

# Project Management and Development of Program-Level Accreditation

Ahmed Sameh

College of Computer and Information Sciences, Prince Sultan University, Riyadh KSA

**Abstract**—The NCAAA is the Quality Assessment arm of the Saudi Ministry of Higher Education (MOHE). We introduce a locally-developed Web-based Outcome-Directed "Program-Level" accreditation system that complies with the NCAAA program accreditation requirements. It automates the process of preparing all relevant accreditation documents, measurements, benchmarks, surveys, tests, statistics, forms, etc. And establish meaningful hyperlinks and structure among all relevant material. It also provides tools for performing important assessments functions such as PLO/CLO mapping, Rubrics direct assessment, surveys indirect assessment, trend analysis, etc. By running this system one can have access to "electronic" showroom that contains all needed forms and evidence materials. This automated system helps to "stream line" the whole accreditation process. It prevents current document and data inflation. It also slims down the process and provides an agile management system.

In this paper we describe the software engineering process used in development and deployment of this Web-Based system. We first describe the high-level system architecture. We then move to describe each subsystem in more details. The development team has been using a SCRUM methodology for both managing and developing the project. Comparison between RUP, XP, and SCRUM methodologies to be used in managing and developing the project and the team is presented. The SCRUM was chosen and its cycles are described. Release management shows how different versions are lunched and deployed. Demonstrating NCAAA compliance and guidance system is exemplified by the case of the "Information Systems" department at Prince Sultan University as a case study. In this context we describe how the SCRUM team measured the "Program Learning Outcomes", and injected guides at both the program and course levels. How assessment and guidance of these outcomes is done. How assessment tools are used in both cases. Describe both direct and indirect measurement tools deployment for both purposes. How to close the loop in assessment? We dig deep to the

course level and describe measurements, assessment, and guidance tools at their levels. We finally show how the system is used to track and strengthen weak areas and direct users' focus to these areas. In many cases the system recommends remedies for improvement.

In fact the accreditation system is a sort of a "Live" system that follows the continuous improvement slogan. In the sense that every semester new added documents, measurements, benchmarks, surveys, tests, statistics, forms, trend analysis, etc. are tallied to the system. At PSU this tracking and guidance system has been lunched as a full-fledged spin-off product. It is intended to be used by both internal and external programs within KSA. Both the Information Systems and Computer Science programs at Prince Sultan University (have recently been fully accredited for 7 years by the NCAAA) have used an early release of the proposed system.

**Keywords**— Assessment, RUP, OpenUP, XP, SCRUM, Rubrics, benchmarks

## I. INTRODUCTION

In fact, Web-Based applications are usually done "quick and dirty" for shorter development time. This leads to low quality, higher operation and maintenance costs. In this paper we have chosen to adapt existing modern software process models that fit such type of web applications. We consider both heavy, light weight and hybrid models. We also adapt known configuration and release management techniques for such type of applications. For example, parallel development of releases where small sub-teams work on different versions concurrently is very much needed in these types of applications with continuous reuse and integration.

In this paper we are comparing four software development processes (see figure 1): RUP/OpenUP, XP, SCRUM, and Hybrid that are potentially fitting the development of the E-Accreditation project. With every process we measure its suitability for the development of the proposed E-Accreditation system. We use qualitative measure for judging each requirement. Agile processes

are generally preferred for Web applications, but have two main obstacles: scalability and complexity, and their high demands on team members. Handling scalability and complexity occurs over the course of several projects. The Hybrid option can help manage such obstacles. We define it as a meta-process methodology. The methodology simply monitors the characteristics of the Web application in terms of project management parameters such as cost, time, quality, human resources, complexity, etc. And if the values of these parameters are on the low side, we initiate "lightweight" pure agile processes, mid-range, we perform a "transition phase". If the values of the parameters are on the high values, we initiate "heavyweight" planned-based processes. So for low complexity we initiate agile processes, and for high complexity we initiate heavyweight processes.

Web projects are different from traditional projects in many ways. For example traditional projects stress at: quality product at lowest cost whereas the web ones target usable system in shortest time. Team member sizes for web projects are much smaller than traditional projects. Web project duration is smaller in horizon with costs in thousands. Traditional large project lend themselves to planned-based more than agile based, while web based projects are the opposite. Traditional projects commonly use "Object Oriented" methods, while web projects rely on component-based and ad-hoc processes. Standard projects rely on rigid processes, while web projects rely on standard high usability.

