

# A Project Based Approach for Teaching System Analysis, Design, and Implementation Courses

*Nabil A. Yousif<sup>1</sup> and Masoud Naghedolfeizi<sup>2</sup>*

**Abstract**-In curricula of Computer Information Systems at Fort Valley State University, students are required to complete a sequence of two courses in System Analysis, Design, and Implementation. These courses are upper level classes that are taught into two consecutive semesters. In the first semester a comprehensive overview of the system development life cycles is presented with emphasis on planning, analysis and design. After students demonstrate sufficient depth of the subject matter, they are divided into team groups in order to carry out their information system project. Teams are required to perform project planning, analysis, inputs/outputs design, and collaboration with an industry related to their project in the first semester course. In the second semester, students review and revise their design and start the implementation phase using appropriate software and hardware tools. Project presentations are delivered in classroom before invited faculty and students.

*Keywords:* SDLC; DFD; ERD.

## INTRODUCTION

Students majoring in CIS (Computer Information Systems) at FVSU (Fort Valley State University) are expected to take a number of pre-requisite courses prior to taking system analysis & design. These courses include CSCI 1301 & 1302, CSIS 2331, CSIS 3450, and CSIS 3701. The schematic diagram of these courses through CIS curriculum is shown in Figure 1. These pre-requisite courses prepare students with necessary theoretical background, as well as programming and computer skills needed for CSIS 4001 and CSIS 4002.

Before 1999, the CSIS 4001 and CSIS 4002 were taught theoretically without any major course project. In 1999, the author took the responsibilities of teaching these courses. Since then, the author has adapted a new method of teaching that integrates a major course project into the curriculum of the two courses. Over the years, the course project framework has been modified, refined, and reinforced to include the complete system development life cycle from both theoretical and practical viewpoints. In addition, the new project framework requires students to work collaboratively in groups for two consecutive semesters. This will help students retain the knowledge gained in both courses and utilize the knowledge in practical applications. Further, the project gives students a sense of appreciation for the subject matter.

### System Analysis & Design I (CSIS 4001)

This first course is designed to study the important aspects of systems analysis and design. Students will learn the theory and practices of analyzing a business environment and design a computer-based solution. They also complete sets of logical systems specifications, logical and physical diagrams, and an implementation plan for a business system of moderate complexity.

---

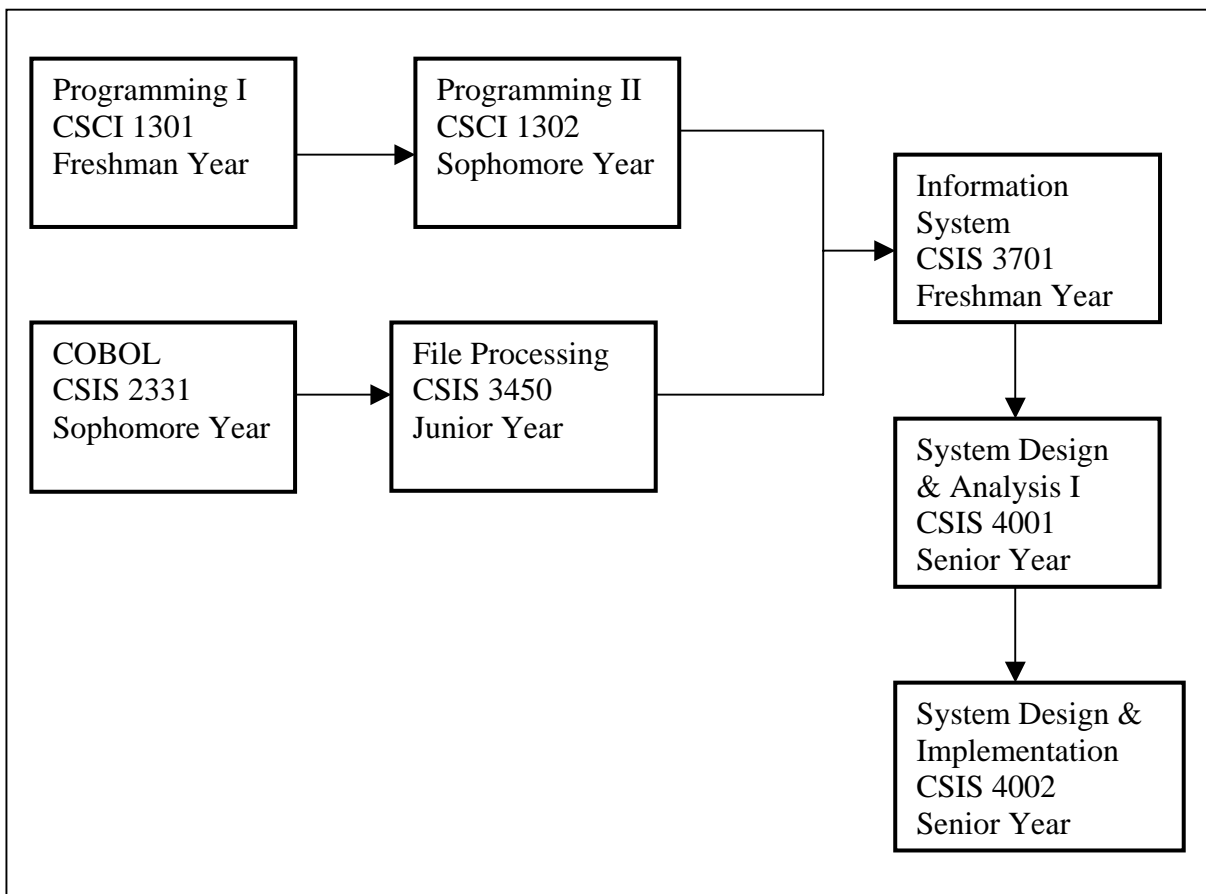
<sup>1</sup> Fort Valley State University, 1005 State University Dr., Fort Valley, GA 31030, yousifn@fvsu.edu

