Date Sheet Scheduler

By

Raja Jahanzeb Khan 2009-NUST-BIT-273

Department of Computing

School of Electrical Engineering & Computer Science

National University of Sciences & Technology

Islamabad, Pakistan

2015

Date Sheet Scheduler

By

Raja Jahanzeb Khan 2009-NUST-BIT-273

Department of Computing

School of Electrical Engineering & Computer Science
National University of Sciences & Technology
Islamabad, Pakistan
2015

CERTIFICATE

It is certified that the contents and form of thesis entitled **"Date Sheet Scheduler"** submitted by *Raja Jahanzeb Khan* (*2009-NUST-BIT-273*) have been found satisfactory for the requirement of the degree.

Advisor: _____

(Dr. Mian Muhammad Hamayun)

Co-Advisor: _____

(Dr. Sharifullah Khan)

DEDICATION

To Allah the Almighty

&

To My Parents and Faculty

ACKNOWLEDGEMENTS

I am extremely thankful to my Advisor and Co-Advisor, Dr. Mian Muhammad Hamayun and Dr. Sharifullah Khan for helping me throughout the course in working on my final year project (FYP) and to Sir. Mujtaba Haider for initiating this idea. Their guidance, support and motivation enabled me in achieving the objectives of the project.

Lastly, I thank my parents, colleagues and friends for their constant encouragement without which this project would not have been possible.

ABSTRACT

At present, SEECS technical personnel use manual process of scheduling exams and in specific date sheet generation. This process is hectic and time consuming which on average takes up to weeks to generate the date sheet for different type of exams such as One Hour Tests (OHTs) and End Semester Exam (ESE).

The intent of this project is to resolve the above issues. Technical staff currently perform manual extraction of information from an excel sheet obtained from LMS portal. The content of the sheet are comprised mainly of students, courses & faculty data carefully filtered after spending an ample amount of time to remove redundancies. The end result of this process is a data and a model file, which are fed into an optimization software to produce optimal scheduling for these exams. We automate this process, thus saving considerable time and manual effort.

Chapter 1

Introduction

This chapter gives a detailed introduction of this project. The discussion in this section would be about the motivation behind this and to discuss the goals of this project.

1.1 BACKGROUND:

Currently exam scheduler is the main issue being faced by management due to introduction of flexible semester system in NUST School of Electrical Engineering & Computer Sciences (SEECS). Hence, there is a need for formulation of date sheet in accordance with the exam policies of SEECS.

I have therefore provided an efficient solution to cater this problem in an effective manner. The solution is windows-based desktop application that utilizes the best of algorithmic and SQL based relational database approach. With the development of this system, the difficulties faced by SEECS technical staff in hectic task of date sheet generation would be resolved.

SEECS in accordance with its own development team is currently working on the same system but since due to the fact that they are following a proper software development lifecycle which not only involves the development of similar system but with more advanced features. They also perform proper testing of the system. Also that due to dynamically changing backend of SEECS database, the development team has to be patiently in line with the changes and adopt to them to provide an end product which is fully functional and is flexible enough to adjust along future changes

1.2 MOTIVATION

As described above as well, SEECS is in dire need of an automated system which not only allows its operational staff to relish the perks of software development system for advancement but to be more efficient and committed to their work by

not only putting all of their efforts in exam scheduling tasks. In short, they need to be more productive in their day to day tasks.

It therefore, becomes a responsibility of the students of SEECS to complete this job. My sole aim behind undertaking this role of developing such an automated system is to compliment the technical staff in achieving their objectives and not the least being a faithful student of this institute for last 4-5 years provide my services to it. This would be a single drop out of a sea of blessings and respect that I have always had and till date having only because of my beloved school and university.

Additionally, this would also allow me to polish my skills in software development and having this reputable project added to my career profile, I might get a better opportunity in industry.

This led me to choose this as my final year project.

GOALS AND OBJECTIVES

As my prime objective was learning and serving, and also my objective was only to develop a system that would perform the automation which would also give me hands-on experience with the latest development tools and techniques and methodologies. The requirement engineering phase was also very interested phase since due to the fact that SEECS database keeps on changing periodically, hence apart from pure technical aspect during engagement throughout this project gave me ample opportunity to polish my management and soft-skills.

From implementation point of view I was to develop a system that has the functionality of date sheet file generation in a specific format.

Chapter 2

Requirement Engineering

This chapter discusses requirement gathering phase of date sheet scheduler and its features. It will discuss the technologies and tools that were used.

The first step in this project was to study the current system of how SEECS staff generate the date sheet, their problem statement and what manual process they go through to accomplish this task. Similarly, I acquired schema of back-end database to analyze the relationship b/w each table and how can I utilize this database into my system. Finally an input dataset in the form of excel sheet was required that was the essence of this whole system and also the input of my system.

2.1 Current Dataset Preparation Method:

As discussed, SEECS currently follows the following method for preparing dataset which is then further processed.

We need to prepare the student data for processing in later steps. We will generate the "Exam Scheduler Report" from LMS for current set of batches, in the following manner.

1- Open Undergraduate Programs under School of Electric Engineering and Computer Science (SEECS)
2- Select the current semester for the latest 4 batches in BSCS, BESE & BEE programs and Click Select Courses
3- Remove the duplicate course resulting from merged courses
   Example: We will keep only one of the following courses checked:
   CS100 Fundamentals of ICT (BESE-5A&B)) (keep this checked)
   CS100 Fundamentals of ICT (BESE-5B) (uncheck this one)
   In case of Meta Course , keep the meta course and uncheck the original A & B courses (all)
4- Download in Excel format for further processing
5- Open the excel file and do the following adjustments
   a- Delete first 6 rows

b- Copy the Course_Code column and call it "Corrected_Course_Code"

c- Sort the sheet in ascending order on Course_Code

d- Delete the following courses form this sheet if any.

- Lab Courses (as we do not schedule them for exams)

e- Rename this sheet as "master"

2.2 Date Sheet Preparation

1. Use the list of courses with different course codes but same course contents (for ACB)
   a. Take the Course_Code for one of these courses, and replace all different Course_Codes in Corrected_Course_Code column.
   b. Move Islamic Studies and Engineering Drawing data to separate sheets, if any. (Delete empty rows)
   c- Now sort the sheet using Corrected Course Code in Ascending order.
2. Copy the Course_ID and Corrected_Course_Code to a new sheet and Remove Duplicates by selecting the Corrected_Course_Code only.
   a- Verify that we have the same number of rows in both Course_ID and Corrected_Course_Code
   b- Rename this sheet as "course_codes" (These are the course conflicts w.r.t the students)
3. Open the Visual Studio Project entitled "WindowsFormsApplication1"
   a- Open Form1 and double-click on "Combine Courses" to view source code executed on its click event

      Search following declarations in code and set values according to input sheets:

      Const int NUM_STUDENTS = 6824;

      Const int NUM_COURSES = 44;

   b- Verify the Excel Workbook path, for example:

      Sting workbookPath = "C:\exam_scheduler_report.xlsx"

   c- Sort the master and course_codes sheet on Corrected_Course_Code.

d- Close the worksheet, run the application and click on the "Combine Courses" button

Wait for the process to finish

e- Name the Last columns as "Combined_Course_ID".

4. Check the source code for Resolve Conflict 2 Button.
   a- Run the verify number of Students equal to the total rows in master sheet.
   b- Also check the conflicts using Course IDs; Courses studies by a student should be in its conflict list.
5. Now we need to create Batches
   a- Insert 4 columns after Course_Name
   b- Data-> Text to Colls (delimited by parenthesis, then delimited on hyphen).
   c- Create Batches (Delimited by Fixed Length)
   d- Concatenate Program and Batch in the fourth column. (e.g. BEE and 6-> BEE6
   e- Copy / Paste the Program and Batch info in a new col, as values.
   f- Delete the formula column.
6. Copy Batch Column (without sections) and Combined_Course_ID to a new sheet.
   a- Remove Duplicates (Both) and Sort Ascending.
   b- Copy batches information, day and slots to Data File (i.e. datesheet.dat)
   Important: Copy batches info through notepad to avoid formatting issue.
   c- Take note of number of courses and number of days.
   (Max. Courses in a Batch should be <= No of Exam days for ESE)
   This constraint should be relaxed for OHTs.
   d- Try to minimize the number of common students effected
   (for example exam sessions per day).
7. Verify the Model file and constraints; tweak if necessary.
8. Run the sw.exe and solve by consulting abc.run file and get output in a text file. Use the following code as an example:

```
Reset;
Option solver gurobi;
model C:\Exam_Scheduler\Spring_OHT12\datesheet_spring2-15.mod
data C:\Exam_Scheduler\Spring_OHT12\datesheet_spring2-15.dat
solve;

for { (d,t) in day_time }
    for { s in subject }
{
    If  X[s,d,t] > 0.1 then
        Printf "%d, %d, %d\n",d,t,s>
C:\Exam_Scheduler\Spring_OTH12\datesheet_spring2015.txt
        };

};
```

9. Import in Excel and move Course_ID to the First Column.
10. Insert Vlookup for Day & Slot in master sheet
11. Copy/Paste Day & Slot as values
    a- Create a new sheet with Day and Slot information
    b- Add Vlookups in master sheet and copy paste the resultant info as values.
    c- Copy the master shet; Rename it as "datesheet"
12. Remove All Columns except Instructor Name, Course Name, Batch, Course Code, Day and Slot
    a- Reorder Cols to: Batch, Day, Slot, Course Code, Course Name and Instructor Name
    b- Remove duplicates, Sort by Batch, Day, Slot & Course Code.
    c- Delete the Day & Slot Columns (Numeric ones only)
    d- Save it as a new workbook "00_Datesheet_XYZ". (Replace XYZ with Semester_Exam)
    e- Verify and Send to ACB for Validation; once verified, proceed to next steps.

