FIT4009 Advanced Topics in Intelligent Systems
Assignment 2 (15%) – Small Chatbot
Due: April 24, 3 pm

The objective of this assignment is to implement a small chatbot using the dialogue markup language *AIML* (*Artificial Intelligence Markup Language*). Ideally, your chatbot should be either a customer who is interested in cutting down his/her energy consumption or a consultant that assists the customer. However, you can select another topic. Regardless of the topic, the chatbot should have a personality and a life, e.g., in the energy domain, it should live in a house/apartment, have a family, pets, etc, have likes and dislikes, and have specific needs and costs.

The main focus of the assignment is on dialogue management issues, such as focus handling, clarification sub-dialogues, and possibly handling non-cooperative behaviour. While it is important to provide useful information, you shouldn't spend all your time acquiring domain expertise. Your chatbot should also be able to chit-chat, and both ask and answer questions. It should be able to handle at least 10 dialogue turns, and answer at least 20 different questions in a particular topic, and possibly provide links to useful web sites.

Below is a list of preliminary tasks you can do in order to get an early start on the assignment. The assignment will be further specified in the following week, but in the mean time you can do the following.

1. Install the environment

1. Install Python 3.5 (or higher) on your machine.
2. Make sure that Python is installed correctly by opening a Terminal and typing
   `python --version`

   If installed correctly, this will print the version number of your Python installation, which should be 3.5 or higher.
3. Download `pyaiml3-master.zip` from moodle.
4. Extract the downloaded zip file to a directory of your choice. A new sub-directory called `pyaiml3-master` will be created there.
5. Download the files `testTK.py` and `test.aiml` into the same sub-directory (`pyaiml3-master`).
6. Test whether `pyaiml` is installed correctly by typing in the terminal:

   `python testTK.py`

   If installed correctly, this will print "Loading test.aiml... done (X.XX seconds)". It will also print a second line containing a prompt like this:

   `aiml>`

   In this line, type "hello" and hit enter. If installed correctly, the response will be "world" followed by another prompt. Type CTRL-D to break out of the AIML prompt.

2. Get acquainted with the AIML language and the pyaiml interpreter

1. Read the AIML overview at `www.pandorabots.com/pandora/pics/wallaceaimltutorial.html`, and the AIML 1.0.1 specifications in `alicebot.org/TR/2011/`. Additional information appears in `www.alicebot.org/documentation/` (all these links are in moodle).

2. Read the contents of the file `test.aiml`. Make sure you understand what each line means. Just to really make sure, edit `test.aiml` so that if you save the file and repeat Step 6 above, the word "hi" is printed instead of "world". Then edit `test.aiml` again so that the system says "hi" as an answer to two additional greetings (other than "hello"). Use AIML's <srai> to implement that functionality.

Note: you can also run the system using `test-startup.py`, which calls `std-startup.xml` to load the functions in `standard/`. Unfortunately, some of these functions have bugs, so use at your own risk.

3. Design a dialogue System for a "save energy at home" domain (or another domain)

1. If your chatbot is in the energy domain, visit the site `switchon.vic.gov.au/more-ways-to-save/top-10-ways-to-save-power`, and inform yourself about the domain. Extend your knowledge by reading further resources related to saving energy at home, e.g., the documents from Assignment 1.

2. To help you get started, imagine an energy consultant whose job it is to help people save energy at home. Imagine three people with different energy consumption patterns: (a) a student who likes video games; (b) a house-holder with three daughters aged 5, 6 and 11; (c) a self-employed engineer who has a workshop in his home.

   Write on paper what a short dialogue between each of these people and energy consultant would look like. The setting should be that all three want to save energy but are unsure how. The consultant tries to find out by talking to them (through questions) how they can save energy at home. For instance:
   STUDENT: Hello.
   CONSULTANT: Hello, how may I help you?
   STUDENT: My last energy bill was way too high and now I'm trying to find a way to get the costs down.
   CONSULTANT: I see. No worries. We can help you with that.
   STUDENT: Great.
   CONSULTANT: Let's see. How many hours per day do you usually spend at home?
   ...

3. While you are doing this, pay particular attention to the dialogue phenomena that you would be addressing. For instance "Let's see" would be a time management contribution.

4. Implement the dialogue system

Your chatbot should have the following specific capabilities (more to be added):

1. Pick up certain answers from its conversational partner, including his/her name, and use them in its own response.
2. Handle different ways of saying the same thing (use <srai> for this).
3. Add dialogue history using <that> or <this>. The chatbot should be able to answer at least three consecutive questions in a domain.
4. Not repeat the same question after it has received an answer.

5. Evaluation

The chatbot will be evaluated by me and by some of your classmates. You should conduct at least two evaluation sessions, so that you can make some changes based on the feedback you receive.