

O uso das técnicas de elicitação de requisitos e processos de software na prototipação de aplicativos mobile

Jonathan Schütz Kronbauer¹, Rafael Rodighero Pereira¹, Evanise Araujo Caldas Ruiz¹

¹FACET – Universidade Federal da Grande Dourados (UFGD)

CEP 79.804-970, Cx. Postal 322, Cx. Postal 322, Dourados MS - Brasil

jonskron@gmail.com, rafaelrodighero10@hotmail.com,
evanisealdas@ufgd.edu.br

Abstract. *This work aims to use the requirements elicitation techniques questionnaire and card sorting, in addition the software process model prototyping for the creation of a prototype of a mobile application aimed at student aid during and post-pandemic. The techniques were used remotely, focusing on requirements collecting and organizing the application, using SynCaps v3 software for card sorting results analysis, Astah UML for diagrams creation, and Marvel app for creating the final prototype.*

Resumo. Este trabalho tem como objetivo utilizar das técnicas de elicitação de requisitos questionário e *card sorting*, além do modelo de processo de software prototipação para a criação de um protótipo de um aplicativo *mobile* que visa o auxílio estudantil durante e pós-pandemia. As técnicas foram utilizadas de forma remota, com foco na coleta de requisitos e na organização do aplicativo, sendo utilizado o *software* SynCaps v3 para análise dos resultados do *card sorting*, do Astah UML para criação de diagramas, e da plataforma *Marvel app* para criação do protótipo final.

Palavras-chave: Processo de *software*; Prototipação; Aplicativos *Mobile*; Engenharia de requisitos; Técnicas de Elicitação de Requisitos.

1. Introdução

A qualidade de um sistema é cada vez mais importante nos dias atuais agregando valor e produtividade nos negócios da empresa, e se não realizado adequadamente poderá gerar manutenção e custos excessivos (ENGHOLM JÚNIOR, 2010). A Engenharia de Requisitos (ER) trabalha adquirindo dados e informações para o desenvolvimento, enquanto cada Modelo de Processo de *Software* (MPS) representa uma abordagem diferente de trabalho para a criação de um *software* (PRESSMAN, 2016).

Os MPS são representações ou abstrações dos objetos e atividades envolvidas no processo de *software*. Dentre os modelos existentes, o modelo prototipação é um dos que mais se destaca no desenvolvimento de produtos de *software*. Este modelo é evolucionário, ou seja, iterativo, que possibilita desenvolver versões cada vez mais completas do *software* (SOMMERVILLE, 2013).

Segundo RIBEIRO E SOUZA (2012), a ER é uma atividade fundamental no processo de desenvolvimento de *software*, assegurando que um conjunto completo das necessidades e requisitos dos usuários sejam capturados e transformados em um conjunto válido de requisitos em todo o ciclo de vida. As técnicas de levantamento de requisitos utilizadas neste trabalho foram o “*Card Sorting*” (arranjo de cartas), que é um método para estudar como as pessoas organizam e classificam seu conhecimento através de cartas, onde os participantes irão categorizá-las como acharem adequado (WOOD E WOOD, 2008), e o “Questionário”, que é ideal quando se deseja atingir um grande número de pessoas, tendo assim uma elicitación de requisitos eficaz (FUZII, SOUZA E TRONCO, 2009).

A implementação de sistemas sem a realização de um processo de *software* apresentarão dificuldades com a definição dos prazos e custos, e sem um bom planejamento, os desenvolvedores não terão motivação e produtividade, resultando em um *software* com baixa qualidade e diversos problemas de manutenção (PRIKLADNICKI, 2004; CABRAL, SILVA E SOUZA, 2014). A elicitación de requisitos se mostrou outro ponto importante, visto que problemas relacionados à ER acarretam projetos a não cobrirem suas funcionalidades ou orçamentos, sendo classificados como problemáticos (STANDISH GROUP, 1995), e segundo ÁVILA E SPÍNOLA (2008), o custo das atividades relacionadas à análise de requisitos é baixo (5%), sendo nesta fase que grande parte dos defeitos são inseridos (55%).

Este trabalho tem como objetivo desenvolver um protótipo de aplicativo *mobile*, e analisar a eficiência do processo de *software* prototipação e das técnicas de elicitación de requisitos *card sorting* e questionário. Teve como objetivo primário o desenvolvimento de um protótipo de aplicativo *mobile* para a terceira idade, entretanto conforme seção 2 explica, foi necessária uma mudança de público alvo, sendo agora desenvolvido com o novo propósito de auxiliar na organização da vida acadêmica, buscando atingir estudantes universitários como novo público alvo.

O desenvolvimento dos protótipos teve como objetivo aplicar mudanças indicadas em cada iteração do modelo de processo de *software* utilizado, e demonstrar aos usuários uma visão mais ampla do futuro aplicativo. Com principal foco na Universidade Federal da Grande Dourados (UFGD), o protótipo foi pensado para primeiramente esta universidade e para posteriormente ser adaptado para outras de acordo com suas necessidades.

O trabalho está organizado em 8 seções: (i) Seção um: Introdução, (ii) Seção dois: Mudança do público-alvo; (iii) Seção três: *Card Sorting* (iv) Seção quatro: Materiais e métodos ; (v) Seção cinco: Trabalhos correlatos ; (vi) Seção seis: Resultados e discussões; (vii) Seção sete: Conclusão e trabalhos futuros, e (viii) Seção oito: Referências.

2. Mudança do público alvo

O decorrer do ano de 2020 foi conturbado, em específico, por conta da pandemia do novo coronavírus (Sars-Cov-2), fazendo com que o mundo tivesse de adotar medidas diferentes para evitar o contágio¹. Todas as áreas do conhecimento foram afetadas por

¹Conselho nacional de Saúde, 2020 - <http://conselho.saude.gov.br/recomendacoes-cns/1103-recomendac-a-o-no-020-de-07-de-abril-de-2020>

esta pandemia, e a educação de uma maneira geral teve um impacto ainda maior, visto que as escolas e universidades em sua maioria tiveram suas aulas presenciais comprometidas e alteradas para um sistema remoto (PEREIRA, NARDUCHI e MIRANDA, 2020), o que levou à alteração da metodologia e do objetivo apresentado.

O objetivo inicial deste trabalho, era realizar a modelagem de um software com o apoio dos MPS Prototipação e da ER com as técnicas “*Card Sorting*” e “Observação” para um projeto voltado à terceira idade, no qual seria executado mais um passo para a futura criação de um aplicativo *mobile* que os auxiliasse no uso de suas medicações. Assim, tanto o público alvo ser de risco² quanto o distanciamento social se mostraram os principais fatores dificultantes da pesquisa, além de que poucos saberiam utilizar ferramentas online para a realização das mesmas. Desta forma, mostrou-se necessária a mudança da técnica de observação para o questionário, já que é um método mais amigável ao meio remoto, e que aborda um número maior de pessoas com facilidade (FUZII, SOUZA e TRONCO, 2009).

Com isso, foi necessária a alteração do objetivo inicial, tendo em vista que o novo público alvo deveria estar ao alcance para a realização dos questionários e do *card sorting*. Então decidiu-se realizar o desenvolvimento de um protótipo voltado para auxiliar os alunos a enfrentarem melhor os desafios durante a pandemia e pós pandemia.

3. Card sorting

A técnica de elicitação de requisitos *card sorting* se trata de uma técnica com finalidade de avaliar ou gerar uma estrutura de informações, ajudando na rotulação e organização de projetos de sistemas (VILELA, CARVALHO, FAGUNDES, 2009). Essa técnica abrange três etapas: planejamento, arranjo dos cartões e elicitação.

