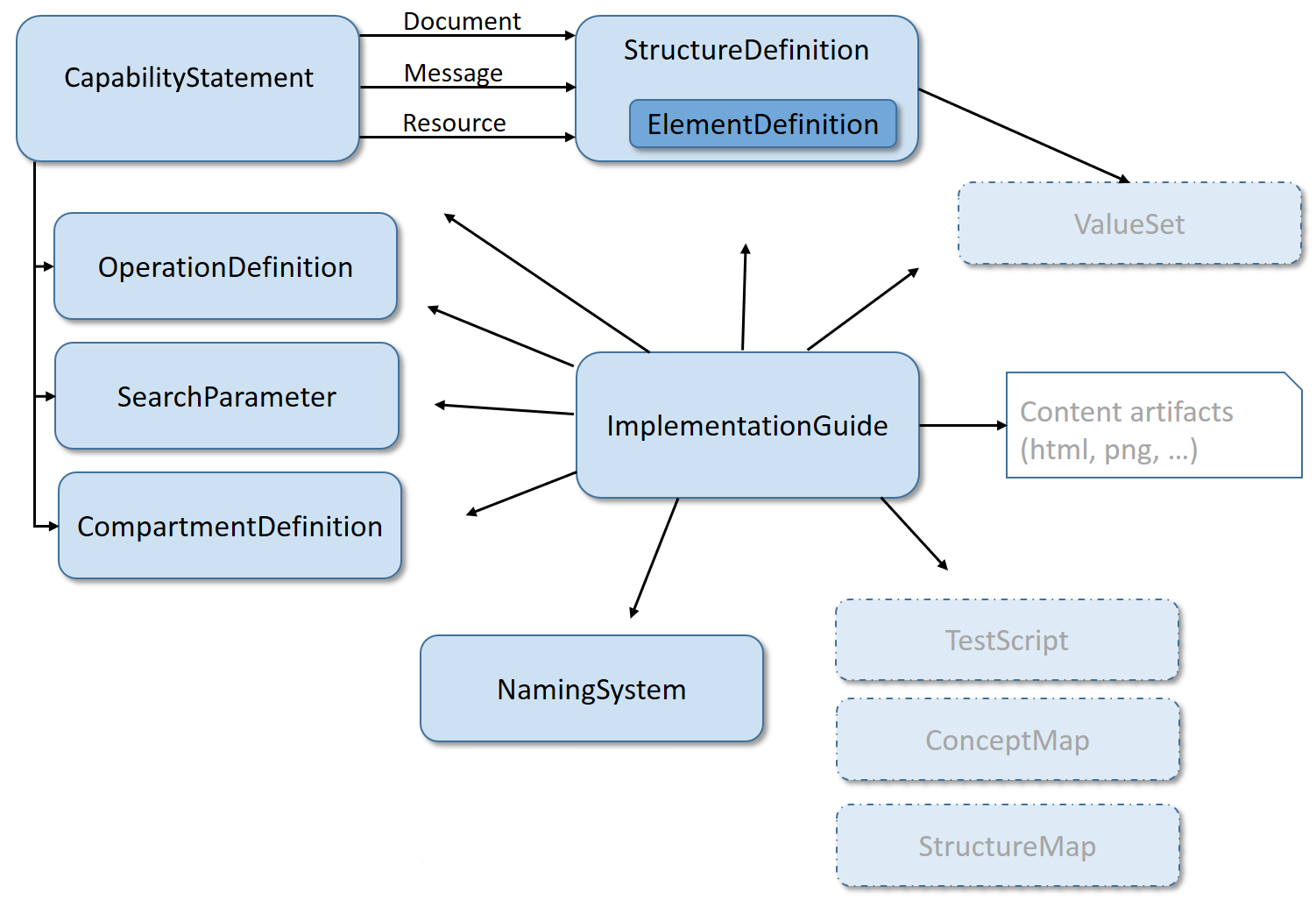
# **1 Guia de implementação**



**Imagem -** O módulo de Conformance do FHIR.

O guia de implementação é um conjunto de regras de como um determinado problema de interoperabilidade ou padrões é resolvido - geralmente por meio do uso de recursos FHIR. Este recurso é usado para reunir todas as partes de um guia de implementação em um todo lógico e publicar uma definição computável de todas as partes em um pacote de interoperabilidade.