<sup>2</sup> Fort Valley State University, 1005 State University Dr., Fort Valley, GA 31030, feizim@fvsu.edu

### Course Outline:

- The Systems Analyst as a Business Problem Solver
- Required Skills of the Systems Analyst
- The Analyst's Role in Strategic Planning
- The Systems Development Life Cycle
- The Analysis Phase in More Details
- System Requirements
- Techniques for Information Gathering
- Data Flow Diagrams (DFD)
- Documentation of DFD Components
- The Entity-Relationship Diagram
- Understanding the Elements of Design
- Design Phase Activities
- The System Flowchart
- The Structure Chart
- Module Algorithm Design: Pseudocode

Figure 1: Schematic diagram of pre-requisite courses



## System Design & Implementation (CSIS 4002)

In the second part of System Design & Implementation, students will implement the system that is designed in the first part. They will write the application programs and then test and document the programs. In addition, they will complete the operational documentation and procedures.

### Course Outline:

- Designing System Outputs
- Designing System Inputs
- Reviewing system's database design
- Establishing entity relationship
- Performing database Normalization
- Completing Systems Implementation
- Developing Systems Documentation
- Designing User Manual for the System

## PROJECT BASED FRAMEWORK

The use of the project-based approach in computing courses has been reported in literature [1-3] as a productive method of enhancing student learning in course concepts. This approach is especially effective for courses that are related to designing computer-based solutions for business or software engineering areas. The following describes a method used to implement project-based approach in System Analysis, Design, and Implementation courses at FVSU.

As part of initiating the project, students are required to form team groups that will work collaboratively and cooperatively for the duration of the project life cycle. This occurs during *the first semester*. Each group will have a minimum of two students and a maximum of four students. However, under certain circumstances, a student may be allowed to carry out the entire project individually. After the students form their groups, each group will sign a no-complain contract stating that they will work and collaborate as a team and resolve any conflicts that may arise among them during the project duration. A group leader is selected if a team has three or four students. Then, each group will select their project information system. All projects must be approved by the instructor for assurance of uniqueness and level of difficulty. Once the project information system is confirmed students will start the planning phase (Phase I).

### Planning Phase

In the planning phase students identify the primary objectives of the phase. These objectives include the scope of the project information system, project feasibility, project schedule, resource planning, and estimated project budget. Upon completion of the planning phase, students will submit a full report to their instructor for review before starting the analysis phase.

### Analysis Phase

Once the instructor confirms the student's planning phase, students will immediately start working on the analysis phase. Emphasis of the analysis phase will be on the traditional approach as opposed to the object-oriented approach. Students will concentrate on *three main activities* of the analysis phase. These activities are *gathering information, defining system requirements, and creating prototypes* for parts of the new system. This phase is an important stage because it will teach students how to implement concepts learned in the classroom and then apply it to a real world problem.

