SHAPE
Smart Homes Altering Physical Environments
Sponsor: Mona Ghandi
Aaron Kwan, Andrew Lewis, Jonah Simon, Nick Strazis, Patrick Bowling

Motivation
Humans have the amazing capability of adapting to their surroundings. But what if their surroundings adapted to them instead? SHAPE gives users the ability to have their environment adapt to their feelings and desires at any given time through intuitive user input devices and emotion analysis.

Requirements
- Develop the following methods of interaction with the wall
  - Touch
  - Voice
  - Movement/Gestures
  - Thoughts
- Develop a form of communication between input devices and physical wall
- Apply statistical analysis and machine learning algorithms to collected data

Solution Approach
- Input Device → Method of Interaction
  - Mobile Application → Touch
  - Amazon Alexa → Voice
  - Affectiva → Emotion Detection
  - Microsoft Kinect → Movement/Gestures
  - Emotiv Headset → Thoughts
- Develop a RESTful API to communicate input data from each input device to the physical wall
- Store input data in relational database on server in preparation for machine learning

Architecture
- Clients: The clients are all very similar. They all read a user input and send that input to the server via HTTP POST request.
- Server: The server is a REST API built on Django and the Django REST Framework. It stores wall configurations in a MySQL server. Configurations are formatted as JSON objects. They are updated via HTTP POST requests and served via HTTP GET requests. Additionally, the server stores and formats the data for future machine learning.
- Wall: The wall is controlled by Rhino3D. Rhino3D receives the data from an extension called GrassHopper. GrassHopper allows custom C# scripts to feed Rhino3D data. The C# script is continuously performing an HTTP GET request to the server asking for the current wall configuration.

Input Devices
- EPOC: Emotive EG Headset. This device and application collects input based on the user’s brainwaves.
- Alexa Skill: This application collects input by listening to the user’s voice commands.
- Web App: The Web App is a touch friendly application that takes the user’s touch and mouse input.
- Kinect: The Kinect App collects gesture and dance data from the user
- Affectiva: Affectiva determines the user’s emotion by reading markers on their face through a web cam.

Results
- Built functioning alpha and beta prototypes for each of the devices listed in the Solution Approach section
- Developed an online server to communicate input device data to the physical wall using HTTP POST/GET requests
- Created a MySQL database on the server to store all data sent from input devices for statistical analysis and machine learning

Future Work
- Design and implement machine learning in server-side database
- Expand input devices to include bioreading smart watch
- Improve ease of use for input devices
- Enable automated testing for input devices and server

Glossary
- HTTP: Hypertext Transfer Protocol. A request/response protocol involving a client and a server
- REST: Representational State Transfer, a way of providing interoperability between computer systems through the Internet
- API: An intermediary allowing a standard programming language to communicate with a piece of software
- JSON: JavaScript Object Notation. A specific text format used to organize data
- MySQL: An open source relational database management system

Acknowledgements
Special thanks to Mona Ghandi, Hamid Esmaeilou and Yang Zhang