A primeira etapa consiste em que o pesquisador identifique os itens que deverão ser organizados pelos participantes. É nesta etapa também que é definido se o *card sorting* será de forma aberta, fechada ou híbrida. Na forma aberta, os participantes organizam os itens da forma que acharem adequada, e dão nome aos grupos de itens. Na forma fechada, os pesquisadores fornecem aos participantes as categorias, e só é necessário o agrupamento nestas categorias pré-definidas. Na forma híbrida, os pesquisadores também fornecem categorias pré-definidas, porém os usuários têm a possibilidade de criar novas categorias se julgarem necessário (PADOVANI E RIBEIRO, 2013). De acordo com WOOD E WOOD (2008), é recomendado começar pesquisas com o método aberto, mesmo que os pesquisadores julguem suas categorias adequadas, pois muitas vezes não estão em sintonia com os usuários.

Na segunda etapa, a de arranjo de cartões, os pesquisadores entregam os cartões aos participantes pedindo para que os mesmos formem grupos com estes cartões de forma que faça sentido a eles, para depois o pesquisador analisar os resultados, e busque formas de unificar os cartões em uma estrutura. O pesquisador deve explicar o funcionamento, mas não opinar como o usuário deve agrupar. ALMEIDA et. al 2014 propuseram em seu trabalho a aplicação desta etapa de forma online.

A terceira etapa, a de elicitação, é uma etapa opcional onde os participantes explicam o porquê da sua escolha de agrupação, e qual a lógica por trás das escolhas.

² <https://www.paho.org/pt/covid19>

Esta informação deve ser armazenada, para auxílio futuro para os pesquisadores (PADOVANI E RIBEIRO, 2013).

4. Materiais e métodos

O desenvolvimento deste artigo teve como premissa a revisão sistemática desenvolvida na primeira etapa do trabalho de conclusão de curso com o título “Uma revisão sistemática sobre técnicas de elicitação de requisitos e processos de *software* no desenvolvimento de sistemas”³, proporcionando resultados para a definição do modelo de processo de *software* (prototipação), bem como a escolha das técnicas de levantamento de requisitos (questionário e *card sorting*).

Os processos de *software* são um conjunto de atividades exigidas para desenvolver um sistema de *software*. A ideia geral é fazer com que através desse processo de atividades se alcance um produto de *software* com qualidade (SOMMERVILLE, 2013; PRESSMAN, 2016).

O MPS aplicado foi a prototipação incremental, utilizado para mostrar ideias, conceitos, experimentar opções para o desenvolvimento, encontrar problemas e desenvolver soluções (SOMMERVILLE, 2013). Conforme PRESSMAN (2016) prototipação é um processo usado para o desenvolvedor se comunicar com o usuário com o objetivo de entender e adquirir os requisitos do sistema. O trabalho conduziu-se através de 4 iterações completas das 5 fases da prototipação (Figura 1):



Figura 1: Modelo de prototipagem. Fonte: (PRESSMAN, 2016).

Na 1ª fase, a de comunicação, foram levantados os requisitos através da técnica de questionário, que é uma técnica onde perguntas são feitas aos usuários de forma simples e objetiva, sem ambiguidades (SILVA, 2012), realizado com um misto de questões objetivas e discursivas com o apoio da plataforma “Formulários Google”⁴, que forneceu gráficos com as respostas gerais, além de possibilitar a análise manual das respostas discursivas. Também fez-se uso da técnica de *card sorting* para identificar e adequar o modelo mental do usuário à navegação de um sistema (SÁ, 2013), realizada

³ <https://drive.google.com/file/d/1mHX8PEIICiYqIaci6IoCj6B0JLc1Rdnd/view?usp=sharing>

⁴ https://docs.google.com/forms/d/e/1FAIpQLSfkboPiI9o2-hmy4g8K0tGPS4CQI_Bf6XdNA9XTC11_8-MkHw/viewform

através da plataforma “KARDSORT”⁵. O público-alvo deste levantamento foram alunos do ensino superior, realizado de forma online.

O questionário realizado focou nas questões que levantam informações como maior público alvo, sistema operacional, meios de comunicação e de estudos, relevância de ferramentas presentes na ideia do aplicativo, dificuldades presentes na comunicação dentro da universidade e *feedback* dos possíveis usuários. Já o *card sorting* oferece opções de pastas e os participantes tinham de colocar as cartas onde julgassem mais coerente, e caso não encontrassem uma boa opção, tinham a possibilidade de criar uma adequada. Assim, com o uso do “Syncaps V3”⁶, um *software* para análise de *card sorting*, foi possível montar dendrogramas para análise de respostas próximas e grupos formados, e uma posterior criação de uma tabela com a relação dos grupos com os itens, conforme dito pelos participantes, para análise manual pelos autores.

Na 2ª e 3ª fase, as de planejamento e modelagem, foram definidos os requisitos funcionais (RF), que são os que definem como o sistema irá reagir e se comportar em determinadas situações, e os requisitos não-funcionais (RNF), que são as restrições à serviços ou funções oferecidas pelo sistema (SOMMERVILLE, 2013). Feita a análise dos requisitos, foram elaborados os diagramas, com suporte do *software* “ASTAH UML”⁷: (i) o diagrama de caso de uso, que identifica os atores envolvidos e representa todas interações descritas nos requisitos; e (ii) o diagrama de classes, que mostra as classes do sistema e suas associações através de atributos e operações (CYSNEIROS, 2001).

Na 4ª fase, a de construção, foi produzido um protótipo como base inicial mais três protótipos incrementais derivados da fase de comunicação, totalizando 4 protótipos: (i) os três primeiros protótipos de baixo nível, que foram realizados no editor de imagens “Adobe Photoshop”⁸, e (ii) o protótipo final, de alto nível, que foi realizado na plataforma “MarvelApp”, uma ferramenta para criação de protótipos interativos em plataformas digitais (PINEDA, 2018).

Na 5ª e última fase, a de *feedback*, foi apresentado os protótipos para um grupo seletivo de 5 a 10 pessoas avaliarem, estas eram em sua maioria pessoas pertencentes à UFGD, visto que o protótipo estava voltado para esta universidade, e que se mostraram proativas durante as fases de elicitação. Então, foram feitas anotações das discussões e analisadas possibilidades para o próximo protótipo. Essa fase foi realizada apenas nas três últimas iterações, ou seja, nos últimos três protótipos, e o grupo que realizou o *feedback* foi formado em sua maioria pelas mesmas pessoas.

Desta forma, com a aplicação do MPS, das técnicas, metodologias e ferramentas, o desenvolvimento do protótipo final é demonstrado com as evoluções alcançadas através destas. Em considerações finais, é feita uma abordagem sobre como este protótipo ainda pode evoluir até tornar-se um aplicativo *mobile*, adicionando funcionalidades e adaptações para projetos futuros.

5. Trabalhos correlatos

Este trabalho teve como base uma revisão sistemática desenvolvida pelos autores, com o título “Uma revisão sistemática sobre técnicas de elicitação de requisitos e processos

⁵<https://study.kardsort.com/URSA>

⁶ <https://syntagm.co.uk/design/cardsortdl.shtml>

⁷ <https://astah.net/>

⁸ <https://www.adobe.com/br/products/photoshop.html>

de *software* no desenvolvimento de sistemas”⁹, que teve como objetivo selecionar e estudar tanto modelos de processo de *software* quanto técnicas de eliciação de requisitos. Os trabalhos correlatos aqui apresentados foram selecionados a partir da revisão sistemática, exceto quando sobre o *card sorting*, os quais foram pesquisados no desenvolvimento deste trabalho.

No estudo "Estudo comparativo sobre as técnicas de eliciação de requisitos do software", BELGAMO E MARTINS (2000) apresentam um estudo comparando as técnicas de eliciação de requisitos, como por exemplo a entrevista e observação, classificando as mesmas através de parâmetros, como confiabilidade e tempo, definidos pelos pesquisadores, e classificando suas vantagens e desvantagens através de uma tabela comparativa. O resultado final foi uma tabela comparativa dos parâmetros para cada técnica, que os autores pretendiam utilizar em uma monografia, com objetivo de auxiliar no ensino da área.

