

Alvaro Portela Figueiredo Neto

Proposta de sistema de design e biblioteca de componentes de prototipação para o Tribunal de Contas do Estado do Rio Grande do Norte

Natal – RN

Dezembro de 2020

Alvaro Portela Figueiredo Neto

**Proposta de sistema de design e biblioteca de
componentes de prototipação para o Tribunal de
Contas do Estado do Rio Grande do Norte**

Trabalho de Conclusão de Curso da Residência em Tecnologia da Informação do Instituto Metrópole Digital da Universidade Federal do Rio Grande do Norte.

Orientador: Itamir de Moraes Barroca Filho

Universidade Federal do Rio Grande do Norte – UFRN

Instituto Metrópole Digital – IMD

Residência em Tecnologia da Informação

Natal – RN

Dezembro de 2020

Alvaro Portela Figueiredo Neto

**Proposta de sistema de design e biblioteca de
componentes de prototipação para o Tribunal de
Contas do Estado do Rio Grande do Norte**

Trabalho aprovado. Natal – RN, 29 de Janeiro de 2021:

Prof. Dr. Itamir de Moraes Barroca Filho - Orientador
UFRN

Prof. Dr. Jair Cavalcanti Leite
UFRN

MSc. André Gustavo Almeida e Silva
TCE-RN

Natal – RN
Dezembro de 2020

RESUMO

É competência do Tribunal de Contas do Estado do Rio Grande do Norte (TCE-RN) o controle externo sobre a administração pública de acordo com o que rege a Constituição Federal de 1988, assim como a prestação de contas à sociedade das entidades jurisdicionadas sob sua guarda. Para a execução dessas ações o TCE-RN desenvolve e utiliza diversos Sistemas Web, seja para permitir que entidades jurisdicionadas cadastrem ou acesse informações referentes ao escopo de interesses do TCE-RN ou permitir a divulgação de dados a população e aos Jurisdicionados como determinada pela Lei Complementar 131 também conhecida como Lei de Transparência. Diversos processos e tecnologias são usados no desenvolvimento dos sistemas do TCE-RN e o objetivo deste trabalho será propor um sistema de design para prototipação de interfaces de usuário que seja ágil, colaborativo, versionável e adaptável, criando um conjunto componentes para web.

Palavras-chaves: Sistemas de design. Prototipação. Biblioteca de componentes. Figma.

ABSTRACT

The Court of Auditors of the State of Rio Grande do Norte (TCE-RN) is responsible for external control over public administration in accordance with the 1988 Federal Constitution, as well as the rendering of accounts to society of the jurisdicted entities under its guard. For the execution of these actions, TCE-RN develops and uses several Web Systems, either to allow jurisdicted entities to register or access information related to the scope of interests of TCE-RN or to allow the disclosure of data to the population and to jurisdicted's as determined by Law Complementary 131 also known as the Transparency Law. Several processes and technologies are used in the development of TCE-RN systems and the objective of this work will be to propose a user interface design system for prototyping that is agile, collaborative, versionable and adaptable, creating a set of components for the web.

Keywords: Design System. Prototyping. tComponent Library. Figma.

LISTA DE ILUSTRAÇÕES

Figura 1 – Figma e seus principais concorrentes.	15
Figura 2 – Análise da biblioteca de componentes existente no TCE-RN.	21
Figura 3 – Exemplo de Composição de elementos.	22
Figura 4 – Resultado Da primeira iteração do Figma.	23
Figura 5 – Diferentes estados de botões da biblioteca de compoentes do TCE-RN.	24
Figura 6 – Separação dos Componentes	24
Figura 7 – Resultado da prototipação do sistema SIAI Concursos.	26

LISTA DE TABELAS

Tabela 1 – Vantagens e desvantagens do nível de fidelidade de protótipos	14
Tabela 2 – Comparativo Figma vs Pencil	20

LISTA DE ABREVIATURAS E SIGLAS

TCE-RN	<i>Tribunal de Contas do Estado do Rio Grande do Norte</i>
API	<i>Application Programming Interface</i>
SIAI	<i>Sistema Integrado de Auditoria Informatizada</i>
DIN	<i>Diretoria de Informática</i>
IMD	<i>Instituto Metr�pole Digital</i>
HTML	<i>HyperText Markup Language</i>
CSS	<i>Cascade Style Sheet</i>
REST	<i>Representational State Transfer</i>
Pencil	<i>Pencil Project</i>
MVP	<i>Produto M�nimo Vi�vel</i>
SD	<i>Sistema de Design</i>

SUMÁRIO

1	Introdução	9
1.1	Apresentação do problema	10
1.2	Apresentação da solução	10
1.3	Estrutura do Trabalho	10
2	Capítulo 2	11
2.1	Afinal o que é um sistema de design	11
2.2	Princípios e Tecnologias usadas para desenvolver um sistema de design	12
3	O Figma	15
3.1	Sketch	16
3.2	AdobeXD	16
3.3	Figma	17
3.4	Ferramenta Adotada	18
4	Capítulo 3	19
4.1	Situação Atual Do Design no TCE-RN	19
4.2	Reconstrução Do Modelo Existente	20
4.3	Teste de Execução e Problemas Encontrados	22
4.4	O Nascimento De Um Proto Sistema De Design	23
5	Capítulo 5	26
6	Capítulo 6	28
6.1	Conclusões	28
6.2	Melhorias	28
	Referências	29

1 INTRODUÇÃO

O Tribunal de contas do Estado do Rio Grande do Norte (TCE-RN), segundo /citeLegislativa, é um órgão de controle externo responsável por auxiliar o poder legislativo estadual, especializado na análise e fiscalização de contas públicas.

A exceção dos dados sob sigilo ou restrições da lei nº 12.557 (JURÍDICOS, 2011) é dever do Poder Executivo Federal promover a cultura de transparência pública, a publicação de dados contidos em bases de dados de órgãos e entidades da administração pública federal direta, autárquica e fundacional sob a forma de dados abertos assim como fomentar o controle social e o desenvolvimento de novas tecnologias destinadas à construção de ambiente de gestão pública participativa e democrática e à melhor oferta de serviços públicos para o cidadão (JURÍDICOS, 2016).

Embora seja um tribunal, o TCE-RN não encontra-se circunscrito nem faz parte do Poder Judiciário, pois seu caráter é de natureza, administrativa (contábil), trabalhando em parceria e não em subordinação ao Judiciário (BARRETO, 2016).

Portanto, como corte cuja finalidade é a fiscalização de Prefeituras dos Municípios e unidades jurisdicionados, o TCE-RN precisa de transparência e correção dos dados a serem avaliados afim de fornecer essa mesma transparência e correção de volta aos cidadãos, aqueles a quem realmente servem como órgão público.

Para atender a crescente demanda da sociedade pela ampliação da transparência, O TCE-RN oferece o acesso a uma Interface de Programação de Aplicativos (API), do inglês *Application Programming Interface*, no formato de Dados Abertos sem restrições de uso possibilitando a criação de aplicações digitais (NORTE, 2020) que façam uso dos dados do seu Sistema Integrado de Auditoria Informatizada (SIAI) permitindo que não só o cidadão desenvolva aplicativos que formentem o controle social, como as equipes de desenvolvimento do tribunal, facilitando o compartilhamento de dados entre aplicações internas ao tribunal desenvolvidas pela Diretoria de Informática do TCE-RN (DIN).

