**UNIVERSIDADE ESTADUAL DO SUDOESTE DA BAHIA – UESB**

**DEPARTAMENTO DE CIÊNCIAS EXATAS E TECNOLOGIAS - DCET**

**CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO**

**ICU SURVIVING (UM SOFTWARE DE GESTÃO DA QUALIDADE DE UTIs):**PROPOSTA DE INTERFACE COM FOCO NA EXPERIÊNCIA DO USUÁRIO DE DISPOSITIVOS COM SISTEMA OPERACIONAL ANDROID

VITÓRIA DA CONQUISTA,

2015

**FRANKLIN TED LELIS SILVA SOUZA**

**ICU SURVIVING (UM SOFTWARE DE GESTÃO DA QUALIDADE DE UTIs):** PROPOSTA DE INTERFACE COM FOCO NA EXPERIÊNCIA DO USUÁRIO DE DISPOSITIVOS COM SISTEMA OPERACIONAL ANDROID

Trabalho de Conclusão de Curso de Graduação apresentado ao Departamento de Ciências Exatas e Tecnologias (DCET) como requisito parcial à obtenção do título de Bacharel em Ciência da Computação pela Universidade Estadual do Sudoeste da Bahia (UESB).

Orientador: Prof. Dr. Hélio Lopes dos Santos

VITÓRIA DA CONQUISTA,

2015

Ficha Catalográfica

CRISTIANE LUZ SANTANA

ICU SURVIVING (UM SOFTWARE DE GESTÃO DA QUALIDADE DE UTI’S): PROPOSTA DE INTERFACE COM FOCO NA EXPERIÊNCIA DO USUÁRIO DE DISPOSITIVOS COM SISTEMA OPERACIONAL ANDROID

Esta monografia foi julgada adequada e aprovada para a obtenção do título de bacharel em Ciência da Computação pela Universidade Estadual do Sudoeste da Bahia- UESB.

Vitória da Conquista, \_\_\_\_ de \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ de 2015.

Apresentada à Banca Examinadora composta pelos professores:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Prof. Dr. Hélio Lopes dos Santos.

Universidade Estadual do Sudoeste da Bahia- UESB

Orientador

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Prof. Dra. Alzira Ferreira da Silva

Universidade Estadual do Sudoeste da Bahia- UESB

Membro

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Prof. Esp. Fabrício de Sousa Pinto

Universidade Estadual do Sudoeste da Bahia- UESB

Membro

Dedico este trabalho a minha família que sempre me incentivou e durante toda esta trajetória compreendeu minha ausência e mesmo distante se fez presente.

**AGRADECIMENTOS**

Primeiramente, agradeço a Deus por permitir e dar forças para concluir esta jornada.

À minha família, pelo apoio e amor incondicional dedicados em toda minha vida.

À minha fiel companheira, Carla Cotrim, por compreender meus defeitos e torná-los menores, provando o seu amor a cada dia.

Ao meu orientador prof. Hélio Lopes dos Santos pela motivação e paciência prestados.

A todos os professores com os quais pude aprender.

Aos colegas de curso pela amizade e companheirismo.

Aos meus colegas de trabalho (Cris Luz, Cláudio, Maurina, Renê Céu, Marcos Pereira, André Luiz, e muitos outros) , pelo apoio integral e amizade sem fins lucrativos.

Ao meu padrinho Geraldo Meira, pelos conselhos diários e compartilhamento de experiências.

Enfim, a todos que de alguma forma contribuíram a esta caminhada.

**RESUMO**

O desenvolvimento de interfaces tornou-se uma atividade desafiadora, visto o grande potencial interativo e heterogeneidade dos dispositivos computacionais, em especial, smartphones e tablets. Conhecer as necessidades, bem como criar soluções que melhorem a experiência do usuário são, certamente, os principais requisitos para o sucesso de uma aplicação mobile. Este trabalho visa exemplificar os principais desafios do designer ao projetar interfaces, a fim de permitir, de forma intuitiva e eficaz, a utilização de aplicativos para dispositivos móveis. Como estudo de caso, elaborou-se um protótipo de alta fidelidade para o Sistema de gestão de qualidade de UTIs, *ICU Surviving*, criando uma situação real de desenvolvimento. As etapas do processo, bem como seus componentes e ferramentas utilizados, foram pesquisados e discutidos com base em padrões de design de interfaces para dispositivos móveis com sistema operacional *Android*.

PALAVRAS-CHAVE: Interface; Design de interfaces; Dispositivos móveis; Android.

**ABSTRACT**

The development of interfaces has become a challenging activity, as the great interactive potential and variety of computing devices, especially smartphones and tablets. Knowing the needs and create solutions that improve the user experience are certainly the main requirements for the success of a mobile application. This paper aims to illustrate the main challenges of the designer when designing interfaces in order to allow, intuitively and effectively, the use of mobile applications. As a case study, we elaborated a high-fidelity prototype for the quality management system of ICU, ICU Surviving, creating a real situation of development. The process steps as well as their components and tools used, have been researched and discussed based on interface design standards for mobile devices with Android OS.

KEYWORDS: Interface; Interfaces design ; Mobile devices; Android.

**LISTA DE ILUSTRAÇÕES**

[Figura 1 - Diversidade de dispositivos com sistema operacional Android. 17](#_Toc417408862)

[Figura 2 - Temas holo-light (esquerda) e holo-dark (direita) 18](#_Toc417408863)

[Figura 3 - Affordance nos componentes da Action Bar 19](#_Toc417408864)

[Figura 4 - Configuração para diferentes densidades de tela 22](#_Toc417408865)

[Figura 5 - Tamanhos de fontes utilizados na UI Android 22](#_Toc417408866)

[Figura 6 - Ícones de lançamento Android 23](#_Toc417408867)

[Figura 7 - Ícones e Identidade visual 24](#_Toc417408868)

[Figura 8 - Guia ilustrado para design de ícones 25](file:///C:\Users\Franklin\OneDrive\Monografia%20finallllllllllllllll\MONOGRAFIA%20CC%20Final.docx#_Toc417408869)

[Figura 9 - Diagrama arquitetura da informação 29](#_Toc417408870)

[Figura 10 - Wireframes 30](#_Toc417408871)

[Figura 11 - Ferramenta de prototipagem Briefs 33](#_Toc417408872)

[Figura 12 - Arquitetura da informação do sistema ICU Surviving 40](#_Toc417408873)

[Figura 13 - Ícone lançador do ICU Surviving 41](file:///C:\Users\Franklin\OneDrive\Monografia%20finallllllllllllllll\MONOGRAFIA%20CC%20Final.docx#_Toc417408874)

[Figura 14 - Laucher icon do ICU Surviving em várias escalas 42](#_Toc417408875)

[Figura 15 - Visão geral dos protótipos 44](#_Toc417408876)

[Figura 16 - ICU Surviving: Tela de login 45](file:///C:\Users\Franklin\OneDrive\Monografia%20finallllllllllllllll\MONOGRAFIA%20CC%20Final.docx#_Toc417408877)

[Figura 17 – ICU Surviving: Lista de internações 46](file:///C:\Users\Franklin\OneDrive\Monografia%20finallllllllllllllll\MONOGRAFIA%20CC%20Final.docx#_Toc417408878)

[Figura 18 – ICU Surviving: Registrar internação do paciente 47](file:///C:\Users\Franklin\OneDrive\Monografia%20finallllllllllllllll\MONOGRAFIA%20CC%20Final.docx#_Toc417408879)

[Figura 19 - ICU Surviving: Detalhe internação 48](file:///C:\Users\Franklin\OneDrive\Monografia%20finallllllllllllllll\MONOGRAFIA%20CC%20Final.docx#_Toc417408880)

[Figura 20 - ICU Surviving: Registrar dados SAPS 3 e APACHE II 49](#_Toc417408881)

**LISTA DE TABELAS**

[Tabela 1 - Equações de admissão SAPS 3 customizadas para diferentes regiões geográficas 36](#_Toc417408882)

**LISTA DE QUADROS**

[Quadro 1 - Termos sobre densidade e seus significados 20](#_Toc417409015)

[Quadro 2 - Qualificadores de configuração que permitem fornecer recursos especiais para diferentes tipos de tela. 21](#_Toc417409016)

[Quadro 3 - Gestos do Android 26](#_Toc417409017)

**LISTA DE SIGLAS**

**ANVISA–** Agência Nacional de Vigilância Sanitária

**APACHE –** *Acute Physiology And Chronic Health Evaluation*

**CSS –** *HyperText Markup Language*

**HTML –** *HyperText Markup Language*

ICU – *Intensive Care Unit*

**iOS –** *iPhone Operation System*

**RDC –** Resolução da Diretoria Colegiada

**SAPS –** *Simplified Acute Physiology Score*

SO – Sistema Operacional

UI – *User Interface*

UTI – Unidade de Terapia Intensiva

**SUMÁRIO**

[1 INTRODUÇÃO 13](#_Toc417409232)

[1.1 OBJETIVOS 14](#_Toc417409233)

[1.1.1 Objetivos Específicos 14](#_Toc417409234)

[1.2 JUSTIFICATIVA 14](#_Toc417409235)

[1.3 ESTRUTURA DO TRABALHO 15](#_Toc417409236)

[2 PROJETO E IMPLEMENTAÇÃO DE INTERFACES PARA DISPOSITIVOS MOVEIS 16](#_Toc417409237)

[2.1 OS DESAFIOS DO MOBILE DESIGN 16](#_Toc417409238)

[2.1.1 Considerações ao design de interfaces para Android 17](#_Toc417409239)

[2.1.2 Padrões de design Android (design patterns) 27](#_Toc417409240)

[2.1.3 Definindo uma proposta 29](#_Toc417409241)

[3 SISTEMAS DE GESTÃO DA QUALIDADE DE UTI’S 35](#_Toc417409242)

[3.1 SAPS 3 36](#_Toc417409243)

[3.2 APACHE II 38](#_Toc417409244)

[4 ANÁLISE CONTEXTUAL E ELABORAÇÃO DOS PROTÓTIPOS 39](#_Toc417409245)

[4.1 Escopo, conceito e planejamento 39](#_Toc417409246)

[4.2 Construção dos protótipos 44](#_Toc417409247)

[5 CONCLUSÕES E TRABALHOS FUTUROS 51](#_Toc417409248)

[REFERÊNCIAS 52](#_Toc417409249)

[ANEXO 1 – TABELA DE PONTUAÇÃO - SAPS 3 54](#_Toc417409250)

[ANEXO 2 – FORMULÁRIO PADRÃO – APACHE II 55](#_Toc417409251)

# INTRODUÇÃO

Indivíduos que dominam novas tecnologias têm vantagens em relação àqueles que sentem dificuldades em manuseá-las. Sistemas computacionais, bem como interfaces acessíveis e interativas são novas tecnologias, cujo processo de disseminação é cada dia mais rápido.

A interface representa a porção de um sistema com a qual o usuário mantém contato, além de ser considerada por alguns, como o sistema propriamente dito (BARBOSA, 2010). Portanto, é a principal responsável em permitir a interação humano-computador, baseando-se na criação de experiências que melhorem e estendam a maneira como as pessoas se relacionam com artefatos tecnológicos.

Para o projeto de interfaces voltado ao desenvolvimento de aplicações móveis, existem regras e padrões específicos recomendados a fim de permitir maior compatibilidade de recursos, visto que o ambiente mobile constitui-se de plataformas distintas, além de atender a usuários de diferentes níveis de experiência e necessidade.

Talvez o termo mais difundido ao tratar-se de princípios para desenvolvimento de interfaces do usuário seja usabilidade. Segundo especialistas, usabilidade diz respeito ao grau qualitativo do uso de um sistema interativo, levando em consideração a relação estabelecida entre usuário, tarefa, interface e demais aspectos do ambiente no qual o usuário utiliza o sistema. Para o desenvolvimento de um sistema com usabilidade, faz-se necessária a análise criteriosa dos vários componentes de seu contexto de utilização, além da participação ativa do usuário nas decisões do projeto em questão. No entanto, há outras práticas de projeto através das quais uma interface poderá favorecer o estabelecimento da usabilidade na relação humano – computador. Estas práticas se fazem respeitando critérios, princípios e/ou heurísticas de usabilidade propostas por diversos autores, especialistas e instituições.

Nos últimos anos, as aplicações de softwares têm crescido suas possibilidades, seja através da utilização da Internet, computação ubíqua, realidade aumentada, entre outros. Atrelado a este cenário está também o crescimento do processo competitivo entre os desenvolvedores destas aplicações, que por sua vez, vem a estimular a criação de soluções cada vez mais interativas.

Para este trabalho tem por interesse mensurar os possíveis benefícios a serem obtidos pela utilização de boas práticas de design de interação em projetos de design de interfaces especificamente mobile, por acreditar que aplicações deste tipo possuem grande potencial interativo, maior disponibilidade em relação aos aplicativos desktop, além de estarem em rápida e constante evolução.

## OBJETIVOS

Analisar as principais dificuldades, bem como algumas soluções relacionadas, ao se propor uma interface para o sistema *ICU Surviving*, com foco na experiência do usuário de dispositivos com sistema operacional *Android*

### Objetivos Específicos

* Analisar as principais definições de design, padrões e estruturas referenciadas no guia de desenvolvimento *Google* – *Android*;
* Identificar quais técnicas e conceitos se adequaram ao projeto proposto, a fim de promover uma boa relação entre elementos visuais, interatividade e funcionalidades;
* Desenvolver protótipos das telas baseando-se em alguns princípios padrões de design propostos.

