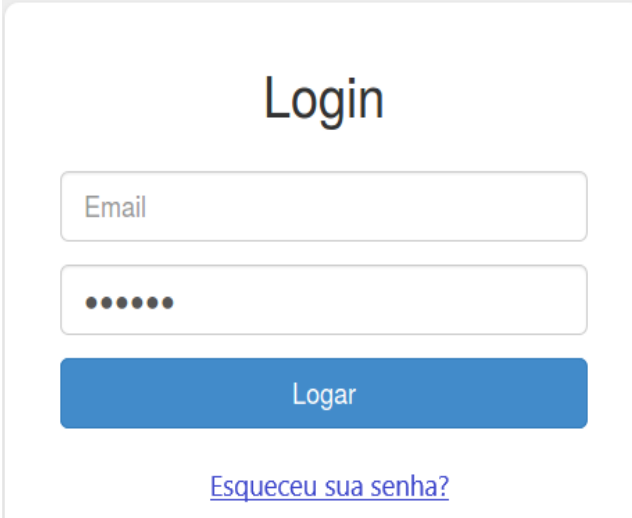
Figura 9: Menu Organizacional.

Fonte: Os autores

5.1.3. Acesso Restrito

O acesso restrito apresenta informações confidenciais, sendo necessária uma política de segurança que restringe tal acesso, conseqüentemente, somente quem possui usuário e senha consegue acessar o sistema e suas informações.

Nesse sentido, o login ao sistema contém o campo de *usuário*, *senha*, o botão *acessar* e o botão *esqueci minha senha*, como ilustra a figura 10. Nos campos de usuário e senha, deverá ser informado, respectivamente, o *usuário* e a *senha*. No botão *acessar*, será validado os dados informados e, se esses estiverem certos o usuário será redirecionado ao SiGFlux, por sua vez, caso estejam incorretos, aparecerá um aviso. No botão *esqueci minha senha*, o usuário terá acesso a uma nova senha através do e-mail que foi cadastrado.



The image shows a login form titled "Login". It contains two input fields: the first is labeled "Email" and the second is a password field with six dots. Below the fields is a blue button labeled "Logar". At the bottom of the form is a blue link labeled "Esqueceu sua senha?".

Figura 10: Login.

Fonte: Os autores.

5.1.4. Controle de Itens

O módulo controle de itens visa sanar um déficit da atlética, uma vez que seus membros não sabem ao certo quais produtos possuem. Portanto, o módulo visa proporcionar controle aos membros dos objetos que a empresa possui, sendo possível, por exemplo, acessar o almoxarifado, ver as movimentações dos itens e cadastrar quais produtos que são necessários comprar.

Sendo assim, a figura 11 ilustra a página onde serão listados todos os itens cadastrados no sistema, no qual é apresentado de forma sucinta, contendo apenas o seu id, descrição, unidade, e se o registro está ou não ativo. Além destas informações, foi desenvolvido três botões: *editar*, *deletar* e *visualizar*; cada um destes botões redireciona ao usuário para a página desejada. Ademais, o botão *novo item* migra o usuário para a página de cadastro.

Itens

Pesquisar Itens



Novo Item

| ID | Nome | Unidade | Ativo | Ações |
|----|-------------------------|--------------|-------|---|
| 1 | BOLA DE FUTEBOL PENALTY | UN (UNIDADE) | Sim |    |
| 2 | COLETES | UN (UNIDADE) | Sim |    |

« 1 »

Figura 11: Controle dos itens cadastrados no sistema.

Fonte: Os autores

5.1.5. Cadastro de itens

O Cadastro de itens é feito de forma bem simples e intuitivo. O protótipo, ilustrado na figura 12, mostra a tela de cadastro, que contém os campos: *nome*, *descrição*, *observação*, *unidade* e o *flag* (responsável por indicar se é um cadastro ativo ou não). Nesta página, o usuário deve indicar as informações conforme necessidade do cadastro.

Após disponibilizar tais informações, o usuário deve utilizar o botão “salvar”, isto é, validar os dados que foram informados. Dessa maneira, se as informações estiverem válidas, o sistema grava-as no banco de dados. No entanto, se não for validado é possível cancelar, ou seja, o usuário será redirecionamento para a tela que inclui informações.