O recurso ImplementationGuide é um recurso único que define o conteúdo lógico do IG, contendo páginas de contextualização importantes para a publicação do pacote lógico que o IG representa. Esse pacote pode ser usado pelos servidores FHIR para confrontar e validar o conteúdo em relação ao guia de implementação.

O módulo de conformidade do FHIR traduz esse pacote lógico em regras de validação para perpetuar as regras de conformidade especificadas pelo guia através de perfis que podem ser construídos em cima do padrão. É importante ressaltar que uma implementação FHIR pode estar em conformidade com vários guias de implementação ao mesmo tempo. Isso requer que os guias de implementação sejam compatíveis.

Os guias de implementação contêm StructureDefinitions que definem perfis que customizam regras de validação para recursos e extensões que adicionam novos campos aos recursos; NamingSystems que determinam identificadores de pessoas físicas, jurídicas e dispositivos; CodeSystems que criam sistemas de codificação para abarcar terminologias, ontologias e sistemas de classificação; ValueSets que são coleções dessas codificações oriundas de sistemas de codificação para reunir códigos em contextos específicos; CapabilityStatements que são especificações de operações e parâmetros de um servidor FHIR.

# **2 Multiversão**

A maioria dos guias de implementação visa uma única versão. Em outros casos, no entanto, a implementação de um guia de implementação não se limita a uma única versão. Normalmente, o requisito para oferecer suporte a várias versões surge à medida que a implementação amadurece e diferentes comunidades de implementação ficam presas em diferentes versões por regulamentação ou dinâmica de mercado. Os aplicativos podem travar em diferentes versões da especificação.

Por esse motivo, os guias de implementação podem descrever como usar várias versões diferentes do FHIR para a mesma finalidade. As diferentes versões podem ter diferentes perfis, extensões e exemplos, enquanto compartilham definições de conjuntos de valores comuns, por exemplo. Por alguns motivos, perfis e exemplos podem ser comuns em todas as versões. E um guia de implementação geralmente terá muitas narrativas comuns descrevendo o problema, abordagens de segurança e outras informações de implantação, independentemente das versões específicas do FHIR.

# **3 Especificando multiversão em um guia de implementação**

Um guia de implementação especifica quais versões do FHIR ele descreve na propriedade ImplementationGuide.fhirVersion:

{

"resourceType": "ImplementationGuide",

"fhirVersion": ["3.0.1","4.0.1"]

}

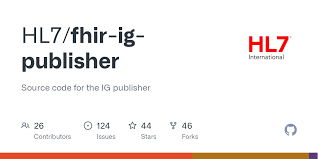
**Snippet** - Especificando um guia de implementação em multiversão.

No exemplo acima, o recurso ImplementationGuide se declara aderente a duas versões do FHIR: 3.0.1 e 4.0.1.

# **4 Perfis Padrões**

Um guia de implementação pode definir perfis padrões usando o atributo **ImplementationGuide.global**. Esses são perfis que se aplicam a qualquer recurso que não tenha um perfil explícito atribuído pelo guia de implementação. Ao definir perfis padrões, um guia de implementação pode evitar a definição exaustiva de perfis em cada tipo de recurso apenas para criar o perfil de cada referência a um determinado tipo de recurso. No entanto, é corriqueira a necessidade de diferentes perfis para um mesmo recurso, portanto essa implementação não é muito utilizada.