Assessment of curriculum outcomes represents a critical stage of any degree program evaluation to support continuous academic improvement. The collection, aggregation and analysis of assessment data are notoriously complex and time-consuming processes. Several actors are involved across the assessment workflows, which we propose to streamline in the context of a portal-based approach. We first describe a model for capturing and propagating assessment data across the assessment cycle as well as tracking potential deficiencies in the assessed program. We then describe our portal-based implementation of the proposed model in the context of an ACM-based IS Curriculum program at PSU which assessment processes respond to NCAAA accreditation requirements. We reveal the results of our assessment experience using the proposed model and the corresponding in-house developed portal. We show the corrective curricular actions facilitated by our assessment tool to close the assessment cycle.

Most previous similar systems focus on how to assess the PLOs, and little has concerned how to equip students with the skills and attitudes specified in those outcomes. Although it is mainly an "assessment" system, but it can work in "reverse" and allow equipping students with the learning outcomes at both course and program levels (CLOs, and PLOs) instead of just measuring them. The proposed system can work in "reverse" to provide guidance and advice on how to achieve compliance. It provides guidance and advice for students, instructors, and administrators of the program to be accredited. This guidance and advice tasks go hand in hand with the assessment tasks such that at every activity level of uploading files, conducting a survey, collecting statistics, running a tool, filling in a form, etc. there is guidance and recommendation on how to proceed, informing about missing items, recommending an assessment tool, diagnosing certain measurements, suggesting helpful hyperlinks, etc.

In [2] we focus on the "Guidance" part of the system. The proposed system can work in "reverse" to provide guidance and advice on how to achieve compliance. It provides guidance and advice for students, instructors, and administrators of the program to be accredited. This is provided through explanation hyperlinks when the end-user hover above any bold italic phrase or under-lined word or phrase. Guidance hyper messages are displayed anywhere there is either "***Bold Italic***" text or any "Under Lined" text. They are displayed in a drop down window and they come in four different colors: Black, Green, Blue, and Red. Examples of guidance hyper messages that elaborates on any ***Bold-Italic*** text when an end-user hover on top of it by displaying the beneath text, and within the text when hovering on top the underlined words.



Fig. 1 Agile Software Development Survey [1]

In a previous paper [2] the guidance activities of the proposed system were demonstrated. The overall description of the proposed system can be found in [1]. The structure of this paper goes as follows: Section 2 describe the architecture of the proposed system, section 3 describes the project management decisions, and compares between RUP, OpenUP, XP, and SCRUM methodologies and their suitability in such project. Section 4 describes the software development process used and displays snap shots of the implemented system in section 5. Section 6 presents training end-users and section 7 presents deployment followed by conclusion.

II. SYSTEM ARCHITECTURE

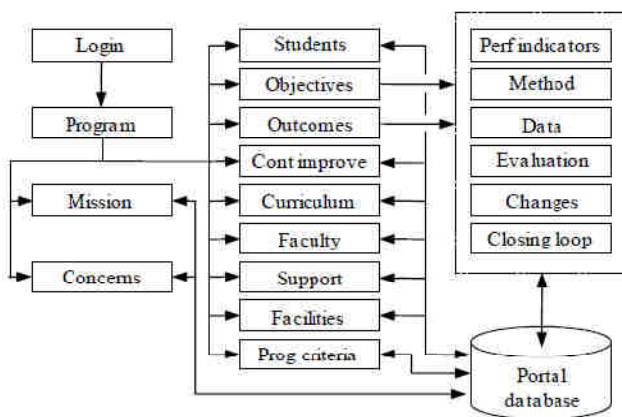


Fig. 2 The high level Logical architecture of the E-Assessment system.

Figure 2 shows the high level architecture of the proposed system. This automated system helps to "stream line" the whole accreditation process. It prevents current document and data inflation. It also slims down the process and

provides an agile management system. Four types of users are allowed into the system: Administrators, Higher-Managers, teachers, and inspectors. Each has a set of allowed permissions. The "Login" is the front-end of the system that displays many tabs such as: "Table of Contents", "Steering Committee", "Institutional Overview", "Introduction", "Standards", "Program Description", "Mission and Vision", "Concerns", "Program Governance", etc. Pressing on any tab will display details similar to the "Standards" tab shows you the eleven standards of NCAAA accreditation.