A DIN é a unidade responsável por desenvolver, gerenciar e fornecer serviços de tecnologia da informação para o Tribunal. Até o primeiro trimestre de 2019 a DIN executou com a ajuda do seu corpo de servidores, estagiários, do Instituto Metrópole Digital (IMD), e da Indra, empresa terceirizada que presta serviços de Software house ao TCE-RN, ao menos 12 projetos e 3718 atendimentos técnicos ao Tribunal (NORTE, 2019).

1.1 Apresentação do problema

A DIN conta entre servidores, estagiários, terceirizados, e outros colaboradores com um total de 58 membros na sua unidade (NORTE, 2019) que, entre outras demandas, precisam desenvolver sistemas e ferramentas para melhor atender não só as necessidades internas da sua diretoria como a de entidades externas jurisdicionadas, seja para consumir ou fornecer dados ao Tribunal de Contas do Estado.

Uma das grandes dificuldades atualmente no tribunal está em manter a coesão entre tantos sistemas criados ao longo da sua história. Outro problema está na velocidade de produção desses sistemas que variam imensamente em escopo e complexidade, passando por longos períodos de levantamento de requisitos, prototipação e produção, até a sua homologação. Além disso, apesar de fornecer acesso a informação por meio de APIs, caso o cidadão deseje desenvolver um sistema ou serviço baseado nos dados abertos fornecidos pelo tribunal, os mesmos não podem contar com outra referência aos modelos de interface humano computador além das telas de sistemas com acesso público providas pelo TCE-RN.

1.2 Apresentação da solução

Para mitigar os problemas anteriormente citados de acesso público aos modelos de interfaces e favorecer a coesão entre os sistemas novos a serem desenvolvidos, a sugestão deste trabalho é propor uma cultura de prototipação e evoluir o modelo de identidade visual existente construindo um sistema de design e uma biblioteca de componentes reutilizáveis que seja colaborativamente editável, evoluído e versionado. Permitindo a rápida prototipação de interfaces de usuários de alta fidelidade que sejam facilmente, testáveis pelos clientes da DIN e que permitam ao cidadão criar sistemas ou serviços com uma experiência de usuário única em torno do ecossistema do tribunal.

1.3 Estrutura do Trabalho

Este trabalho apresenta inicialmente uma introdução sobre as obrigações e motivações do Tribunal de Contas do Estado do Rio Grande do Norte e dos fatores que levanta as justificativas da elaboração deste trabalho. Na sequência o Capítulo 2 contendo o embasamento teórico e o estado da arte para a elaboração deste trabalho, em seguida o Capítulo 4 introduz modelos e exemplos dos documentos gerados pela ferramenta desenvolvida assim como a descrição do trabalho produzido. O Capítulo 3 trata dos resultados obtidos com o que foi desenvolvido enquanto no Capítulo 5 são abordadas as conclusões e possíveis melhorias a este trabalho.

2 ESTADO DA ARTE E FUNDAMENTAÇÃO TEÓRICA

Neste capítulo será apresentado o estado da arte referente a construção de sistemas de designs e prototipação, assim como o material utilizado como embasamento teórico para o seu desenvolvimento.

2.1 Afinal o que é um sistema de design

Realizando uma breve pesquisa na internet a expressão mais comumente encontrada para descrever o que seria um sistema de design remetem a Nathan Curtis e a frase "*A Design System isn't a Project. It's a Product, Serving Products*". (CURTIS, 2016) Poderíamos traduzir para Um sistema de design não é um projeto. É um produto servindo outros produtos.

Um sistema de design é a representação viva de um conjunto de documentos com o intuito de unificar a identidade visual de um ecossistema de produtos de uma instituição. Para servir a este propósito esses documentos não podem ser estáticos, devem permitir constantes mudanças e conseguir manter a ideia central de poder encaixar formas sem perder consistência (FEDERAL, 2019a). Em 2019, o Governo Federal decidiu iniciar a construção do seu sistema de design com uma motivação clara

[...]A iniciativa potencializa a eficiência e a eficácia dos usuários na utilização de interfaces para acesso aos serviços e aos sistemas de Governo, possibilitando uma única curva de aprendizado e garantindo a previsibilidade na utilização dos diferentes sistemas. (FEDERAL, 2019b)

Para Gonzalez um sistema de design é "[...]um conjunto de padrões para design e código, componentizado que unificam as duas práticas". (GONZALEZ, 2019) Para o Governo Federal um *Design System* "[...]tem como objetivo guiar todos os responsáveis pela construção de interfaces interativas orientadas à experiência única do usuário, considerando a acessibilidade e a usabilidade dos sistemas". (FEDERAL, 2019b)

É fato que grandes entidades vêm adotando a elaboração de sistemas de design com o intuito de se beneficiar das vantagens proporcionadas pela unificação da linguagem de comunicação com usuários de seus sistemas, acelerar o processo de prototipação de seus layouts e interfaces e, diminuir o tempo de desenvolvimento das aplicações por meio do respaldo de bibliotecas de componentes mantidas por esses sistemas.

Para citar algumas das empresas e entidades que possuem ou estão em processo de desenvolvimento de um sistema de design:

- [Google](#)
- [Microsoft](#)
- [Apple](#)
- [IBM](#)
- [GitLab](#)
- [Governo Federal](#)
- [Governo do Reino Unido](#)
- [Governo da Australia](#)

2.2 Princípios e Tecnologias usadas para desenvolver um sistema de design

Existem uma quantidade virtualmente infinita de maneiras de se construir um *Design System*. Contudo uma das maneiras mais comuns de se proceder com seu desenvolvimento consiste em separar o conjunto de documentos gerados, agrupando-os em categorias de acordo com a sua finalidade.

No caso do Governo Federal, por exemplo, temos seu sistema de design separado em: Fundamentos Visuais, Componentes, Templates, Guias e Orientações. Esta separação foi feita com o intuito de melhor guiar a decisões de design, ser autênticos e específicos, em vez de genéricos e ser facilmente memorizáveis, fáceis de lembrar e utilizáveis no dia a dia. (FEDERAL, 2019b)

o Gitlab por sua vez optou por uma estrutura organizacional de seu sistema em: Fundação, Layout, Componentes, Visualização de dados, Regiões, Objetos, Conteúdo, Usabilidade e Recursos de Design. Para a equipe do GitLab o foco era que a companhia precisaria informar seus princípios, valores e propósito, promover a produtividade e promover empatia (GITLAB, 2020).

A Microsoft vem passando sucessivas mudanças. Com um grande leque de produtos que variam de serviços de nuvem a sistemas operacionais para dispositivos embarcados, unificar uma linguagem de comunicação sempre foi um problema para esta empresa. Atualmente, um dos focos da microsoft vem sendo o desenvolvimento de um linguagem visual chamada Fluent Design com o intuito criar engajamento e permitir acessibilidade e

internacionalização em um ecossistema de programas e serviços focados em produtividade que se estende por múltiplas plataformas e sistemas operacionais. Separando inicialmente seu protótipo por diferentes plataformas. Seu projeto de design se organiza em elementos básicos, layout, controles, estilo, movimento, "casca".

