St. Lucie County School Concurrency Information System

Project Report

URP 6905
Internet GIS
Yong Hong Guo

Department of Urban and Regional Planning
University of Florida

December 16, 2008
Introduction

This project attempts to use the Internet to map schools and provide school concurrency information to the user for St. Lucie County, Florida. Currently, desktop GIS are applied by the school board and the planning department to track whether developments meet the school concurrency requirements in the State of Florida. Using the Internet to provide school information has recently emerged as an alternative. This project explores a way to use the Internet to map schools and provide relevant school concurrency information for the user. It is hoped that using the Internet for school concurrency can replace the desktop applications, and this project explores a way to get to that destination in certain degrees.

Audiences

The users of this application are the developers, the school board, and the planning department of St. Lucie County. Because residential development will generate students for elementary schools, middle schools, and high schools, based on the State of Florida Statutes, residential developments must meet the school concurrency requirements to ensure adequate public school
capacity is provided or under construction when a development is approved. As a result, new single-family and multi-family residential developments will be affected by the concurrency law.

Because new single-family or multi-family developments will generate students for local public schools, impact fees will be assessed by the planning departments during the development approval process. Therefore, a proper evaluation of the impact of a new development will be important for the local government. Developers will also need to evaluate the impact of a development because they need to estimate the impact fees when submitting their development proposals for approval. The school board will need an application to help them plan the school capacities based on the current as well as the projected enrollment incurred by the new residential developments. This project will provide such an application to be used by the developers, the local government, and the school board.

In St. Lucie County, Florida, there are three CSAs (Concurrency Service Areas) for public schools, which namely are Blue, Green, and Red. Each CSA has enrollment, capacity, and LOS (Level of Service) numbers for elementary schools, middle schools, and high schools. Basically, if enrollment is more than capacity, it will create a concurrency problem because the school with the existing capacity is not able to accommodate more students. The local school board evaluates school capacity situations based on the LOS figure. If a new development makes the LOS in a CSA more than 100%, it means the development will generate more students than the capacity, and that is not considered concurrency-oriented. The LOS criteria may vary according to the different school district. Some school districts may adopt the criteria of 110% – 120%. This project will adopt the 100% threshold, however.
Data

The data of this project is from the St. Lucie County School Board. The data includes GIS data, Fishkind data, and PSFE (Public School Facility Element) data. GIS data provide information, such as geographic locations, for schools and CSAs. Fishkind data provide capacity, enrollment, and LOS information for each school and each CSA. PSFE data provide multipliers for each type of development, such as single-family and multi-family developments.

GUI

The graphic user interface (GUI) uses a splitter to create a frame to accommodate the Microsoft Virtual Earth (VE) map and the relevant functions. The left side contains the VE map with the controlling dashboard; the right side contains two modules of functions. Module I shows CSAs as well as the enrollment, capacity, and LOS information for each school on the top. Module II calculates the LOS for each school type in a CSA on the bottom. In addition, checkboxes, buttons, GridViews, dropdownlists, textboxes, tables, and labels are used in the two modules. The school information will be presented in GridViews in Module I, and the calculated results will be populated in the tables marked in red labels in Module II. The GUI is presented in Figure 1.

Software Architectures

The software architectures used in this project are the Virtual Earth Mashup, the ASP.NET, the SQL server, Virtual Basic codes, and Java Scripts. Using Virtual Earth as a platform to map schools is based on its Bird’s Eye function. In addition, for an elegant design of the map frame,
css and splitter.js are applied. The above elements are put together to form the design and functions of the application.

![Image of the St. Lucie County School Concurrency Information System](image.png)

**Figure 1**

**Functions**

The functions of this application are three-fold. First, it is a mashup. School and CSA information is presented by the popup boxes. When users click on a school icon, the contact information as well as schools’ URL links will show up. Also, when double clicked, the map will
zoom in to that school. Users can also interact with the CSAs and schools by turning on/off CSAs. Second, it is a school concurrency information display application. Users can seek the school enrollment, capacity, and LOS information through selecting a typical school in a CSA. Third, it is a school concurrency calculation system. Users can calculate school concurrency based on the user input. New development units will be converted into concurrency information through computer calculation. User inputs and the school concurrency information are therefore interactive. In addition to the above functions, a tutorial link is provided.

**How to Make It Work**

First, the CSA shapes are created based on the latitudes and longitudes of each polygon’s vertexes, which are extracted from the GIS data. Then, they are added into the Virtual Earth by JAVA scripts to form three CSAs with different colors correspondent with their CSA names. JAVA scripts are also written to control the on/off function of the CSAs. After that, schools are input into the Virtual Earth map utilizing the Mashup technique. Pushpins are customized using school icons and marked different colors representing different types of schools. In the mashup, school addresses and URLs are populated in popup boxes, and CSA information such as CSA enrollment, capacity, and LOS are also printed in the CSA popup boxes.

ASP.NET is applied to create user interfaces on the right side of the frame. For this purpose, Fishkind data are populated in a table stored in the SQL server. This allows the data to be used by the application’s user interface. When users query the enrollment, the capacity, and the LOS information from selecting a school name in the school dropdownlist, relevant information on the sever side will be passed into the GridViews. SqlDataSource is used to achieve this function.
A similar process is undertaken for the calculation function. In this function, two textboxes, a dropdownlist, three tables, and a button are created. When users input a value into the textbox, the texted information will be converted into numeric numbers and calculations are made for CSA concurrency, LOS, and the added students. When non-numeric numbers are entered, the application will automatically convert them into zero. Visual basic codes drive this function.

**Evaluations**

The application offers a user friendly interface which performs excellently in displaying the school and the school concurrency information. Also, the school concurrency calculation function performs outstandingly to allow users to interact with the concurrency information through user input. Important school concurrency information, such as whether a new development meets the school concurrency requirement, calculated LOS for different type of schools in a CSA, and the number of new students added for each type of schools, is provided. These are meaningful functions that are applicable in the real-world operations.

The application has rooms to improve, however. First, interactions between the school concurrency information display (Module I) and the mashup map should be considered. Currently, the school concurrency information display is a static process. When users choose a school from the school list in the right panel, the corresponding school on the mashup map on the left does not interact with it. Therefore, a function which can zoom in to the selected school on the mashup map on the left panel but is controlled by the dropdownlist on the right panel should be created. Second, hyperlinks of URLs should be added in the school mashup popup boxes. In addition, lines should be broken in the CSA mashup. Third, school pictures, videos, and so on
should be added to the school mashup. In this case, feeds can be considered to be used which allows information to be updated real time and remotely.

**Knowledge Learned**

From the project, the knowledge such as how to combine the mashup and the ASP.NET together into one interface is learned. In addition, how to create a calculation function in the ASP.NET, which is to put JAVA scripts and Visual Basic codes together into one aspx file, is learned. Moreover, how to use a customized, user friendly interface using the css and splitter.js files is also learned.

**Conclusions**

From the project, it is obvious that functions of desktop applications can be realized on-line. The Internet provides an excellent platform to achieve on-line information display and calculation; as a result, more information such as pictures, videos, and hyperlinks can be added onto the application. This will create more functions and bring users more convenience than the desktop applications. School concurrency is a good example that well illustrates the advantages of the Internet platform as this project shows.

For the project per se, more calculation modules can be added into the application. For example, a function that calculates the specific monetary impact fees based on the number of students generated can be added into the application. In addition, newly added modules can be customized so as to meet different customers’ needs. Moreover, as mentioned previously,
interactions between the mashup map and the school concurrency information display should be added. They will make the application more user-friendly, versatile, and flexible to use.