

# Project Proposal

Prepared for: David Hanson

Prepared by: Mike Pan

October 8, 2014

---

---

# EVA BOT ROADMAP

## Goal

To help create a software system that allows an external AI engine to control a realistic looking digital avatar in Blender. Specifically:

1. Motion authoring: Create a well defined Blender model rig and supporting scripts, that allow artists to create believable expressions that can be converted into motor movements that can be stored into a library for later use.
2. Motion playback: Create a system that allows the digital avatar to take command from the 'AI engine' and displays various expressions via some undecided channel (ROS, simple commands, etc). This playback happens in realtime.

## Project Outline (Starting October 1st, 2014)

### Week 1: Survey, Setting up. Planning.

The first week was spend on getting up to speed and communicating with different people about the requirement for the project, and what's the optimal way to rig the model.

Also worked on setting up a dev environment so I can test various things. But it's probably not critical for now, as most work will be done inside Blender.

### Week 2-3: Rig Control

With the rig almost ready (by Beorn), I can start by writing Python scripts that controls the Rig in a low level. The idea is to have a system that can control the following:

- A wrapper that is able to execute codes inside Blender's event loop continuously. (useful for networking and animation playback)
- Individual shape keys (as well as a combination of them)
- An 'animation player' wrapper that's able to create arbitrary expressions out of shape keys in realtime.
- These scripts will be written in Python3 and makes heavy use of the Blender API.

### Week 4: Rig Control: Saves and Load

- Ways to save and load arbitrary shape keys
  - Further work on animation player (from week 1)
-

---

## Week 5: Servo Control

As it's critical that the animations Beorn creates can be translated to servo movement for the real robot, I will work on a translation layer that transforms bone/shapekey motion in Blender into servo motion, and vice versa.

- Bone/Shapekey motion to servo
- Servo to bone/shapekey

Note, as this point, there isn't any code to communicate with any external engine.

## Week 6-7: High Level Rig Control

At this point, the code should have full control over the digital avatar by manipulating individual bones and shape keys. The goal for this week is to create a higher level abstraction to define a collection of shape keys as an expression. Candidates include:

- Eye target(s)
- Eye Blinks
- Head target(s)
- Facial expressions (e.g. smile, wink, confused)
- Physiological ( e.g. Blushing, sweat, nervousness)
- Speech (lipsync)

As this list should mirror closely with the API of the external 'AI Engine', the exact goal will need to be discussed in detail with the core programmers. Once the detail is finalized, we can start mapping these emotions to bone/shapekey controls inside Blender.

## Week 8: Polish and Interfacing with the AI Engine (PAU)

Part of this week will be saved for polishing of existing code and functionalities.

Start work on communication with PAU using Blender.

## Week 9+: Continue interfacing with PAU and Future Work

---