## JUSTIFICATIVA

A escolha do tema justificou-se pela importância de se analisar os padrões relacionados ao design de interfaces para dispositivos com sistema operacional *Android* que contribuem como guia de desenvolvimento de aplicações de qualidade.

O mercado de aplicações para dispositivos móveis está em alta e por isso, promove uma crescente demanda por produtos cada vez mais competitivos. Consequentemente, necessita-se cada vez mais de mão de obra especializada, uma ótima oportunidade para aqueles que possuem conhecimento nesta área de desenvolvimento.

É fundamental considerar que, por se tratar de sistema construído para *Android* e voltado à área de saúde, sua contribuição será de grande valia às instituições usuárias da aplicação devido à sua ótima relação custo-benefício.

A análise permitiu compreender muitos dos diversos aspectos envolvidos no design de interfaces móveis para *Android* como desenvolver protótipos de alta fidelidade utilizando os guias de estilo e padrões em um estudo de caso real.

Além disto, este projeto de pesquisa poderá servir de fonte para trabalhos futuros de outro acadêmicos ou entusiastas da área.

## ESTRUTURA DO TRABALHO

Este projeto é constituído por cinco capítulos. O primeiro é a introdução, no qual são contemplados o tema e suas delimitações. Trata-se da apresentação do proposta de pesquisa, seus objetivos geral e específicos e a justificativa do trabalho.

Os capítulos 2 e 3 constituem-se dos referenciais teóricos que fundamentam os conceitos abordados na pesquisa, servindo de apoio ao estudo. A seção 2 discute os conceitos e desafios voltados ao projeto e implementação de interfaces para dispositivos móveis, levantando questões sobre requisitos da plataforma, guias estilo, padrões de design e boas práticas para elaboração de uma proposta de interface. Na seção 3 são abordados os conceitos sobre o estudo e aplicação dos índices de prognósticos *SAPS*[[1]](#footnote-1) 3 e *APACHE* *II*[[2]](#footnote-2) para gestão da qualidade de UTIs[[3]](#footnote-3) .

No capítulo 4 são apresentadas as etapas utilizadas na análise e construção dos protótipos, bem como tópicos relacionados aos requisitos da aplicação *ICU Surviving*[[4]](#footnote-4), Apresenta ainda um esboço do protótipo e notas explicativas que justificam a aplicação das técnicas e conceitos utilizados.

O capítulo 5 contempla as considerações finais, pelas quais são dispostas as impressões sobre os resultados da pesquisa e do protótipo proposto, além de sugestões à trabalhos futuros. São apresentados, também, as referências e o apêndice do trabalho.

# PROJETO E IMPLEMENTAÇÃO DE INTERFACES PARA DISPOSITIVOS MOVEIS

Esta seção apresenta os requisitos, termos técnicos e padrões do design de interface para dispositivos móveis necessários à implementação do protótipo da interface do sistema *ICU Surviving.*

## OS DESAFIOS DO MOBILE DESIGN

Segundo Cuello (2013), desenhar aplicações para dispositivos móveis não é uma tarefa fácil, especialmente quando o designer é iniciante nesta modalidade ou está habituado a trabalhar com outros ambientes de desenvolvimento, tais como o WEB, um contexto bastante distinto do abordado neste trabalho.

Além do entendimento sobre os aspectos gerais de *IHC* [[5]](#footnote-5)e a arquitetura da aplicação a ser desenvolvida, projetar para dispositivos móveis, requer conhecimentos específicos desta modalidade de desenvolvimento (*Desktop* x *Smartphone*/*tablet*), bem como da plataforma a ser utilizada (*IOS[[6]](#footnote-6), Android, Windows Phone*, etc.).

Ainda de acordo com Cuello (2013), diferente de um fenômeno aparentemente recente, as aplicações já convivem entre nós há bastante tempo. Ultimamente, complementa o autor, estas aplicações não só se tornaram mais populares e atrativas aos usuários, como também aos designers e desenvolvedores que, por sua vez, aproveitam-se das possibilidades oferecidas pelos telefones com telas de maior qualidade. Seguindo este raciocínio, afirma que apesar de um desafio, projetar para dispositivos móveis é uma boa oportunidade de se entrar em um mercado de crescente demanda por melhores ferramentas de comunicação e promoção.

2.2 PROJETO DE INTERFACES PARA ANDROID

Diante de um projeto de design para aplicativos *Android*, o designer precisa se atentar a uma série de requisitos e características que tendem a influenciar no resultado final da produção. Faz-se necessário conhecê-los de forma que o projetista possa identificar o contexto de cada módulo ou funcionalidade do aplicativo e consequentemente utilizar os princípios e padrões adequados. No guia oficial para desenvolvedores *Android* (disponível em *http://developer.Android.com/design/index.html*), entre outras informações, são abordados diversos guias de estilo, tais como: Heterogeneidade de dispositivos e displays; Temas; Respostas ao toque; Resolução de tela; Tipografia; Cor; Iconografia; Customização e Textos amigáveis.

### Considerações ao design de interfaces para Android

A seguir, uma abordagem aos guias de estilo pertinentes ao design de interfaces para dispositivos com sistema operacional *Android*.

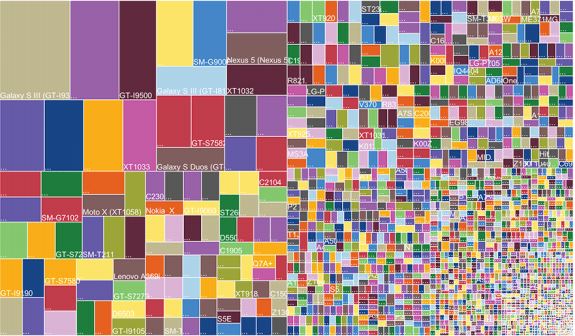
1. O ecossistema multi-screen Android

Atualmente, o SO[[7]](#footnote-7) *Android* está presente em diversos dispositivos dos mais variados tamanhos e formatos. Esta heterogeneidade exige da plataforma um alto grau de flexibilidade. É possível projetar telas adaptativas que suportem desde pequenos smartphones à *tablets* utilizando-se de técnicas que produzam layouts (ANDROID DEVELOPERS, 2014):

* Escalonáveis - adaptáveis à largura e altura do display;
* Otimizados – possibilitando combinações multi-painel para total aproveitamento em telas de dispositivos maiores;
* Sensíveis à resolução – Configuração de recursos para diferentes densidades de tela (DPI) visando garantir que o aplicativo tenha a mesma aparência em qualquer dispositivo;

De acordo com Leal (2015), o sistema *Android* está presente em mais de um bilhão de dispositivos que em sua maioria, são *smartphones* e *tablets*. No entanto, também pode ser encontrado em outros dispositivos como automóveis, TVs, relógios inteligentes, etc. Esses números refletem o sucesso da plataforma, mas torna a tarefa de seus desenvolvedores bastante desafiadora. A Figura 2.1 mostra a diversidade de dispositivos que utilizam o sistema *Android*, segundo estudo da *OpenSinal*.

Figura - Diversidade de dispositivos com sistema operacional Android.

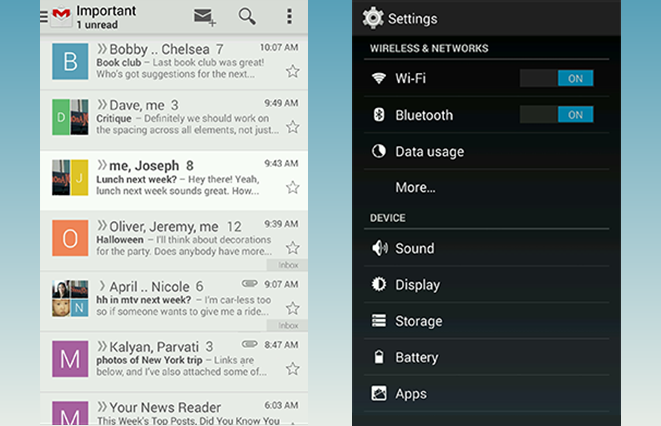


Fonte: http://pt.opensignal.com/reports/2014/android-fragmentation/

1. Temas

Com o objetivo de oferecer um padrão visual consistente aos aplicativos ou atividades, a API *Android* permite, à partir da versão 4.0, a utilização de temas padrão, cujos estilos definem as propriedades visuais dos elementos que compõem a interface do usuário (LEHTIMAKI, 2013). Cabem aos desenvolvedores e designers optar por qual tema mais se adequará ao contexto de sua aplicação, utilizando-os como base para a criação. Além disto, é possível customizá-los implementando estilos visuais próprios. A Figura 2 mostra dois exemplos de telas utilizando os temas *Holo Light* e *Holo Dark*, respectivamente.

Figura - Temas holo-light (esquerda) e holo-dark (direita)



Fonte: http://developer.android.com/design/style/themes.html

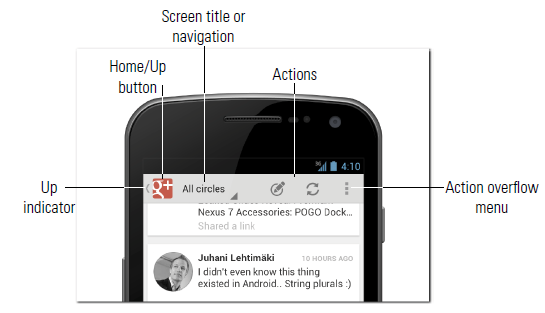
1. Resposta ao toque

Durante utilização de determinada aplicação em dispositivo com SO *Android*, o usuário poderá sentir dificuldades quanto a que passos seguir, a fim de se chegar a uma porção desejada do sistema ou recurso. Neste cenário, o produto deve estar preparado para responder adequadamente às ações do utilizador, permitindo uma navegação fluida e intuitiva. Também conhecido pelo termo *Feedback*, em outras palavras, significa que o usuário nunca diria: 'O que está acontecendo?’ (NEIL, 2014).

Segundo Lehtimaki (2013), é errôneo considerar a interação com telas sensíveis ao toque como algo natural, visto que seres humanos são acostumados a manipular objetos tridimensionais, os quais possuem atributos como peso, textura, entre outros que dão mais sentido à interação. O autor ressalta, por estes e outros fatores, a importância do designer em planejar cuidadosamente a interface como qualquer outro método de interação.

Por padrão, a API *Android* permite a utilização de recursos visuais que vão desde sombreamento, iluminação suave e esmaecimento de itens de layout como indicadores do estado sistema (ANDROID DEVELOPERS, 2014), à aplicação de técnicas de *affordance*, com as quais, o designer se utiliza de elementos ou ações que inferem significado durante os processos de interação com o usuário (LEHTIMAKI, 2013).

Figura - Affordance nos componentes da Action Bar



Fonte: Lehtimaki, 2013

1. Métricas e grids

Como mencionadas na seção 2.1.1, além da variação de tamanho, as ofertas de displays disponíveis também se diferem quanto à densidade de pixels, criando uma demanda por parte do designer em planejar um layout otimizado, que permaneça consistente em qualquer contexto. De acordo com as recomendações do portal Google Developers (2014), é preciso pensar no layout, analogamente, como se o despejasse em áreas particulares de tamanho e densidade de pixels. Estas áreas, quanto ao tamanho, são classificadas como *Phone* (inferior a 600dp) e *Tablet* (maior ou igual a 600dp). Quanto à densidade, são definidos como LDPI, MDPI, HDPI, XHDPI, XXHDPI e XXXHDPI.

* Pixel, px, ppi, dpi, dp e sp - Existem muitos termos utilizados quando se remete à densidade de tela, e é comum que haja confusão ao interpretá-los. O Quadro 1 apresenta alguns destes termos e seus verdadeiros significados (KOMATINENI et al., 2013).

Quadro - Termos sobre densidade e seus significados

|  |  |  |
| --- | --- | --- |
| Termo | Definição | Significado |
| Pixel | Um ponto de cor em uma imagem digital | Um pixel pode ser uma coisa física em uma tela, ou podem ser os pontos que compõem uma imagem digital (em arquivo ou na memória) |
| Px | Pixel da tela *Android* | Um minúsculo ponto físico em uma tela de vídeo que exibe uma cor. Também conhecido como pixel absoluto. O termo px é usado como unidade de dimensão no *Android*. |
| PPI | *pixels per inch* (pixels por polegada) | O número real de pixels por polegada exibidos na tela |
| DPI | *dots per inch* (pontos por polegada) | DPI começou como um conceito para mídia impressa, mas às vezes é usado para descrever monitores de vídeo. No entanto, um elemento de imagem de vídeo é geralmente composto de vários pixels, assim o termo DPI fica confuso em termos de vídeo e seu uso, não recomendado neste contexto. |
| DP | *density-independent pixel* | Técnica utilizada para facilitar a consistência visual de elementos de layout em displays com diferentes densidades. Toma-se por referência o layout MDPI (~ 160 DPI). Automaticamente o SO aumenta ou diminui as escalas dos elementos e seleciona os ícones adequados de forma transparente ao usuário. |
| SP | *scale-independent pixel* | Muito semelhante à dp, mas utilizado para fontes apenas. |

Fonte: Komatineni et al.

* Qualificadores de configuração - O *Android* suporta diversos qualificadores de configuração que permitem ao desenvolvedor controlar como o SO seleciona os recursos alternativos baseado nas características da tela do dispositivo em uso. Um qualificador de configuração é uma notação que pode ser acrescentada a um diretório de recursos em um projeto *Android* e especifica a configuração de finalidade do conteúdo de cada repositório. O quadro 2 apresenta detalhes sobre os qualificadores de configuração *Android* (ANDROID DEVELOPERS, 2014).