Já o artigo de ALVES E VANALLE (2001) “Ciclo de vida de desenvolvimento de sistemas - visão conceitual dos modelos clássico, espiral e prototipação” teve como objetivo realizar uma análise do ciclo de vida dos modelos clássico, espiral e prototipação, avaliando seus problemas de desenvolvimento, bem como o rigor dos requisitos e da metodologia aplicada. A partir de um estudo de pesquisas de vários autores, constatou-se que os ciclos de vida dos modelos em questão convergem no processo de refinamento dos requisitos, porém cada modelo deve ser aplicado considerando suas características vitais no desenvolvimento do sistema. Enquanto a prototipação pode pressionar os desenvolvedores, impactando na qualidade do sistema, o modelo espiral necessita de uma participação mais consciente, faltando assim flexibilidade para sistemas terceirizados.

Em "Prototipação de *software* e Design Participativo: uma Experiência do Atlântico" por ROSEMBERG et al. 2008 é realizada uma prototipação de baixa fidelidade para eliciação de requisitos através de uma abordagem de design participativo. Primeiramente solucionaram desafios com os stakeholders e então definiram o perfil dos usuários do sistema, e com ajuda do *software* Microsoft Visio, realizaram simples protótipos que mostraram somente elementos do fluxo da discussão. Como resultado, houve um aumento no envolvimento e na criatividade da equipe de desenvolvimento, possibilitando soluções mais efetivas que proporcionaram mais experiência e qualidade.

Segundo LESSA E JUNIOR (2009) no artigo “Modelos de Processos de Engenharia de *Software*”, os processos de *software* vieram para corrigir problemas e otimizar o desenvolvimento de sistemas. Seus objetivos foram apresentar os modelos clássico, prototipação e espiral, e um guia chamado SWEBOK, que desmembra a engenharia de *software* em dez áreas de conhecimento. Como resultado, classificaram o SWEBOK como uma boa opção de material completo, constituído por vários pesquisadores. Por fim, o artigo salienta o quão crítico é o fator de sucesso da engenharia de *software* no desenvolvimento de sistemas.

VILELA, CARVALHO, FAGUNDES (2009), em “Utilização do *Card Sorting* na Implementação de um Sistema de *Balanced Scorecard* para Instituições de Ensino Superior”, demonstraram o uso do *Card Sorting* de modo a avaliar a usabilidade de uma proposta de um sistema de "*Balanced Scorecard*", para o ensino superior. Recomendando a técnica para desenvolvimento ou redesenho de websites, os autores afirmaram que o *card sorting* não é uma avaliação técnica, e que não mostra os erros de

⁹ <https://drive.google.com/file/d/1NIwT4ru-Zs-35toBYsHk1kTg4P8K7C2B/view?usp=sharing>

um website. Através de sua aplicação em 4 usuários de modo aberto, conseguiram-se 9 categorias (juntando os nomes parecidos), e concluiu-se que foi possível avaliar a usabilidade do sistema, além de que o agrupamento realizado ter sido idêntico com todos usuários, com apenas um cartão estando em categorias distintas.

Escrito por SILVA E SAVOINE (2010), o artigo "Análise comparativa de ferramentas computacionais para prototipação de interfaces" apresenta uma comparação de duas ferramentas utilizadas no desenvolvimento de protótipos de interface. As ferramentas analisadas foram o GUI Design Studio e o OverSite, e tiveram como parâmetros comparativos suas funcionalidades para a construção de uma interface simples. A tabela final concluiu que o GUI Design Studio possui maior liberdade de criação enquanto o OverSite demonstra uma falta de flexibilidade em criar variados tipos de protótipos.

PADOVANI E RIBEIRO (2013) falam em "*Card sorting*: adaptação da técnica para aplicação ao design de sistemas de informação não digitais" sobre a adaptação do *card sorting* para vários sistemas. No artigo, é explicado o uso do *card sorting* fazendo uma adaptação de conceitos, procedimentos e métricas do mesmo. O artigo detalha o uso da técnica em três sistemas (embalagens alimentícias, manual de eletrodomésticos e mapas turísticos), e como foi feita sua aplicação e análise de dados. Após as aplicações, concluíram que a mudança de objetivo nesta adaptação permitiu aos usuários participarem de um estágio de design, proporcionando flexibilidade da elicitação de requisitos, ouvindo as opiniões e dificuldades dos usuários, diferente da forma tradicional. Os resultados mostraram que os participantes compreenderam a aplicação da versão adaptada da técnica, e os autores crêem que esta possa ajudar a desenvolver soluções mais adequadas para produtos em nível de escopo e arquitetura de informação.

No artigo "Arquitetura de informação na web: um estudo de caso da utilização do método *card sorting* para redesign do portal da Universidade Federal de Pelotas" (ALMEIDA, BANDEIRA E WEYMAR, 2014), propuseram uma arquitetura nova para o website de sua universidade, enfatizando a usabilidade. Foram realizados testes de usabilidade, analisando o tempo gasto nas tarefas propostas, e o *feedback* dos usuários. Com os resultados, um redesign dos menus do website foi realizado, onde a técnica *card sorting* foi utilizada, definindo o melhor posicionamento de cada item nos menus. O objetivo de propor a nova arquitetura obteve sucesso, conseguindo uma organização superior de acordo com o método utilizado.

De acordo com SEGUNDO(2015) em "Pesquisa exploratória em elicitação de requisitos: teoria e prática", erros em requisitos e a não-adequação dos mesmos são os fatores que mais tem valor de correção entre problemas de um projeto de software. Foi realizada uma avaliação das técnicas de elicitação na literatura junto com uma análise de como é feita a elicitação em si, através de entrevistas, onde também procurou-se saber quais problemas ocorriam nas aplicações das técnicas. Como resultado, 14 técnicas foram encontradas na literatura, e a mais utilizada no mercado foi a entrevista. Tanto na pesquisa quanto na revisão, os requisitos são tratados de acordo com sua metodologia, e também foi concluído que as técnicas mais recentes buscam resolver problemas mais precisos dos requisitos.

No trabalho "Roda da reprodução IOS: aplicativo móvel para gestão de rebanho leiteiro", PINTON, NETO e VISOLI (2017) utilizam a plataforma Marvel app para construção de um protótipo da versão IOS do aplicativo roda da reprodução, que serve para auxiliar na gestão do rebanho leiteiro. Os resultados foram obtidos através de feedback dos testadores. Com os resultados, uma versão teste foi obtida.

Os trabalhos sobre a elicitação de requisitos ajudaram no desenvolvimento do trabalho ao realizar uma análise das técnicas de elicitação, mostrando para qual situação cada uma é mais adequada, testes de usabilidade e comparações entre as mesmas, além de fornecerem exemplos de aplicações, principalmente com o *card sorting*. Já na parte dos processos de *software*, os trabalhos correlatos ajudaram na escolha do método utilizado neste trabalho (prototipação), e a como realizar sua aplicação, visto que os artigos mostraram análises e comparações de ferramentas dos métodos de processo de *software*.

6. Resultados e discussão

Na primeira iteração somente foi realizado o primeiro protótipo de baixo nível que não considerou nenhum resultado das técnicas abordadas, este protótipo serviu como uma base do que era desejado antes de utilizar as técnicas de elicitação de requisitos para adquirir os dados (Figura 2).

