

Introduction

The aim of the CXA 2010 entry was to develop a lab observer software which will allow a staff or technical officer to remotely monitor equipment in a laboratory and switch off or shutdown the equipment when they are not in use so as to conserve energy and maintain a green eco-friendly environment.

A 3D model of the laboratory under observation was developed using Google Sketchup 7 and interactive features added using Germanium Web so as to make the application user-friendly and intuitive. An IP camera was integrated into the system to enable the staff to monitor the laboratory and identify unused computers.

The application was hosted on a server so as to enable the staff to monitor the lab remotely. The staff will be able to click the equipment on the 3D model or choose the computer identifier from a drop-down list to initiate a remote shutdown or hibernation. This will cause an email to be sent to the target computers. A mail client residing on the target computer, on receipt of the email, triggers off the appropriate action – shutdown or hibernate.

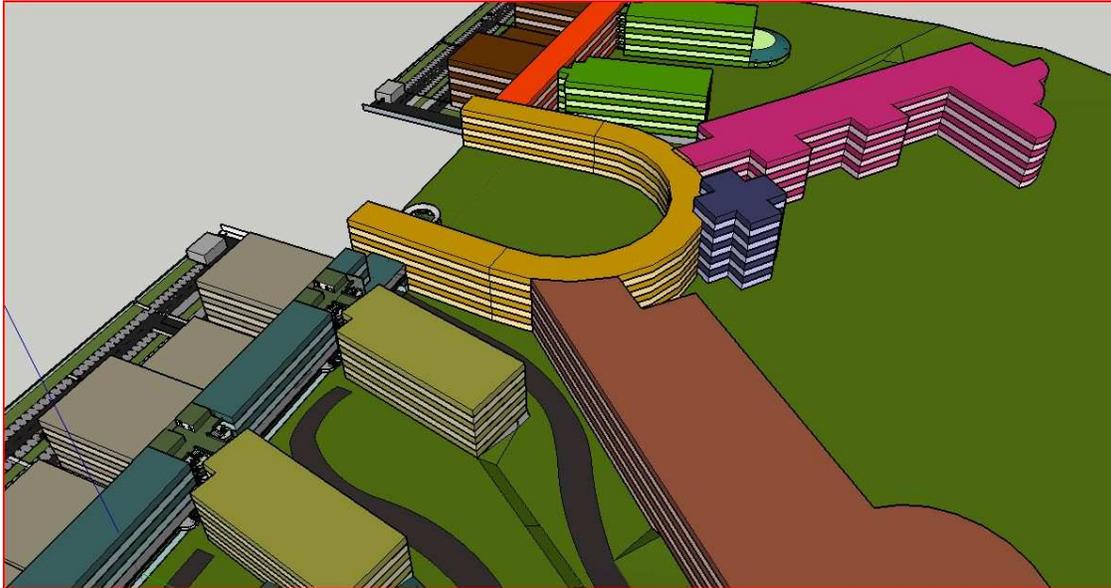


The use of the GermaniumWeb API was a pleasant experience, because the API is easy to understand. Furthermore, the sample codes helped a lot in facilitating my team's understanding of how to write our own set of codes. The ease of implementing the codes using Javascript also led to our increased interest in learning how to use Javascript.

Model Description

The model is mostly just coloured rectangular boxes that are made to be as close of a dimension as the actual Temasek Polytechnic architecture as possible, based mostly on imagination and memory. Some 3D models are downloaded from Google Sketchup's 3D Warehouse, but approximately 95% of the model is done by us.

There are some details included to depict the Engineering School's Concourse as well as Level 3 of Block 21, where our project labs are.



Design Processes

Initially, we had some problems with stacking 2 levels of a building in the Building Composer, and they kept overlapping each other. However, we were very glad that we attended the GermaniumWeb Training, organized for participants of Code::XtremeApps2010. The instructors were very helpful and taught us that we would have to separate the building levels during the 3D modeling process.

The simple instructions and tips on the GermaniumWeb's official site also helped to improve our 3D modeling experience. An important tip was to reduce the number of "triangles" used in our 3D model by reducing the extent of details where possible.

Coding Processes

In order to position our placemarks, we had to obtain the corresponding coordinates needed. We're thankful that amongst the sample codes, one of them was to obtain coordinates by double clicking. With this information, we edited the code such that a list of coordinates will be printed out upon a mouse-click generated event. We also had a pleasant time experimenting with the codes that will move the "Eye" around and create automated changes to the viewers' perspective. We were so fascinated with the API that we created a special presentation slide based on GermaniumWeb's API.

Conclusion

Overall, it has been a great experience using GermaniumWeb as a development tool. We enjoyed the process of writing the codes