

## **TABLE OF CONTENTS**

- 1. Introduction
  - 1.1 Overview of the Project
  - 1.2 Objectives of the Project
  - 1.3 The Need for the Project
  - 1.4 Overview of Existing Systems and Technologies
  - 1.5 Scope of the Project
  - 1.6 Deliverables.
- 2. Feasibility Study
  - 2.1 Financial Feasibility
  - 2.2 Technical Feasibility
  - 2.3 Resource and Time Feasibility
  - 2.4 Risk Feasibility
  - 2.5 Social/Legal Feasibility
- 3. Considerations
- 4. References

## 1. Introduction

### 1.1 Overview of the Project

OES is an online examination system which automates the process of creating question papers, performing online quizzes, file submissions and marking criteria. OES is a web based application that provides interfaces for various stake holders (lecturers, students and teaching assistances).

Questions can be added to the system with their associated parameters (number of allocated marks, time allowed), and the lecturer can create quizzes and assignments using them. Moreover OES is capable of holding online quizzes and homework assignments.

OES is capable of marking MCQ type questions, and has the ability to grade essay type questions based on key words. Users can get statistical distributions of the examination results as well as performance for each question in many forms (graphs, tabular form).

### 1.2 Objectives of the Project

The objectives of this project is to-

- Develop a central database of questions for each subject
- Automate the process of creating quizzes
- Provide an online examination platform
- Provide the ability to submit files (for homework and other detailed kind of assignments)
- Automate the grading process of MCQ type questions
- Makes it easy to check essay type questions based on key words
- Provide statistical information regarding examinations, questions and student performance in many forms (including graphs, tabular form).

### 1.3 The Need for the Project

Manual management of examinations is a cumbersome tasks when it comes to a large audience like in a university. Project OES looks into this matter and automates this process. Other than examinations, OES is capable of managing assignments and other file submissions.

In the current context, getting statistical information about the performance of a student and a set of students needs extra effort to do things separately. OES provides a solution to this problem by providing automatically generated statistical information in many aspects.

- Student performance
- Performance of a batch of students in an examination
- Performance for an individual question
- Overall statistical distribution for a particular subject over several years

## 1.4 Overview of Existing Systems and Technologies

University Moodle system: Moodle system consists some of the use cases that are implemented in the OES. But OES is mainly concerned with examinations and assignments, unlike the Moodle system which contains subject materials.

Main Technologies associated with OES.

- Web programming technologies (JS,JSP,HTML,CSS)
- MySQL(Database)
- Diagram and design tools (Visio, Nclass, Draw.IO, Microsoft project)

## 1.5 Scope of the Project

Main actors of this system

- Lecturers
- Teaching assistants
- Students
- Administrative staff (examination division, undergraduate division)

Main use cases associated:

1. Lecturer and teaching assistant can
  - Submit new questions
  - Create papers using existing questions
  - Schedule an online quiz
  - Schedule a homework assignment or a file submission assignment.
  - Grade file submissions
  - View results in a quiz
  - View statistical details of students, a quiz, a section of a subject, a subject.
2. Student
  - Do an online quiz
  - Submit assignment files
  - View grades and performance (Statistical details)
3. Administrative Staff
  - View statistical details

## 1.6 Deliverables.

A web based software system. This contains a central database and functionalities for various stakeholders. Since many number of stakeholders are involved, different GUIs will be provided to different users.

## 2. Feasibility Study

### 2.1 Financial Feasibility

Being a web application OES will have an associated hosting cost. Since the system doesn't consist of any multimedia data transfer, bandwidth required for the operation of this application is very low.

The system will follow the freeware software standards. No cost will be charged from the potential customers. Bug fixes and maintaining tasks will have an associated cost. At the initial stage the potential market space will be the local universities and higher educational institutes.

Beside the associated cost, there will be many benefits for the customers. Especially the extra effort that is associated with paper making and marking will be significantly reduced while the effort to create descriptive statistical reports will be eliminated, since reports generation is fully automated.

From these it's clear that the project OES is financially feasible.

### 2.2 Technical Feasibility

Project OES is a complete web based application. The main technologies and tools that are associated with OES are

- HTML
- CSS
- JSP
- MySQL
- JS
- NetBeans
- Diagram drawing tools
  - NCLASS
  - Microsoft Project
  - Visio
  - Draw.IO

Each of the technologies are freely available and the technical skills required are manageable. Time limitations of the product development and the ease of implementing using these technologies are synchronized.

Initially the web site will be hosted in a free web hosting space, but for later implementations it will be hosted in a paid web hosting space with a sufficient bandwidth. Bandwidth required in this application is very low, since it doesn't incorporate any multimedia aspect.

From these it's clear that the project OES is technically feasible.

## 2.3 Resource and Time Feasibility

### Resource feasibility

Resources that are required for the OES project includes,

- Programming device (Laptop)
- Hosting space (freely available)
- Programming tools (freely available)
- Programming individuals

So it's clear that the project OES has the required resource feasibility.

## 2.4 Risk Feasibility

Risk feasibility can be discussed under several contexts.

### Risk associated with size

Estimated size of the product in line of codes:

Being a web application with many number of stakeholders, OES will contain significant amount of code lines. As the system doesn't contain any multimedia aspect, the file sizes and the complete project size will not exceed 200MB.

Estimated size of product in number of programs:

Though the application supports many stakeholders, it will be constructed as a single web application with a single login page rather than having many number of sites for different users. Depending on the access rights, the contents will be showed or hidden.