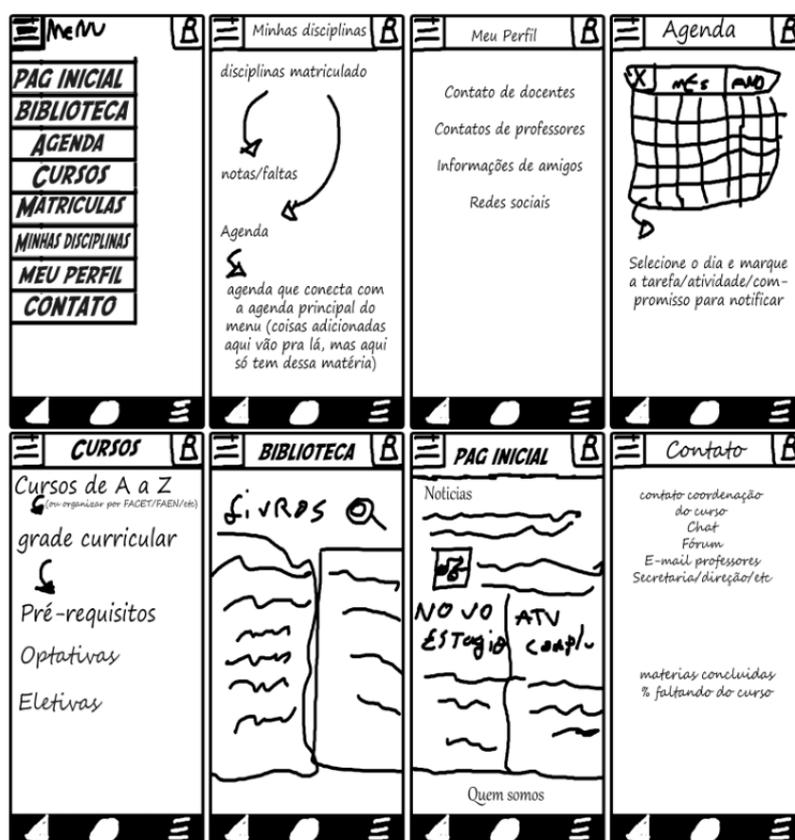


Figura 2: primeiro protótipo. Fonte: os autores (Adobe Photoshop).

Após definir a ideia do aplicativo, iniciou-se a 2ª iteração onde foram realizados o questionário e o *card sorting*. Nesta iteração do protótipo foram obtidas e avaliadas 10 respostas, adquiridas através do envio das técnicas de elicitação para acadêmicos com mais proximidade dos autores. O diagrama de casos de uso foi realizado para obtenção dos primeiros requisitos do sistema (figura 3).

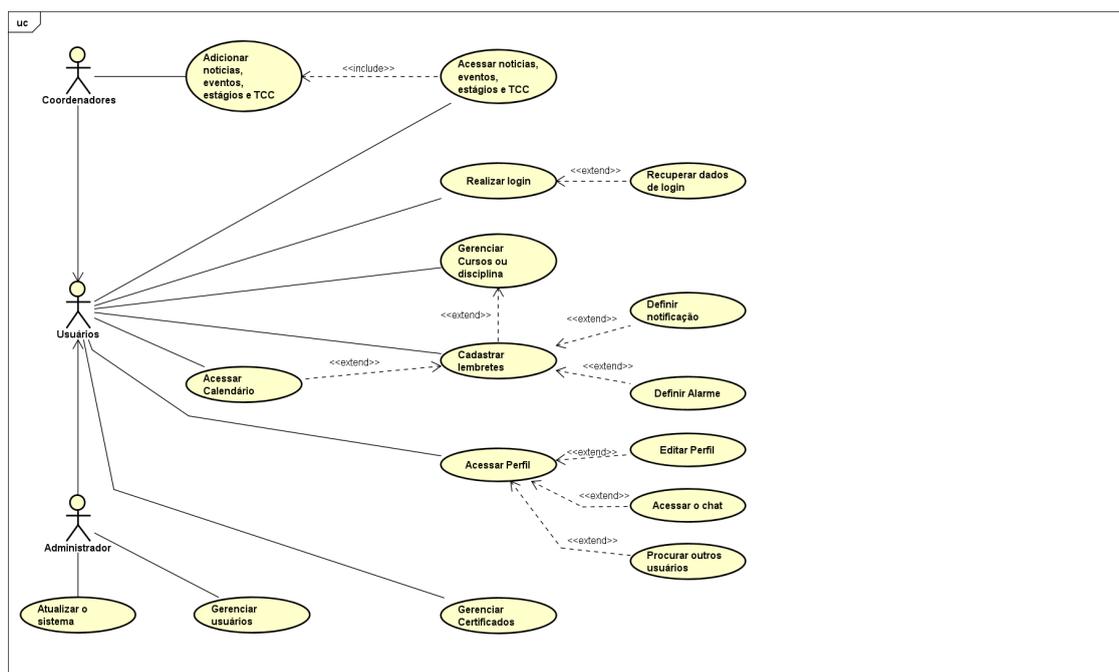


Figura 3: Diagrama de casos de uso. Fonte: os autores (Astar UML).

Através do questionário, notou-se que em sua maioria utilizavam a plataforma Android. Também foi avaliada a importância dada à ideia do aplicativo sugerido, que mostrou 9 respostas favoráveis e 1 neutra. Destacou-se principalmente as funções de notas e de divulgação de estágios (10 respostas favoráveis), seguido por agenda, atividades e comunicação entre usuários, (9 respostas favoráveis). O nível de dificuldade de acesso às informações da universidade foi classificado como médio (6 respostas), e na parte da comunicação, o maior problema foi o atraso no tempo de respostas dos profissionais da instituição (7 respostas).

No *card sorting* apareceram os primeiros grupos por proximidade, totalizando 8. Algumas funções nesta fase foram consideradas de baixa relevância, visto que ou apareceram em poucos resultados, ou divergiram muito nas categorias selecionadas, dentre elas, *podcasts*, pré-requisitos, cardápio e download/upload de arquivos. Com a análise das 10 primeiras respostas de ambas técnicas, o primeiro protótipo (Figura 4) foi desenvolvido e apresentado a esse mesmo grupo de pessoas participantes da pesquisa para obtenção de um *feedback* buscando a evolução do mesmo.



Figura 4: segundo protótipo. Fonte: os autores (Adobe Photoshop).

Na terceira iteração, 50 novas respostas foram levantadas pelo questionário e 15 do *card sorting*, totalizando 60 e 25 respostas respectivamente. Neste momento, a pesquisa ficou próxima do resultado final, e foi nela que as maiores mudanças aconteceram. Dentre as principais mudanças, estão:

- Página de login criada, que não havia sido feita no protótipo anterior;
- Cursos e Disciplinas: foram colocados no mesmo grupo devido a alta proximidade vista no *card sorting*, e sofreram redução nas funções propostas que serão consideradas como trabalho futuro;
- Contato: foi realocado para a aba “Meu perfil”, permitindo usuários procurarem outros usuários com suas respectivas informações para contato disponibilizadas, mudança realizada devido as classificações do *card sorting*;
- Certificados: foi criada uma nova aba chamada certificados, uma vez que houveram solicitações no questionário para a abertura de um espaço de controle de certificados;
- Deslogar: função colocada no menu para rápido acesso;
- Página inicial: teve um aprimoramento visual, e foram adicionadas as funções de “pesquisa” e “quem somos”, visto que estas tiveram maior proximidade na pasta do *card sorting*;

- Meu perfil: houve mudança da proposta inicial de ter uma lista separada dos usuários, mostrando agora informações de contato, acesso ao chat, e busca por outros usuários. Essa mudança foi uma junção dos grupos de contatos e perfil devido à análise do *card sorting*;
- Cursos: alterado para ser mais amigável visualmente já que possuía muita informação desnecessária, abrindo assim mais liberdade para os usuários colocarem as informações que desejarem no cadastro, dispondo também de uma opção de anotações. Esta mudança ocorreu devido aos dados principalmente do questionário, mostrando que os usuários apresentavam problemas ao encontrar informações e se organizarem na entrega de atividades, e,
- Lembretes: Foi criada uma nova aba chamada lembretes, usada para ativar notificações e definição de alarmes para entregas de atividades, provas e trabalhos.

Com a avaliação presente, realizou-se o diagrama de classes conceitual (figura 5), que auxiliou no alinhamento dos requisitos, tornando-os mais refinados para o desenvolvimento do terceiro protótipo.

