Ben-Gurion University
of the Negev

Exam Management System

Application Requirements Document (ARD)

Students: Roi Tiefenbrunn, Ofek Nov, Idan Aharoni, Mor Abo

Customer: Dr. Mayer Goldberg

Contents

Chapter 1 – Introduction 3

The Problem Domain 3

Context 4

Vision 4

Stakeholders 4

Software Context 5

Chapter 2– Usage Scenarios 6

User Profiles — The Actors 6

Use cases 7

Special Usage Considerations 7

Chapter 3– Functional Requirements 8

Chapter 4 – Non-functional Requirements 10

Implementation constraints 10

Platform constraints 10

Special restrictions & limitations 10

Chapter 5 – Risk Assessment & Plan for the Proof of Concept 11

Risk Assessment 11

Plan for the Proof of Concept 11

Appendices 13

Chapter 1 – Introduction

The Problem Domain

With the increase in student body size, it has become difficult to manage grading within the given time constraints. This has led many courses to adopt a multiple-choice format in their exams, which is faster to grade, and easier to analyze. Still, creating challenging multiple-choice tests in the sciences is a non-trivial task, and creating one-time-use questions is costly and unsustainable.

Over the years, Dr Goldberg has developed a unique approach to writing multiple-choice questions in science, based on the idea of meta-questions:
A meta-question represents an abstraction over a multiple-choice question and is a structure from which it is possible to derive several, often a great many, multiple-choice questions. To support the creation of exam questions and the synthesis of exams, Dr. Goldberg has defined a DSL for describing various kinds of meta-questions and a software system that synthesizes exams from a given selection of meta-questions.
The system is written in Scheme and uses text files to store meta-questions.

As the number of meta-questions grew, the weaknesses of the current system have become apparent, and a re-design is in order. The most notable change in the new design is that it is designed as a workflow-management system that enables different course staff members to collaborate in creating and developing new questions.

Our proposed solution is to develop a system that will be an upgrade to the existing one, while still relying on the same philosophy and approach of meta-questions maintenance, offering better solutions where the current system fails - with a focus on scalability, accessibility, labor management all while being easy to learn and use.

Context

The system is designed to manage work inside of an academic course between the lecturer(s) and the Tas.

The course administration will develop, test, and create multiple-choice questions and generate exams through the system.

The system will be installed remotely on the university's servers and accessed through web applications, while the data will be stored in databases.

Vision

- Workflow Management System:
  - Support various roles (for TAs, Lecturers, System Admins).
  - Algorithmic generation of tasks based on priority and needs.
  - Review and accept the TA's suggestions about editing questions.
  - Tracking productivity (= blame feature).
- UI:
  - Web-based system.
  - Dashboard based on Role.
- Content creation:
  - Manage content by subjects/keywords.
  - Creation and management of meta-questions, questions, and appendices.
  - Creation and management of possible solutions/distractors per question.
- Output creation:
  - Flexible LaTeX-based creation of exams/keys/solutions.
- System:
  - Have both WMS and content handled in DB.
  - Version control of questions, handled in DB.
  - Easy install, migration, backup, and cloning of the system.

Stakeholders

Client: Dr. Mayer Goldberg – Supervisor Lecturer for the "Compiler Construction" course.

Users: "Compiler Construction" course administration – lecturers and TAs.

Additional people of interest: Other courses' supervisors are interested in the system when it comes to existence.

Software Context

Access

Access to the system will be through browsers and only while connected to the university's network.

Regular Use (Course downtime)

Users will access the system to maintain question stock based on various tasks completed based on their respective roles.
High roles will be able to track work velocity and quality of lower-level roles.

Pre-Exam

High roles will be able to produce exams/keys/solutions through the system and by choosing subject distribution.

Post-Exam

Inputting students' answers will be used to further analyze the questions/answers/distractors quality.

Main Expected Functionality:

The system should support the following functionalities:

- **Add meta question:** Add meta-questions (questions, solutions distractors) to a specific course.
- **Ask for a task:** A user can get a task to work on, a task could be to validate the correctness of the meta-question solution and distractors. The task will be pulled from a task pool which will prioritize the tasks
- **Complete a task:** complete a task received from the task pool
- **Add/Remove course staff:** The course administrator can add and remove staff from his course.
- **Generate exams:** Course lecturers can automatically generate exams which will be based on the meta-questions.

