

UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ  
CAMPUS CURITIBA  
ENGENHARIA DE COMPUTAÇÃO

MARCELLE REIS PIRES

**SISTEMA DE GERENCIAMENTO DE APENADOS EM  
REGIME DE PRESTAÇÃO DE SERVIÇOS À COMUNIDADE**

TRABALHO DE CONCLUSÃO DE CURSO

CURITIBA

2021

MARCELLE REIS PIRES

**SISTEMA DE GERENCIAMENTO DE APENADOS EM  
REGIME DE PRESTAÇÃO DE SERVIÇOS À COMUNIDADE**

Trabalho de Conclusão de Curso do curso de Engenharia de Computação do Departamento Acadêmico de Informática e Departamento Acadêmico de Eletrônica da Universidade Tecnológica Federal do Paraná como requisito parcial para obtenção do grau de Engenheiro da Computação.

Orientador: Prof. Msc. Christian Mendes

**CURITIBA**

**2021**

**MARCELLE REIS PIRES**

**SISTEMA DE GERENCIAMENTO DE APENADOS EM REGIME DE  
PRESTAÇÃO DE SERVIÇOS À COMUNIDADE**

Trabalho de Conclusão de Curso de Graduação  
apresentado como requisito para obtenção do título  
de Bacharel em Engenharia de Computação da  
Universidade Tecnológica Federal do Paraná  
(UTFPR).

Data de aprovação: 28 de junho de 2021

---

**CHRISTIAN CARLOS DE SOUZA MENDES**

Mestrado em Engenharia Elétrica e Informática Industrial pela Universidade Tecnológica Federal do  
Paraná  
Universidade Tecnológica Federal do Paraná (UTFPR)

---

**ANA PAULA MYSZCZUK**

Doutorado em Direito pela Pontifícia Universidade Católica do Paraná  
Universidade Tecnológica Federal do Paraná (UTFPR)

---

**LEANDRO BATISTA DE ALMEIDA**

Doutorado em Informática pela Universidade Federal do Paraná  
Universidade Tecnológica Federal do Paraná (UTFPR)

**CURITIBA**

**2021**

## **AGRADECIMENTOS**

Quero agradecer primeiramente a Deus, pela força, sabedoria, capacitação e sustentação diária.

Dedico este trabalho à minha família, que apesar da distância, tem sido uma sólida base de apoio. Especiais agradecimentos aos meus pais, Marcelo Assis Pires e Sylvia Agum dos Reis Pires, por todo amor e dedicação em minha criação e pelo apoio e incentivo em toda a minha vida. Aos meus irmãos, Marina Reis Pires e Murilo Reis Pires, pelas boas energias e entusiasmo, com certeza vocês tornaram a caminhada mais leve. À minha tia e amiga, Suelen Agum dos Reis, pelas palavras de incentivo, pelos conselhos e confortos nos momentos difíceis. E ao meu namorado, Armstrong Godoy Junior, pela nossa cumplicidade, por todos os abraços nos momentos de dificuldades, carinho e paciência nestes anos. Amo muito vocês.

Quero agradecer também toda equipe do Laboratório de Projetos de Tecnologia da Informação (LAPTI) pelo acolhimento. Agradeço a Bruna Araújo Pinheiro, que aceitou fazer parte deste projeto, implementando o aplicativo para dispositivos móveis, e que se tornou uma companheira neste desafio. Agradeço também meu professor orientador, Christian Mendes, por me aceitar como sua aluna e pela dedicação em me ensinar e orientar durante o tempo em que trabalhamos juntos.

Por último, mas não menos importante, também quero agradecer aos muitos amigos que fiz durante estes anos de curso. Amigos estes que levarei para toda a vida e que, com certeza, foram peças primordiais para que eu chegasse até o final desta etapa. A cada um de vocês, meus sinceros agradecimentos e desejos de sucesso.

## RESUMO

PIRES, Marcelle R. SISTEMA DE GERENCIAMENTO DE APENADOS EM REGIME DE PRESTAÇÃO DE SERVIÇOS À COMUNIDADE. 129 f. Trabalho de Conclusão de Curso – Engenharia de Computação, Universidade Tecnológica Federal do Paraná. Curitiba, 2021.

O Código Penal em vigência no Brasil prevê em seu artigo 43 penas alternativas em substituição às penas em prisões tradicionais, as chamadas de Penas Restritivas de Direito. A prestação de Serviço Comunitário, é uma dessas penas alternativas e pode ser aplicada por qualquer juiz mediante análise e enquadramento das características do réu e de seu crime nas regras pré estabelecidas em lei. Entretanto, vê-se uma baixa atribuição da mesma no dia a dia dos tribunais. Apesar da sua comprovada eficiência em ressocializar e reintegrar os indivíduos à sociedade, que são princípios que norteiam o código penal brasileiro, as dificuldades em gerenciar e monitorar as atividades prestadas pelos apenados se apresentam como uma barreira ainda a ser superada para que de fato essa modalidade de pena seja largamente aplicada. Visando fomentar a aplicação de tal modalidade de pena, este trabalho de conclusão de curso (TCC) propõe um sistema para a Justiça Federal brasileira que facilita o gerenciamento das penas e dos apenados que porventura venham a se enquadrar no cumprimento de penas através de serviços comunitários. Este gerenciamento se dará por meio de um portal *web* onde as equipes técnicas e juízes responsáveis poderão gerir as atividades cadastradas pelas entidades envolvidas, monitorar as atividades sendo executadas pelos apenados e gerenciar dados de apenados e entidades. Será por meio desse mesmo portal que as entidades poderão divulgar suas atividades para os apenados automaticamente, o que torna, portanto, esse sistema um canal de comunicação mais eficiente entre as partes envolvidas. Além disso, através de uma aplicação para dispositivos móveis, os apenados realizarão *check-ins* e *check-outs* de suas atividades referentes à pena e todos os dados referentes à atividade (local, horário, etc) serão apresentados no portal, possibilitando o monitoramento em tempo real. Dessa forma, o sistema se apresenta como um meio bastante eficiente e com excelentes possibilidades de auditoria e rastreabilidade, permitindo uma economia considerável de recursos do governo federal. Sobre a parte técnica, este TCC se propõe, exclusivamente, à implementação do *back-end* da aplicação, constituído pelo Banco de Dados e *Web service*, e portal *Web*. Justos estes elementos representam o sistema que será utilizado pelas entidades, juízes e equipes técnicas da Justiça Federal, e possibilitará a manipulação e fornecimento de dados para o aplicativo para dispositivos móveis

utilizado pelos apenados. Este aplicativo, apesar de ser parte importante do sistema, está fora do escopo do presente trabalho, mas está sendo desenvolvido em paralelo pelas equipes do Laboratório de Projetos de Tecnologia da Informação (LAPTI).

**Palavras-chave:** Poder Judiciário, Serviço Comunitário, Automatização, Sistemas de Informação

## ABSTRACT

PIRES, Marcelle R. MANAGEMENT SYSTEM OF INMATES UNDER THE PROVISION OF COMMUNITY SERVICES. 129 f. Trabalho de Conclusão de Curso – Engenharia de Computação, Universidade Tecnológica Federal do Paraná. Curitiba, 2021.

The Penal Code in force in Brazil provides in its article 43 alternative penalties to replace the penalties in traditional prisons, the so-called Restrictive Penalties of Law. The provision of Community Service is one of these alternative penalties and can be applied by any judge by analyzing and framing the characteristics of the defendant and his crime in the pre-established rules by law. However, there is a low allocation of it in the day-to-day of the courts. Despite its proven efficiency in resocializing and reintegrating individuals into society, which are principles that guide the Brazilian penal code, the difficulties in managing and monitoring the activities provided by inmates present themselves as a barrier yet to be overcome for this modality penalty is widely enforced. Aiming to encourage the application of this type of penalty, this course conclusion work proposes a system for the Brazilian Federal Justice that facilitates the management of penalties and inmates who may fall under the terms of serving sentences through community services. This management is collected through a web portal where the technical teams and judges carry out the activities registered by the entities involved, monitor the activities being performed by the inmates and manage data from inmates and entities. It will be through this same portal that the entities will automatically disclose their activities to the inmates, which makes, therefore, this system a more efficient communication channel between parties involved. In addition, through an application for mobile devices, inmates will carry out check-ins and check-outs of their activities related to the penalty and all data related to the activity (location, time, etc.) will be presented on the portal, enabling real time monitoring. Thus, the system presents itself as a very efficient, with possibilities for auditing and traceability, allowing for an attenuating saving of federal government resources. Regarding the technical part, this work offers itself, exclusively, to the implementation of the back-end application, constituted by the Database and Web service, and a Web portal. These elements represent the system that will be used by the entities, judges and technical teams of Justice Federal, and will enable the manipulation and provision of data for the application for mobile devices used by inmates. This application, despite being an important part of the system, is outside the scope of this work, but it is being developed

in parallel by the teams of the Information Technology Project Laboratory (LAPTI).

**Keywords:** Judicial power, Community Services, Automation, Information Systems

## LISTA DE FIGURAS

FIGURA 1	– Arquitetura Tradicional de desenvolvimento <i>web</i> .....	37
FIGURA 2	– Arquitetura Moderna de desenvolvimento <i>web</i> .....	38
FIGURA 3	– Fluxo de funcionamento de um <i>framework</i> .....	43
FIGURA 4	– Diagrama de Casos de uso do gerenciador da Justiça Federal .....	62
FIGURA 5	– Estrutura Geral do Projeto de Gerenciamento de Apenados .....	69
FIGURA 6	– Diagrama Entidade-Relacionamento do Banco de Dados .....	74
FIGURA 7	– <i>Mockup</i> da Tela de Login .....	76
FIGURA 8	– <i>Mockup</i> da Tela inicial .....	76
FIGURA 9	– <i>Mockup</i> da Tela do <i>Dashboard</i> .....	76
FIGURA 10	– <i>Mockup</i> da Tela do Mapa .....	76
FIGURA 11	– <i>Mockup</i> da Tela dos Apenados .....	77
FIGURA 12	– <i>Mockup</i> da Tela das Entidades .....	77
FIGURA 13	– <i>Mockup</i> da Tela de Relatórios .....	77
FIGURA 14	– <i>Mockup</i> da Tela de Logs .....	77
FIGURA 15	– Tela inicial para um usuário entidade do sistema .....	80
FIGURA 16	– Tela inicial para um usuário juiz do sistema .....	81
FIGURA 17	– Tela inicial para um usuário técnico da JF do sistema .....	82
FIGURA 18	– Tela de Logs .....	83
FIGURA 19	– Tela de Dados de um apenado .....	85
FIGURA 20	– Exemplo do sistema de sugestão de Atividades .....	87
FIGURA 21	– Tela de Análise de Gráficos - Informações Gerais .....	88
FIGURA 22	– Relatórios Gráficos .....	89
FIGURA 23	– Tela de Login .....	91
FIGURA 24	– Fluxo de Esquecimento da senha .....	92
FIGURA 25	– Tela do <i>Dashboard</i> .....	93
FIGURA 26	– Tela do Mapa contendo as localizações dos apenados no momento da realização das atividades agendadas .....	94
FIGURA 27	– Tela dos Apenados .....	95
FIGURA 28	– Tela das Entidades .....	95
FIGURA 29	– Tela das Atividades - visão de um Juíz ou Técnico da JF .....	96
FIGURA 30	– Tela das Atividades - visão de uma entidade .....	97

## LISTA DE TABELAS

TABELA 1	– Exemplo de uso do sistema pela entidade para adicionar Nova atividade .....	63
TABELA 2	– Exemplo de utilização do sistema para aprovação de novas atividades adicionadas .....	64
TABELA 3	– Fluxo esperado de interação entre sistema e usuário quando realizada a ação de cadastro de nova atividade .....	66
TABELA 4	– Fluxo esperado de interação entre sistema e usuário quando realizada a ação de aprovação de uma nova atividade .....	68
TABELA 5	– Caso de Uso 1 - Cadastrar Apenados e entidades .....	107
TABELA 6	– Caso de Uso 2 - Visualizar Informações dos Apenados e entidades cadastrados .....	108
TABELA 7	– Caso de Uso 3 - Visualizar Mapa e <i>Dashboard</i> .....	108
TABELA 8	– Caso de Uso 4 - Visualizar todas as atividades cadastradas .....	109
TABELA 9	– Caso de Uso 5 - Gerar Relatórios .....	109
TABELA 10	– Caso de Uso 6 - Aprovar escalas agendadas .....	110
TABELA 11	– Caso de Uso 7 - Visualizar os gráficos da análise de dados do sistema .....	110
TABELA 12	– Caso de Uso 8 - Inativar Apenados e Entidades .....	111
TABELA 13	– Caso de Uso 9 - Ver logs do sistema .....	111
TABELA 14	– Caso de Uso 10 - Aprovar escalas finalizadas .....	112
TABELA 15	– Caso de Uso 11 - Adicionar Nova atividade .....	112
TABELA 16	– Caso de Uso 12 - Aprovar atividades cadastradas .....	113
TABELA 17	– Caso de Uso 13 - Visualizar as atividades cadastradas pela sua entidade .....	113
TABELA 18	– Fluxo Básico - Caso de Uso 1 .....	115
TABELA 19	– Fluxo Básico - Caso de Uso 2 .....	116
TABELA 20	– Fluxo Básico - Caso de Uso 3 .....	117
TABELA 21	– Fluxo Básico - Caso de Uso 4 .....	118
TABELA 22	– Fluxo Básico - Caso de Uso 5 .....	119
TABELA 23	– Fluxo Básico - Caso de Uso 6 .....	120
TABELA 24	– Fluxo Básico - Caso de Uso 7 .....	121
TABELA 25	– Fluxo Básico - Caso de Uso 8 .....	122
TABELA 26	– Fluxo Básico - Caso de Uso 9 .....	123
TABELA 27	– Fluxo Básico - Caso de Uso 10 .....	124

TABELA 28 – Fluxo Básico - Caso de Uso 11 .....	125
TABELA 29 – Fluxo Básico - Caso de Uso 12 .....	126
TABELA 30 – Fluxo Básico - Caso de Uso 13 .....	127

## LISTA DE SIGLAS

TCC	Trabalho de Conclusão de Curso
DEPEN	Departamento Penitenciário Nacional
Infopen	Levantamento Nacional de Informações Penitenciárias
CPB	Código Penal Brasileiro
PSC	Prestação de Serviços à Comunidade
JF	Justiça Federal
E-proc	Sistema de Transmissão Eletrônica de Atos Processuais da Justiça Federal da Primeira Região
SGDB	Sistema Gerenciador de Banco de Dados
ACID	Atomicidade, Consistência Isolamento e Durabilidade
XML	Extended Markup Language
HTTP	Hypertext Transfer Protocol
SOA	Service-Oriented Architecture
SOAP	Simple Object Access Protocol
REST	Representational State Transfer
W3C	World Wide Web Consortium
WSDL	Web Service Description Language
JSON	JavaScript Object Notation
URI	Uniform Resource Identifier
CRUD	<i>Create, Read, Update and Delete</i>
API	Application Programming Interface
LAPTI	Laboratório de Projetos de Tecnologia da Informação - Campus Curitiba
SPA	Single Page Applications
DOM	Document Object Model
CERN	Conseil Européen pour la Recherche Nucléaire
WWW	World Wide Web
AJAX	Asynchronous JavaScript and XML
HTML	Hypertext Markup Language
WHATWG	Web Hypertext Application Technology Working Group
CSS	Cascading Style Sheets
DER	Diagrama Entidade Relacionamento

## SUMÁRIO

<b>1</b>	<b>INTRODUÇÃO</b>	<b>13</b>
<b>2</b>	<b>PROBLEMÁTICA E MOTIVAÇÃO</b>	<b>17</b>
2.1	CONTEXTO ATUAL E SISTEMA E-PROC	17
2.2	DIGNIDADE DA PESSOA HUMANA, DIREITO PENAL E PENAS POR PRESTAÇÃO DE SERVIÇOS À COMUNIDADE	18
2.3	A IMPORTÂNCIA DO SISTEMA PARA APLICAÇÃO DAS PENAS POR PRESTAÇÃO DE SERVIÇOS À COMUNIDADE	21
<b>3</b>	<b>OBJETIVO</b>	<b>24</b>
3.1	OBJETIVO GERAL	24
3.2	OBJETIVOS ESPECÍFICOS	24
<b>4</b>	<b>TECNOLOGIAS EMPREGADAS</b>	<b>26</b>
4.1	BANCO DE DADOS	26
4.2	<i>WEB SERVICE REST</i>	31
4.3	DESENVOLVIMENTO <i>WEB</i>	36
<b>5</b>	<b>METODOLOGIA</b>	<b>46</b>
5.1	LEVANTAMENTO DOS PROBLEMAS E NECESSIDADES	47
5.1.1	Problemas	47
5.1.2	Necessidades	48
5.2	LEVANTAMENTO DE REQUISITOS	50
5.2.1	Requisitos do Portal <i>Web</i>	50
5.2.2	Requisitos do <i>Web Service</i>	59
5.3	DIAGRAMAS DE CASOS DE USO	61
5.3.1	Descrição dos casos de uso	63
5.3.2	Fluxo Básico dos casos de uso	64
5.4	ESTRUTURA GERAL DO SISTEMA	69
5.5	MODELAGEM	70
5.5.1	Modelo Entidade Relacionamento do Banco de Dados	71
5.5.2	<i>Mockups</i> do Portal <i>Web</i>	75
5.6	IMPLEMENTAÇÕES ESPECÍFICAS DO SISTEMA	77
5.6.1	Módulos de segurança - <i>Security by Design</i>	78
5.6.2	Geração de relatórios com informações dos apenados	84
5.6.3	Sistema Baseado em Regras - sugestão de atividades	85
5.6.4	Relatórios gráficos	88

<b>6</b>	<b>RESULTADOS FINAIS .....</b>	<b>90</b>
<b>7</b>	<b>CONSIDERAÇÕES FINAIS .....</b>	<b>98</b>
	<b>REFERÊNCIAS .....</b>	<b>103</b>
	<b>Apêndice A – CASOS DE USO DO SISTEMA .....</b>	<b>107</b>
	<b>Apêndice B – FLUXO BÁSICO DOS CASOS DE USO DO SISTEMA .....</b>	<b>114</b>
	<b>Apêndice C – EXEMPLO DE UM RELATÓRIO GERADO PELO SISTEMA .....</b>	<b>128</b>

## 1 INTRODUÇÃO

A partir das revoluções industriais, época em que ocorreu um crescimento exponencial da indústria, do comércio e da tecnologia, as cidades e vilas começaram a se organizar ao redor das fábricas e pólos industriais. Muito mais do que a necessidade de estarem próximos ao local de trabalho, segundo Lúcio Costa, importante arquiteto e urbanista francês que foi pioneiro da arquitetura no Brasil (COSTA, 1995):

Cidade é a expressão palpável da necessidade humana de contato, comunicação, organização e troca, numa determinada circunstância físico-social e num contexto histórico.

Em outros termos, a organização em cidades parte da necessidade do ser humano em ter convívio social e elas surgiram com o intuito de proteger os cidadãos que ali viviam. Muitos foram os benefícios colhidos a partir dessa nova organização, uma vez que a sociedade proporciona aos seus integrantes maiores facilidades de compra e venda, através de seus mercados e feiras, maior proximidade à recursos médicos, infraestrutura de sistemas de esgoto e saneamento básico, por exemplo.

Em contrapartida, aliados a esses benefícios, diversos problemas sociais surgiram, entre eles, a criminalidade. Assim como a organização em cidades, a criminalidade também é um fenômeno social. É um fato social já estudado e identificado desde o final do século XIX (COSTA, 1995). Segundo Carlos Gomes, a criminalidade é multiforme e cresce se infiltrando na estrutura social através das oportunidades existentes no espaço urbano. E, no Brasil, o fenômeno da criminalidade também se alastra pelas cidades, se concentrando em partes da mesma onde as

dificuldades econômicas são crescentes e o acesso aos seus benefícios são escassos. Esta fragmentação das cidades é um fenômeno resultante das relações sociais, mas que favorece ainda mais o crescimento da criminalidade (GOMES, 2005). Este, ainda hoje, é um problema no Brasil e tem como principal consequência o aumento desenfreado da população carcerária nacional, resultados da dificuldade em preservar os direitos fundamentais do cidadão em todo o território urbano.

Isso pode ser facilmente constatado pelos relatórios estatísticos divulgados pelo DEPEN (Departamento Penitenciário Nacional), que é o órgão do Ministério da Justiça e Segurança Pública responsável pela divulgação de relatórios que compilam estatísticas a respeito do sistema penitenciário brasileiro, o Infopen (Levantamento Nacional de Informações Penitenciárias). Estes relatórios estão disponíveis no site do departamento e trazem dados históricos da população carcerária e suas diversas atividades. No relatório divulgado em 2014, a população prisional do Brasil era composta por 622.202 indivíduos e o país ocupava o 4º lugar no ranking mundial dos países com a maior população carcerária do mundo (INFOPEN, 2014). Segundo as atualizações feitas no início do ano de 2020 (DEPEN, 2020), este número chegou, no segundo semestre de 2019, à marca de 748.009 indivíduos, o que levou o Brasil ao 3º lugar no ranking mundial com uma porcentagem de 0,354% da população encarcerada, ficando atrás apenas dos EUA com 2,1 milhões de indivíduos preso (0,63% da sua população total), e da China com 1,6 milhão presidiário, (0,114% da população total do país).

Em meio a esse contexto, é interessante ressaltar que boa parte da população carcerária do Brasil foi condenada por delitos com potencial para serem enquadrados nos requisitos para cumprimento de penas alternativas. Estas, inclusive, representam a longo prazo, um processo de mitigação para o problema da superlotação carcerária no Brasil. De fato, o Código Penal Brasileiro (CPB) em vigor prevê em seu Art. 43 (BRASIL, 1940a) penas alternativas às penas de privação de liberdade que são chamadas Penas Restritivas de Direito. Estas podem ser aplicadas por qualquer juiz, desde que o crime e o réu se enquadrem em requisitos descritos no Art. 44 do CPB (BRASIL, 1940b), que são os seguintes:

Art. 44. As penas restritivas de direitos são autônomas e substituem as privativas de liberdade, quando: (Redação dada pela Lei nº 9.714, de 1998)

I – aplicada pena privativa de liberdade não superior a quatro anos e o crime não for cometido com violência ou grave ameaça à pessoa ou, qualquer que seja a pena aplicada, se o crime for culposo;(Redação dada pela Lei nº 9.714, de 1998)

II – o réu não for reincidente em crime doloso; (Redação dada pela Lei nº 9.714, de 1998)

III – a culpabilidade, os antecedentes, a conduta social e a personalidade do condenado, bem como os motivos e as circunstâncias indicarem que essa substituição seja suficiente. (Redação dada pela Lei nº 9.714, de 1998)

Destaca-se entre estas a Prestação de Serviços à Comunidade (PSC) ou a entidades públicas, que consiste no cumprimento da pena através da prestação de serviços a qualquer instituição que queira receber um apenado. Estas penas estão em consonância com um dos nortes do código penal brasileiro, a ressocialização, visto que atribui à pena a possibilidade de colocar o indivíduo em sociedade e dar a ele a chance da reinserção à comunidade.

Tendo em vista o cenário apresentado, a seguinte questão é suscitada: Por que as penas alternativas, mesmo quando atendendo aos requisitos, não são aplicadas? Uma das possíveis respostas para este questionamento é a dificuldade de gerenciamento dos apenados e dos serviços a serem prestados. Sabe-se que, legalmente, este gerenciamento é de responsabilidade tanto do órgão de justiça quanto da instituição que receberá os serviços comunitários. Atualmente, o órgão de justiça responsável por este gerenciamento são as varas locais da Justiça Federal (JF), que disponibiliza equipes técnicas para atendimento presencial dos apenados, assim como os recursos necessários. Entretanto, este gerenciamento não é uma tarefa fácil considerando as dificuldades com relação às burocracias de comprovação das atividades e ao monitoramento dos apenados durante a execução das atividades.

Em vista disso, este trabalho se propõe a desenvolver a base principal de um sistema de gerenciamento que, com auxílio das equipes técnicas da JF, garantirá maior eficiência e efetividade na aplicação das penas por prestação de serviços à comunidade. Para isso, foi desenvolvido um *back-end*, que neste projeto são representados pelo

*web service* e banco de dados, e um *front-end*, representado pelo Portal *Web*. Tudo isso é apresentado em detalhes no decorrer deste documento, onde será exposto toda a plataforma *Web* onde os juízes e suas equipes terão acesso aos dados dos apenados e das instituições/entidades, através da qual será possível cadastrar novos indivíduos e gerenciar os serviços em desenvolvimento. Além disso, as instituições interessadas, após o devido cadastramento, também utilizarão esta plataforma para divulgar as atividades disponíveis dentro de suas dependências. Um aplicativo para dispositivo móvel será desenvolvido em paralelo a este trabalho utilizando o *back-end* criado, e será de uso exclusivo dos apenados. Através desse aplicativo, os apenados poderão acompanhar o andamento do cumprimento de sua pena, receber as ofertas de oportunidades de novos serviços e também utilizar para fazer um processo de *check-in* e *check-out* referente à sua participação em determinada atividade. Dessa forma, ele poderá comprovar, através de uma foto no local e sua geolocalização, que a atividade foi realizada. Este aplicativo está fora do escopo do presente trabalho, mas está sendo desenvolvido em paralelo pelas equipes do LAPTI.

## **2 PROBLEMÁTICA E MOTIVAÇÃO**

Neste capítulo é abordado com mais detalhes o contexto geral de trabalho das equipes da justiça federal, enfatizando o sistema utilizado atualmente e suas limitações. Além disso, é apresentado o amparo jurídico pelo qual este TCC se mostra importante para a sociedade brasileira como um todo.