Fluent experiences listen and adapt. They feel natural on the devices people use, from tablets to laptops, from PCs to televisions. They travel from the office to the living room to virtual worlds (MICROSOFT, 2020).

Estruturas de organização semelhantes se repetem e podem ser encontradas nos sistemas de design desenvolvidos, ou em desenvolvimento, de todas as empresas citadas na listagem da seção anterior.

O agrupamento desses documentos são ações subjetivas e que devem ser executadas em parceria e comum acordo entre a equipe de design e de desenvolvimento. A categorização de estilos e componentes precisa não somente manter uma identidade visual única através das suas diferentes categorizações como fazer sentido de modo facilitar a sua executabilidade pelos desenvolvedores.

Essa abordagem conjunta é importante, pois o desenvolvimento de um sistema de design em silos isolados por designers resulta em: Desperdício de tempo e recursos humanos, sentimentos negativos e falta de compreensão sobre o design proposto (GONZALEZ, 2019). É possível extrapolar facilmente esta mentalidade de crescimento de membros de uma equipe para a quantidade de equipes num projeto.

Softwares são muitas vezes construídos por times, que podem variar de times pequenos até equipes muito grandes e a medida que cresce a quantidade de membros em um time cresce a dificuldade em manter uma experiência de usuário coesa, pois cada pessoa traz opiniões e soluções com estilos diferentes que irão facilmente divergir (SAARINEN, 2016)

Para promover essa integração entre desenvolvedores e designers, ferramentas de desenvolvimento e comunicação são utilizadas para não só construir e manter, mas também para compartilhar e compreender as decisões e práticas adotadas. Dentro do vasto conjunto de ferramentas existentes para tal, podemos citar algumas das mais conhecidas:

- [Sketch](#)
- [AdobeXD](#)
- [Figma](#)
- [Studio](#)
- [Pencil](#)

- [Balsamiq](#)

Vale salientar que as ferramentas anteriormente citadas são usadas principalmente para prototipação e não são as únicas existentes no mercado, mas algumas das principais. Quanto a prototipação, ela é uma das etapas dos processos envolvidos no desenvolvimento de um produto e é fundamental, pois acelera etapas de validação e na consolidação de alguns elementos obtidos durante o levantamento de requisitos.

Os protótipos criados podem ser de baixa ou alta fidelidade, estes termos de qualificação servem para descrever a distância obtida pela aparência ou comportamento entre o protótipo e o produto final a ser desenvolvido e não é incomum serem usados ambos, em diferentes etapas do processo de desenvolvimento do sistema ou da prototipação dependendo de quão avançado está a validação ou quão refinado deseja-se descrever o comportamento do sistema ou serviço.

Tabela 1 – Vantagens e desvantagens do nível de fidelidade de protótipos

Fidelidade	Vantagens	Desvantagens
Baixa	Baixo custo de desenvolvimento Avaliação de múltiplos conceitos Aponta problemas de telas Prova de conceito	Checagem de erro limitada Especificação de detalhamento pobre Voltado ao facilitador Sem valor após especificação de requisitos Valor limitado para testes de usabilidade Fluxo e navegação limitados
Alta	Funcionalidade Completa Interativo Voltado ao usuário Esquema de navegação claros Permite testes exploratórios Sensação do produto final Atua como especificação viva	Desenvolvimento mais custoso Demorado para criar Ineficiente para provas de conceito Ineficiente para obtenção de requisitos

Fonte: (RICHARDSON et al., 2015)

É comum que o desenvolvimento do sistema de design anteceda um passo de muita importância que é a codificação desse design, criando assim, uma biblioteca de componentes utilizáveis tanto para o desenvolvimento do protótipo como do produto. Todas as empresas e entidades citadas no início do capítulo possuem essa implementação em maior ou menor grau de cobertura dos componentes, algumas em diferentes linguagens de programação para uma ou mais plataformas, como a web ou mobile. O Tribunal de contas do estado conta com uma implementação feita anteriormente a criação do sistema de design e com ajuda deste trabalho iremos ampliar o escopo dessa biblioteca.

3 O FIGMA E AS MOTIVAÇÕES DE SUA ESCOLHA COMO FERRAMENTA DE PROTOTIPAÇÃO.



Figura 1 – Figma e seus principais concorrentes.

Fonte da imagem: [Medium](#)

Como foi dito no capítulo anterior existem diversas ferramentas de design disponíveis no mercado e os principais concorrentes são o Sketch e o AdobeXD. Ambas ferramentas muito boas e em diferentes estágios de desenvolvimento. No decorrer deste capítulo falarei mais sobre cada uma das ferramentas e explicarei os motivos que levaram a adoção do Figma para o desenvolvimento deste trabalho.

3.1 Sketch

Sketch foi desenvolvido pela Bohemian Coding e foi liberado originalmente para macOS em 2010. É considerado uma dos primeiros softwares de design de experiência de usuário e também um dos primeiros softwares disruptivos no cenário UX/UI.

Como funcionalidades-chaves o Sketch possui suporte a plugins assim como um preview instantâneo das alterações dos protótipos, presets de exportação automáticos de componentes, grades e linhas de guia e suporte a salvamento dos documentos em nuvem.

Algumas das vantagens do Sketch:

- Desin facilmente escalável.
- permite a construção de bibliotecas facilitando a manutenibilidade de componentes.
- A funcionalidade: Pixel-perfect torna o design muito preciso.
- Renderização rápida e precisa em relação ao que será mostrado no browser.
- Alto engajamento e suporte da comunidade.

Algumas desvantagens do Sketch:

- Não possui versão gratuita.
- Disponível apenas para macOS.
- Não possui mecanismos de edição e retoque de imagem.
- não possui funcionalidades de prototipação próprias dependendo de plugins.
- Muito dependente da comunidade em relação a manutenção de funcionalidades providas por plugins.
- Exigente em relação a recursos de hardware.

3.2 AdobeXD

O AdobeXD é a mais nova das ferramentas de design citadas neste capítulo, tendo sido lançada em 2017 pela Adobe, o software foi publicado suportando execução tanto no mac quanto no windows com extensão a testes de protótipos no ios e no android.

Possui as principais funcionalidades a co-edição, o suporte a web, desktop e mobile, drag and drop de apartir do artboard, animações automáticas e integração multi plataforma.

Algumas das vantagens do AdobeXD:

- Ferramenta de design mantida pela Adobe passa a sensação de robuste e confiança.
- Permite a edição e o reuso de elementos com suporte a redimensionamento em grupos.
- Prototipos interativos com suporte a multiplas plataformas.
- Facilidade de uso por usuarios com experiência com outros produtos da Adobe.
- Salvamento em nuvem e edição de permissões de acesso como por exemplo apenas visualizar.

Algumas desvantagens do AdobeXD:

- Edição de formas muito limitada, permitindo apenas formas geometricas basicas.
- Necessario adição de plugins para permitir visualização do CSS dos componentes.
- Sem suporte a edição de imagens, valendo-se do uso do Photoshop para tal.
- Ainda em desenvolvimento, recebe alterações constante o que pode quebrar a experiencia do uso.
- Apesar de gratuito durante a fase de desenvolvimento a Adobe pretente passar a cobrar mensalidade nos moldes de outras de suas ferramentas.