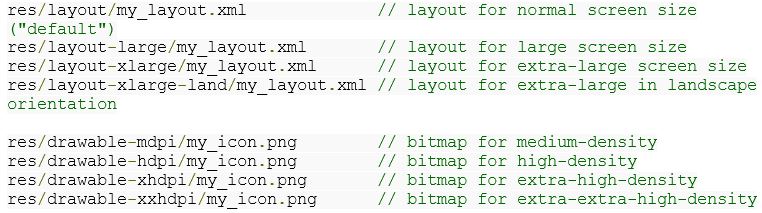
Entrando um pouco no mérito da programação, a figura 4 exemplifica a configuração dos recursos em um aplicativo que fornece modelos de layouts distintos (para diferentes tamanhos de tela) e imagens de vários tamanhos para suportar diferentes densidades (MDPI, HDPI e XHDPI).

Quadro - Qualificadores de configuração que permitem fornecer recursos especiais para diferentes tipos de tela.

|  |  |  |
| --- | --- | --- |
| Característica da tela | Qualificador | Descrição |
| Tamanho | *Small* | Recursos para pequenas tela. |
| *Normal* | Recursos para  telas de tamanhos normais . (Este é o tamanho de referência). |
| *Large* | Os recursos para as telas consideradas grandes . |
| *Xlarge* | Recursos para telas de tamanho extragrandes . |
| Densidade | *LDPI* | Recursos para telas de baixa densidade (LDPI ) (~ 120 dpi). |
| *MDPI* | Recursos para telas de média-densidade (MDPI ) (~160dpi). (Isto é a densidade de linha de base). |
| *HDPI* | Recursos para telas de alta densidade (HDPI ) (~ 240 dpi). |
| *Xhdpi* | Recursos para-extra-alta densidade ( xhdpi ) telas (~ 320dpi). |
| *Xxhdpi* | Recursos para-extra-extra-alta densidade ( xxhdpi ) telas (~ 480dpi). |
| *Xxxhdpi* | Recursos para-extra-extra-extra-alta densidade ( xxxhdpi usos) (~ 640dpi) |
| *Nodpi* | Recursos para todas as densidades. Estes são recursos independentes de densidade. O sistema não dimensionará recursos marcados com esta qualificação, independentemente da densidade da tela atual. |
| *Tvdpi* | Recursos para telas em algum lugar entre MDPI e HDPI; cerca de 213dpi. É principalmente destinado a televisores Sendo necessário prever recursos tvdpi, devem-se dimensioná-los em um fator de 1.33 \* MDPI. |
| Orientação | *Landscape* | Recursos para telas na orientação paisagem (horizontal proporção). |
| *Portrait* | Recursos para telas na orientação retrato (vertical relação de aspecto). |
| Relação de aspecto | *Long* | Recursos para telas que têm uma relação de aspecto significativamente mais alto ou mais largo (quando na orientação retrato ou paisagem, respectivamente) do que a configuração da tela inicial. |
| *Notlong* | Os recursos para telas de uso que possuem uma relação de aspecto que seja semelhante ao ecrã de configuração da linha de base. |

Fonte: Adaptado de http://developer.android.com/guide/practices/screens\_support.html

Figura - Configuração para diferentes densidades de tela

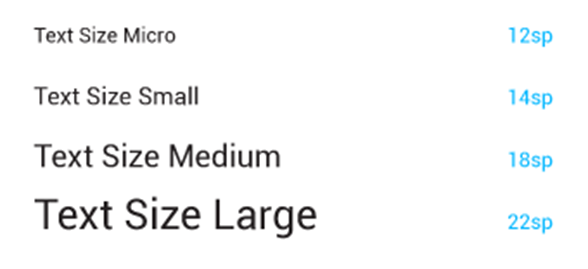


Fonte: http://developer.android.com/guide/practices/screens\_support.html#qualifiers

1. Tipografia

Não menos importante que outros elementos de interface, a tipografia tem por objetivo representar de forma clara, as informações contidas na tela. A linguagem de *design* *Android* conta com recursos tipográficos tradicionais, tais como escala, espaçamento, ritmo e alinhamento. O guia de estilo do portal *Android* developers (2014), além de considerações importantes quanto à padrões de cores existentes no *framework*, apresenta uma discussão no que tange a utilização de textos em tamanhos distintos para se criarem layouts compreensíveis. Neste cenário, recomenda-se pouca variação de escala, bem como a especificação da scale-independent pixels ( sp ), sempre que possível, a fim de apoiar recursos de acessibilidade. A figura 5 mostra o conjunto limitado de tamanhos utilizados na *UI*[[8]](#footnote-8) *Android*.

Figura - Tamanhos de fontes utilizados na UI Android



Fonte: http://developer.android.com/design/style/typography.html

1. Iconografia

Devido à forma de interação com smartphones e *tablets* (entre outras especificidades), os ícones tem um papel importantíssimo no que diz respeito à representatividade de funcionalidades, estado da aplicação e identidade visual (ANDROID DEVELOPERS, 2014). Em dispositivos providos de telas muito pequenas, por exemplo, os ícones têm maior potencial em transmitir uma mensagem ao usuário, em comparação à utilização de blocos textuais, considerando que seria necessário bem menos espaço físico em tela. Por estes e outros fatores, não menosprezar a importância da iconografia e dar-lhe a atenção que merece, é um passo relevante ao sucesso da aplicação.

Pensando em termos de mercado e monetização, é preciso imaginar a aplicação como um produto entre muitos outros, imerso em um ambiente competitivo no qual o ícone de lançamento (*launcher icon*) é quem o representa. Em sua maioria, este ícone terá a responsabilidade de convencer o usuário em realizar o *download* e instalação do aplicativo (CUELLO et al., 2013).

Figura - Ícones de lançamento Android

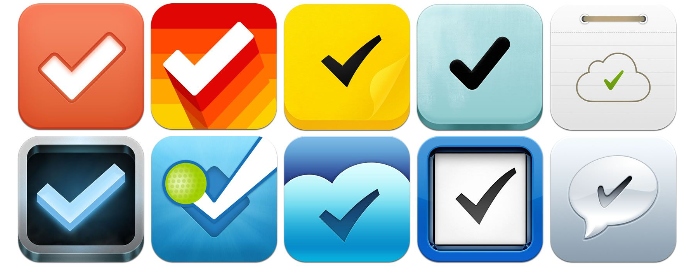


Fonte: http://developer.android.com/design/style/iconography.html

De acordo com Cuello et al. (2013), considerando uma aplicação já instalada no dispositivo e, portanto, em convivência com outras, o ícone passa a ter caráter distintivo e representativo. Distintivo porque deve distinguí-la das demais (incluindo aquelas que possuem funções similares), e representativo, pois sua linguagem visual deve transparecer o objetivo principal da aplicação. O autor aponta como de maior efetividade, o design baseado em formas simples, contendo pouca carga de informação visual e com detalhes bem trabalhados.

Um ícone de lançamento de um app deve ter personalidade para distinguí-la das demais, além de representar o que faz, de forma clara. Na Figura 7, o segundo ícone (fila inferior) representa uma rede social muito famosa e pode ser facilmente reconhecido (CUELLO et al., 2013)

Figura - Ícones e Identidade visual



**Fonte: Cuello et al.**

Além de suporte a diferentes densidades de tela, como citado na seção 2.1.1, outras considerações sobre o lançador devem ser abordadas (ANDROID DEVELOPERS, 2014):

* Visibilidade - O ícone de lançamento deve ser claramente visível em qualquer tipo de fundo;
* Tamanhos e escala - em um dispositivo móvel, ícones de lançamento devem ter **48x48 dp** e para visualização no *Google Play*, **512x512 *pixels*** ;
* Proporções – Caso o ícone seja envolto por uma plano de fundo, valem as métricas do item anterior;
* Estilo – sugere-se utilizar uma forma distintiva, com aspecto tridimensional, vista frontal e ligeira perspectiva (como se visto por cima), a fim de produzir sensação de profundidade.

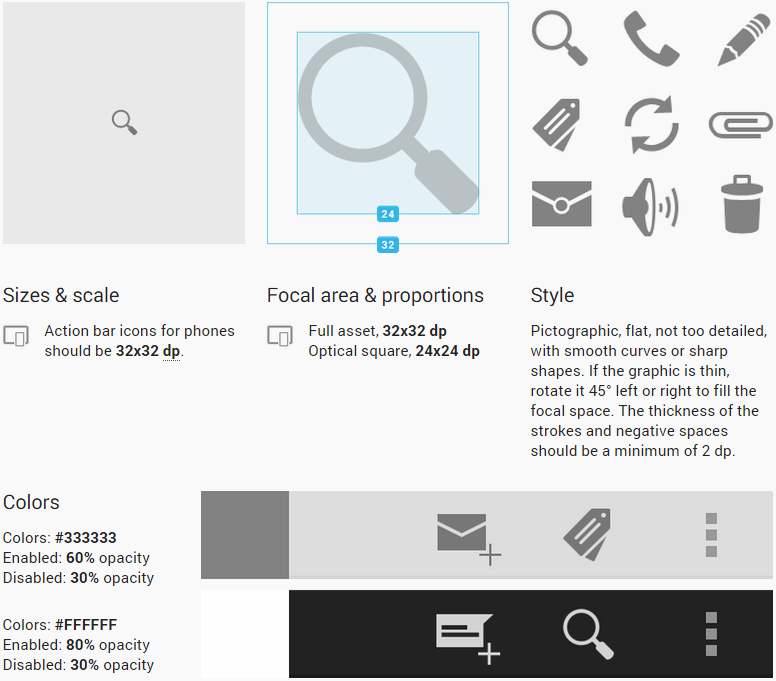
Ícones que fazem parte do corpo da aplicação tem como papel principal representar as ações que os usuários podem tomar dentro do *app*. Segundo Cuello et al. (2013), a utilização desta modalidade de iconografia está associada a três cenários. No primeiro, como auxílio visual, a fim de enfatizar a informação (como em um quadro de diálogo com um alerta, por exemplo). No segundo, os ícones podem servir como complemento de elementos interativos, quando encontrados em listas, botões, *tabs*, etc. Por último, podem ser utilizados como otimizadores do espaço ocupado na tela, pois resumem visualmente uma informação que em forma de texto seria muito extensa ou de difícil compreensão.

No portal Android Developers (2014), são feitas várias considerações sobre esta modalidade. Estes ícones são agrupados em três categorias: integrantes da *Action Bar*, pequenos / contextuais e de notificações.

Os ícones da *Action Bar*, como o próprio nome sugere, representam as ações mais importantes a serem tomadas pelos usuários. Cada um destes deve representar uma metáfora de fácil associação com seu objetivo de forma que não produza interpretações ambíguas.

Os ícones pequenos / contextuais devem sem utilizados no corpo da aplicação com o objetivo de evidenciar ações ou status de itens específicos, diferentemente dos ícones da *action bar*, que possuem visão e controle da aplicação de forma mais globalizada.

Por fim, existem os chamados ícones de notificação, a serem mostrados na barra de status do *Android*, de forma a alertar o usuário sempre que novas notificações estiverem disponíveis.



Fonte: http://developer.android.com/design/style/iconography.html

Figura 8 - Guia ilustrado para design de ícones

### Padrões de design Android (design patterns)

Nesta seção serão tratados alguns dos principais *design patterns* Android a serem utilizados no projeto.

1. Gestos

Utilizando seus computadores no dia a dia, usuários tornaram-se experts em manipular teclado e mouse. Gestos como duplo clique, clicar e segurar, acessar menus ao clique do botão direito, clicar e arrastar para seleção de objetos , entre outros, são triviais e até então suficientes à eficaz utilização de desktops. Usuários avançados aprenderam a usar atalhos para operar sem ter que mover suas mãos do teclado. Entretanto, segundo Lehtimaki (2013), um novo paradigma de controle primário tomou a indústria de smartphones como uma tempestade. Isto porque, afirma o autor, a maior parte de smartphones já possuem displays sensíveis ao toque. Diz ainda que esta modalidade de interação é, por vezes, tida como natural ao usuário.

Quadro - Gestos do Android

|  |  |  |
| --- | --- | --- |
| Nome do Gesto | Descrição | Ícone |
| *Tap* (bater de leve) | Usuário simplesmente toca uma região da tela. |  |
| *Double Tap* | Usuário toca rapidamente a tela duas vezes no mesmo local. |  |
| *Swipe* (também  Conhecido por *Fling*, *flick*) | Usuário arrasta o dedo pela tela rapidamente em uma direção. |  |
| *Drag* (arrasto) | Usuário arrasta o dedo na tela. O movimento é mais lento do que o Swipe e não tem de ser em uma direção específica. |  |
| Beliscar, fechar, pinça | Usuário arrasta dois dedos na tela em um movimento de beliscar. |  |
| *Spread* (propagar, expandir) | Usuário arrasta dois dedos na tela distanciando-os entre si |  |
| *Press* | Usuário pressiona a tela por um tempo de aproximadamente 2 segundos ou mais. |  |

Fonte: Lehtimaki

1. Estrutura da aplicação

A estrutura de um aplicativo depende muito do conteúdo e tarefas definidos para atender as necessidades dos seus usuários. Para tanto, existem padrões de design definidos no portal Android Developers (2014), afim de guiar o *designer* na definição e organização do conteúdo, implementação de ações e navegação consistentes e de fácil utilização.

A estrutura geral de uma típica aplicação Android consiste em camadas de nível superior e visualizações de detalhes ou edição.

1. Navegação

A navegação consistente é um componente essencial à experiência geral do usuário. Poucas coisas causam mais frustrações aos usuários do que comportamento inconsistente e inesperado da navegação. De uma forma mais básica, podem-se utilizar os botões *Up* (navegação sentido camadas filho-pai dentro do app) e *Back* (retorna à camada interior na pilha do sistema).

