



Processo demandas PJe

Elaborado em 29/2/2016



Roteiro

- ▶ Sobre o Pje
- ▶ Principais ferramentas
- ▶ Visão geral do fluxo da fábrica do Pje
- ▶ Detalhes importantes do fluxo
- ▶ Relatórios Gerenciais



Sobre o projeto Pje

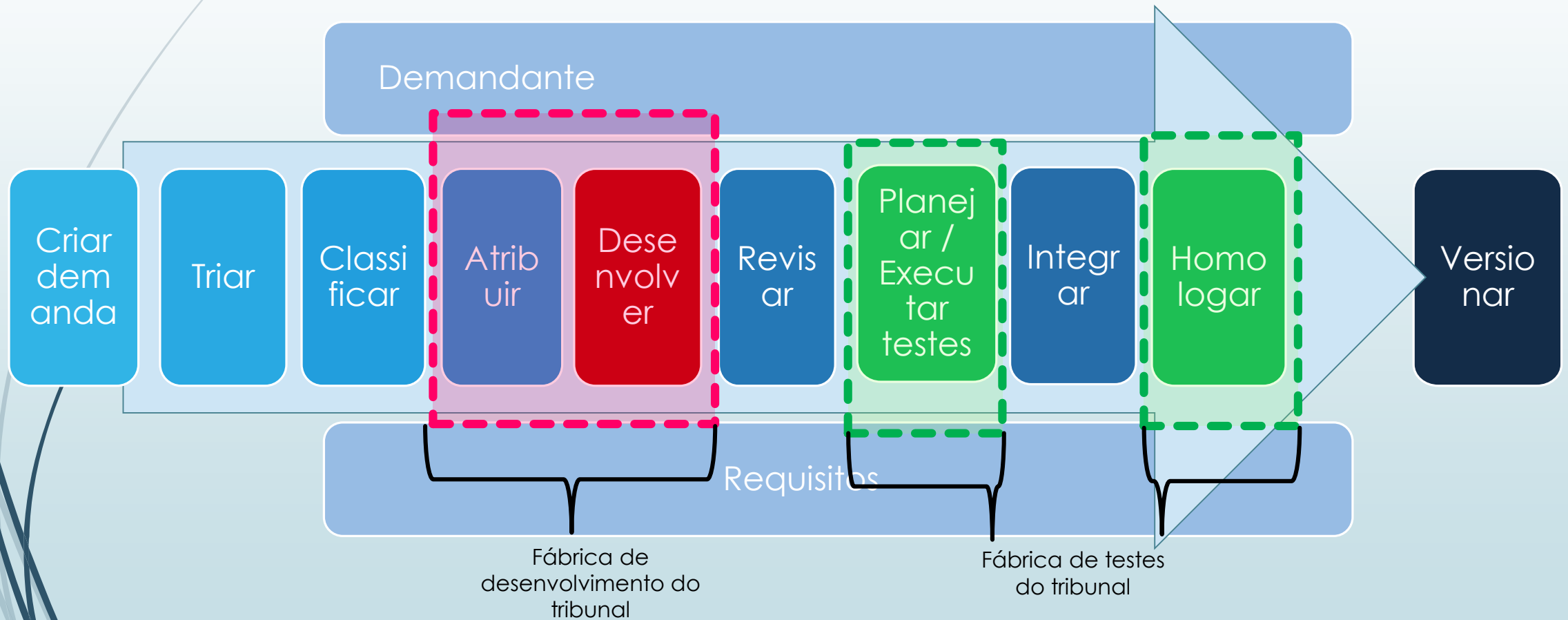
- ▶ Sistema de tramitação de processos judiciais
- ▶ Atender às necessidades dos diversos segmentos do Poder Judiciário brasileiro (Justiça Militar da União e dos Estados, Justiça do Trabalho e Justiça Comum, Federal e Estadual)
- ▶ Sistema elaborado pelo Conselho Nacional de Justiça (CNJ) a partir da experiência e com a colaboração de diversos tribunais brasileiros.
- ▶ Objetivo principal: manter um sistema de processo judicial eletrônico capaz de permitir a prática de atos processuais pelos magistrados, servidores e demais participantes da relação processual diretamente no sistema, assim como o acompanhamento desse processo judicial.
- ▶ O CNJ pretende convergir os esforços dos tribunais brasileiros para a adoção de uma solução única, gratuita para os próprios tribunais e atenta para requisitos importantes de segurança, de interoperabilidade e racionalizando de gastos com elaboração e aquisição de softwares.