# **5 Estrutura de um guia de implementação**



**Imagem -** FHIR Implementation Guide Publisher.

O guia de implementação tem um boilerplate que deve ser seguido para ser compilado. O FHIR IG publisher pega o conteúdo do guia de implementação onde é possível definir os recursos que sustentam os recursos do Guia de Implementação (Guia de Implementação, Recursos de Conformidade, Declarações de Capabilidade, e Exemplos), usando FHIR Shorthand (FSH), pacotes (TGZ) ou planilhas (XLS) a fim de desenvolver a estrutura e conteúdo narrativo que transforma os recursos em um guia de implementação. O resultado do processo de publicação é um conjunto de arquivos HTML que representam o guia de implementação. Esses arquivos podem ser postados em um servidor web.

# **6 Publicando um guia de implementação**

Normalmente, a publicação do guia de implementação acontece a partir de um repositório GitHub público, onde os editores podem executar sua própria compilação local e compartilhar o conjunto de arquivos com quem quiserem. Uma compilação contínua é publicada em URL canônica do pacote do guia de implementação.

# **7 Versionando um guia de implementação**

A editora canônica de guias de implementação é mantida em <http://build.fhir.org> , onde é publicada de forma contínua. Lá são mantidas as seguintes versões:

* 1.0.2: DSTU2;
* 1.4.0: versão preliminar de maio de 2016 (suportada para algumas partes interessadas que a utilizam e tornam isso possível);
* 3.0.1: DSTU3;
* 4.0.1: R4;
* 4.3.1: R4b;

O IG pode funcionar com outras versões (por exemplo, bifurcações de versões candidatas R4), mas isso não é garantido e não é recomendado. Se você estiver usando uma versão diferente da compilação atual, precisará especificar a versão no arquivo de controle.

# **8 Versionando perfis**

A versão de perfil implica que perfis não precisam ter suas URLs canônicas alteradas quando são feitas alterações substanciais. Isso é uma questão de ordem que impacta significativamente em eventual retrabalho dos integradores. Em vez disso, todas as referências de perfil em um IG simplesmente usam um URL canônico consistente e são concomitantes a versão do guia de implementação.

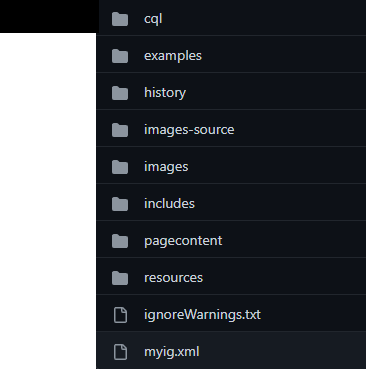
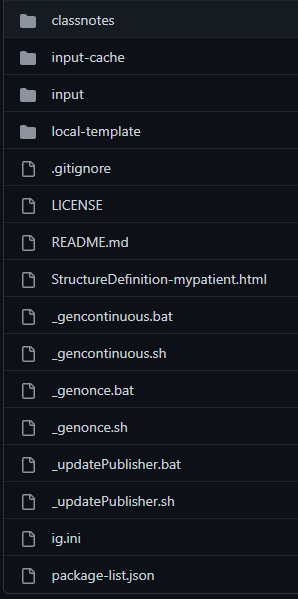
O versionamento de perfis só é permitido para Extensões. Neste caso, quando as extensões declaram o perfil em uma versão específica, elas combinam o URL canônico com a versão da extensão da seguinte forma:

[URL canônica] **|** [versão da Extensão]

**http://example.org/fhir/StructureDefinition/someExtensionId|1.2**

Isso informa ao validador qual versão do perfil deve ser usada ao validar a extensão.