### **2.1 CONTEXTO ATUAL E SISTEMA E-PROC**

O processo de comprovação das horas de serviços prestados por apenados, mencionado na seção anterior, acontece da seguinte forma: após prestada a atividade e comprovada através de documentação, esta deve ser entregue pessoalmente pelos apenados às varas locais da JF, onde as horas serão devidamente contabilizadas e os documentos, após terem sido verificados, arquivados no sistema E-proc (Sistema de Transmissão Eletrônica de Atos Processuais da Justiça Federal da Primeira Região) em planilhas eletrônicas ou, em sua maioria, em arquivos de documentos em papel.

O E-Proc consiste em um sistema de peticionamento processual que permite o encaminhamento de petições à Justiça Federal eletronicamente. Advogados e equipes técnicas da JF possuem cadastro no sistema e, através da sua autenticação, conseguem encaminhar os arquivos das petições. Este é o sistema majoritariamente utilizado nas varas da Justiça Federal para armazenamento das prestações de contas relacionadas às penas por Prestação de Serviço à comunidade. Entretanto, segundo relatos obtidos das equipes técnicas, este sistema não tem se mostrado eficiente no desempenho dessas funções, visto que não foi desenvolvido para estes fins e apresenta limitações.

Como fica claro na própria descrição do serviço, este sistema não foi criado especificamente para ser usado no controle das atividades desta modalidade de pena, uma vez que ele se trata apenas de um sistema de encaminhamento de processos em arquivos. A entrada de dados na plataforma se dá através de arquivos em formato PDF e, como na maioria dos casos os documentos apresentados são físicos, em papel, o seu armazenamento no sistema se dá através do processo de escaneamento, inviabilizando a pesquisa dentro dos arquivos. Ou seja, dependendo da informação procurada pela equipe técnica, um grande trabalho manual é necessário, sendo preciso abrir arquivo por arquivo e ler todos os dados a fim de encontrar o que se procura.

Em consequência às dificuldades de utilização do sistema, internamente em cada vara, é comum a utilização de planilhas eletrônicas para o controle dos apenados. As planilhas apresentam a facilidade de serem formatos já conhecidos pelas equipes técnicas, facilitando sua utilização, mas dificultam a coleta dos dados e estudo dos mesmos de forma organizada a níveis acima das varas locais, visto que seus formatos são livres e cada vara organiza seus dados da forma que a convenha. Diante disso, um levantamento estatísticos a nível estadual ou nacional são inviabilizados, pois os dados limitam-se a estatísticas regionais, de acordo com o domínio de cada vara, sem qualquer padronização ou centralização em um sistema de fácil acesso.

Além disso, para cumprir o processo mencionado anteriormente, as varas locais mantêm uma equipe técnica que trabalha no atendimento ao público e que é capacitada para a contabilização e documentação das horas. Ou seja, existe a necessidade de alocar uma carga horária da equipe administrativa para o desempenho dessas funções.

## 2.2 DIGNIDADE DA PESSOA HUMANA, DIREITO PENAL E PENAS POR PRESTAÇÃO DE SERVIÇOS À COMUNIDADE

Apesar de todas as dificuldades processuais e documentais que o Poder Judiciário e as entidades parceiras enfrentam com a utilização apenas do E-proc, as PSCs se mostram ainda vantajosas para todas as partes envolvidas: sociedade, apenado, poder judiciário e entidades parceiras.

O ordenamento jurídico brasileiro está pautado na Dignidade da Pessoa Humana, que consiste na garantia dos direitos mais fundamentais e vitais para o homem, como respeito e proteção. Sendo estes intrínsecos à existência do ser humano, independente de sua cor, raça, sexo ou idade (MOTTA, 2013).

Prevista no artigo 1º, III da Constituição Federal de 1988, a Dignidade Humana é princípio fundamental do nosso país e de acordo com Ingo Wolfgang Sarlet é considerada:

A qualidade intrínseca e distintiva reconhecida em cada ser humano que o faz merecedor do mesmo respeito e consideração por parte do Estado e da comunidade, implicando, neste sentido, um complexo de direitos e deveres fundamentais que assegurem a pessoa tanto contra todo e qualquer ato de cunho degradante e desumano, como venham a lhe garantir as condições existenciais mínimas para uma vida saudável, além de propiciar e promover sua participação ativa e corresponsável nos destinos da própria existência e da vida em comunhão com os demais seres humanos mediante o devido respeito aos demais seres que integram a rede da vida. (SARLET, 2011)

O Direito Penal brasileiro, ramo do Direito que regula o poder punitivo do Estado, possui como norteador o conceito de Dignidade da Pessoa Humana. O mesmo busca intervir apenas em casos de conflitos sociais e atualmente tem trabalhado os princípios da prevenção (ANGELIM, 2016). As penas são as formas de punição, previstas pelo Direito Penal, para todo cidadão que desobedecer alguma de suas normativas.

A finalidade da aplicação das penas é retribuição, reeducação e ressocialização. A retribuição consiste na punição do indivíduo. A forma justa, prevista em lei, em resposta à ação injusta, que foi o crime cometido. Entretanto, a crítica que se faz à este objetivo é a falta de uma finalidade prática na punição, se apresentando apenas como um vingança do Estado. A reeducação, por sua vez, é orientada pelas ideias preventivas que vem crescendo no Direito Penal. Busca-se reeducar o indivíduo, objetivando que o mesmo não cometa novos crimes e conseqüentemente esteja preparado para um retorno à sociedade. A consequência de um eficiente processo de reeducação do indivíduo atinge-se a finalidade de ressocialização e reintegração do indivíduo à sua comunidade.

Atualmente, as punições podem ser aplicadas através das penas privativas de liberdade ou das penas restritivas de direito, apresentadas no Art. 43 do CPB. As primeiras são divididas em três regimes: fechado, semiaberto e aberto, que se diferenciam pelo grau de liberdade do preso. As penas restritivas de direito, por sua vez, são penas que substituiriam as penas restritivas de liberdade e podem ser aplicadas a um determinado perfil de crime.

Acerca dos requisitos para aplicação das penas restritivas de direito, tem-se que, podem ser aplicadas a delitos de pequeno e médio potencial ofensivo. O Art. 44 do Código Penal determina que estas penas podem substituir penas restritivas de liberdade não superiores a quatro anos, quando não for cometida violência ou grave ameaça à qualquer pessoa, e, independente da pena aplicada, quando o crime é culposos, ou seja, quando não houve a intenção de cometer tal ação. Esta modalidade de pena se apresenta como uma alternativa aos efeitos traumáticos do cárcere em prisões.

Apesar das vantagens das penas restritivas de direito, as penas restritivas de liberdade ainda são, em sua maioria, aplicadas no Brasil. A prisão, em especial, se mostra ao longo dos anos ineficaz no cumprimento do objetivo de ressocialização dos indivíduos, como é mostrado no trecho a seguir extraído do livro Lições de Direito Penal:

A pena necessariamente deforma a personalidade, ajustando-se à subcultura da prisão. A reunião coercitiva de pessoas do mesmo sexo num ambiente fechado, autoritário, opressivo e violento, corrompe e avilta. [...] Conclui-se assim, que o problema da prisão é a própria prisão, que apresenta um custo social demasiadamente elevado (FRAGOSO, 2009)

É possível concluir que, a partir da citação destacada, para alguns casos específicos de crimes, a prisão pode ser mais prejudicial do que educativa e, por consequência, pode prejudicar o objetivo final que é a ressocialização. Esse é o caso de réus por crimes de transporte de drogas entre fronteiras, por exemplo. Os indivíduos que praticam este tipo de crime, na sua maioria, não caracterizam a raiz do problema e a prisão dos mesmos não se mostra eficiente na diminuição do tráfico de drogas entre

fronteiras. Pelo contrário, a prisão representa um processo de deformação de caráter que pode resultar em um indivíduo mais propenso a atividades criminosas no futuro.

Neste caso e em outros similares, a atribuição de penas alternativas por meio da prestação de serviços comunitários se apresenta como uma boa opção. O problema é que uma das grandes dificuldades na aplicação das penas restritivas de direito é a dificuldade de gerenciamento e controle. A lei que rege a pena por PSC prevê que é de responsabilidade das entidades e da equipe do poder judiciário garantir que as penas estão sendo cumpridas e de controlar as atividades dos apenados, como é possível verificar no trecho a seguir extraído do CPB:

Art. 150. A entidade beneficiada com a prestação de serviços encaminhará mensalmente, ao Juiz da execução, relatório circunstanciado das atividades do condenado, bem como, a qualquer tempo, comunicação sobre ausência ou falta disciplinar. (BRASIL, 1984)

Por conta das dificuldades de cumprimento desta cláusula por ambas as instituições, muitas vezes as PSCs deixam de ser aplicadas, mesmo sendo mais benéficas para a sociedade e para os apenados, além de estarem de acordo com as diretrizes do CPB. Mais ainda, às vezes, apesar de aplicadas, não são monitoradas como deveriam e resultam numa eficácia parcial da medida em razão das dificuldades administrativas das instituições responsáveis.

### 2.3 A IMPORTÂNCIA DO SISTEMA PARA APLICAÇÃO DAS PENAS POR PRESTAÇÃO DE SERVIÇOS À COMUNIDADE

Diante do que foi exposto, se houvesse um instrumento eficaz, que auxiliasse as equipes do judiciário e as entidades parceiras, as PSC's poderiam ser implementadas, executadas e controladas com maior tranquilidade e eficácia pelas equipes. Assim, as instituições melhorariam a sua atuação profissional, o apenado cumpriria a pena adequada nos termos da legislação penal brasileira, e a sociedade seria beneficiada com a aplicação de penalidades que efetivamente cumprem seu papel de ressocialização.

Portanto, o sistema computacional proposto neste projeto se mostra bastante benéfico a todos os envolvidos neste tipo de processo e poderá futuramente contribuir para o aumento da aplicação das Penas Restritivas de Direito.

Em relação à sociedade, a modalidade de PSC favorece a aplicação das penas por meio da aplicação de serviços comunitários, o que, além de trazer benefícios à comunidade através dos serviços prestados, também se mostram mais eficientes para a reinserção dos indivíduos à sociedade. Segundo Cavalcante, advogado especialista de Direito Penal (CAVALCANTE, 2016), a PSC assume uma função que as outras penas alternativas não conseguem, que é a reparação do dano causado à sociedade e a preocupação com a vítima. Além disso, ele destaca os satisfatórios resultados no que diz respeito à recuperação dos condenados para o processo de sociabilidade.

Aos apenados os benefícios se mostram através da maior viabilidade de aplicação da pena e também nos benefícios que o sistema trará para o gerenciamento da mesma. No modelo atual, os apenados precisam dispor de parte da sua semana para a prestação das horas cumpridas em horários comerciais, horários estes que poderiam estar sendo melhor utilizados em outros serviços, favorecendo assim o cumprimento da pena em um período mais curto.

As entidades que recebem os apenados, além de estarem ajudando a justiça na destinação das atividades dos indivíduos e, diretamente, auxiliando na ressocialização dos mesmos, são beneficiadas pelos serviços prestados, que são gratuitos por lei. Além disso, o sistema computacional proposto aqui também facilitará o contato dos apenados com as entidades, as quais terão maior segurança ao recebê-los em suas instituições, possibilitando inclusive o aumento das entidades interessadas em receber apenados.

As Varas locais da Justiça Federal se beneficiam na forma de disporem de uma menor quantidade de pessoal para atendimento dos apenados ou mesmo controle das atividades. Sendo assim, estes servidores poderão ser alocados e melhor aproveitados em outras áreas que demandem conhecimentos técnicos especializados.

Mediante a tudo que foi apresentado até aqui, tem-se que, apesar do Brasil

prever em suas leis as penas alternativas de direito, existem obstáculos de aplicação que ainda precisam ser superados. A PSC é uma modalidade deste tipo de pena, que está alinhada às finalidades do Código Penal e se apresenta como uma alternativa eficiente de punição.

As dificuldades em monitorar as atividades prestadas somada a falta de um sistema destinado a este controle, dificultam a aplicação da PSC. Portanto, o sistema proposto neste trabalho objetiva auxiliar o monitoramento das atividades prestadas pelos apenados, atribuindo eficiência e eficácia na gerência das penas. Consequentemente, a partir deste sistema, pretende-se alcançar um crescimento, a longo prazo, das aplicações das PSCs no país.

Portanto, a pergunta que impulsiona este projeto é a seguinte: como fazer com que a modalidade de pena PSC seja de fato aplicada aos crimes em que se enquadra? E, mais ainda, como possibilitar que o monitoramento e gerenciamento das mesmas seja realizado de forma eficiente e otimizada?

### 3 OBJETIVO

#### 3.1 OBJETIVO GERAL

Em vista do que foi apresentado até aqui, o objetivo geral deste trabalho é desenvolver um sistema informatizado estruturado para facilitar o gerenciamento de penas e apenados que se incluem na modalidade de PSCs. Este ambiente é composto por um *Web service*, um Banco de dados e também uma aplicação *Web*, que será utilizada pelas entidades, juízes e equipes técnicas da JF.

#### 3.2 OBJETIVOS ESPECÍFICOS

- Produzir um sistema com nível de segurança planejado desde a sua concepção, *security by design*. Serão aplicadas as técnicas de segurança: autenticação multi fator e sistema de autorização para usuários com diferentes perfis de autorização, na aplicação *Web*; autenticação via credenciais para acesso ao Banco de dados e ao *Web service*, além de um sistema de *log* implementado para fins de auditoria do sistema;
- Desenvolver uma aplicação *Web* por meio da qual a entidade interessada em disponibilizar uma vaga de atividade a ser cumprida por um apenado, a fim de contabilizar horas para sua pena, possa se cadastrar, adicionar as atividades e divulgá-las apropriadamente aos apenados;
- Produzir uma aplicação *Web* que possibilite ao Juiz e à sua equipe técnica realizar o cadastrar os apenados, cumprindo pena de PSC, acompanhar suas atividades em

desenvolvimento e gerenciar suas futuras escalas;

- Implementar um sistema especialista baseado em regras que sugere, através de um ordenamento prioritário, as atividades disponíveis no sistema, levando em consideração o perfil de habilidades e crimes do apenado;
- Gerar relatórios relacionados aos apenados, dados pessoais, dados da pena e contendo um histórico de atividades, concluídas e agendadas;
- Gerar relatórios periódicos sobre características dos apenados e entidades registrados no sistema e suas atividades. Estes relatórios auxiliariam as equipes técnicas na tomada de decisão e otimização de recursos das varas locais, visto que traçam um perfil da região onde atuam;

## 4 TECNOLOGIAS EMPREGADAS

Este capítulo apresenta as tecnologias empregadas neste trabalho e justifica a escolha de cada uma delas a partir das especificidades do projeto.

### 4.1 BANCO DE DADOS

Os bancos de dados estão presentes em diversas situações do dia a dia no mundo atual. Desde um saque no banco até mesmo uma compra no supermercado resultam, em sua maioria, em uma transação em um banco de dados, sejam eles de papel ou digital. Tornando-os indispensáveis em alguns sistemas, como é o caso da aplicação aqui desenvolvida. Para introduzir as discussões relacionadas ao banco de dados que foi desenvolvido neste projeto, serão apresentadas algumas definições importantes para o entendimento desta tecnologia, bem como suas características.

Um Banco de Dados, segundo Elmasri e Navathe (NAVATHE; ELMASRI, 2010), pode ser definido como uma coleção de dados logicamente relacionados que possuem um público e uma finalidade específicos e que representam eventos da vida real. Um Sistema Gerenciador de Banco de Dados (SGDB) é a união desta coleção de dados e programas que os manipulam, possuindo como principal objetivo o armazenamento e recuperação de informações.

No decorrer da história dessa tecnologia, os bancos de dados sofreram alterações e evoluções. Os primeiros bancos de dados utilizados na década de 1960, eram implementados em computadores mainframes e possuíam sistemas hierárquicos

ou de redes, onde os dados possuem relacionamento pai-filho e estão estruturados em árvores. Além disso, esses sistemas não dispõem de interfaces gráficas, portanto, todas as interações devem ser feitas por interfaces baseadas em linguagem de programação, o que dificulta as consultas e transações. Foi em 1970, que Codd (CODD, 1970), na época funcionário da *IBM Research*, propôs o modelo de dados relacional, que baseia-se na lógica de predicados de primeira origem e na teoria de conjuntos.

Segundo Silberschats e Korth (SILBERSCHATZ; KORTH, 2006), o modelo relacional se destacou pela simplicidade se comparado aos modelos rede ou hierárquico, facilitando o trabalho dos programadores. Este modelo se baseia em uma coleção de relações. Estas relações são representadas em tabelas e as colunas das tabelas demonstram um relacionamentos entre os dados armazenados. Além disso, a união e correlação das tabelas, proporcionada pela lógica relacional, aumenta exponencialmente a quantidade de relações que um banco de dados pode representar.

Um SGBD baseado em lógica relacional precisa obedecer a propriedades conhecidas como ACID, que é a sigla para: Atomicidade, Consistência, Isolamento e Durabilidade. Estas propriedades objetivam garantir o correto funcionamento do banco de dados e a integridade dos seus dados (LÓSCIO et al., 2011). A seguir, será apresentada uma sucinta explicação de cada uma dessas propriedades:

- A Atomicidade objetiva garantir que uma transação é atômica, ou seja, indivisível. Dessa forma, ou ela é executada completamente ou não é executada.
- A Consistência objetiva garantir que os dados são coerentes ao final de cada transação. Na prática, essa propriedade consiste em assegurar que, em caso de falhas, todas as ações executadas são desfeitas.
- O Isolamento diz respeito a não interferência de uma transação em outra executando paralelamente.
- E a Durabilidade garante que os dados disponíveis permanecem corretos e íntegros no decorrer do tempo, mesmo após uma queda do sistema.

Apesar de todas as vantagens trazidas pela aplicação prática das propriedades descritas, os bancos de dados que utilizam o modelo relacional possuem limitações que inviabilizam a sua utilização em determinados tipos de sistemas. A principal delas diz respeito a escalabilidade. Com a crescente geração de dados, principalmente em sistemas como as redes sociais, surgiu a necessidade de utilizar um novo modelo de relação de dados. Modelo este que permite uma menor estruturação dos dados e que se comporta mais eficientemente quando a quantidade de dados gerada diariamente é muito grande (GARCIA; SOTTO, 2019).

Com isso, surgem os bancos de dados não relacionais ou *NoSql*, com a proposta de uma distribuição simplificada e, conseqüentemente, um bom desempenho no quesito escalabilidade, mas em contrapartida, não garantindo rigidamente as propriedades ACID. Esta é uma das principais diferenças entre o modelo relacional e o não relacional. A seguir estão listadas algumas das características dos bancos de dados não relacionais:

- **Ausência de esquemas:** que consiste em não ter uma definição padrão do esquema de relação dos dados. Possibilitando adicionar e remover campos de dados de forma simples e atribuindo maior escalabilidade, devido a não utilização de espaços vazios ou por valores nulos.
- **Controle de Concorrência multiversão:** que consiste em não possuir travas durante a execução das transações, possibilitando execuções em paralelo.
- **Escalabilidade horizontal:** por conta do controle de concorrência explicado, é possível manipular uma quantidade crescente de dados.
- **Consistência eventual:** que consiste em garantir que as diversas versões do banco de dados distribuídos estão consistentes entre si eventualmente e não em todo o tempo.

Levando em consideração as características dos bancos de dados que utilizam o modelo relacional e o não relacional, serão apresentados tópicos de comparação entre os modelos, posteriormente utilizados para justificar a escolha da tecnologia.

**Sobre a estruturação dos dados:** Em um banco de dados relacional os dados são organizados em tabelas e possuem entre si relacionamentos caracterizados pela utilização de chaves. Portanto, utilizam estruturação em linhas e colunas. As tabelas e estes relacionamentos precisam ser definidos no início dos projetos e as alterações nos campos não é simples.

Os bancos de dados não relacionais, por sua vez, trabalham com esquemas chave-valor e esquemas flexíveis. Ou seja, os atributos podem ser adicionados ou removidos sem uma pré definição padrão para todos os tipo. Esta flexibilidade proporciona maior facilidade de alteração dos dados salvos, mas não garante a integridade dos mesmos.

**Sobre a escalabilidade:** Como já mencionado acima, os bancos de dados não relacionais apresentam melhor desempenho quando aplicados a sistemas que geram um grande volume de dados. Como é o caso das redes sociais. Isso se dá pela falta de bloqueios, ou seja, não há travas nos recursos e eles podem ser acessados concorrentemente pelos sistemas e usuários.

Já os bancos de dados relacionais apresentam maiores dificuldades em serem escalados, devido a sua estrutura de dados rígida. Consequentemente, gera-se um alto custo quando existe a necessidade de ser escalonamento de um banco de dados relacional, devido à necessidade de várias máquinas e pelo seu complexo modelo lógico.

**Sobre a consistência dos dados:** Para garantir a grande escalabilidade, os bancos de dados não relacionais sacrificam a consistência dos dados. Neste tipo de banco diversos recursos podem ser acessados e alterados ao mesmo tempo por diferentes sistemas. Apesar disso, não significa que não há nenhum tipo de garantia de consistência dos dados. Através da consistência eventual garante-se que, caso não sejam alterados os recursos, todas as requisições receberão a última versão dos dados armazenados no banco.

Por outro lado, os bancos de dados relacionais, amplamente utilizados e testados, garantem a característica de consistência a todos os seus dados, principalmente

através da utilização de travas.

**Sobre Disponibilidade Disponibilidade:** Em questão de disponibilidade, os bancos de dados não relacionais possuem vantagens, pois possuem um alto nível de distribuição dos dados. Esta característica faz com que o sistema fique indisponível por menos tempo, pois uma vez indisponível é possível disponibilizar gradualmente o sistema, não sendo necessário carregar toda a estrutura. Além disso, possibilita que mais requisições sejam feitas ao mesmo tempo.

Os bancos de dados relacionais possuem dificuldades em distribuir os dados e por isso estão em desvantagem nesta característica.

As características analisadas acima foram utilizadas para selecionar, neste projeto, o modelo de banco de dados a ser utilizado. A partir dos pontos acima concluiu-se que, em questão de escalabilidade, o sistema aqui apresentado não se caracteriza pela crescente e rápida geração de dados diariamente. A maior movimentação de dados se dará no processo de cadastramento dos usuários e apenas, o que ainda assim caracteriza uma quantidade muito inferior de geração de dados se comparado a redes sociais.

Com relação ao modelo e consistência dos dados, o modelo relacional se apresenta como mais adequado, visto que, os dados desta aplicação são muito bem definidos e estruturados. Também, o modelo de tabelas e planilhas já é parte da rotina do cotidiano das equipes técnicas, inclusive, a maior parte dos dados atualmente estão estruturados em modelos de planilhas eletrônicas. Finalmente, a garantia das propriedades de integridade e consistência dos dados é importante para este projeto e o banco de dados relacional garante, não só estas duas propriedades, mas todas as propriedades ACID.

Sobre a disponibilidade, devido a intenção de desenvolver um aplicativo que consuma os dados deste banco, o banco de dados não relacional se mostraria mais adequado ao projeto. Entretanto, tendo em vista as outras características discutidas, que

são consideradas mais relevantes nesta aplicação, será dispensada esta característica. Além disso, o impacto causado pela indisponibilidade do sistema por um determinado período de tempo é pequeno se comparado às perdas de integridade ou consistência dos dados e os problemas com indisponibilidade do sistema ou mesmo de acesso à internet serão tratados localmente no aplicativo móvel e também na aplicação *Web*.

Para o desenvolvimento do sistema foi escolhida a utilização do Banco de Dados Relacional *MySQL* utilizando o programa *MySQL Workbench* como auxiliar no desenvolvimento. A escolha das tecnologias de banco de dados empregadas neste projeto estiveram limitadas às necessidades das equipes técnicas de Tecnologia da Informação da Justiça Federal, que futuramente trabalharão com a manutenção e suporte deste sistema. Portanto, tendo em vista os padrões utilizados atualmente em outros sistemas mantidos pela equipe, escolheu-se a utilização de bancos de dados relacionais. Entretanto, apesar da limitação na escolha atribuída pela equipe técnica da JF, a escolha de um banco de dados relacional se mostrou muito adequada ao sistema aqui proposto se comparados aos modelos não relacionais e suas características, como explicado no decorrer deste capítulo.

#### 4.2 *WEB SERVICE REST*

A partir da escolha do Banco de dados, foi possível planejar e pesquisar sobre as tecnologias de *web services* a serem utilizados no sistema. Nesta subseção, será apresentada a revisão bibliográfica e a teoria por trás desta tecnologia.

Nos primeiros anos de utilização da internet, o principal *software* utilizado para acessar os conteúdos disponíveis eram os navegadores, também conhecidos pelo seu nome em inglês *browsers*. E os *sites* e conteúdos disponíveis, em sua maioria, eram páginas simples e normalmente estáticas de conteúdos. Entretanto, com a evolução da internet, sua utilização massiva e a evolução natural destes conteúdos demandou integração com outros sistemas, como bancos de dados e outros servidores. Este processo abriu no mercado um nicho tecnológico ainda não explorado nesta época

(ALBINADER NETO, 2006).

Uma das dificuldades de integrar os diferentes sistemas neste período era a falta de padronização na internet, que obrigava muitas empresas a definirem formatos de comunicação proprietários, a fim de permitir as integrações entre os seus sistemas. Entretanto, as padronizações eram restritas aos sistemas de uma única empresa, dificultando a integração com sistemas de empresas ou organizações parceiras. Portanto, surge então a necessidade de se desenvolver esta padronização, facilitando a integração de sistemas de diferentes organizações através da internet. É neste contexto que os primeiros *Web Services* são desenvolvidos, juntamente com outras tecnologias de comunicação entre sistemas distribuídos, bem como a linguagem XML (*Extended Markup Language*), como um padrão de envio de dados entre os sistemas.

Um serviço *web* pode ser definido como uma interface que oferece integração entre sistemas e permite a comunicação entre aplicações heterogêneas (EULALIO et al., 2016). Ou seja, é um serviço que trabalha como uma camada de interfaceamento entre as aplicações, isolando-as e tornando a comunicação independente da plataforma de desenvolvimento ou da linguagem de programação. Esta comunicação se dá através do uso de um padrão de transmissão dos dados que serão enviados por ambas as partes.