Chapter 3

Project methodology and implementation

This chapter discusses different methodologies, tools & techniques used during the implementation phase.

Tools:

Programming Languages:

C#

Development Tools:

Visual Studio, MSSQL

Data analysis Tools:

Microsoft Excel

Output Files:

.dat, .mod

Components:

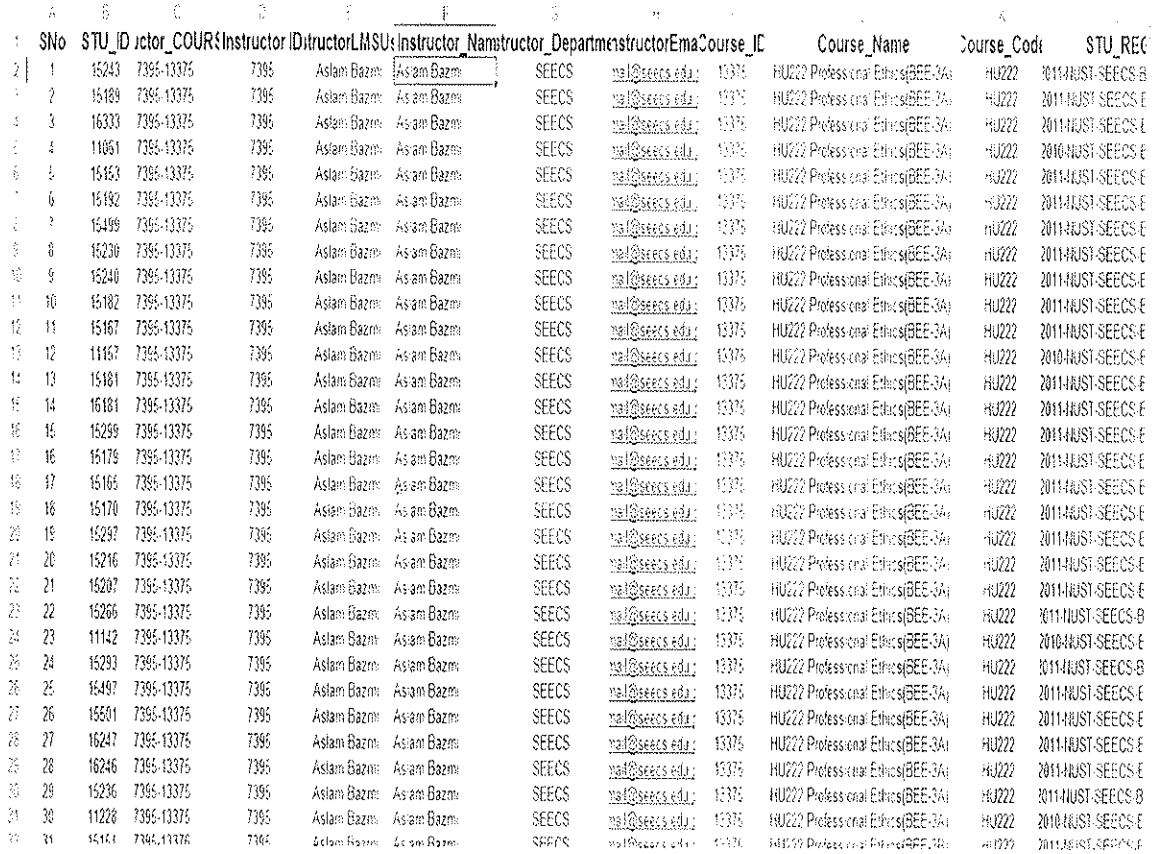
- Dataset
- Database
- Software System

In this section, I would briefly explain about each component of the system that was designed and developed by me, its architectural significance and what the core of each component is.

At the end, I would also summarize the flow of complete system and what is interoperability of each of the building-blocks.

Dataset:

The dataset that was provided to me was such that it contained almost 5000 records in the form of an excel sheet containing but not limited to details of courses, their respective instructors, course codes and their respective batches, students enrolled in each course and their further details. The screenshot attached under is from the sample data:



	A	B	C	D	E	F	G	H	I	J	K	L
1	SNo	STU_ID	ctor_COURS	Instructor ID	tructorLMSUs	Instructor_Nam	tructor_Departmen	structorEma	Course_IC	Course_Name	Course_Cod	STU_REC
2	1	15243	7395-13375	7395	Aslam Bazm	Aslam Bazm	SEECS	@seecs.edu.	13375	HU222 Professional Ethics(BEE-3A)	HU222	2011-NUST-SEECS-B
3	2	15189	7395-13375	7395	Aslam Bazm	Aslam Bazm	SEECS	@seecs.edu.	13375	HU222 Professional Ethics(BEE-3A)	HU222	2011-NUST-SEECS-E
4	3	16333	7395-13375	7395	Aslam Bazm	Aslam Bazm	SEECS	@seecs.edu.	13375	HU222 Professional Ethics(BEE-3A)	HU222	2011-NUST-SEECS-E
5	4	11061	7395-13375	7395	Aslam Bazm	Aslam Bazm	SEECS	@seecs.edu.	13375	HU222 Professional Ethics(BEE-3A)	HU222	2010-NUST-SEECS-E
6	5	15153	7395-13375	7395	Aslam Bazm	Aslam Bazm	SEECS	@seecs.edu.	13375	HU222 Professional Ethics(BEE-3A)	HU222	2011-NUST-SEECS-E
7	6	15192	7395-13375	7395	Aslam Bazm	Aslam Bazm	SEECS	@seecs.edu.	13375	HU222 Professional Ethics(BEE-3A)	HU222	2011-NUST-SEECS-E
8	7	16499	7395-13375	7395	Aslam Bazm	Aslam Bazm	SEECS	@seecs.edu.	13375	HU222 Professional Ethics(BEE-3A)	HU222	2011-NUST-SEECS-E
9	8	15230	7395-13375	7395	Aslam Bazm	Aslam Bazm	SEECS	@seecs.edu.	13375	HU222 Professional Ethics(BEE-3A)	HU222	2011-NUST-SEECS-E
10	9	15240	7395-13375	7395	Aslam Bazm	Aslam Bazm	SEECS	@seecs.edu.	13375	HU222 Professional Ethics(BEE-3A)	HU222	2011-NUST-SEECS-E
11	10	15182	7395-13375	7395	Aslam Bazm	Aslam Bazm	SEECS	@seecs.edu.	13375	HU222 Professional Ethics(BEE-3A)	HU222	2011-NUST-SEECS-E
12	11	15167	7395-13375	7395	Aslam Bazm	Aslam Bazm	SEECS	@seecs.edu.	13375	HU222 Professional Ethics(BEE-3A)	HU222	2011-NUST-SEECS-E
13	12	11157	7395-13375	7395	Aslam Bazm	Aslam Bazm	SEECS	@seecs.edu.	13375	HU222 Professional Ethics(BEE-3A)	HU222	2010-NUST-SEECS-E
14	13	15181	7395-13375	7395	Aslam Bazm	Aslam Bazm	SEECS	@seecs.edu.	13375	HU222 Professional Ethics(BEE-3A)	HU222	2011-NUST-SEECS-E
15	14	16181	7395-13375	7395	Aslam Bazm	Aslam Bazm	SEECS	@seecs.edu.	13375	HU222 Professional Ethics(BEE-3A)	HU222	2011-NUST-SEECS-E
16	15	15299	7395-13375	7395	Aslam Bazm	Aslam Bazm	SEECS	@seecs.edu.	13375	HU222 Professional Ethics(BEE-3A)	HU222	2011-NUST-SEECS-E
17	16	15179	7395-13375	7395	Aslam Bazm	Aslam Bazm	SEECS	@seecs.edu.	13375	HU222 Professional Ethics(BEE-3A)	HU222	2011-NUST-SEECS-E
18	17	15165	7395-13375	7395	Aslam Bazm	Aslam Bazm	SEECS	@seecs.edu.	13375	HU222 Professional Ethics(BEE-3A)	HU222	2011-NUST-SEECS-E
19	18	15170	7395-13375	7395	Aslam Bazm	Aslam Bazm	SEECS	@seecs.edu.	13375	HU222 Professional Ethics(BEE-3A)	HU222	2011-NUST-SEECS-E
20	19	15297	7395-13375	7395	Aslam Bazm	Aslam Bazm	SEECS	@seecs.edu.	13375	HU222 Professional Ethics(BEE-3A)	HU222	2011-NUST-SEECS-E
21	20	15216	7395-13375	7395	Aslam Bazm	Aslam Bazm	SEECS	@seecs.edu.	13375	HU222 Professional Ethics(BEE-3A)	HU222	2011-NUST-SEECS-E
22	21	15207	7395-13375	7395	Aslam Bazm	Aslam Bazm	SEECS	@seecs.edu.	13375	HU222 Professional Ethics(BEE-3A)	HU222	2011-NUST-SEECS-E
23	22	15266	7395-13375	7395	Aslam Bazm	Aslam Bazm	SEECS	@seecs.edu.	13375	HU222 Professional Ethics(BEE-3A)	HU222	2011-NUST-SEECS-B
24	23	11142	7395-13375	7395	Aslam Bazm	Aslam Bazm	SEECS	@seecs.edu.	13375	HU222 Professional Ethics(BEE-3A)	HU222	2010-NUST-SEECS-E
25	24	15293	7395-13375	7395	Aslam Bazm	Aslam Bazm	SEECS	@seecs.edu.	13375	HU222 Professional Ethics(BEE-3A)	HU222	2011-NUST-SEECS-B
26	25	16497	7395-13375	7395	Aslam Bazm	Aslam Bazm	SEECS	@seecs.edu.	13375	HU222 Professional Ethics(BEE-3A)	HU222	2011-NUST-SEECS-E
27	26	15501	7395-13375	7395	Aslam Bazm	Aslam Bazm	SEECS	@seecs.edu.	13375	HU222 Professional Ethics(BEE-3A)	HU222	2011-NUST-SEECS-E
28	27	16247	7395-13375	7395	Aslam Bazm	Aslam Bazm	SEECS	@seecs.edu.	13375	HU222 Professional Ethics(BEE-3A)	HU222	2011-NUST-SEECS-E
29	28	16246	7395-13375	7395	Aslam Bazm	Aslam Bazm	SEECS	@seecs.edu.	13375	HU222 Professional Ethics(BEE-3A)	HU222	2011-NUST-SEECS-E
30	29	15236	7395-13375	7395	Aslam Bazm	Aslam Bazm	SEECS	@seecs.edu.	13375	HU222 Professional Ethics(BEE-3A)	HU222	2011-NUST-SEECS-B
31	30	11228	7395-13375	7395	Aslam Bazm	Aslam Bazm	SEECS	@seecs.edu.	13375	HU222 Professional Ethics(BEE-3A)	HU222	2010-NUST-SEECS-E
32	31	15141	7395-13375	7395	Aslam Bazm	Aslam Bazm	SEECS	@seecs.edu.	13375	HU222 Professional Ethics(BEE-3A)	HU222	2011-NUST-SEECS-E