1. Navigation Drawer

O *Navigation Drawer* (gaveta de navegação) é um painel/menu que faz a transição a partir da borda esquerda da tela e exibe principais opções de navegação do aplicativo. A estrutura do seu aplicativo deve orientar a escolha de qual padrão a ser usado. Alguns exemplos de boas práticas de uso deste recuso são (ANDROID DEVELOPERS 2014):

* Mais de 3 camadas de nível superior - O *Navigation Drawer* é excelente para exibir um grande número de alvos de navegação ao mesmo tempo. Recomenda-se seu uso se o app tiver mais de 3 camadas únicas de nível superior. Caso contrário, a utilização de abas fixas é mais recomendada facilitar a descoberta e interação;
* Navegação cruzada a partir de níveis mais baixos - Neste contexto, por ser acessível a partir de qualquer lugar do app, a gaveta permite a navegação eficiente a partir de telas de nível mais baixo para outros lugares importantes do app;
* Ramos de navegação profundas - O *Navigation Drawer* permite saltos rápidos para níveis superiores do aplicativo, removendo a necessidade de repetitivo uso do botão *Back* ou *UP;*

1. Layouts Multi – painel

Ao escrever um aplicativo para *Android*, deve-se ter em mente os diferentes tamanhos de tela e tipos e fornecer um layout consistentemente equilibrado e esteticamente agradável, ajustando o seu conteúdo para diferentes tamanhos de tela e orientações. Entre as técnicas utilizadas é possível construir layouts com dimensões e escalas e combinações de painéis adaptáveis.

### Definindo uma proposta

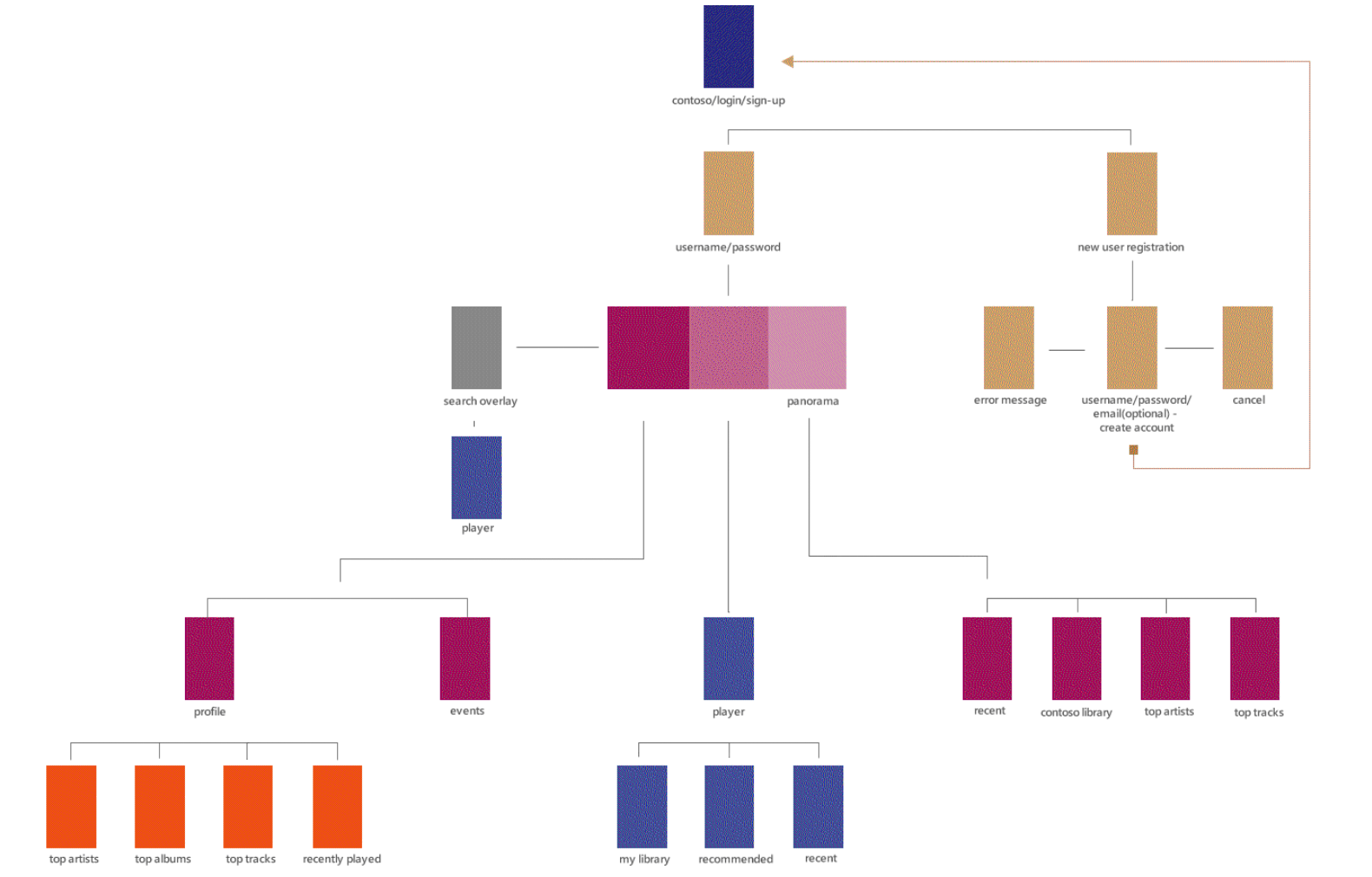
Após conhecer os guias de estilo, padrões de design e outras peculiaridades do universo *Android* referenciados na seção anterior, o designer passa a ter um novo desafio: aplicar estes conceitos de forma organizada e coerente com um projeto proposto. Antes de começar a projetar, faz-se necessário entender e estudar sobre cenários, escopo de utilização, requisitos funcionais, etc., bem como, o que os usuários precisam e como o projeto responderá às suas necessidades. (CUELLO et al., 2013).

Partindo do pressuposto que o designer conheça o papel e necessidades do usuário, bem como os requisitos funcionais da aplicação a ser projetada, a etapa seguinte é propor uma visão da aplicação.

1. Arquitetura da informação

A arquitetura da informação é uma maneira de organizar o conteúdo e funcionalidades de toda a aplicação afim de proporcionar fácil acesso por parte do usuário. Trata-se, em sentido global, de estabelecer uma relação entre os conteúdos de diferentes telas e a nível de uma tela em particular, a organização de seus componentes. (CUELLO et al., 2013).

Figura - Diagrama arquitetura da informação



Fonte: Cuello et el., 2013

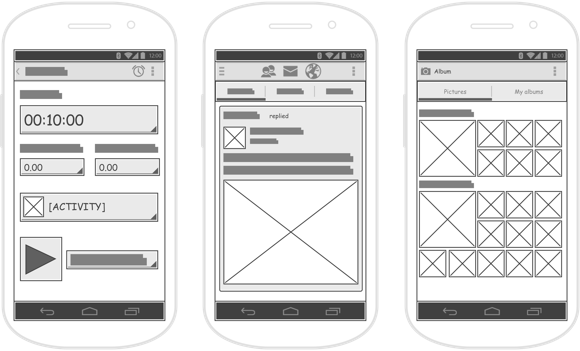
1. Wireframes

Definido por Cuello et al. (2013), um *wireframe* ( quadro de arame, em tradução livre) é uma forma simplificada de representar uma tela individual, permitindo aos designers e usuários terem uma ideia preliminar da organização dos elementos que a constitui, identificando e possibilitando classificá-los em informativos ou interativos.

Fazendo uma analogia, para uma aplicação, o *wireframe* é como o plano arquitetônico para uma casa, afirma o autor. Desta forma, apresenta de uma forma clara e simplificada os espaços e elementos funcionais que a constitui.

Assim como em um plano arquitetônico de uma casa, *wireframes* são desenhados de forma linear e numa mesma cor, enfatizando o design estrutural. Evitando detalhes estéticos, como texturas, sombras e volumes, esta técnica de design permite elaboração e alterações mais rápidas em comparação a protótipos de alta fidelidade.

Figura - Wireframes



Fonte: https://www.visual-paradigm.com/tutorials/android-wireframe.jsp

1. Protótipos e métodos de prototipagem

Por definição, a prototipagem é o processo de criação de uma amostra antes de se produzir uma versão final. No design de software, prototipagem é usada para criar uma versão do software que podemos testar e aprender. O objetivo final é "resolver as torções" antes que seja feita uma versão final. Prototipagem móvel é crucial para obter experiências de usuário. Idealmente, utiliza-se prototipagem móvel para se ter *feedback* sobre três aspectos (MENDONZA, 2014):

* Fluxo das telas - No que diz respeito à navegação, o fluxo e relação entre as páginas e outras telas da aplicação devem fazer sentido para o usuário. A utilização do protótipo sob este aspecto tende a responder ao designer se a proposta atende às expectativas de experiência do usuário ou haverá necessidade em alterar alguma forma de interação;
* Interface de interação do usuário - Os gestos de toque são claros? Quando o usuário aciona o teclado ou outro componente, a lógica de funcionamento faz sentido? Estão sendo utilizados os elementos de *UI* que o usuário espera ver? Com este *feedback* é possível focar-se na otimização dos mecanismos de uma tela específica, tarefa, ação ou função;
* Interação de dispositivos - Segundo o autor, a utilização de protótipos neste contexto serviriam de auxílio à obtenção métricas relacionadas à compatibilidade do estado esperado da aplicação proposta quando utilizando os elementos de hardware do dispositivo real. Este *feedback* ajudaria a otimizar a escala, proporção, e interação da experiência com o dispositivo, por exemplo.

Vale ressaltar que apesar de parecidos, *wireframes* e protótipos têm características e objetivos distintos. *Wireframe* é de caráter mais abstrato e por ser de menor fidelidade ao produto final, é uma técnica com maior eficácia quando necessário representar as estrutura da aplicação. Em contrapartida, protótipos são artefatos de média a alta fidelidade e seu uso é mais adequado em etapas mais próximas da produção final.

Existem diversas formas de se produzirem protótipos: podem-se fazê-los com desenhos em papel, utilizando softwares tradicionais de desenho ou até mesmo aplicações dedicadas a sua construção. Em Cuello et al.( 2013) as formas de prototipagem são classificadas como:

* Documentos navegáveis - Os protótipos podem apresentar-se em forma de documentos de uso popular como PDF. Usam-se também arquivos de apresentação como os documentos gerados pelo *Microsoft PowerPoint, Apple KeyNote e OpenOffice*, etc.;
* Versões web - Representações da aplicação em formato de sítios web e baseados em *HTML5*[[9]](#footnote-9)e *CSS3*[[10]](#footnote-10) (necessitando conexão com a internet e uma navegador web). Possíveis de serem criados por muitos softwares dedicados (*Desktop* ou na nuvem), estes protótipos permitem a utilização de interações e animações que muito se assemelham às do produto final. *Codiqa*[[11]](#footnote-11), *FluidUI*[[12]](#footnote-12), *Framer*[[13]](#footnote-13) e *Flinto*[[14]](#footnote-14) são alguns exemplos dessas ferramentas;
* Formatos alternativos - Existem softwares com propostas diferentes, porém não menos interessantes. Um deles é aplicação *Briefs*[[15]](#footnote-15), disponível para Mac e *iOS*. Com este software é possível produzir protótipos profissionais e de alta fidelidade, possibilitando testes em tempo real, enquanto trabalha (basta instalar um aplicativo no smartphone). Caso o designer deseje criar protótipos de forma mais rápida e utilizando o próprio dispositivo, uma ótima opção é o *POP*[[16]](#footnote-16), que permite utilizar a câmera do aparelho para fotografar *wireframes* em papel e dotá-los de interação para navegar entre as diferentes telas.

Figura - Ferramenta de prototipagem Briefs



Fonte: http://giveabrief.com/

# SISTEMAS DE GESTÃO DA QUALIDADE DE UTI’S

Esta seção referencia os aspectos sobre estudo e aplicação dos índices de prognósticos para apoio à melhoria da qualidade em UTIs, além de conceitos sobre os scores SAPS 3 e APACHE II.

Balsanelli et al. (2006), define as Unidades de Terapia Intensiva (UTIs) como sendo setores críticos do hospital destinados aos pacientes graves que carecem de acompanhamento contínuo e suporte terapêutico especializado. Neste cenário, o enfoque interdisciplinar torna-se indispensável para que o paciente tenha suas necessidades de tratamento à saúde atendidas da melhor forma e tão rápido quanto possível.

Para Balsanelli et al. (2006), considerando o elevado número de procedimentos terapêuticos para acompanhamento de pacientes graves, os custos hospitalares chamam a atenção. Dessa forma, os índices de prognósticos utilizados para apoiar à triagem dos pacientes de UTI, baseando na gravidade e probabilidade de morte,tornaram-se instrumentos de medida que permitem uma forma eficaz de avaliação dos resultados e investimentos.

De acordo com Silva Junior et al. (2010, p.1) “Os índices prognósticos quantificam desarranjos fisiológicos agudos e crônicos durante a admissão, estimando mortalidade, com objetivo de corrigir os erros e melhorar o desempenho da unidade de terapia intensiva.”

Terzi et al. (2002, p.1),resume a importância da aplicação dos índices de prognósticos em UTIs como ferramenta de auxílio à triagem de pacientes graves, mensuração do desempenho das unidades e à avaliação de novos procedimentos e tecnologias.

