

# Sophomore Block Project 2003-2004

**Many of the terms used throughout this description will be covered during the course of the semester. Don't panic if you are not familiar with the terminology!**

This is a joint project across the JDEP Sophomore block. The fall semester portion of the project is to create a "mini-MRP" system. It should be scalable and maintainable because it will be extended in the spring by converting your "mini-MRP" system into an ERP system. The fall semester portion of the project is fairly well structured in terms of features that your system must include. The spring semester will be much more open ended, allowing you to identify those features of an ERP system that you believe to be most useful and capable of building before final deadlines.

Your fall semester "mini-MRP" system will be able to forecast orders, generate a master production schedule (MPS), and create planned order schedules and exception reports when the production schedule is not feasible. The focus of this project is the computational engine. A fancy graphical interface is not required, nor is it expected.

The primary components of your system will be: a forecasting module, a bill of materials (BOM) data structure, an inventory records/inventory status data structure, a MPS module, and an MRP module.

The forecasting module will read a file of historical orders to be used for your forecasting. This file will be provided. The forecasting module will support, at a minimum, one forecasting algorithm per team member. Users will be able to select the forecasting method of choice. Not all of the forecasting methods chosen can come from the business textbooks for the sophomore block. Your system should also analyze the accuracy of the set of forecasting algorithms and recommend the most accurate algorithm based on the historical data. After the forecasting algorithm has been selected, the module will forecast product orders for the next  $k$  time buckets, where the user inputs the desired  $k$ .

The MPS module will accept the monthly order forecast as input and create a master schedule to satisfy forecasted demand. The forecast schedule will be displayed on the screen and the user will be given the opportunity to make changes to the forecast. Once the changes (if any) are made and the user accepts the schedule, MPS will be set.

The MRP module will accept as input the MPS, read a BOM file, inventory records/inventory status file, and determine the inventory and purchasing needs to meet the MPS. The MRP system will generate reports that detail the planned order schedule that are organized by release date, order date, supplier, and by product. The user will have the option of selecting any of these order schedule reports to be displayed on the screen. If a feasible schedule cannot be created, an exception report will be generated that identifies the problem part and the earliest point in time at which the schedule becomes infeasible. This is often called *pegging*.

The format and content of the inventory and BOM files will be provided. Your team will define the format of the displays and reports generated. The user must be able to either specify the filenames as arguments to the program, or should be able to enter them during execution.

The MRP *engine* must execute on a Linux/Unix system and be written in C++.

Your team will be required to create a set of milestones for your project. You will send email to the course instructors (CS and Business) at each milestone that summarizes your status, indicates any problems encountered, and describes any risk mitigation strategies being considered or adopted.

A final CS report will be required that introduces the problem you are solving, describes your solution with moderate detail, such as the types of data structures and algorithms used for major components, discusses alternative designs and why your design was chosen, analyzes the time and space complexity of your main algorithms, and describes how your system was evaluated. The report should follow the standard CS template available from Dr. Goddard's Writing Resources pages (linked from the course webpage).

A user manual and training session are also required. The user manual is due December 3. The training sessions will be held Dec 6 and 8. The final CS system report and source code for your project is due 9:00pm Saturday, December 11, 2004. Work with the communications faculty (i.e., Rick and Mary) during the preparation of your user manual, CS final report, and training session.

As usual, the coding standard must be followed and a Makefile must be submitted with the source code. Use CSE Web handin to submit your code, and submit a hardcopy of the report in class.

This project will be scored as follows for your CS course:

Program correctness (as defined)	30%
Quality of design/readability	25%
In-line documentation/coding standard	10%
Final report	30%
Thoroughness of test cases	05%

While the final grade in the business course will be developed by the combination of business and communications faculty, the following breakdown provides an overall picture of the importance of the different areas of emphasis:

Accuracy of Results (forecasts and reports)	30%
Ease of Use	10%
User's Manual	20%
Training Session	20%
Professionalism	20%