3.3 Figma

O Figma foi lançado em 2016 com a proposta de ser uma ferramenta de design baseada inteiramente na nuvem. Ser um hub que permita prototipação e design colaborativo. Esta mentalidade e filosofia fez com que tornase-se uma das melhores ferramentas de prototipação e design disponiveis permitindo que times trabalhem no mesmo arquivo em tempo real.

Como principais vantagens podemos destacar que é uma ferramenta totalmente baseada em nuvem, que suporta edição tanto no desktop (windows e mac) como no browser em que arquivos podem ser editados e testado tempo real pelas equipes envolvidas. Alem disso recentemente passou a dar suporte a extensões e desenvolveu um app para android e ios chamado Figma Mirror que permite testes de prototipos usando essas plataformas mobile.

Algumas das vantagens do Figma:

- Facilidade de trabalhar com grandes equipes em tempo real de forma transparente.

- Compartilhamento de arquivos de maneira fácil e rápida na nuvem torna os elementos de design altamente acessíveis, pulando etapas de importação e exportação.
- Por ser baseado em nuvem o figma se torna agnostico de plataformas.
- A comunidade muito ativa vem criando uma ampla variedade de plugins.
- Como ferramenta de prototipação e design integrados e baseado em nuvem permitindo desenvolvimento e teste colaborativo o Figma se torna um pacote completa para empresas.
- Possui historico de versões semelhante ao Git.

Algumas desvantagens do Figma:

- Possui muito poucas funcionalidades offline.
- Necessita de maquinas com desempenho intermediario para ater uma boa experiencia de uso'.
- A ferramenta vem recebendo constantes alterações e evoluções que podem confundir o usuário.
- Interface confusa.

3.4 Ferramenta Adotada

O Figma possui serias limitações de trabalho quando a internet passa por problemas ou esta indisponivel, pois é inteiramente baseada em computação na nuvem, contudo é uma ferramenta capaz de concentrar a documentação os elementos do design e os prototipos de sistemas alem de permitir testes de usabilidade com um bom grau de fidelidade no fluxo de interações.

Além disso o figma é capaz de rodar em maquinas com desempenho intermediario, não sendo tão exigente de hardware como Sketch por exmeplo. Quanto a precificação o Figma assim como o AdobeXD é ofertado por meio de pacotes mensais, contratados por ano ou mes a mes. contudo o figma se posiciou por manter uma serie de funcionalidades gratuitas enquanto o AdobeXD não deu garantias de mante algum pacote de serviços disponiveis como freemium. Por fim a mensalidade do Figma ainda é menor que a planejada pela Adobe para a sua ferramenta.

Por esses motivos o figma foi escolhido como ferramenta de design e prototipagem usada neste trabalho.

4 SITUAÇÃO ATUAL E DESENVOLVIMENTO DO SISTEMA DE DESIGN NO TCE-RN

Tendo em vista, o potencial que ferramentas de design e prototipação possuem em acelerar o desenvolvimento e a validação de sistemas e serviços além da necessidade do Tribunal de Contas do Estado em agilizar os seus processos de desenvolvimento e a crescente demanda por novos sistemas e pela evolução dos sistemas e serviços prestados, não somente a suas diretorias como aos entes jurisdicionados, foi abordada junto a DIN se haveria por parte deles o interesse no desenvolvimento de um sistema de design. A Diretoria de Informática se mostrou interessada e foi dado início ao desenvolvimento do trabalho.

Como visto anteriormente, um sistema de design é um produto, que por sua vez tem a intenção de servir outros produtos, que seriam os sistemas desenvolvidos pela instituição baseados nas diretivas criadas por esse sistema, é de responsabilidade do Sistema de Design (SD) possibilitar tanto a prototipação rápida, de baixa fidelidade, como a de alta fidelidade, assim como validar as evoluções feitas em sistemas preexistentes junto aos clientes. Sendo essas apenas algumas das responsabilidades desse produto.

4.1 Situação Atual Do Design no TCE-RN

A história do Tribunal de Contas do Estado começou oficialmente em 12 de janeiro de 1961, data oficial da sua criação (BARBOSA, 2020) O tribunal vem ao longo desses anos evoluindo sempre e gerenciando projetos cada vez mais complexos, agregando as responsabilidades de fiscalização e divulgação dos pareceres do poder legislativo Municipal. Contudo do ponto de vista do design o TCE-RN avançou pouco nos últimos anos. Apesar de possuir em suas políticas de desenvolvimento a obrigatoriedade da construção de um protótipo que precisa ser validado junto ao cliente que irá usar o software ou serviço, para saber se o mesmo tem suas necessidades atendidas essa prototipação atualmente é feita por meio da ferramenta de prototipação Pencil Project (Pencil) que possui várias limitações em relação a ferramentas de design mais modernas como demonstrado na tabela 2 Vantagens Figma vs Pencil.

Em 2019 o TCE-RN iniciou o desenvolvimento de uma nova biblioteca de componentes para construção de interfaces web. Essa biblioteca é construída sobre uma plataforma de

desenvolvimento em Javascript chamada Angular, Javascript por sua vez é uma linguagem de Programação de propósito geral, multiparadigma que juntamente com as tecnologias HTML (HyperText Markup Language), uma linguagem de marcação de hipertexto e CSS (Cascade Style Sheet), o mecanismo de acionamento de estilos de interpretadores web representam os 3 pilares fundamentais usados no desenvolvimento de sistemas web modernos.

Para a construção do seu front-end, que são sistemas com a qual os usuários interagem diretamente, o tribunal usa tecnologias modernas e em constante evolução, contudo para a prototipação, o Pencil oferece poucas vantagens em relação a maioria das ferramentas citadas no Capítulo 2 como por exemplo testes de usabilidade ou a presença de animações ou transições que aproximem o usuário da experiência final do uso do software em Desenvolvimento.

Tabela 2 – Comparativo Figma vs Pencil

Características	Figma	Pencil
Facil de usar	sim	sim
Colaborativo	sim	não
Animação	sim	não
Arrastar e soltar	sim	sim
Prototipo de Software	sim	não
Prototipação de Interfaces	sim	sim
Prototipação de UX	sim	não
Teste de Usabilidade	sim	não
Controle de Versao	sim	não

Fonte: (G2, 2020)

O fluxo de trabalho até o momento para o desenvolvimento de protótipos consiste em copiar um projeto preexistente e utilizar a maior quantidade possível de elementos de layout desse documento do Pencil. No entanto como foi dito anteriormente o tribunal passou a evoluir a biblioteca de componentes de front-end dos seus sistemas, e por não possuir um sistema de versionamento o Pencil não é capaz de refletir essas modificações. Também não é permitido que o usuário teste o fluxo da experiência de uso dos sistemas com o usuário. Uma lista de comparações entre as funcionalidades do Figma e do Pencil pode ser encontrada na tabela 2

4.2 Reconstrução Do Modelo Existente

O ponto de partida foi recriar modelos dos elementos preexistentes nos layouts de sistemas do TCE-RN. Foram favorecidos os sistemas que implementassem a biblioteca de componentes recém desenvolvida em detrimento as demais, pois estes expressam de

maneira mais acurada as intenções do tribunal quanto a aparência e usabilidade a serem experimentadas pelos usuários num futuro próximo.