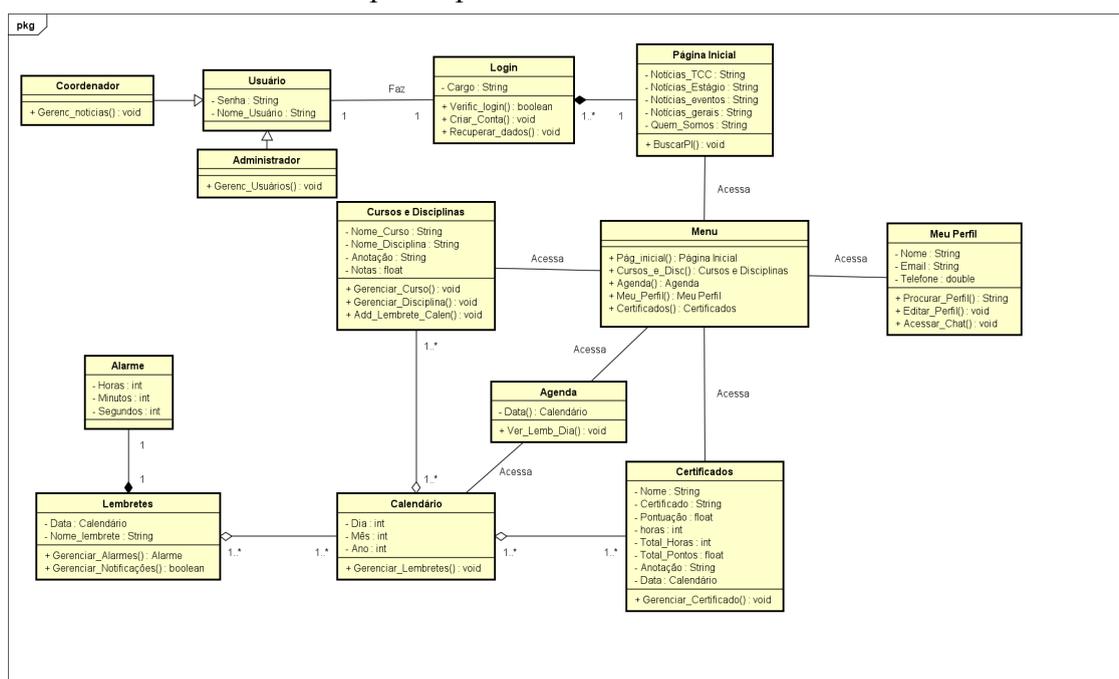


Figura 5: Diagrama de classes. Fonte: os autores (Asth UML).

Após a análise dos requisitos, aplicação das mudanças e realização do diagrama de classes, um novo protótipo foi desenvolvido (Figura 6).

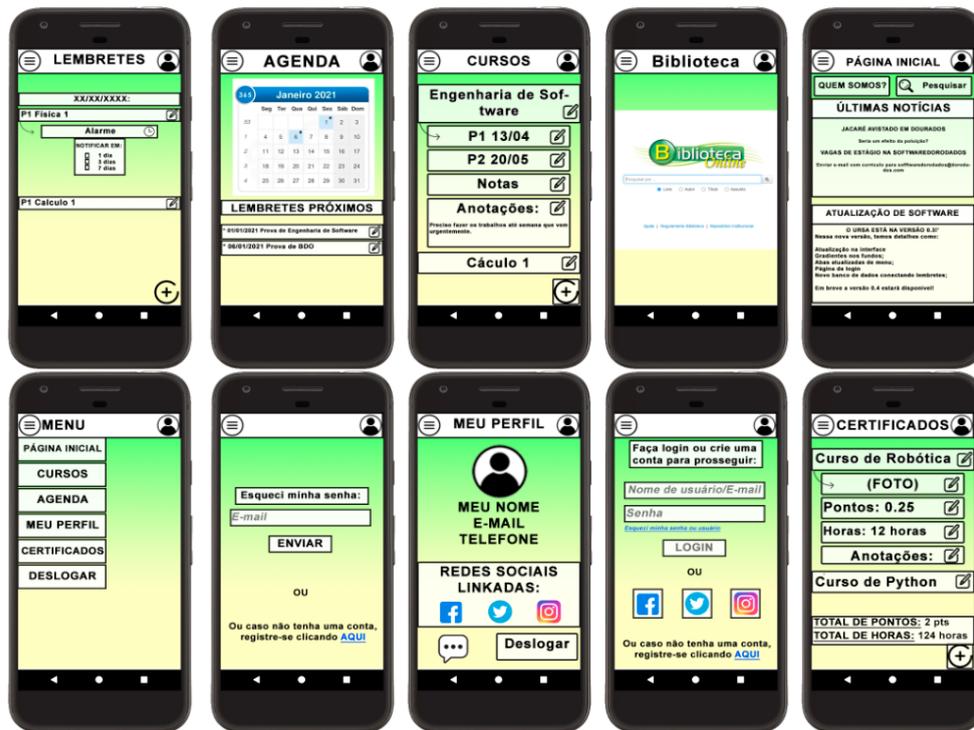


Figura 6: terceiro protótipo. Fonte: os autores (Adobe Photoshop).

Na última iteração, foram obtidas para análise mais 31 respostas do questionário e mais 7 respostas do *card sorting*, totalizando 91 e 32 respostas respectivamente. Como resultado final do questionário observou-se que:

- 88% dos respondentes têm idade entre 18 e 30 anos, definindo assim a faixa etária do público alvo;
- 72,5% usam o sistema operacional Android enquanto 27,5% IOS, o que definiu a plataforma Android como base para ser desenvolvida a aparência do protótipo final;
- O uso de redes sociais foi alto, variando do mais baixo (*twitter*) com 30,8% para o mais alto (*whatsapp*) com 95,6%. Essas informações foram úteis para definir quais redes sociais poderiam ser usadas como forma de login, e também para adequar a função de chat a algo mais próximo do cotidiano;
- A funcionalidade de “aulas gravadas” foi colocada como trabalho futuro, visto que a ideia principal seria a organização pessoal do aluno, e não uma relação professor-aluno. Já “horário de ônibus” e “cardápio RU” foram retirados devido à suas baixas relevâncias. Segue a lista de relevância das atividades de acordo com o questionário:

| | |
|-------------------------------|----------------|
| ■ Divulgação de estágios | 86,8% (79/91); |
| ■ Sala de chat (c/ professor) | 83,5% (76/91); |
| ■ Atividades | 83,5% (76/91); |
| ■ Notas | 82,4% (75/91); |
| ■ Aulas Gravadas | 78% (71/91); |

- | | |
|----------------------------|----------------|
| ■ Agenda | 76,9% (70/91); |
| ■ Divulgação de minicursos | 74,7% (68/91); |
| ■ Biblioteca | 73,6% (67/91); |
| ■ Notícias da faculdade | 59,3% (54/91); |
| ■ Matrículas | 58,2% (53/91); |
| ■ Horário de ônibus | 48,4% (44/91); |
| ■ Lista de eventos | 48,4% (44/91); |
| ■ Cardápio RU | 40,7% (37/91). |
- Algumas funcionalidades extras informadas foram ou poderão ser adaptadas futuramente: opção que auxilie na organização dos certificados, acessibilidade a portadores de deficiência visual e auditiva, geolocalização, e conexão com outros sistemas, como o dropbox;
 - Sobre as informações dos cursos e disciplinas universitárias: 23,1% consideram de baixa dificuldade as formas de encontrarem o que procuram, 67% consideraram como razoável, e 9,9% como alta. Isso mostrou uma necessidade de simplificar o protótipo inicial, apresentando uma interação humano computador (IHC) mais adequada;
 - Das formas que os respondentes mais buscam conteúdos extras, vídeo-aulas (94,4%), artigos (66,3%) e livros (64%) foram os que se destacaram. As importâncias desses dados irão servir para aprimoramentos em trabalhos futuros.

Tratando-se do *card sorting*, que teve grande importância na definição final das abas, organização e localização dos recursos do protótipo, não houve mudanças significativas das respostas anteriores às finais. O Syncaps V3 mostrou uma relação de porcentagem conforme mostra a figura 7.

