Design and Implementation of an Interactive Building Information System.

Graduate Project

By
Sreekanth Uppaluru

Committee Members

Dr. Mario Guimaraes, Ph.D.
Chairman

Dr. R. Stephen Dannelly, Ph.D.
Member

Dr. David Thomas, Ph.D.
Member

Texas A&M University - Corpus Christi
Department of Computer Science
III. Operation of the Input and Review Program

This is the program used by Operators.

A. Program Entry

1. From the operator's terminal while the date and time are displayed on the screen press the Return Key. If the program is in memory the computer will respond with the first request for input. If the program is not in memory it will be loaded into memory from the disk. The computer will then respond Loading INV&REV. Loading will take about 20 seconds.

2. From the main terminal while the date and time are displayed on the screen press the Return Key. Choose the appropriate code from the Master Menu and the computer will then go to the start of INP&REV or will load it and then go to the start of the program.

B. Logging Options

1. The first question is the date and time the user wants for logging or review. There are two options - the current time or fill in the date and time. Press 1 or 2 and then the Return Key. If the date is filled in follow the exact format shown on the screen Mn/Dy/Yr Hr (e.g. 01/07/81 13). A 24 hour clock is in use. The hour will not change while logging is in progress. The hour changes on the hour so logging should be done shortly after the hour, not before the hour.

2. The computer will then display the logging menu. Choose the appropriate code. If "1.0p On Duty" is chosen a menu of operator names will be displayed. Choose the code for the appropriate name. The engineer can change the list of names as needed. All other codes will lead to the display of a logsheet table.

3. Logsheet headers will be displayed along with one line of data and a horizontal one line logging menu as follows:

   1=LOG, 2=LOGNXT, 3=LOG@, 4=CORR, 5=NXT, 6=BCK, 7=FWD, 8=RVRS, 9=DONE

Choose the appropriate code

1=LOG  This allows the user to log data for the date and time specified at the beginning of the TNF1&REV program.

2=LOGNXT This allows the user to log data for the next hour or day (depending on whether it is an hourly or daily logsheet) after the final line displayed on the screen.

3=LOG@  This allows the user to fill in any valid date and time for logging.

4=CORR  This allows the user to correct any data on the last line displayed on the screen.
## Table of Contents

1. Abstract .................................................05
2. Background & Rationale .....................................06
3. Narrative ..................................................09
4. Environment ..............................................27
5. Procedure ................................................28
6. Results and future work ...................................46
7. References .................................................47
8. Appendix - A (Table details) ..............................50
9. Appendix -B (Flow Diagram) ..............................53
10. Appendix - C (ER Diagram) ...............................54
11. Appendix - D (Form-Tables details) ....................55
12. Project code ..............................................56
   a. Module .................................................56
   b. Login .................................................57
   c. Splash ...............................................62
   d. Main Form (FrmMain) ..................................65
   e. User Choice for Viewing (FrmUCV) ....................77
   f. View Building Details (FrmVBD) .......................81
   g. View Room Details (FrmVRD) ..........................91
   i. Query Database (FrmQuery) ..........................101
j. User Choice for Modificatiob (FrmUCM) ...........................................105
k. Add Building Record (FrmBDA) ..................................................111
l. Update Building Record (FrmBuildingsUpdate) ..............................119
m. Delete Building Record (FrmBDD) ..............................................129
n. Report on Buildings (FrmCDBR) ...............................................137
o. Add Room Record (FrmRDA) ....................................................140
p. Input for Room Update (FormInput) ...........................................151
q. Input for Room delete (FormInputDelete) .....................................163
r. Delete room Record (FormRDD) .................................................166
s. Report on Rooms (FrmCDRR) ...................................................176
t. Export Database (FormExport) ..................................................180
u. FTP (FormFTPVJR) .................................................................184
v. Module FTP ................................................................195
w. Database Admin. (FrmDBA) .......................................................200
x. About BIS (FrmAboutBIS) .........................................................207
y. User Details (FrmUsers) .............................................................212
# Table of Figures