Executando uma instância de exemplo da biblioteca de componentes criadas em Angular e usando o navegador Brave, foi possível utilizar uma das funcionalidades desse navegador, o CSS Overview, para inspecionar as características do projeto e a situação atual da biblioteca, tornando possível realizar o levantamento dos diferentes aspectos dos componentes como por exemplo a quantidade de cores diferentes presentes nas páginas, onde essas cores eram usadas, quais os códigos hexadecimais referente a cada cor, quantos e quais os breakpoints de tamanho das mídias, e outras informações relevantes. Um exemplo de inspeção de página pode ser encontrado na figura 4 Análise da biblioteca de componentes existente no TCE-RN.

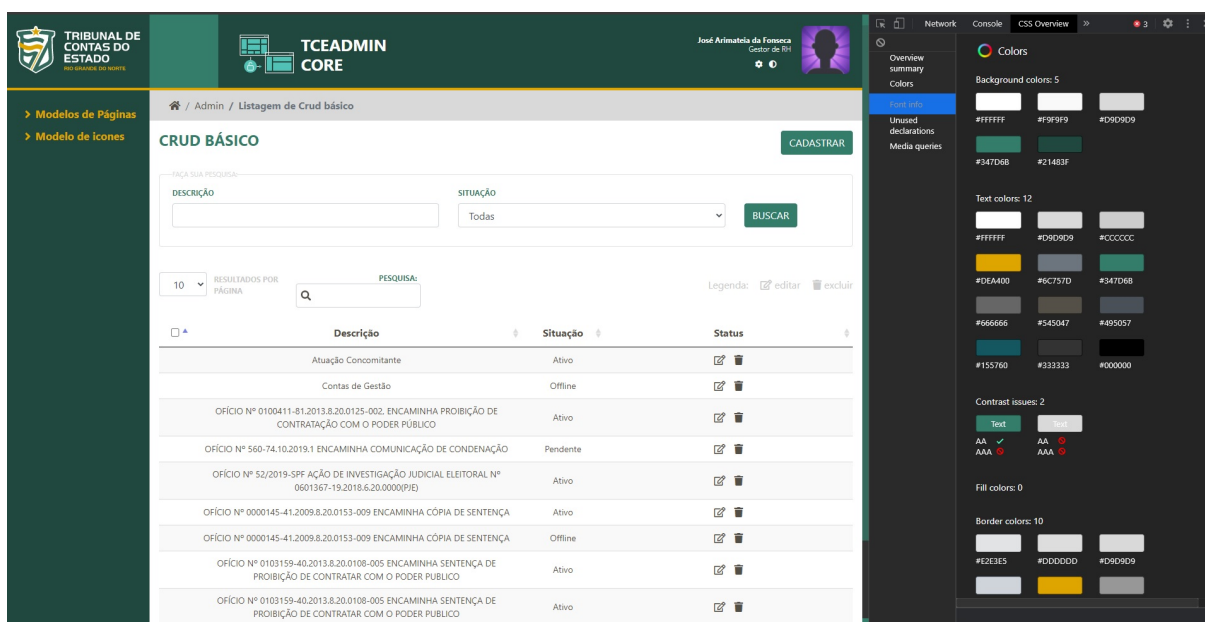


Figura 2 – Análise da biblioteca de componentes existente no TCE-RN.

De posse dos elementos fundamentais que já estavam sendo usados em conjunto com as métricas que os mesmos possuíam o passo seguinte foi a construção dessa instância atual na ferramenta Figma proposta pelas vantagens citadas no capítulo 2. Em um primeiro momento foi utilizada a metodologia atomica de design para o desenvolvimento desse modelo inicial "In the natural world, atomic elements combine together to form molecules. These molecules can combine further to form relatively complex organisms"(FROST, 2020). Construir elementos mais complexos a partir de elementos mais simples associado a uma ação facilitada por uma funcionalidade do Figma de permitir a construção de "componentes mestres", estruturas das quais podem ser criadas instâncias e que quando edições são realizadas no componente mestre as mesmas são herdadas para todos os elementos copiados a partir deste, o que permitiu reduzir a carga de trabalho, pois mudanças e melhorias nos elementos não precisavam ser replicadas em elementos acima na construção hierárquica

dos componentes.

O intuito desta primeira abordagem era o de separar os elementos existentes em elementos mais fundamentais que pudessem ser identificados como reaproveitáveis por componentes com hierarquias e complexidades mais elevadas. Esta separação segue uma sequência intuitiva de agrupamentos por átomos, moléculas e organismos, em que, átomos constituem os blocos fundamentais usados por moléculas, sendo estas estruturas um pouco mais complexas como por exemplo um campo de entrada de texto ou um botão, que por sua vez agrupados compõem organismos, estruturas ainda mais complexas que englobam múltiplos átomos e moléculas. Um exemplo dessa Composição e reaproveitamento pode ser visto na figura 3

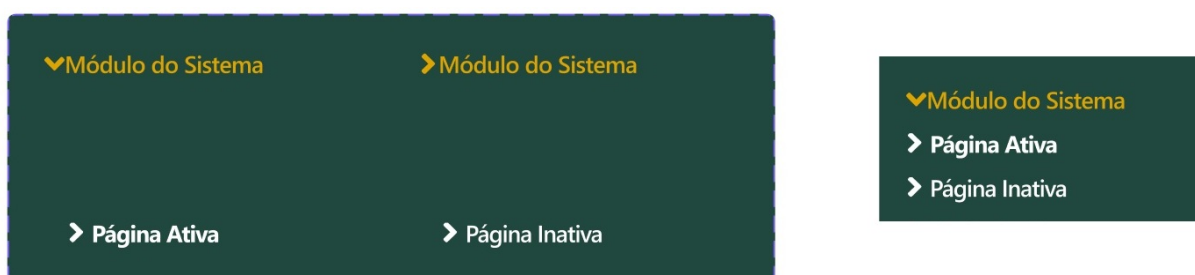


Figura 3 – Exemplo de Composição de elementos.

- *Atoms are the basic building blocks of all matter. Each chemical element has distinct properties, and they can't be broken down further without losing their meaning. (Yes, it's true atoms are composed of even smaller bits like protons, electrons, and neutrons, but atoms are the smallest functional unit.)*
- *Molecules are groups of two or more atoms held together by chemical bonds. These combinations of atoms take on their own unique properties, and become more tangible and operational than atoms.*
- *Organisms are assemblies of molecules functioning together as a unit. These relatively complex structures can range from single-celled organisms all the way up to incredibly sophisticated organisms like human beings.*

(FROST, 2020)

4.3 Teste de Execução e Problemas Encontrados

Após a criação do modelo existente o conjunto de componentes criado com esse modelo foi utilizado para realizar a prototipação de um projeto de desenvolvimento de sistema em andamento no TCE-RN com o intuito de validar com o clientes desse sistema, que era a Diretoria de Atos de Pessoal, as interfaces a serem concluídas junto com o sistema.