Principais ferramentas

- Jira – gerenciador de desenvolvimento de demandas ([link](#))
- Git – sistema de controle de versões do código ([link](#))
- Jenkins – gerenciador de integração ([link](#))
- TFS – gerenciador de casos de teste ([aqui](#))



Fluxo básico da fábrica PJe



Etapas

► Criar demanda

- Nesta etapa a necessidade do tribunal é transformada em demanda no sistema gerenciador de demanda (Jira);
 - As demandas podem ser criadas tanto pelos tribunais, quanto por qualquer pessoa participante dos times envolvidos no desenvolvimento do PJe;
 - Antes da criação da demanda, deve-se fazer uma pesquisa preliminar no Jira em busca de demandas já relacionadas, demandas duplicadas ou até de informações de configuração para resolver a necessidade do tribunal;
 - Foi configurado no próprio Jira, no momento da criação da demanda, um *template* com informações mínimas para auxiliar tanto a equipe de desenvolvimento e de requisitos, quanto a equipe de revisão e a de homologação. A descrição da issue deve ser considerada a documentação da demanda.
-
- Em média são criadas 6 demandas por dia* no Jira:
 - 2 bugs / 3 defeitos / 1 melhoria



Etapas

► **Triar**

- A triagem é uma etapa realizada pela equipe do CNJ;
- Nesta etapa a equipe responsável, faz uma avaliação das informações mínimas de entendimento das demandas e verifica se a necessidade já não está sendo atendida por outra demanda;
- Nesta etapa é verificado se a demanda não é duplicada e aqui também são identificadas as demandas relacionadas;
- Durante esta etapa as demandas passam por uma pré-classificação e demandas urgentes (normalmente relacionadas a problemas de produção) já são encaminhadas diretamente à equipe responsável para um atendimento prioritário, as demais demandas seguem para a etapa de classificação e priorização.

Etapas

► Classificar

- A classificação das demandas depende da prioridade dos tribunais, da complexidade das demandas e da sua criticidade;
- A prioridade dos tribunais é informada no e-mail prioridade@cnj.jus.br e pode ser visualizada de forma consolidada no link: <https://goo.gl/tiHR4c>;
- As demandas classificadas ganham uma ordem de classificação que pode ser verificada no campo *Business Value*, a partir deste campo as demandas são ordenadas e ficam na fila para serem encaminhadas a uma das fábricas de desenvolvimento dos tribunais participantes do PJe Nacional.

Demandas que estão nesta etapa podem ser vistas no seguinte filtro (Jira):

[**Adm2.59 Classificar \(backlog\)](#)

Etapas

► Classificar*

► Frentes de trabalho

► Após a identificação da prioridade das demandas estas são classificadas em:

Urgente

Demandas do tipo “Bug em produção”, são classificadas assim, pois estão normalmente relacionadas a correção de dados pontuais que estão acontecendo em produção em documentos, processos ou cadastros específicos. Normalmente, para a solução da necessidade não há código Java.

Estabilização da versão

Demandas que venham a surgir por uma desestabilização da versão atualmente em homologação, são classificadas desta forma, pois devem ter prioridade no desenvolvimento e na sua verificação.

Sprint 1.7.2.x (homologação)

Quando o escopo da versão que deverá ser homologada é definido, as demandas (preferencialmente já desenvolvidas) são identificadas desta forma, para que seja possível a identificação do que deve ser homologado e em caso de recusa da solução, para que a sua correção seja priorizada pelas fábricas de desenvolvimento e de homologação.

Próxima Release (priorizado)

Outras demandas que envolvam codificação Java são classificadas como próxima release e são encaminhadas para uma das fábricas de desenvolvimento.



Etapas



► Atribuir

- Após a etapa de classificação a demanda aguarda a disponibilidade de uma das fábricas de desenvolvimento;
- Quando há uma fábrica disponível a demanda é direcionada para o líder da equipe de desenvolvimento, esta é a pessoa responsável pela distribuição interna e pelo acompanhamento do desenvolvimento das demandas;
- Não há um padrão para a atribuição das demandas aos desenvolvedores dentro de cada fábrica.
 - Em algumas o líder da equipe repassa para seus desenvolvedores demanda por demanda;
 - Em outras o líder já aloca várias demandas a cada desenvolvedor;
 - Em outras os desenvolvedores têm liberdade de se atribuir demandas.

