**Design Side**

* Information text is tiny and will be unreadable on a phone.
* A simplified rating system would be a welcome feature, considering how small the space is that you have to work with.
* The characters may need emoji speech bubble or something to convey emotion, considering how small they would be on-screen.
* 5-12 is a massive age range. The differences in level of comprehension and attention are significant and you will not be able to cater to that broad of an audience. Narrow your focus so you can provide better quality content for the range you settle on.
* You could take the pop-up style a step further and have your information boxes be pull-out tabs and flaps etc.
* Be wary of straying from the scientific data when balancing your game. Sure, it’d be an interesting gameplay feature to have a cheap nuclear plant, but it’s far from reality. Try to find a balance between representing the reality of the situation, and making a fun experience.

**Tech Side**

* Explain why your application needs any rights or needs to collect data using a consent screen that is displayed when the application is run for the first time.
* Don’t trust the client will behave itself, an attacker will write their own that can do what it likes.
* Make sure your demo is robust and will work on the day.
* When writing the essay remember to justify why you made a particular design or implementation choice and what path you did not take.
* We need ethics approval to have anyone other than ourselves do usability testing so ideally just ask your team members to test it.
* Scalability includes issues such as the caching of static assets to minimise downloading (either at install time or on demand).
* Automated tests that are run regularly are good.
* Ideally allow users to be able to remove themselves and all their data from the system.
* Fine not to use OAuth if doesn’t make sense but if rolling your own scheme you should think about password resets etc.
* When writing your essay consider Heroku tools add-ons you would have used or would be useful for scaling and how they would help.
* Can you application work in offline mode?
* Do you need an administrative interface for your system.
* Remember to sanitize anything and think about issues such as appropriateness of images especially where images can be shared between users (same applies to text). Do you need a scheme for moderation or reporting offensive language or images?