Com a ajuda da biblioteca de componentes as telas do sistema foram criadas em algumas horas possuindo alta fidelidade e permitindo a experimentação dos clientes

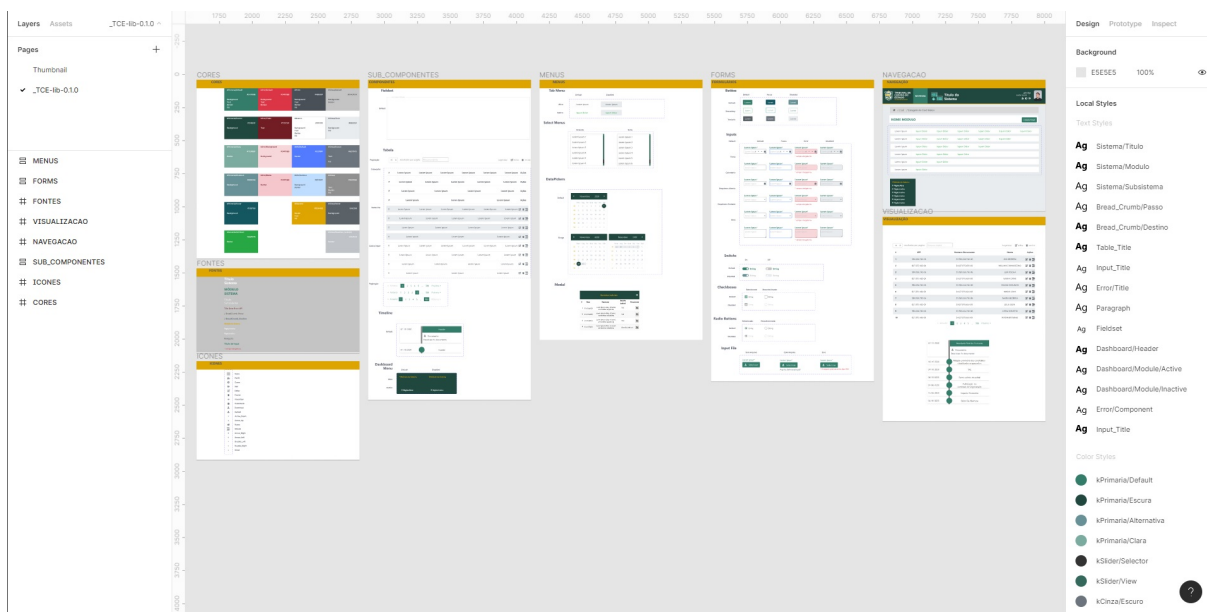


Figura 4 – Resultado Da primeira iteração do Figma.

para testar os diferentes fluxos do sistema. Contudo, os testes de usabilidade não foram feitos de maneira extensiva e múltiplas alterações foram solicitadas ou descobertas outras necessidades durante a etapa em que o desenvolvimento da aplicação havia sido iniciado. Essas alterações passaram por novas fases de validação por meio de prototipagem e ocorreram de forma mais rápida do que o desenvolvimento tradicional em que as alterações seriam feitas diretamente na aplicação no front para serem testadas e validadas junto ao cliente.

Durante essas refatorações foram encontrados alguns problemas com a escolha da arquitetura da biblioteca. A separação dos componentes usando o modelo atômico por si só e a estrutura de separação adotada não tornava a biblioteca de componentes intuitiva de ser usada. A nomenclatura dos componentes também poderia ser melhorada de modo a favorecer localização e facilitar o uso desses componentes. Por sua vez, a biblioteca de componentes criadas pelo tribunal usando Angular tinha uma amplitude muito grande no gradiente de cinza e possuía elementos que poderiam apresentar confusão dependendo da sua prioridade e estados como botões terciários habilitados e desabilitados que não comunicavam bem a sua função pois misturavam as tonalidades com seus estados secundários.

4.4 O Nascimento De Um Proto Sistema De Design

A medida que se avançava o desenvolvimento deste trabalho, maior foi o contato com sistemas de design em estágios de desenvolvimento mais maduros ou consolidados como fonte de estudo e referência, então, notou-se que a complexidade e o esforço despendido para a construção de um sistema de design mais abrangente, contendo não só uma biblioteca

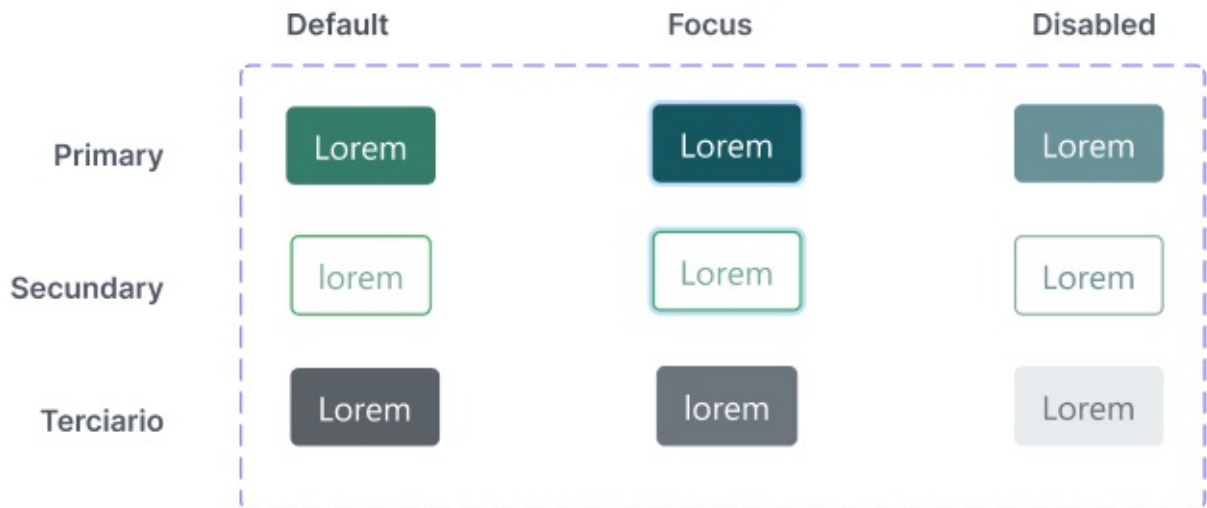


Figura 5 – Diferentes estados de botões da biblioteca de componentes do TCE-RN.

de componentes para prototipação mas também a sua implementação eram muito maiores do que eram possíveis obter da equipe dedicada a este trabalho. Surgiu então a ideia de se criar as diretrizes para a evolução da biblioteca de componentes atualmente existentes no TCE-RN, melhorando refinando o estilo dos elementos atualmente presentes para melhor comunicar a informação com os usuários, diminuindo as variações existentes para obter biblioteca com maior conformidade.