Cadastro de Item

Nome:

Descrição:

Observação:

Unidade: Ativo:

Figura 12: Cadastro de itens.

Fonte: Os autores.

5.1.6. Visualização de itens

A página de visualização refere-se apenas a exibição das informações citadas no campo anterior, item 5.5. Portanto, não é possível, nessa tela, que o usuário edite nenhum campo, como ilustra a Figura 13. Sendo assim, a página apresenta apenas a opção voltar, no qual se tem o redirecionamento do usuário para a página que possui os itens listados. Destaca-se que caso seja necessário a inclusão ou modificação de alguma informação é necessário que o usuário volte para a opção “Inserir item”.

Figura 13: Visualizar item.

Fonte: Os autores

Visualizar Item

Nome:
BOLA DE FUTEBOL PENALTY

Descrição:
BOLA DE FUTEBOL DA MARCA PENALTY

Observação:

Unidade:
UN (UNIDADE)

Ativo:

[Voltar](#)

5.1.7. Edição do cadastro de item

A tela de edição do cadastro de algum item, ilustrado a prototipação na figura 14, é semelhante a tela de cadastro. Dessa forma, ela apresenta os campos com as informações preenchidas, sendo seu diferencial apenas a função de editar algum item selecionado. Portanto, há os campos: *nome*, *descrição*, *observação*, *unidade* e o *flag*, que indica se é um cadastro ativo ou não. Assim, nesta página o usuário deve alterar os campos das informações apenas conforme necessidade.

Da mesma forma que no cadastro dos itens, deve-se salvar, a fim de validar os dados informados e salvar no banco de dados do sistema. Caso haja alguma discrepância ou erro, deve-se cancelar, visando o redirecionamento da página para realizar as modificações necessárias nos itens.

Figura 14: Edição de cadastro de item.

Fonte: Os autores

Editar Item

Nome:

Descrição:

Observação:

Unidade: Ativo:

5.1.8. Exclusão de Itens

Por motivos de segurança e integridade das informações, o sistema não permite a exclusão do cadastro de itens, apenas o desativa. Dessa forma, a figura 15 ilustra o protótipo da tela que oferece a opção do usuário realizar essa ação.

Figura 15: Exclusão de item.

Fonte: Os autores

Excluir Cliente ×

Deseja realmente excluir este Cliente?

Portanto, caso o usuário queira desativar algum item, ou seja, excluí-lo da página e não do sistema, deverá clicar o botão *Excluir*, no qual haverá a opção *Sim* e *Não*, a fim do usuário escolhe-lo, confirmando sua decisão.

5.1.9. Tela de listagem de cursos

Para o cadastro dos membros no sistema, é necessário que este apresente uma listagem dos cursos disponíveis no Centro Universitário de Franca, conforme ilustra o protótipo da figura 16, uma vez que todos os integrantes da associação estudantil devem fazer parte da faculdade em questão.

Dessa forma, vincula-se o ID do aluno ao seu curso e descrição desse, além de ser possível acrescentar alguma informação relevante e verificar sua situação perante à universidade.

Figura 16: Listagem Cursos.

Fonte: Os autores

Cursos

Pesquisar cursos



Novo Curso

| ID | Nome | Descrição | Observação | Ativo | Ações |
|----|------------------------|------------------------|---------------|-------|-------|
| 3 | ADMINISTRACAO | ADMINISTRACAO | ADMINISTRACAO | Sim | |
| 2 | ENGENHARIA DE SOFTWARE | ENGENHARIA DE SOFTWARE | E.S. | Sim | |
| 1 | SISTEMAS DE INFORMACAO | SISTEMAS DE INFORMACAO | S.I. | Sim | |

5.1.10. Membros

A listagem dos membros, ilustrado na figura 17, refere-se à identificação do ID do aluno, ID da modalidade, observação e situação deste. Com isso, é possível analisar quais alunos participam da associação atlética e em quais modalidades estão inseridos, criando, assim, um banco de dados para a atual gestão, assim como, para as futuras gestões.

Figura 17: Listagem membros.

Fonte: Os autores.