My further action taken on this sample data was to perform certain tweaking such that my program would find it possible to read, parse and write this data into database.

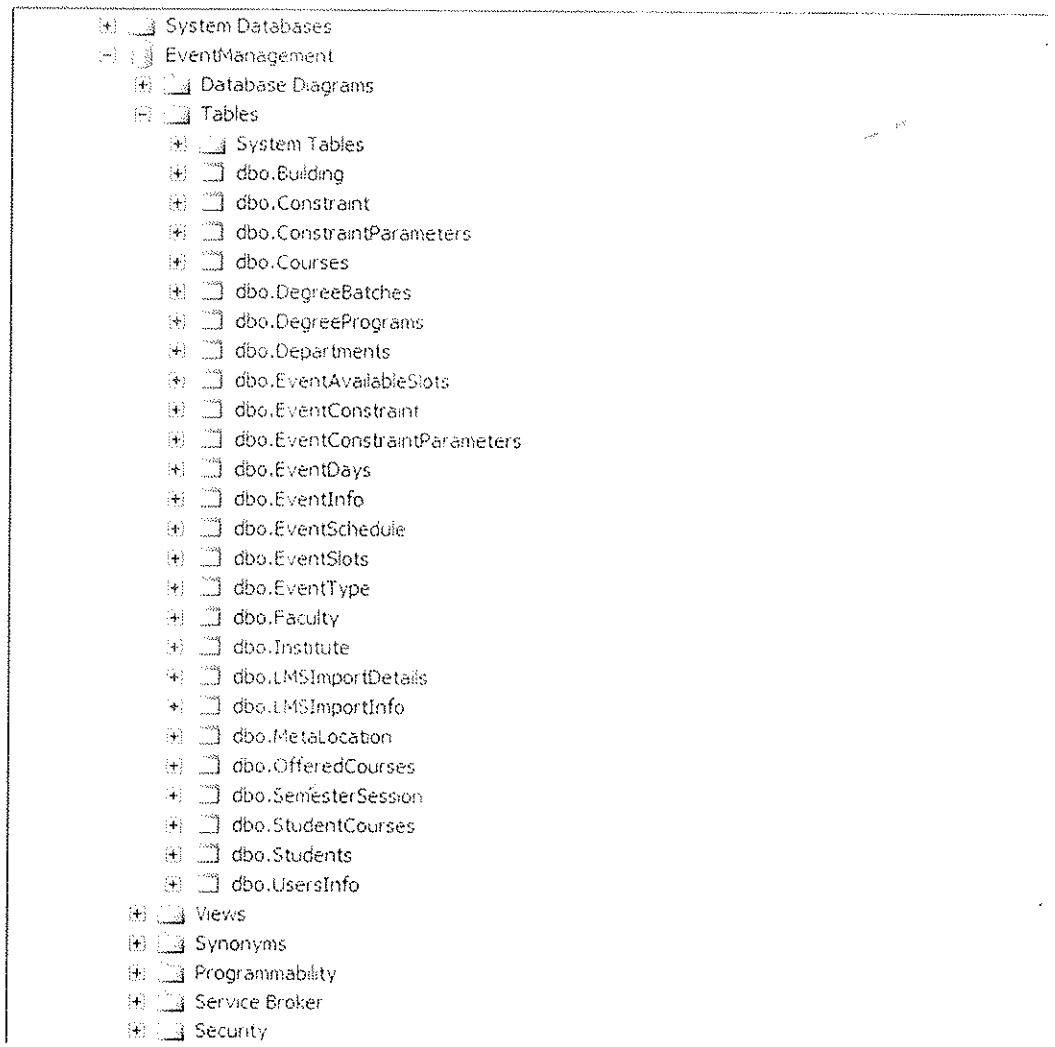
Speaking of parsing, there was one column that needed further editing e.g. Course_Name was in the format which was missing any special character between Degree and its respective batch e.g. BEE3 and not in the format that would allow my system to make this differentiation of degree and batches. Therefore, the first operation that I performed was to insert a special character in between the degree and batch which would now look like BEE-3

Secondly, since currently the input file contained the data of students and courses regardless of any department such as in SEECS are Department of Computing (DOC) and Department of Electrical Engineering (EE). This part was unavoidable as the database schema would have a special table for department. Therefore, I inserted an additional column in the sheet named "Department" with its value set to SEECS for each record.

This would allow this system to generate the output files without having to worry about certain department that Students were a part of or a course that a batch was enrolled in.

Database:

The database provided to me was also very structured with the proper schema and its constraints well-defined and the maximum effort that is generally required to design a schema-based db was not required. This is also because SEECS was following a specific db architecture to be followed in the subsequent releases of this product which would incorporate seating plan functionality. The detail of what tables were contained in that database are in the figure as under:



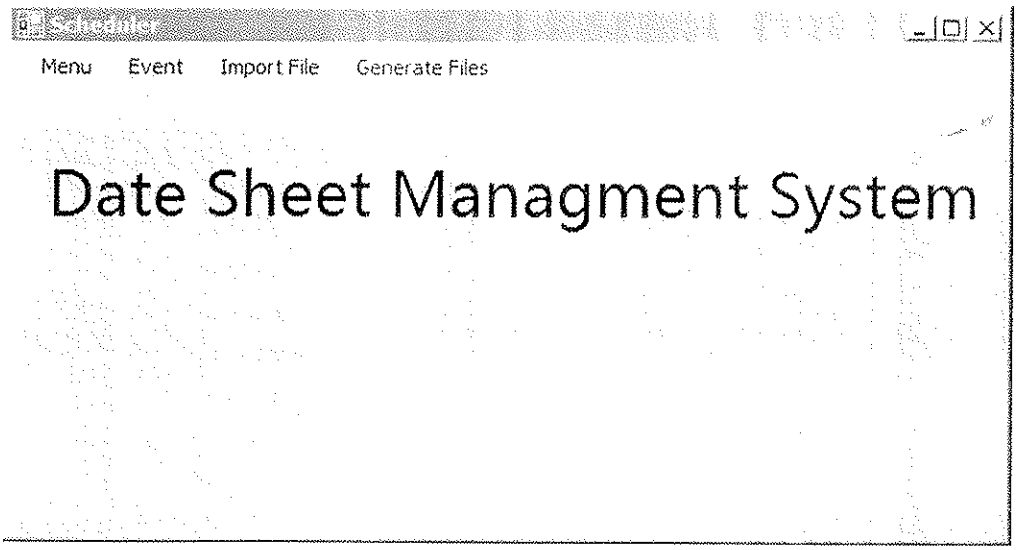
I was supposed to use all of the tables except for the following:

- UsersInfo
- Building
- MetaLocation

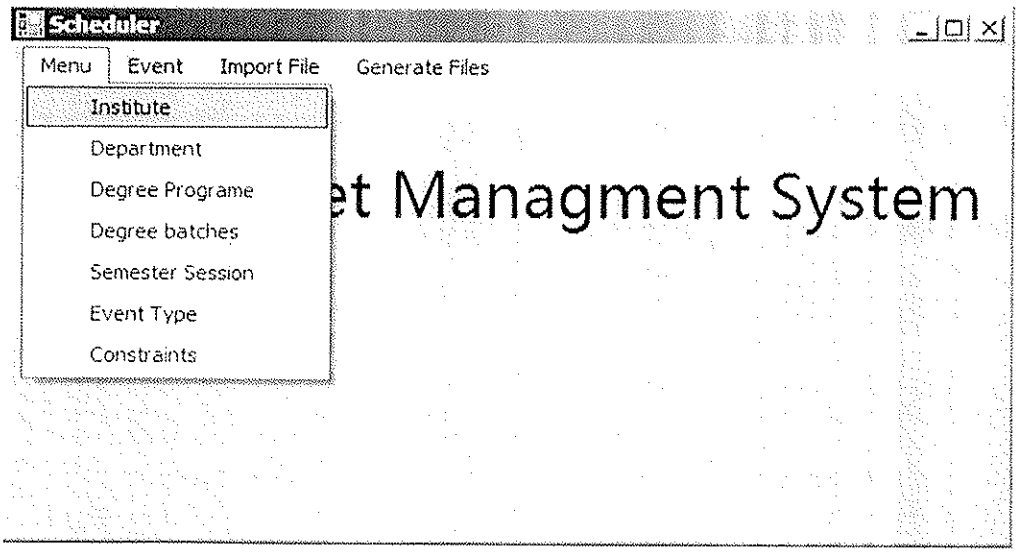
This because as mentioned above as well, these tables would be part of future product enhancements such as extending it to provide seating plan functionality as well.