A padronização de formatos estimulou não apenas as integrações, mas também a criação de novas soluções, como é o caso da comunicação cliente-servidor. A mesma se caracteriza como um sistema cliente que requisita ações ao servidor sem a necessidade de interferência humana. Esta comunicação acontece basicamente através de requisições realizadas pela internet, em sua maioria, utilizando o protocolo HTTP (*Hypertext Transfer Protocol*) e que utilizam um padrão de formato dos dados que serão enviados e recebidos nas requisições (COULOURIS et al., 2006).

Antes de iniciar uma discussão mais aprofundada sobre os tipos de *web services* existentes e suas diferenças, serão apresentadas as arquiteturas de aplicação que são usadas como base teórica para esta tecnologia.

O modelo cliente-servidor é uma arquitetura que consiste em uma integração

em duas camadas. O computador que oferece a interface para o usuário, é chamado de cliente, e o servidor possui a comunicação com outros sistemas e com o banco de dados e é executado em um robusto computador que centraliza as atividades (EULALIO et al., 2016). Toda arquitetura que se assemelha a este comportamento pode ser chamada de cliente-servidor. Algumas literaturas relativas a Bancos de dados mencionam as arquiteturas de três camadas ou mais para as aplicações *web* (NAVATHE; ELMASRI, 2010). Estas aplicações se caracterizam por serem cliente-servidor, onde o servidor processa as informações e executa consultas ou comandos no banco de dados, que seria a terceira camada desta comunicação. Ou seja, o *web service* é uma camada de interfaceamento do banco de dados e da aplicação usada pelo cliente.

Outra arquitetura que é importante ser mencionada quando tratando-se de *web services* é a Arquitetura Orientada a Serviços (SOA, do inglês *Service-Oriented Architecture*) que consistem em um conceito de arquitetura que oferece a integração através de um conjunto de rotinas que podem ser reutilizados pelas diferentes plataformas (EULALIO et al., 2016). Ou seja, especifica-se o formato de troca de dados, tanto para envio quanto para o recebimento de informações, permitindo melhor interoperabilidade entre sistemas através da padronização. O *web service* é apenas uma das formas de implementar esta arquitetura e, para tal, ele precisa atender a algumas características essenciais, que são: possuir interfaces e métodos claros e bem definidos e estar publicado em alguma intranet ou repositório, disponível para ser utilizado por seus potenciais usuários.

Apresentadas as arquiteturas que os *web services* estão baseados, serão discutidos os modelos mais comuns dos mesmos. Existem dois padrões principais que serão adotados: SOAP (*Simple Object Access Protocol*) e REST (*Representational State Transfer*). A seguir serão descritas as principais características de cada uma delas, como forma de justificar a escolha da tecnologia empregada neste trabalho.

O SOAP é um protocolo padrão para transmissão de dados dentro da arquitetura *web service* proposto pela *World Wide Web Consortium* (W3C), organização de padronização da internet. Este protocolo utiliza o formato de mensagem XML e pode

utilizar qualquer protocolo de transmissão, normalmente utiliza-se o HTTP. Além disso, é um serviço *stateful*, isso significa que o servidor responderá a requisição de acordo com o estado atual da máquina.

Por padrão, as especificações dos métodos de um *web service* SOAP precisam estar apresentadas em um arquivo chamado WSDL (*Web Service Description Language*). Este descreve os parâmetros e os nomes dos métodos que podem ser utilizados por um cliente. Ou seja, em posse deste arquivo, o cliente está apto a fazer as solicitações (MUMBAIKAR; PADIYA, 2013). O SOAP utiliza um padrão de envio de mensagens que envolve três elementos: cabeçalho, corpo e um envelope. Este formato de envio é considerado um problema, visto que a sobrecarga de informações em cada mensagem pode gerar latência na rede pela alta transferência de dados.

Já o *web service REST*, diferentemente do SOAP, não é um protocolo e sim uma definição de boas práticas na modelagem de um *web service* propostas por Roy Fielding em sua tese de doutorado, apresentada em 2000 na Califórnia (FIELDING, 2000). Os princípios propostos objetivam resultar em um *web service* de boa eficiência e simples utilizando o protocolo HTTP. Neste modelo, não há a necessidade de um arquivo WSDL, simplificando o seu desenvolvimento. Também o formato padrão dos dados fica a cargo do programador, podendo escolher o que mais lhe convenha, diferentemente do SOAP. Pela simplicidade, muitos desenvolvedores priorizam a utilização do formato JSON (*JavaScript Object Notation*). Para sintetizar estes itens, Mumbaikar e Padiya (MUMBAIKAR; PADIYA, 2013), resumiram em três princípios básicos de *design*:

**Endereçabilidade:** o REST modela todos os métodos para serem identificados e rastreáveis por uma URI (*Uniform Resource Identifier*);

**Stateless:** ou seja, não há interferência do estado da máquina; cada transação é independente e não influencia no processamento de outras, visto que, todas as informações necessárias estão contidas na solicitação;

**Interface Uniforme e padrão:** usa-se um conjunto fixo de métodos HTTP para as rotinas, são elas: GET, PUT, POST e DELETE - excluindo a necessidade do

WSDL;

Estes princípios definem um *web service REST*, se um modelo é desenvolvido seguindo todos estes princípios, ele é chamado de *RESTful*.

Para fazer o intermédio entre o Banco de dados e as interfaces com os usuários, tanto o portal *Web* quanto a aplicação para dispositivos móveis, um *Web Service* foi escolhido como sistema de integração. O mesmo é responsável pelo interfaceamento CRUD - *Create* (Criação), *Read* (Consulta), *Update* (Atualização) e *Delete* (Deleção) - e também algumas pelas manipulações lógicas e processamentos dos dados. A partir de toda a explicação dada anteriormente, escolheu-se pelo desenvolvimento de um *web service RESTful*, pela sua facilidade de desenvolvimento e simplicidade na utilização, utilizando como formato de dados o JSON.

Para o projeto do *Web Service* foi utilizada a linguagem Java, utilizando o IDE Eclipse e o *framework Spring Boot* como auxiliares na implementação.

O Java é uma linguagem amplamente utilizada no mercado atual, ela foi desenvolvida pela *Sun Microsystems* e consiste em uma linguagem orientada a objeto capaz de criar aplicativos para sistemas *web* e *desktop* (CLARO; SOBRAL, 2008). Por ser muito utilizada, esta linguagem é estável e possui suporte de integração com diversas outras tecnologias, entre elas a integração com o *MySQL*, banco de dados apresentado na sessão anterior, e a possibilidade de utilizar o *framework Spring*. O *Spring* é um *framework* desenvolvido para tornar a programação em JAVA mais rápida, fácil e segura. Ele possui uma gama extensa de bibliotecas que já estão preparadas para interagir com as tecnologias, proporciona aos desenvolvedores a criação do sistema sem precisar se preocupar com o processo de configuração do ambiente local, pois executa o *Tomcat* - servidor *web* Java. Além disso, proporciona um ambiente de teste rápido, que disponibiliza a aplicação para ser executada rapidamente e recarrega automaticamente apenas as partes alteradas da estrutura após cada salvamento, em tempo de execução. Portanto, esta ferramenta auxiliou no processo de desenvolvimento agilizando e acelerando os testes.

Foi utilizado o *software Postman* para testar os *endpoints* criados. Este sistema consiste em uma plataforma de desenvolvimento de APIs (*Application Programming Interface*). Ele permite ao desenvolvedor configurar a requisição HTTP de acordo com as suas necessidades, adicionando a configuração do *Web Service*, *Headers* e *Body* necessários.

Este *web service*, juntamente com o Banco de dados, foram disponibilizados em um servidor do LAPTl para estarem sempre disponíveis e em execução, pois serão utilizados não apenas pelo Portal *Web*, mas também pelo aplicativo para dispositivos móveis.

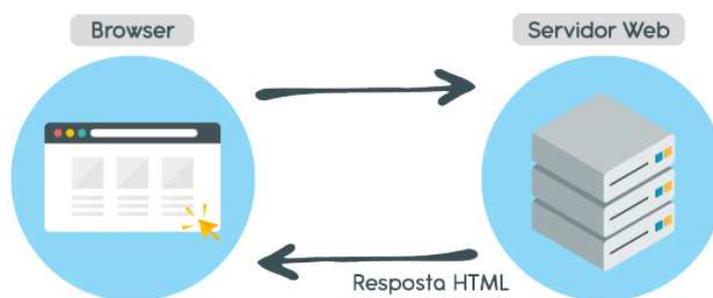
### 4.3 DESENVOLVIMENTO WEB

Até aqui foi apresentado o *back-end* do sistema, ou seja, a parte do *software* que é *server-side*, que roda do lado do servidor, acessa o banco de dados, garante as regras de negócio, segurança e escalabilidade (AFONSO; FARIA, 2018). Nesta seção será discutido o *front-end*, que é a aplicação que executa do lado do cliente e fornece uma interface de interação com o sistema. Um *front-end* possui como base as linguagens HTML, CSS e JavaScript e atualmente usam-se muitas bibliotecas e *frameworks* para auxiliar o desenvolvimento. As Aplicações são formadas de 3 camadas (EIS; FERREIRA, 2012), são elas:

1. A **camada da informação**: é a mais importante e é controlada pelo HTML que marca o significado da informação. Neste caso, entende-se informação como tudo que é consumido pelo usuário;
2. A **camada de formatação**: que é responsável pelo controle da aparência que a informação será exibida para o usuário, através da aplicação de cores, texturas, estilização de fontes, etc. Esta camada é controlada pelo CSS;
3. A **camada do comportamento**: esta camada define qual será o comportamento da informação, se ela será recarregada a partir de uma outra ação, se ela irá

transacionar ou rotacionar no decorrer do tempo, etc. Este comportamento é ditado pelo JavaScript através da manipulação do CSS.

As aplicações *web*, assim como as tecnologias no geral, passaram por diversas etapas de melhoramento e otimização no decorrer dos anos. As primeiras aplicações utilizavam o conceito da Arquitetura Tradicional de desenvolvimento, que consistia em páginas HTML que se comunicam com servidores localizados na internet ou na intranet, através da comunicação cliente-servidor, como mostra a Figura 1.



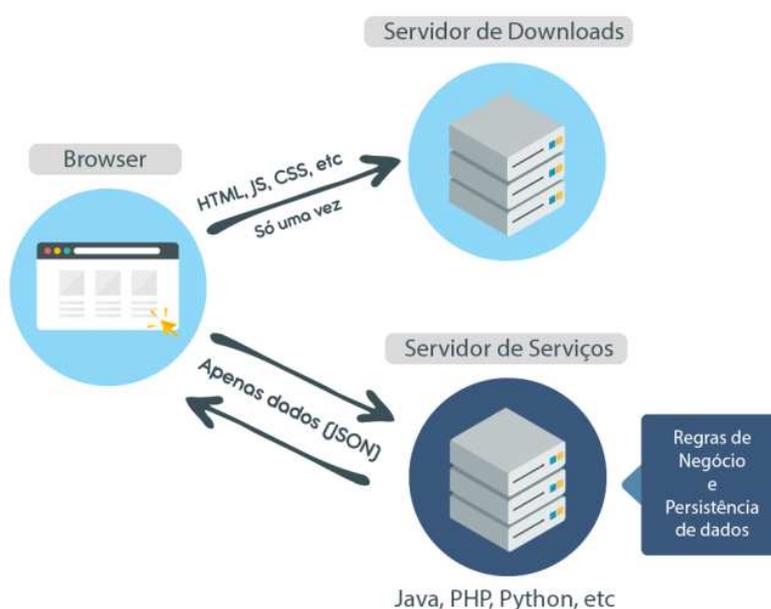
**Figura 1: Arquitetura Tradicional de desenvolvimento *web***

**Fonte: (AFONSO; FARIA, 2018)**

Após a requisição ser recebida, o servidor processa o pedido, e monta dinamicamente um código HTML que será retornado para o cliente e renderizado no *browser* (AFONSO; FARIA, 2018). Este tipo de arquitetura foi muito utilizada no começo das comunicações via internet e ainda é encontrada em algumas aplicações. Sua principal característica é que o processamento é feito majoritariamente do lado do servidor e ao navegador cabe apenas renderizar a página e apresentá-la ao usuário.

No decorrer dos anos, uma nova arquitetura, nomeada de Arquitetura Moderna, surgiu, nela existe a presença de dois servidores interagindo com o *browser*, o servidor de *download* e o servidor de serviços, como mostra a Figura 2. Neste tipo de arquitetura, preza-se pela combinação de processamento *client-side* e *server-side*. Além disso, o *back-end* não envia arquivos HTML e está totalmente desvinculado do *front-end* da

aplicação, uma das grandes vantagens deste modelo. Quando alguma requisição é feita ao servidor de serviços, que está localizado na internet, o mesmo retorna os dados solicitados pelo cliente utilizando um formato padrão de dados. Dessa forma, é o servidor de serviços que faz a interação com o banco de dados e se preocupa com segurança e aplicação das regras de negócio da aplicação. Retornada a informação, ela será tratada pelo servidor de *downloads*, que se caracteriza por ser um *back-end* que executa localmente na máquina do cliente e que é formado por HTML, CSS e JavaScript e passa para o navegador o que precisa ser apresentado ao usuário.



**Figura 2: Arquitetura Moderna de desenvolvimento web**

**Fonte: (AFONSO; FARIA, 2018)**

Segundo Zorzo e Bernardi (ZORZO; BERNARDI, 2016), a arquitetura de aplicação Web tem deixado de ser *server-side* para aplicações com utilização de códigos *client-side*. Pois a primeira é caracterizada por um grande volume de processamento de informações no servidor na internet e, conseqüentemente, utiliza grande quantidade de banda para o envio de informações. Enquanto a priorização por

processamento *client-side* é feito na máquina do cliente e as requisições ao servidor externo são feitas quando há necessidade de recuperação de um dado. Além disso, esta arquitetura busca trabalhar de forma assíncrona, evitando os bloqueios do sistema enquanto a requisição está sendo processada.

Uma das grandes vantagens da arquitetura moderna é o desacoplamento do *back-end* e do *front-end*. Esta característica permite que um mesmo *back-end* seja utilizado por diversos *front-ends*. Este é um dos principais motivos pelos quais utilizou-se como base teórica para o desenvolvimento deste projeto a arquitetura moderna, possibilitando que o portal *Web* e o aplicativo para dispositivos móveis utilizem um mesmo servidor de serviços.

Com o crescimento das implementações de aplicações *web* utilizando a arquitetura moderna, um outro conceito ganha formas, são as *Single Page Applications (SPA)*. Estas consistem em aplicações que utilizam uma única página HTML que engloba todas as páginas da aplicação *web* e toda a interação do usuário com a página é feita usando HTML, JavaScript e CSS (FINK; FLATOW, 2014). Este tipo de aplicação renderiza todo o programa quando iniciado, evitando que a cada interação o HTML tenha que ser recarregado. Quando uma requisição é feita dentro de uma aplicação SPA usa-se o *AJAX* para solicitar dados ao servidor de serviços. Retorna a requisição, o documento, em formato *JSON* ou *XML*, é interpretado pelo *Document Object Model (DOM)* e carregado no *browser*, do lado do cliente, através do JavaScript e do HTML. Serão apresentadas detalhadamente cada uma das tecnologias mencionadas anteriormente e a evolução histórica das mesmas, até chegar nos *frameworks*, muito utilizados hoje em dia pela facilidade no desenvolvimento das aplicações.

Foi em 1990 que Tim Berners-Lee (BERNERS-LEE, 1989), na época funcionário do CERN, antigo acrônimo de *Conseil Européen pour la Recherche Nucléaire*, hoje conhecido como a Organização Europeia para a Pesquisa Nuclear, escreveu o primeiro servidor e cliente *web* usando HTML e protocolo HTTP. O objetivo era a criação de uma forma fácil de resgatar arquivos de qualquer computador dentro da internet, auxiliando as divulgações bibliográficas da época. Surgiu-se deste projeto

a *World Wide Web (WWW)*. Neste período os conteúdos das páginas eram estáticos, em sua maioria textos e imagens. As páginas eram desenvolvidas do lado do servidor e enviadas para o cliente, utilizando a arquitetura tradicional, como já explicado anteriormente.

Algumas linguagens de *script* foram criadas, como o PHP, para tornar as páginas mais dinâmicas, mas não se tornaram imediatamente populares entre os programadores. Em 1995 surge o JavaScript, uma linguagem de *script* executada do lado do cliente, com a intenção de ser um validador de dados. Neste período as conexões de internet eram lentas e, por conta da arquitetura tradicional, era preciso aguardar a requisição ser retornada pelo servidor para verificar se todo o preenchimento de dados estava correto. Essa demora era problemática para os usuários, e, por conta disso, o JavaScript surgiu com o objetivo de validar os dados localmente, antes da interação com o servidor, reduzindo o tempo de espera do usuário. Mas, com a evolução dos anos, esta linguagem se transformou em muito mais do que uma linguagem de validação e hoje é considerada uma linguagem completa, capaz de fazer cálculos complexos e interações, incluindo fechamentos, funções anônimas e metaprogramação (que consiste na capacidade da linguagem em passar funções como parâmetros de outras funções) (ZAKAS, 2012).

O JavaScript então surge como uma opção para criar processamento lógico do lado do cliente e de criar páginas *web* mais dinâmicas (FINK; FLATOW, 2014). Entretanto, a maioria dos programadores neste período estava acostumado a fazer sistemas com a lógica centrada no servidor e usaram essa linguagem apenas para dinamizar a interface das páginas, não fazendo uso do seu grande potencial. Mas foi em 2000, quando o *Java Applets* e o *Flash, plug-ins* para *browsers*, começaram a se popularizar que o JavaScript cresceu como linguagem de *script*. Diferentemente dos *plug-ins* mencionados anteriormente, o JavaScript fazia parte do *browser* e isso proporcionou uma renderização mais rápida das páginas e a possibilidade de mudar uma página sem precisar recarregá-la.

Todavia, foi a partir do aumento da utilização do *AJAX (Asynchronous*

*JavaScript and XML*), em 2005, que o JavaScript começou a ser ainda mais utilizado. O *AJAX* é uma combinação das técnicas de desenvolvimento com o objeto *XMLHttpRequest*, que é um objeto do JavaScript que recebe o XML enviado pelo servidor. O programa em JavaScript consegue manipular este objeto através do *DOM*, uma *API* que mapeia o *XML* em estruturas de nós hierárquicos (ZAKAS, 2012), proporcionando ao desenvolvedor um controle maior do conteúdo e das estruturas do *XML*, possibilitando remoção, adição, substituição de nós, entre outras ações. Além disso, o *AJAX* também fornecia a possibilidade de fazer requisições assíncronas, ou seja, não é mais preciso bloquear a interface do usuário enquanto o servidor processa e retorna a resposta. A interface continua em execução e apenas a parte necessária é recarregada quando a informação é recebido pelo JavaScript. O *Google* foi uma das primeiras empresas a utilizar todo este potencial do *AJAX* no *Gmail* e também no *Google Maps*, seguidos pela *Microsoft* e também o *Yahoo* (FINK; FLATOW, 2014). Hoje em dia esta linguagem é utilizada em diversos sistemas e seu potencial tem sido muito explorado na dinamização das páginas e processamento do lado do cliente.

Durante todo este período, também o HTML e o CSS evoluíram para acompanhar a evolução das tecnologias.

O *HTML (Hypertext Markup Language)* é uma linguagem de marcação, ou linguagem de Hipertexto, que consiste em uma linguagem de publicação de conteúdo (seja ele texto, imagem, vídeo ou outras multimídias) na *web*. O seu próprio nome sugere este comportamento, pois o hipertexto é uma forma de organização do conteúdo não linearmente, mas através de nós, ou *tags*, como é característico do HTML. Por sua vez, entende-se por marcação a identificação dos conteúdos, de forma que o *browser* consiga diferenciar o que é um título, um texto, um parágrafo, uma imagem, etc (EIS; FERREIRA, 2012).

Atualmente, a versão do HTML que está em vigor é o HTML5, esta linguagem foi criada no mesmo episódio mencionado anteriormente da criação do *WWW*, por Tim Berners-Lee. Ela evoluiu através de várias versões, até que no ano de 2004 um grupo de desenvolvedores das empresas *Mozilla*, *Apple* e *Opera* se juntaram para formar

o *WHATWG (Web Hypertext Application Technology Working Group)*. Este grupo tinha como objetivo escrever o que hoje é conhecido como HTML5. Na época, outra organização, a *W3C*, trabalhava para o desenvolvimento da XHTML, mas esta iniciativa não estava agradando a comunidade dos desenvolvedores, e por isso a *WHATWG* iniciou suas atividades. Em 2006 o próprio Tim Berners-Lee se junta ao grupo neste projeto.

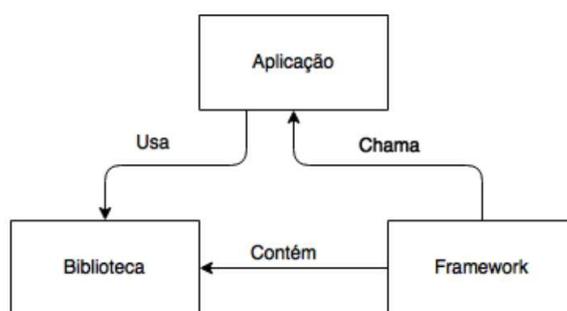
O HTML5 fornece suporte para a utilização do CSS e do JavaScript e o principal objetivo da equipe era facilitar a manipulação dos elementos, tornando isso transparente ao usuário. Por conta disso, durante todo o período de desenvolvimento, outras APIs foram definidas, formando uma base cada vez mais sólida para o desenvolvimento *web*. Houve a introdução de novas *tags*, modificação de funções e a padronização de determinadas técnicas do desenvolvimento *web* com o objetivo de introduzir mais semântica ao código em menos quantidade de linhas. Além disso, houve uma preocupação com a retrocompatibilidade com as versões anteriores, não sendo necessário reescrever todos os *sites* já disponíveis na *web*, e também com a compatibilidade de *browsers*. Segundo Eis e Ferreira (EIS; FERREIRA, 2012), a criação de código interoperável, ou seja, capaz de operar ou atuar com outros sistemas, torna esta nova versão pronta para futuros dispositivos e facilita a reutilização de informações.

O CSS (*Cascading Style Sheets*) por sua vez, é um mecanismo para adicionar estilo e formatação as informações do HTML. Ela pode aplicar formatação a qualquer tipo de formato de dados disponível na página HTML, seja ela imagem, vídeo ou texto. Segundo Eis e Ferreira (EIS; FERREIRA, 2012), a versão atualmente disponível no mercado é o CSS3, que trouxe, se comparado às suas versões anteriores, mudanças drásticas para a manipulação e transição visual dos elementos do HTML. Já era possível fazer com o CSS formatação de tabelas, fontes, mudanças de cor, *background*, adicionar *margins* e *padding*s e posicionar os itens. Entretanto, no CSS3 também é possível fazer uma seleção mais específica das *tags* do HTML (primeira ou última, por exemplo), fazer bordas arredondadas, sombras nos elementos, manipular opacidade e gradiente de cores, além de ser possível controlar rotação e perspectiva, adicionar animação e

inclusive controlar um objeto em 3D. Estas características foram um grande avanço para os *designers* gráficos de aplicações *web* e juntamente com todo este poder de formatação dos elementos, com o JavaScript e o HTML5. Consequentemente, os *sites* evoluíram no decorrer dos anos, adicionando complexidade e interação, transformando a experiência do usuário, não só com relação ao visual, mas com relação a responsividade dos *sites* e tempo de carregamento.

Com tantas evoluções, aumentou-se a complexidade das aplicações *web* desenvolvidas e isso exigiu dos desenvolvedores mais dedicação na implementação do *front-end* das aplicações. Para auxiliar neste processo, foram desenvolvidos *Frameworks* que auxiliam nestes desenvolvimento e tornaram a experiência mais fluida e agradável para o desenvolvedor.

Um *Framework*, segundo Duarte (DUARTE, 2015), consiste em um conjunto de padrões de desenvolvimento que devem ser seguidos para tornar o código mais fácil de entender e flexível. As aplicações que utilizam este tipo de tecnologia, possuem uma grande quantidade de templates e códigos de configuração já desenvolvidos e existe um espaço específico para o código trabalhado pelo desenvolvedor. Este código é chamado quando alguma situação específica. Ou seja, um *Framework* contém uma série de bibliotecas e comanda a aplicação, conforme mostra a Figura 3:



**Figura 3: Fluxo de funcionamento de um *framework***

**Fonte: (DUARTE, 2015)**

Existem diversos *frameworks* disponíveis no mercado atualmente, como: *AngularJS*, *EmberJS*, *React*, *CanJs* ou *KnoukoutJS*. Entre eles, o *Angular* é um *Framework* que se destaca, desenvolvido pela *Google* (AFONSO; FARIA, 2018), ele auxilia na criação de interfaces utilizando o HTML, CSS e possibilita a utilização de JavaScript ou *TypeScript*, que é uma linguagem criada pela *Microsoft* que se define como um *superset* do JavaScript, ou seja, oferece tudo que o JavaScript oferece e mais algumas funcionalidades, a principal delas é a verificação de possíveis erros de digitação, ou *typo* em inglês. Segundo a documentação da linguagem, o *typo* é um dos erros mais comuns na execução de programas em JavaScript (MICROSOFT, 2012).

O *Angular* possui elementos básicos que auxiliam o desenvolvimento, entre eles: os componentes, os serviços, os templates, a injeção de dependências e ferramentas de infraestrutura. Estas ferramentas automatizam tarefas, como a execução de testes locais e agilizam o desenvolvimento do sistema. A seguir serão explicados estes principais:

**Componente:** os componentes são encapsuladores de comportamentos e regras de interface. Eles são compostos de um template HTML, uma folha de estilo CSS e uma classe que gerencia suas propriedades em JavaScript. Um componente pode conter outros componentes, tornando o código reutilizável e modular. Por exemplo, uma barra de pesquisa pode ser desenvolvida em formato de componente e ser reutilizada em headers de outras páginas. (ANGULAR, 2009)

**Templates:** todo componente no *Angular* tem seu template HTML, como já mencionado, que descreve como o componente é renderizado e permite a reutilização do mesmo em diversas partes do código.

