Saudi Arabian Airlines
Training Department
Houston, Texas

STUDENT PAYROLL SYSTEM
(SPSYSTEM)

Programmer's Guide

MASTER GRADUATE PROJECT

Prepared by
FAHAD M. S. KAYAL
TEXAS A&M UNIVERSITY AT CORPUS CHRISTI
November 15, 1993

COMMITTEE MEMBERS

Mr. William Allen Bush
Chairman

Dr. Roy S. Ellzey
Member

Mr. David Perkins
Member
LOGO SCREEN

SAUDIA
SAUDI ARABIAN AIRLINES
Training Department
Houston, Texas       U.S.A.
Student Payroll System
SPSystem v1.0 by Fahad Kayal
Copyright 1993
# TABLE OF CONTENTS

ACKNOWLEDGEMENTS .................................................. 1

ABSTRACT ................................................................. 2

LIST OF FIGURES ...................................................... 3

INTRODUCTION .......................................................... 4

PROJECT ENVIRONMENT ............................................. 6

LOGICAL DESIGN ....................................................... 7

DESIGN IMPLEMENTATION .......................................... 11

SUMMARY & CONCLUSIONS ......................................... 28

REFERENCES ............................................................ 29

APPENDIX I 
DATA MODEL ........................................................ 30

APPENDIX II 
DATA DICTIONARY ................................................... 44

APPENDIX III 
MAIN & SUB-MENUS ............................................... 50

APPENDIX IV 
USER'S INPUT FORMS ............................................. 66

APPENDIX V 
REPORT EXAMPLES ................................................ 83

APPENDIX VI 
LISTING REPORT EXAMPLES ....................................... 91

APPENDIX VII 
PROCEDURE ........................................................ 100

APPENDIX VIII 
REPORT FORMAT FILES ........................................... 224

APPENDIX IX 
QUERY-BY-EXAMPLE FILES ....................................... 292

APPENDIX X 
PROGRAMS NAMES AND USAGE .................................. 302

INDEX ................................................................. 307
ACKNOWLEDGEMENTS

The Author wishes to thank (Saudia) Saudi Arabian Airlines Training department, Houston, TX U.S.A especially Dr. Paul Rusk and Mrs. Bobby Rabackoff for providing the time and support of this project. Thanks also to my graduate project chairman Prof. William Allen Bush for his expert guidance and helpful suggestions. Special thanks are also given to my family for all their love, understanding, support, and encouragement.
ABSTRACT

This project is the design and implementation of a student financial monitoring system for the Training Department of Saudi Arabian Airlines. The system is a menu-driven database management system that creates and maintains a database to provide student payroll information including monthly allowance, book and tool allowance, clothing allowance, tuition payment, medical expenses, and ticketing expenses. The system calculates the monthly payment due for each student and provides a monthly payroll report to the finance department for issuing monthly pay checks.
## LIST OF FIGURES

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>System Overview, module flow of control</td>
<td>12</td>
</tr>
<tr>
<td>2</td>
<td>System Main module Flowchart</td>
<td>14</td>
</tr>
<tr>
<td>3</td>
<td>Edit Menu Flowchart</td>
<td>16</td>
</tr>
<tr>
<td>4</td>
<td>Post Menu Flowchart</td>
<td>19</td>
</tr>
<tr>
<td>5</td>
<td>Report Menu Flowchart</td>
<td>22</td>
</tr>
<tr>
<td>6</td>
<td>Listing Menu Flowchart</td>
<td>25</td>
</tr>
<tr>
<td>7</td>
<td>Maintenance Menu Flowchart</td>
<td>27</td>
</tr>
<tr>
<td>8</td>
<td>Diagram of SPSYSTEM DBMS Relationship</td>
<td>30.A</td>
</tr>
</tbody>
</table>
INTRODUCTION

The Training Department of Saudi Arabian Airlines was relocated to Houston, Texas and opened on November 1, 1986. Prior to that time, it was located in Kansas City, Missouri where it was originally started by Mr. Hussain Bondagji in 1980 to provide the support previously contracted to an outside company called AmidEast.

The prime function of this department is to monitor the academic performance of students sponsored by Saudi Arabian Airlines who are attending various academic and technical programs in the U.S.A. Upon completion of their studies, the students return to Saudi Arabia where they are placed into positions relevant to their skills. The objective of this training is that each student, through his respective program, be a knowledgeable, productive employee who will be capable of fulfilling his duties in support of the airline company.