The Portal database is the backend of the system. Web pages and related documents (of all types) are classified into the categories as shown in figure 2. Each category is again divided into tabs for further classification of the related material. Each tab has facilities to upload documents, measurements, benchmarks, surveys, test, statistics, forms, etc. It also has facilities to display and edit such material in various formats. Establishing structured hyperlinks among the tabs and material is a powerful feature of the system. Each tab reminds the end-user of the relevant accreditation material needed for such tab, and displays "Green" and "Red" indicators that show "compliance" or "non-compliance".

In fact the backend database is a sort of a "Live" archive that follows the continuous improvement slogan. In the sense that every semester new added documents, measurements, benchmarks, surveys, tests, statistics, forms, trend analysis, etc. are tallied to the system. The system has both a tracking and guidance functions that are lunched at all levels.

The idea of "reminding" end users of the accreditation body requirements is part of the "Guidance" function of the system. This is a feature that allows the system to work in "reverse" for guidance and advice such as to equip students with learning outcomes instead of just measuring their degree of compliance. In this context, it identifies and describes instructional techniques and methods that should effectively prepare students at both the course and program levels to achieve target learning outcomes. Another reverse example is to equip and guide instructors on how to achieve NCAAA compliance by identifying missing documents, measurements, teaching methods, benchmarks, surveys, tests, statistics, forms, etc. Suggesting meaningful hyperlinks, tools to perform important assessment functions, etc. can also be considered an important guidance function. Further details of the "Guidance and Advice" function of the system can be found in [2].

The system also provides tools for performing important assessments functions such as PLO/CLO mapping (Program Learning and Course Learning Outcomes), Rubrics direct assessment, surveys indirect assessment, trend analysis, Key Performance Indicators, etc.

This automated system is designed to streamline faculty accreditation responsibilities such as compilation, management, analysis, tracking, benchmarking, and reporting activities, while meeting accreditation standards. It also provides desktop access for chairs, deans, and others responsible for accreditation. NCAAA accreditation team does not have to visit a program showroom to inspect physical evidence and documents. By running the E-Accreditation system the NCAAA team can have access to an "Electronic" showroom that contains all needed materials.

Figure 3 shows the Client-Server model of the proposed Web-Based On-Line system. The physical structure at both the client and the server sides are shown in addition to the networking link between the two. The Client side is thin and almost all the functionalities are developed and provided at the server side.

**III. PROJECT MANAGEMENT DECISIONS**

Figure 4 shows the configuration management setup of the development team of this project where 4 servers (environments) are shared among development team, testers, quality assurance, and pre-production/production. The software team is made up of 7 members. All of them are above average .NET programmers. The system repository contains all "configuration Items" related to the project distributed over the four environments. This include source codes, data, forms, test-cases, documents, assessment tools, etc.

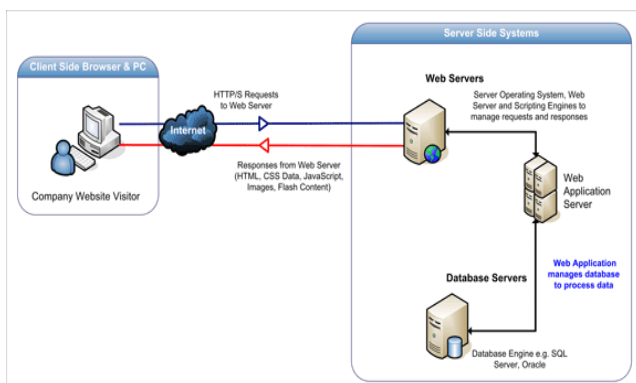


Fig. 3 E-Accreditation Web Application Physical Architecture

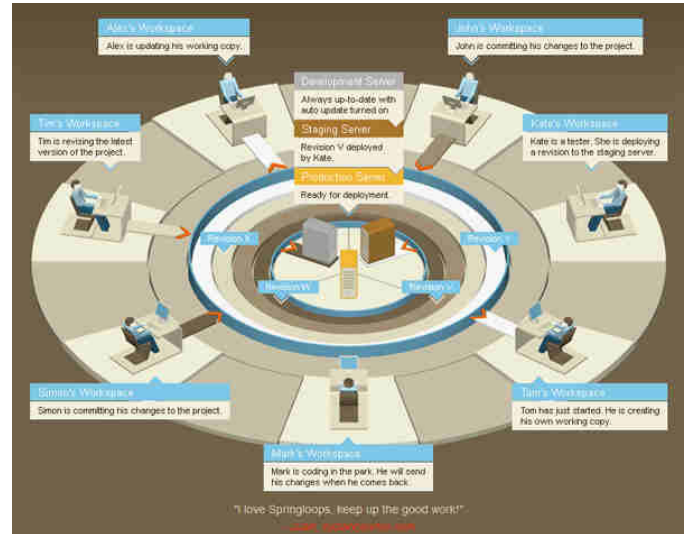


Fig. 4 Development, Testing, and Production Environments

The team has been using the V-Model as shown in figure 5, where flow goes one side is analysis and decomposition, and the bottom edge of the V is implementation, where the other side of the V is for testing, quality assurance and deployment.

