HONG KONG INSTITUTE OF VOCATIONAL EDUCATION

Case Study for Software Project Management & Quality Assurance

Ai-Dai Consultants Co. is a well-established consultant company with 5000 employees worldwide that offers consultancy services to professionals in different business sectors. In Ai-Dai, all internal projects are sponsored by the Board of Directors (The Board).

A new web-based system will be implemented as a part of the intranet to perform library management facility (librarian system). The new system will replace the current manual system, which has a lot of errors and much manual work involved. The management expects this system to increase the efficiency of searching and borrowing resources and improve resource utilisation.

Librarians have good experience with electronic records and operating web-based systems. The IT department has strong knowledge of its existing intranet, web-based system, databases and Internet technology, but they have not designed any librarian system before. The ISP should scale its services to accommodate the new library system.

The project team will likely consist of three or fewer people. Management committed a few librarian experts to assist the project team in system design. The project time frame is flexible. The development budget of the system is around 500k HKD, and the annual maintenance cost is about 30k. It is expected to save 1M of administrative charges and labour costs in the first year.

The company board has strong support in the project, and the project manager, Mr Freerider, is a knowledgeable web-based system expert. The users of the system are expected to appreciate the new system. Ai-Dai consultants already have a consultant with expertise in similar applications to assist with the project, and the librarians would need some training before using the system.

Freerider's initial plan is to develop an online resource searching function and an interface to the Human Resources System (HR system) to collect employees' records as borrowers' profiles. The second version will add the borrowing and reminder function. Freerider plans to build the system by an in-house team, and he estimates that a few part-time experts (expert A, expert B etc.…) would be needed in the implementation phase.

The project team begins performing analysis activities, and the following is a summary of the as-is condition.

When the department purchases a book, clerical staff (i.e. clerk) records the book details and assigns a unique ID number. The ID number is a sequence number, and it is in ascending order. The clerk looks up the "book record file" to find the following un-assigned ID number and assign it to the book. The book is then stocked in the cabinet for borrowing.

When a staff member recommends the manager purchase a book, the team will look into the "book record file" to make sure the book is not in the departmental library.

When a staff wants to borrow a book, he first finds whether the book is already stocked in the department. They look at the "book record file" to find the book. If the book exists, the staff will ask the clerk to borrow the book. The clerk will then go to the shelf to get the book. If the book is not on the shelf, the clerk will check the "loan record file" to find if another staff already borrows the book. The clerk will then inform the borrower to return the book as soon as possible.

There are some identified problems with the current system:

- *Clerical staff sometimes are bustling, resulting in a duplicate ID. When new books arrive, and a teaching staff needs to borrow the book urgently, the clerk typically makes a photocopy of the book's front page, looks at the "book record file", assign, mark and stick the new ID label to the book. The clerk will then mark the ID on the photocopy and put it aside. The "book record file" will be updated later. The problem arises when the clerk forgets to update the "book record file" and sets another new book with the same unique ID.*
  As the borrowed books are not given any due date, whoever borrows the books may forget to return.
- *Sometimes a book is borrowed, but the record is not found. It usually takes a long time to locate the book by asking every individual staff.*
- *At the end of the financial year, the department needs to report the amount used to purchase the books to obtain funds for the following year. Currently, the clerk takes the "book record file" and sums up all the financial year costs.*
- *Staff is not able to find books according to categories or publishers. They can only look through the "book record file" sequentially to locate the book.*

The hardware, software, and networks would need to be integrated into the current infrastructure for the new librarian system. Currently, Ai-Dai uses Linux as the operating system for the server and Oracle's database, and the new system is expected to be written in Java. The current system configuration could likely handle the traffic for system requirements.

It is decided that during unit tests, the test plan for Web Interface and System Interface would be developed directly from the program specification. System Interface would also review the source code.

For the integration tests for the web, interface components would test user interfaces and use scenarios to ensure that the interface worked adequately. The system interface component would test to ensure that the system performed calculations correctly.

During Systems tests, requirement tests would be conducted on all system parts to ensure that all requirements were met. The user-facing components test to see how many transactions they could handle before not providing a response time of 3 seconds. Also, usability tests would be conducted on the web interface.

Acceptance tests would be conducted with made-up data only. Tests would be done during the training of the Ai-Dai staff.

Appendix

	Duration Day(s)	Predecessor
System Requirement	1	
Feasibility Analysis	3	
Project Schedule	1	System Requirement Feasibility Analysis
Finalize Project Plan	3	System Requirement Feasibility Analysis Project Schedule
Project Plan Presentation	1	Finalize Project Plan
Project Plan Confirmation	5	Project Plan Presentation
Create Requirement Specifications	3	Project Plan Confirmation
Analysis Current System & Identify improvement	2	Project Plan Confirmation
Develop concept for the new system	3	Analysis Current System & Identify improvement
Create Functional Requirement	2	Analysis Current System & Identify improvement
Create Non-Functional Requirement	2	Analysis Current System & Identify improvement
Finalize System Proposal	3	Create Functional Requirement Create Non-Functional Requirement
Confirm Version System Proposal	3	Finalize System Proposal
Architecture design	1	Confirm System Proposal
Process design in UML	1	Confirm System Proposal
UI design	1	Confirm System Proposal
Report design	1	Confirm System Proposal
Database design	1	Confirm System Proposal
Program coding	5	Architecture design Process design in UML UI design Report design Database design
Confirm version	5	Program coding
Testing	15	Confirm version
Documentations	40	Confirm System Proposal