Chapter 2– Usage Scenarios

User Profiles — The Actors

1. System Administrator:
    - Characteristics:
        - Manages the overall WMS at the department/university level.
        - Handles tasks like migration, installation, updates, and system management.
        - Manages courses within the WMS.
    - Data Exchange:
        - Migrates, installs and updates the WMS system.
        - Manages courses, including creating, backing up, and removing course entries.
2. Course Lecturers:
    - Characteristics:
        - Responsible for managing courses, defining course staff, and overseeing exams.
        - Assigns roles to course staff and defines exam parameters.
    - Data Exchange:
        - Defines course staff and assigns roles.
        - Defines exam parameters, including type, length, date, direction, and stylistic elements.
        - Generate exam documents.
3. TAs:
    - Characteristics:
        - Involved in the creation and validation of questions, stems, keys, distractors, and solutions.
        - Performs various activities based on assigned roles.
    - Data Exchange:
        - Works on existing questions, validates, edits, and adds new content.

Use cases

1. Manage Courses:
   - Actor: System Administrator and Course Lecturer
   - Description: Managing courses, includes tasks such as creating new courses, updating course information, and removing courses when necessary.
2. Manage Exams:
   - Actor: Course Lecturer
   - Description: Define and manage exams. This involves specifying exam type, length, direction, date, etc.
3. Manage Course Staff:
   - Actor: Course Lecturer
   - Description: Define the course staff roles, including instructors, TAs, and graders. They assign roles to individuals, track their activities, and manage any changes in the course staff.
4. Edit and Validate Questions:
   - Actor: Course Staff (Course Lecturers, TAs)
   - Description: Course staff members edit, validate, and manage questions within the system. They can add, delete, or modify question elements, such as stems, keys, distractors, and solutions.
5. Generate Exam Documents:
   - Actor: Course Lecturer
   - Description: Generate exam documents based on the defined parameters. This includes creating different versions of exams.
6. Inspect Changes by Course Staff:
   - Actor: Course Lecturer
   - Description: Course administrators review and inspect changes made by course staff members. They can accept, reject, modify, or conditionally validate these changes based on the quality and relevance of the modifications.
7. Search and Edit Questions:
   - Actor: Course Staff (Course Lecturer, TAs)
   - Description: Course staff members can search for specific meta-questions and edit them. The system allows users to locate questions based on keywords and strings, facilitating efficient editing and validation processes.
8. View Dashboard:
   - Actor: All Users
   - Description: Users view a personalized dashboard based on their roles in the current/given course. The dashboard provides an overview of relevant information and tasks for the user.
9. Generate Solutions for Exams:
   - Actor: Course Lecturer
   - Description: Course lecturers generate solved exams for distribution to students after an exam is completed. This document includes keys and explanations.

Special Usage Considerations

The system should be tangible and attractive, due to the high need for user cooperation for the system to be beneficial.