Os objetivos propostos para o emprego dos índices prognósticos em pacientes graves podem ser resumidos em quatro grandes áreas de interesse para o intensivista:

1. Permitem aos médicos focalizarem sua atenção àqueles pacientes que podem mais se beneficiar do tratamento intensivo.
2. Permitem complementar o juízo clínico na limitação ou suspensão do suporte avançado de vida.
3. Permitem a comparação de desempenho entre diferentes unidades.
4. Permitem estratificar grupos de pacientes para a avaliação de novas tecnologias e procedimentos terapêuticos.

Foram determinados pela ANVISA[[17]](#footnote-17), e em vigor desde 24 de fevereiro de 2010, os requisitos mínimos para funcionamento das unidades de terapia intensiva, publicados na RDC[[18]](#footnote-18) nº 7. Segundo informações contidas no portal da agência, a resolução se aplica a todas as Unidades de Terapia Intensiva do país, sejam públicas, privadas ou filantrópicas; civis ou militares. Informa ainda que, o objetivo é reduzir os riscos aos pacientes, visitantes, profissionais e meio ambiente, além de buscar elevar a qualidade do atendimento, com a consequente redução do tempo de tratamento de pacientes graves nesses setores. Assim, mais pacientes poderão usufruir do tratamento especializado oferecido nas unidades.

Dentre os requisitos dispostos na RDC7, o art. 48 descreve a obrigatoriedade de serem monitorados e mantidos registros de avaliações do desempenho e do padrão de funcionamento global da UTI, assim como de eventos que possam indicar necessidade de melhoria da qualidade da assistência, com o objetivo de estabelecer medidas de controle ou redução dos mesmos. Entre outras, as seguintes regras são citadas:

* Deve ser calculado o Índice de Gravidade / Índice Prognóstico dos pacientes internados na UTI por meio de um Sistema de Classificação de Severidade de Doença recomendado por literatura científica especializada;
* O Responsável Técnico da UTI deve correlacionar a mortalidade geral de sua unidade com a mortalidade geral esperada, de acordo com o Índice de gravidade utilizado.
* Os pacientes internados na UTI devem ser avaliados por meio de um Sistema de Classificação de Necessidades de Cuidados de Enfermagem recomendado por literatura científica especializada.

Como requisito ao sistema ICU Surviving, proposto neste trabalho e visando atender às determinações supracitadas, serão utilizados os índices de prognóstico SAPS 3 e APACHE II.

## SAPS 3

O SAPS 3 é um sistema de pontuação, cuja aplicação produz informações para cálculo da predição de mortalidade de pacientes internados em unidades de terapia intensiva. Este índice de prognóstico inclui dados de 307 UTIs de 35 países. Em média, cada UTI contribuiu com o estudo de 50 pacientes. Para avaliar a heterogeneidade de resultados entre diferentes regiões geográficas, foram definidas sete áreas: Austrália, América Central e América do Sul, Europa central e ocidental, Leste Europeu, América do Norte, Norte e Sul da Europa e países do Mediterrâneo (METNITZ et al., 2005).

Segundo Silva Junior et al. (2010), o Escore Fisiológico Agudo Simplificado (SAPS 3) possui 20 variáveis de fácil mensuração na admissão do paciente na UTI. Estas variáveis são divididas em três grupos: dados demográficos, razões pela admissão na UTI e dados fisiológicos. Para cada informação coletada é atribuído um peso, de acordo com gravidade do distúrbio fisiológico. O somatório do valores determina a pontuação do escore, que, na teoria, pode variar entre 16 (menor valor admitido) e 217 pontos. Uma descrição destas variáveis e seus respectivos valores e pesos podem ser consultados no ANEXO 1.

Posteriormente, a relação entre o SAPS 3 e status vital no momento da alta hospitalar é dada pela equação geral (MORENO et al., 2005):

Logit = -32.6659 + ln (SAPS 3 score +20.5958) x 7.3068

Por sua vez, a probabilidade de morte pode ser adquirida pela fórmula:

Probabilidade de morte = elogit/(1+elogit).

Tabela - Equações de admissão SAPS 3 customizadas para diferentes regiões geográficas

|  |  |
| --- | --- |
| Área | Equação |
| Austrália | Logit = -22.5717 + ln (SAPS 3 score + 1) x 5.3163 |
| Américas Central e do Sul | Logit = -64.5990 + ln (SAPS 3 score + 71.0599) x 13.2322 |
| Europa Central e ocidental | Logit = -36.0877 + ln (SAPS 3 score + 22.2655) x 7.9867 |
| Europa Oriental | Logit = -60.1771 + ln (SAPS 3 score + 51.4043) x 12.6847 |
| Norte da Europa | Logit = -26.9065 + ln (SAPS 3 score + 5.5077) x 6.2746 |
| Sul da Europa, países do Mediterrâneo | Logit = -23.8501 + ln (SAPS 3 score + 5.5708) x 5.5709 |
| América do Norte | Logit = -18.8839 + ln (SAPS 3 score + 1) x 4.3979 |  |
|  |  |

Fonte: Adaptado de Moreno et al., 2005

## APACHE II

De acordo com Gonçalves et al. (1999), em 1981, foi introduzido por Knaus et al., o primeiro índice prognóstico de avalição de gravidade da doença o Acute Phisiology and Chronic Health Evaluation (APACHE), cuja base para o seu desenvolvimento firmou-se na hipótese de que a gravidade da doença aguda poderia ser mensurada através da quantificação do grau de anormalidades das múltiplas variáveis fisiológicas.

Devido à complexidade do APACHE original, houve a necessidade de um modelo mais simplificado. Para tanto, fruto de esforços continuados, surgiu o sistema APACHE II, uma versão modificada e aprimorada do original, desenvolvida pelo mesmo grupo de estudo, baseada num banco de dados com mais de 5 mil pacientes. Esta nova versão, mais acurada quanto ao poder preditivo e de uso clínico mais simples, foi apresentada a comunidade científica em 1985 (GONÇALVES et al., 1999).

Juntamente com o SAPS 3, o APACHE II foi escolhido como um dos recursos a serem implementados no aplicativo proposto neste projeto.

De acordo com Knaus et al (1985), este modelo de prognóstico é composto por 12 variáveis fisiológicas, idade e estado prévio de saúde utilizados de forma a fornecerem uma medida geral de gravidade da doença. Uma pontuação crescente (faixa de 0 a 71) foi estreitamente relacionada com o risco de óbito hospitalar para 5815 admissões de terapia intensiva de 13 hospitais.

Ainda segundo Knaus et al (1985), o método de aplicação do APACHE II dar-se pelo somatório dos pontos coletados com base em dados fisiológicos, demográficos e estado prévio de saúde do paciente. São utilizadas as variáveis mais alteradas nas primeiras 24 horas, contadas à partir da internação na UTI. Para coleta dos dados utiliza-se um formulário padrão

A equação final para cálculo do índice APACHE II e expressa:

In (R/1-R) = -3,517 + ( ESCORE APACHE II x 0,146)

+ (0,603 somente se cirurgia de emergência) + (Diagnóstico específico)

O ANEXO 2 apresenta o modelo de formulário padrão utilizado na coleta dos valores para o APACHE II.

# ANÁLISE CONTEXTUAL E ELABORAÇÃO DOS PROTÓTIPOS

Os tópicos a seguir descrevem as questões pertinentes ao desenvolvimento e funcionamento da aplicação a ser construída, tais como objetivo da aplicação, cenários e dinâmica de uso, recursos, pré-requisitos de software, hardware, etc.

## Escopo, conceito e planejamento

1. Visão geral do sistema ICU SURVIVING

A aplicação proposta tem por objetivo, servir como ferramenta de apoio à gestão da qualidade assistencial em unidades de terapia intensiva. Seu papel principal é calcular as probabilidades de óbito de cada paciente cadastrado utilizando o SAPS 3 e APACHE II. Estes escores e outras informações relacionadas, deverão ser armazenados na base de dados do sistema, podendo ser consultados em forma de relatórios a serem requisitados pelo gestor da unidade.

O sistema deverá ser construído para a plataforma *Android*, especificamente para *tablets*. Desta forma, pensa-se em produzir uma aplicação de baixo custo, que ofereça mobilidade e fácil manuseio, sem abrir mão da qualidade de recursos e segurança da informação.

1. Procedimentos operacionais padrão

Em relação a dinâmica de uso do sistema proposto, devem-se ter em mente as seguintes considerações:

* Quando na admissão do paciente, o profissional de saúde responsável iniciará a obtenção de informações do mesmo, utilizando-se de conversas com familiares, amigos, etc.;
* Caso haja dificuldade, ou mesmo, impossibilidade em adquirir essas informações, faz-se necessária a utilização de dados estimados;
* São realizados exames clínicos e solicitados exames laboratoriais;
* A obtenção dos resultados de exames laboratoriais depende da logística do hospital;

1. Requisitos funcionais (descrição das funções que o software deve executar):

* O software deve permitir o cadastro de pacientes/internação;
* O software deve permitir o pesquisa de pacientes/internações cadastrados;
* O software deve mostrar um resumo estatístico da UTI (Módulo WEB);
* O software deve permitir a geração de relatórios de desempenho das UTI’s (Módulo WEB);

1. Não-funcionais (condições que o software deve atender ou qualidades específicas que o software deve ter)

* O software será destinado somente a tablets;
* A autenticação do usuário será realizada somente mediante aprovação do administrador do sistema. Ao executar a aplicação pela 1ª vez, uma requisição será enviada ao servidor que por sua vez, irá capturar informações do hardware do tablet;
* A arquitetura do sistema será no modelo CLIENTE-SERVIDOR, sendo indispensável conexão com a internet;

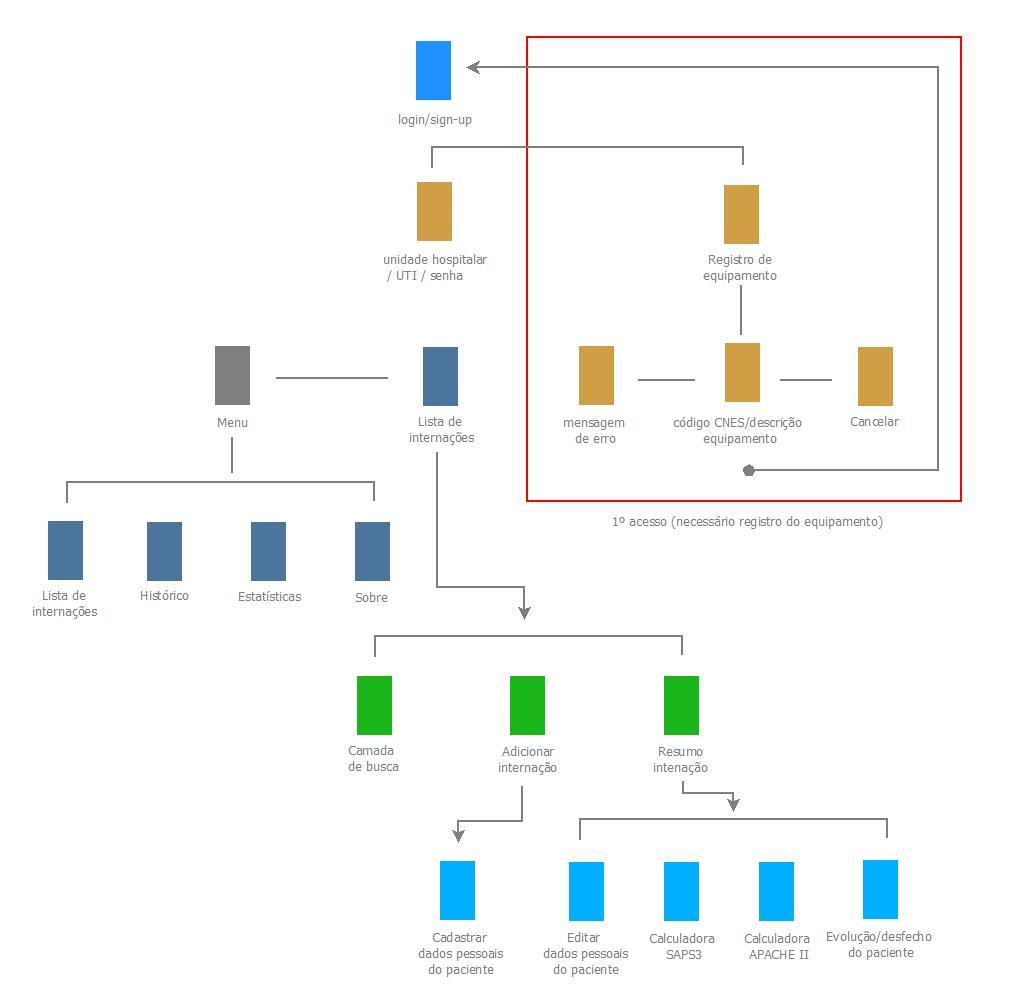
1. Requisitos de software

* Deverão ser utilizados os índices de prognóstico SAPS III e APACHE II

1. Arquitetura da informação

Levantados os requisitos do sistema, é possível propor uma visão prévia e resumida da aplicação a ser desenvolvida permitindo melhor análise e compreensão de sua complexidade . A Figura 12 apresenta como o conteúdo e funcionalidades do aplicativo ICU Surviving poderão ser organizados. Notam-se facilmente os diferentes níveis de profundidade e as relações entre os conteúdos. Cada retângulo representa uma porção da interface, podendo ser uma funcionalidade ou uma tela propriamente dita. A hierarquia deve ser estabelecida de forma que as camadas superiores são responsáveis em fornecer acesso às camadas inferiores.