Software System:

The below figure illustrates how the main GUI of my developed system looks like:

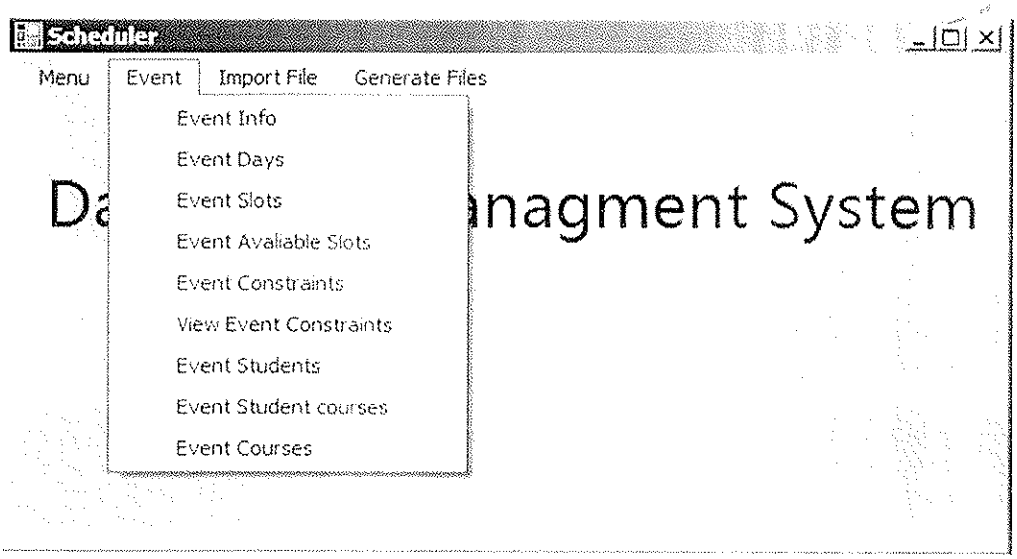


Menu button provides the starting functionality of the system as it allows administrator of the system to enter the details of Institute, Department, Degree Programs, Degree Batches, Semester Sessions and Type of events (such as OHT1, OHT2 & ESE) to be entered manually. All of this input when provided is stored in backend database in its respective table.

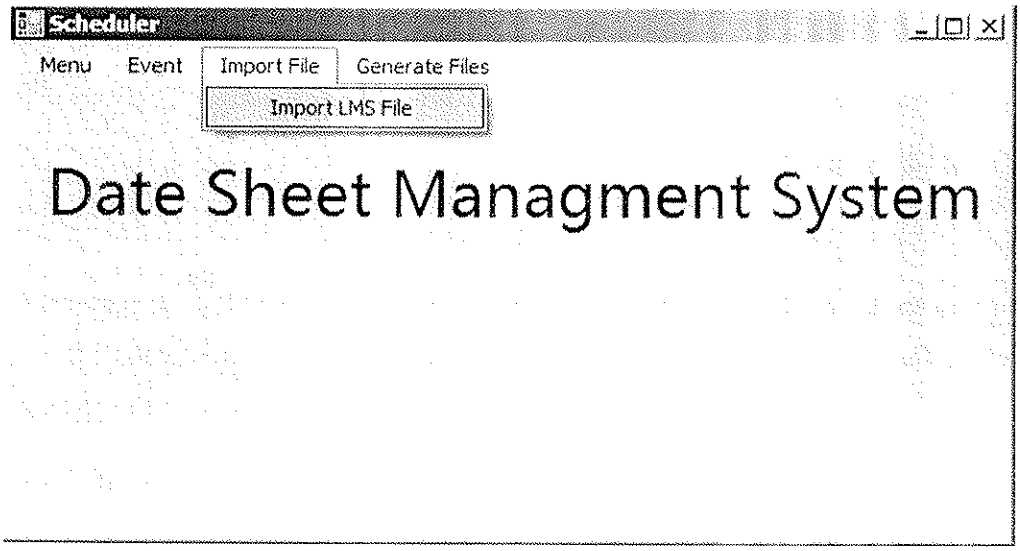


The **Event** screen allows user to define a certain event and its complete details such as the no. of days that event would be covered in and how many slots does each day could have at maximum, moreover, it also allows users to add different constraints and parameters against those constraints which would then be associated to the events going to be scheduled. The event students, courses offered

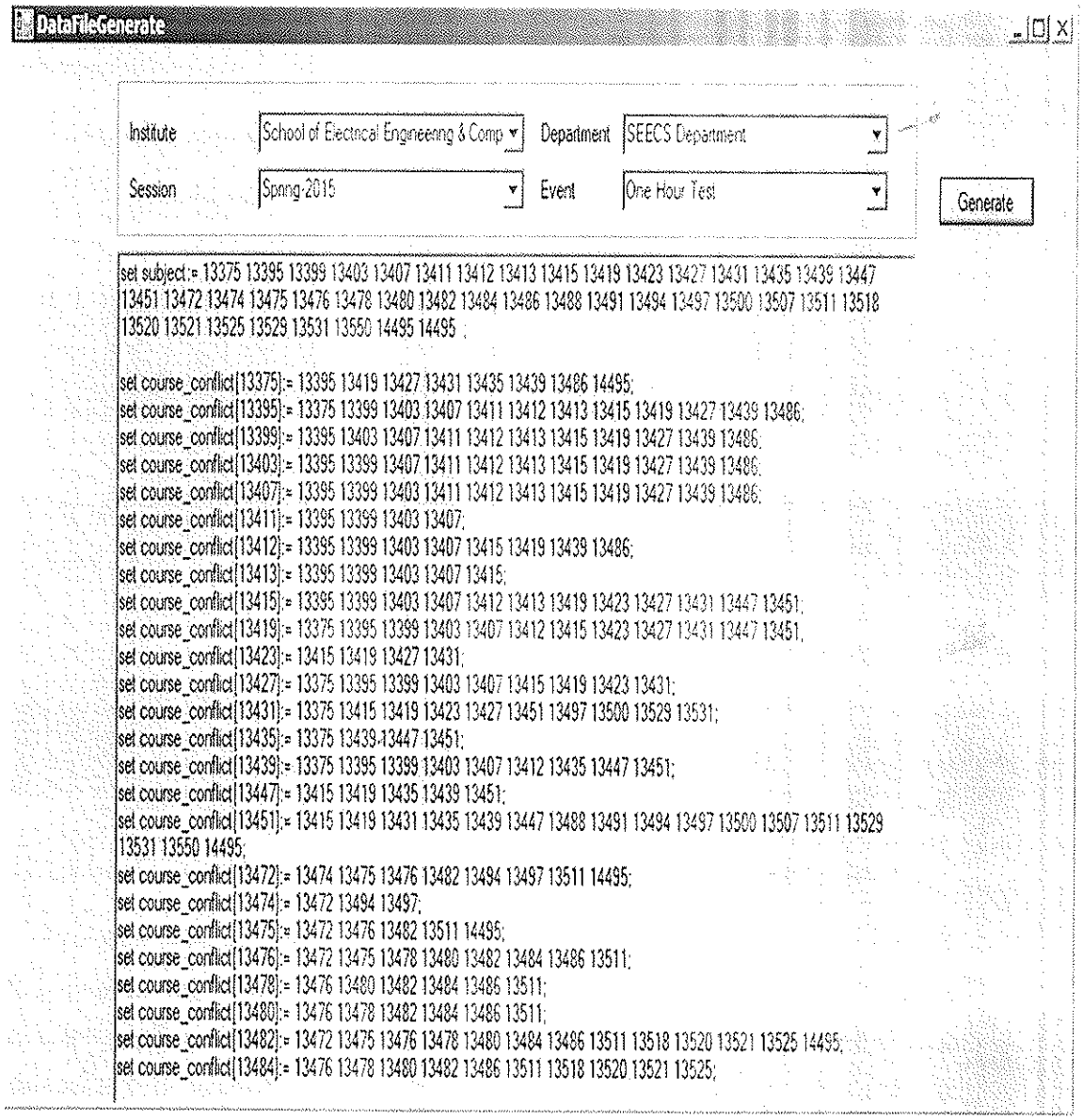
and faculty information is extracted from excel sheet and then inserted into the tables that matches it.