# **9 Estrutura de arquivos de um guia de implementação**



**Imagem -** Estrutura de caminhos para arquivos em um IG.

O IG define os seguintes caminhos que podem ser configurados:

* **input**
  + O diretório onde todos os recursos FHIR de entrada são encontrados (geralmente recursos de conformidade e exemplos). Vários diretórios podem ser usados para permitir que os arquivos de origem sejam agrupados.
* **input-cache**
  + Pasta de cache de artefatos sintáticos como Perfis, Extensões. Traz celeridade ao processo compilação e checagem da interoperabilidade sintática.
* **temp**
  + Um diretório temporário onde são mantidos arquivos temporários utilizados pelo IG publisher para construir o guia.
* **output**
  + A pasta de saída onde é gerado o conteúdo da página estática que é o guia de implementação.
* **txCache**
  + Cache do serviço de terminologia. Pasta de cache de artefatos semânticos como CodeSystems e ValueSets. Traz celeridade ao processo compilação e checagem da interoperabilidade sintática.
* **history**
  + A página que lista o histórico do guia de implementação.
* **images**
  + Contém imagens que são utilizadas dentro do guia.
* **includes**
  + Normalmente define menus do guia.
* **pagecontent**
  + Define páginas do guia.
* **resources**
  + Define perfis, extensões, terminologias e conjuntos de valores do guia.

# **10 Lista de Dependências**

Um guia pode ter seus perfis baseados em recursos de outros guias. Para isso, é necessário definir a dependência desse guia com o outro guia. Para fazer essa definição de dependência, dentro do recurso de IG, é necessário referenciar o outro guia através da sua URL canônica e versão.

{

"resourceType": "ImplementationGuide",

"dependsOn": [

{

"nome": "uscore",

"location": "http://hl7.org/fhir/us/core",

"versão": "1.1.0"

}

]

}

**Imagem -** Dependência com outros guias.

# **11 URL Canônica e redirecionamento**

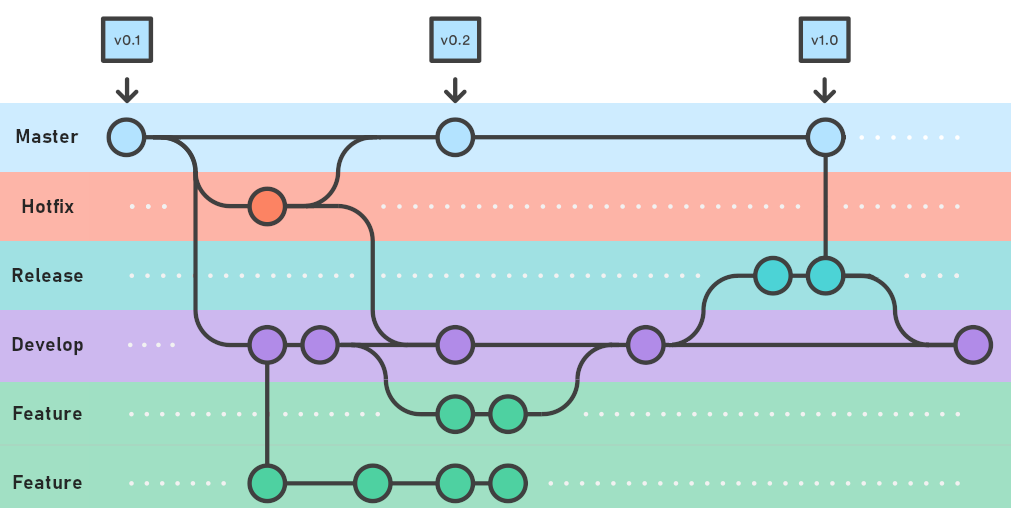
O Guia de Implementação deve nomear uma URL canônica. Essa URL canônica é usada para definir todos os recursos. A URL canônica deve sempre apontar para a versão atual do IG que você está publicado, portanto, se alguém inserir o URL canônico em seu navegador, obterá a página inicial do IG (por exemplo, <http://example.com/fhir/meuPerfilDePaciente>).

