

- How the system is to be executed
 - used a mac for all programming, haven't tested it on anything else
 - execute using eclipse
 - programmed in java
- Step-by-step instructions on how to install your system. All hardware and software requirements for your system need to be listed (e.g., OS, browser).
 -
- A high level summary of what your system does. A list of the A+ features integrated into this project, and a list of any additional features over and above these features that we should be aware of.
 - System displays moods: There is a different degrees of 'angry' for the girlfriend. The more you fight with her the more angry she gets. She analyzes your responses to determine her mood and also yours and reacts based on that.
 - System accuses user of Cheating: The more the user and system fight, the more likely she is to accuse you of cheating. If you fight 4 times you enter a trust issues which she brings up past things you have said as evidence that you are cheating on her
 - Analytics of conversation: At the end of the conversation 'Dr. Phil' lets you know what you said wrong and why it made her angry. He tells you what you could say next time in order to make/keep her happy.
 - There are A LOT of different minor paths that don't go down a completely new branch, but there is different sentence structure based on the answer given by the user (especially when a break up or when she's accusing him of cheating)
- All the source files and dependent files that are used in your project, along with any subdirectories you placed them in (if any). For each file, give a brief (one to two lines) description of what the code or information in the file is for. Be sure to include any data file your system uses.
 - happy.java- the main conversation, where all the conversation happens
 - girlfriendmethods.java- where some methods are to check common spelling mistakes, check like words in order to handle them the same
- If your system uses any external libraries or APIs, you will need to list all of them, along with a link of where you got those libraries and APIs. For each external file you use, please indicate which parts of your system interface with them (the name of the file (s) containing the methods that interface with them).
 - n/a
- Lastly, enumerate the list of known bugs or issues with your system. Bugs include cases you have come across and didn't have time to fix before the submission date. Explain how it is triggered and what happens to the system (e.g., error message, exception caught, crashes). Issues include desired features that would make the system better and are to be considered for future improvement.
 - as mentioned in the not good conversation:
 - saying whatever leads to a break up.
 - when entering the conversation talking about relationship qualities, if the user says something the computer doesn't recognize its supposed to say "that's a

good one!" but regardless of what you enter, it outputs this, sometimes 5 times.

- doesn't spell check everything
- sometimes doesn't matter what the user says, the system just responds the same for everything.