1. Are there any inconsistencies in the design? That is, does the document contradict itself?  
   None that we could find.

2. Are there omissions in the design? That is, are there elements that are mentioned but never discussed, or obvious pieces that are missing?  
   a. The use of SSL/encryption was stressed in Part 3 of the Requirements document, but there is no mention of it in the design document.  
   b. There are no UML diagrams, which makes it quite hard to understand the relationships between the various classes. Some classes are simply mentioned by their public interfaces are not specified.  
   c. It is said that movement will be animated but it is not specified how movement animation will be done and how object collision will be handled, given the various synchronization issues related to this.  
   d. The specification of the game UI is not detailed enough. It is only specified that the GameEngine class will do the rendering, but what will be the classes that it will be using? Will there be classes that will represent the various game objects: players, ships, ports, land, etc?

3. Are any parts of the design unclear?  
   a. (Part 4) In the description of the GameData class, they describe ClientData. Is this a typo? Mislabeled?  
   b. (Part 4) ClientData contains x velocity, y velocity. How is this determined?  
   c. How are ship type (will different ships look different?), amount of money stored?  
   d. Does each client have their own database that updates the Server Database (a one to many relationship between the Server Database and Client Database)? If so, why does each client have their own copy of all player information and not just their own Row?  
   e. How are the missions associated with each port stored?

4. Are there technical errors in the design? Is there any statement of fact that you know is false?  
   None that we could find.

5. Has thought been given to testing? How would you test this design? What, if anything, could be done to make the design easier to test?
a. Load testing? They are hoping for hundreds or even thousands of players. So, load testing becomes very critical especially since this is a graphics heavy project and poor performance in rendering might cause problems.

6. Does the design make realistic assumptions about the environment? That is, will the team have trouble getting access to important external components are the systems the project needs to interact with suited to the purpose?
   a. Found a suitable server as described in Required System Components (Part 6 of Requirements and Specification Document)?

7. Does the plan seem realistic? Are tasks at a reasonable level of granularity and is it clear what each task means? Do the time estimates seem appropriate? Do any parts of the plan seem risky in the sense that they are likely to become a bottleneck to further progress?
   a. The timeline for tasks in 6, Plan, is very aggressive. Possibly unrealistic?
   b. Some parts of the design are very vaguely specified and so it is very hard to tell how much work will be required to implement those parts of the program. For example, it not clear how much work will go into implementing the GUI because the design of it is not really described.

8. Any other comments?
   a. What determines velocity? weather?
   b. It is mentioned that upon login, all of the data needed to initialize the player's world (the location of other players, ships, etc) will be sent to the client. It seems that the current plan is that this data will be sent through the main command queue, which can create a serious latency for other players already in the game.