Wentworth Institute of Technology
Comp650
Senior Project in Computer Science
Summer 2012

Members: Arthur Charlton, Alex Jackson, Timothy Brantley II

Project: Multisaver

***Project Aims and Description**:* Our project is to design and develop a multiple monitor screensaver that is enjoyable to look at. The intended user of our project is the multiple monitor enthusiasts. Windows comes with a poor selection of multiple monitor compatible screensavers and the two that it does have do not take advantage of effects only capable on multiple monitors. We want to remedy this oversight by supplying a free to use screensaver that takes full advantage of all of the user's monitors. The completed project will be a downloadable program that a user can simply double click to install the screensaver in their computer.

One of our key features will be for individual monitors to function independent or cooperative in any mixture. Users will be able to configure the screensaver to choose what mode each monitor will run in. There are three main display modes that monitors can participate in.

- **Multiview Maze:** Each monitor acts as a different perspective of an AI solving a maze. The central monitor will render what would normally be seen in front of the user. A left monitor at the same height would render the left peripheral vision, etc. When multiple monitors cooperate in this manner, they will give the illusion of an expanded field of view. This is similar to Nvidia Surround or AMD Eyefinity.
- **Multiple AI Maze:** Each monitor can host a separate AI trying to solve the maze. For example, if you had three monitors you could watch three different AIs trying to solve the maze all at the same time. Furthermore one monitor can be selected to display a map showing each AI's progress.
- **Advanced Photo Album:** Each mode displays a slideshow of images from a selected photo album. Like the other modes each monitor can display a separate album and work independently or all monitors can work out of the same album. Unlike the traditional Windows slide show screensaver transitions from one image to another will take advantage of 3D graphics for entertaining fade in and fade out effects.

Monitors can mix and match independence or cooperation in all of the modes at the same time too. For example: 3 monitors could cooperate in the multiview maze and a fourth and fifth could be displaying two separate photo albums independently.

Finally the project will be open source and opened up to the community for development after the end of the semester. This way if anyone wants to continue development of the project on their own they can. Also they can use the source code as a learning resource if they are curious about multiple monitor graphics.

***Conduct of the Project**:* Our project will take advantage of the open source development model with the variation of weekly meetings of our group members. We will take advantage of source control technologies such as Git and collaboration services such as Github to manage building the project. Our weekly meetings will be used to review issues from the previous week and set our goals for the

upcoming week.

Necessary Skills:

- **Screensaver Integration:** How to make the application itself be controlled by the Windows OS.
- **Screensaver Installation:** How to allow users to install the screensaver and any necessary dependencies.
- **DirectX 9**:
    - **HLSL**: For the 3D graphics and shaders
    - **XNA**: Our programming interface with DirectX
    - **Render Targets:** For rendering to multiple screens at the same time.
- **WPF**: For displays the configuration panel for the user to select the screensaver options.
- **C# Optimization:** While each monitor itself may be doing a fairly simple task rendering to 3+ monitors can be quite taxing on the GPU. It will be necessary for us to learn techniques for optimizing the processing so that the end result is a smooth experience for the user.

As a group we will need to research any skills we are missing. These skills will be acquired in a combination of members teaching each other and independent exploration via MSDN and stackoverflow. Collectively as a group we have all of the necessary skills, it currently is a matter of bringing all of us up to speed.

Necessary Software:

- **Windows 7 x64:** For running using the rest of the software.
- **Visual Studio 2010 Ultimate:** For all of the programming
- **Office 2010:** For all of the documentation
- **Git:** For source control
- **Github.com:** For programming collaboration
- **Git Extensions:** GUI for Git

Statement of Deliverables: The project will consist of four types of deliverables. Documentation, source, tests, and the installer.

- **Documentation:** All of the projects documentation
    - **Proposal** - Our initial goal for the project.
    - **Project Specification** - The requirements document detailing the aspects of the project.
    - **Design Document** - The document explaining how the program will be implemented and achieved.
    - **Class Diagram** - A visual graph of our source code's class structure.
- **Source Code:** The complete codebase for the project.
    - **Git Repository** - The complete history all of the source code and how it progressed forward. The final state of the repository will be the source code ready to compile to our installer.
    - **Github** - A copy of the git repository will be available on Github for anyone to download or fork if they want to start their own development.
- **Tests:** Programs and routines to ensure program functionality.
    - **Stress Test:** A test designed to push the program to its limits so we can discover what those limitations are.

    - **Autotest:** Visual Studio's autotest functionality for ensuring that a program functions as desired. Will require writing all of the logical asserts that challenge the program state at different points ensuring that they are correct.
- **Installer:** The finished program will be compiled and wrapped by the installer. The installer will contain all of the necessary dependencies for the project and will place the screensaver application in the correct location.