Se possível, também deve ser um servidor FHIR que hospeda recursos - portanto, um GET de [canônico]/[tipo]/[id] deve retornar um recurso FHIR. Por exemplo, http://example.com/fhir/Patient/mysamplepatient forneceria a você um exemplo de recurso do paciente usado em toda a especificação.

Outras versões anteriores de um guia podem ser acessadas através da URL canônica acompanhada da versão. É por esse motivo que uma versão anterior de um guia é colocada como uma subpasta da versão atual de um guia de implementação. Por exemplo, <http://example.com/fhir/Patient/2.0.3/mysamplepatient>.

# **12 Repositório do guia de implementação**

# **12.1 Processo de manutenção de um repositório**



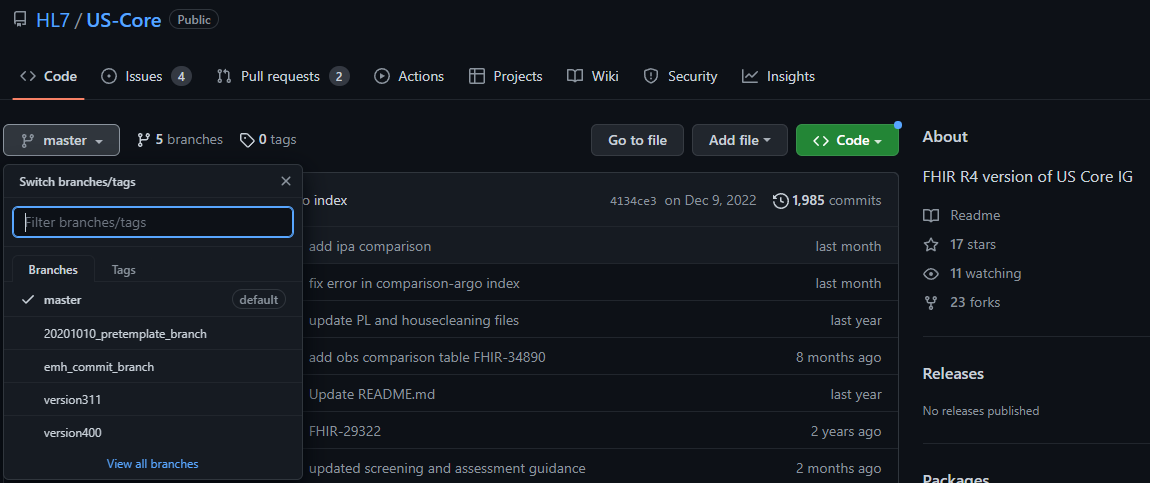
**Imagem -** Fluxo de repositório para um guia de implementação.

O guia de implementação é, portanto, mantido em um repositório de acordo com a estrutura de pastas sugerida para o guia de implementação que é compilado através do FHIR Implementation Guide Publisher. O ciclo de manutenção do repositório de um guia de implementação é melhor mantido seguindo o processo de *Git Flow.* O Gitflow é um processo de utilização do Git que envolve o uso de ramificações de linhas primárias e secundárias de desenvolvimento para manter um repositório.

O Git Flow trabalha com duas branches principais, a Develop e a Master, que duram para sempre; e três branches de suporte, Feature, Release e Hotfix, que são temporários e duram até realizar o merge com as branches principais. Então, ao invés de uma única branch Master, esse fluxo de trabalho utiliza duas branches principais para registrar o histórico do projeto. A branch Master armazena o histórico do lançamento oficial, e a branch Develop serve como uma ramificação de integração para recursos.

É ideal que todos os commits na branch Master sejam marcados com um número de versão e sejam estáveis e publicáveis em produção.