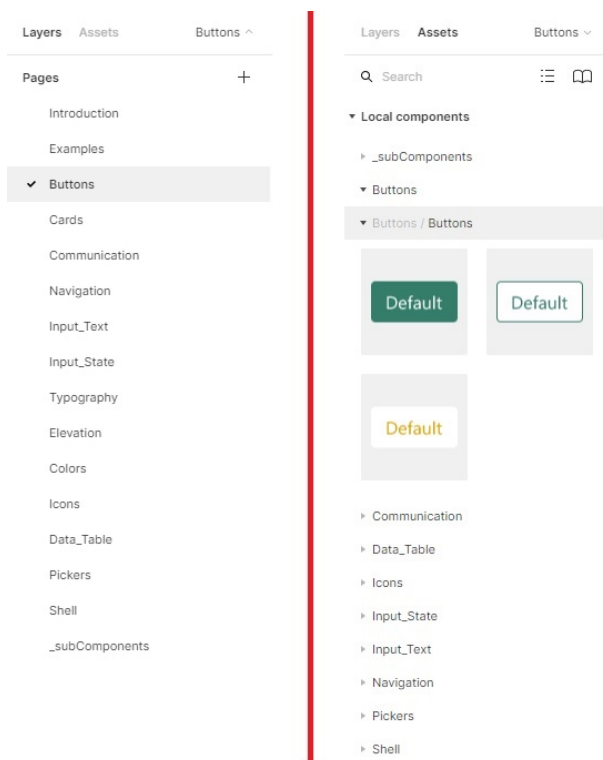


Figura 6 – Separação dos Componentes

Os elementos adicionados no Figma são classificados segundo a própria ferramenta como frames, groups e objects, enquanto os dois primeiros representam estruturas organizacionais o último compreende um outro conjunto de elementos como por exemplo textos, formas e imagens. A associação desses elementos é usada para criar componentes, que são as entidades mais importantes do Figma usadas para representar os itens do nosso SD.

Um arquivo do Figma separa a estrutura de interação com os elementos em dois grandes grupos, layers, onde se encontram páginas que contém elementos que podem ser trabalhados, e assets, que contém apenas elementos mestres para importação em páginas na forma de instâncias dos mestres. Em assets também encontramos quais

elementos estão disponíveis para serem exportados para a biblioteca de componentes.

No fim, valendo-se de uma funcionalidade do Figma de tornar páginas com nomes iniciados com sublinhado próprias apenas para o uso interno o projeto seguiu a seguinte estruturação: `_subComponents` contém elementos herdados da arquitetura atômica inicialmente desenvolvida, esta página contém elementos que são fundamentalmente utilizados em outras estruturas e em tese não deveriam ter seus elementos subdivididos nem usados independentemente. Na sequência cada página contendo elementos em um frame respectivo a suas versões web sendo esses `Buttons`, `Cards`, `Communication`, `Navigation`, `Input_Text`, `Input_State`, `Data_Table`, `Pickers`, `Shell Colors`, `Typography`, `Icons`. Deste modo chegou-se em uma boa estruturação dos componentes para exportação da biblioteca e viabilizando uma experiência fácil e intuitiva no desenvolvimento de protótipos

Adicionalmente uma página contendo exemplos de construção de telas foi adicionada a biblioteca para melhor guiar o desenvolvimento de templates por parte dos utilizadores do sistema de design assim como uma página introdutória, contendo explicações sobre a separação do documento assim como os meios para utilizar a biblioteca. Por último, uma página com um guia contendo os estilos e como aplicar diferentes elevações a elementos do front-end para iniciar o desenvolvimento de elementos para dispositivos móveis.

5 RESULTADOS OBTIDOS

Ao final deste trabalho obtivemos a construção de um sistema de design usando a ferramenta Figma. Este SD não só contém os principais elementos usados atualmente na biblioteca de front-end, escrita em Angular, pelo Tribunal de contas do Estado do Rio Grande Do Norte como também inclui algumas sugestões de melhoria e evolução para esta biblioteca, permitindo que com algumas alterações a mesma possa manter uma identidade visual mais consistente entre seus sistemas a serem desenvolvidos no futuro e permita aos usuários manter uma experiência mais uniforme ao transitar entre sistemas mantidos ou desenvolvidos pelo TCE-RN.

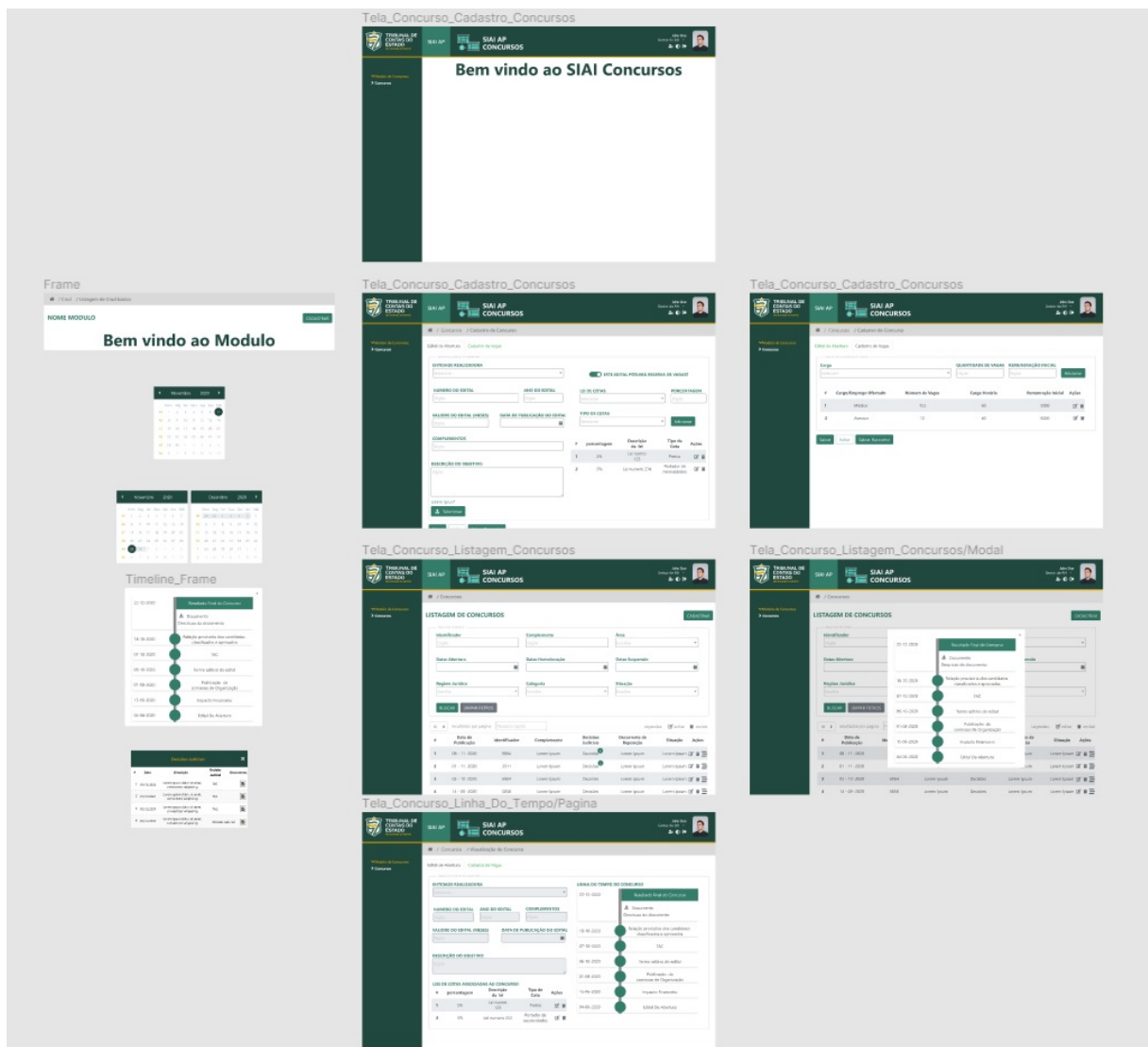


Figura 7 – Resultado da prototipação do sistema SIAI Concursos.