**Serviços:** os serviços possuem o objetivo de concentrar as regras de negócios e podem ser usados por um ou mais componentes. Isso significa que alguns serviços são usados, por exemplo, para encapsular as comunicações com o servidor de serviços, através de chamadas usando o protocolo HTTP. (AFONSO; FARIA, 2018)

**Injeção de dependências:** a injeção de dependências é um padrão de projeto e consiste em prover para a classe a instância que ela precisa para usar outros componentes ou elementos de bibliotecas (AFONSO; FARIA, 2018). Esta funcionalidade promove menor acoplamento entre as partes e permite escrever códigos mais testáveis e flexíveis (MICROSOFT, 2012).

A partir de todas estas funcionalidades, é possível desenvolver um *front-end* mais complexo e inteligente, além de ser dinâmico e rápido para o usuário.

Para o desenvolvimento *Web* deste projeto utilizou-se o *Framework Angular* em sua versão 9, utilizando TypeScript, HTML5 e CSS3. O processo de criação é muito facilitado pela utilização deste *framework*, visto que o mesmo implementa todas as configurações *core* do sistema e algumas opcionais que podem ser facilmente importadas para dentro da aplicação. Além disso, utilizou-se como biblioteca de componentes o *Angular Material*, que é um repositório desenvolvido pelo time do *Angular* onde são disponibilizados componentes pré-formatados. Esta biblioteca de componentes facilita ainda mais a utilização do *framework* e o desenvolvimento das interfaces.

Para testar e desenvolver a plataforma foi utilizado o *Browser Chrome* como plataforma de teste e a própria infraestrutura de testes que o *Angular* oferece. As telas e recursos implementados levaram em consideração os *mockups* desenvolvidos para este trabalho que foram apresentados para as equipes técnicas e serão apresentados nas seções a seguir.

## 5 METODOLOGIA

Até aqui foram apresentados conceitos técnicos das tecnologias que foram escolhidas para o desenvolvimento deste trabalho. Entretanto, o sucesso de um artefato não é determinado apenas pelas suas qualidades técnicas, mas sim, pela sua capacidade de atingir o objetivo final da sua aplicação. Portanto, para um desenvolvimento apropriado de um sistema como o aqui proposto, é necessário primeiramente entender quais são os problemas e as necessidades que o artefato se propõe a resolver. Bem como conhecer seus usuários e suas características específicas. A partir desta análise, é possível desenvolver o Levantamento de Requisitos, descrevendo previamente o comportamento esperado do sistema. Esta etapa inicial, antes da implementação, é determinante para um resultado final satisfatório do artefato, visto que é nesta etapa que é feito o alinhamento entre as expectativas dos usuários e o que será desenvolvido.

Nesta seção são apresentados estes estudos, realizados anteriormente à implementação, objetivando desenvolver um sistema que entenda e atenda as necessidades do seu público alvo.

Para esse fim, foi desenvolvido um estudo de caso com a equipe técnica da 4ª Vara Local da Justiça Federal, localizada em Foz do Iguaçu-PR, juntamente com seu juiz responsável. A escolha desta vara como representante, se deu pelo fato da grande quantidade de apenados cumprindo pena por PSC em Foz do Iguaçu. A cidade é divisa entre Brasil e Paraguai e, por conta do fácil acesso a fronteira, muitas apreensões são realizadas de indivíduos tentando atravessá-la ilegalmente. Um exemplo, são as tentativas de entrada ou saída dos países transportando drogas ou entorpecentes. Uma

característica comum a maioria dos apreendidos nesta situação é o não envolvimento direto com o tráfico em si. Muitos destes indivíduos foram aliciadas por traficantes e, em troca de alguma remuneração ou benefício, aceitam correr o risco deste tipo de transporte. Por consequência disso, muitos se enquadram na aplicação das penas por PSC e esta modalidade de pena é aplicada com demasiada frequência pela vara local.

Os estudos aqui apresentados deram origem ao levantamento de requisitos do sistema, englobando o presente trabalho, através do *back-end* e do Portal *Web*, bem como o aplicativo para dispositivos móveis.

## 5.1 LEVANTAMENTO DOS PROBLEMAS E NECESSIDADES

Tendo como base as reuniões com a equipe técnica da JF foi possível traçar os seus problemas e necessidades. Este levantamento foi revisado e ajustado até resultarem nos pontos expostos listados a seguir, objetivando estarem coerentes e caracterizarem, da melhor forma possível, a situação a ser resolvida. Como já exposto anteriormente, os tópicos mapeados fazem referência à 4ª Vara da Justiça Federal. Portanto, para evitar repetições, em todos os tópicos lê-se que quem executa a ordem do verbo imperativo é a 4ª Vara da Justiça Federal.

### 5.1.1 PROBLEMAS

- P01** Deve garantir um meio tecnológico seguro de armazenamento de informações geradas e coletadas dos apenados e das entidades interessadas.
- P02** Deve garantir um meio tecnológico para gerar relatórios relacionados às atividades dos apenados, sob pena de dificultar a documentação das atividades prestadas por cada apenado.
- P03** Deve garantir um meio tecnológico de gerenciar as horas de serviços voluntários prestados pelos apenados, sob pena de dificultar a contabilização de horas já cumpridas pelos mesmos.

- P04** Deve garantir um meio tecnológico para que o apenado tenha controle do tempo restante de horas de serviços à serem prestadas, sob pena do mesmo não cumpri-las em tempo hábil para redução da pena.
- P05** Deve garantir um meio tecnológico e seguro de gestão das escalas dos apenados nas instituições, sob pena de prejudicar o serviço nas instituições.
- P06** Deve garantir um meio tecnológico das entidade interessadas em receber voluntários exporem suas necessidades, sob pena de não possuir entidades interessadas.
- P07** Deve garantir um meio tecnológico para notificar diariamente os apenados que possuam acesso ao aplicativo em seus aparelhos celulares sobre as horas que o mesmo ainda deve cumprir dentro de um determinado período de tempo.
- P08** Deve garantir um meio tecnológico de manter o controle diariamente das atividades dos apenados pela equipe técnica interna, sob pena de aumentar a quantidade de infrações nas atividades.
- P09** Deve garantir um meio tecnológico de manter os dados dos apenados, onde serão definidas senhas de acesso e cadastro no aplicativo mobile, sob pena de impossibilidade de utilização do aplicativo por parte do apenado.
- P10** Deve garantir um meio tecnológico dos apenados exporem suas habilidades e cursos, sob pena de não encontrar serviço comunitário compatível com o mesmo.

#### 5.1.2 NECESSIDADES

- N01** Necessita de uma aplicação *web* com autenticação e autorização para diferentes perfis: Juiz, Administrativo e entidades interessadas em receber apenados.
- N02** Necessita de uma aplicação *web* onde será permitido o cadastramento dos apenados e das instituições e, também, onde serão disponibilizados estes dados para acesso aos autorizados.

- N03** Necessita de uma aplicação *web* onde será possível gerar relatórios relacionados às atividades realizadas pelos apenados.
- N04** Necessita de aplicativo mobile onde será feito o registro das entradas e saídas dos apenados das instituições, juntamente com foto e geolocalização, com intuito de reduzir os problemas relacionados ao controle das atividades junto às entidades.
- N05** Necessita de aplicativo mobile onde as horas de serviço comunitária dos apenados será calculada e as informações sobre redução da pena e horas restantes serão disponibilizadas para o mesmo.
- N06** Necessita um meio tecnológico que permita à equipe técnica interna, validar a indicação de voluntariado realizada pelo apenado ou não, devido a características da pena.
- N07** Necessita de aplicação *web* que seja possível divulgar as demandas das entidades junto aos apenados.
- N08** Necessita de um aplicativo mobile que permita ao apenado ser informado sobre atividades/vagas existentes e respectivas durações para que possa se voluntariar em tais ações.
- N09** Necessita de aplicativo mobile que seja possível notificar diariamente os apenados sobre sua situação no cumprimento de sua pena.
- N10** Necessita um *dashboard* que permita à equipe técnica interna, acompanhar quais são as atividades sendo realizadas no dia e/ou semana, de forma geral listando Entidade / Apenado / Atividade / Duração.
- N11** Necessita um ambiente para cadastro e criação de uma conta de acesso para cada apenado que será utilizada durante o processo de logon na aplicação mobile.
- N12** No futuro, pensar na possibilidade que o próprio apenado possa atualizar os dados referente as suas habilidades.

## 5.2 LEVANTAMENTO DE REQUISITOS

Baseado nos problemas e necessidades, expostos no item anterior, foram esboçados os Requisitos Funcionais do sistema, apresentados a seguir. Os requisitos nortearam tanto a modelagem quanto a implementação do sistema. É importante ressaltar que os requisitos aqui apresentados excluem aqueles que dizem respeito ao aplicativo para dispositivos móveis.

### 5.2.1 REQUISITOS DO PORTAL *WEB*

**[RF01] Exibir tela de Login**

**Descrição** A página *web* para login é disponibilizada para o usuário.

**Pré-condições** O usuário deverá acessar em seu *browser* o domínio previamente criado para visualizar a tela de login.

**Pós-condições** Entrada na tela para realização de login.

**[RF02] Realizar Login**

**Descrição** O sistema verifica se a combinação da credencial do usuário e senha fornecidos estão corretas.

**Pré-condições** Estar na tela de Login. Preencher os campos obrigatórios e selecionar o botão Entrar.

**Pós-condições** O sistema autentica o usuário e o redireciona para a tela principal do sistema atribuindo ao mesmo as autorizações devidas, conforme o seu perfil de autorização (Juiz, Administrativo ou Entidade).

**[RF03] Realizar Logout**

**Descrição** O sistema irá encerrar a sessão ativa.

**Pré-condições** O usuário precisa estar autenticado no sistema e selecionar o botão Logout.

**Pós-condições** O usuário é redirecionado para a tela de login.

**[RF04] Recuperar senha**

**Descrição** O sistema gera aleatoriamente uma nova senha e envia para o email do requisitante.

**Pré-condições** O usuário deve possuir uma conta válida previamente cadastrada no sistema. O usuário deve selecionar a opção de Recuperar Senha na tela de Login, preencher os campos de Usuário e Email e selecionar o botão Recuperar.

**Pós-condições** O usuário recebe uma nova senha via email.

**[RF05] Adicionar novo usuário**

**Descrição** O sistema cadastra um novo usuário do sistema.

**Pré-condições** Estar autenticado no sistema e possuir nível de Permissão de Juiz ou Administrador. Selecionar a opção de Cadastro de novos usuários. Preencher os dados do usuário, incluindo seu nível de permissão, e salvar.

**Pós-condições** O sistema envia um e-mail com os dados de acesso para o e-mail cadastrado.

**[RF06] Exibir tela principal**

**Descrição** O sistema exibe a tela inicial que contém o menu de navegação entre telas.

**Pré-condições** Estar autenticado no sistema.

**Pós-condições** A tela principal é exibida para o usuário.

- [RF07] Navegar entre telas**  
**Descrição** O sistema realiza navegação entre telas baseado na escolha do usuário.  
**Pré-condições** Estar autenticado no sistema e selecionar uma opção do menu principal.  
**Pós-condições** O usuário é redirecionado de acordo com a opção selecionada e sua permissão de acesso à tela.
- [RF08] Gerar Logs**  
**Descrição** Cada ação realizada no sistema gera um registro no banco de dados.  
**Pré-condições** Estar autenticado no sistema e realizar alguma ação.  
**Pós-condições** O registro da ação é salva no banco de dados
- [RF09] Exibir registros de logs**  
**Descrição** O sistema exibe os registros de logs existentes no banco de dados.  
**Pré-condições** Estar autenticado no sistema, Possuir nível de Permissão de juiz. Selecionar a opção Logs no menu. Existir ao menos um registro de log no banco de dados.  
**Pós-condições** Os registros de logs cadastrados são exibidos.
- [RF10] Cadastrar apenado**  
**Descrição** O sistema cadastra um novo apenado.  
**Pré-condições** O usuário precisa estar autenticado no sistema; Possuir nível de Permissão com autorização para executar esta ação; Selecionar o botão de cadastro de apenados, preencher os campos obrigatórios e selecionar o botão Salvar.  
**Pós-condições** Um novo apenado é cadastrado e o sistema retorna uma mensagem de confirmação.

- [RF11]            Alterar registro de apenado**
- Descrição**        O sistema altera um registro de um apenado já existente no banco de dados.
- Pré-condições**    Estar autenticado no sistema; Possuir nível de Permissão com autorização para executar esta ação. O banco de dados deverá conter ao menos um registro de apenado. Selecionar o registro a ser alterado, realizar modificações e selecionar o botão Alterar.
- Pós-condições**    Os dados são atualizados no banco de dados.
- 
- [RF12]            Inativar cadastro de apenado**
- Descrição**        O sistema inativa um cadastro de apenado do banco de dados. Registros de apenados não serão removidos do banco de dados, apenas permanecerão inacessíveis para alteração. O cadastro é inativado selecionando no botão de edição e selecionando Inativar.
- Pré-condições**    Estar autenticado no sistema; Possuir nível de permissão de Juiz. O banco de dados deverá conter pelo menos um cadastro de apenado. Estar na tela de Apenados, selecionar o cadastro a ser removido e selecionar o botão Inativar.
- Pós-condições**    Uma flag de inativo é colocada nas tabelas do Banco de Dados que possuem este registro.
- 
- [RF13]            Encontrar apenado registrado**
- Descrição**        O sistema exibe os registros de apenados cadastrados compatíveis com a busca do usuário.
- Pré-condições**    Estar autenticado no sistema; Selecionar a opção de Pesquisa, preencher o campo com alguma informação e selecionar o botão Pesquisar.
- Pós-condições**    Os registros de usuários resultantes da busca serão exibidos e poderão ser selecionados para exibir os dados.

- [RF14] Exibir registros de apenados**
- Descrição** O sistema exibe os registros de apenados do sistema *web* existentes no banco de dados. Somente os usuários classificados como Juiz e Administrativo conseguem visualizar dados de apenados cadastrados.
- Pré-condições** Estar autenticado no sistema; Possuir nível de Permissão de Juiz ou Administrativo; Existir ao menos um registro de apenado no banco de dados; Selecionar o apenado desejado a partir da busca ou na página com todos os registros.
- Pós-condições** Os registros de penado do sistema *web* são exibidos.
- 
- [RF15] Gerar relatório de apenado**
- Descrição** O sistema reúne as informações contidas no sistema relacionadas à determinado apenado e exporta um arquivo padrão de relatório, que poderá ser em formato: PDF, Planilha, CSV, JSON, imagem, entre outros.
- Pré-condições** Estar autenticado no sistema; Possuir nível de Permissão de Juiz ou Administrativo; Pesquisar o apenado, abrir as informações, selecionar o botão gerar relatório e selecionar o botão ok.
- Pós-condições** O sistema realizará o download do relatório
- 
- [RF16] Cadastrar nova entidade**
- Descrição** O sistema cadastra uma nova entidade interessada em receber serviços comunitários.
- Pré-condições** O usuário precisa estar autenticado no sistema; Possuir nível de Permissão autorizada; Selecionar o botão de cadastro de apenados, preencher os campos obrigatórios e selecionar o botão Salvar.
- Pós-condições** Um novo apenado é cadastrado.

**[RF17] Alterar registro de entidade**

**Descrição** O sistema altera um registro de uma entidade já existente no banco de dados.

**Pré-condições** Estar autenticado no sistema; Possuir nível de Permissão com autorização para executar esta ação. O banco de dados deverá conter ao menos um registro de entidade interessada. Selecionar o registro a ser alterado, realizar modificações e selecionar o botão Alterar.

**Pós-condições** Os dados são atualizados no banco de dados.

**[RF18] Inativar cadastro de entidade**

**Descrição** O sistema inativa um cadastro de entidade do banco de dados. Os registros serão mantidos no Banco de Dados, mas serão inacessíveis para alteração. O cadastro é inativado selecionando o botão de edição e selecionando Inativar.

**Pré-condições** Estar autenticado no sistema; Possuir nível de permissão de Juiz. O banco de dados deverá conter pelo menos um cadastro de entidade. Estar na tela de Entidades, selecionar o cadastro a ser removido e selecionar o botão Inativar.

**Pós-condições** Uma flag de inativo é colocada nas tabelas do Banco de Dados que possuem este registro.

**[RF19] Exibir registros de entidade**

**Descrição** O sistema exibe os registros de entidade do sistema *web* existentes no banco de dados. Somente os usuários classificados como Juiz e Administrativo conseguem visualizar dados de entidades cadastradas.

**Pré-condições** Estar autenticado no sistema; Possuir nível de Permissão de Juiz ou Administrativo; Existir ao menos um registro de entidade no banco de dados; Selecionar a entidade desejada.

**[RF20] Visualizar escala do apenado**

**Descrição** O sistema exibe para o usuário, que deve possuir registro de Juiz ou Administrativo, a escala selecionada pelo apenado pelo aplicativo mobile.

**Pré-condições** Estar autenticado no sistema; Possuir nível de Permissão de Juiz ou Administrativo; Selecionar o apenado que deseja visualizar a escala, selecionar o botão de informações.

**Pós-condições** O sistema exibirá todas as escalas definidas pelo apenado no aplicativo mobile.

**[RF21] Trocar as escalas dos apenados**

**Descrição** O sistema altera a escala previamente estabelecida pelo apenado e envia ao mesmo uma notificação de aviso.

**Pré-condições** Estar autenticado no sistema; Possuir nível de Permissão de Juiz ou Administrativo; Selecionar o apenado que deseja alterar a escala, selecionar o botão visualizar escala, selecionar o botão alterar escala, alterar da forma como deseja e selecionar o botão salvar.

**Pós-condições** A escala do apenado será atualizada no banco de dados, será também atualizada no aplicativo mobile e uma notificação o informará da alteração.

**[RF22] Aprovar as escalas dos apenados**

**Descrição** O sistema aprova ou reprova a escala escolhida pelo apenado e envia ao mesmo uma notificação de aviso.

**Pré-condições** Estar autenticado no sistema; Possuir nível de Permissão de Juiz ou Administrativo; Selecionar o apenado que deseja analisar a escala ou selecionar a escala com aprovação pendente diretamente, selecionar o botão visualizar escala. Após a análise, selecionar o botão aprovar ou reprovar.

**Pós-condições** O status da escala do apenado será atualizado no banco de dados, será também atualizada no aplicativo mobile e uma notificação o informará da aprovação ou reprovação.

**[RF23] Aprovar as escalas finalizadas**

**Descrição** Após finalizada a atividade e realizado o check-out, a entidade que recebeu o apenado precisa aprovar a finalização, confirmando que a atividade foi realizada

**Pré-condições** Estar autenticado no sistema; Possuir nível de Permissão de Entidade; Selecionar a escala que deseja analisar que possua aprovação pendente; Selecionar o botão visualizar escala. Após a análise, selecionar o botão aprovar ou reprovar.

**Pós-condições** O status da escala do apenado será atualizado para concluída no banco de dados, será também atualizada no aplicativo mobile e uma notificação o informará da aprovação ou reprovação. Além disso, as horas reduzidas serão efetivamente calculadas.

- [RF24]**            **Visualizar *Dashboard* de atividade**
- Descrição**        O sistema exibe todas as escalas que estão sendo cumpridas naquele momento com um sistema de cores que resume a atividade em questão, bem como atividades que acontecerão em um futuro próximo.
- Pré-condições**    Estar autenticado no sistema; Possuir nível de Permissão de Juiz ou Administrativo; Selecionar a página do *Dashboard*.
- Pós-condições**    O sistema exibe a tela do *dashboard* atualizada com as últimas informações das escalas que estão sendo cumpridas.
- 
- [RF25]**            **Cadastrar novas atividades**
- Descrição**        O sistema deve permitir o cadastramento de uma nova atividade.
- Pré-condições**    O usuário precisa estar autenticado no sistema; Possuir nível de Permissão de Entidade; Selecionar o botão de cadastro de atividade, preencher os campos obrigatórios e selecionar o botão Salvar.
- Pós-condições**    Um nova atividade é cadastrada no sistema, mas com aprovação pendente da Justiça Federal.
- 
- [RF26]**            **Aprovar as atividades cadastradas**
- Descrição**        Após criadas as atividades, é preciso haver uma aprovação da equipe técnica da JF antes da atividade ser divulgada para os apenados.
- Pré-condições**    Estar autenticado no sistema; Possuir nível de Permissão de Juiz ou Administrativo; Existir ao menos um registro de atividade com aprovação pendente no banco de dados; Selecionar o botão de aprovação ou reprovação.
- Pós-condições**    O status da atividade será atualizado no banco de dados e a mesma estará apta a ser divulgada nos aplicativos para dispositivos móveis.

- [RF27] Exibir registros das atividades cadastradas**
- Descrição** O sistema exibe os registros de atividades no sistema *web* existentes no banco de dados. Somente os usuários classificados como Juiz e Administrativo conseguem visualizar todas as atividades cadastradas. Um usuário autenticado como entidade só consegue visualizar suas próprias atividades.
- Pré-condições** Estar autenticado no sistema; Quando possuir nível de permissão de Juiz ou Administrativo é preciso existir ao menos um registro de entidade no banco de dados; Quando o nível de permissão é de entidade, é necessário existir ao menos uma atividade cadastrada da entidade em questão.
- Pós-condições** Os registros das atividades são exibidos.
- 
- [RF28] Visualizar Mapa**
- Descrição** O sistema possuirá um mapa que conterà as localizações dos apenados em atividades naquele momento.
- Pré-condições** Estar autenticado no sistema; Possuir nível de Permissão de Juiz ou Administrativo; Selecionar a tela do Mapa; Existir ao menos um apenado realizando atividade no momento.
- Pós-condições** O sistema exibe o mapa com os marcadores de onde os apenados estão cumprindo atividades.

### 5.2.2 REQUISITOS DO *WEB SERVICE*

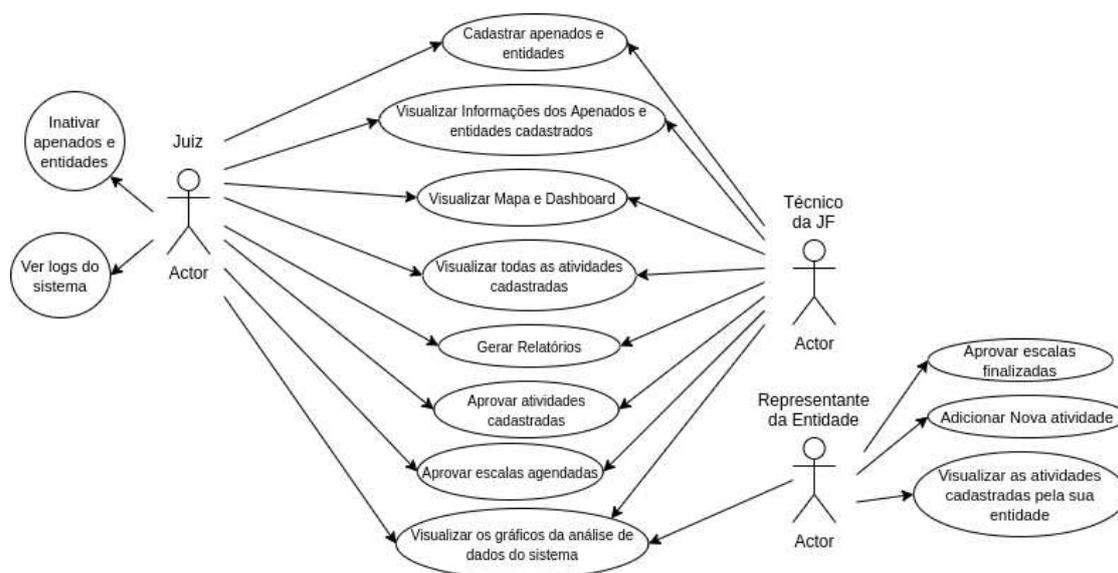
- [RF29] Divulgação das Informações dos apenados**
- Descrição** A aplicação irá realizar consultas na tabela Apenados e emití-las em formato JSON via *web*.
- Pré-condições** Conexão com o banco de dados.
- Pós-condições** Os dados são propagados via *web* em formato texto, utilizando a notação JSON.

- [RF30]            Divulgação das Informações das entidades**
- Descrição**        A aplicação irá realizar consultas na tabela Entidades e emiti-las em formato JSON via *web*.
- Pré-condições**    Conexão com o banco de dados.
- Pós-condições**    Os dados são propagados via *web* em formato texto, utilizando a notação JSON.
- 
- [RF31]            Divulgação das Informações das atividades voluntárias disponíveis**
- Descrição**        A aplicação irá realizar consultas na tabela Atividades e emiti-las em formato JSON via *web*.
- Pré-condições**    Conexão com o banco de dados.
- Pós-condições**    Os dados são propagados via *web* em formato texto, utilizando a notação JSON.
- 
- [RF32]            Divulgação das Informações das aprovações das escalas**
- Descrição**        A aplicação irá realizar consultas na tabela Acompanhamento de escalas e emiti-las em formato JSON via *web*.
- Pré-condições**    Conexão com o banco de dados.
- Pós-condições**    Os dados são propagados via *web* em formato texto, utilizando a notação JSON.
- 
- [RF33]            Receber Informações das escalas dos apenados**
- Descrição**        A aplicação irá realizar inserções na tabela Escalas via POST e PUT proveniente do aplicativo mobile, contendo informações sobre a atividade selecionada pelo apenado juntamente com os horários de cumprimento da atividade.
- Pré-condições**    O *web* Service deverá possuir conexão com a internet.
- Pós-condições**    Os dados serão registrados no banco de dados.