A second, and equally important, function of the training department is to provide financial support to the students, as outlined by the company. This support includes payment of tuition fees, monthly living allowances, book and clothing allowances, and payment of medical expenses. The objective of the financial support provisions is to allow the student the opportunity to focus completely upon his educational training.

Other functions of this department include support for several Training departments in Jeddah, Saudi Arabia, including registration of employees for short courses and seminars in the U.S.A., and the purchase and shipment of various training materials for use in Jeddah, Saudi Arabia. In addition, this department researches various subjects and provides information, as requested.

The recent growth of the training department of Saudi Arabian Airlines Houston, Texas has increased management's awareness of the need for an automated system to monitor the potential growth of the number of students and their monthly payroll. Presently the department consists of one manager, two Academic and Technical consultants, one coordinator, and two secretaries. The status of each student trainee is kept in a confidential file; therefore, the system enforces security protection. The project database system solves many of the time dependent reports required by management.

To help the management plan their work, a student payroll report is generated for the next month showing monthly allowance, book and tool allowance, clothing allowance, tuition payment, medical expenses, and ticketing expenses. The system calculates the monthly payment due for each student and provide a monthly payroll report to the finance department for issuing monthly pay checks.

Management does not have an easy method to produce the monthly payroll report; therefore, the implementation and design of this database project has provided a tool to assist in this process.
One of the goals of this project is to follow the procedures used at the Training Department as closely as possible so that the users will feel comfortable with the resulting system. Forms that are designed with the software resemble closely those that were used in the past with necessary additions and changes having been made with the approval of the users of the system.

Since some of the employees at the Training Department have very little experience using a computer, another goal was to ensure that these individuals experience a pleasant change and improvement in their daily work routine with the use of the resulting system.

The final phase of the project is to provide training for the users to familiarize them with the workings of the system so that they will know how to protect their data better and if changes are required in the future, the users will be knowledgeable enough to make the changes themselves.
PROJECT ENVIRONMENT

This database system was created for the Saudi Arabian Airlines Training Department, Houston, Texas. The system was implemented on DEL 486DX IBM compatible computer. The DEL 486DX is equipped with 640K contentional memory and 4MB extended memory, a math co-processor, a 340MB hard drive, a 1.44MB 3-1/2 disk drive, 1.2 MB 5-1/4 disk drive and VGA color monitor. The DEL 486DX runs under the MS-DOS 5.0 operating system.

The entire application including procedures and reports was written using Borland's Database IV version 2.0 relational database program. Dbase IV runs on systems that are a 286 system or higher, with at least 2 MB of extended memory running DOS 3.3 or 5.0 and higher, a CGA, EGA, or VGA color monitor, at least 30 buffers and 99 files and shell environment size of 1024 bytes. Database IV 2.0 uses virtual memory manager(VMM) that lets Dbase access up to 16 MB of extended memory and to create a swap file when we do not have much memory on our system. Dbase IV 2.0 is designed to take advantage of extended and expanded memory that is available in 286 and higher machines. A disk buffering management system that improves how Dbase uses the disks. To install Database, at least 10MB of free space is required on the system hard drive.
LOGICAL DESIGN

The principal entity classes in the Student Payroll System DBMS are: students, academic, family, Ramadan, schools, specialty, training, vendor, transact, checkreg, and xpenditr. A one-to-many relationship exists between an student and the student activities with which the student is associated. Student activities are defined as academic, family, Ramadan, schools, specialty, training, vendor, and transact. Academic refers to the student current academic performance. Family refers to the student family status, while Ramadan refers to the student bonus amount. Checkreg has a one-to-many relationship, while xpenditr has a many-to-many relationship with the student. The entity class repmemo is reference table is used for internal purposes to store up to ten lines of comments that can be printed at the end of the monthly expenditure report. To keep the database application manageable and normalized the system was designed so that each entity class has its own table.

ACADEMIC DATABASE:

The ACADEMIC table is keyed by ACADEMIC_C (Academic Code) which is defined as the primary key. This database is designed to store the codes and dollar amounts assigned to each code. These codes (attributes) are then assigned to a student for future reference. For Example: - Student 101 has an academic code of "B". Last year - the school paid an academic bonus of $225.00 for all students with this code. This year - the training department decides to pay each "B" student a bonus of $300.00. Instead of having to modify each and every database record for all students assigned a "B" code - the user simply change the dollar amount assigned "B" - and all reports that reference this code now reference the updated (and correct) amount.