Figura - Arquitetura da informação do sistema ICU Surviving



Fonte: O autor

1. Wireframes

Como tratado na seção 2.2.3, a utilização de *wireframes* é uma forma pratica e barata de apresentar uma visão do conteúdo e funcionalidades de telas que constituem uma aplicação.

Para o estudo de caso deste projeto, visando-se a criação rápida dos protótipos, foram utilizados *wireframes* desenhados a mão e a ferramenta POP para teste de usabilidade. Após definidos a organização e tipos de elementos de interface, a proposta passou a ter como foco na estética e experiência do usuário.

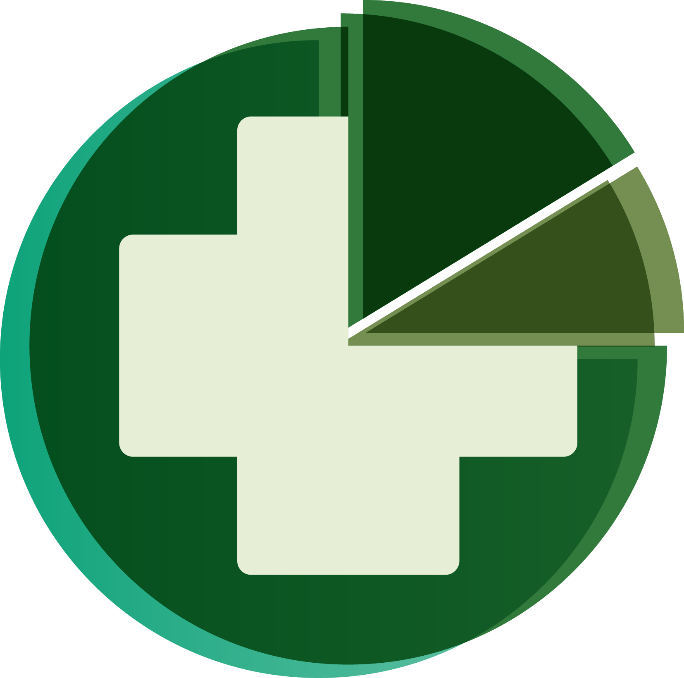
1. Iconografia

A criação dos ícones para o sistema ICU Surviving foi baseada no conceito de simplicidade, utilizando elementos do flat design em harmonia com a proposta da aplicação. O foco desta etapa foi produzir alto grau de experiência do usuário, apoiando-se em princípios como *affordance*e guias de estilo para design de aplicações *Android*.

1. Ícone lançador

Visto que o objetivo principal do sistema é produzir dados estatísticos referentes às taxas de mortalidade real e estimada das UTIs, pensou-se na identidade visual do ícone principal de forma que o usuário intuitivamente o associasse à área de assistência hospitalar (símbolo da cruz e predominância de tons em verde), bem como à uma aplicação que manipulasse dados estatísticos (formato de gráfico de pizza).

Figura 13 - Ícone lançador do ICU Surviving

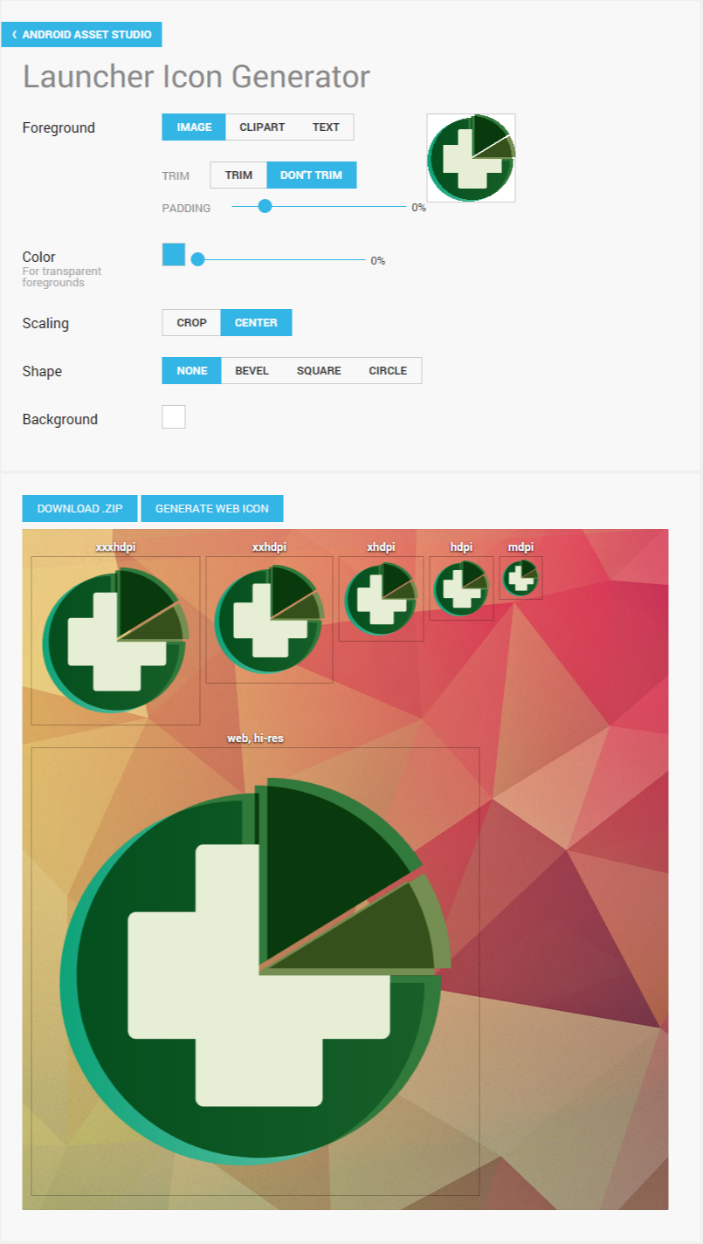


Fonte: O autor

Outro aspecto importante sobre a identidade visual do ícone lançador do ICU Surviving diz respeito ao seu formato circular, que, segundo os guias de estilo para aplicações *Android* (seção 2.2.1), reforça o caráter distintivo da marca quando disposta entre ícones de outros aplicativos desenvolvidos para o SO da Google, seja quando instalado no dispositivo ou se encontrado em lojas de *apps*, como *Google Play Store* ou *Amazon Appstore*, por exemplo.

Pensando em garantir suporte a diferentes densidades de tela, utilizou-se a ferramenta *Launcher Icon Generator* ( *Android* Asset *Studio*) para se criar réplicas com dimensões especificas para cada densidade de tela (mdpi, hdpi, xhdpi, xxhdpi, xxxhdpi e web, hi-res).

Figura - Laucher icon do ICU Surviving em várias escalas



Fonte: O autor

1. Ícones internos

Nesta etapa os ícones foram criados com a finalidade de servirem como complemento de elementos interativos (listas, botões, etc.) e complementar mensagens da barra de notificações. Para representar as ações da *Action* *bar* e outras informações contextuais, pensou-se em utilizar o pacote de ícones baseados no tema *holo*-*light*, disponibilizado para *download* na seção *design* do portal *Android Developers*. Este pacote inclui ícones escalados para diversas densidades de tela. Inclui ainda arquivos editáveis de ícones que podem ser modificados para combinar com o tema da aplicação, caso necessário.

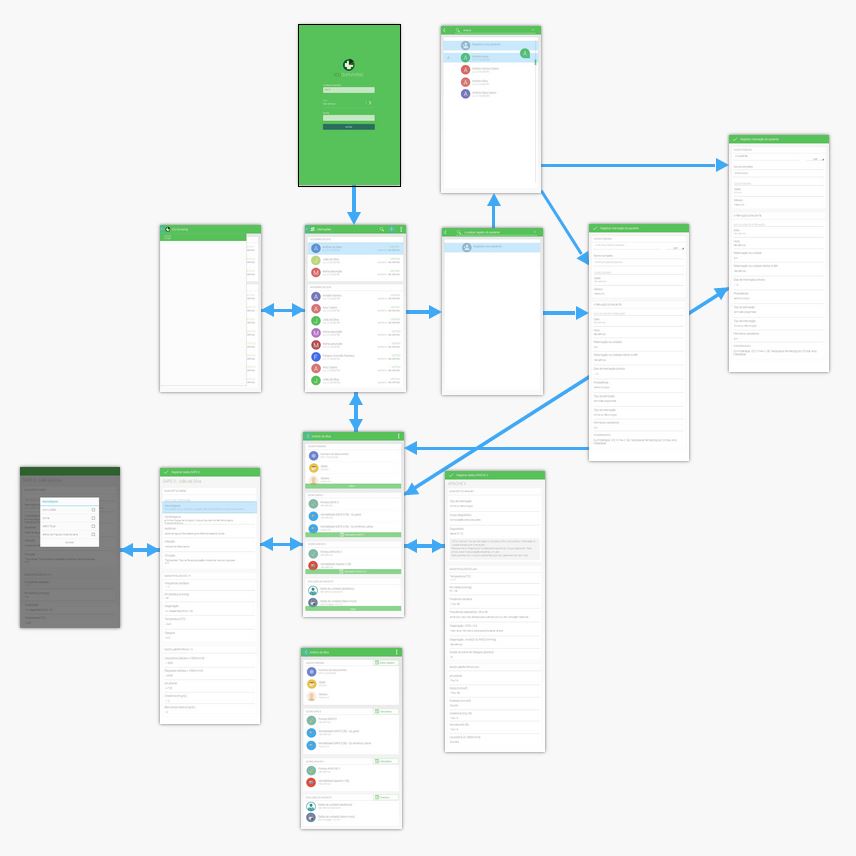
## Construção dos protótipos

Para construção dos protótipo de alta fidelidade foram utilizados como referência o diagrama de arquitetura de informação, além da estrutura e os componentes definidos nos *wireframes*. O objetivo nesta etapa foi definir detalhes estéticos e validar as funcionalidades e interações segundo os guias de estilo e padrões de projeto do *Google developers*.

Em todo o processo de criação das telas utilizou-se o sistema *FluidUI*, que se mostrou adequado à essa função. Trata-se de uma aplicação web dedicada a prototipagem móvel para *Android*, *iPhone*, *iPad* e *Windows* *8*. Como mencionado na seção 2.1.2, e de fácil aprendizado.

A Figura 15 mostra uma visão geral do resultado final de design do protótipo no qual é possível visualizar a relação entre as telas e estruturas dos elementos que as constituem. Pode-se observar também detalhes estéticos como dimensões, cores, sombreamento, tipografia, entre outros.

Figura - Visão geral dos protótipos



Fonte: O autor

Os tópicos a seguir apresentam discursões referentes a metodologia utilizada no projeto de criação dos protótipos.

1. Tela de login

interface responsável pela autenticação do usuário. Possui os seguintes componentes:

* Unidade hospitalar (automaticamente preenchido pela aplicação);
* UTI (definição da unidade a ser acessada);
* Senha (chave para acesso);
* Entrar (botão para submeter os dados)

Figura 16 - ICU Surviving: Tela de login

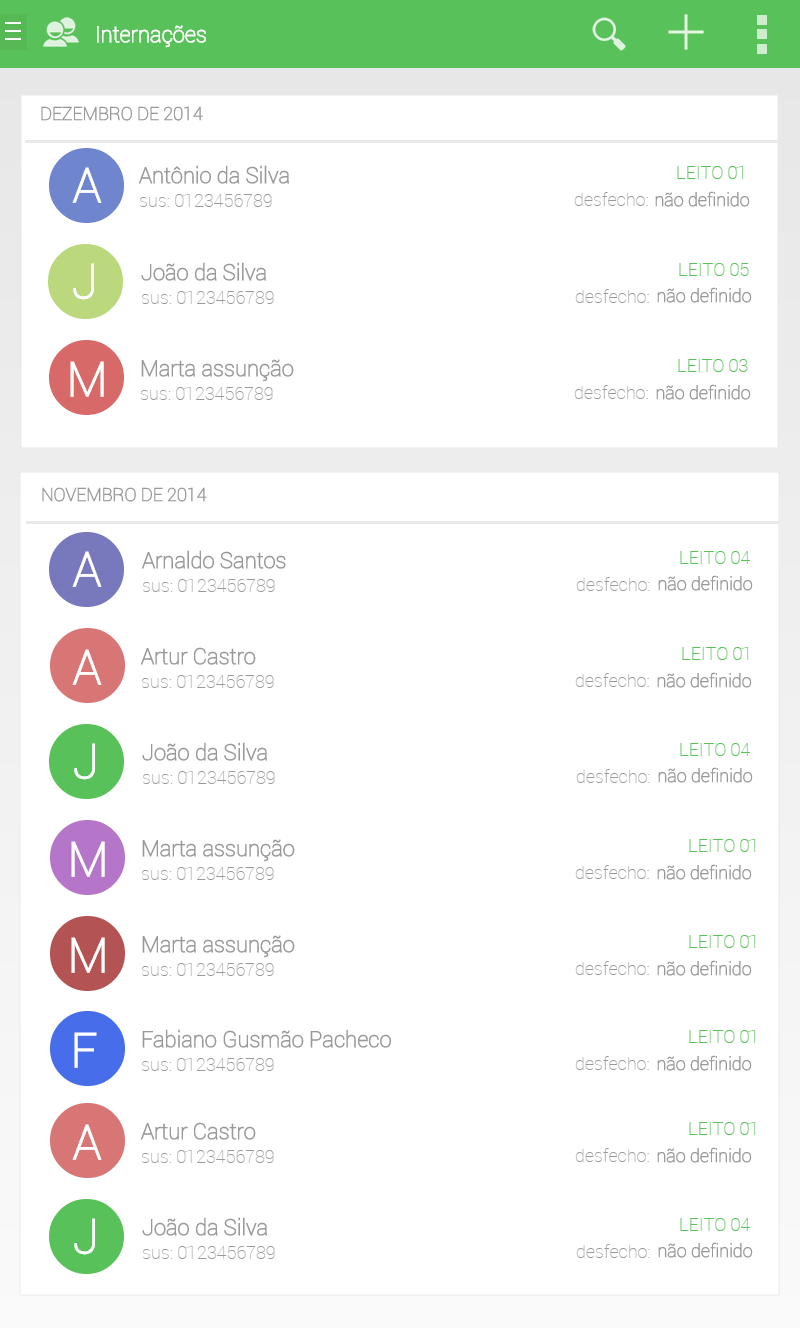


Fonte: O autor

1. Internações

Nesta proposta, é considerada a tela principal da aplicação. Apresenta a lista de internações agrupadas por período. Segue o padrão de design *card view*, no qual as blocos que representam porções da interface possui a aparência de um cartão. Na *action bar* o usuário encontrará links de acesso ao menu principal (*Drawer menu*), à pesquisa de internações (*search button*) e o botão “+” para uma nova internação.

