O caso Jarbas demonstra exatamente a necessidade e a importância de um controle de versionamento. Não só do controle da versão de seu aplicativo como das versões das bibliotecas que utilizava. Através do PyPI ele controlava o versionamento de todas as bibliotecas que utilizava em seu App. Ao perceber, através de testes foi possível diagnosticar que o problema ocorria em uma parte do código que não foi alterada. Já desconfiado que o problema poderia ser em uma das bibliotecas, efetuou o mesmo teste com uma versão anterior de seu app (mas com as mesmas bibliotecas atualizadas pelo PyPI). Ao notar que o erro persistia, ficou mais fácil diagnosticar que o problema certamente estava relacionado com alguma das bibliotecas atualizadas. Daí pra frente foi apenas uma questão de efetuar novos testes (agora com as bibliotecas nas versões anteriores á atualização) para constatar que a falha referia-se a algum problema de incompatibilidade entre as bibliotecas atualizadas e a nova versão do seu aplicativo.

Graças ao controle de versionamento, foi possível que Jarbas utilizasse facilmente tanto as versões anteriores de seu aplicativo, quanto das bibliotecas utilizadas por ele.

É importante também ressaltar a estrutura do versionamento, que também serve para indicar a compatibilidade ou incompatibilidade entre as versões do mesmo aplicativo, conforme o modelo de versionamento semântico.

Neste modelo, o número da função é formado por 3 conjuntos numéricos, por exemplo “X.Y.Z”, onde a grandeza da alteração é determinada da direita para a esquerda, sendo as alterações no conjunto Z (denominado “Patch ou Correções”) as menores, referentes apenas á correções. Já uma mudança no conjunto Y representa novas features\funcionalidades, e por fim(“Menor”) e as mudanças no conjunto chamado “Maior”, neste caso representado por X representam mudanças que implicam na compatibilidade da API.

Podemos ainda alinhar o uso deste tipo de versionamento ao Headline do projeto, de forma a tornar as mudanças de versões, subversões e releases menos destoantes em suas numerações, a exemplo de muitos apps que em uma versão chegam ao release 50, e em outra versão, apenas 9.