FAMILY DATABASE:

The FAMILY table is keyed by FAMILY_C (Family Code) which is defined as the primary key. The attributes of this database describe and stores the codes (M, M1, S, etc) for each family type and the dollar amount for that code. The codes and amounts assigned in this database are used as a method to pay living expenses for a student (and family). As with the Academic Database - the code and dollars are stored here instead of with each student record.

RAMADAN DATABASE:

The RAMADAN table is keyed by RAMADAN_C (Ramadan Code) which is defined as the primary key. The attributes of this database describe and stores the codes (1, 0, etc) for each type of bonus and the dollar amount for that code. As with the Academic Database - the code and dollars are stored here instead of with each student record.
SCHOOLS DATABASE:

The SCHOOLS table is keyed by SCHOOL_C (School Code) which is defined as the primary key. The attributes of this database describe and stores the code, Short Name (Abbreviation), City, State, and two (2) lines of description for each school used in the system. As with the ACADEMIC database - the schools are defined in this database so the system can reference each school and extract the current and correct information.

SPECIALTY DATABASE:

The SPECIALTY table is keyed by SPECIALTY_C (Specialty Code) which is defined as the primary key. The attributes of this database describe and stores the code (or abbreviation) and the name (description) for each different type of specialties used in the system. The same principal as with the dollar amount is used with the SPECIALTY database - except code names are retrieved instead of dollar amounts whenever referenced.

TRAINING DATABASE:

The TRAINING table is keyed by TRAINING_C (Training Code) which is defined as the primary key. The attributes of this database describe and stores the code (or abbreviation) and the name (description) for each different types of training used in the system. Training are linked by training code (TRAINING_C) to the student table. The same principal as with the dollar amount databases is used with the TRAINING database - except code names are retrieved instead of dollar amounts whenever referenced.

VENDOR DATABASE:

The VENDOR table is keyed by VENDOR (Vendor Code) which is defined as the primary key. The attributes of this database describe and stores the vendor code(Vendor) and the vendor name (Vendorname) for each VENDOR used in the system. Vendor are linked by vendor code(Vendor) to the xpendituir table. The same principal as with the dollar amount databases is used with the VENDOR database - except vendor names are retrieved instead of dollar amounts whenever referenced.

TRANSACTION DATABASE:

The TRANSACT table is keyed by TRANSACODE (Transaction Code) which is defined as the primary key. The attributes of this database describe and stores the Transaction Code(Transacode), Name(Transaname), AC_CODE (Account Code), CC_CODE (Cost Center Code), and the LOC_CODE (Location Code) for each different types of Transaction used in the system. The transaction code(Transacode) is linked to student table. The most important item in this table is the logical field (AUTOPOST) flag. By turning "ON" (setting to TRUE) this flag for either MNTH, ACDM, BOOK, CLOT transaction codes - the user can "automatically post" the expenditure records for all students in the database for these codes, for a given expense period.
XPENDITR DATABASE:

The XPENDITR table maintains information about each student's current expenditure. The attributes of this database describe and stores the Transaction Code (Transacode), Vendor code (Vendor), AC_CODE (Account Code), CC_CODE (Cost Center Code), LOC_CODE (Location Code), Student code (Studencode), Expense Period (Expnsperiod), Entry date (Entry_Date), invoice amount (Invoic_amt), invoice name (Invoicename), invoice date (Invoicdate), invoice Number (Invoicenum), and Reimbursable expense (Reimbursbl) for each student expenditure used in the system. The logical field (Reimbursbl) flag. By turning "ON" (setting to TRUE) this flag - the user can reimbursable the student(s) for this expenditure record(s), for a given expense period. The transaction code (Transacode), AC_CODE (Account Code), CC_CODE (Cost Center Code), and LOC_CODE (Location Code) are linked to transaction table, vendor code (Vendor) is linked to vendor table, and student code is linked to student table. Students are paid both a monthly amount based on their family code, and also are reimbursed any approved expenses. These expenses are added to the XPENDITR database either automatically (AUTOPOST) or manually. All expenses are tied to an Expense Transaction Code (TRANSACT Database) to ensure full accountability in the system. Reports are generated from this Expense data that are used for General Ledger purposes as well as Payroll purposes.