Além disso a biblioteca de componentes é graças ao uso do Figma versionável dentro da própria ferramenta, permitindo que evoluções na identidade visual não se tornem

incompatíveis com sistemas existentes em situação legada. O versionamento dessas versões é feito por meio da publicação de alterações em um padrão muito parecido com o utilizado pela ferramenta de versionamento de código usada pelo tribunal, o GitLab.

O Sistema de Design desenvolvido permite por meio do uso de instâncias de componentes preexistentes que sejam criados protótipos de tela de alta fidelidade de maneira rápida e eficiente, permite também que os protótipos sejam testados por meio de um fluxo de telas associados a ações disparadas por componentes, garantindo durante a fase de levantamento de requisitos, que usuários sintam uma experiência mais próxima da real, seja testando um sistema ou uma nova funcionalidade em desenvolvimento.

Além disso, a biblioteca de componentes pode ser compartilhada publicamente, permitindo a qualquer cidadão com interesse de desenvolver alguma funcionalidade ou sistema que consuma dados fornecidos pelo TCE-RN mantenha a mesma conformidade com a identidade visual dos sistemas produzidos pelo Tribunal de Contas.

Contudo, o sistema carece de testes de uso mais extensivos por parte dos desenvolvedores do TCE-RN não só para assegurar e cobrir eventuais falhas mas também para assegurar o treinamento dos desenvolvedores e líderes de projetos no uso de protótipos mas também na cultura de prototipação antes do desenvolvimento, de modo a permitir que mudanças em fases iniciais da construção e testes de sistemas e funcionalidades causem um impacto menor no tempo de produção dos mesmos.

O produto resultado deste trabalho pode ser acessado pelo link público a seguir: [TCE-RN/Design System](#).

6 CONSIDERAÇÕES FINAIS

Neste capítulo trago minhas considerações finais sobre o que foi alcançado e propostas para o amadurecimento do que foi desenvolvido.

6.1 Conclusões

É fato que um sistema de design é um documento vivo e em evolução permanentemente e com este trabalho foi possível realizar o desenvolvimento de um SD com a sobriedade, a objetividade e o senso de propósito pertinente ao Tribunal de Contas do Estado do Rio Grande do Norte, sendo assim, é considerado pelo autor que as intenções e objetivos do trabalho foram alcançados com êxito, evidentemente não sem sacrifícios em relação as escolhas adotadas, pois o desenvolvimento de um sistema de design não é algo que se conclui, mas que se abandona, pois o fim dos prazos chegam e o produto precisa ser entregue de um jeito ou de outro, mas sempre passível de melhorias, críticas e reinterpretações. Neste trabalho não foi diferente e apesar do desejo de continuar a evolução do que vinha-se desenvolvendo a entrega tinha que ser efetuada.

6.2 Melhorias

Dentro das muitas melhorias possíveis, planejadas ou simplesmente encontradas podemos destacar, acima de tudo, a falta de comunicação direta com a biblioteca de componentes de front-end tce-ng-lib. Era esperado que as alterações propostas fossem incluídas nos componentes de front-end e a nomenclatura e descrição das duas poderiam ser melhoradas para entrar em conformidade, tornando a busca e compreensão facilitadas tanto para os designers quanto para os desenvolvedores, referenciando os componentes e as classes de estilo pelos mesmos nomes tanto no angular quanto no figma. Outro grande problema foi a ausência de um conjunto de componentes para dispositivos móveis, que foi pensado em se fazer inicialmente mas o estudo e preparatório e as sucessivas revisões do trabalho consumiram muito tempo e o receio de não conseguir terminar os documentos a tempo inibiram o início desta fase do trabalho.

REFERÊNCIAS

BARBOSA, I. R. Tribunal de contas do estado do rio grande do norte. 2020. Disponível em: <<https://irbcontas.org.br/associado/tribunal-de-contas-do-estado-do-rio-grande-do-norte/>>.

BARRETO, W. da S. Tribunais de contas: conceito, funções, competências, histórico, natureza jurídica e acórdão do tcu em anexo. 2016. Disponível em: <<https://ambitojuridico.com.br/cadernos/direito-constitucional/tribunais-de-contas-conceito-funcoes-competencias-historico-natureza-juridica-e-acordao-do-tcu-em-anexo/>>.

CURTIS, N. A design system isn't a project. it's a product, serving products. 2016. Disponível em: <<https://medium.com/eightshapes-llc/a-design-system-isn-t-a-project-it-s-a-product-serving-products-74dcffef935>>.

FEDERAL, G. Sobre o design system. 2019. Disponível em: <<https://www.gov.br/governodigital/pt-br/transformacao-digital/ferramentas/design-system>>.

FEDERAL, G. Sobre o design system. 2019. Disponível em: <<https://dsgov.estaleiro.serpro.gov.br/ds/introducao/sobre>>.

FROST, B. The atomic design methodology. 2020. Disponível em: <<https://atomicdesign-bradfrost.com/chapter-2/>>.

G2. Compare figma and pencil. 2020. Disponível em: <<https://www.capterra.com/prototyping-software/compare/175027-176481/Figma-vs-Pencil-Project>>.

GITLAB. Gitlab design system. 2020. Disponível em: <<https://design.gitlab.com/get-started/principles>>.

GONZALEZ, G. O que é, e porque criar um design system? 2019. Disponível em: <<https://uxdesign.blog.br/design-system-90036c034225>>.

JURÍDICOS, S. para A. Lei nº 12527. 2011. Disponível em: <http://www.planalto.gov.br/ccivil_03/_Ato2011-2014/2011/Lei/L12527.htm>.

JURÍDICOS, S. para A. decreto nº 8777. 2016. Disponível em: <http://www.planalto.gov.br/ccivil_03/_Ato2011-2014/2011/Lei/L12527.htm>.

MICROSOFT. Fluent design system. 2020. Disponível em: <<https://www.microsoft.com/design/fluent/#/>>.

NORTE, T. D. C. D. E. D. R. G. D. Relatório de atividades do tce-rn. 2019. Disponível em: <http://www.tce.rn.gov.br/as/PlanejamentoEstrategico/RelatorioTrimestralTribunal/Relatorio_Atividades_1_trimestre_2019.pdf>.

NORTE, T. de Contas do Estado do Rio Grande do. Portal dos dados abertos tce/rn. 2020. Disponível em: <<http://apidadosabertos.tce.rn.gov.br/>>.

RICHARDSON, K. et al. Presentation on theme: "comp 5620/6620/6626 chapter 8: Design, prototyping and construction"— presentation transcript. 2015. Disponível em: <<https://slideplayer.com/slide/7666413/>>.

SAARINEN, K. Building a visual language: Behind the scenes of our airbnb design system. 2016. Disponível em: <<https://medium.com/airbnb-design/building-a-visual-language-behind-the-scenes-of-our-airbnb-design-system-224748775e4e>>.