Figura 17 – ICU Surviving: Lista de internações

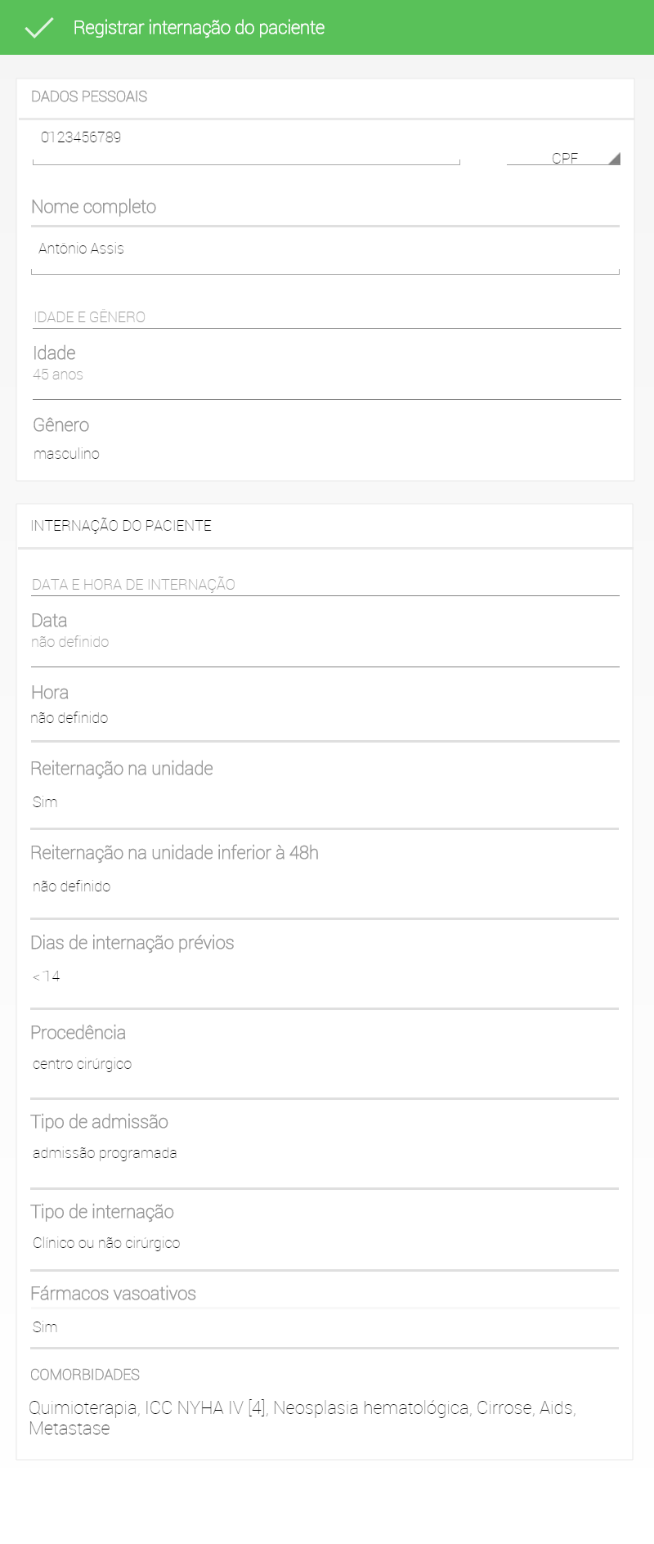


Fonte: O autor

1. Registrar internação do paciente

Dispõe de um formulário dividido em dois blocos, solicitando do usuário informações sobre os dados pessoais e internação do paciente, respectivamente. É composta em sua maioria por elementos de lista. Este arranjo tende a facilitar a entrada dos dados, pois basta que o usuário selecione um ou mais itens pré-definidos ao invés de digitá-los.

Figura 18 – ICU Surviving: Registrar internação do paciente

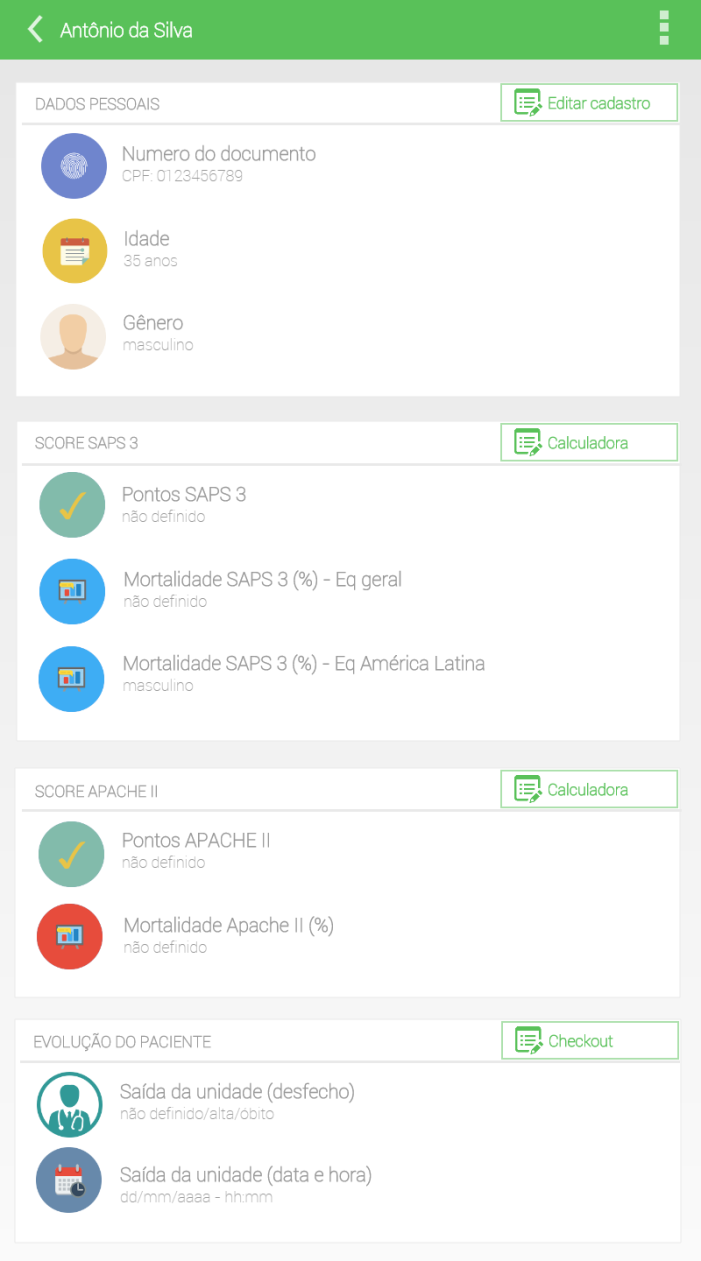


Fonte: O autor

1. Detalhe internação

Após concluir o cadastro da internação, o usuário será enviado à uma nova tela contendo o resumo do internação dividido em quatro blocos: dados pessoais, score SAPS 3, Score APACHE II e evolução do paciente. Na Figura 19 é possível visualizar o elemento “*up* *navigation*” da *action bar* além de outros botões responsáveis por possibilitar a modificação dos dados.

Figura 19 - ICU Surviving: Detalhe internação



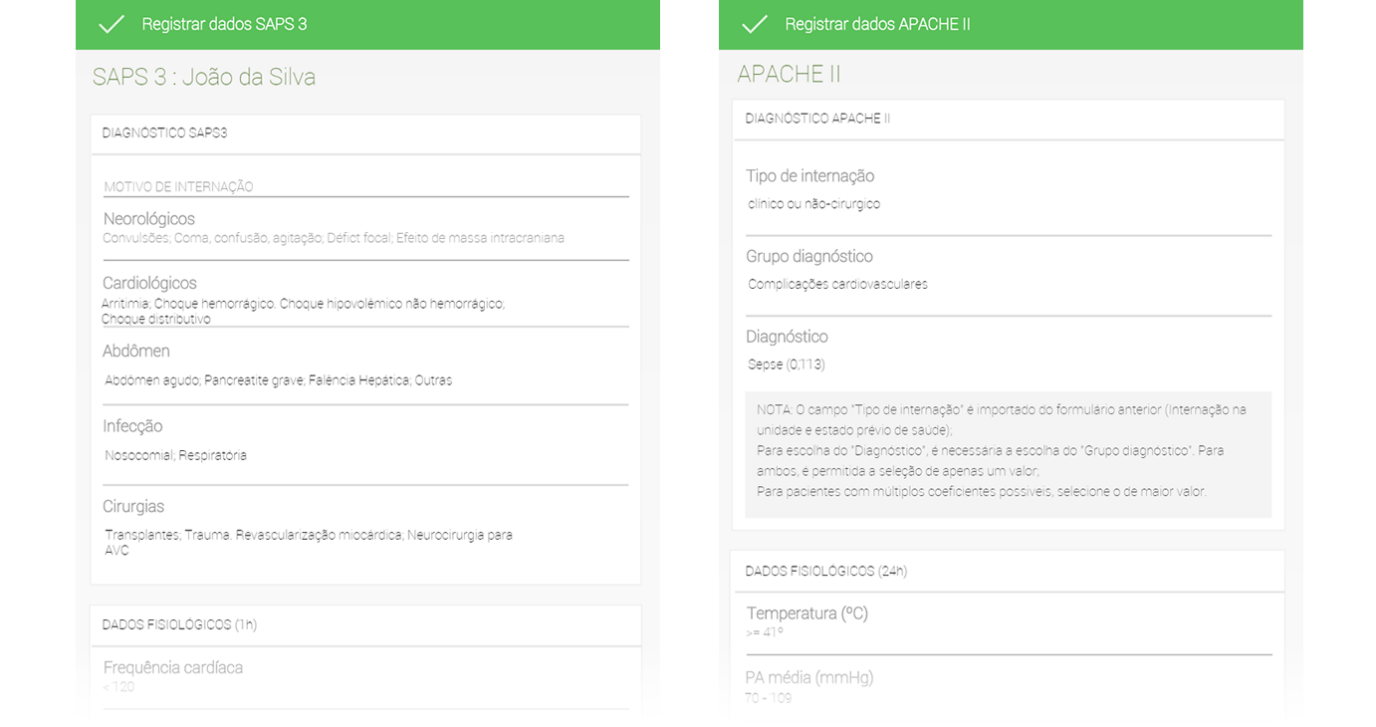
Fonte: O autor

1. Registrar dados SAPS 3, APACHE II e Evolução do paciente

Em posse dos parâmetros necessários ao cálculo do SAPS 3 e APACHE II, o usuário poderá acessar os formulários de registro (figura 20) partindo da tela de detalhe da internação (Figura 19).

O Planejamento destes formulários foi baseado nos guias de estilo e padrões aplicados às outras seções da aplicação. Podem-se notar a utilização padronizada das cores, tipografia clara, textos objetivos, listas de única ou múltipla escolha (facilitando a inserção de dados, utilização de notas (Assistente de preenchimento), entre outros elementos layout focados na experiência do usuário.

Figura - ICU Surviving: Registrar dados SAPS 3 e APACHE II



Fonte: O autor

# CONCLUSÕES E TRABALHOS FUTUROS

Este trabalho teve como foco investigar os desafios que envolvem o design móvel para dispositivos com sistema operacional *Android* utilizando como estudo de caso a proposta de elaboração de protótipos de alta fidelidade para o sistema ICU Surviving, desde a concepção das ideias até a arte final.

A cada etapa desenvolveu-se os conceitos considerados mais importantes ao sucesso do projeto, utilizando-se do manual de boas práticas fornecido pelo *Google* e outras referências de qualidade equivalente.

A elaboração da proposta de interface para o sistema ICU Surviving permitiu, a análise das as principais dificuldades, bem como algumas soluções relacionadas, ao se propor um design focado na experiência do usuário de dispositivos com sistema operacional *Android*.

Numa visão mais específica, foi possível analisar as principais definições de design, padrões e estruturas referenciadas no guia de desenvolvimento *Google* – *Android* e outras bibliografias da área.

Identificou-se quais técnicas e conceitos se adequariam ao projeto proposto, pelos quais foi possível estabelecer uma boa relação entre elementos visuais, interatividade e funcionalidades para a aplicação.

Por último, pela exploração de algumas ferramentas de design, desenvolveu-se um protótipo de alta fidelidade, permitindo testes e simulações de interação focados na experiência do usuário.

Após as validações e adaptações do protótipo proposto, o mesmo foi utilizado como referência ao desenvolvimento da aplicação.