Etapas

► Atribuir*

► Filtros

- Os filtros gerais das issues que estão nesta etapa são:

[**Adm01_BugsGeral \(atribuir\)](#)

[**Adm3.00 Atribuir \(homologacao\)](#)

[**Adm3.02 Atribuir \(priorizado\)](#)

- No entanto criamos filtros para cada uma das fábricas de desenvolvimento, com as demandas encaminhadas para cada tribunal (exemplo):

[*EQ05.22 TSE \(atribuir - priorizado\)](#)

[*EQ07.23 TJPR \(atribuir - priorizado\)](#)

[*EQ08.22 TJDFT \(atribuir - priorizado\)](#)

[*EQ14.22 TJMG \(atribuir - priorizado\)](#)

[*EQ16.22 TJCE \(atribuir - priorizado\)](#)



Etapas



► Desenvolver

- Ainda dentro da fábrica de desenvolvimento dos tribunais, há a etapa de desenvolvimento;
- Esta etapa é de responsabilidade do desenvolvedor, que deve ter seu nome atribuído à demanda no Jira;
- O desenvolvimento deve seguir os padrões do projeto, de acordo com as informações disponíveis na wiki do projeto ([aqui](#));
- Os desenvolvedores devem ficar atentos ao campo “Sprint do grupo”, pois deve-se priorizar o desenvolvimento na seguinte ordem (do mais prioritário, para o menos): Urgente, Sprint 1.7.2.x, Proxima Release, ficando atento às demandas “Rejeitadas pela homologação comercial”, “Solução recusada” ou “MR Reprovado”, pois estas são demandas já desenvolvidas que precisam de ajustes para serem entregues novamente;
- O desenvolvedor deve ficar atento à descrição da própria demanda, às regras de negócio do projeto e caso haja o planejamento dos cenários de teste incluídos na issue do jira;
- Ao final do processo de desenvolvimento, o desenvolvedor deverá:
 - abrir um novo MR no GIT para o *branch master* – com a identificação da issue do Jira;
 - transitar a issue (do Jira) para a situação “Homologação Técnica”;
 - Incluir evidência (preferencialmente um vídeo) comprovando o funcionamento da solução dada.



Etapas

► Revisar

- Esta etapa é centralizada pela equipe técnica do PJe do CNJ;
- A etapa tem o objetivo de aumentar a qualidade geral do código e de auxiliar na disseminação do conhecimento sobre as ferramentas do projeto entre os desenvolvedores;
- Nesta etapa serão feitos testes gerais da solução e será cobrada a evidência e a explicação técnica da solução dada pelo desenvolvedor;
- Mais detalhes deste processo e as boas práticas que são avaliadas encontram-se [aqui](#);
- Ao final desta etapa o revisor poderá solicitar ajustes ao desenvolvedor ou poderá aprovar a solução dada e integrá-la ao *branch master* do projeto.



Etapas

- ▶ **Planejar / Executar testes** (no *branch master*)
- ▶ Primeiramente, vamos à explicação do objetivo desta etapa:
 - ▶ O projeto PJe é um projeto muito grande e o seu processo de desenvolvimento está em constante adaptação em busca da melhor solução ao seu contexto execução;
 - ▶ Ao longo dos últimos anos, muitas demandas foram desenvolvidas e revisadas, mas a equipe de homologação não conseguiu acompanhar este desenvolvimento e consequentemente houve acúmulo de demandas pendentes de homologação;
 - ▶ Desta forma, acrescentamos a etapa de planejamento e execução de testes após a integração das demandas no *branch master*, para que posteriormente, na etapa de homologação da versão as demandas que ali chegarem já terem passado por um planejamento de testes e já tenham sido aprovadas em um teste prévio da solução;
 - ▶ A intenção é que futuramente consigamos colocar a etapa de “Planejamento de testes” antes da etapa de atribuição das demandas aos desenvolvedores e que a etapa de “executar testes” seja de responsabilidade do próprio desenvolvedor e do revisor de códigos, ficando a etapa de homologação mais focada em testes de integração e dos cenários alternativos das soluções.



Etapas

- ▶ **Planejar / Executar testes** (no *branch master*)

- ▶ Esta etapa deve ser executada pelas fábricas de teste dos tribunais e é separada em duas fases:

- ▶ **Planejamento de testes**

- ▶ Nesta etapa são criados e documentados os cenários de teste, para isso utiliza-se:

- ▶ a descrição da própria demanda no Jira;

- ▶ a documentação das regras de negócio hoje disponíveis na [wiki do projeto](#);

- ▶ o próprio Jira, em demandas relacionadas;

- ▶ e principalmente a experiência e documentação dos cenários de teste das equipes de teste do projeto.