- [RF34] Receber Informações de check-in dos apenados**
- Descrição** A aplicação irá realizar inserções na tabela Escala via POST e PUT proveniente do aplicativo mobile, contendo informações da atividade que o apenado está realizando, horário de chegada ao local, geolocalização e uma foto.
- Pré-condições** O *web service* deverá possuir conexão com a internet.
- Pós-condições** Os dados serão registrados no banco de dados.
- 
- [RF35] Receber Informações de check-out dos apenados**
- Descrição** A aplicação irá realizar inserções na tabela Escala via POST e PUT proveniente do aplicativo mobile, contendo horário de saída ao local, geolocalização e uma foto.
- Pré-condições** O *web service* deverá possuir conexão com a internet.
- Pós-condições** Os dados serão registrados no banco de dados.

### 5.3 DIAGRAMAS DE CASOS DE USO

Fundamentado no que foi estudado sobre o problema a ser resolvido, iniciou-se a modelagem do sistema. O primeiro estudo realizado foi a listagem das várias formas de funcionamento do Portal *Web* pelos diferentes usuários. Para tal, utilizou-se do desenvolvimento do Diagrama dos casos de uso, apresentado na Figura 4. Nele estão representados os diferentes atores que utilizarão o sistema e as respectivas ações possíveis que estão disponíveis para cada um deles.



**Figura 4: Diagrama de Casos de uso do gerenciador da Justiça Federal**

**Fonte: Autoria Própria**

No diagrama é possível observar os 3 atores do portal *web*, são eles: os técnicos da JF, os juízes e os representantes das entidades que se interessam por receber apenados. Devido as particularidades de uso de cada um dos atores, determinadas ações são restritas a apenas um grupo, como é o caso da inserção de novas atividades, que pode ser realizada apenas pelos representantes das entidades. Em contrapartida, a ação de visualização dos relatórios gráficos do sistema é comum a todos os usuários.

Quem determina quais serão as ações disponíveis para o usuário é o sistema de autorização, que será descrito nas seções a seguir. Seu comportamento básico é: quando um usuário se autentica no sistema, o mesmo identifica qual é o seu nível de privilégio. Baseado nesta informação, são disponibilizadas as diferentes funções para os diferentes perfis de usuários.

### 5.3.1 DESCRIÇÃO DOS CASOS DE USO

Cada uma das ações possíveis, descritas no diagrama, estão detalhadas no Apêndice A através das descrições de cada caso de uso. Nesta seção será apresentado um destes casos de uso, o fluxo de criação da atividade. Que se inicia com a entrada de dados por parte do representante de uma entidade e finaliza com a aprovação por parte da JF. Após este processo, as atividades estão prontas para serem propagadas, via *Web service* para os apenados.

Levando em consideração que a entidade já está devidamente cadastrada no sistema e possui seus acessos, seu representante irá adicionar uma atividade, conforme o caso de uso da Tabela 1. Nela está descrito em cada linha, respectivamente: quem é o ator envolvido na ação, qual é a descrição da ação que será tomada, quais são as pré-condições, necessárias para a execução da ação e quais as pós-condições, esperadas de uma execução bem sucedida da ação.

**Tabela 1: Exemplo de uso do sistema pela entidade para adicionar Nova atividade**

<b>Nome</b>	<b>UC011 - Adicionar Nova atividade</b>
<b>Atores</b>	Representante da Entidade
<b>Descrição</b>	Caso de uso executado quando o Representante de uma entidade quer adicionar uma nova atividade
<b>Pré-condições</b>	- Usuário com privilégio de Entidade devem estar logados no sistema;
<b>Pós-condições</b>	Atividade cadastrada no sistema.

Após devido cadastramento da atividade, a mesma será disponibilizada na página de Atividades da JF, com um status de pendente aprovação. Seguindo o ciclo de vida das atividades, ela precisa ser devidamente analisada, por um técnico da JF ou o juiz, e, após aprovação, poderá ser propagada para os apenados, seguindo o caso de uso da Tabela 2.

**Tabela 2: Exemplo de utilização do sistema para aprovação de novas atividades adicionadas**

<b>Nome</b>	<b>UC012 - Aprovar atividades cadastradas</b>
<b>Atores</b>	Juiz e Técnicos da JF
<b>Descrição</b>	Caso de uso executado quando uma entidade cadastra nova atividade no sistema, sendo necessário a aprovação da JF para que ela seja disponibilizada para os apenados.
<b>Pré-condições</b>	<ul style="list-style-type: none"> <li>- Usuário com privilégio de Juiz ou Técnico da JF devem estar logados no sistema;</li> <li>- Atividade nova cadastrada no sistema por uma entidade.</li> </ul>
<b>Pós-condições</b>	Atividade aprovada e divulgada para os apenados

Os casos de uso descrevem o comportamento esperado do sistema em uma situação de normalidade, ou seja, entendendo que tudo esteja em funcionamento, as conexões devidamente estabelecidas e que as regras de negócio sejam seguidas pelos atores. Para descrever de forma detalhado o comportamento do sistema e, inclusive, levar em consideração erros, é preciso analisar os fluxos básicos de cada caso de uso, que é explicado na seção a seguir.

### 5.3.2 FLUXO BÁSICO DOS CASOS DE USO

Como já exposto acima, os casos de uso e suas descrições geram os fluxos básicos de cada um deles. Estes descrevem detalhadamente o comportamento desejado do sistema, especificando, por exemplo, quais botões precisam ser pressionados na interface para que tal ação seja realizada. Também é o fluxo básico que descreve quais são os comportamentos de exceção do sistema, ou seja, qual é o comportamento esperado do mesmo quando algo não é realizado corretamente ou quando há problemas

de conexão, por exemplo. Os fluxos básicos de cada caso de uso, estão apresentados no Apêndice B.

A seguir será apresentado o fluxo básico do sistema desejado para o caso de uso descrito anteriormente. Quando a entidade quer adicionar uma atividade, o comportamento esperado, tanto do usuário quanto do sistema, seguem o fluxo da Tabela 3. A numeração utilizada nesta tabela determina a ordem das ações a serem tomadas. Na coluna à esquerda estão descritas as ações do usuário e a coluna à direita apresenta a resposta do sistema as ações do mesmo. Além disso, são expostas as regras de negócio que este fluxo está implementando, ou seja, qual o requisito funcional que o mesmo está atendendo e também os fluxos de exceção, que descrevem a resposta do sistema a situações que fogem do comportamento esperado.

**Tabela 3: Fluxo esperado de interação entre sistema e usuário quando realizada a ação de cadastro de nova atividade**

<b>Fluxo Básico - UC011</b>	
<b>Ações dos atores</b>	<b>Ações do sistema</b>
1 - Usuários com privilégio de Entidade logam no sistema com sucesso	
2 - Usuários selecionam a página Atividades	
	3 - Sistema abre a página selecionada
4 - Usuário seleciona o botão “Adicionar nova atividade”	
	5 - Sistema abre o <i>pop-up</i> de cadastro das atividades
6 - Usuário preenche os dados da atividade (nome, descrição, horas da atividade por dia, quantidade de apenados necessários, Início e fim do período, habilidades necessárias para a atividade e as penas excluídas).	
7 - Usuário seleciona o botão “Salvar”	
	8 - Sistema salva as informações no Banco de dados
	9 - Sistema retorna mensagem de confirmação para o usuário
<b>Regras de Negócio</b>	
[RF25] Cadastrar novas atividades	
<b>Fluxo de exceção</b>	
<b>Ações dos atores</b>	<b>Ações do sistema</b>
7.a - Caso os campos obrigatórios (nome, descrição, horas da atividade por dia, quantidade de apenados necessários, Início e fim do período) não estejam preenchidos, o sistema não permite o salvamento.	
	9a - Caso algum error ocorra durante o salvamento, o sistema retorna mensagem de erro.

A partir do fluxo apresentado acima sabe-se que inicialmente é preciso que o usuário acesse o sistema com uma conta com privilégio de entidade e que navegue até a página de atividades. Em seguida, o usuário deve selecionar o botão que indica um novo cadastro de atividade e preencher os dados. Após selecionar o botão de salvamento, os dados serão registrados no banco de dados e uma mensagem de confirmação é exibida na tela. Entretanto, é preciso preparar o sistema também para a aparição de erros e comportamentos que fujam ao esperado, neste caso, um erro nos dados do cadastramento ou na conexão com o *back-end*. O fluxo de exceção é o responsável por fazer a descrição destes comportamento. Neste caso, apresentado na parte inferior da Tabela 3, o fluxo de exceção descreve o que o sistema deve fazer caso ocorram erros no preenchimento dos dados no cadastro. O sistema irá verificar que há erros, não salvará os dados no banco e retornará uma mensagem de erro identificando que é preciso revisitar os dados de entrada ou mesmo a conexão com a internet.

Após finalizada a transação anterior com sucesso, a atividade é salva no banco de dados e estará disponível para a equipe da JF, como exposto no caso de uso apresentado. O fluxo da Tabela 4 descreve o ciclo de aprovação que a atividade é submetida.

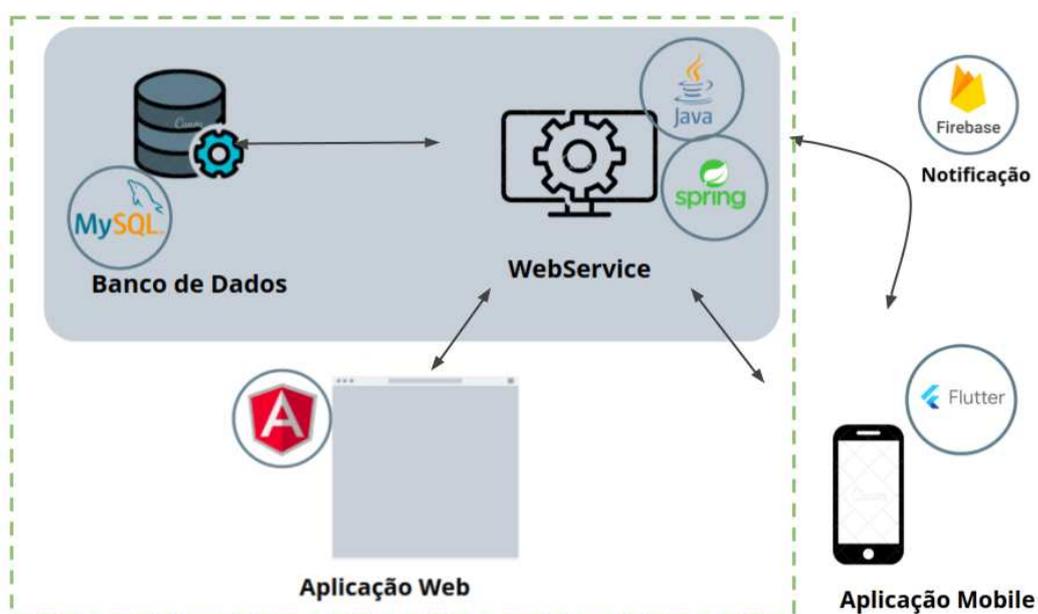
**Tabela 4: Fluxo esperado de interação entre sistema e usuário quando realizada a ação de aprovação de uma nova atividade**

<b>Fluxo Básico - UC012</b>	
<b>Ações dos atores</b>	<b>Ações do sistema</b>
1 - Usuários com privilégio de Técnico ou Juiz loga no sistema com sucesso	
2 - Usuários selecionam a página Atividades	
	3 - Sistema abre a página selecionada
	4 - O sistema exibe todas as atividades, incluindo aquelas que ainda não foram aprovadas.
5 - Usuário seleciona uma atividade ainda não aprovada	
6 - Usuário Analisa a atividade	
7 - Usuário seleciona o botão com um ícone que representa aprovação	
	8 - Sistema salva os dados no banco de dados, incluindo o log desta ação.
	9 - Sistema retorna a mensagem informando que a escala foi aprovada com sucesso
<b>Regras de Negócio</b>	
[RF26] Aprovar as atividades cadastradas	
<b>Fluxo de exceção</b>	
<b>Ações dos atores</b>	<b>Ações do sistema</b>
7.a - Usuário seleciona o botão com um ícone que representa reprovação da atividade.	
	8.a - O sistema registra no banco de dados a atividade como reprovada.

Neste caso, o fluxo de normalidade do sistema é a atividade ser aprovada pela equipe e então ser divulgada para os apenados através do *web service*, seguindo as etapas expostas anteriormente. O fluxo de exceção é a ação de reprovação da atividade. Quando a equipe da JF analisar uma atividade e concluir que ela não é apropriada, o técnico responsável deve selecionar o botão que indica a reprovação da atividade. Após esta ação, o banco de dados registrará que a atividade foi reprovada e a mesma não será propagada para os apenados.

#### 5.4 ESTRUTURA GERAL DO SISTEMA

A partir da análise realizada nos itens anteriores, entende-se que o comportamento esperado do sistema e as suas respectivas respostas estão bem determinadas e pode-se seguir para as etapas de modelagem do sistema. O diagrama da Figura 5 apresenta a modelagem e um breve resumo das tecnologias escolhidas.



**Figura 5: Estrutura Geral do Projeto de Gerenciamento de Apenados**

**Fonte: Autoria Própria**

Este diagrama apresenta um resumo geral do sistema, apresentando sucintamente as tecnologias centrais empregadas em cada componente bem como a comunicação entre cada um deles. Os elementos englobados pelo quadrado em linha pontilhada representam os componentes que fazem parte do escopo deste trabalho.

Resumindo o que já foi apresentado em detalhes nas seções anteriores: o Banco de Dados é um banco relacional *MySQL*, sua função é armazenar os dados de toda a aplicação, mantendo suas relações e integridade; O *Web Service*, por sua vez, utilizou o *framework Spring*, em linguagem *JAVA*. Ele é responsável pelo interfaceamento dos dados do Banco de dados e da aplicação *web*. Além disso, executa os processamentos dos dados e envia para as aplicações apenas as informações necessárias. Por último, o portal *web* foi desenvolvido seguindo os princípios do desenvolvimento *web* moderno e utilizando o *framework Angular*. O portal é a interface visível aos usuários finais, ela carrega os dados dos bancos de dados em estruturas lógicas, visivelmente organizadas e claras ao usuários, utilizando *Typescript*, *HTML* e *CSS*.

Neste diagrama a comunicação entre os componentes pode ser identificada pelas setas e por suas direções. É importante destacar que a comunicação do Banco de dados com o mundo externo, seja ele pelo Portal *web*, pela aplicação para dispositivos móveis ou mesmo por *softwares* como o *Postman*, é feita exclusivamente através do *web service*. Este comportamento garante maior segurança aos dados armazenados e nas manipulações dos mesmo. As comunicações do *web service* com os componentes externos é feita através de requisições *HTTPS* e os dados enviados em formato *JSON*.

## 5.5 MODELAGEM

Continuando o processo de modelagem de uma forma mais aprofundada, levando em consideração os estudos expostos anteriormente, foram realizadas duas modelagens principais: o Modelo ou Diagrama Entidade Relacionamento do Banco de Dados (*DER*) e os modelos das telas do Portal *Web*, que foram usados como base para o desenvolvimento do *front-end*. Estes dois processos estão detalhados nas seções

a seguir.

### 5.5.1 MODELO ENTIDADE RELACIONAMENTO DO BANCO DE DADOS

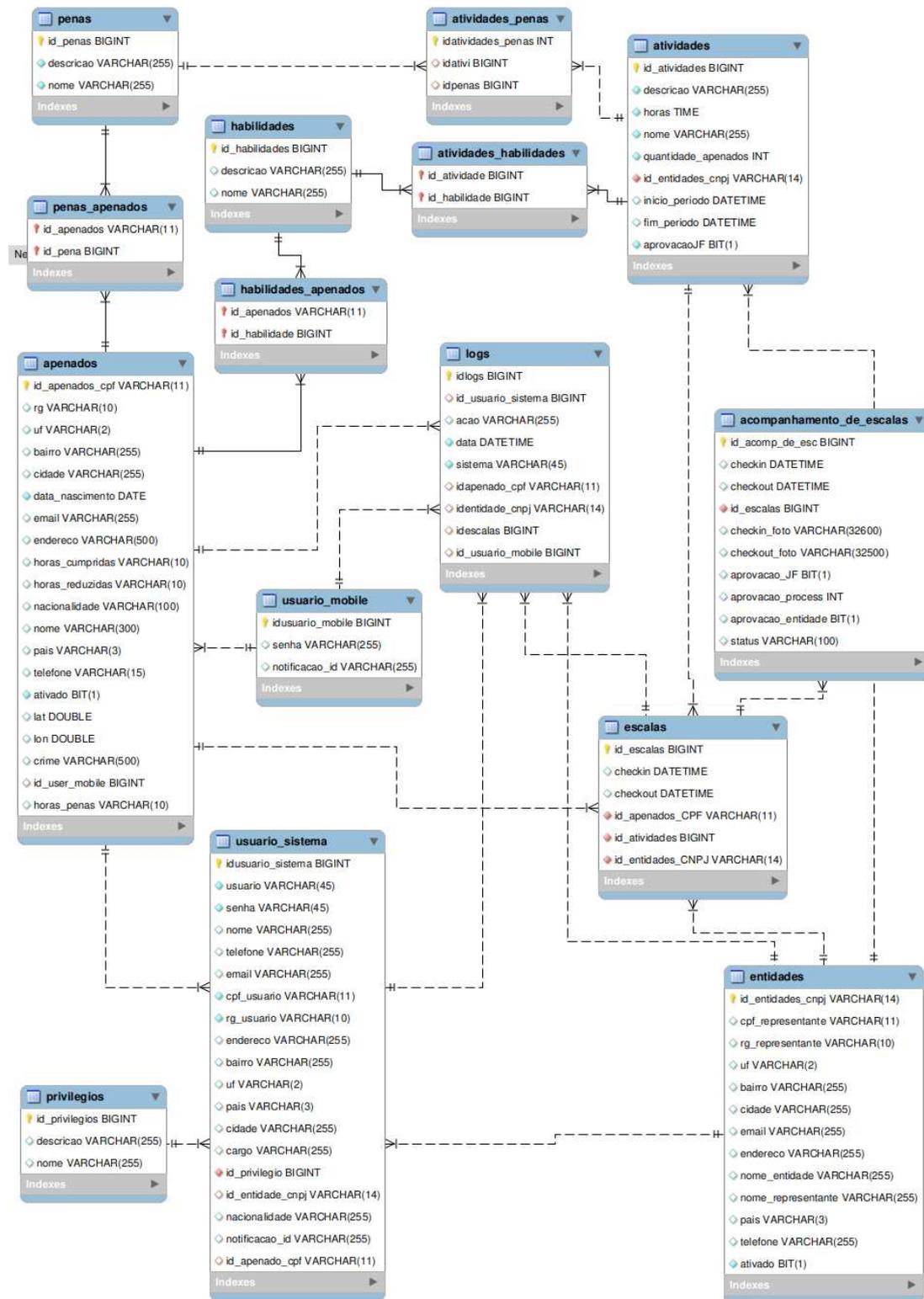
O DER é um diagrama que descreve todas as tabelas do banco de dados, bem como determinado as relações entre cada uma delas. A partir do estudo dos diversos indivíduos que interagem com o sistema e partindo das relações entre os componentes, foi possível trabalhar no DER do Banco de Dados desta aplicação, apresentado na Figura 6.

O Banco de dados aqui implementado possui um total de 15 tabelas, que registram não apenas as informações, mas também as relações entre elas. A seguir, estão descritas as características de cada uma delas:

1. **apenados**: Esta tabela armazena os dados dos apenados. Os principais dados armazenados são: os dados pessoais (endereço, informações de contato e documentos), os dados de localização, que serão utilizados no mapa, as informações do crime e do cumprimento da pena. Também é registrada a relação com a tabela usuario\_mobile, utilizada no login do aplicativo.
2. **penas**: a tabela penas armazena as características dos diferentes crimes, através de nome e descrição.
3. **penas\_apenados**: esta tabela armazena a relação existente entre a tabela apenados com a tabela penas. Como este relacionamento é caracterizado por ser muitos para muitos, ou seja, os apenados podem possuir vários crimes e um mesmo crime pode ser cometido por mais de um apenado, foi preciso criar uma tabela para relacionar estes dados, conforme as teorias de bancos de dados. Ela armazena basicamente o identificador de apenado e da pena atribuída à ele.
4. **habilidades**: nesta tabela, como na tabela penas, são armazenados nome e descrição de cada habilidade possível de um apenado.

5. **habilidades\_apenados**: esta tabela armazena o relacionamento entre os apenados e as suas habilidades, caracterizada por ser uma relação muitos para muitos.
6. **atividades**: Esta tabela armazena as atividades que as entidades cadastram, juntamente com todas as informações relativas à disponibilização da atividade, como data de início, data de fim e quantidade necessária de apenados.
7. **atividades\_habilidades**: Esta tabela armazena o relacionamento da tabela atividades com a tabela habilidades, pois, quando uma entidade entra com uma atividade, ela cadastra quais são as habilidades esperadas do apenado que pode se cadastrar para cumprir a atividade. Este dado é usado no sistema de sugestão que foi implementado no trabalho.
8. **atividades\_penas**: Esta tabela armazena o relacionamento da tabela atividades com a tabela penas. Quando uma entidade cadastra uma nova atividade, é preciso informar quais são os crimes que devem ser exclusivistas para tal atividade. Isso significa que um apenado condenado por tal crime não pode receber a atividade cadastrada. Este dado também é utilizado no sistema de sugestões de atividades.
9. **entidades**: esta tabela armazena os dados das entidades, que assim como os dados dos apenados, é utilizada em muitas outras tabelas. Ela tem relação com a tabela usuário\_sistema, que será explicada a seguir, importante para o sistema de login do gerenciador.
10. **escalas**: nesta tabela são armazenadas as informações das escalas que são agendadas pelos apenados. Ou seja, quando um apenado se interessa por uma atividade e se candidata a ela, em um determinado dia e horário, uma entrada é feita nesta tabela, registrando este compromisso.
11. **acompanhamento\_de\_escalas**: todas as informações do que foi realizado na escala, como *check-in*, *check-out* e suas respectivas fotos, são armazenadas na tabela de acompanhamento das escalas. Dessa forma, é possível comparar o compromisso, registrado na tabela escalas, com o que foi efetivamente realizado.

12. **usuario\_mobile**: O usuário mobile será utilizado pelo aplicativo para dispositivos móveis durante o processo de autenticação. Como já mencionado anteriormente, ela está ligada aos dados do apelado, como estratégia para evitar a redundância de informações.
13. **usuario\_sistema**: esta tabela armazena os usuários do sistema *web*, ou seja, os dados dos juízes, técnicos e representantes das entidades. São armazenados seus dados pessoais e informações para login. Além disso, existem uma relação com a tabela *privilegios*, que será explicada a seguir.
14. **privilegios**: esta é a tabela que armazena os diferentes níveis de privilégio que um usuário pode ter dentro do sistema, são basicamente 3 tipos: juiz, técnico e entidade. Cada um destes perfis possuem diferentes tipos de acesso e autorizações, que o sistema limita de acordo com a relação identificada no ato do login.
15. **logs**: Por último, temos a tabela *logs*, que armazena as atividades que são executadas dentro do sistema. Ela registra ações tomadas, quem as executou e a data do registro. Esta tabela é muito importante, pois torna o sistema auditável e rastreia ações.



**Figura 6: Diagrama Entidade-Relacionamento do Banco de Dados**

Fonte: Autoria Própria

### 5.5.2 MOCKUPS DO PORTAL WEB

A última modelagem necessária foi o desenvolvimento das telas que serão exibidas para os usuários. Para modelar o *front-end* do portal *Web* foram utilizadas 3 ferramentas: o *Canvas*, o *inVision* e o *Angular Material*. Os dois primeiros são plataformas de *design* usadas para o desenvolvimento de *mockups* de telas em geral. A grande vantagem em utilizá-los é a possibilidade de implementar os elementos do *Angular Material*, terceira ferramenta utilizada. O *Angular Material* consiste em um repositório de componentes padrão do *framework* que podem ser utilizados para o desenvolvimento do *front-end*. Utilizando este recurso as telas esboçadas tornam-se muito próximos dos resultados que são alcançáveis com a utilização do *framework Angular*. Além disso, o *software inVision* permite ao desenvolvedor criar interação entre os elementos das tela, permitindo uma navegabilidade entre mesmas. Este recurso foi muito importante na apresentação de aprovação das telas do sistema para as equipes técnicas da JF, através deste recurso os usuários puderam identificar como seria o funcionamento do sistema de uma forma mais próximos a realidade.

A seguir estão apresentadas as principais telas desenvolvidas e utilizadas como padrão para o desenvolvimento deste trabalho. São elas: a tela de Login, Figura 7; tela padrão inicial, Figura 8; Tela do *Dashboard*, Figura 9; Tela com o Mapa e a localização dos apenados, Figura 10; Tela dos Apenados, Figura 11; Tela das Entidades, Figura 12; Tela para a Geração de Relatórios, Figura 13 e a tela de Logs Figura 14.



**Figura 7: Mockup da Tela de Login**

**Fonte: Autoria Própria**



**Figura 8: Mockup da Tela inicial**

**Fonte: Autoria Própria**



**Figura 9: Mockup da Tela do Dashboard**

**Fonte: Autoria Própria**



**Figura 10: Mockup da Tela do Mapa**

**Fonte: Autoria Própria**



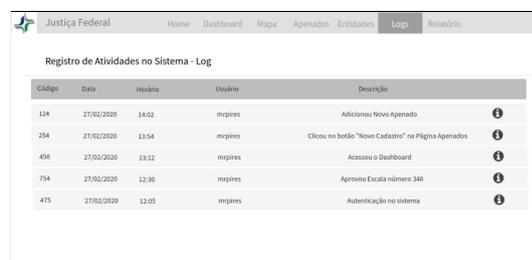
**Figura 11: Mockup da Tela dos Apenados**  
**Fonte: Autoria Própria**



**Figura 12: Mockup da Tela das Entidades**  
**Fonte: Autoria Própria**



**Figura 13: Mockup da Tela de Relatórios**  
**Fonte: Autoria Própria**



**Figura 14: Mockup da Tela de Logs**  
**Fonte: Autoria Própria**

Estas telas foram utilizadas como inspiração para o sistema final do Portal *Web*. No decorrer do desenvolvimento e de acordo com os componentes disponíveis, elas foram adaptadas. As telas finais serão apresentadas na seção de Resultados.

