**The procedure**

We started off by having several conversations with each other by imagining different scenarios and different paths the dialogue may take, always assuming the student character. These sample dialogues are provided in the appendix. After that, we combined the distilling phase with implementation and while adding all the dialogues to the AIML file, we removed duplicates, tried to combine similar questions or answers and added alternative ways of saying the same questions.

Based on these dialogues and given our limited time, we decided to do three sub-topics, viz. fridge, Xbox, and solar panels by adding relevant questions and answers and also steering the user towards those topics through other mechanisms (see the Handling Different Phenomena Section).

We than abstracted out some keywords for each of these subtopics and added common synonyms of the keyword using a thesaurus dictionary.

Given the nature of AIML, we tried to merge as many questions as we can into one and took a greedy approach with responses. For instance, any question that includes the word “pay” will be responded with the amount of summer bill and winter bill. This greedy answer that tries to give as much information as they are relevant in one go, is very helpful in reducing the number of questions that must be answered, however, it sometimes make the answers mechanic and non-organic.

We used binary variables to keep track of discussed topics. Once a topic is discussed, its flag variable is set to true so the bot doesn’t neither initiates another talk on the topic nor allows the user to bring it up. A limitation of AIML is that, these variables should be initialised in a category and if for any reason that category isn’t called (the user never asked the question in that category), the variables will not hold any value. We placed the initialisation in the “HELLO” category and by adding a greetings as the first utterance of the bot, tried to encourage the user to say a greetings and therefore initialise the variables.

**Handling Different Phenomena**

1. Not understanding users utterance: this case is inevitable and the way we addressed it is by
2. Pronouns: this cannot really be handled in AIML as it doesn’t perform any natural language processing. We tried to include pronouns in the categories we designed just to give an illusion of intelligence.

**The bot**

We chose the student characteristic and coded several attributes for the bot such as his name, number of people living in the house, their power bill amount, etc. These can be found in the “*assignment2.py*” file.

**Language Limitations**

There are many limitations in AIML, some of which were very hard to work with:

1. Not being able to initialise variables safely.
2. The “*that”* tag is very limited.
3. The conditionings are very basic. Nested if-else structures are not allowed.
4. No natural language processing
5. Being punctuation blind