Membros

Pesquisar Membros

Novo Membro

| ID | ID Aluno | ID Modalidade | Observação | Ativo | Ações |
|----|--|--------------------|---------------------------|-------|-------|
| 1 | TESTE DE PESSOA (SISTEMAS DE INFORMACAO) | ATLETICA DIRETORIA | presidente eleito em 2017 | Sim | |

« 1 »

Ademais, criou-se um módulo responsável pelo armazenamento dos dados pessoais dos integrantes da atlética, apresentado na figura 18, ou seja, uma relação entre o ID do estudante, seu nome, data de nascimento, número (código do aluno) e e-mail. Isso facilita não apenas a atual gestão, mas também as próximas, já que cria um banco de dados com o contato dos indivíduos que trabalharam na associação.

Figura 18: Dados Membros.

Fonte: Os autores.

Pessoas

Pesquisar Itens



Nova Pessoa

| ID | Nome | Data de Nascimento | Número | Email | Ações |
|----|-------------|------------------------------|--------|-------|-------|
| 1 | TESTE TESTE | 11 de Abril de 1996 às 00:00 | 1 | | |

« 1 »

Por fim, é necessária a determinação das funções de cada integrante dentro da associação atlética, ou seja, qual o cargo e responsabilidades dos membros individualmente, conforme ilustra a figura 19.

Figura 19: Funções dos membros.

Fonte: Os autores

| ID | Nome | Descrição | Observação | Ativo | Ações |
|----|------------|---|---------------------------|-------|---------------------------|
| 2 | ATLETA | MEMBRO QUE REPRESENTA A FACULDADE EM COMPETIÇÕES ESPORTIVAS | | False | Visualizar Editar Excluir |
| 1 | PRESIDENTE | PRESIDENTE DA ATLÉTICA | RESPONSÁVEL PELA ATLÉTICA | False | Visualizar Editar Excluir |
| 4 | TECNICO | TECNICO | | True | Visualizar Editar Excluir |
| 6 | TESOUREIRO | RESPONSÁVEL PELO DINHEIRO | | True | Visualizar Editar Excluir |
| 3 | ZJCS | SLKDCS | LKSMCWS | True | Visualizar Editar Excluir |

5.1.11. Modalidades

O campo de modalidades, figura 20, apresenta ao usuário quais modalidades ele está participando, isto é, quais são seus compromissos com a atlética. Com isso, é possível criar uma série de dados que disponibiliza quais esportes os alunos têm mais interesse, quais estão precisando de pessoas, por exemplo.

Portanto, as informações tornam-se mais acessíveis e seguras, consequentemente, torna as análises feitas pela associação mais precisas e corretas.

Figura 20: Modalidades.

Fonte: Os autores

| ID | Nome | Descrição | Observação | Ativo | Ações |
|----|--------------------|---|--|-------|---------------------------|
| 1 | ATLETICA DIRETORIA | MODALIDADE PARA OS MEMBROS DA DIRETORIA DA ATLÉTICA XX-DE-MARÇO | MODALIDADES PARA CARGOS COMO PRESIDENTE, DIRETORES, TESOUREIROS, ENTRE OUTROS. | Sim | Visualizar Editar Excluir |
| 4 | BASQUETE MASCULINO | BASQUETE MASCULINO | BASQUETE MASCULINO | Sim | Visualizar Editar Excluir |
| 3 | FUTSAL FEMININO | FUTSAL FEMININO | FUTSAL FEMININO | Sim | Visualizar Editar Excluir |
| 2 | FUTSAL MASCULINO | FUTSAL MASCULINO | FUTSAL | Sim | Visualizar Editar Excluir |

6. Considerações Finais

Este projeto teve como seu objetivo principal apresentar um software que elimine o uso de interfaces manuais e pessoais das informações, criando padrões

dentro da associação atlética estudada. Portanto, visou-se suprir as necessidades de uma organização estudantil, principalmente, em relação aos seus bens e membros, por meios de cadastros de todos os seus itens no sistema, mantendo atualizado e salvo todas as informações necessárias para o gestor.

De acordo com o que foi levantado em entrevistas com membros da atlética em relação aos seus processos e rotinas, juntamente com o que foi ministrado no curso de sistemas de informação e também com o conhecimento dos autores em experiências vivenciadas no mercado de trabalho, o desenvolvimento e a documentação desse projeto foi concluída com sucesso.

