

## Milestone 5 Implementation Status Document & Performance Testing

### Purpose:

To identify the current status of implementation in relation to the proposed increments as stated in the SRS and to comment on any issues that are still outstanding.

### Current Status:

At the beginning of the project we defined three [3] increments we would produce using an agile approach. A purchase order module, a vendor module, and an inventory module were the three increments that we planned to implement in that order. We have currently implemented all three modules and feel confident that we have nearly reached our goals for the project by completing all three increments. The SRS and ER Data Model were revised for testing and progress toward revising the remainder of the documentation is well under way.

### Desired Functional Improvements:

The system that we have will continue to be improved for functionality until final application turnover at the end of Milestone 6. Some of the issues that should be addressed before completion as well as in future modifications include:

- A method for archiving data – This process is complicated because of the related nature of all the data. For instance, a vendor cannot be permanently deleted if a purchase order is linked to that vendor. For the vendor to be deleted, the purchase orders linked to said vendor would also have to be deleted or stored in a archived state.
- A pre-report engine that takes inputs from the user to dynamically create reports rather than having to rely of a series of individual reports.
- A preview function for the reports.
- A full blown search module to allow the user to search for purchase orders based on any number of different criteria such as a date range or vendor.
- A way to change the number sequence for the purchase order.

### Performance Issues:

Our database application has very little overhead processing to perform other than to store records and populate lists. We realize that as the database begins to fill, the use of indexes in the database will become necessary. Because of the amounts of data that the database will support, we have intentionally stored values in each purchase order that we think will be necessary for speedy lookups and computations for queries and other functional adaptations in the future. Examples of these values are *SubTotal* in the transaction form and eventually *Total* in the purchase order forms. While all of the information exists in the database to compute these values on the fly, we think that planning for performance in these issues by including an extra field in a table or two is worth the price of disk space. After testing the database for usability and considering the project scale, we did not encounter any area that was in need of performance modifications and so we feel that our time would be best spent adding functionality.