For a given expense period (usually monthly) - students are issued a check for whatever expenses as well as bonuses and living allowances. All data must have been entered in the XPENDITR database for it to appear in the payroll check. A report is available (Monthly Trainee Expenses) that outlines each expenditure that student has for that expense period. Once this report is satisfactory (and any changes or additions have been made to the XPENDITR database) - payroll checks can be printed. Historical payroll data (Check Register) is maintained in the CHECKREG database (Check Register) - and two (2) reports are available from this data.

As the financial period progresses - students may have invoices that need to be reimbursed to them. This data is entered into the XPENDITR (Expenditure) table. This table is primarily keyed by STUDENCODE. In this table - the financial information (Transaction Code, Vendor Code, and Dollar Amount) is entered. Each record entered for a student is a "line item" on the check stub - IF THE REIMBURSBL FLAG IS = TRUE. If this flag is set to FALSE for a given expense entry - that amount WILL NOT be paid to the student when checks are produced. The data in this table is used to generate payroll checks, expense reports, journal voucher reports. Data is entered into this database either via the AUTOPOST function - or manually entered via the POST menu.

Once all the invoices and expenses for a financial period have been entered and checked. The system will provide a (Monthly Trainee Expenses) report that outlines each expenditure that student has for that expense period to the finance department for issuing their monthly pay checks.
STUDENTS DATABASE:

All of the above databases are used in conjunction with the STUDENTS database. In this table (keyed by STUDENCODE) which is the primary key. Each student is assigned an identification code (STUDENCODE) that is unique (checked at entry). His/her first (First_Name) and last (Last_Name) name are entered and then - using the codes entered in the above "Support" databases - the SCHOOL, TRAINING, SPECIALTY, FAMILY, RAMADAN, and ACADEMIC codes are assigned. A Student is entered into the Student Database where all his attributes are maintained (School Code (School_C), Specialty Code (SPECIALTY_C), Family Code (FAMILY_C), Academic code (ACADEMIC_C), Ramadan code (RAMADAN_C), and Training code (TRAINING_C) are all assigned here. The amounts paid a student are determined by what codes have been assigned to them. The user will use the Student code (STUDENCODE) throughout the application to edit or delete student information. For the student to be eligible for AUTOPOST - at a minimum - that students FAMILY code has to be set. The data assigned to a student when first introduced to the system can be easily maintained by using the correct menu function.
DESIGN IMPLEMENTATION

The Student Payroll System DBMS consists of six modules: Edit Menu, Post Menu, Reports Menu, Listing Menu, Maintenance Menu and Exit Menu. The first module allows the user to add, edit/update, browse, and print student, school, training, specialty, academic, Ramadan, family, vendor, and transaction. The second module consists of three parts. The first part is the Expenditure sub-menu which allows the user to add, edit/update, browse, and print expenditure records for all the students in the system. The second part is the Check Register Sub-menu which allows the user to add, edit/update, browse, and print the check numbers and amounts for each student's expenses and allowance for an expense period. The third part is the AutoPost function which is the quickest and easiest method to post recording data to the Expenditure database for every student in the system. The Reports module provides the reports requested by management. The Listing module provides listings for students, schools, trainings, specialty, academic, Ramadan, family, vendor, and transacting. The Maintenance module consists of back up and restores data from and to disk, archive data from and to disk, deletes menu with a password to delete an individual record for student training, etc. or the entire expense data by expense period.

The edit and delete procedures in the modules were written to prompt the user for information such as student I.D., student name, and school code, etc. A look up table is used to check the validity of the information entered. The look up command checks the table specified to determine if the value exists. Also a pop-up table is used to display all the existing records allowing the user to choose a record instead of typing it. The delete procedure in the module allows the user to delete one record at a time or one expense period at a time. The record is displayed on the screen and the user is asked to verify deletion.

Custom made forms were designed to add, edit or delete individual records. The form and data from one record is displayed on the screen and the user is allowed to modify the displayed data. To assist the user with using the displayed form, the top menu selection area displays function and combined keys to perform certain actions.

The application begins with a logo screen and menu selection at the top of the screen. The user has the option to proceed to the other sub-menus or exit to DOS.

Throughout the application the user is shown a menu screen, with a menu selection area at the top of the screen. The menu screen displays the selection available to the user. Selections are made from the top menu area using left and right arrow keys and pressing return. The return option found in the menu selections returns to the previous menu.