Neste projeto foram aplicados conceitos universais de desenvolvimento de software, cumprindo com as etapas de levantamento de requisitos, modelagens de banco de dados, prototipação, casos de uso, diversos tipos de diagramas que são responsáveis por representar com exatidão o passo a passo do sistema e foram usados para a representação de como o processo era feito anteriormente. Portanto, tais conceitos foram usados para definir uma rota, a fim de que o desenvolvimento fosse realizado da melhor maneira possível.

As disciplinas de engenharia de software, linguagens de programação, banco de dados e aspectos avançados em banco de dados, empreendedorismo, entre outras que foram ministradas durante o curso, foram essenciais para o desenvolvimento do projeto e também da documentação, juntamente com auxílio dos professores durante o processo de realização do trabalho de conclusão de curso.

Alinhado a tudo o que já foi citado anteriormente, o estudo feito para a elaboração do referencial teórico, complementou conceitos que até então não eram desconhecidos para os autores e como eles devem ser implementados em um projeto de software.

Depois destas etapas seguidas, os autores levam para o mercado de trabalho técnicas de desenvolvimento e estruturação de projetos que até então não são aplicadas como deveriam, ampliando sua visibilidade no mercado e também seu conhecimento numa área pouco explorada atualmente. Ademais, favorece a manutenção e a implementação de novos projetos, melhorando processos nas empresas, no qual se cria um mercado mais competitivo e mais lucrativo, tanto para funcionários, quanto para as empresas em que estes profissionais estão envolvidos.

Este projeto dá aos autores a oportunidade de explorar outras áreas e também de ajudar as empresas em que estão inseridos a aumentar seus lucros removendo qualquer tipo de processo que não seja extremamente útil, seguindo padrões internacionais, facilitando também a vinda de outros profissionais de outros países para serem inseridos em projetos dessa empresa, abrindo assim um grande leque de possibilidades, vislumbrando novos desafios e oportunidades.

Também traz a experiência de aprender uma nova linguagem que está em alta no mercado de trabalho, agregando a experiência que esta já foi aplicada em um projeto e conhecendo as dificuldades e imprevistos em que o trabalho nesta linguagem pode trazer, aumentando a visibilidade no mercado e o conhecimento dos autores, este último que é um meio indestrutível para qualquer ser humano.

Para a implementação do software e continuidade do projeto, os autores farão reuniões com todos os membros da atlética para exibição de todas as funcionalidades que foram desenvolvidas no sistema, junto com a melhor forma de usar, dados a serem colocados e formatações, também será disponibilizado um suporte durante um período de 3 meses, para qualquer dúvida restante com relação ao uso do software.

