# O que é o inferno das dependências? O que houve no caso do Jarbas (https://github.com/okfnbrasil/Jarbas), fale um pouco deste projeto

# Quanto mais o sistema cresce, e mais pacotes são adicionados a ele, maior será a possibilidade de, um dia, você encontrar-se neste poço de desespero, para controlar as versões dos pacotes.

# No caso do Jarbas, a versão 0.4.0 foi testada e aprovada, porem começou a apresentar erro ao tentar carregar os dados de reembolsos e foi difícil identificar o que de fato tinha mudado na versão, com isto foi reinstalada a versão anterior 0.3.0 e rodar o comando para confirmar se o reprint havia mesmo alterado o código.

# Se o reprint não visionasse as alterações seriam impossíveis identificar aonde ocorreu a falha na versão 0.4.0. Esse caso torna claro de que ter ocorrido o incremento da versão para acompanhar a mudança no código tornou possível e relativamente fácil de voltar a usar localmente a versão 0.3.0 e não afetar o funcionamento do sistema.

# Dependency Hell

Dependency hell é um termo usando quando não se tem o controle das versões de alguns componentes e/ou pacotes de software e até mesmo documentações; dificultando assim a correção de erros e identificações de melhorias que o software poderá ter.

# PyUP

PyUP são atualizações automatizadas de dependência do Python, desenvolvida para resolver estruturas e problemas com ambientes , através de analise de código e atualizações. Juntamente pode-se usar o Safety que verifica se existem vunerabilidade de seguranças nas dependências.

O PyUP é uma ferramenta muito importante para o gerenciamento de configuração de software, pois é usado em conjunto com sistema de controle de versão, assim sendo possível controlar atualizações de dependências e vulnerabilidade em códigos ou configurações, algumas empresas como Google e Mozilla estão usando está ferramenta no seu dia-a-dia para automatização de segurança e dependências.

# Versionamento Semântico

Com o versionamento semântico pode-se transmitir o significado do componentes.

Dando um número de versão ***MAJOR.MINOR.PATCH*** para os componentes:

1. **Versão Maior(MAJOR)**: Quando as mudanças incompatíveis na API,
2. **Versão Menor(MINOR)**: Novas funcionalidades mantendo compatibilidade;
3. **Versão de Correção(PATCH)**: Correções de falhas, porém mantém compatibilidade.