# **12.2 Contextualizando o git flow para guias de implementação**



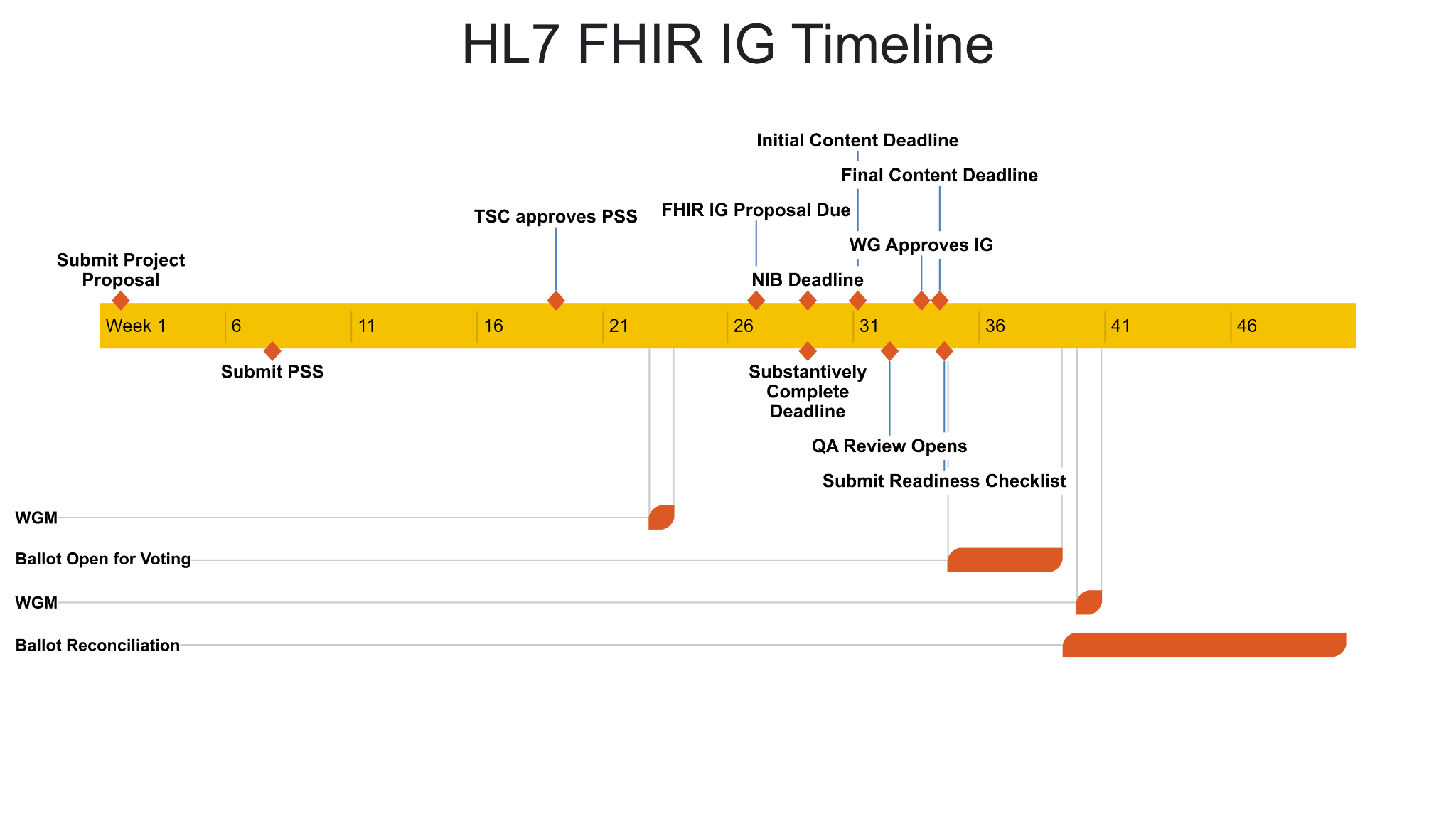
**Imagem -** Estrutura de branches do US-Core.