## 5.6 IMPLEMENTAÇÕES ESPECÍFICAS DO SISTEMA

Além de toda a implementação para tornar o sistema funcional, este projeto possui características e funcionalidades específicas que serão apresentadas nas seções a seguir. Neste capítulo serão apresentados os módulos de segurança, o sistema de geração de relatórios individuais e gráficos do portal *web* e também o sistema de sugestão de atividades.

### 5.6.1 MÓDULOS DE SEGURANÇA - *SECURITY BY DESIGN*

Este projeto levou em consideração os princípios do *Security by Design*, que consiste em desenvolver o projeto do sistema já pensando em módulos de segurança e não apenas aplicá-los ao final do desenvolvimento. Segundo estudantes da escola Naval de Pós-graduação (LEVIN et al., 2007), que desenvolveram um relatório sobre os princípios de segurança para sistemas computacionais, o aumento da quantidade de sistemas envolvidos nos processos e da complexidade dos mesmos, influencia diretamente no aumento da quantidade de vulnerabilidades a que estão expostos. Pensando nisso, desde o princípio do projeto, planejou-se um sistema seguro, através da implementação de recursos diversos disponíveis nos *frameworks* e nas infraestruturas utilizadas.

Todos os sistemas disponíveis na internet, deveriam ser implementados pensando nos requisitos de segurança, evitando exposição dos seus dados e até roubo dos mesmos. O sistema aqui apresentado em especial precisa se preocupar pontualmente com os requisitos de segurança, visto que lida diariamente com dados sensíveis relacionados a situação legal de um indivíduo e inclusive sua localização.

Este projeto cobre os três A's da segurança de sistemas, provendo recursos de: autenticação, autorização e auditoria. Cada uma das três grandes áreas da segurança de sistemas será apresentada a seguir levando em consideração a teoria apresentada por Maziero (MAZIERO, 2013).

**Autenticação** consiste em um conjunto de técnicas empregadas para identificar o usuário ou o recurso que quer acessar o sistema. De forma básica, o usuário se identifica para o sistema, utilizando credenciais ou características biométricas por exemplo, e o sistema confirma que o usuário é quem ele diz ser.

**Autorização** também conhecida como controle de acesso, consiste em técnicas usadas para definir ações permitidas para os diferentes tipos de usuários. Para cada um deles são definidas regras de acesso aos recursos e o sistema de autorização se propõe a garantir que essas regras serão aplicadas.

**Auditoria** consiste em técnicas de registro das atividades efetuadas no sistema. Este tipo de recurso garante que as atividades sejam salvas permitindo sua análise e utilização futura, podendo auxiliar, por exemplo, na identificação de comportamentos indevidos ou suspeitos do sistema.

Primeiramente, sobre a grande área da Autenticação, tem-se que esta pode ser subclassificada em três técnicas. Duas delas foram utilizadas neste trabalho. A seguir são apresentadas as características sobre cada uma das subclassificações de uma autenticação:

”**Algo que você sabe**” que é um fator de autenticação que se baseia no conhecimento do usuário, um exemplo são as credenciais, *login* e senha.

”**Algo que você tem**” que se baseia, normalmente, em um objeto físico e tangível para autenticar o usuário, exemplo: *smartcard* ou um cartão físico;

”**Algo que você é**” que são as autenticações feitas através de características biométrias, levando em consideração características físicas do usuário, como a íris ou a digital.

A combinação de mais de um destes fatores de autenticação é chamada de autenticação multifator. Esta combinação torna a autenticação mais segura, visto que se baseia em mais de uma fase de comprovação da identidade do usuário. Neste sistema utilizou-se a autenticação multifator combinando o primeiro e o segundo princípio. Para acessar no Portal *web* o usuário deve estar previamente cadastrado e possuir credenciais, ou seja, *login* e senha. Esta autenticação se baseia no conhecimento do usuário. Passada essa fase e confirmada a senha, o sistema envia para o número de celular cadastrado no sistema um SMS com um código de 6 dígitos que deve ser inserido na aplicação, cobrindo o segundo princípio, entendendo que o celular se caracteriza como o objeto de posse do usuário que precisa estar em mãos na hora da autenticação do sistema.

Sobre o princípio da autorização, este sistema é baseado em três perfis diferentes, são eles: o juiz, o técnico da JF e o representante da entidade. Cada um

possui um nível de autorização diferente dentro do sistema. Na etapa de *login*, o sistema identifica quais são os privilégios do usuário e carrega apenas os itens que estão acessíveis para seu perfil.

Um perfil de representante de uma entidade, consegue visualizar todas as atividades que estão planejadas ou aconteceram em sua instituição. É importante ressaltar que a escala planejada só é visualizada por uma entidade após ser previamente aprovada pelas equipes técnicas da JF. Também no perfil de entidade, é possível adicionar atividades novas, que serão criadas no sistema, mas que ainda precisarão ser aprovadas pelas JF. Veja na Figura 15 a tela inicial do sistema para um representante de entidade.



**Figura 15: Tela inicial padrão do sistema para um usuário logado com acesso de entidade**

**Fonte: Aatoria Própria**

Já um perfil de técnico e juiz são semelhantes, eles podem gerenciar os apenados e entidades do sistema, possuem acesso aos seus dados e podem alterá-los, tem acesso a todas as atividades cadastradas e podem aprová-las ou reprová-las. Além disso, conseguem visualizar o mapa com todas as escalas em andamento e o *dashboard* que contém um resumo histórico de todas as atividades. A diferença entre estes perfis

é que com privilégio de juiz é possível visualizar todos os logs gerados no sistema e também é possível gerenciar os usuários do sistema, podendo cadastrá-los e alterá-los. Na Figura 16 é possível visualizar a tela inicial de um juiz e na Figura 17 é possível visualizar essa mesma tela para um técnico da JF.



**Figura 16: Tela inicial padrão do sistema de um usuário logado com acesso de juiz**

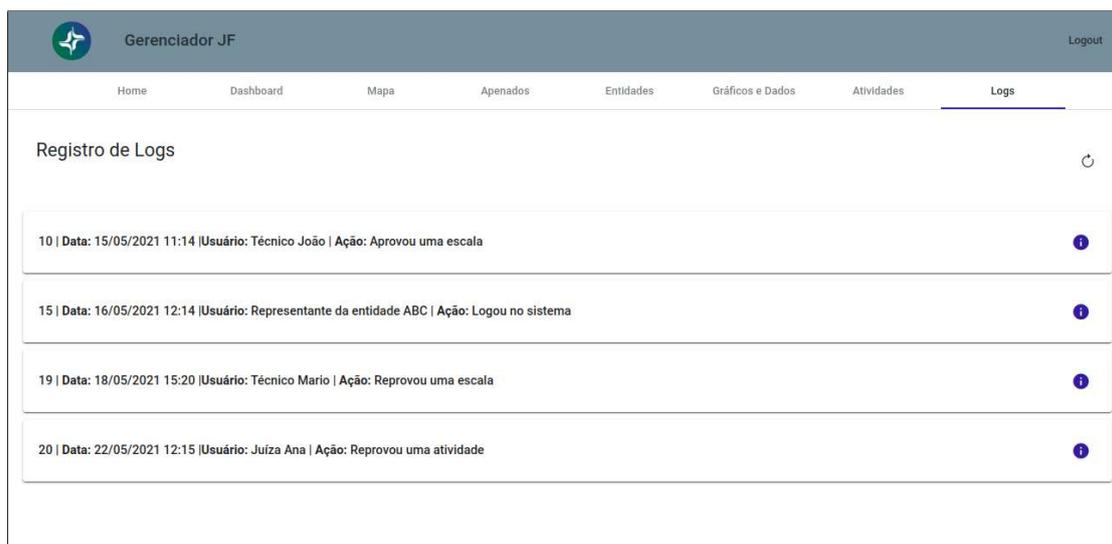
**Fonte: Autoria Própria**



**Figura 17: Tela inicial padrão do sistema de um usuário logado com acesso de Técnico da JF**

**Fonte: Autoria Própria**

Para completar o último "A" dos três A's da segurança, o sistema registra logs com todas as ações realizadas no sistema, desde as autenticações, as aprovações e reprovações e também a adição de novos cadastros. Dessa forma, o sistema se torna auditável para ações futuras. É possível visualizar hora e data da realização da ação, quem foi o usuário da ação e também qual foi, especificamente, a ação realizada. É possível visualizar na Figura 18 a tela, visível pelos juízes, com os logs do sistema.



Registro de Logs				🔄
10	Data: 15/05/2021 11:14	Usuário: Técnico João	Ação: Aprovou uma escala	🔍
15	Data: 16/05/2021 12:14	Usuário: Representante da entidade ABC	Ação: Logou no sistema	🔍
19	Data: 18/05/2021 15:20	Usuário: Técnico Mario	Ação: Reprovou uma escala	🔍
20	Data: 22/05/2021 12:15	Usuário: Juíza Ana	Ação: Reprovou uma atividade	🔍

**Figura 18: Tela de Logs**

**Fonte: Autoria Própria**

Além disso, o sistema do *Web Service* conta com a implementação do *Web Security*, pacote do *framework Spring* que fornece uma série de mecanismos de segurança. Estes foram utilizados para aumentar o nível de segurança da aplicação, é possível implementar através deste pacote, por exemplo, filtros de *hosts* autorizados a receber *requests* e credenciamento com autenticação via *HTTP Header Options*. Por último, também o banco de dados possui um sistema de autenticação utilizado *login* e senha.

Sabe-se que todos os sistemas, mesmo implementados os mecanismos de segurança, estão sujeitos a novos e agressivos ataques. Entretanto pode-se afirmar que muitas medidas foram tomadas para garantir a integridade da plataforma, devido a preocupação com a manipulação dos dados de apenados e entidades. Como já expresse anteriormente, esta característica torna a necessidade de proteger os dados ainda mais relevante, evitando expor qualquer uma das partes envolvidas na utilização do sistema.

## 5.6.2 GERAÇÃO DE RELATÓRIOS COM INFORMAÇÕES DOS APENADOS

Outra funcionalidade implementada neste sistema, que foi requisitado pelas equipes técnicas da JF, é o sistema de geração de relatórios. Um perfil de técnico da JF ou de juiz, pode gerar relatórios das atividades dos apenados, contendo todas as suas informações básicas e também as atividades cumpridas ou cadastradas por ele no sistema. A seguir será desenvolvido um exemplo de como isso pode ser feito, bem como um exemplo do relatório gerado.

Quando acessado o sistema com os devidos acessos, o usuário que deseja gerar um relatório deve abrir a tela de Apenados, através do menu, e selecionar o apenado que deseja gerar as informações. Neste ponto, ao selecionar o botão correspondente às informações o usuário visualizará as informações do apenado, como apresentado na Figura 19.

**Apenado João**

**Informações Pessoais**

Nome Completo: Apenado João

Data de Nascimento: 10/04/1995      CPF: 23456789011      RG: 2345678

Telefone: +5541999999999      Email: apenado2@email.com

Endereço: Avenida Silva Jardim, n 1000

Cidade: Curitiba      Estado: PR      País: BRA

Habilidades: Motorista de veículos grandes, Serviços Gerais

**Informações da Pena**

Crimes: Transporte de drogas      Horas total da Pena: 150

Descrição do crime: Apenado sendo condenado por crime de transporte de drogas entre fronteiras. Réu primário.

**Resumo da Pena**

Horas Cumpridas:

Horas Reduzidas:

**Histórico de atividades**

Apenado João | Entidade ABC | Atividade Reforma  
Status: Em andamento

Fechar      Exportar Relatório

**Figura 19: Tela de Dados de um apenado**

**Fonte: Autoria Própria**

Todas essas informações serão disponibilizadas em um PDF, que será gerado através do clique no botão "Exportar Relatório". Este mecanismo de geração de relatórios possibilita para as equipes técnicas o armazenamento das informações dos apenados em documentos físicos, por exemplo, quando necessário. O Exemplo de PDF gerado pelo sistema está apresentado no Apêndice C

### 5.6.3 SISTEMA BASEADO EM REGRAS - SUGESTÃO DE ATIVIDADES

Parte importante da implementação deste projeto consistiu no sistema de sugestão de atividades para os apenados, que cria uma lista de atividades única para cada apenado, levando em consideração suas características.

Futuramente, o sistema receberá implementações de técnicas de *Machine Learning* que auxiliarão no reconhecimento facial dos apenados, validando as fotos inseridas no sistema durante os processos de *check-ins e check-outs*, limitando as possibilidades de fraudes nas execuções das penas. Entretanto, neste presente trabalho, implementou-se um sistema de tomada de decisão automática, um Sistema Baseado em Regras, que sugere uma ordem de prioridade das atividades para os apenados.

Um sistema baseado em regras, segundo Negnevitsky (NEGNEVITSKY, 2005), pode ser resumido em uma técnica que foi desenvolvida no começo dos anos 1970 e que é a fundação para os sistemas mais modernos de Inteligência Artificial. O mesmo consiste em um sistema de decisão que utiliza como base uma série de regras, traduzidas para o *software* em formato de Cláusulas SE, que ajudam o sistema a decidir qual é a escolha correta.

Neste projeto o sistema de decisões se baseou nas habilidades e crimes dos apenados para sugerir as atividades que mais se enquadrem ao seu perfil. Quando uma entidade cadastra uma nova atividade, é preciso que ela identifique quais são as habilidades necessárias para execução da tarefa e se existem crimes que devam ser excluídos de tais atividades. Na Figura 20a é possível verificar, à esquerda da imagem, um exemplo com três atividades cadastradas, lista de habilidades desejadas e a lista de crimes que devem ser excluídos. Neste exemplo, temos que uma pessoa que foi apreendida traficando ou transportando drogas, não pode se cadastrar para uma atividade de motorista, visto que, existe a possibilidade deste serviço ser utilizado indevidamente.

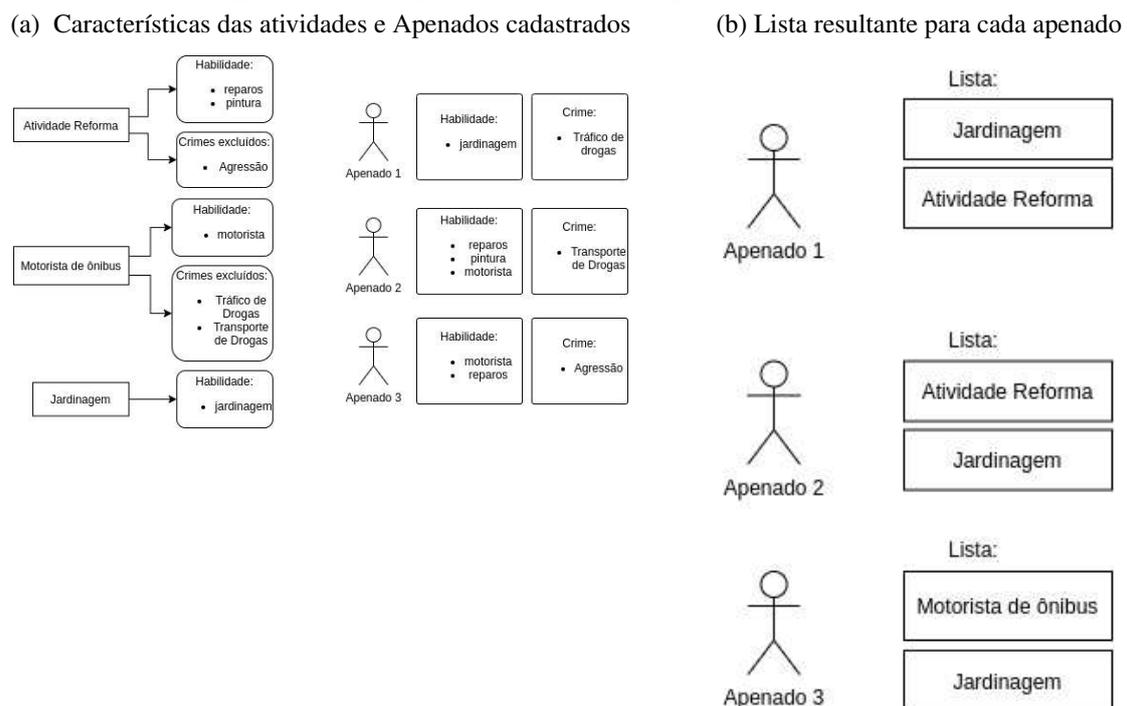
Em posse das informações de pena e habilidade, o *web service* elabora uma lista de atividades específica para cada apenado, que é requisitada enviando como parâmetro o seu CPF. Nesta lista o sistema exclui todas as atividades que possuem os crimes do apenado em sua lista de exclusão e coloca em ordem as atividades restantes baseadas na maior compatibilidade das habilidades.

Para exemplificar este processo, será utilizada a Figura 20a, onde tem-se o cadastro de 3 apenados, com suas listas de habilidades e seus respectivos crimes. Na Figura 20b pode-se verificar o resultado do sistema de sugestão para cada um dos

apenados. Para o apenado 1, devido a sua pena, não é disponibilizada a atividade de motorista e, por conta de suas habilidades com jardinagem, a atividade de jardinagem é priorizada no ordenamento de sua lista. O apenado 2, por sua vez, possui as mesmas restrições de pena do apenado 1, mas, devido a suas habilidades com reformas e pintura, as atividades aparecem com priorização diferente. Por fim, o apenado 3, não pode participar da atividade de reforma, pois sua pena consta na lista de exclusões da mesma, e o ordenamento das outras duas atividades se dá levando em consideração suas habilidades como motorista.

É importante destacar que a atividade de jardinagem, utilizada no exemplo, aparece para todos os apenados, pois não há uma restrição de pena em seu cadastro, entretanto, o ordenamento das atividades é personalizado para cada um dos apenados.

**Figura 20: Exemplo do sistema de sugestão de Atividades**



**Fonte: Autoria Própria**

#### 5.6.4 RELATÓRIOS GRÁFICOS

Por último, mas não menos importante, o sistema de geração de gráficos é uma das implementações adicionadas como forma de agregar ainda mais valor à utilização do sistema. Nesta tela, serão apresentados alguns gráficos sintetizando características de apenados, entidades e atividades. É possível gerar muitos gráficos através das informações existentes no banco de dados. Atualmente, os gráficos focam nas características de faixa etária dos apenados e também de suas habilidades.

Entende-se que, a partir dessa síntese de informações, as equipes da JF estarão mais preparadas para uma tomada de decisão. A partir de dados concretos relacionados as habilidades dos apenados, por exemplo, aumenta-se a possibilidade das equipes da JF realizarem campanhas de incentivo para que novas entidades se inscrevam e, a partir das informações das idades gerar atividades compatíveis com o público e sua faixa etária.

As Figuras 21, 22a e 22b são exemplos de como os gráficos são visíveis no sistema:

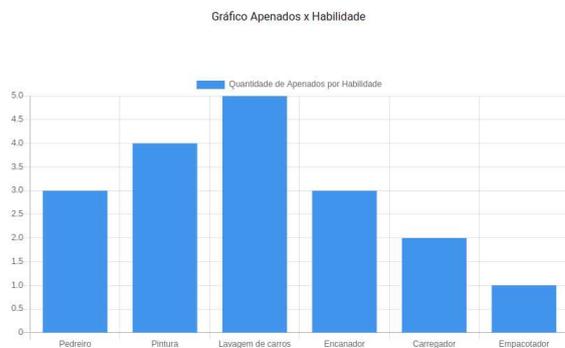


**Figura 21: Tela de Análise de Gráficos - Informações Gerais**

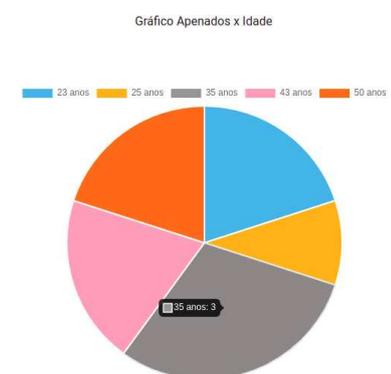
**Fonte: Autoria Própria**

**Figura 22: Relatórios Gráficos**

(a) Gráfico mostrando a quantidade de apenados por habilidade



(b) Gráfico mostrando a quantidade de apenados por idade



**Fonte: Autoria Própria**

## 6 RESULTADOS FINAIS

A partir do que foi apresentado nas seções anteriores, é possível discutir os resultados finais das aplicações aqui desenvolvidas. Com relação ao Banco de Dados e ao *Web Service*, os mesmos estão sendo consumidos tanto pela aplicação *Web* quanto pelo aplicativo para dispositivos móveis, e apresentam estabilidade na propagação e manipulação dos dados.

A respeito do Portal *Web*, além das telas já apresentadas anteriormente, é interessante destacar outras funcionalidades do sistema.

A Figura 23a e a Figura 23b apresentam as telas de Login e a tela de confirmação do celular, respectivamente, que será utilizada para confirmar o número do celular que receberá o SMS com o código de confirmação. Em seguida, um *pop-up* pedindo o código aparece e fica aguardando até que o mesmo seja inserido corretamente:

**Figura 23: Tela de Login**

(a) Entrada de credenciais                      (b) Confirmação do celular - Autenticação Multifator

The figure displays two screenshots of a mobile application's login process. Screenshot (a) is the credential entry screen, featuring a logo at the top, a 'Usuário' field containing '08697638522', a 'Senha' field with masked characters, an 'Entrar' button, and a link for 'Esqueceu sua senha?'. Screenshot (b) is the multi-factor authentication screen, showing a confirmation message, a 'Celular' field with the number '+5541999999999', and a 'Confirmar' button.

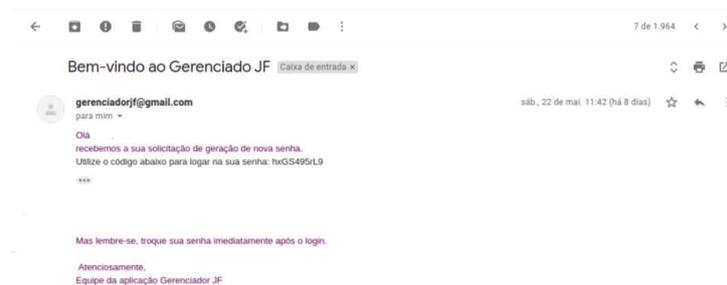
**Fonte: Aatoria Própria**

Um processo que é interessante de ser mencionado aqui é a geração automática de senhas e envio por email, tanto após a criação do usuário *mobile* e quanto do usuário do portal *Web*. O sistema gera automaticamente uma senha com 10 dígitos que é encaminhada para o email registrado no ato do cadastramento. Este mesmo processo é feito quando a pessoa seleciona no portal a opção de esquecimento de senha, onde ele mostra a tela apresentada na Figura 24a para que o usuário confirme o email que receberá a nova senha e, após confirmado, um email no formato do apresentada na Figura 24b é recebido no email indicado. Este processo é necessário para que, em momento nenhum, o juiz ou técnico da JF entrem em contato com a senha do usuário, evitando fraudes. Além disso, seguindo princípios de segurança, todas as senhas são salvas através de *Hash* utilizando o algoritmo *MD5* de criptografia, sendo que nenhuma senha em texto puro está armazenada nos sistemas.

### Figura 24: Fluxo de Esquecimento da senha

(a) Tela de confirmação do email  
para recuperação da senha

(b) Email recebido pelo usuário com senha aleatória

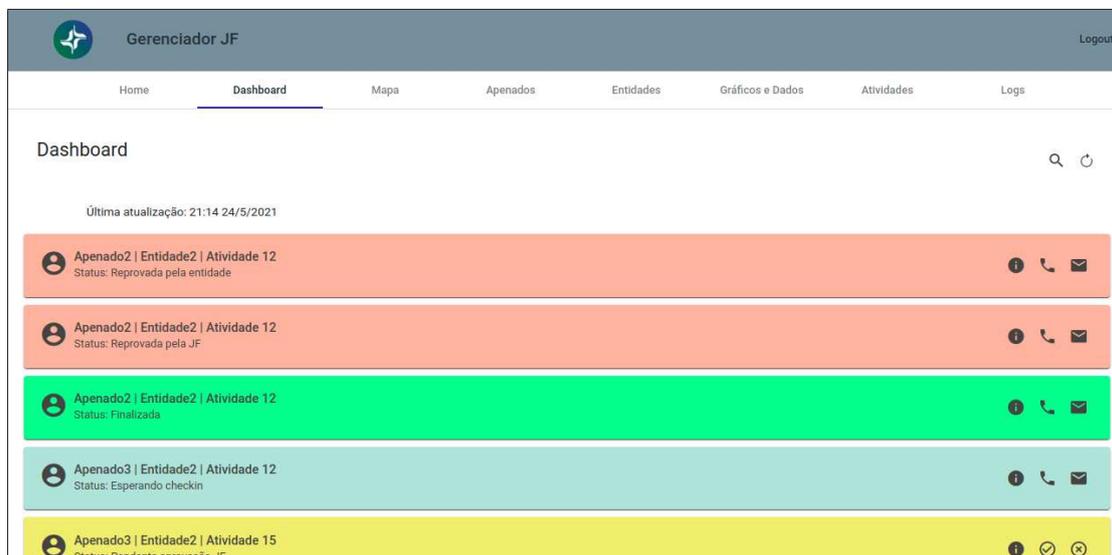


**Fonte: Autoria Própria**

Após logado, o sistema vai abrir a tela inicial referente ao usuário logado, o que já foi apresentado no item anterior, quando discutidas as questões de segurança deste sistema. A partir deste ponto, se o perfil logado pertence a um técnico da JF ou a um juiz, existem 4 telas que precisam ser apresentadas.