Vale ressaltar a importância da interdisciplinaridade para ambas as áreas envolvidas. Essa relevância se justificou por permitir o compartilhamento de conhecimentos até então discrepantes, com a finalidade de viabilizar o desenvolvimento de tecnologias inovadoras.

Espera-se com este trabalho, contribuir como referência a outras pesquisas e projetos futuros. Para tanto, antes de tudo, propõe-se investigação mais profunda sobre os assuntos abordados, visto que cada tema tem uma riqueza infinita, impossível de se tratar em um projeto de monografia.

# REFERÊNCIAS

ANDROID DEVELOPERS.Guia de design para apps Android. **Android Design.** Disponível em:<http://developer.android.com/design/index.html>. Acesso em 26/02/2015.

BARBOSA, Simone D. J; SILVA, Bruno Santana da; **Interação Humano-Computador**. São Paulo: Editora Elsevier,2010.

BRASIL. ANVISA. Agência Nacional de Vigilância Sanitária. **Resolução RDC nº 07, de 24 de fevereiro de 2010**. Dispõe sobre os requisitos mínimos para funcionamento de Unidades de Terapia Intensiva e dá outras providências. ANVISA Publicações Eletrônicas. 2010. Disponível em: <http://bvsms.saude.gov.br/bvs/saudelegis/anvisa/2010/res0007\_24\_02\_2010.html>. Acesso em 26/02/2015.

CUELLO, Javier; VITONE, José; **Diseñando apps para móviles.** Editora José Vitone. 2013.

GONÇALVES, Waldiere Machado at al. Mortality evaluated by apache II prognostic system in a surgical critical care unit. **Revista do Colégio Brasileiro de Cirurgiões.** Vol. 26, n. 2, p. 115-118. Janeiro,1999.

KNAUS, Willian A. et al; APACHE II: A severity of disease classification system. **Critical Care Medicine.**Vol. 13, n. 10, p. 818-819, Outubro, 1985**.**

KOMATINENI, Satya; MCLEAN, Dave; **Expert Android.** Editora Apress, 2013.

LEAL, Nelson Glauber; **Dominando o Android: Do básico ao avançado.** São Paulo: Editora Novatec, 2015.

LEHTIMÄKI, Juhani; **Smashing Android UI:Responsive user interfaces and design patterns forAndroid phones andtablets.** USA:Editora John Wiley & Sons, 2013

MENDONZA, Adrian; **Mobile User Experience: Patterns to make sense of it all.** USA: Editora Elsevier, 2014.

METNITZ, Philipp G. H. et al; SAPS 3—From evaluation of the patientto evaluation of the intensive care unit. Part 1: Objectives, methods and cohort description. **Intensive Care Med**. Vol. , n. , p. 1336-1344, Agosto, 2005.

MORENO, Rui P. et al; SAPS 3—From evaluation of the patient to evaluation of the intensive care unit. Part 2: Development of a prognostic model for hospital mortality at ICU admission**. Intensive Care Med**. Vol. , n. , p. 1345-1355, Agosto, 2005.

NEIL**,** Theresa;**Mobile Design Pattern Gallery: UI patterns for smartphone Apps.** Canadá: Editora O’Reilly Media, 2014.

NIELSEN, Jacob. **Design Usability Engineering**. Academy Press. 1993.

ROCHA, Heloísa V. da; BARANAUSKAS, Maria C. C. **Design e Avaliação de Interfaces Humano-Computador**. Unicamp. 2003.

SILVA JUNIOR, João Manoel da et al.Aplicabilidade do Escore Fisiológico Agudo Simplificado (SAPS 3) em Hospitais Brasileiros. **Revista Brasileira de Anestesiologia**, V. 60, n. 1, p. 20-31, Janeiro/Fevereiro, 2010.

TERZI, Renato G. et al.Índices prognósticos em Medicina Intensiva. **Revista Brasileira deTerapia Intensiva**, Vol. 14, n. 1, p. 6-21, Janeiro/Março, 2002.

# ANEXO 1 – TABELA DE PONTUAÇÃO - SAPS 3

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Demográfico / estado prévio de saúde** | | **Categoria diagnóstica** | | **Variáveis fisiológicas na admissão** | |
| ***Variáveis*** | ***Pontos*** | ***Variáveis*** | ***Pontos*** | ***Variáveis*** | ***Pontos*** |
| **Idade** |  | **Admissão programada** | 0 | **Glasgow** |  |
| < 40 | 0 | **Admissão não programada** | 3 | 3 - 4 | 15 |
| ≥ 40-<60 | 5 | **Status cirúrgico** |  | 5 | 10 |
| ≥ 60-< 70 | 9 | Não cirúrgico | 5 | 6 | 7 |
| ≥ 70-< 75 | 13 | Eletiva / Planejada | 0 | 7 - 12 | 2 |
| ≥ 75-<80 | 15 | Emergência | 6 | ≥ 13 | 0 |
| ≥ 80 | 18 | **Tipo de operação** |  | **Frequência cardíaca** |  |
| **Comorbidades** |  | Transplantes | -11 | < 120 | 0 |
| Outras | 0 | Trauma | -8 | ≥ 120-< 160 | 5 |
| Quimioterapia | 3 | RM sem valva | -6 | ≥ 160 | 7 |
| ICC NYHA IV | 6 | Cirurgia no AVC | 5 | **Pressão arterial sistólica** |  |
| Neoplasia hematológica | 6 | Outras | 0 | < 40 | 11 |
| Cirrose | 8 | **Admissão na UTI acrescentar 16 pontos** | 16 | ≥ 40-< 70 | 8 |
| Aids | 8 | **Motivo de internação / Admissão** |  | ≥ 70-< 120 | 3 |
| Metástase | 11 | **Neurológicas** |  | ≥120 | 0 |
| **Dias de internação prévios** |  | Convulsões | -4 | **Oxigenação** |  |
| < 14 | 0 | Coma, confusão, agitação | 4 | com VM e relação PaO2/FiO2 < 100 | 11 |
| ≥ 14-28 | 6 | Déficit Focal | 7 | com VM e relação PaO2/FiO2 ≥ 100 | 7 |
| ≥ 28 | 7 | Efeito de massa intracraniana | 11 | Sem VM e PaO2 < 60 | 5 |
| **Procedência** |  | **Cardiológicas** |  | Sem VM e PaO2 ≥ 60 | 0 |
| Centro cirúrgico | 0 | Arritmia | -5 | **Temperatura** |  |
| Pronto Socorro | 5 | Choque hemorrágico | 3 | < 34,5 | 7 |
| Outra UTI | 7 | Choque hipovolêmico não hemorrágico | 3 | ≥ 34,5 | 0 |
| Outros | 8 | Choque distributivo | 5 | Leucócitos |  |
| **Fármacos vasoativos** |  | **Abdômen** |  | < 15.000 | 0 |
| Sim | 0 | Abdômen agudo | 3 | ≥ 15.000 | 2 |
| Não | 3 | Pancreatite grave | 9 | **Plaquetas** |  |
|  |  | Falência hepática | 6 | < 20.000 | 13 |
|  |  | Outras | 0 | ≥ 20.000-< 50.000 | 8 |
|  |  | **Infecção** |  | ≥ 50.000-< 100.000 | 5 |
|  |  | Nosocomial | 4 | ≥ 100.000 | 0 |
|  |  | Respiratória | 5 | **pH** |  |
|  |  | Outras | 0 | ≤ 7,25 | 3 |
|  |  |  |  | > 7,25 | 0 |
|  |  |  |  | **Creatinina** |  |
|  |  |  |  | < 1,2 | 0 |
|  |  |  |  | ≥ 1,2-< 2,0 | 2 |
|  |  |  |  | ≥ 2,0-< 3,5 | 7 |
|  |  |  |  | ≥ 3,5 | 8 |
|  |  |  |  | **Bilirrubina** |  |
|  |  |  |  | < 2 | 0 |
|  |  |  |  | ≥ 2-< 6 | 4 |
|  |  |  |  | ≥ 6 | 5 |

# ANEXO 2 – FORMULÁRIO PADRÃO – APACHE II

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Variáveis fisiológicas** | **+4** | **+3** | **+2** | **+1** | **0** | **+1** | **+2** | **+3** | **+4** |
| Temperatura retal (C) | > 41 | 39-40,9 |  | 38,5-38,9 | 36-38,4 | 34-35,9 | 32-33,9 | 30-31,9 | < 29,9 |
| Pressão arterial média mmHg | >160 | 139-159 | 110-129 |  | 70-109 |  | 50-69 |  | < 40 |
| Freqüência cardíaca bpm | > 180 | 140-179 | 110-139 |  | 70-109 | 55-69 | 40-54 | < 39 |  |
| Freqüência respiratória irpm   (ventilados ou não) | > 50 | 35-49 | 25-34 | 12-24 | 10-11 | 6-9 |  | < 5 |  |
| Oxigenação A-aDO2 |  |  |  |  |  |  |  |  |  |
| a) FiO2 > 0,5 A-aDO2 | >500 | 350-499 | 200-349 |  | < 200 |  |  |  |  |
| b) FiO2< 0,5   PaO2 |  |  |  |  | >70 | 61-70 |  | 55-60 | < 55 |
| pH Arterial | > 7,7 | 7,6-7,69 |  | 7,5-7,59 | 7,33-7,49 |  | 7,25-7,32 | 7,15-7,24 | < 7,15 |
| Sódio sérico (mEq/L) | > 180 | 160-179 | 155-159 | 150-154 | 130-149 |  | 120-129 | 111-119 | < 110 |
| Potássio sérico (mEq/L) | > 7 | 6-6,9 |  | 5,5-5,9 | 3,5-5,4 | 3-3,4 | 2,5-2,9 |  | < 2,5 |
| Creatinina sérica (mg/dL) dobrar pontos se IRA | > 3,5 | 2-3,4 | 1,5-1,9 |  | 0,6-1,4 |  | < 0,6 |  |  |
| Hematócrito (%) | > 60 |  | 50-50,9 | 46-49,9 | 30-45,9 |  | 20-29,9 |  | < 20 |
| Número de leucócitos | > 40 |  | 20-39,9 | 15-19,9 | 3-14,9 |  | 1-2,9 |  | < 1 |
| Escala de Glasgow para o coma (CGS) | | | | | | | | | |
| **(A)** Total do escore fisiológico agudo (APS) = **soma dos valores** | | | | | | | | | |
| Bicarbonato sérico (mEq/L) (usar se não coletar gasometria) | > 52 | 41-51,9 |  | 32-40,9 | 22-31,9 |  | 18-21,9 | 15-17,9 | < 15 |
| **(B) Pontos por Idade**   |  |  | | --- | --- | | Idade(Anos) | Pontos | | <= 44 | 0 | | 45-54 | 2 | | 55-64 | 3 | | 65-74 | 5 | | >=75 | 6 |   **Pontuação Apache =**  **(A) + (B) + (C)** | **(C) Pontos por doença crônica**  Se o paciente tem uma história de insuficiência grave de órgãos ou é imunocomprometido; assinale pontos como se segue:  a) Para pacientes não-cirúrgicos ou pós-operatórios de emergência: 5 pontos  b) Para pacientes de pós-operatórios eletivos: 2 pontos  **Definições**  A insuficiência de órgão ou o estado de imunodepressão deve ser evidente antes da admissão hospitalar e deve obedecer o seguinte critério:  Fígado: Cirrose comprovada por biópsia, hipertensão portal documentada; episódios passados de hemorragia gastrintestinal atribuídos à hipertensão portal; episódios anteriores de insuficiência hepática, encefalopatia ou coma.  Cardiovascular: New York Association classe IV.  Respiratória: Doença crônica restritiva, obstrutiva ou vascular resultando em grave restrição ao exercício, isto é, incapaz de subir escadas ou fazer serviços domésticos; hipóxia crônica documentada, hipercapnia, policitemia secundária, hipertensão pulmonar grave (> 40 mmHg); dependência de prótese ventilatória.  Renal: Recebendo diálise cronicamente.  Imunocomprometido: Paciente tem recebido terapia que suprime a resistência à infecção, isto é, imunossupressores, quimioterapia, radioterapia, corticóides cronicamente ou recente em altas doses; doença que é suficientemente avançada para suprimir a resistência à infecção, isto é, leucemia, linfoma, AIDS. | | | | | | | | |

1. Simplified Acute Physiology Score [↑](#footnote-ref-1)
2. Acute Physiology And Chronic Health Evaluation [↑](#footnote-ref-2)
3. Unidades de Terapia Intensiva [↑](#footnote-ref-3)
4. Nome sugerido à aplicação proposta [↑](#footnote-ref-4)
5. Interação Humano-Computador [↑](#footnote-ref-5)
6. iPhone Operation System [↑](#footnote-ref-6)
7. Sistema Operacional [↑](#footnote-ref-7)
8. User Interface [↑](#footnote-ref-8)
9. *HyperText Markup Language* [↑](#footnote-ref-9)
10. *Cascading Style Sheets* [↑](#footnote-ref-10)
11. https://codiqa.com/ [↑](#footnote-ref-11)
12. https://www.fluidui.com/ [↑](#footnote-ref-12)
13. http://framerjs.com/ [↑](#footnote-ref-13)
14. https://www.flinto.com/ [↑](#footnote-ref-14)
15. http://giveabrief.com/ [↑](#footnote-ref-15)
16. https://popapp.in/ [↑](#footnote-ref-16)
17. Agência Nacional de Vigilância Sanitária [↑](#footnote-ref-17)
18. Resolução da Diretoria Colegiada [↑](#footnote-ref-18)