# Feedback Sprint ending Mar 13

## Client

* I see some implementation code being put in header files which are not template classes. This might come back to cause problems later in the project. There are a few things the compiler cannot do if the code is implemented that way.
* It looks like events are delivered to all grid squares rather than finding the grid to receive the event once and just delivering it to that gridsquare.
* In ClientAPI.h all of the resource collections are stored as vectors and then, when you need to find something or delete something, a linear search is performed. Each collections is two vectors, one for the keys and one for the values. Can ANYONE tell me why we are not using maps for this???????? Do we need to slow the game down? Do we avoid proper object-oriented techniques? Just give me a reason.

## Database

* There is a lot of untested code here
* There is no way to load the entire game
* Design of saving should use a sequence chart to show how it is to be done
* There is no information on the players

## Networking

* The design of client and server liaison looks very simplistic to me. I have suggested a more sophisticated architecture that will allow you to use the same listener on the client and server side. This will save a lot of time in the long run and be more robust. Your currect design does not even have a thread pool in it and it is not obvious how the multi-threaded aspect is managed. The design looks to be a collection of methods rather than a true design involving the several classes it should use.
* Netbroadcaster design should have some sort of relationship to the NetClient class in the class diagram.

## Server

* Designs like the teams one tell me nothing that I would not see reading the code. In fact, it is the code. It should have associations and cardinality rather than vectors.
* The lobby remains a mystery to me. I can find no design and it has few connections to the rest of the server. It does not have a list of games nor access a list from the local DB. I wonder how this is supposed to work?