Chapter 3– Functional Requirements

MH = Must Have

NTH = Nice to Have

No	Description	Priority	Risk
1	**System Administrator**	-	-
1.1	The system shall support the System Administrator in performing a migration, installation, updating, and overall management of the WMS system.	MH	Medium
1.2	The system shall provide the System Administrator with the functionality to install the WMS on departmental/university systems.	MH	Medium
1.3	The system shall implement the ability for the System Administrator to clone the WMS for duplication or testing purposes.	MH	Low
1.4	The system shall enable the System Administrator to update the WMS with new features or patches.	MH	Medium
1.5	The system shall support the System Administrator in the creation of new course entries within the WMS.	NTH	Low
1.6	The system shall provide functionality for the System Administrator to back up and remove course entries as needed.	NTH	Low
1.7	The system shall allow the System Administrator to assign initial roles, particularly the administrator role, for each course.	MH	Low
1.8	The system shall provide functionality for the System Administrator to remove and change roles as needed in course staff.	MH	Low
2	**Lecturer**	-	-
2.1	The system shall enable the Lecturer to define course staff.	MH	Low
2.2	The system shall allow the Lecturer to assign roles to course staff.	MH	Low
2.3	The system shall enable the Lecturer to define exams.	MH	Low
2.4	The system shall allow the Lecturer to define the exam type (test, quiz).	NTH	Low
2.5	The system shall enable the Lecturer to define the exam direction (RTL, LTR).	MH	Low
2.6	The system shall allow the Lecturer to define the exam length.	NTH	Low
2.7	The system shall enable the Lecturer to define the exam date.	NTH	Low
2.8	The system shall allow the Lecturer to define stylistic elements (fonts & sizes).	NTH	Low
2.9	The system shall enable the Lecturer to define frontal matter for the exam, and test.	NTH	Low
2.10	The system shall allow the Lecturer to define basic headers.	NTH	Low
2.11	The system shall enable the Lecturer to define instructions for an exam.	NTH	Low
2.12	The system shall allow the Lecturer to define the basic layout (number of columns, number of items (4 or 5)).	NTH	Low

No	Description	Priority	Risk
2.13	The system shall enable the Lecturer to define the number of versions for the exam.	NTH	Low
2.14	The system shall allow the Lecturer to select questions.	MH	Low
2.15	The system shall enable the Lecturer to select appendices.	MH	Low
2.16	The system shall allow the Lecturer to generate exam documents.	MH	Medium
2.17	The system shall enable the Lecturer to generate exam versions.	MH	Low
2.18	The system shall enable the Lecturer to generate special versions for reading-impaired (A4, but landscape, and 41% larger).	NTH	Low
2.19	The system shall enable the Lecturer to generate exam keys (As PDF, As CSV).	MH	Low
2.20	The system shall enable the Lecturer to generate exam catalog documents.	MH	Low
2.21	The system shall enable the Lecturer to generate solved exams (for handing out to students post-exam).	MH	Low
2.22	The system shall enable the Lecturer to inspect changes made by course staff and take actions such as reject, accept, modify, or condition upon further validation.	MH	Low
3	**TAs**	-	-
3.1	The system shall enable TAs to add, delete, edit, and validate questions.	MH	Low
3.2	The system shall enable TAs to add, delete, edit, and validate stems.	MH	Low
3.3	The system shall enable TAs to add, delete, edit, and validate keys.	MH	Low
3.4	The system shall enable TAs to add, delete, edit, and validate distractors.	MH	Low
3.5	The system shall enable TAs to add, delete, edit, and validate solutions.	MH	Low
4	**WMS Activities**	-	-
4.1	The system shall allow users to log into the system using an ID and password.	MH	Low
4.2	Users should be able to search for a specific meta-question and edit it.	MH	Medium
4.3	The system shall allow users to ask to edit the current exam, in which case, the current exam settings give focus.	NTH	Low
4.4	Users can still write new questions, but those are now suggested to the Course Administrator rather than added to the exam.	NTH	Medium
4.5	Users can work on existing questions based on their role.	MH	Low
4.6	The system shall allow users to ask for a task, the system should offer tasks based on categories and sorted by urgency.	NTH	High
4.8	If a user is active in more than one course, users should be able to select the course.	NTH	Low

Chapter 4 – Non-functional Requirements

Implementation constraints

1. Performance (Speed, Capacity, Throughput, etc.)
    1. The system should be able to support multiple users (of the same or different types) using the system at the same time.
2. Reliability & Stability
    1. the distractors and answers as well as past exams will be persisted in a database and will support data recovery.
    2. In case of errors in internet connection, crashes, or hardware failure, the system will roll back all related updates until reaching the last stable version.
3. Safety & Security
    1. The system will save any sensitive data such as password only encrypted.
    2. The system will not allow any access to its sensitive data to users without permission.
4. Portability
    1. The system is web-based and can be accessed only while connected to the university's network.
    2. The system should be accessible from different browsers.
    3. The current requirement is to support the Hebrew language only.
5. Usability
    1. The system's users do not have any special expertise in computers or programming; therefore, the system's interface should be as simple and clear as possible.
6. Availability
    1. Unless the system is undergoing updates, the system should be available 24/7

Platform constraints

SE Project constraints

1. The system will be interactive, the inputs come naturally from the end users.
2. The system will need access to student exam answers to analyze questions/answers/distractors quality.
3. The system shall use samples of actual data in case of simulation.