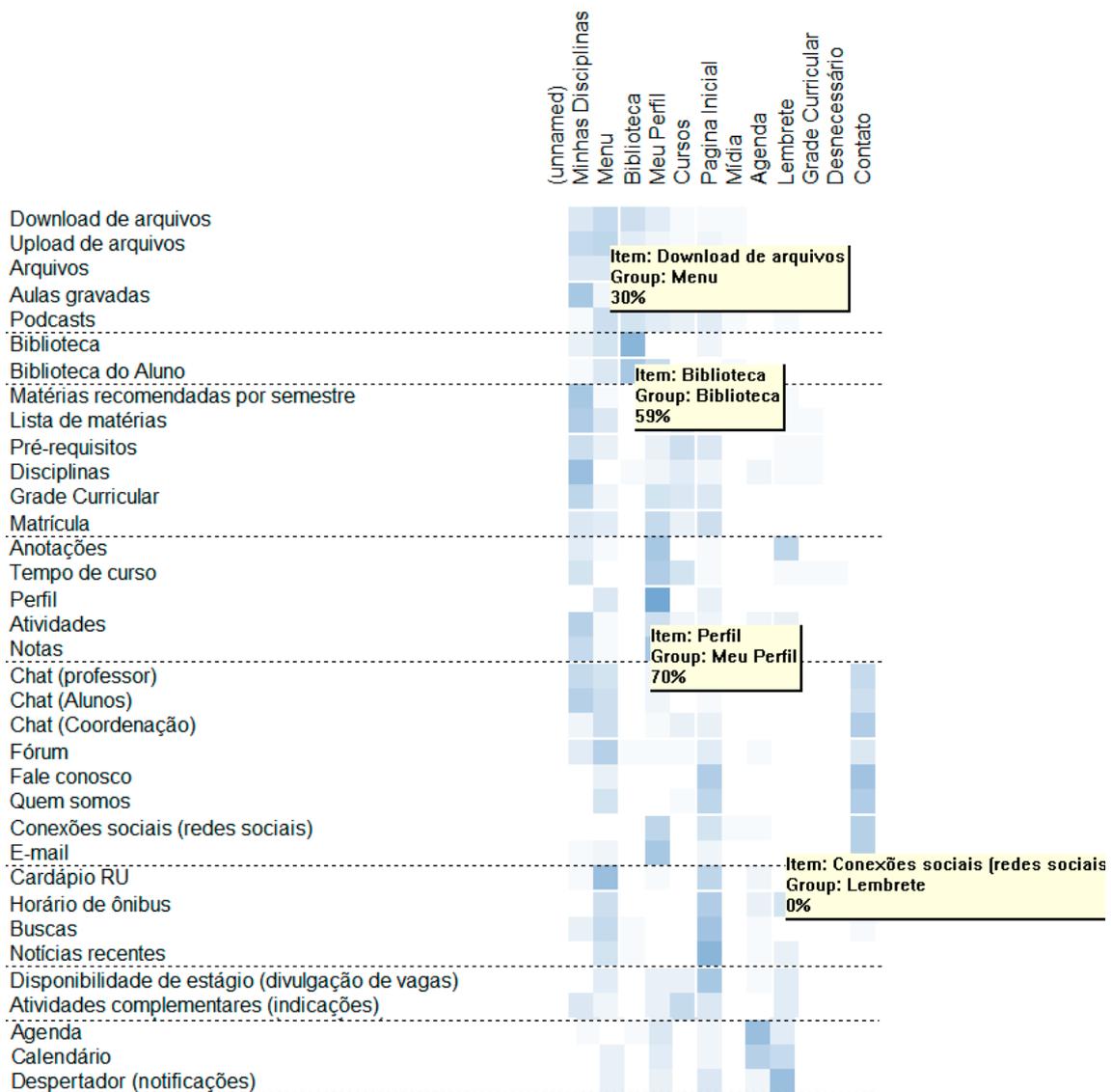


Figura 7: grupo x itens. Fonte: os autores (SynCapsV3 - card sorting).

Estes dados representam os cartões agrupados de outros cartões por proximidade, ou seja, cartões que possuem similaridades nas categorias colocadas. Quanto mais escuro o azul, mais próximos são classificados. Os resultados nele apresentados foram de extrema importância na decisão de em que página colocar funcionalidades que possuem identidades próximas.

Para melhor demonstração, foi montada uma tabela com os resultados apresentados pelo SynCaps¹⁰.

Como outro exemplo de proximidade das cartas, segue a figura 8, representando um dendrograma, também proporcionada pelo SynCaps V3:

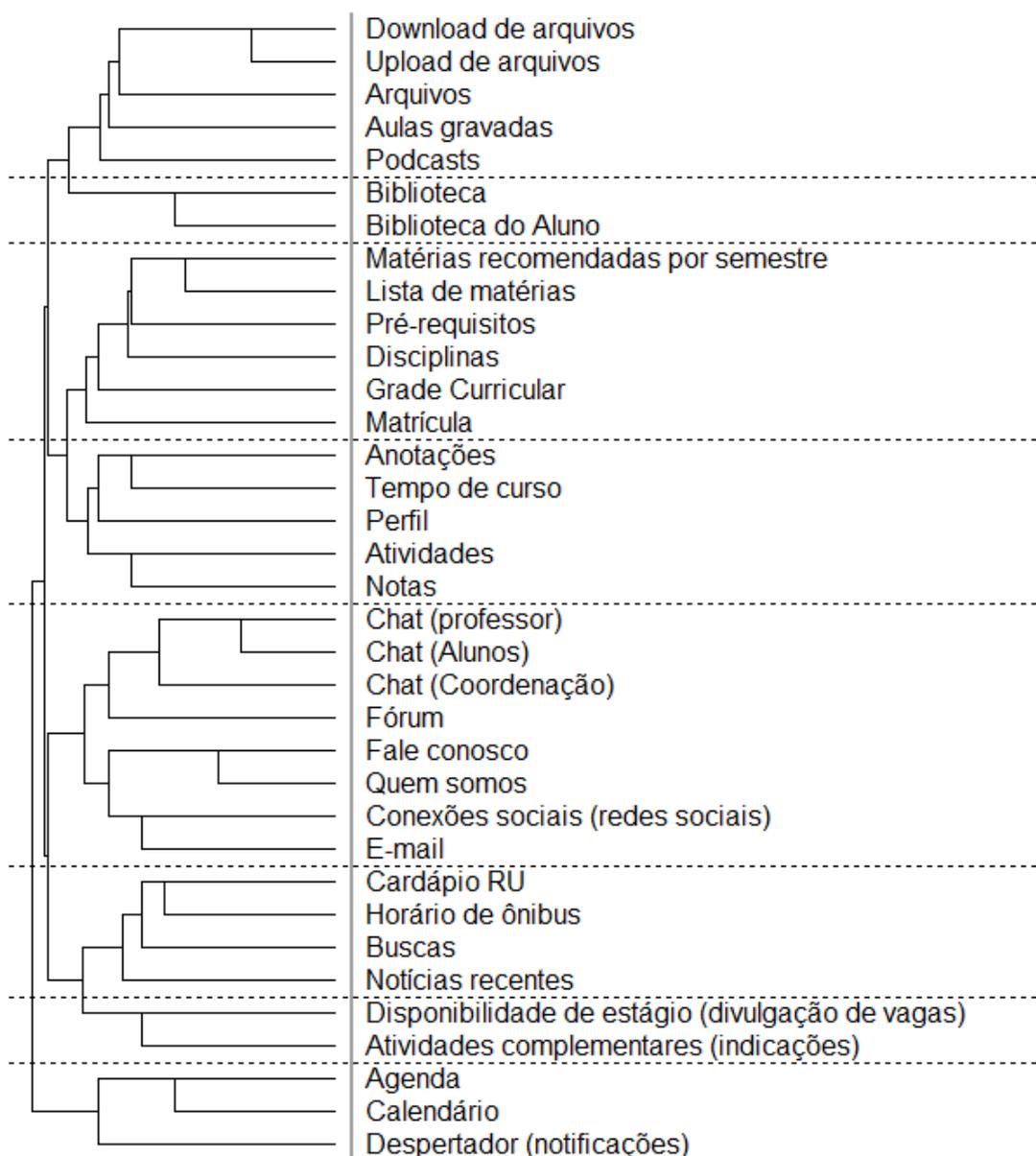


Figura 8: Dendrograma de itens. Fonte: os autores (SynCaps V3 - *card sorting*).