**Gathering information:** students make several informal visiting trips to the designated organization where they will be designing their project information system. During these trips students will gain a valuable knowledge about the business of the organization, people who will be working with

the system, and the hierarchical structure of the organization. Students will also identify key staff members such as users, programmers, and managers who will be using the information system. Students will then launch their gathering information activity by conducting interviews with the identified key staff members. During the interview session students spend a great amount of time by asking the users many questions about the existing system and examining all the necessary documents. **(See typical student's information system project)**. This activity promotes critical thinking, and problem solving, as well as teaching students how a professional system analyst operates in a real world. After this activity students should have all the information needed to define what the new system must do.

**Defining system requirements:** In this activity students will give emphasis on both functional and technical requirements. They will learn that functional requirements describe activities or processes the system must perform (e.g., calculate student over all GPA) whereas technical requirements describe an operating environment or performance objective (e.g., system must be run on Windows XP, UNIX; or it must response in 20 seconds). Once functional and technical requirements are defined, students will develop both the logical and the physical models. First, they need to differentiate between a **logical model** and a **physical model** [4, 5]. A logical model illustrates what a system must do without committing to a specific technology. For example, a logical model could specify output as a list of data, without committing to a specific format. A physical model would include details about the format. The models that are created by students here are in the form of data flow diagrams (DFD) and entity relationship diagrams (ERD). The DFD represents data movement among processes, data stores, and external entities. Prior to drawing their context diagram students are required to identify all the entities needed for their project information system. The **context diagram** of their system will then be constructed followed by **diagram 0**, **diagram 1** and so on. At this stage (system analysis) students are challenged by how well they understood their information system during the analysis phase. Students who performed a thorough investigation and analysis would not have any difficulties to draw the DFD for their information system. They will then draw a rough draft of the ERD where the relationships between entities are determined. For example, the relationships could be one-to-one, one-to-many, or many-to-many.

Upon completion of this activity, each team presents its DFD and ERD to the entire class for discussions and comments. The instructor also provides guidelines and suggestions as needed for each team group. Each team group then goes through a modification phase to make necessarily corrections and improvement to their DFD and ERD. By the end of this activity students will be able to explain what the system needs to do in detail.

**Creating prototypes:** This is an important step during system analysis phase. Each team will design a prototype of the system according to the system requirements specified during the analysis phase. The prototype must help the users visualize what the new system would do and how it would be operated. For example, a typical prototype should include GUI input and report screens for visualization purposes.

**At the beginning of the second semester** each team will have the opportunity to review and revise their system and make any necessary changes as needed to improve their project. Design phase, implementation, system maintenance and support, and user manual will be completed during this semester.

## Design phase

Each team should be aware of the importance of a **graphical user interface (GUI)** when designing input and output screens. A graphical user interface should allow users to utilize all the capabilities that PC can offer. These include using a pointing device (such as a mouse) and visual images (such as icons, and other graphical images) to issue commands. Each group will design a GUI interface based on a user's perspective. After completing the input and the output designs, students will start their database design. During database design they learn about **normalization** techniques and applications. Normalization improves the quality of a

table design by minimizing data redundancy. Usually students will normalize their table design to the third normal form.

### Implementation phase

This is the stage where teams launch their **application development**. Teams at this phase demonstrate and apply all their previously learned skills, knowledge, and techniques to start their application development. A great amount of time and effort are required from students during this phase to finish their project information system. The main steps that teams apply for developing their system are briefly mentioned below:

- **Program design:** They apply the top-down design in developing their system. This development method is the most popular method that students learned through out their programming courses. They start with the main modules and then break them into smaller, more manageable modules.
- **Review DFD and ERD diagrams:** Prior to coding they review all their DFD diagrams, ERD diagrams, and algorithms for the information system.
- **Programming environments:** They will choose their own programming language. Majority of students use Visual Basic with Microsoft Access to implement their information system. After deciding programming environments they will begin coding their programs.
- **Testing the system:** During this stage, a good and coherent team effort is needed to successfully remove all deficiencies and debug their programs. Each member of the group will be assigned a specific task to carry out during testing stage. For example, some members will search and fix **syntax errors**; others will search and fix **logical errors**. **Desk checking** technique is used for searching and correcting logical errors. During desk checking students develop a various sets of test data and determine the expected result for each set of test data. Following this step they examine the program code using one set of test data and determine the actual result. The expected result is then compared to the actual result. If the results differ, it probably indicates that logical errors exist in the program. Thereafter, they perform system testing in three stages. In **Unit Testing**, which is the first stage, each team tests system modules individually before integrating with other modules. In the second stage, **Integration Testing** is a critical activity where students run various types of testing. For example, they could pass incorrect data type (such as alphanumeric instead of numeric) from one module to another and observe the result. After running these tests students will gain a valuable experience of how integration is performed between modules. Finally, **System Testing** is performed after the entire system is compiled and linked ("built").
- **Documentation:** Documentation is emphasized for both CS and CIS majors in all programming classes for its importance. For this project, each team must provide a full documentation for programs, system, and user (user's manual). These documentations include programs flowcharts, algorithms, system flowcharts, data flow diagrams (DFD), and entity relationship diagrams (ERD). The user's manual must be designed in a manner that can guide none technical persons how to operate the system with minimum or no difficulty. System documentation should be produced throughout the SDLC.

### A TYPICAL INFORMATION SYSTEM PROJECT DESIGNED BY A TEAM OF STUDENTS

The following is an example of a project that was completed by a team group during an academic year of 2001-2002. This team applied the five phases of the SDLC procedures. Due to the large size of the project few excerpts are selected from each phase of the SDLC. In addition, Figure 2 shows the context diagram for the data flow diagram (DFD) and Figure 3 shows the entity relationship diagram (ERD). Students designed both diagrams.