- ▶ Os cenários devem conter tanto os caminhos “felizes” para o teste da demanda, quanto os caminhos alternativos;

- ▶ Os cenários criados para teste são incluídos em uma sub-tarefa da demanda que está sendo testada, para auxiliar às pessoas que venham a testá-la ou que venham a corrigir defeitos relatados na solução.



Etapas

► **Planejar / Executar testes** (no *branch master*)

► **Executar testes**

- Para a execução dos testes, a fábrica de testes do tribunal deverá montar um ambiente de testes com as últimas atualizações do *branch master* do projeto PJe;
- Os cenários devem ser executados um-a-um e o resultado de cada cenário deve ser reportado e filmado (ex. de aplicativo de filmagem: www.screencast-o-matic.com);
- Ao final deste processo deve-se incluir à demanda a lista de cenários testados, dos resultados obtidos (se aprovado ou se reprovado) e o filme dos testes executados;
- Se houver algum cenário reprovado a demanda deverá ser devolvida para o desenvolvedor fazer os ajustes necessários, pela transição (do Jira) “Reprovar solução”;
- Se todos os cenários forem aprovados a demanda deverá ser encaminhada para a próxima etapa (Integração), pela transição “Aprovar solução”.

Etapas

- ▶ **Planejar / Executar testes** (no *branch master*)

- ▶ Filtros relacionados:

- ▶ Filtro geral das issues que estão nesta etapa:

- [**Adm8.21 Escopo \(testar-master - priorizado\)](#)

- ▶ No entanto, podemos criar filtros para cada uma das fábricas de testes, com as demandas encaminhadas para cada tribunal (exemplo):

- [*EQ07.21 TJPR \(testar-master - priorizado\)](#)

Etapas

► Integrar

- Nesta etapa a solução que foi integrada (e aprovada) no *branch master* deverá ser transferida para o contexto da próxima versão que será homologada;
- Dependendo da quantidade de demandas que estão nesta etapa do fluxo, elas deverão ser integradas em grupo para compor o escopo de homologação da próxima versão (ou release) do sistema;
- As demandas normalmente ficarão aguardando a homologação da versão corrente e comporão o escopo da próxima versão a ser homologada, por exemplo:

Se hoje há a versão 1.7.2.3 em homologação, demandas classificadas como sendo do grupo "Sprint 1.7.2.x" serão integradas na versão 1.7.2.3 assim que caírem nesta etapa.

Mas se uma demanda, por exemplo: PJElI-20339, que estiver no grupo "Próxima Release" não será integrada na 1.7.2.3, mas quando esta versão for lançada, a demanda PJElI-20339 será uma forte candidata a compor a versão 1.7.2.4.



Etapas

► Homologar (Negocialmente)

- Esta etapa é executada pela fábrica de testes de um tribunal e durante esta etapa são executados os testes dos cenários planejados de cada demanda (ou para a realidade atual, são criados e executados os cenários para cada demanda);
- O resultado dos cenários é inserido na demanda e ela será devolvida ao desenvolvedor ou homologada e comporá o pacote da versão em avaliação;
- A diferença desta etapa para a “Executar testes” é que nesta etapa a demanda deverá ser testada no contexto de funcionamento da versão (que muitas vezes não contém todas as mudanças existentes no *branch master*) e no funcionamento de 1º e 2º graus, também nesta etapa são realizados testes de integração gerais do sistema;
- Problemas identificados de desestabilização do sistema que não tenham uma relação direta com as demandas da versão são abertos em um projeto em separado, por exemplo: “PJE-VERSAO_1.7.2” e marcados como “Estabilização da versão”, pois deverão ser tratados com prioridade;
- Outros problemas encontrados na versão que já ocorriam devem ser avaliados e dependendo da sua criticidade deverão integrar o escopo da própria versão.