Não foram apresentados problemas ou sugestões consideráveis nas respostas do questionário, já no *card sorting*, várias pessoas relataram problemas com o uso do framework pelo celular, uma vez que este não permitia muitas vezes soltar uma carta em uma pasta, ou ainda colocava outra carta erroneamente.

Com os resultados finais em mãos, foram apresentados os requisitos funcionais (RF) do sistema proposto:

- Página de login: funcionalidades de criar conta (login, senha, email), realizar login com os dados cadastrados ou por redes sociais conectadas;

- Página Inicial: lista de notícias, eventos, estágios, Trabalho de Conclusão de Cursos(TCC) e a atualização do *software*, função de busca por *TAG* e botão superior de “quem somos”;
- Menu com acesso à página inicial, agenda, cursos, meu perfil, certificados e logout;
- Cursos: opção de cadastro e edição de cursos e disciplinas, bem como adicionar lembretes e anotações para provas e atividades;
- Agenda: permitir acessar os dias do calendário bem como os lembretes mais próximos, permitir criar e alterar lembretes, alarmes, anotações e notificações na data;
- Meu perfil: possibilitar edição de informações pessoais, pesquisa de outros usuários e chat de texto;
- Certificados: possuir campos de nome e anotação do certificado e ter uma função de adicionar o link ou imagem do certificado;

Como este trabalho teve o intuito de realizar somente um protótipo, os requisitos não funcionais apresentaram somente recursos que não necessitam de programação para serem definidos. São eles:

- Sistema operacional Android;
- Configuração de acesso por redes sociais (facebook, instagram, twitter e gmail);
- Cargo de coordenadores para atualização de notícias, TCC's e estágios;
- Configuração de e-mail para confirmação e recuperação de contas;
- Notícias conectadas às fontes originais;
- Uso de *TAG* pesquisada para buscas por informações e usuários;
- Base de dados geral para notícias, TCC's e estágios, e,
- Base de dados particular para organização própria.

Concluída a fase de modelagem, deu-se início à produção do protótipo final. Este protótipo foi bem próximo do terceiro, tendo em vista que muito pouco se alterou nas respostas da última fase de coleta das técnicas. As mudanças simples foram a retirada da biblioteca, pois ficou em sexto lugar na relevância. Também foi necessário adicionar cargos para que coordenadores pudessem publicar estágios e TCCs.

Com as funções do Marvel App, foi possível representar as diversas funções do aplicativo, dando uma aparência mais sofisticada ao protótipo (figura 9).

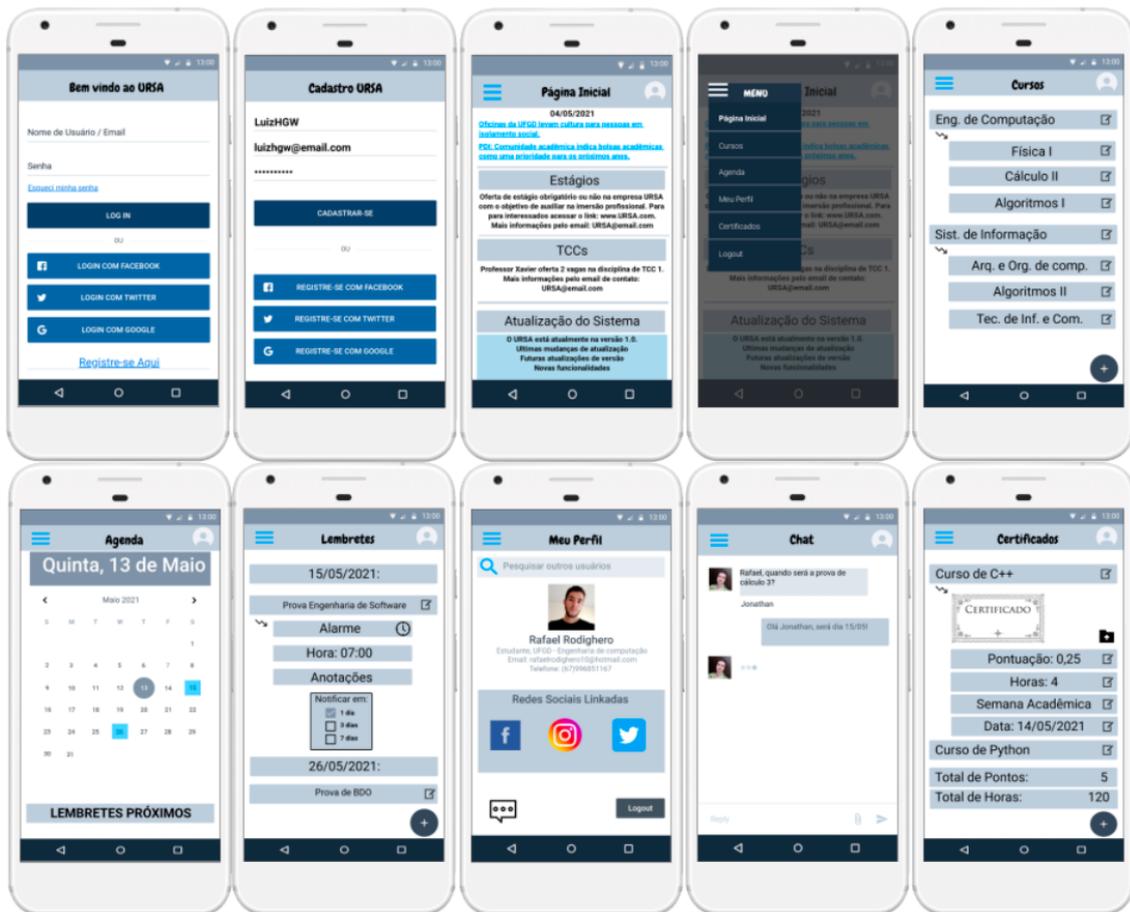


Figura 9: Protótipo final. Fonte: os autores (Marvel App).

No *feedback* final, os respondentes transmitiram estar satisfeitos com a preocupação expressa aos usuários, avaliando o questionário como direto nas questões, porém com ressalvas nas explicações, que alguns classificaram como desnecessariamente extensas. Também classificaram a metodologia de trabalho como agradável, visto que ficou muito simples imaginar o aplicativo, e se sentiram fazendo parte do processo.

7. Conclusão e trabalhos futuros

A prototipação incremental mostrou-se competente quando trabalhada em um processo com várias fases de pesquisa. Ela foi responsável pela organização de cada passo dado no desenvolvimento do protótipo, permitindo com que se retornasse à fase de elicitação de requisitos sempre que necessário. Quanto mais respostas se obtinha nas técnicas abordadas, menos o protótipo se alterava, como aconteceu na evolução do protótipo 3 para o protótipo final. Isso ocorreu, pois as respostas cada vez mais se aproximavam uma das outras, estabelecendo fatores em comum.

O *card sorting*, junto com o SynCaps V3, mostrou-se competente para organizar itens, detectar redundâncias, grupos e itens desnecessários quando há uma coleta prévia de requisitos. O questionário também teve sua efetividade comprovada na coleta de uma amostra de usuários extensa, permitindo alcançar mais de 90 respostas em um período

de aproximadamente 12 semanas, apesar de ser mais dependente do preparo das questões.

O processo do MPS prototipação repetiu-se por quatro vezes, gerando quatro protótipos, um como base da ideia inicial e outros três incrementais, que com o auxílio da técnica questionário, atingiu uma proximidade com as opiniões dos futuros usuários nas questões levantadas. Assim, foi alcançado um processo evolutivo eficiente e que se aproximou dos usuários pretendidos.