Size of database created or used by the product:

Database size will not exceed the values supported by MySQL (65526 entries per table). Number of relations and entities are minimized by using best practices of normalization theories.

Users of the product:

- Lecturers
- Teaching assistants
- Students
- Administrative staff (examination division, undergraduate division)

Number of projected changes to the requirements for the product? Before delivery? After delivery:

The requirements are clearly identified before the implementation phase. Being a general product (not specific to a single user) the requirements will be changed only if new functionalities are added to the system.

Amount of reused software:

Though the main logics are implemented throughout the project, OES will use some JSP libraries to incorporate additional functionalities such as to support file uploads.

Business impact risks

Effect of this product on company revenue:

OES can be implemented either as an individual system, or can be integrated to an existing system such as university Moodle system. Since it automates some key features associated in college education process, the users can increase the revenue.

Reasonableness of delivery deadlines:

Being a 14 weeks project, the project OES will have several deadlines and deliverables that are scheduled successively. Depending on the coding and designing cost and effort, the deadlines are quite reasonable.

Number of customers who will use this product and the consistency of their needs relative to the product:

As mentioned above, we can categorize stakeholders into 4 main categories. This system can support many number of users simultaneously due to the low bandwidth requirements.

Number of other products/systems with which this product must be interoperable:

This product can be integrated with current university Moodle system with slight modifications. Doing so will add a significant value to both systems.

Sophistication of end users:

OES is designed while maintaining the complexity at a very low level. Usability is highly improved by providing help documents and making GUIs easy to use.

Amount and quality of product documentation that must be produced and delivered to the customer:

Customer will be provided with a complete online user manual. As the software is implemented as a freeware and open source system, the code will be available for free

Costs associated with delivery:

At the initial stage the associated cost will be for the hosting cost.

Customer related risks:

OES is a general type of product (not designed just for a single college). Before implementing the system in an educational institute, there will be some basic modifications required.

Development environment risks

Is a software project management tool available?

Microsoft Project will be used as the main project management tool.

Are tools for analysis and design available?

OES will require several designing software

- Draw.IO (database design)
- NCLASS (class diagram)
- Visio (Software related diagrams)

Are compilers or code generators available and appropriate for the product to be built?

JSP will be used as the main scripting language. All the libraries and interpreters will be freely available.

Are testing tools available and appropriate for the product to be built?

JUNIT is the main testing tool that will be used. JUNIT is freely available tool that supports automated testing.

Are software configuration management tools available?

Configuration management will be done using GIT that is freely available.

Does the environment make use of a database or repository?

This is a database oriented system that will use MySQL.

Are all the software tools integrated with one another?

Main deliverables will be packaged under a single project. All the stake holders will have a single login page.

#### Process issue risks

OES will follow the RUP software development process. This provides the flexibility to accommodate changing software requirements of OES.

#### Technical issue risks

Are specific conventions for code documentation defined and used?

Software code will be freely available and the code documentation will be provided.

Do you use a specific method for test case design?

JUNIT will be used as the main testing tool that automates the testing process.

Are configuration management software tools used to control and track change activity throughout the software process?

GIT will be used throughout the software implementation process.

#### Technology risks

Is the technology to be built new?

All the technologies are very well established and old enough (but not obsolete).

Do the system requirements demand the creation of new algorithms, input or output technology?

OES will have several algorithms to generate quizzes, marking of essay type questions and to generate statistical distributions.

### 2.5 Social/Legal Feasibility

OES uses freely available development tools, and provide the system as an open source system. Only the maintenance cost will be charged from potential customers.

JSP Software libraries that are used in this system are free open source libraries.



Since this new system eliminates the effort to make statistical distributions, it will have a great impact in a university system.

### 3. Considerations

#### Performance:

OES requires a very low bandwidth, hence the performance will not degrade with increasing number of potential users. At the development stage, a free hosting service will be used. But when installing the system to a real university environment, it will be hosted in a much more reliable server to increase the performance.

MySQL will provide the adequate speed for database transactions. Since no big data analysis is done, MYSQL is the ideal database for this project.

Response time: less than 2 seconds

Processing time: Less than 2 seconds (no batch processing involved)

Query and reporting times: yet to be tested

Throughput: yet to be tested

Storage: yet to be tested

#### Security:

Security measures are provided in many aspects in this system.

#### User authentication:

Users will have to authenticate using the username and passwords.

Depending on the access level each user will gain functionality of the system. Passwords can be changed by the user.

#### Login details:

Each user's login time and logout time will be recorded in the system, to make the tractability process easy in case of a faulty action.

#### Usability and ease of use:

Users will be provided with a complete user manual as a pdf. The interfaces are designed to make it easy for any potential user to get familiar with the system within one hour. No additional training is required to use the system.

#### Capacity and scalability:

OES system can accommodate many simultaneous users. The system is designed to make it easy to integrate to an existing system like the Moodle system.

#### Availability:

System will be available throughout the 24 hours. Mean time to failure and mean time to repair will be decided to increase the availability. With a paid hosting space, the availability can be guaranteed to a great precision.

Maintainability:

OES is designed using the best practices of RUP and OOP. Since every single segment in the system is very well structured, the system is highly maintainable.

4+1 view model will be used as the main architectural pattern in this system. Hence the separation of each task is improved, hence maintainability improved.

#### 4. References

([groups.engin.umd.umich.edu/CIS/course.des/cis375/projects/](http://groups.engin.umd.umich.edu/CIS/course.des/cis375/projects/), n.d.)  
([eyefodder.com/2011/06/](http://eyefodder.com/2011/06/), n.d.)