Na Figura 25 é apresentada a tela do *Dashboard* que recebe todas as informações de escalas, durante todo o seu ciclo de vida, ou seja, desde que são criadas, aprovadas pela JF, realizado o *check-in*, etc. Sobre a escolha das cores que são disponibilizadas em cada uma das atividades, ela é configurada de acordo com o andamento das mesmas. O *Dashboard* possui quatro cores: vermelho, amarelo, verde e azul. Quando uma atividade está em amarelo significa que alguma ação precisa ser tomada, indicando necessariamente aprovações pendentes, seja por parte da JF ou mesmo das entidades. A cor vermelha indica que algo está errado, ou a atividade foi reprovada em alguma das etapas de aprovação ou o seu *check-in* ainda não foi realizado e seu horário de entrada já expirou, ou seja, indica um atraso. A cor azul é utilizada na atividade quando ela está em andamento ou aguardando a realização do *check-in*, que não está atrasado. Por último, a cor verde indica a finalização da atividade e de todo o processo de aprovação, sendo que ela é configurada apenas ao final do processo

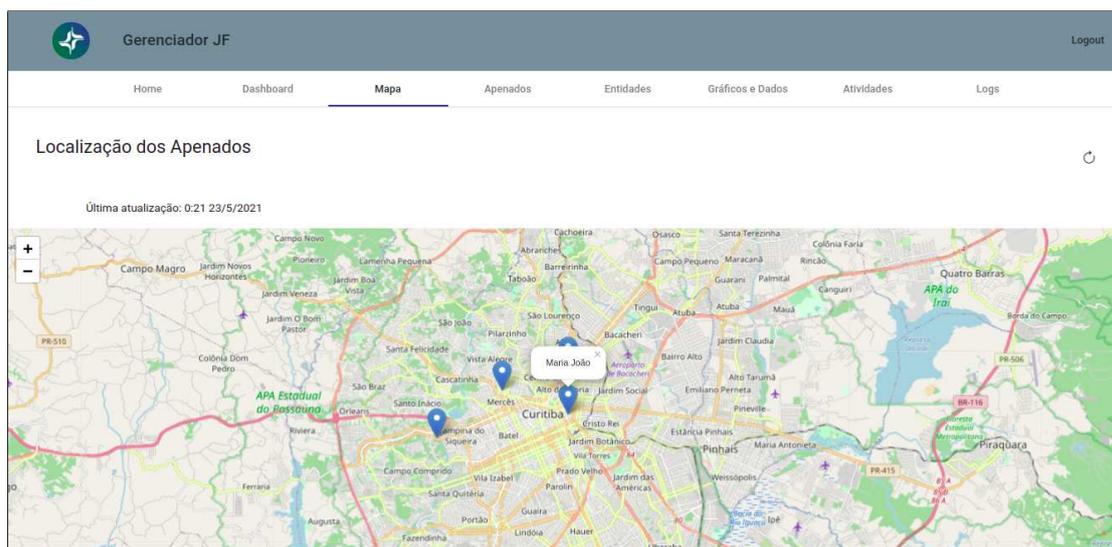
de aprovação da entidade. As cores foram escolhidas para sinalizar visualmente para a equipe técnica o que está acontecendo no momento da visualização da lista, tornando a identificação mais simples.



**Figura 25: Tela do *Dashboard***

**Fonte: Autoria Própria**

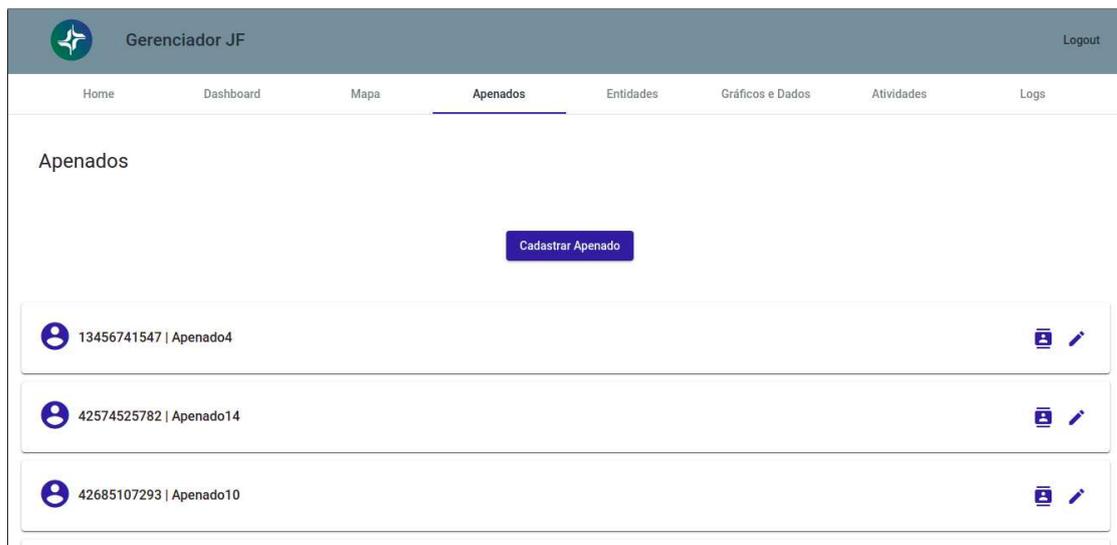
Além disso, logado a partir desses perfis, é possível visualizar o Mapa, apresentando na Figura 26. No mapa é possível saber a localização exata do apenado que está naquele momento prestando alguma atividade. É preciso destacar que é possível identificar o nome do apenado simplesmente clicando em cima do ícone.



**Figura 26: Tela do Mapa contendo as localizações dos apenados no momento da realização das atividades agendadas**

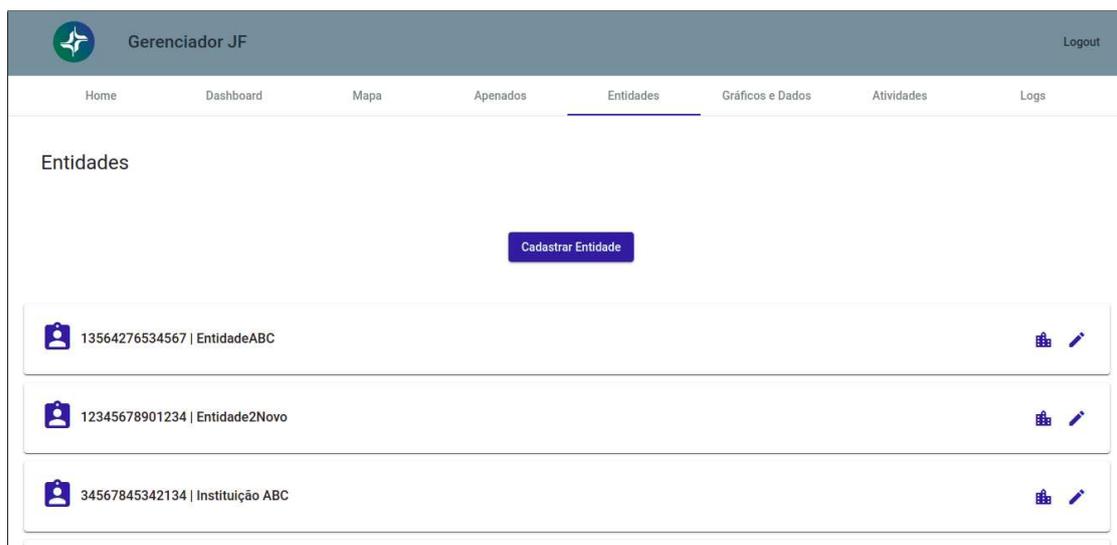
**Fonte: Autoria Própria**

Também é possível visualizar as informações dos apenados e das entidades, as quais são apresentadas da maneira como está representado nas Figuras 27 e 28, respectivamente. A organização em lista favorece a visualização e torna mais simples as atividades do dia a dia. Além disso, nos botões disponíveis no final do *card* é possível verificar todas as informações de apenados e entidades e também atualizá-las.



**Figura 27: Tela dos Apenados**

**Fonte: Autoria Própria**

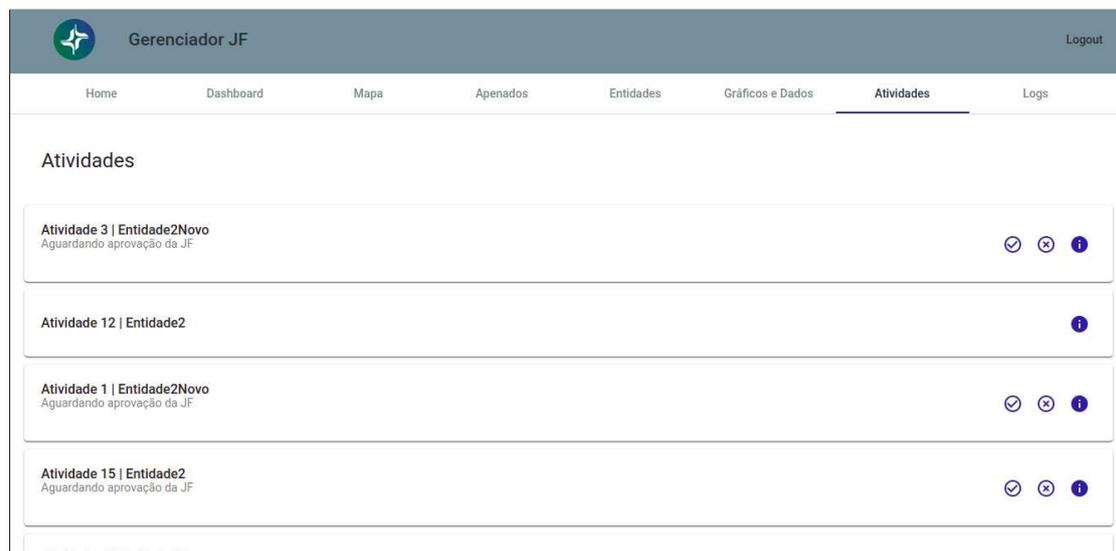


**Figura 28: Tela das Entidades**

**Fonte: Autoria Própria**

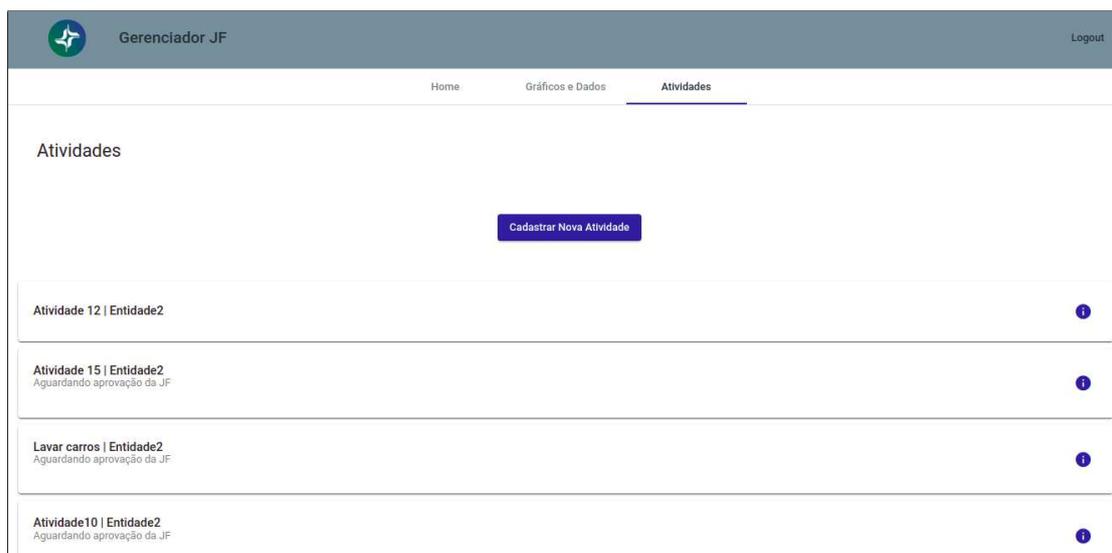
Outra tela importante é a tela que reúne as atividades. Na Figura 29 é possível

visualizar todas as atividades disponíveis no sistema no momento da consulta, pois este é um exemplo de um perfil de técnico ou juiz. Mas, como mostra a Figura 30, na tela de um perfil com cadastro de entidade, é possível visualizar apenas as atividades referentes às suas atividades.



**Figura 29: Tela das Atividades - visão de um Juíz ou Técnico da JF**

**Fonte: Autoria Própria**



**Figura 30: Tela das Atividades - visão de uma entidade**

**Fonte: Autoria Própria**

Dois telas, mencionadas anteriormente, são interessantes de serem destacadas: a tela dos relatórios gráficos, que sintetiza graficamente as características dos apenados, entidades e atividades no banco de dados e está disponível para todos os perfis, com um diferencial para o perfil das entidades, que pode visualizar apenas gráficos relacionados às suas atividades e não recebe informações dos apenados; e a tela de Logs, que mostra todos os registros de logs do sistema, mas só pode ser visualizada por um perfil com privilégio de juiz.

## 7 CONSIDERAÇÕES FINAIS

Este trabalho se propôs a desenvolver um sistema que facilita o gerenciamento das penas e apenados que se incluem na modalidade de Prestação de Serviços à Comunidade. O sistema apresentado oferece mecanismos eficientes para o controle de dados dos apenados e das entidades envolvidas, ferramentas de monitoramento em tempo real das atividades que estão sendo realizadas pelos apenados, assim como também o registro das que foram concluídas. Além disso, o sistema oferece uma rotina de etapas bem definidas para a aprovação de cada uma das atividades requisitadas pelas entidades antes de serem encaminhadas aos apenados, e possibilita também, após a finalização das mesmas, o cálculo automatizado das horas cumpridas. Considerando que tudo isso é realizado por meio de um computador, o sistema proposto se apresenta também como um meio eficiente de comunicação entre as entidades, equipes técnicas e apenados.

A partir do estudo de caso construído por meio de reuniões com a equipe técnica da JF da 4ª Vara de Foz do Iguaçu-PR e seu juiz responsável, foi possível entender os problemas e necessidades que um sistema como o proposto deve cobrir e traçar os requisitos funcionais aqui apresentados. Após pesquisas bibliográficas, entendeu-se que o problema da aplicação das PSCs no Brasil precisa ser discutido e necessita de novas ferramentas para efetivamente obter um maior aproveitamento desta modalidade de pena, a qual se mostra importante para a reinserção e reintegração do apenado à sociedade, além de ser uma forma do apenado ressarcir em certa maneira a sociedade civil pelo crime cometido.

Por meio do portal *web*, é possível realizar a inserção de dados necessários para o funcionamento do ambiente, disponibilizando formulários adequados para o cadastro de apenados, entidades e novos usuários, sendo que este último só é possível de ser feito a partir de um perfil com privilégio de juiz. Os cadastramentos possibilitam um controle maior da identificação de todas as partes envolvidas nas prestações das PSCs e também auxilia na geração de estatísticas que caracterizam a região, no que diz respeito a esta modalidade de pena. Atualmente, este levantamento de dados se mostra problemático para as equipes técnicas, visto que não há um sistema centralizador de dados ou mesmo uma padronização dos formatos dos armazenamentos, resultando em uma baixa geração de informações úteis, principalmente quando pesquisa-se a nível nacional ou mesmo estadual.

Através do *Dashboard* e do mapa, o portal oferece às equipes técnicas um monitoramento em tempo real das atividades em andamento, concluídas e agendadas. Estes recursos apresentam, em formato visualmente objetivo, as informações necessárias para gerenciar as atividades em execução, de forma que o técnico que está monitorando consiga identificar com facilidade as informações que procura. Através do mapa, é possível inclusive verificar quais são as localizações das atividades em execução no momento da visualização. Este recurso se mostra imprescindível para o dia a dia das entidades e das equipes técnicas, visto que, perante a lei, é de responsabilidade destas duas instituições a gestão e garantia de que a pena está sendo cumprida corretamente.

Em concordância com o que foi discutido sobre a geração de estatísticas sobre as PSCs, o sistema oferece formas de geração automática de relatórios. Este foi um requisito muito enfatizado pelas equipes no início da pesquisa e se mostra essencial para suas atividades. Atualmente, existem duas formas de gerar relatórios dentro do sistema:

(i) A primeira delas é gerar relatórios individuais de cada apenado, que trazem todas as suas informações pessoais, os dados da sua pena e também o seu histórico de atividades, além de um resumo da situação do cumprimento da pena no momento da

geração do relatório. Este documento representa um importante comprovante gerado pelo sistema que pode ser usado para atestar que as atividades foram cumpridas e que o indivíduo quitou sua dívida com a justiça.

(ii) A segunda forma são os relatórios gráficos que são gerados através dos registros diários de utilização do sistema e sintetiza, em forma de gráficos, as informações dos apenados, suas atividades e habilidades, por exemplo. As informações geradas por esta forma podem ser exploradas quando as equipes técnicas estiverem planejando novas atividades para os apenados, políticas públicas ou mesmo programas de incentivo. Ou seja, essa segunda forma pode auxiliar tomadas de decisão, tendo como base as características dos cadastros do sistema.

É também pelo portal *web* que os representantes das entidades interessadas em receber um apenado, após cadastramento, conseguem adicionar as atividades disponíveis para serem executadas pelos apenados. As mesmas, após aprovadas no processo de aprovação da JF, chegam aos apenados através dos aplicativos para dispositivos móveis. Dessa forma, as vagas são divulgadas efetivamente para quem precisa receber e os apenados têm um rápido acesso a tudo que há disponível para que ele cumpra sua pena.

Seguindo para as etapas mais especializadas de implementação e mantendo as atenções na divulgação de atividades, o sistema especializado baseado em regras se mostrou um primeiro passo para a implementação de mecanismos inteligentes neste sistema. O mesmo fornece uma lista de atividades única para cada um dos apenados levando em consideração sua pena, ou seja, excluindo previamente todas as atividades que não podem ser cumpridas por ele e ordenando as atividades restantes de forma a priorizar aquelas que demandam habilidades específicas do apenado em questão. A partir desse ordenamento prioritário, entende-se que o apenado terá maior facilidade em se inscrever e cumprir as atividades, resultando em um cumprimento mais rápido e efetivo da sua pena.

Por último, mas não menos importante, este sistema apresenta mecanismos de segurança que foram planejados e desenvolvidos para suprir as três grandes áreas da

segurança, que são: autenticação, autorização e auditoria. Para acessar o sistema, é preciso passar por um processo de autenticação multi fator, que confirma não apenas as credenciais fornecidas pelo usuário, mas que também autentica o mesmo via uma entrada de *token* encaminhada via SMS para o celular cadastrado. Este tipo de autenticação leva em consideração um conhecimento prévio do usuário, as credenciais, e um objeto pertencente a ele, o aparelho celular. A autenticação multi fator é um mecanismo que auxilia na prevenção de acessos indevidos ao sistema. Além disso, em relação a autenticação, os sistema do *web service* e do banco de dados também são acessados via uso de credenciais. No que diz respeito à autorização, tem-se um sistema baseado em 3 tipos de perfis: juiz, técnico administrativo e representante de entidade. Cada um deles tem níveis de acesso e autorização para visualizarem seções diferentes do portal. Por exemplo, com relação às atividades cadastradas no sistema, enquanto um juiz e um técnico conseguem visualizar todas as atividades e também aprovar ou reprovar as mesmas, um perfil de entidade consegue visualizar apenas as atividades que foram cadastradas pela sua entidade.

Finalmente, temos a auditoria, que neste sistema se apresenta na forma de logs, que são gerados em determinadas ações e que rastreiam as ações que são executadas dentro do sistema salvando no banco de dados todas as informações do usuário e qual foi a ação tomada. Estes registros só conseguem ser visualizados utilizando um perfil de juiz e, em casos de suspeitas de fraudes ou ações indevidas, estes registros devem ser cuidadosamente analisados e processados com o intuito de encontrar o responsável.

Com relação a trabalhos futuros, a implementação de processos utilizando *Machine Learning* para identificação e validação das fotos inseridas no sistema durante os processos de *check-in* e *check-out* é uma das mais promissoras propostas que podem ser desenvolvidas para melhorar os processos de autenticação e rastreabilidade do sistema. Através de reconhecimento facial, o sistema automaticamente poderá dizer se uma foto não compatível com aquela registrada naquele determinado login foi enviada para o cumprimento da atividade. Outra possibilidade de trabalho futuro é a ampliação dos fatores de segurança, entre eles a implementação de autenticação baseada em

biometria, o que tornaria o sistema ainda mais seguro e robusto em relação aos acessos. A respeito dos resultados gráficos do sistema, era parte do planejamento inicial passar os *mockups* pela análise da equipe de *design* do laboratório, entretanto, esta atividade teve que ser adiada e ficará para as etapas futuras do projeto. Este tipo de análise se mostra muito importante, não se limitando apenas às características estéticas da aplicação, mas podendo também verificar demandas de acessibilidade e inclusão do sistema, tornando-o acessível a todas as pessoas e adaptando-o quando necessário.

## REFERÊNCIAS

AFONSO, A.; FARIA, T. **Fullstack Angular e Spring - Guia para se tornar um desenvolvedor moderno**. [S.l.]: AlgaWorks Softwares, Treinamentos e Serviços Ltda., 2018.

ALBINADER NETO, J. A. **Web Services em Java**. [S.l.]: Brasport Livros e Multimídia Ltda, 2006.

ANGELIM, A. C. d. C. L. Penas alternativas como instrumento de reintegração social do apenado no sistema penal brasileiro. **Âmbito Jurídico**, 2016. Disponível em: <<https://ambitojuridico.com.br/cadernos/direito-penal/penas-alternativas-como-instrumento-de-reintegracao-social-do-apanado-no-sistema-penal-brasileiro/>>. Acesso em: 30 de Junho de 2021.

ANGULAR, E. **What is Angular**. 2009. Disponível em: <<https://angular.io/guide/what-is-angular>>. Acesso em: 09 de Abril de 2021.

BERNERS-LEE, T. **Information Management: A Proposal - W3C**. 1989. Disponível em: <<https://www.w3.org/History/1989/proposal.html>>. Acesso em: 31 de Abril de 2021.

BRASIL. **Constituição de 1988. Artigo 43 do Decreto Lei nº 2.848 de 07 de Dezembro de 1940**. 1940. Disponível em: <<https://www.jusbrasil.com.br/topicos/10635539/artigo-43-do-decreto-lei-n-2848-de-07-de-dezembro-de-1940>>. Acesso em: 31 de Janeiro de 2021.

BRASIL. **Constituição de 1988. Artigo 44 do Decreto Lei nº 2.848 de 07 de Dezembro de 1940**. 1940. Disponível em: <<https://www.jusbrasil.com.br/topicos/10635539/artigo-43-do-decreto-lei-n-2848-de-07-de-dezembro-de-1940>>. Acesso em: 31 de Janeiro de 2021.

BRASIL. **Constituição de 1988. Artigo 150 da Lei nº 7.210 de 11 de Julho de 1984**. 1984. Disponível em: <<https://www.jusbrasil.com.br/busca?q=Art.+150+da+Lei+de+Execu%C3%A7%C3%A3o+Penal++Lei+7210%2F84#:text=150.,sobre%20aus%C3%Aancia%20ou%20falta%20disciplinar.>>. Acesso em: 31 de Janeiro de 2021.

CAVALCANTE, I. R. Capacidade de ressocializaÇÃo das penas alternativas: prestação de serviço à comunidade. **Tribunal de Justiça do Piauí**, 2016. Disponível em: <<http://www.tjpi.jus.br/site/uploads/htmlcontent/1057.pdf>>. Acesso em: 22 de Maio de 2021.

CLARO, D. B.; SOBRAL, J. B. M. **Programação em Java**. [S.l.]: Pearson Education., 2008.

CODD, E. F. A relational model of data for large shared data banks. **Communications of the ACM**, v. 13, n. 6, 1970. Disponível em: <<https://www.seas.upenn.edu/~zives/03f/cis550/codd.pdf>>. Acesso em: 22 de Outubro de 2020.

COSTA, L. **Registro de uma vivência**. Rio de Janeiro: Empresa das Artes, 1995.

COULOURIS, G. et al. **Sistemas Distribuídos. Conceitos e Projeto**. [S.l.]: Editora Bookman Companhia Editorial Ltda, 2006.

DEPEN. **Depen atualiza dados sobre a população carcerária do Brasil**. 2020. Disponível em: <<https://www.gov.br/depen/pt-br/assuntos/noticias/depen-lanca-paineis-dinamicos-para-consulta-do-infopen-2019-1>>. Acesso em: 26 de Abril de 2020.

DUARTE, N. C. **Frameworks e Bibliotecas JavaScript**. Dissertação (Mestrado) — Instituto Superior de Engenharia do Porto, 2015. Disponível em: <[http://recipp.ipp.pt/bitstream/10400.22/8223/1/DM\\_NunoDuarte\\_2015\\_MEI.pdf](http://recipp.ipp.pt/bitstream/10400.22/8223/1/DM_NunoDuarte_2015_MEI.pdf)>. Acesso em: 31 de Abril de 2021.

EIS, D.; FERREIRA, E. **HTML5 e CSS3 com farinha e pimenta**. [S.l.]: Clube de Autores, 2012.

EULALIO, A. D.; CORDEIRO, D.; SOUZA, R. Web services: Integração de sistemas orientado a serviços com uma proposta de aplicação na ead. **Revista de Informática Aplicada**, v. 12, n. 2, 2016. Disponível em: <[https://www.seer.uscs.edu.br/index.php/revista\\_informatica\\_aplicada/article/view/6918](https://www.seer.uscs.edu.br/index.php/revista_informatica_aplicada/article/view/6918)>. Acesso em: 6 de Abril de 2021.

FIELDING, T. R. Architectural styles and the design of network-based software architectures. **International Journal of Scientific and Research Publications**, 2000. Disponível em: <<https://www.ics.uci.edu/~fielding/pubs/dissertation/abstract.htm>>. Acesso em: 6 de Abril de 2021.

FINK, G.; FLATOW, I. **Pro Single Page Application Development: Using Backbone.js and ASP.NET**. Nova Iorque: Apress Media LLC., 2014.

FRAGOSO, H. C. **Lições de direito penal, 1994. In: QUEIROZ, Paulo. Direito Penal, parte geral.** [S.l.]: Lumen Juris Editora, 2009.