Special restrictions & limitations

1. The system will be installed remotely on the university's servers.
2. The system will rely on the cooperation of end users, without it no data will flow in the system.

Chapter 5 – Risk Assessment & Plan for the Proof of Concept

Risk Assessment

Risk 1: Resistance from users to adapt to the new system.

People usually resist changes, especially if they are accustomed to existing systems. This resistance could result in a useless system, that’s due to the need of users to complete tasks.

Mitigation:

- Conduct training sessions to familiarize users with the new system.
- Communicate the benefits of the new system, emphasizing efficiency gains and improved functionalities.

Risk 2: Vulnerability to unauthorized access and leaking of sensitive information.

Given the nature of the system dealing with exams and sensitive data, there is a risk of unauthorized access leading to data leakage of sensitive information.

Mitigation:

- Implement security measures, including encryption and secure authentication protocols.
- Allow connection to the system only via the University network or/and using VPN.

Plan for the Proof of Concept

Research:

- **Examination of Web-Based Workflow Tools:**
  - Investigate popular web-based workflow management tools.
  - Assess their capabilities for role-based dashboards and task distribution.
- **Algorithmic Task Distribution:**
  - Research and develop a preliminary algorithm for task distribution.
- **User’s needs.**
  - Ask potential users for functionalities that may help with creating an exam
  - Ask potential users about what could be done to improve their experience with creating exams.
- **Security**
  - Investigate popular secure protocols that can be implemented in the system for authentication and data transfer.
  - Investigate how to use VPN and limit access to a server

Preliminary:

- **Selection of Web-Based Framework:**
    - Based on research, choose a web-based framework suitable for developing the workflow management system.
- **Basic UI Development:**
    - Implement basic UI features, such as role-based dashboards and content organization.
    - Integrate a simple manual task assignment mechanism.
- **Security:**
    - Implement basic security using JWT which will be created by the server.

Proof of Concept:

- **System Implementation:**
    - Develop a basic version of the system with the following components:
        1. **Task Distribution:**
            - Implement a simplified version of the algorithm for task distribution.
        2. **LaTeX-Based Exam Creation:**
            - Develop a module for flexible LaTeX-based exam, key, and solution creation.
        3. **Database Integration:**
            - Establish basic database integration for handling workflow and content data.
        4. **Security:**
            - Implement a simple server-client who will communicate using secure HTTPS protocol over the internet.
            - Authentication will be done using JWT which the server will create and send to the client after the authentication process.
        5. **User needs:**
            - Implement a simplified workflow management system that will be useful to use for managing a course.

Appendices

I/O Format Information

- **Exam Output**: LaTeX files for exams, keys, and solutions. These files can be compiled to generate PDF documents.
- **Database Format**: Structured in a relational database to store meta-questions and user data.

User Surveys

- **Purpose**: To gather feedback on the usability of the current system and expectations for the new system.
- **Participants**: Lecturers, TAs, and system administrators.
- **Findings**: Users highlighted the need for better collaboration tools, a more intuitive interface, and robust security features.

Background Information

- **Current System Limitations**: The existing system, written in Scheme, stores meta-questions in text files, which has led to scalability issues and difficulties in collaboration.
- **Proposed System Improvements**: The new system will be web-based, support multiple user roles, and offer a scalable database solution.

Glossary

- **Meta-Question**: An abstract structure from which multiple-choice questions can be derived.
- **WMS (Workflow Management System)**: A system that defines and manages the flow of work within an organization.
- **LaTeX**: A typesetting system commonly used for technical and scientific documentation.
- **TA (Teaching Assistant)**: A person who assists the lecturer in managing the course and creating exam materials.
- **System Administrator**: A person responsible for managing the installation, updates, and overall maintenance of the system.