Para a manutenção de guias de implementação, o conceito utilizado é basicamente o mesmo. Trata-se de uma especialização do processo de *Git Flow* para guias de implementação em FHIR. O repositório é mantido em duas linhas principais de produção e desenvolvimento e as linhas secundárias são utilizadas para manter *Features, Releases e Hotfixes.*

Cada Feature pode ser um Perfil, Exemplo, Extensão, Identificador, Terminologia ou Conjunto de Valor. A história de cada Feature pode ser trazida para a linha principal de desenvolvimento e agrupada para a entrega de releases do projeto quando forem atingidos marcos do projeto ou modelo de informação. Alternativamente, as correções de erros encontrados na linha de produção podem acontecer na linha secundária de *hotfix*. Lembrando que devem produzir uma nova versão quando eventualmente retornadas à linha principal.

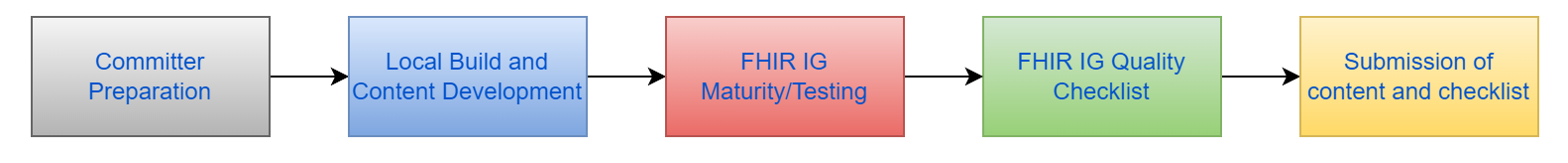
Observe que na imagem acima estão representadas as ramificações do US-CORE que é o guia de implementação nacional dos Estados Unidos. Cada release tem sua própria ramificação enquanto atualizada e pode ser descontinuada a medida que um projeto de guia de implementação muda de direção ou transiciona para uma nova fase.

# **13 Gestão de configuração para um guia de implementação**



**Imagem -** Timeline de um guia de implementação.

As tarefas de um guia de implementação podem tanto ser provenientes de um roadmap de um projeto de saúde digital, de um backlog de bugs ou de solicitações de mudança oriundas de um consenso construído em um foro mantenedor desta iniciativa.



**Imagem -** Ciclo de vida de cada mudança em um guia de implementação.

Cada mudança em um guia de implementação passa por um ciclo de preparação, build, teste, checklist de qualidade e submissão de conteúdo. O ciclo de vida para preparação de uma mudança é mais disciplinado para guias de implementação canônicos, mas esses processos não são obrigatórios para guias nacionais. As atividades de preparação de ambiente estão associadas a instalação de ferramentas e configuração de ambiente para submissão da mudança, o build local e desenvolvimento de conteúdo contempla atividades corriqueiras do grupo de trabalho para a entrega de novos artefatos. Os processo de teste e maturidade do perfil estão associados ao Modelo de Maturidade do FHIR (FMM) e estão associados à condução de conectatonas, teste e avaliação dos artefatos propostos.

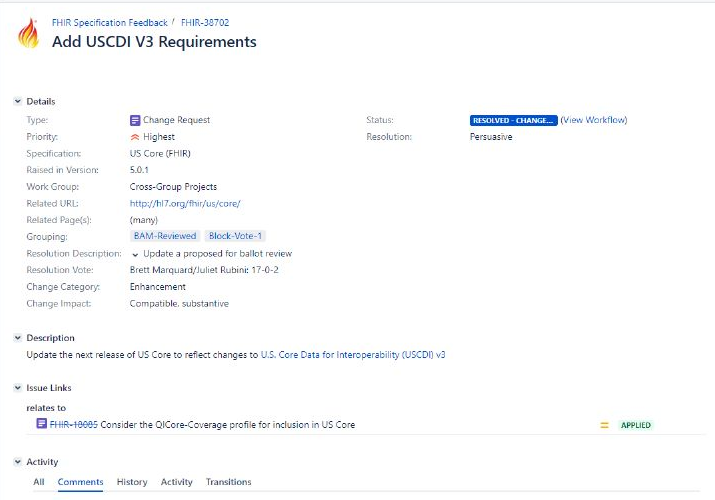
Uma vez entregues, passam por um checklist de qualidade