The **Import File** screen is the second last GUI that actually takes excel sheet file as an input and scatters the data after performing a series of events to the rest of the tables. A sample figure describing how import file screen looks like is shown as under:



Last but not the least is the final GUI **Generate Files** which is the final process of generating the files i.e. .dat and .mod file the figure below shows it:

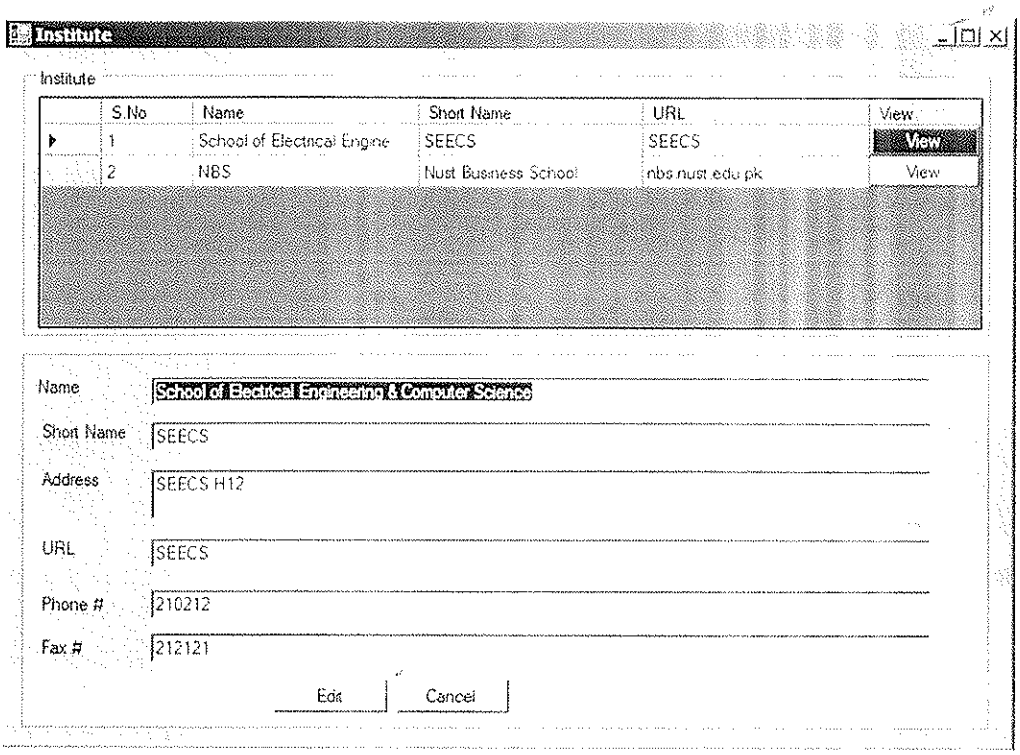


This image is further navigation into Generate Files and actually is screen shot of how the content of .dat file look like. Note that besides displaying the data in the panel of interface a backend file is also generated which is the key.

How the whole process works:

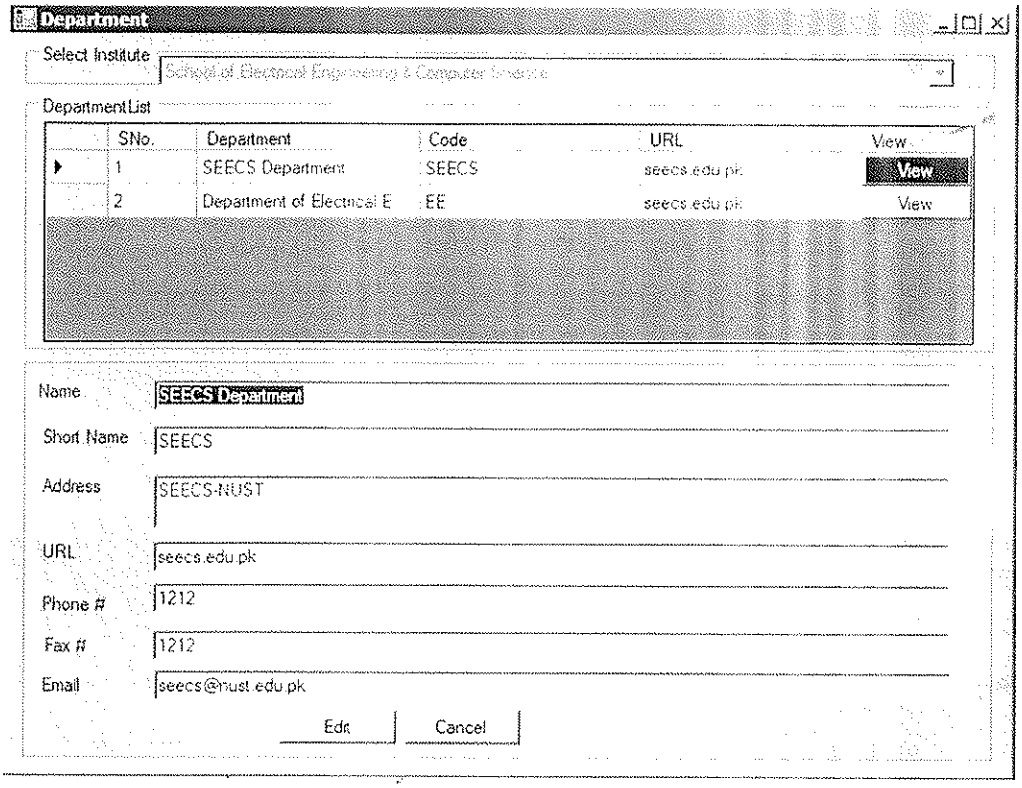
In this section, I will describe what steps my system would go through to generate files.

- In first step, user enters the details of an institute e.g. SEECS, its URL, contact information and email address as show in the figure below:



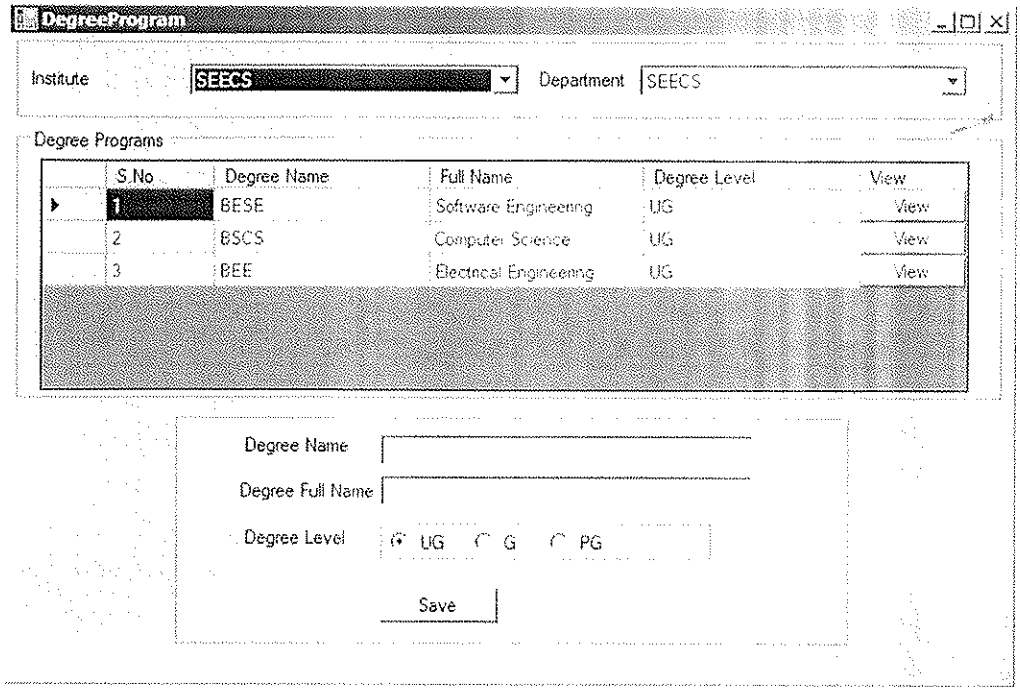
The entered information gets written directly into Institute tables in the database

- User then goes to department GUI and enters department information, since in this case as the excel file data does provide consolidated information all the departments e.g. DOC and EE it is therefore required to enter SEECS as a department in generic or the same value that was entered in excel sheet in Department column. Below figure illustrates this example:

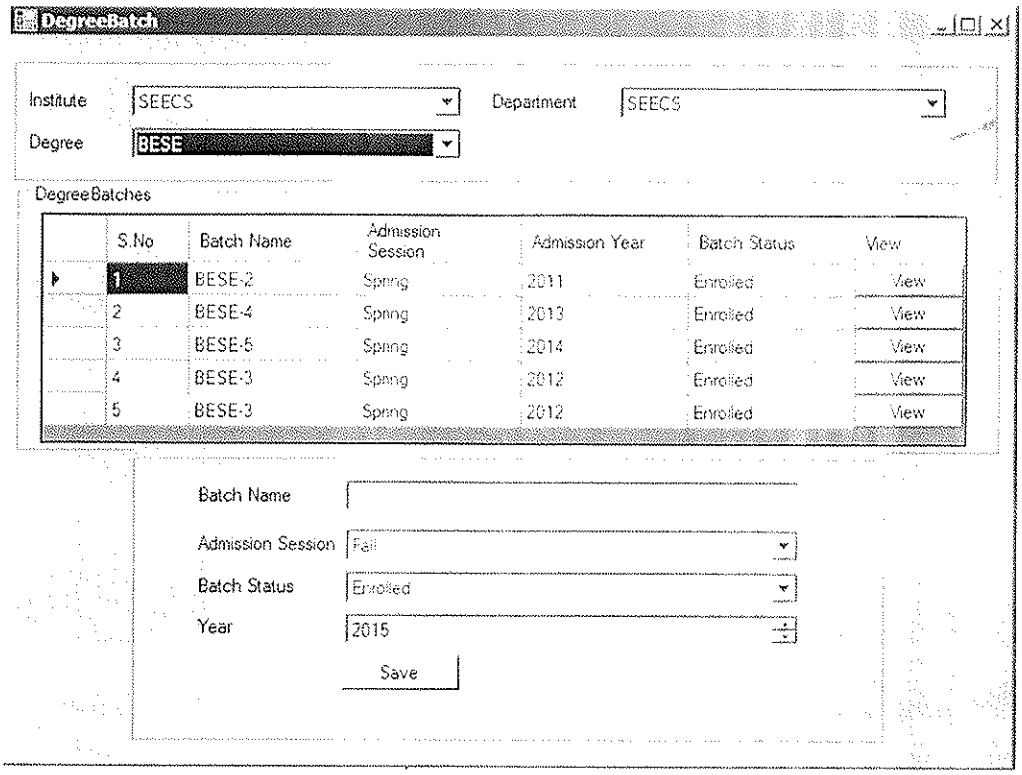


Same as with the institute this information also gets written into database but into department table.