Etapas



► **Homologar** (Negocialmente)

► Filtros relacionados:

► Filtros gerais das issues que estão nesta etapa:

[**Adm8.0 Escopo \(homologacao\)](#)

[**Adm8.01 Escopo \(pendentes - homologacao\)](#) – subconjunto do filtro anterior, mostrando apenas demandas ainda não homologadas

► Criamos alguns filtros por fábrica de testes do tribunal:

[*EQ07.01 TJPR \(testesTJPR - homologacao\)](#)



Etapas

► Versionar

- Esta etapa é executada pela equipe técnica do PJe do CNJ;
- Para gerar uma nova versão o pré-requisito é que todas as demandas que compõe o escopo da versão tenham sido homologadas;
- Nesta etapa é gerado um pacote .war e um arquivo de scripts com todos os scripts necessários para a evolução do banco de dados do tribunal, para que seja possível implantar a versão liberada;
- Adicionalmente é gerado um *release notes*, contendo uma descrição geral da versão e a lista de issues relacionadas, informação que pode ser obtida na wiki do projeto ([aqui](#));
- É encaminhado um e-mail aos líderes das equipes dos tribunais com a informação da disponibilização do pacote da versão no FTP do projeto;
- Por fim, é encaminhada também uma mensagem aos inscritos no canal do (público) do PJe Nacional no Telegram: PJE News (<http://telegram.me/pjenacional>).

Filtros

- Demandas por tribunal:

- Nestes filtros estão incluídas demandas criadas pelo tribunal, que estejam sob responsabilidade do tribunal e as que o tribunal tenha solicitado prioridade:

[*EQ08.00 TJDFT \(urgente/homologacao\)](#) – demandas com prioridade 0, possuem erros em produção ou são demandas que estão sendo homologadas para a próxima versão que será lançada;

[*EQ08.20 TJDFT \(priorizado\)](#) – demandas priorizadas pela comunidade do PJe e que têm relação (criada, priorizada ou atribuída) com o tribunal, estas demandas estão em desenvolvimento ou ainda não foram homologadas.

[*EQ08.21 TJDFT \(desenvolver - priorizado\)](#) – sub-grupo da lista anterior, contempla as demandas já priorizadas restringindo àquelas que ainda faltam ser desenvolvidas.

Solicitamos às equipes de desenvolvimento dos tribunais que utilizem este filtro para acompanhar o que devem desenvolver e qual a situação do que está sendo desenvolvido.

[*EQ08.22 TJDFT \(atribuir - priorizado\)](#) – sub-grupo das pendentes de desenvolvimento, contem as demandas ainda não atribuídas aos desenvolvedores, o responsável do tribunal por repassar as demandas à equipe de desenvolvimento deve sempre acompanhar este filtro.

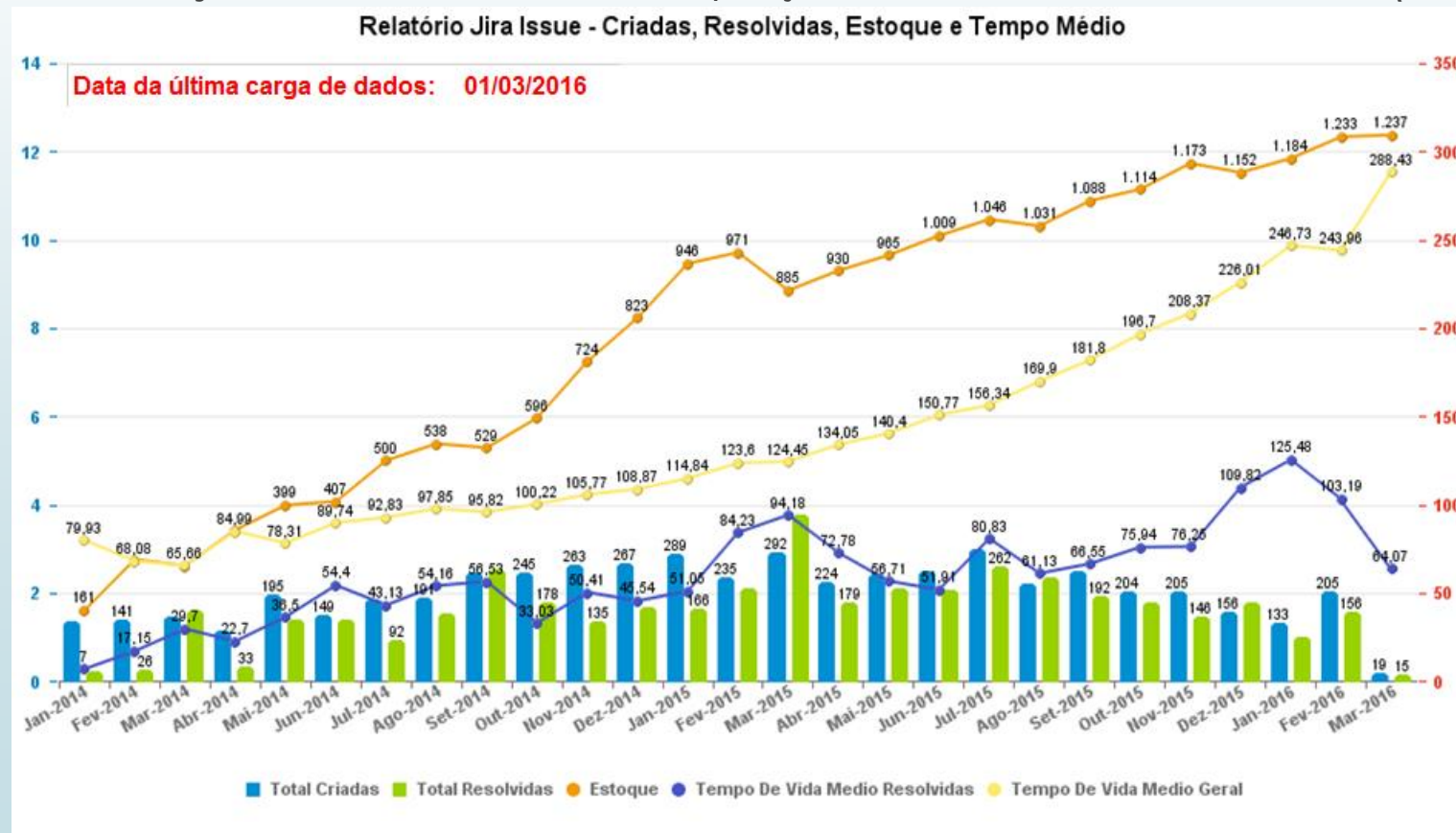
[*EQ08.9 TJDFT \(backlog\)](#) – demandas neste filtro são aquelas ainda não priorizadas pelo tribunal, ou caso estejam priorizadas ainda não foram encaminhadas à fábrica de desenvolvimento pois esta ainda está com muitas demandas a desenvolver.

