**New feature design process**

**Analysis**

The analysis stage should state what the purpose of the new feature will be, why it’s useful and briefly describe what it is (should not exceed 100 words)

**Design**

The design stage should go into almost painfully meticulous detail about what the feature is, exactly what it does, how it will work, how it will look and even a brief description of how it will work in terms of code. There should be no word limit but use just enough so that even a moron could understand what the feature does and how it works the use of diagrams should be heavily present in this stage.

**Implementation**

The implementation (code) stage should be well recorded in the dev log and follow all the best practices. Great care should be taken to ensure the modularity, quality and readability of code with extensive comments, consistent indentation structure and meaningful variable names.

**Testing**

Rigorous testing should be done to ensure the functionality as well as efficiency of the code is as close to perfect as is realistically achievable by the developer. To this end modules and the code as a whole should be designed such that probing of particular functions and variables should be easy and fast (no one likes testing).

**Docs**

Really just a super brief description of what each new variable means and what each new function does

**Evaluation**

Due to the non-commercial nature of this project the evaluation stage is really just a summary with any additional notes or information which doesn’t really fit anywhere else for example if a feature had to be majorly redesigned (which should have accompanying designs) , any problems or even learning experiences popped up they should be noted down here.

Oh and try and throw in an awful reference or pun here and there 😊

Quick example

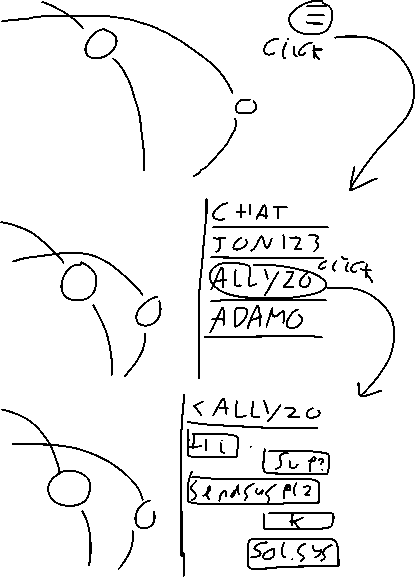
*Italics* denotes an aspect which would not be in the read design process.

**Analysis**

The new feature is a simple chatting function which allows users to communicate over the internet and share their systems. This is useful because it adds vital user functionality and a possibility for viral spreading of the software (the good kind). it should seamlessly integrate with the overall design and aesthetic of the program so far and barely use any resources at all. *(63 words)*

**Design**

basic idea



**Testing**

|  |  |  |
| --- | --- | --- |
| Variable / function | Test type | Worked? (if not explain) |
| SendText (String) | Works? | Yes |
| sendDataOverNet (function) | Works? | No function syntax was incorrect fixed now (see dev log) |
| userID(int) | Comprehensive range ( 0 - 1000) | Yes |

**Docs**

Var userID – the id the user will be given to identify them more easily to the server

Var sendText – the text which is being sent over the internet to another user

Function sendDataOverNet(userID, sendText) – takes in ID of user which is sending data and the data they are sending then used a socket based P2P model to send the data

**Evaluation**

No major problems or deviations occurred while developing this feature however the chat feature does have some problems with efficiency in transferring files I would suggest using some form of compression in later updates.