Figure 2: Context diagram

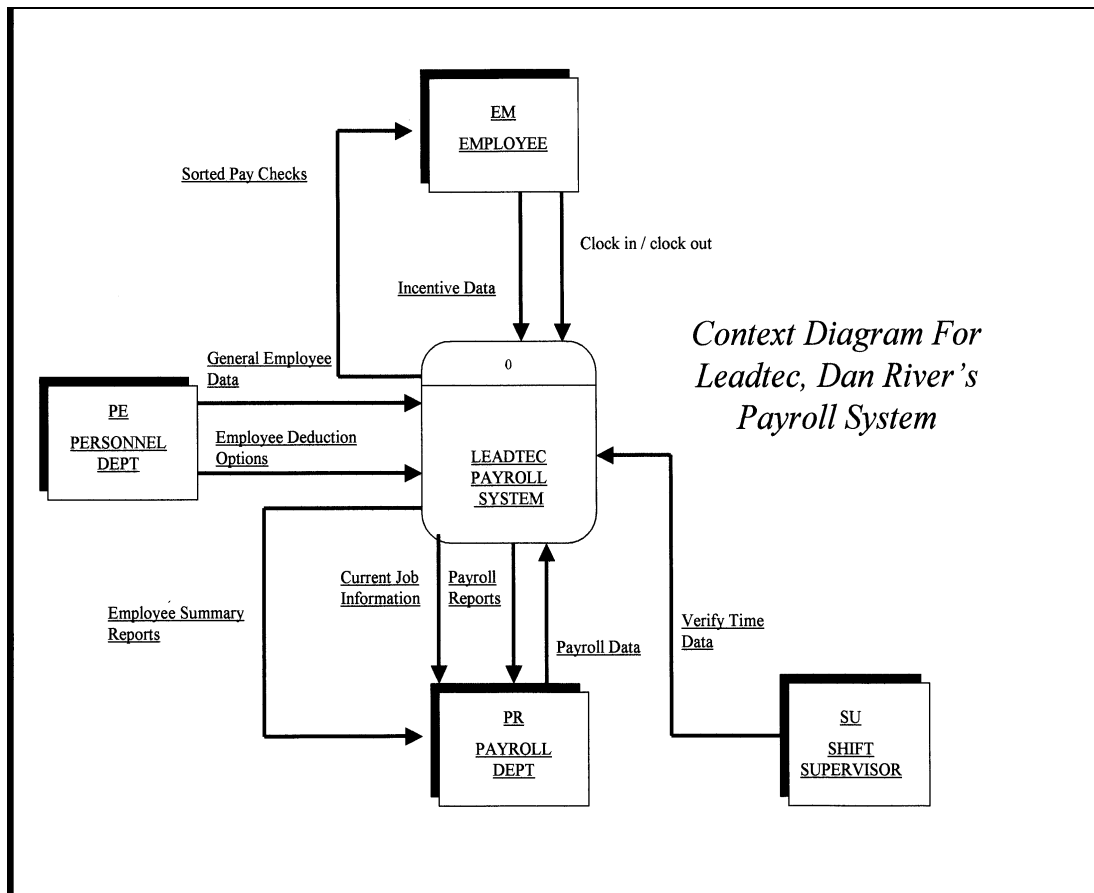
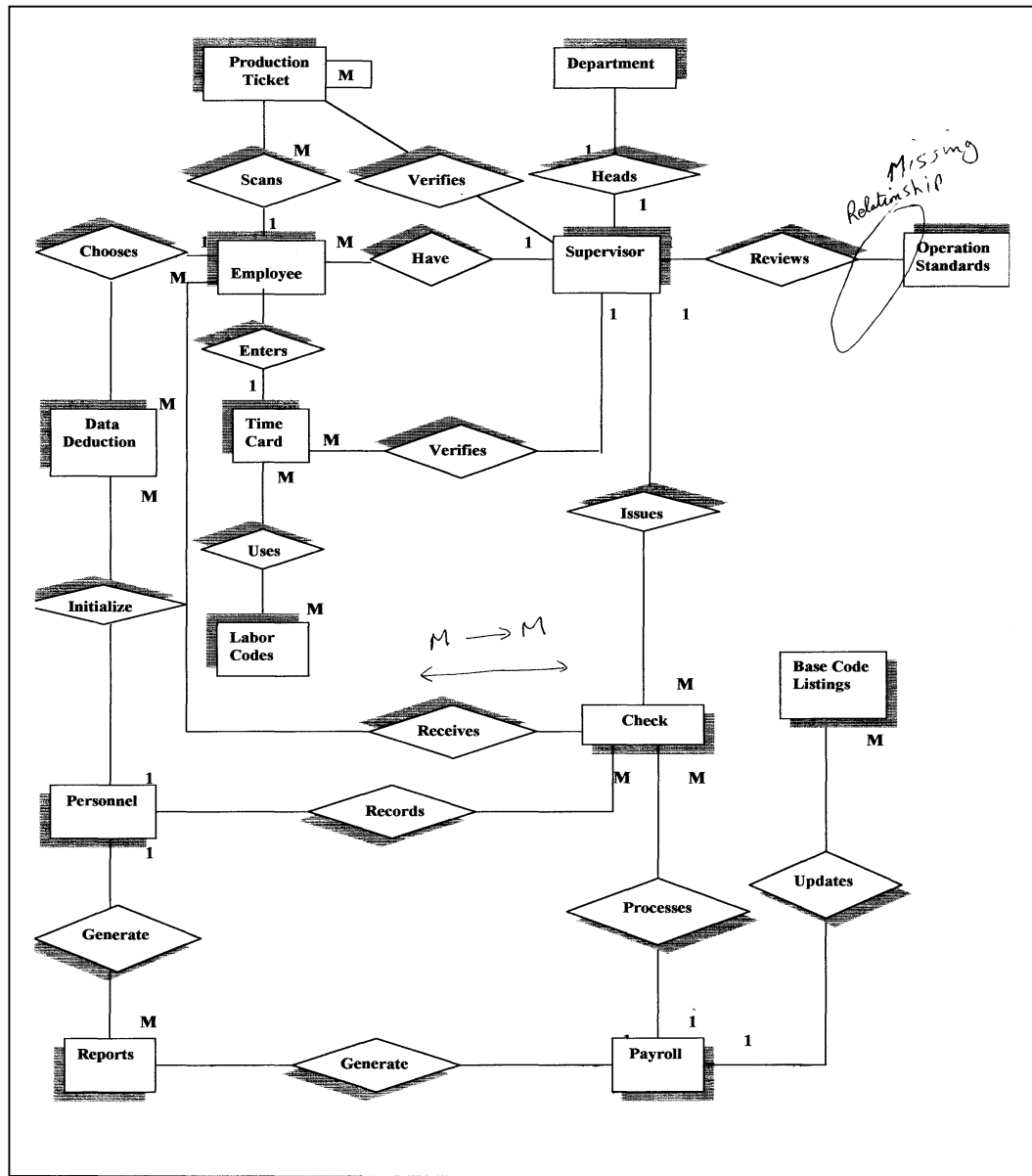


Figure 3: Entity Relationship Diagram (ERD)



## Preliminary Investigation Report

### Introduction

**Group one** (numbers of each group are assigned by instructor) of the System Design and Analysis class at Fort Valley State University has completed a preliminary investigation of the payroll system at Dan River, Inc. This investigation was for the partial fulfillment for graduation by the instructor.