<table>
<thead>
<tr>
<th>Number</th>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Fig 1</td>
<td>Login</td>
<td>11</td>
</tr>
<tr>
<td>2</td>
<td>Fig 2</td>
<td>System Permissions.</td>
<td>12</td>
</tr>
<tr>
<td>3</td>
<td>Fig 3</td>
<td>User Choice for Viewing.</td>
<td>13</td>
</tr>
<tr>
<td>4</td>
<td>Fig 4</td>
<td>View Building Details.</td>
<td>13</td>
</tr>
<tr>
<td>5</td>
<td>Fig 5</td>
<td>View Room Details.</td>
<td>14</td>
</tr>
<tr>
<td>6</td>
<td>Fig 6</td>
<td>Query Database.</td>
<td>15</td>
</tr>
<tr>
<td>7</td>
<td>Fig 7</td>
<td>User Choice for Modification.</td>
<td>16</td>
</tr>
<tr>
<td>8</td>
<td>Fig 8</td>
<td>Add Building Records</td>
<td>18</td>
</tr>
<tr>
<td>9</td>
<td>Fig 9</td>
<td>Update Building Details</td>
<td>19</td>
</tr>
<tr>
<td>10</td>
<td>Fig 10</td>
<td>Delete Building Details</td>
<td>20</td>
</tr>
<tr>
<td>11</td>
<td>Fig 11</td>
<td>New Room Records</td>
<td>21</td>
</tr>
<tr>
<td>12</td>
<td>Fig 12</td>
<td>Update Rooms</td>
<td>22</td>
</tr>
<tr>
<td>13</td>
<td>Fig 13</td>
<td>Delete Room Record</td>
<td>23</td>
</tr>
<tr>
<td>14</td>
<td>Fig 14</td>
<td>Report On Rooms Database</td>
<td>24</td>
</tr>
<tr>
<td>15</td>
<td>Fig 15</td>
<td>FTP</td>
<td>26</td>
</tr>
</tbody>
</table>
Abstract

This project is the development of an interactive, user interface tool for the Texas A&M University-Corpus Christi Physical plant. The project also requires the development of a database containing the information for all existing buildings on campus. This database will be used to keep track of all those buildings used by different departments on the campus including details like, building conditions, locations, area, volume, room details and special facilities of that particular building. The system will provide a user-friendly environment to access the database for producing various reports.
BACKGROUND AND RATIONALE

The Texas A&M University–Corpus Christi physical plant is responsible for the maintenance of all buildings on campus. When the physical plant has to process a work order on a particular building, the work team needs to know the existing conditions and other data such as height, width, area, location and perimeter, etc.

In view of the increase in pending jobs and the complexity of the university facilities added strain was placed on both administrative and maintenance staff, thus a database was created by the Conrad Blucher Institute (CBI) on behalf of the university. This was aided by a computer-based combination of digital map and relational database technology. The two technologies are collectively called Geographic Information Systems (GIS). The system, which was developed, was called the Facilities and Management Information System (FMIS). A GIS can combine graphics in the form of digital maps with a relational database. This combination of graphics and data allows a user to identify an object on an electronic map and retrieve attribute data about that object from the database.

But most available GIS packages are supplied with a software interface that permits functions within the GIS to make queries to a database. Also, most GIS software will interface directly with popular relational database products.
But there are several disadvantages in this:

- Reliance on vendor-supplied GIS software requires links to third party databases. These database interfaces are generally implemented at binary-code level and may change with each release of the database software, which would require continual code modification by the GIS software vendor to retain database compatibility.

- Restriction on the user to the subset of queries, query structures and file structures supported by the GIS software vendor.

In order to overcome this difficulty the FMIS has achieved database and GIS interface independence by creating a separate X-windows/Motif application.

The FMIS project uses GIS software developed by Genasys System. Inc. This software is called Genamap. It is intended that the FMIS will also be available to the CBI to provide hands on experience with a working database to its GIS students. Ingress has been selected as the relational database in use on the Unix platform primarily due its availability at the CBI.

Commands are entered into Genamap through the Genamap shell, which is modeled after the Unix bourne shell. The Genamap shell interprets the entered commands and calls the appropriate program to execute or act upon the data.
The shell approach to the command entry permits access to all available routines with Genamap. However, it is an inefficient user interface. Most commands within the map require several options to be passed as parameters to the program, and users generally must execute several commands in a specific sequence to achieve the desired results. Consequently, command-line entry results in a great deal of typing even for rudimentary operations.

In other words, this system is intended only for an expert user who is familiar with GIS and Unix. This makes the existing system very difficult to use for many potential users. Since the database is residing on a Unix platform, most of the potential users don’t have access to it.

A user-friendly, PC-based system would be an efficient and cost effective alternative to the existing one. This will be quick, dependable and very convenient to modify and query. The data will be dynamic, easy to modify and can be viewed by more than the person maintaining the database. This should be made available to all the required people throughout the campus.

This project is centered on designing, developing and maintaining a user interface system for the existing buildings on campus using Microsoft® Visual-basic® as a front end and Microsoft Access as a back-end tool. The end product will enable the authorized users to browse, add, edit and delete the data.
Narrative

The goal of this project is to develop a user interactive tool for the existing Buildings and Rooms part of the FMIS database. This system is developed with user-friendly features, so that user can navigate with the use of mouse, buttons and check boxes. At the same time this system has sufficient security features to avoid unauthorized users from entering the database and to restrict the unwanted data manipulation by authorized users.

The existing FMIS database is a relational database with two primary tables as explained in Appendix - A. The building table consists of 16 columns and 55 rows and contains the details of all buildings on campus. The room table consists of 21 columns with 1655 rows and describes each and every room of those buildings. Most of the columns in the database are in numerical form instead of actual value (i.e., if a column is describing the location of a building, then it is represented with 1, 2, 3, etc.. where as 1 stands for on campus and 2 stands for off campus.) But, in the final product actual values will be displayed instead of numeric code values.