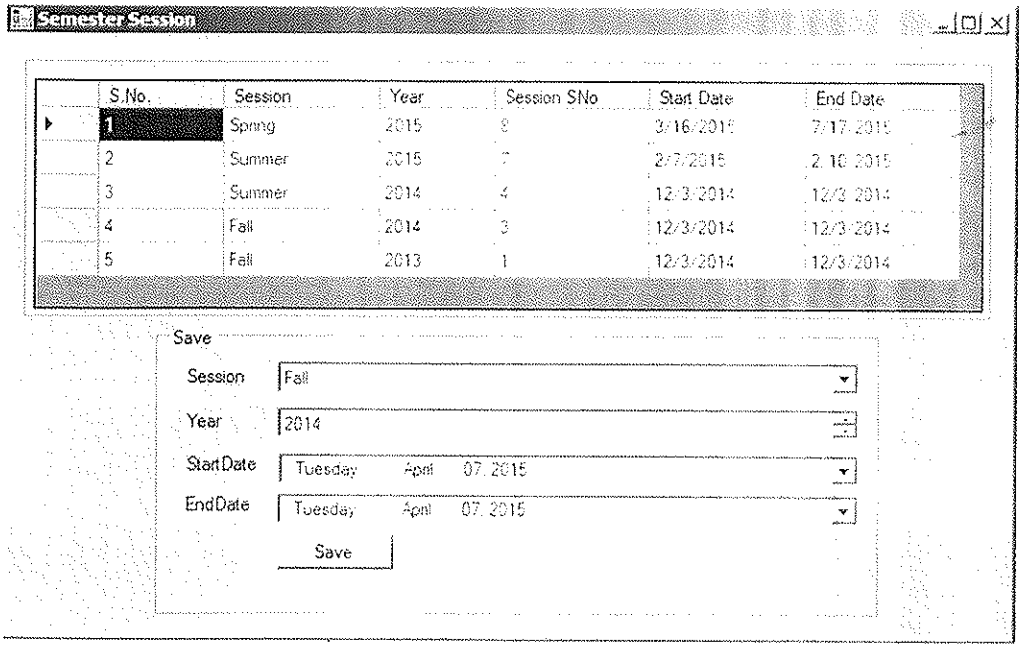
- Then comes the turns to enter all the degree programs being offered in the above institute and its department in our case would be SEECS both as department and institute as well.



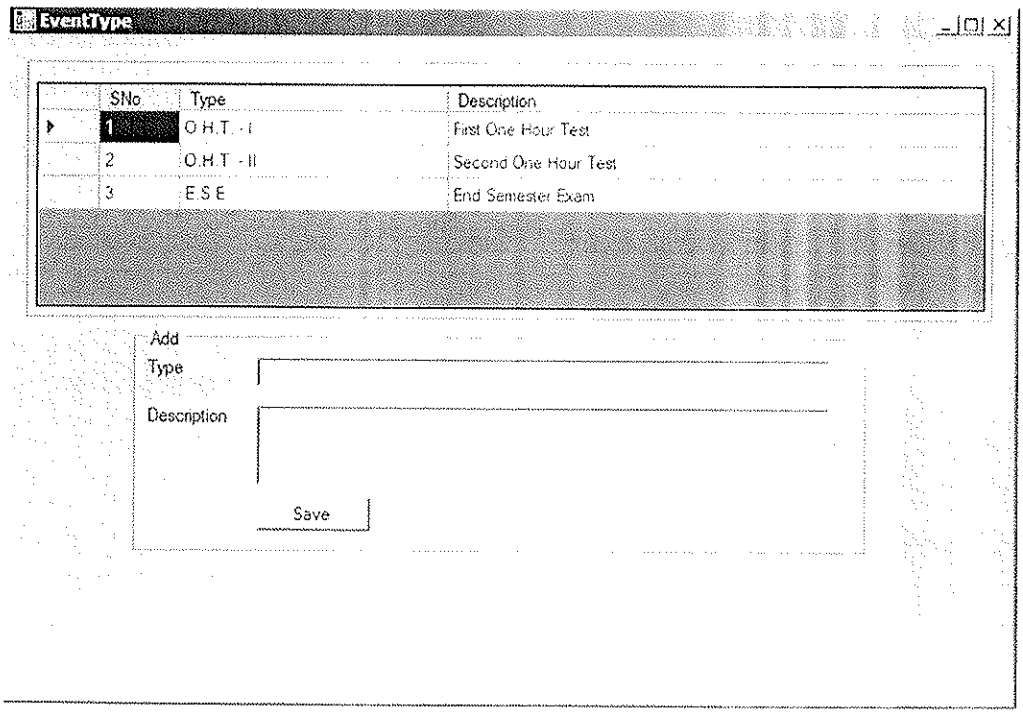
- The next information to be entered would be all the degree batches pertaining to the degree programs, it is required for the user to enter the batches against their corresponding degree programs otherwise the output should get affected:



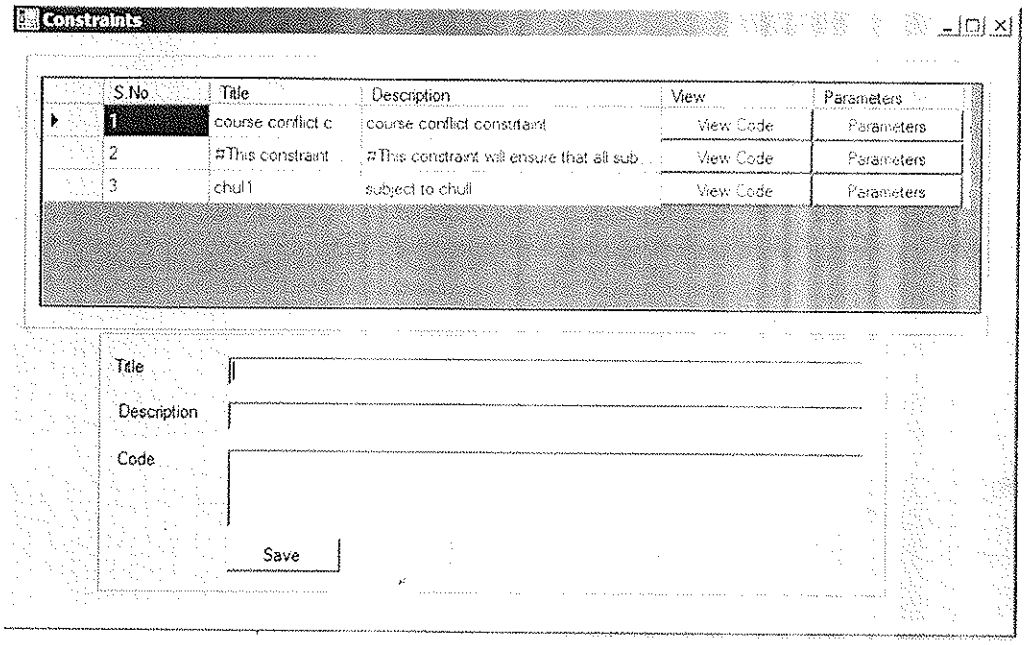
- At this stage semester session info is to be provided to system, since it is well known how many semester sessions are being followed at UG level which is mostly 3 sessions in a year including summer session, note that the below information is more than 3 sessions which means they might be for different years or to be more precise this is old data which needs to get removed:



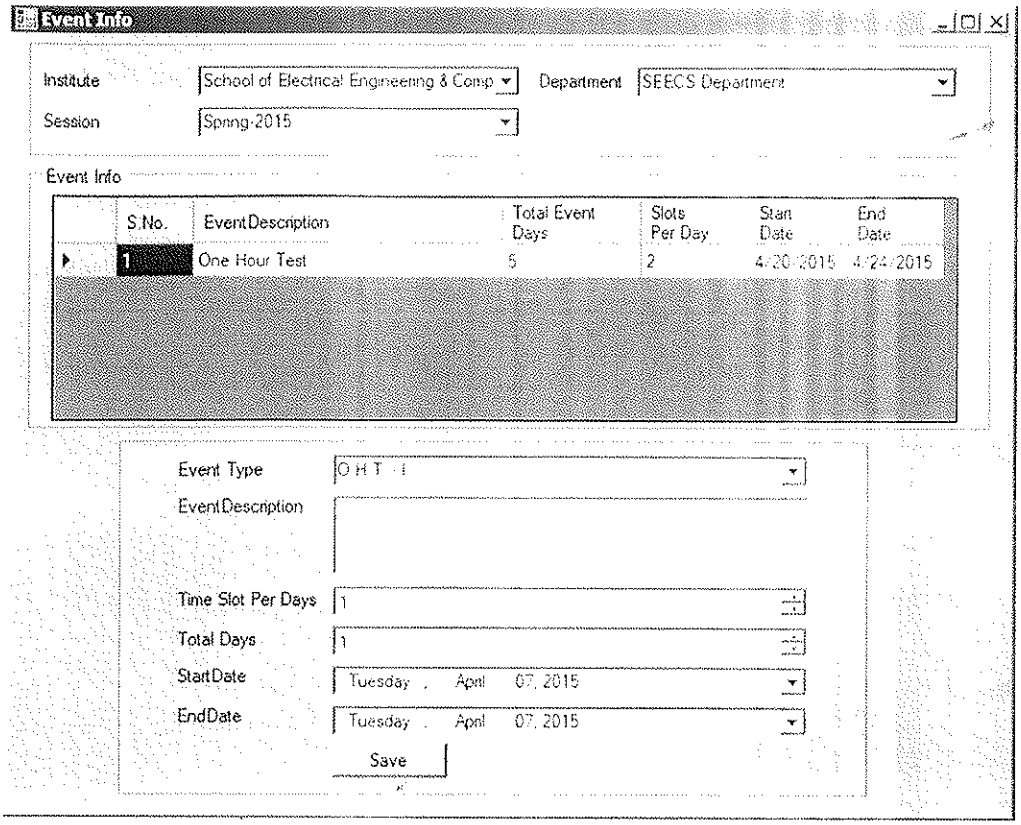
- It then follows types of events for which date sheet is required to be generated. As matter of fact, SEECS has only three type of events ranging from two One Hour Tests (OHTs) to one End Semester Exam (ESE):