## System Request Summary

Two problems were mentioned in the request for information system services: The payroll system has problems with the overtime pay according to company policy. Also, the system does not have a long-term data storage system.

### Preliminary Investigation Findings

The following problems were found during the investigation:

1. The payroll system has problems with overtime pay according to company policy during weeks with holiday pay. The system was not initially designed to handle this type of situation.
2. The system has to be manually backed up and shutdown every night. The system is not capable of backing up data automatically.

Sample questions from an interview session.

### Phase II- System Analysis

Interview with Mr. Larty Hunt, Industrial Engineer

1. Define all inputs involved in the payroll process and explain each of their functions as they relate to the payroll system? (Ex. Payroll dept, Accounting Dept, Bank)

**Ans.** Payroll data, general employee data, deduction options, employee time data, incentive data, and verified time data.

2. Define all outputs that created from the payroll system and explain the destination of each output as they relate to the payroll system? (Who receives the output)

**Ans.** Date of hire report, employee data summary report, employee time report, employee earnings report, W2 report, employee by department report, and check stub report.

3. At what point does the payroll system at the Fort Valley Plant send its data to the payroll system in Danville, VA? Explain.

**Ans.** During each payroll cycle, it accumulates payroll data for each employee; processes it, and sends it to the corporate Dan River payroll system where actual federal and state deductions are made along with other deductions

## CONCLUSION

Student comments and feedbacks received through the course evaluation instrument showed that they were very confident regarding the knowledge and skills gained through performing and completing their projects. In fact, the performance data from exams and projects indicate that students who did well on projects generally performed well on the course exams. It should be noted that some students did express their initial frustrations about assigned duties and rare personality clashes that eventually resolved through dialogues and professional communications among them. In addition to the above comments, a number of students who have been hired as system analysts have sent their sincere appreciations about the experience gained in these courses and how it helped them in performing their duties professionally and efficiently.

The aforementioned approach of teaching opens a gateway for students to gain a practical experience that will prepare them for the real world challenges in the area of system analysis and design and software engineering. It also teaches students how to carryout the course concepts from the classroom to real-world applications. In addition, students gain a thorough practical knowledge of system development life cycle



(SDLC). Moreover, they learn how to work collaboratively and resolve conflicts in the interest of completing their project tasks.

This practical approach also helps students retain knowledge and concepts learned through the project for a long period of time. In fact, this result has been evident in performance of our students in departmental exit exams where system analysis, design and implementation is one of the main subjects for CIS majors.

## **REFERENCES**

- [1] Sheryl Duggins, "Constructivism in Software Engineering," 2000 ASEE Southeastern Conference Section, Roanoke, Virginia, April 2-4, 2000
- [2] David A. Dampier "Teaching Practical Problem-Solving in Software Engineering Through Industrial Experience," 2002 ASEE Southeastern Conference Section, Gainesville, Florida, 7-9 April 2002
- [3] Barbara Bernal Thomas "Usability Problem Solving: Designing Good Interactive Software," 2002 ASEE Southeastern Conference Section, Gainesville, Florida, 7-9 April 2002
- [4] John W. Satzinger, Robert B. Jackson, Stephen D. Burd , "System Analysis and Design in a Changing World, Fourth Edition", Course Technology, 2007.
- [5] Gary B. Shelly, Thomas J. Cashman, and Harry J. Rosenblatt, "Systems Analysis and Design Six Edition," Course Technology, 2006

### **Nabil A. Yousif**

Mr. Yousif is an assistant professor of computer science at Fort Valley State University. His area of teaching interest includes database systems, system, analysis and design, computer programming, and business information systems. His area of research interest includes database and information system designs.

### **Masoud Naghedolfeizi**

Dr. Naghedolfeizi is an associate professor of computer science and engineering at Fort Valley State University. His teaching interests include computer-based measurement and instrumentation, data communications, computer networks, and programming languages. His current research interests include applied artificial neural networks, signal processing, and measurement systems.