A principal dificuldade encontrada no método *card sorting* de forma remota foi que a grande maioria dos participantes (64,84%) não responderam ao *card sorting* através da plataforma online, muitas vezes pela incompatibilidade da plataforma KardSort com o celular (sendo necessário acesso a um computador). O tipo de *card sorting* usado foi o híbrido, onde os participantes puderam criar e sugerir novas categorias, garantindo uma visão mais aberta e permitindo além da organização dos requisitos propostos, a extração de novos.

A plataforma de prototipação Marvel App possui uma interface amigável para se trabalhar e se mostrou fácil de aprender, além de ter diversos tutoriais disponíveis na forma de vídeo ou texto. Apresentou funcionalidades suficientes para a realização deste trabalho mesmo na forma gratuita, sendo que com as funcionalidades pagas notaram-se diversos complementos que podem aprimorar ainda mais a experiência dos usuários, como por exemplo, adicionar links que direcionam a outras páginas. Como resultado, o protótipo final recebeu uma aparência mais lapidada, permitindo demonstrar o funcionamento do aplicativo de forma simples e eficaz.

Como trabalhos futuros, aprimorar a ideia do protótipo permitindo que outras universidades tenham acesso à ideia, trazendo conexões para com a biblioteca e permissões mais abrangentes para coordenadores e professores no aplicativo. Também se pretende aplicar essa metodologia na produção de um aplicativo *mobile* para terceira idade conforme detalhado na seção 2.

8. Referências

- ALMEIDA, Mariana Leite; BANDEIRA, Ana; WEYMAR, Lúcia BC. Information architecture in web: a case study of using the card-sorting method for redesign of the portal of Universidade Federal de Pelotas. Blucher Design Proceedings, v. 1, n. 2, p. 1841-1847, 2014.
- ALVES, Rafael Ferreira; VANALLE, Rosângela Maria. Ciclo de Vida de Desenvolvimento de Sistemas-visão conceitual dos modelos clássico, espiral e prototipação. XXI Encontro Nacional de Engenharia de Produção (ENEGEP). Anais Eletrônicos... Salvador: ABEPRO, 2001.
- ÁVILA, Ana Luiza; SPÍNOLA, Rodrigo Oliveira. Introdução à Engenharia de Requisitos. Engenharia de Software Magazine, Rio de Janeiro, p.46-52, 27 ago. 2019.
- BELGAMO, Anderson; MARTINS, Luiz Eduardo Galvão. Estudo comparativo sobre as técnicas de elicitação de requisitos do software. In: XX Congresso Brasileiro da Sociedade Brasileira de Computação (SBC), Curitiba-Paraná. 2000.

- CABRAL, Ademilson Angelo; DA SILVA, Diego Bezerra; DE SOUZA, Alan Pinheiro. A Problemática do desenvolvimento de software: crise ou calamidade crônica?. Faculdades Integradas de Três Lagoas.
- PINEDA, Castillo; Fabián Humberto et al. Desarrollo de una aplicación móvil que ayude a coordinar la comunicación de la comunidad de transporte de la UPBBGA utilizando bases de datos Nosql para mostrar información en línea. 2018.
- CYSNEIROS, Luiz Marcio; LEITE, J. C. S. P. Requisitos não funcionais: da elicitação ao modelo conceitual. PhD Tese, PUC-RJ, 2001.
- DA SILVA, Silvio Sanches; SAVOINE, Márcia Maria. Análise comparativa de ferramentas computacionais para prototipação de interfaces. Revista Científica do ITPAC, v. 3, 2010.
- VILELA, Rafael Silvestre de Souza; DE CARVALHO, Katia Morais; FAGUNDES, Fabiano. Utilização do Card Sorting na Implementação de um Sistema de Balanced Scorecard para Instituições de Ensino Superior. 2009
- ENGHOLM JR, Hélio. Engenharia de software na prática. Novatec Editora, 2010.
- FUZII, Ricardo Yoshihiro Mastrocolla; SOUZA, Rogéria Cristiane Gratão; TRONCO, Mário Luís. Apoio Automatizado para Aplicação de Técnicas de Elicitação de Requisitos. Revista Eletrônica de Sistemas de Informação, v. 8, n. 1, 2009.
- LESSA, Rafael Orivaldo; LESSA JUNIOR, Edson Orivaldo. Modelos de Processos de Engenharia de Software. Link para o PDF: http://xps-project.googlecode.com/svn-history/r43/trunk/outros/02_Artigo.pdf, 2009.
- PADOVANI, Stephania; RIBEIRO, Murilo Amgarten. Card sorting: adaptação da técnica para aplicação ao design de sistemas de informação não digitais. InfoDesign-Revista Brasileira de Design da Informação, v. 10, n. 3, p. 293-312, 2013.
- PEREIRA, Alexandre; NARDUCHI, Fábio; DE MIRANDA, Maria Geralda. Biopolítica e Educação: os impactos da pandemia do covid-19 nas escolas públicas. Revista Augustus, v. 25, n. 51, p. 219-236, 2020.
- PINTON, Erich Nelson Silveira; CAMARGO NETO, J.; VISOLI, Marcos Cezar. Roda da Reprodução iOS: aplicativo móvel para gestão de rebanho leiteiro. In: Embrapa Informática Agropecuária-Artigo em anais de congresso (ALICE). In: MOSTRA DE ESTAGIÁRIOS E BOLSISTAS DA EMBRAPA INFORMÁTICA AGROPECUÁRIA, 13., 2017, Campinas. Resumos expandidos... Brasília, DF: Embrapa, 2017., 2017.
- PRESSMAN, Roger S.; MAXIM, Bruce R.. Engenharia de Software: UMA ABORDAGEM PROFISSIONAL. 8. ed. São Paulo: Amgh Editora Ltda, 2016. 939 p. Revisão técnica: Reginal Arakaki, Julio Arakaki, Renato Manzan de Andrade.
- PRIKLADNICKI, Rafael. Problemas, desafios e abordagens do processo de desenvolvimento de software. 2002. Trabalho Individual I, FACIN-PPGCC, PUCRS, Porto Alegre, 2004.
- RIBEIRO, Fabíola Gonçalves; SOUZA, Leandro Rodrigues da Silva. A IMPORTÂNCIA DA ENGENHARIA DE REQUISITOS PARA O CICLO DE DESENVOLVIMENTO DE SOFTWARE DE TEMPO REAL. 2012. 1 f. Resumo

aprovado e apresentado - Curso de Sistemas Para Internet, Centro de Ensino Superior de Catalão, Catalão, 2012.

ROSEMBERG, Carlos et al. Prototipação de software e design participativo: uma experiência do Atlântico. IHC, v. 8, p. 312-315, 2008.

SÁ, MI da F. O ensino da disciplina de arquitetura de informação: uma aplicação da técnica de Card Sorting. In: XXV Congresso Brasileiro de Biblioteconomia, Documento e Ciência da Informação–Florianópolis, SC, Brasil. 2013.

SEGUNDO, Cristovam Antonio. Pesquisa Exploratória em Elicitação de Requisitos: Teoria e Prática. Monografia - Curso de Bacharelado em Ciência da Computação, Universidade Federal Rural de Pernambuco, Recife, 2015.

SILVA, Samuel Fabiano Barbosa. ENGENHARIA DE REQUISITOS: Uma análise das técnicas de levantamento de requisitos. 2012. 42 f. Monografia (Especialização) - Curso de Ciência da Computação, Universidade Fumec, Belo Horizonte, 2012.

SOMMERVILLE, Ian. Engenharia de Software. 9. ed. São Paulo: Pearson Prentice Hall, 2013. 544 p. Tradução de Kalinka Oliveira, Ivan Bosnic. Revisão Técnica de Prof. Dr. Kechi Hirama.

STANDISH GROUP. CHAOS report. Dennis, MA 02638, USA, 1995

WOOD, Jed R.; WOOD, Larry E. Card sorting: current practices and beyond. Journal of Usability Studies, v. 4, n. 1, p. 1-6, 2008.