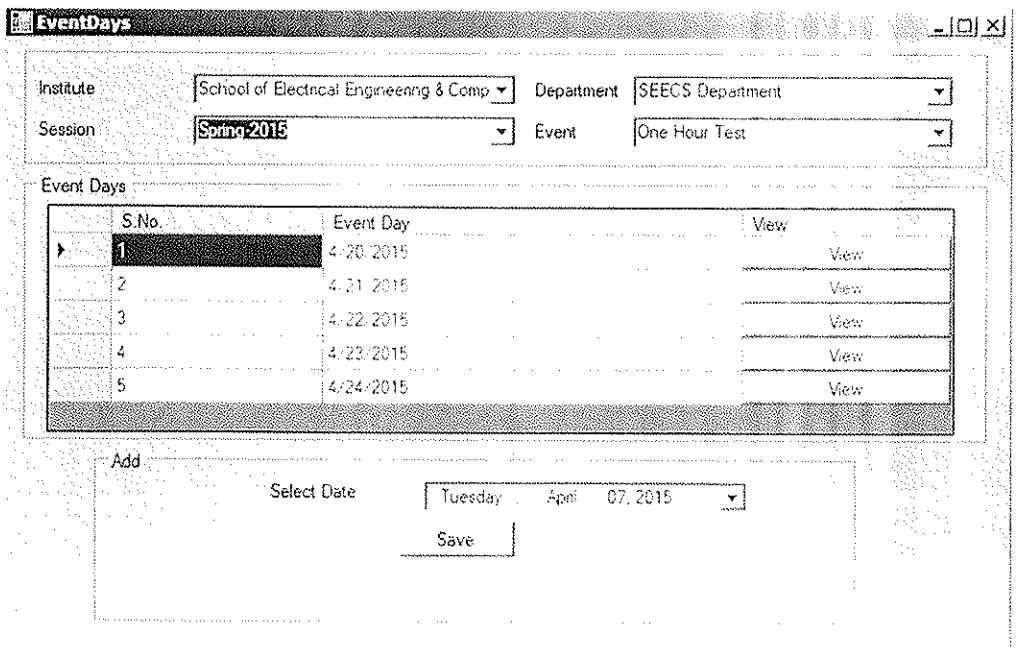
- The below screen in the figure shows different constraints for event which need to be defined. These constraints would later become part of model file, hence required to not make any human error as possible. To provide flexibility and error control the data is editable:



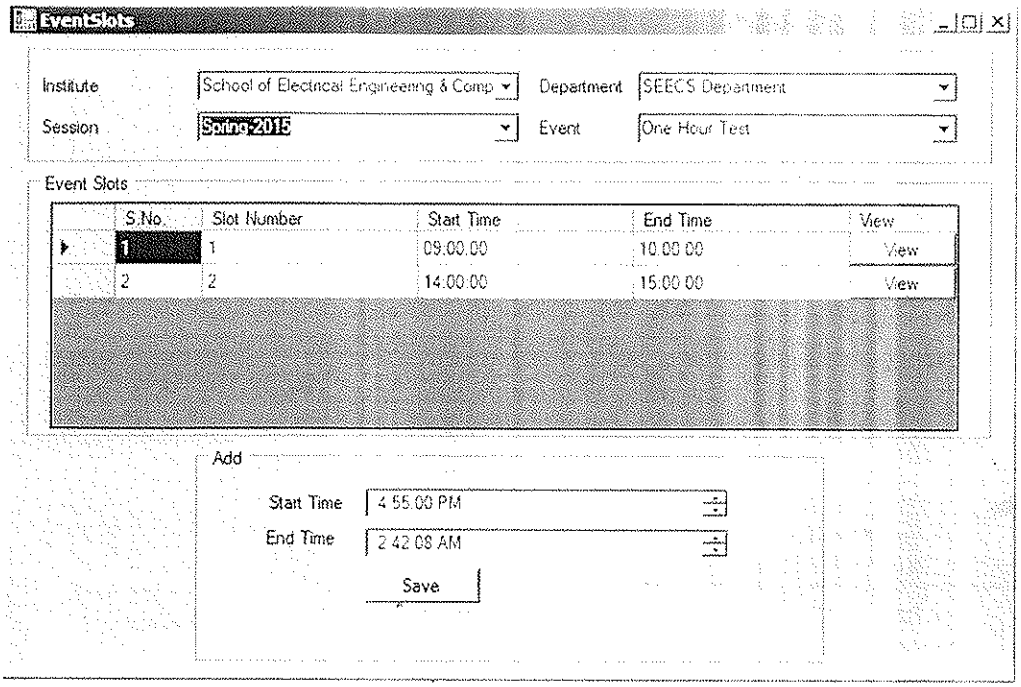
- After comes the event information details such as how long would this event e.g. OHT would remain i.e. no. of day and no. of slots available for each day:



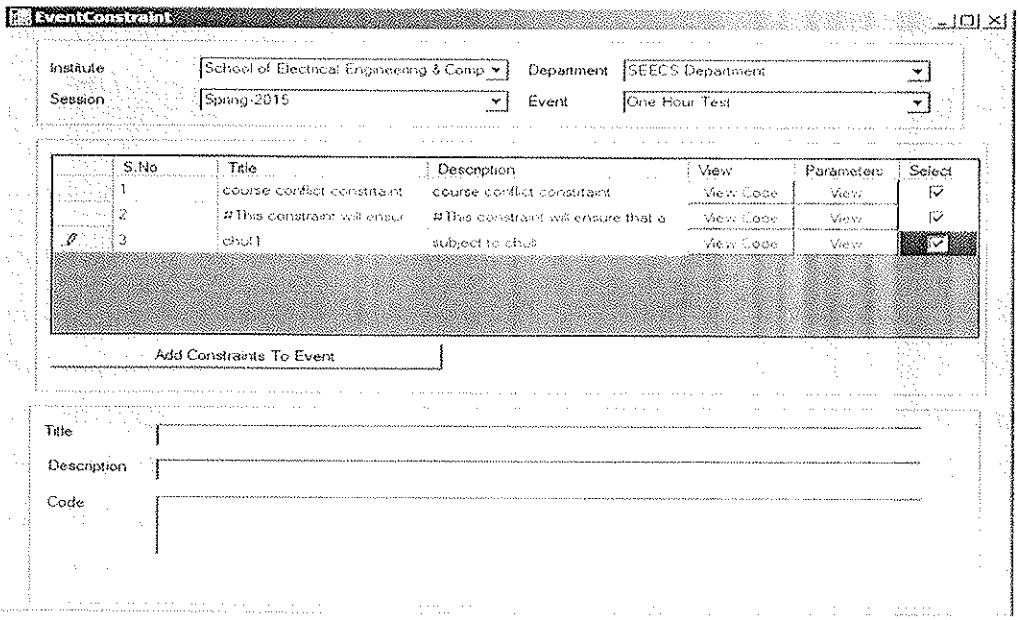
- User now has to enter the exact calendar days by choosing from the calendar option. Note that this information must reflect the Start and End date chosen in the above screen:



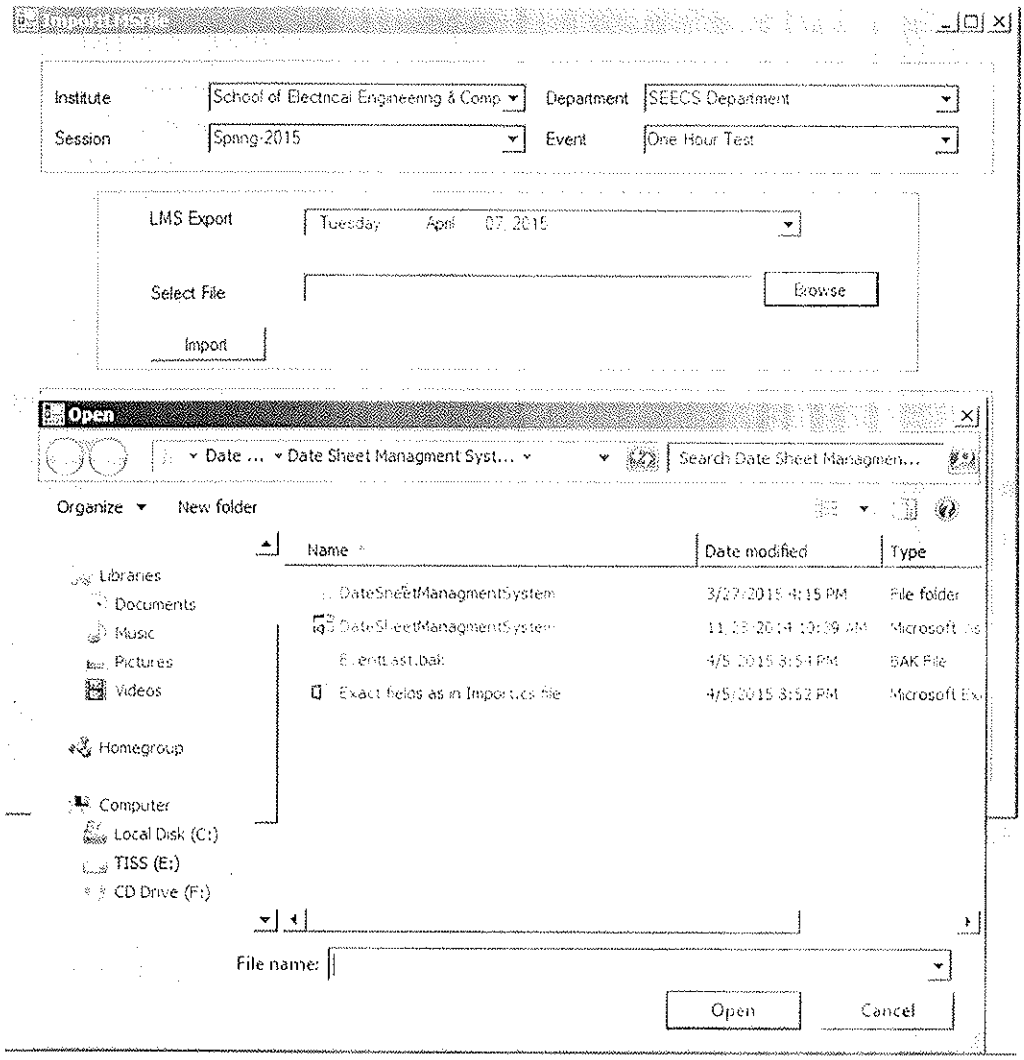
- Just as the days have been selected, slots and their respective time must also be configured for this particular event going to be scheduled for the selected department in the given institute:



- The event constraints defined in the constraints GUI must now be added to the scheduled event by checking the **Select** checkbox, then click **Add Constraints to Event**:

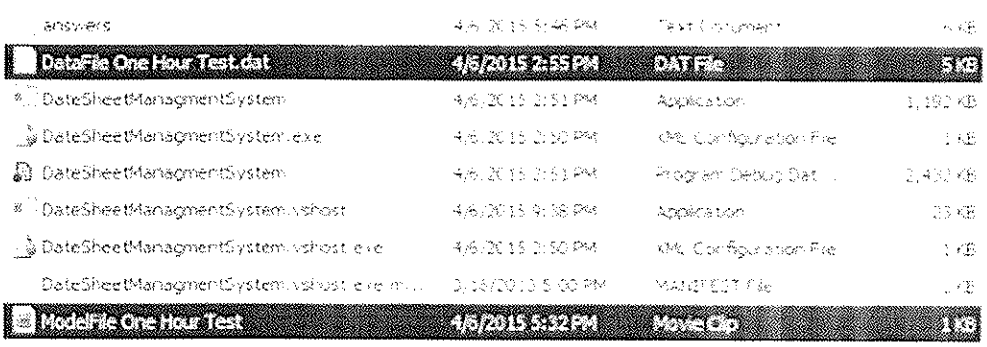


- This is all input related to db which was supposed to be required by user, its time to import the excel file:



Once the user has clicked import button, then depending upon the amount of records the time taken by the system may take up to couple of minutes, at the backend the data is read split into different tables such as faculty, students, unique courses being offered and students enrolled in each course which is an obvious reason for the extended time taken at this step, user is expected to wait the successful importation file, my test file with sample size 5000-6000 would take no more than a couple of minutes.

- This is the final step where use has to generate both model and data file one by one. The file contents are also displayed in the window but to see the actual file please go to debug folder inside project directory:



Challenges faced:

One of the major challenge faced during the development and testing phases was the dynamic requirements and continuously changing database which had various implications on the source code obviously and keeping up to date with the latest database fields.

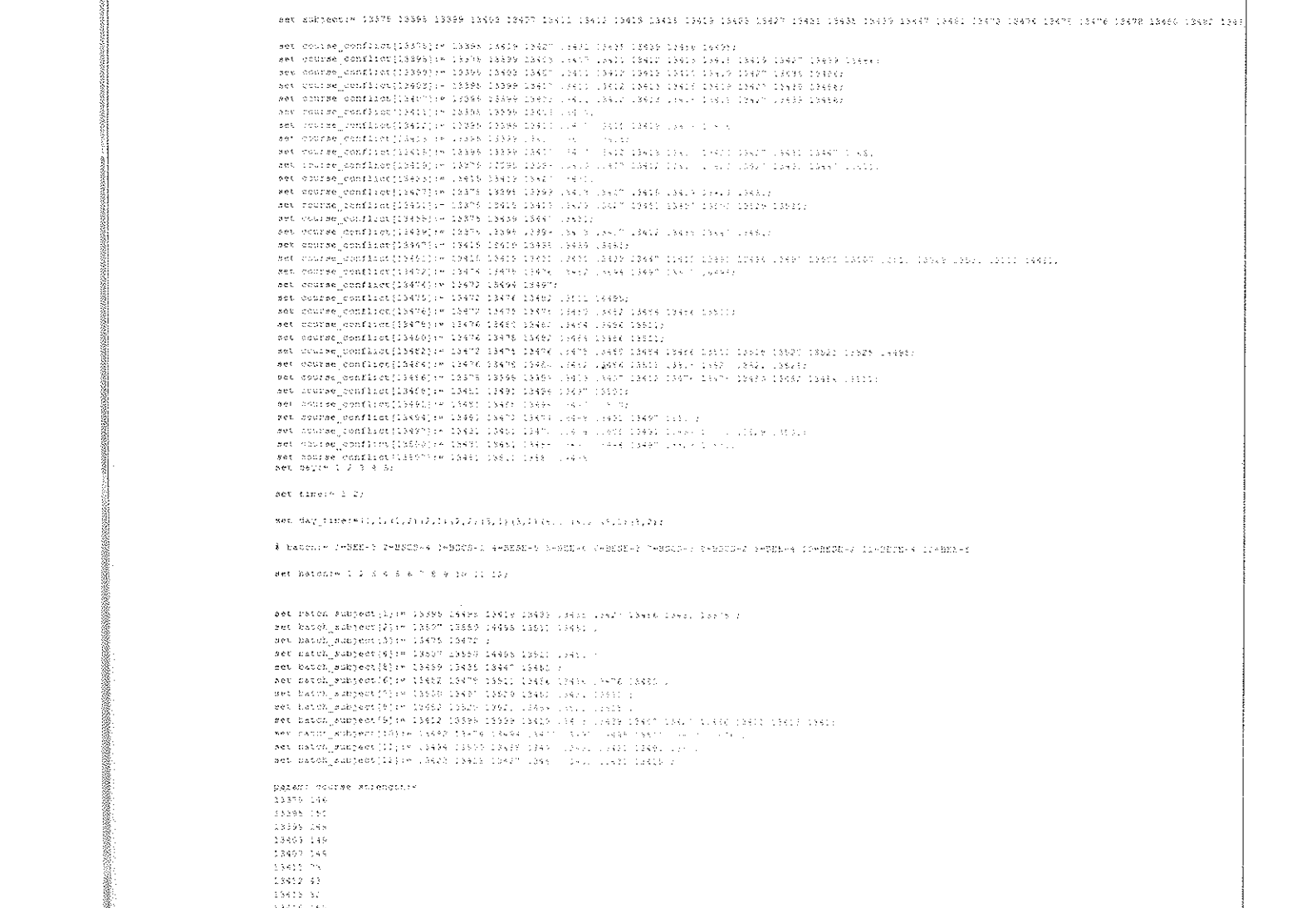
Secondly, there were various version of LMS file being given to me, therefore, so finalize the file that would contain maximum data and that too accurate was a painful task. Most of the time, one file would take days to meet the requirements of the system since major algorithm was already written hence, changing the code seemed to be less painful than changing the file itself. I have shared with the format of the excel sheet and what column information is mandatory for the system to correctly parse it and to perform successful operations towards generation of both output files.

Result

The end result consists of two files one is the data file while the other being model file. As far as the data file is concerned it contains the information majorly related to optimized and unduplicated results of courses, their strengths, their conflicts,

schedule of the event for which the date sheet is being generated and the information if batches and what course is being taken by each batch.

The model file mainly consist of constraints defined at the time of GUI wizard which is majorly the data being instructed to AMPL based software so that it can perform its step accurately and with the desired information given to it. The sample of data file is pasted below:



These images above do not provide the complete information in the data file, it is just to show the type of data that is being written into the file.

Chapter 4

Conclusion

This chapter describes the learning outcome of this project.

There are many learning outcomes of this project including good understanding of database concepts, hands-on experience with MSSQL database tools as well as its detailed configuration procedure. I also established strong grasp over SEECS date sheet generation process. Practice of algorithms using in-depth understanding of C# programming language. Requirements Engineering process was also a fruitful effort.

The summary of things addressed in this report include:

- Problems with current date sheet generation process
- Review of how the process works in details
- The proposed solution and the technologies used for this project
- Working of my project
- Different tips and tricks to avoid being ending up into undesired results
- Learning outcome

Chapter 6

References

[1] Microsoft Documentation on MSSQL

[2] Microsoft & other forums

[3] Stackoverflow

[4] DotNet professional forums

[5] msdn https://msdn.microsoft.com

[6] Previous date sheet scheduling process documentation

[7] dotNet forums http://forums.asp.net