**Imagem -** Checklist de qualidade de uma requisição de mudança.

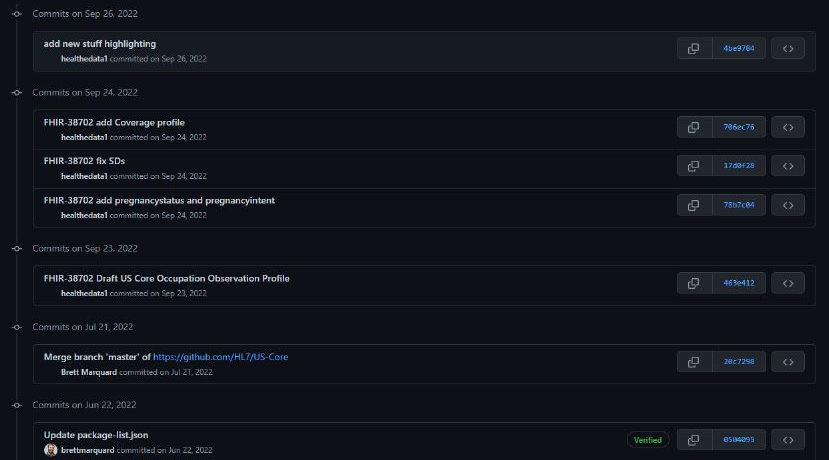
As qualidades das mudanças submetidas passam por um checklist. O autor da submissão precisa de um facilitador com acesso às linhas principais do repositório e que faça a auditoria da mudança proposta. O guia desta mudança deve estar compilado e o relatório de erros e warnings precisa estar vazio. Quando erros e warnings forem encontrados, precisam ser justificados junto ao facilitador e reportados ao grupo de trabalho que mantém o guia.

# **14 Detalhando a requisição de mudança para um guia de implementação**



**Imagem -**  Tarefa criada para deliberar sobre uma solicitação de mudança em um projeto de saúde digital.

A requisição de mudança acontece para um guia de implementação quando é oriunda de uma solicitação de mudança para o projeto. Por diversas razões já mencionadas na seção anterior, ela é cadastrada em um sistema de gestão de configuração. No caso do HL7 Internacional, é mantida no Jira, mais especificamente dentro do projeto criado para o grupo de trabalho que mantém esse projeto. A solicitação de mudança é julgada pelos facilitadores do projeto e, se aprovada, segue como tarefa de implementação para o grupo de trabalho mantenedor deste guia. Repare no exemplo acima que o ticket FHIR-38702 foi criado e encaminhado para desenvolvimento do projeto do US-CORE.



**Imagem -** Histórico de mudanças em um guia de implementação.

A solicitação de mudança vira, portanto, uma tarefa de desenvolvimento que é realizada por um membro do time e que passa por todo o processo de validação do facilitador antes de ser finalmente submetida para linha de desenvolvimento e pode ser agrupado para um release do projeto. Veja a tarefa FHIR-38702 no histórico de mudanças do repositório do US-CORE.

# **15 Conclusão**

Concluímos portanto o sobrevôo sobre um típico processo de manuteção de um guia de implementação. Desde conceitos básicos sobre um guia, passando pelo versionamento de perfis, extensões e artefatos semânticos e, finalmente, apresentando o processo de governança e gestão de um guia de implementação entrando em detalhes sobre como acontece uma solicitação de mudanças para um guia.

**16. Referências**

* 1. Guia de Fluxo de Processo para Guias de Implementação. HL7 Internacional. <https://confluence.hl7.org/display/FHIR/FHIR+Implementation+Guide+Process+Flow> Último acesso em 20/01/2023.