Aqui no CNJ buscamos sempre acompanhar a lista de demandas priorizadas pendentes para desenvolvimento e tendemos a repassar mais demandas de defeito às fábricas de desenvolvimento. De qualquer forma, buscamos atender às prioridades dadas pelos tribunais enviadas ao e-mail prioridadepje@cnj.jus.br, esta lista de demandas priorizadas por cada tribunal pode ser vista aqui: <https://goo.gl/tiHR4c>.

*Para os demais tribunais, basta pesquisar nos filtros criados por “Zeniel”, trocando a sigla do tribunal, ex.: TJMG, TJPR, TJCE, TSE, TRF1

Alguns relatórios do projeto

- Evolução das demandas no projeto PJe nos últimos 2 anos (ver anexo):



Alguns relatórios do projeto

- Demandas pendentes de solução por assunto e por tipo:

#	Epícteme macro	Total	Total Bugs	Total Defeitos	Total Melhorias	Total Outros
1	[Fluxo servidores+magistrados]	175	-	61	99	15
2	[Cadastro básico]	145	2	58	69	16
3	[Documentos e expedientes]	137	-	33	83	21
4	[Processo]	108	3	38	59	8
5	[Infra]	89	-	30	46	13
6	[Permissões e Visibilidade]	82	-	28	50	4
7	[Layout+Usabilidade]	58	-	28	28	2
8	[Push+Acesso externo]	53	2	29	22	-
9	[Outros perfis]	37	-	11	22	4
10	[Audiências e sessões]	34	-	14	18	2
11	[Representante processual]	24	-	9	14	1
12	[Intercomunicação]	22	-	8	12	2
13	[Grupo de usuários]	19	-	9	10	-
14	[Dívida ativa+execução fiscal]	19	-	8	9	2
15	-	18	-	3	5	10
16	[Movimentação]	12	-	6	4	2
17	[Mais dados processos]	10	-	5	5	-
18	[Criminal]	9	-	2	7	-
19	[Layout+Usabilidade]	8	-	-	2	6
20	[Fluxo servidores+magistrados]	7	-	-	1	6
21	[Representante processual]	2	-	-	1	1
22	[Outros Sistemas]	1	-	-	1	-
23	[Outros perfis]	1	-	-	-	1
24	[Cadastro básico]	1	-	-	-	1
25	-	1	-	-	-	1
26	[Grupo de usuários]	1	-	-	-	1
27	[Outros]	1	-	1	-	-

*dados de 1/3/16