Referências

- BONAZZI, F. L.Z; MEIRELLES, D.S. Mobile: A evolução de um modelo de negócios à frente de seu tempo. SEMEAD – Seminários em Administração FEA/USP, São Paulo, SP, BRASIL, 2013.
- BLANK, STEVEN, Évora; Edição: 1ª, Do Sonho à Realização em Quatro Passos. (1 de setembro de 2012)
- DANTAS, D. C. G.; ALVES, R. F. Sistemas de informação ERP: Uma visão gerencial de ciclo de vida de implantação. Anais do Encontro Nacional de Engenharia de Produção, n. 12, p. 1-5. 2002.
- DAVENPORT, T.H. Comoditização de Processos – a nova onda, Harvard Business Review, edição em português, p. 76-84, Jun. 2005.
- DORNELAS, J.C.A. Empreendedorismo: Transformando ideias em negócios. Rio de Janeiro: Elsevier Brasil, 2008.
- ELISANGELA PEREIRA s.d.. Business Model Canvas: O que é? E como fazer um para minha empresa? Disponível em: <<http://atitudeenegocios.com/business-model-canvas/>> Acesso em 20. jun. 2017.
- FIESP/FEA-USP; Idigital – Pesquisa Perfil da Empresa Digital Resultados Completos da terceira edição (2003/2004). Disponível em: <<http://www.idigital.fea.usp.br/iDigital>>. Acesso em: 03 jul. 2017.
- GARTNER GROUP. The Gartner glossary of information technology acronyms and terms, 2004. Disponível em: <www.tc.suny.edu/pdf_docs/gartner glossary.pdf>. Acesso em: 08 jun. 2017.
- GLASS, R.L. Agile versus Traditional: Make Love not War. Cutter IT Journal, v. 14, n. 12, December 2001.
- GLOBO UNIVERISIDADE. Saiba como surgiram os primeiros jogos universitários no Brasil, 2012. Disponível em: <<http://redeglobo.globo.com/globouniversidade/noticia/2012/12/saiba-como-surgiram-os-primeiros-jogos-universitarios-no-brasil.html>>. Acesso em: 03 jul. 2017.
- GUEDES, G. UML 2: Guia Prático. São Paulo: Novatec, 2014.
- Guido van Rossum (Maio de 1996). Foreword for "Programming Python" (1st ed.) . Python Software Foundation. Consultado em 12 de junho de 2008
- HILL, T. Manufacturing Strategy: the strategic management of themanufacturing function. SecondEdition, London, Macmillan, 1993.
- IDG-Now. Pequenas e médias devem investir mais em TI em 2003. ComputerWorld – Negócios. Disponível em <www.computerworld.com.br/AdPortalV3/>. Acesso em: 03 jul 2017.
- KIRZNER, LM. Competition and entrepreneurship. Chiago University Press, 1973.
- KRUCHTEN, P. Agility with RUP. Cutter IT Journal, v. 14, n. 12, December 2001.
- LAUDON, K.; LAUDON, J. Sistemas de informações gerenciais. São Paulo: Pearson, 2010.
- O'BRIEN, J. A.; MARAKAS, G. M. Administração de sistemas de informação: Uma introdução.13ª Edição. São Paulo: McGraw-Hill. 2007.
- OSTERWALDER; Alexander.Business Model Generation - Inovação em Modelos de Negócios: ummanual para visionários, inovadores e revolucionários f AlexanderOsterwalder, Yves Pigncur. Rio de Janeiro, RJ : Alta Books, 2011.
- PADILHA, T. C. C.; MARINS, F. A. S. Sistemas ERP: Características, custos e tendências. Revista Produção, 15(1), p. 102-113. 2005.

- PAULK, M. Extreme Programming from a CMM perspective. IEEE Software, November/December 2001.
- PORTER, M. Estratégia Competitiva. 5 ed. Rio de Janeiro: Campus, 1986.
- RATTNER, H. Pequena Empresa: o comportamento empresarial na acumulação e na luta pela sobrevivência. Conselho Nacional de Desenvolvimento Científico e Tecnológico de Brasília. São Paulo: Brasiliense, 1985.
- RIES, E. Lean Startup. Prime books, 2012.
- TURBAN, E.; VOLONINO, L. Tecnologia da informação para gestão: em busca do melhor desempenho estratégico e operacional. Porto Alegre: Bookman, 2013.
- SCHUMPETER, J. The Theory of Economic Development. Harvard University Press, 1949
- SHAHAWAI, S. S.; IDRUS, R. Pre-considered factors affecting ERP system adoption in Malaysian SMEs. Proceedings of the 12th UKSim International Conference on Computer Modelling and Simulation, p. 323-328. 2010.
- SLACK, N. et al. Administração da Produção. 2. ed. São Paulo: Ed. Atlas, 2003.
- SOMMERVILLE, I. Software Engineering. 6. ed. Addison Wesley, 2003.
- SOMMERVILLE, I. Engenharia de software. São Paulo: Pearson, 2011.
- SOUZA, C. A.; ZWICKER, R. Ciclo de vida de sistemas ERP. Caderno de Pesquisas em Administração, 1(11), p. 46-57. 2000.
- SUTHERLAND, J. Agile can scale: Inventing and Reinventing SCRUM in Five Companies. Cutter IT Journal, v. 14, n. 12, December 2001.
- SU, Y.; YANG, C. A structural equation model for analyzing the impact of ERP on SCM. Expert Systems with Applications, 37, p. 456-469. 2010.
- ZABJEK, D.; KOVACIC, A.; STEMBERGER, M. I. The influence of business process management and some other CSFs on successful ERP implementation. Business Process Management Journal, 15(4), p. 588. 2009.