A abordagem usada tem base no Quorum Consensus, com algumas alterações com objetivo de melhorar a eficiência do projeto. Nesse sentido, é assumido que existem três réplicas Station, e que não pode haver mais que uma falha em simultâneo. Além disso, o servidor Binas pode falhar ocasionalmente, o que pode causar a perda ou corrupção na informação dos utilizadores.

Uma das alterações realizadas foi na estrutura da tag. Foi decidido que a tag seria um número sequencial, que iria ser incrementado apenas quando é realizado a operação *setBalance,* ou seja, quando é realizada a operação de escrita. Além disso, a tag mais recente é também guardada no *Front End* do utilizador. Isso é possível porque existe apenas um cliente, pelo que não é necessário guardar o clientId. Assim, a leitura realizada antes de se fazer uma escrita, com o objetivo de procurar qual a tag mais recente, passa a ser desnecessária.

Outra otimização realizada consiste no armazenamento, por parte de cada utilizador, da tag respetiva à ultima escrita (*setBalance)* de cada utilizador. Em leituras subsequentes, o cliente vai comparar este valor (a tag, que se trata do número de sequência) com o seu próprio número de sequência, que é guardado do lado do cliente, como explicado acima. Assim, se o número de sequência for igual, não há necessidade de ler o valor nas stations, já que existe a garantia que o saldo guardado pelo utilizador é o mais atual.

Numa situação de falha do servidor Binas, em que os registos de utilizadores são completamente perdidos, a associação de email e saldo de cada utilizador persiste nas réplicas station. Assim sendo, na realização de operações returnBina e getBina posteriores, admitindo uma verificação da existência do utilizador (via getBalance), este será re-instanciado no servidor Binas com os valores por omissão.