GARCIA, V. S.; SOTTO, E. C. S. Comparativo entre os modelos de banco de dados relacional e não-relacional. **Revista Interface Tecnológica**, v. 16, n. 2, 2019. Disponível em: <<https://revista.fatectq.edu.br/index.php/interfacetecnologica/article/view/673>>. Acesso em: 22 de Outubro de 2020.

GOMES, C. A. d. C. Espaço urbano e criminalidade: uma breve visão do problema. **Revista de desenvolvimento Econômico**, v. 7, n. 11, 2005. Disponível em: <<https://revistas.unifacs.br/index.php/rde/article/viewFile/97/101>>. Acesso em: 15 de Abril de 2021.

INFOPEN. **Levantamento Nacional de Informações Penitenciárias - INFOPEN.** 2014. Disponível em: <[http://depen.gov.br/DEPEN/depen/sisdepen/infopen/infopen\\_dez14.pdf](http://depen.gov.br/DEPEN/depen/sisdepen/infopen/infopen_dez14.pdf)>. Acesso em: 26 de Abril de 2020.

LEVIN, T. E. et al. **Design Principles and Guidelines for Security.** [S.l.], 2007. Disponível em: <<https://apps.dtic.mil/sti/pdfs/ADA476035.pdf>>. Acesso em: 22 de Maio de 2021.

LÓSCIO, B. F.; OLIVEIRA, H. R. d.; PONTES, J. C. d. S. Nosql no desenvolvimento de aplicações web colaborativas. **VIII Simpósio Brasileiro de Sistemas Colaborativos**, v. 8, 2011. Disponível em: <<https://www.researchgate.net/publication/268201466>>. Acesso em: 22 de Outubro de 2020.

MAZIERO, C. A. **Sistemas Operacionais: Conceitos e Mecanismos.** [S.l.]: Editora da UFPR, 2013.

MICROSOFT, E. **Typescript - What is TypeScript?** [S.l.], 2012. Disponível em: <<https://www.typescriptlang.org/>>. Acesso em: 14 de Abril de 2021.

MOTTA, A. F. M. R. A dignidade da pessoa humana e sua definição. **Âmbito Jurídico**, 2013. Disponível em: <<https://ambitojuridico.com.br/cadernos/direitos-humanos/a-dignidade-da-pessoa-humana-e-sua-definicao/>>. Acesso em: 30 de Junho de 2021.

MUMBAIKAR, S.; PADIYA, P. Web services based on soap and rest principles. **International Journal of Scientific and Research Publications**, v. 3, 2013. Disponível em: <<https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.414.7936&rep=rep1&type=pdf>>. Acesso em: 6 de Abril de 2021.

NAVATHE; ELMASRI, R. **Sistemas de Banco de Dados.** [S.l.]: Pearson Universidades, 2010.

NEGNEVITSKY, M. **Artificial Intelligence: A Guide to Intelligent Systems**. [S.l.]: Pearson Education Limited, 2005.

SARLET, I. W. S. **Dignidade da Pessoa Humana e Direitos Fundamentais na Constituição Federal de 1988**. [S.l.]: Livraria do Advogado Editora, 2011.

SILBERSCHATZ, A.; KORTH, H. S. **Sistemas de Banco de Dados**. [S.l.]: Elsevier Brasil, 2006.

ZAKAS, N. **Professional JavaScript for Web Developers**. Rio de Janeiro: Wrox Press;, 2012.

ZORZO, R.; BERNARDI, E. F. F. . **ESTUDO E DESENVOLVIMENTO DE CASO DE USO COM FRAMEWORK ANGULARJS**. 2016. Disponível em: <<https://painel.passofundo.ifsul.edu.br/uploads/arq/20160331181740335398980.pdf>>. Acesso em: 13 de Abril de 2021.

## APÊNDICE A – CASOS DE USO DO SISTEMA

A seguir, são detalhados os casos de uso do sistema, que foram desenvolvidos levando em consideração o requisitos funcionais e o comportamento esperado do sistema aqui apresentado:

**Tabela 5: Caso de Uso 1 - Cadastrar Apenados e entidades**

<b>Nome</b>	<b>UC001 - Cadastrar Apenados e entidades</b>
<b>Atores</b>	Juiz e Técnicos da JF
<b>Descrição</b>	Caso de uso executado quando um juiz ou técnico da JF precisam cadastrar um novo apenado ou entidade.
<b>Pré-condições</b>	<ul style="list-style-type: none"> <li>- Juiz ou técnico devem estar logados no sistema;</li> <li>- Informações do Apenado ou Entidade preenchidas.</li> </ul>
<b>Pós-condições</b>	Apenado ou Entidade cadastrado no sistema.

**Tabela 6: Caso de Uso 2 - Visualizar Informações dos Apenados e entidades cadastrados**

<b>Nome</b>	<b>UC002 - Visualizar Informações dos Apenados e entidades cadastrados</b>
<b>Atores</b>	Juiz e Técnicos da JF
<b>Descrição</b>	Caso de uso executado quando um juiz ou técnico da JF precisam visualizar as informações das entidades e dos apenados cadastrados no sistema.
<b>Pré-condições</b>	<ul style="list-style-type: none"> <li>- Juiz ou técnico devem estar logados no sistema;</li> <li>- Existir ao menos um apenado ou uma entidade cadastrada no sistema;</li> </ul>
<b>Pós-condições</b>	Informações dos apenados ou entidades cadastrados no sistema são exibidos na tela.

**Tabela 7: Caso de Uso 3 - Visualizar Mapa e Dashboard**

<b>Nome</b>	<b>UC003 - Visualizar Mapa e Dashboard</b>
<b>Atores</b>	Juiz e Técnicos da JF
<b>Descrição</b>	Caso de uso executado quando um juiz ou técnico da JF precisam visualizar o <i>Dashboard</i> e o Mapa.
<b>Pré-condições</b>	<ul style="list-style-type: none"> <li>- Juiz ou técnico devem estar logados no sistema;</li> </ul>
<b>Pós-condições</b>	Mapa ou <i>Dashboard</i> são exibidos na tela

**Tabela 8: Caso de Uso 4 - Visualizar todas as atividades cadastradas**

<b>Nome</b>	<b>UC004 - Visualizar todas as atividades cadastradas</b>
<b>Atores</b>	Juiz e Técnicos da JF
<b>Descrição</b>	Caso de uso executado quando um juiz ou técnico da JF precisam visualizar todas as atividades disponíveis no sistema
<b>Pré-condições</b>	- Juiz ou técnico devem estar logados no sistema;
<b>Pós-condições</b>	As atividades cadastradas são exibidos na tela

**Tabela 9: Caso de Uso 5 - Gerar Relatórios**

<b>Nome</b>	<b>UC005 - Gerar Relatórios</b>
<b>Atores</b>	Juiz e Técnicos da JF
<b>Descrição</b>	Caso de uso executado quando um juiz ou técnico da JF precisam gerar relatórios sobre os apenados ou entidades e suas informações.
<b>Pré-condições</b>	- Juiz ou técnico devem estar logados no sistema;
<b>Pós-condições</b>	Relatório gerado e documento salvo na máquina local

**Tabela 10: Caso de Uso 6 - Aprovar escalas agendadas**

<b>Nome</b>	<b>UC006 - Aprovar escalas agendadas</b>
<b>Atores</b>	Juiz e Técnicos da JF
<b>Descrição</b>	Caso de uso executado quando o Técnico da JF ou o juiz precisam analisar e aprovar as escalas dos apenados.
<b>Pré-condições</b>	<ul style="list-style-type: none"> <li>- Juiz ou técnico devem estar logados no sistema;</li> <li>- Escala deve estar cadastrada no sistema.</li> </ul>
<b>Pós-condições</b>	Escala do apenado é aprovada ou reprovada

**Tabela 11: Caso de Uso 7 - Visualizar os gráficos da análise de dados do sistema**

<b>Nome</b>	<b>UC007 - Visualizar os gráficos da análise de dados do sistema</b>
<b>Atores</b>	Juiz, Técnicos da JF e Representantes de entidades
<b>Descrição</b>	Caso de uso executado quando o usuário precisa visualizar os gráficos sobre os apenados, suas habilidades e atividades recentes.
<b>Pré-condições</b>	<ul style="list-style-type: none"> <li>- Usuário devem estar logados no sistema;</li> </ul>
<b>Pós-condições</b>	Sistema exibe os gráficos gerados a partir do Banco de dados

**Tabela 12: Caso de Uso 8 - Inativar Apenados e Entidades**

<b>Nome</b>	<b>UC008 - Inativar Apenados e Entidades</b>
<b>Atores</b>	Juiz
<b>Descrição</b>	Caso de uso executado quando o Juiz precisa inativar um apenado ou uma entidade.
<b>Pré-condições</b>	<ul style="list-style-type: none"> <li>- Usuário com privilégio de Juiz devem estar logados no sistema;</li> <li>- Cadastro do apenado deve estar ativo</li> </ul>
<b>Pós-condições</b>	Apenado inativado no sistema

**Tabela 13: Caso de Uso 9 - Ver logs do sistema**

<b>Nome</b>	<b>UC009 - Ver logs do sistema</b>
<b>Atores</b>	Juiz
<b>Descrição</b>	Caso de uso executado quando o Juiz queira ver os logs salvos no sistema
<b>Pré-condições</b>	<ul style="list-style-type: none"> <li>- Usuário com privilégio de Juiz devem estar logados no sistema;</li> </ul>
<b>Pós-condições</b>	Sistema exibe os logs registrados no sistema

**Tabela 14: Caso de Uso 10 - Aprovar escalas finalizadas**

<b>Nome</b>	<b>UC010 - Aprovar escalas finalizadas</b>
<b>Atores</b>	Representantes da Entidade
<b>Descrição</b>	Após realizado o <i>check-out</i> a entidade deve aprovar a atividade.
<b>Pré-condições</b>	<ul style="list-style-type: none"> <li>- Usuário com privilégio de Entidade devem estar logados no sistema;</li> <li>- <i>Check-out</i> realizado por apenado que cumpriu escala na entidade em questão.</li> </ul>
<b>Pós-condições</b>	Escala aprovada ou reprovada pela entidade.

**Tabela 15: Caso de Uso 11 - Adicionar Nova atividade**

<b>Nome</b>	<b>UC011 - Adicionar Nova atividade</b>
<b>Atores</b>	Representante da Entidade
<b>Descrição</b>	Caso de uso executado quando o Representante de uma entidade quer adicionar uma nova atividade
<b>Pré-condições</b>	<ul style="list-style-type: none"> <li>- Usuário com privilégio de Entidade devem estar logados no sistema;</li> </ul>
<b>Pós-condições</b>	Atividade cadastrada no sistema.

**Tabela 16: Caso de Uso 12 - Aprovar atividades cadastradas**

<b>Nome</b>	<b>UC012 - Aprovar atividades cadastradas</b>
<b>Atores</b>	Juiz e Técnicos da JF
<b>Descrição</b>	Caso de uso executado quando uma entidade cadastra nova atividade no sistema, sendo necessário a aprovação da JF para que ela seja disponibilizada para os apenados.
<b>Pré-condições</b>	<ul style="list-style-type: none"> <li>- Usuário com privilégio de Juiz ou Técnico da JF devem estar logados no sistema;</li> <li>- Atividade nova cadastrada no sistema por uma entidade.</li> </ul>
<b>Pós-condições</b>	Atividade aprovada e divulgada para os apenados

**Tabela 17: Caso de Uso 13 - Visualizar as atividades cadastradas pela sua entidade**

<b>Nome</b>	<b>UC013 - Visualizar as atividades cadastradas pela sua entidade</b>
<b>Atores</b>	Representante da Entidade
<b>Descrição</b>	Caso de uso executado quando o Representante de uma entidade quer visualizar as atividades cadastradas por sua entidade.
<b>Pré-condições</b>	<ul style="list-style-type: none"> <li>- Usuário com privilégio de Entidade devem estar logados no sistema;</li> </ul>
<b>Pós-condições</b>	Atividade da entidade cadastradas no sistema são exibidas na tela.

## **APÊNDICE B – FLUXO BÁSICO DOS CASOS DE USO DO SISTEMA**

Neste apêndice são apresentados os Fluxos Básicos de execução do sistema para cada Caso de Uso apresentado no Apêndice A.

**Tabela 18: Fluxo Básico - Caso de Uso 1**

<b>Fluxo Básico - UC001</b>	
<b>Ações dos atores</b>	<b>Ações do sistema</b>
1 - Usuários com privilégio de Juiz ou técnicos logam no sistema com sucesso	
2 - Usuários selecionam a página Apenados ou Entidades (de acordo com o que desejam cadastrar)	
	3 - Sistema abre a página selecionada
4 - Usuário seleciona o botão para Adicionar cadastro	
	5 - Sistema abre o pop-up de cadastro
6 - Usuário preenche as informações de cadastro	
7 - Usuário seleciona a opção “Salvar”	
	8 - Sistema salva os dados no banco de dados, incluindo o log desta ação.
	9 - Sistema fecha a janela de cadastro.
	10 - Sistema retorna uma mensagem informando o sucesso no cadastramento
<b>Regras de Negócio</b>	
[RF10] Cadastrar apenado [RF16] Cadastrar nova entidade [RF08] Gerar Logs	
<b>Fluxo de exceção</b>	
<b>Ações dos atores</b>	<b>Ações do sistema</b>
	8.a - Caso haja algum campo obrigatório faltando, o sistema não salva os dados e mostra uma mensagem indicando o que está errado.
	10.a - Caso aconteça algum erro com o salvamento, o sistema mostra uma mensagem de erro pedindo ao usuário que tente novamente em alguns minutos.

**Tabela 19: Fluxo Básico - Caso de Uso 2**

<b>Fluxo Básico - UC002</b>	
<b>Ações dos atores</b>	<b>Ações do sistema</b>
1 - Usuários com privilégio de Juiz ou técnicos logam no sistema com sucesso	
2 - Usuários selecionam a página Apenados ou Entidades (de acordo com o que desejam visualizar)	
	3 - Sistema abre a página selecionada
	4 - Sistema exibe todos os apenados ou entidades cadastrados no sistema
5 - Usuário seleciona o Apenado ou Entidade que deseja visualizar as informações	
	6 - Sistema abre um <i>pop up</i> com os dados
7 - Usuário seleciona o botão “Fechar”	
	8 - Sistema fecha a janela de visualização
<b>Regras de Negócio</b>	
[RF14] Exibir registros de apenados	
[RF19] Exibir registros de entidade	
[RF29] Divulgação das Informações dos apenados	
[RF30] Divulgação das Informações dos entidades	

**Tabela 20: Fluxo Básico - Caso de Uso 3**

<b>Fluxo Básico - UC003</b>	
<b>Ações dos atores</b>	<b>Ações do sistema</b>
1 - Usuários com privilégio de Juiz ou técnicos logam no sistema com sucesso	
2 - Usuários selecionam a página <i>Dashboard</i> ou Mapa (de acordo com o que desejam visualizar)	
	3 - Sistema abre a página selecionada
	4 - Sistema exibe o <i>Dashboard</i> (com todas as informações de escalas) ou o Mapa com as localizações dos apenados.
<b>Regras de Negócio</b>	
[RF20] Visualizar escala do apenado [RF24] Visualizar <i>Dashboard</i> de atividade [RF28] Visualizar Mapa [RF32] Divulgação das Informações das aprovações das escalas	

**Tabela 21: Fluxo Básico - Caso de Uso 4**

<b>Fluxo Básico - UC004</b>	
<b>Ações dos atores</b>	<b>Ações do sistema</b>
1 - Usuários com privilégio de Juiz ou técnicos logam no sistema com sucesso	
2 - Usuários selecionam a página Atividades	
	3 - Sistema abre a página selecionada
	4 - Sistema exibe as atividades cadastradas no sistema
5 - Usuário seleciona uma atividade para visualizar mais detalhes	
	6 - Sistema abre um pop up com os detalhes da atividade
7 - Usuário seleciona o botão “Fechar”	
	8 - Sistema fecha o pop up
<b>Regras de Negócio</b>	
[RF27] Exibir registros das atividades cadastradas	
[RF31] Divulgação das Informações das atividades voluntárias disponíveis	

Tabela 22: Fluxo Básico - Caso de Uso 5

<b>Fluxo Básico - UC005</b>	
<b>Ações dos atores</b>	<b>Ações do sistema</b>
1 - Usuários com privilégio de Juiz ou técnicos logam no sistema com sucesso	
2 - Usuários selecionam a página Apenados	
	3 - Sistema abre a página selecionada
	4 - Sistema exibe a lista de apenados
5 - Usuário seleciona qual usuário quer gerar o relatório	
	6 - Sistema abre um <i>pop-up</i> com as informações do apenado
7 - Usuário seleciona o botão "Exportar relatório"	
	8 - Sistema gera o arquivo e exibe em outra tela, além de fazer o download para a máquina local do usuário.
<b>Regras de Negócio</b>	
[RF15] Gerar relatório de apenado	

Tabela 23: Fluxo Básico - Caso de Uso 6

<b>Fluxo Básico - UC006</b>	
<b>Ações dos atores</b>	<b>Ações do sistema</b>
1 - Usuários com privilégio de Técnico ou Juiz loga no sistema com sucesso	
2 - Usuários selecionam a página <i>Dashboard</i>	
	3 - Sistema abre a página selecionada
	4 - O sistema exibe todas as escalas com seus status
5 - Usuário seleciona uma escala ainda não aprovada e seleciona para abrir suas informações	6- Sistema abre um <i>pop-up</i> com os dados da escala
7- Usuário Analisa a escala	
8 - Usuário seleciona o botão com um ícone que representa aprovação	
	9 - Sistema salva os dados no banco de dados, incluindo o log desta ação.
	Sistema retorna a mensagem informando que a escala foi aprovada com sucesso
<b>Regras de Negócio</b>	
[RF21] Trocar as escalas dos apenados	
[RF22] Aprovar as escalas dos apenados	
[RF24] Visualizar <i>Dashboard</i> de atividade	
[RF32] Divulgação das Informações das aprovações das escalas	
<b>Fluxo de exceção</b>	
<b>Ações dos atores</b>	<b>Ações do sistema</b>
8.a - Usuário seleciona o botão com um ícone que representa reprovação da escala	
	9.a - O sistema registra no banco de dados a escala como reprovada.

**Tabela 24: Fluxo Básico - Caso de Uso 7**

<b>Fluxo Básico - UC007</b>	
<b>Ações dos atores</b>	<b>Ações do sistema</b>
1 - Usuários logado no sistema com sucesso	
2 - Usuários selecionam a página Análise de dados	
	3 - Sistema abre a página selecionada
	4 - Sistema exibe os gráficos dos dados.

Tabela 25: Fluxo Básico - Caso de Uso 8

<b>Fluxo Básico - UC008</b>	
<b>Ações dos atores</b>	<b>Ações do sistema</b>
1 - Usuário com privilégio de Juiz logado no sistema com sucesso	
2 - Usuários selecionam a página Apenados ou Entidades, de acordo com o que deseja desativar.	
	3 - Sistema abre a página selecionada
	4 - Sistema exibe todos os apenados ou entidades cadastrados no sistema
5 - Usuário seleciona qual apenado ou entidade deseja desativar.	
6 - Usuário seleciona o botão "Edição" do apenado ou entidade selecionado.	
	7 - Sistema abre <i>pop-up</i> de edição.
8 - Usuário seleciona o botão "inativar"	
	9 - Sistema inativa o registro no banco de dados
	10 - Sistema fecha a tela de edição
	11 - Sistema mostra uma mensagem indicando sucesso na ação
<b>Regras de Negócio</b>	
[RF12] Inativar cadastro de apenado	
[RF18] Inativar cadastro de entidade	
<b>Fluxo de exceção</b>	
<b>Ações dos atores</b>	<b>Ações do sistema</b>
	11.a - Caso aconteça algum erro ao inativar, o sistema exibirá uma mensagem indicando para o usuário tentar novamente mais tarde.

**Tabela 26: Fluxo Básico - Caso de Uso 9**

<b>Fluxo Básico - UC009</b>	
<b>Ações dos atores</b>	<b>Ações do sistema</b>
1 - Usuário com privilégio de Juiz loga no sistema com sucesso	
2 - Usuários selecionam a página Logs	
	3 - Sistema abre a página selecionada
	4 - Sistema exibe todos os logs do sistema
<b>Regras de Negócio</b>	
[RF09] Exibir registros de logs	

**Tabela 27: Fluxo Básico - Caso de Uso 10**

<b>Fluxo Básico - UC010</b>	
<b>Ações dos atores</b>	<b>Ações do sistema</b>
1 - Usuários com privilégio de Entidade logam no sistema com sucesso	
	2 - O sistema exibe a página inicial com as escalas relacionadas a entidade logada. Incluindo as atividades pendentes de aprovação.
3 - Entidade seleciona a escala que quer revisar. seleciona o botão de informações da escala	
	4 - Sistema abre um <i>pop-up</i> com as informações da escala.
5 - Usuário analisa os dados	
6 - Usuário seleciona o botão de aprovação da escala	
	7 - Sistema salva os dados no banco de dados.
	8 - Sistema retorna uma mensagem de confirmação do fim do processo de aprovação
<b>Regras de Negócio</b>	
[RF23] Aprovar as escalas finalizadas	
<b>Fluxo de exceção</b>	
<b>Ações dos atores</b>	<b>Ações do sistema</b>
6.a - O usuário pode selecionar o botão de reprovação da escala	
	8.a - O sistema registra no banco de dados a escala como reprovada.

**Tabela 28: Fluxo Básico - Caso de Uso 11**

<b>Fluxo Básico - UC011</b>	
<b>Ações dos atores</b>	<b>Ações do sistema</b>
1 - Usuários com privilégio de Entidade logam no sistema com sucesso	
2 - Usuários selecionam a página Atividades	
	3 - Sistema abre a página selecionada
4 - Usuário seleciona o botão “Adicionar nova atividade”	
	5 - Sistema abre o <i>pop-up</i> de cadastro das atividades
6 - Usuário preenche os dados da atividade (nome, descrição, horas da atividade por dia, quantidade de apenados necessários, Início e fim do período, habilidades necessárias para a atividade e as penas excluídas).	
7 - Usuário seleciona o botão “Salvar”	
	8 - Sistema salva as informações no Banco de dados
	9 - Sistema retorna mensagem de confirmação para o usuário
<b>Regras de Negócio</b>	
[RF25] Cadastrar novas atividades	
<b>Fluxo de exceção</b>	
<b>Ações dos atores</b>	<b>Ações do sistema</b>
7.a - Caso os campos obrigatórios (nome, descrição, horas da atividade por dia, quantidade de apenados necessários, Início e fim do período) não estejam preenchidos, o sistema não permite o salvamento.	
	9a - Caso algum error ocorra durante o salvamento, o sistema retorna mensagem de erro.

**Tabela 29: Fluxo Básico - Caso de Uso 12**  
**Fluxo Básico - UC012**

<b>Fluxo Básico - UC012</b>	
<b>Ações dos atores</b>	<b>Ações do sistema</b>
1 - Usuários com privilégio de Técnico ou Juiz loga no sistema com sucesso	
2 - Usuários selecionam a página Atividades	
	3 - Sistema abre a página selecionada
	4 - O sistema exibe todas as atividades, incluindo aquelas que ainda não foram aprovadas.
5 - Usuário seleciona uma atividade ainda não aprovada	
6 - Usuário Analisa a atividade	
7 - Usuário seleciona o botão com um ícone que representa aprovação	
	8 - Sistema salva os dados no banco de dados, incluindo o log desta ação.
	9 - Sistema retorna a mensagem informando que a escala foi aprovada com sucesso
<b>Regras de Negócio</b>	
[RF26] Aprovar as atividades cadastradas	
<b>Fluxo de exceção</b>	
<b>Ações dos atores</b>	<b>Ações do sistema</b>
7.a - Usuário seleciona o botão com um ícone que representa reprovação da atividade.	
	8.a - O sistema registra no banco de dados a atividade como reproitada.

Tabela 30: Fluxo Básico - Caso de Uso 13

<b>Fluxo Básico - UC013</b>	
<b>Ações dos atores</b>	<b>Ações do sistema</b>
1 - Usuários com privilégio de Entidade logam no sistema com sucesso	
2 - Usuários selecionam a página Atividades	
	3 - Sistema abre a página selecionada
	4 - Sistema exibe as atividades cadastradas no sistema da entidade.
5 - Usuário seleciona uma atividade para visualizar mais detalhes.	
	6 - Sistema abre um <i>pop-up</i> com os detalhes da atividade
7 - Usuário seleciona o botão “Fechar”	
t1	8 - Sistema fecha a janela de cadastro de atividade.
<b>Regras de Negócio</b>	
[RF27] Exibir registros das atividades cadastradas	
[RF31] Divulgação das Informações das atividades voluntárias disponíveis	

**APÊNDICE C - EXEMPLO DE UM RELATÓRIO GERADO PELO SISTEMA**

# Relatório de Apenado

## Informações Pessoais

Nome Completo		Data de Nascimento
Apenado2		10/04/1995
CPF	RG	Nacionalidade
23456789011	2345678	brasileiro
Endereço		
Avenida Silva Jardim, n 1000		
Bairro	Cidade	UF
Rebouças	Curitiba	PR
Email		Telefone
apenado2@email.com		+5541999999999

## Habilidades

Habilidade3, Habilidade5

## Informações da Pena

Pena3, Pena

## Situação Atual da Pena

Total da Pena	Horas Reduzidas
200	15:00

## Histórico de Atividades

Nome	Descrição	checkin	checkout	Total
Atividade 12	Testando habilidades 12	09/05/2021 13:00:00	09/05/2021 18:00:00	5h
Atividade 12	Testando habilidades 12	09/05/2021 13:00:00	09/05/2021 18:00:00	5h
Atividade 12	Testando habilidades 12	09/05/2021 13:00:00	09/05/2021 18:00:00	5h

*Este documento foi gerado pelo Gerenciador de Apenados, sob ordem do usuário portador do CPF . Lembrados que as informações são confidenciais e de responsabilidade do gerador do relatório.*