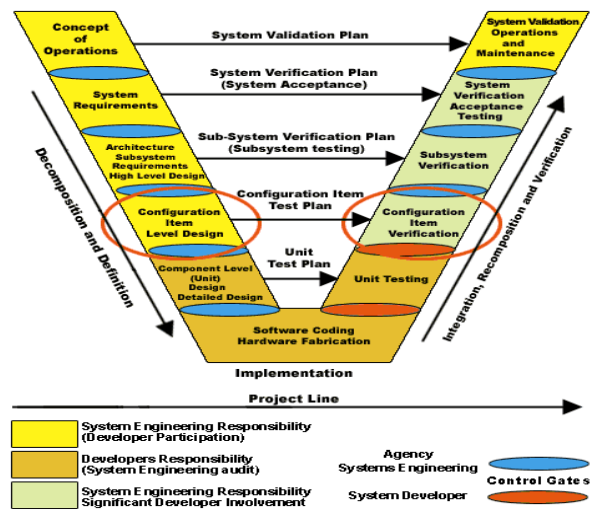


Fig. 5 Software Life Cycle

Promotion policies for developers, testers, quality assurance, and pre-production/production have been set by the project leader and system stakeholders. Examples of some of these policies are: - For developers to promote their codes from the development environment to the testing environment they should have a compiled code that has been tested at least using 20 test cases. Another policy for quality assurance team members to promote their codes to pre-production/production environment they should have fulfilled a complete test plan that satisfies a test completeness criteria. At least 300 test

cases should have been passed successfully before promotion of a new release. Releases are sent for deployment for end-users (e.g faculties) at their production environments.

A Software Process Development tool has been used to store "User Stories" and track their progress from "To-Do" to "In-Progress" to "Completed". It is also used to generate "test cases" to be used for testing and quality assurance. The tool have also provided for Continuous integration, and source code management.

Various testing methodologies and tools have been used for: Unit testing, integration testing, static and dynamic analysis, risk-based testing, regression testing, acceptance testing, and maintenance testing. All the above has helped in increasing the productivity of the team and improving the quality of the final product.

As for the software engineering development and project management three approaches were investigated: OpenUP [3], SCRUM [4], and Extreme Programming [5]. The purpose was to select the best approach that can handle the management of this Web-Based system, efficiently steer both the development team and the process itself, and deal effectively with project's risks.

Many projects (specially the web-based ones) are done "Quick and Dirty" for shorter development time, but the consequences are low quality, and higher operation and maintenance costs. It was found that for medium and large size Web projects, one should adapt existing agile software project management and development models. Agile models are light weight models and can still serve the first objective which is fast delivery. Most Agile models are goal-oriented and are composed of a set of distinct iterations that results in software releases. Some steps may occur several times and reuse accumulated knowledge. They have short expected duration time, can handle dynamic restructuring of data, accommodate evolving technologies and standards, have strong customer involvement, disposable releases to demonstrate functionality, small teams working on different versions of the system concurrently, and strong emphasis on communication.

The proposed E-Accreditation system has a number of characteristics which are: Medium size project, changing requirements, risk of faculty non-participation, large heterogeneous data types and formats, large data sets, dynamic backend database, several add-on tools, strict NCAAA deadlines, and complex interactions with several parties (e.g. data suppliers, and benchmarking).

The development team has been experimenting with the above three Agile methodologies for both managing and developing the project. For each methodology a suitable tool has been chosen to investigate the suitability of the method for the target E-Accreditation project. The tool was used to analyze project's cycles, and investigate release management and how different versions are lunched and deployed. Tracking NCAAA compliance is exemplified by the case of the "Information Systems" department at Prince Sultan University as a case study. In this context we make use of the chosen tool to describe in terms to tasks: how "Learning Outcomes" are defined at both the program and course levels. How assessment of these outcomes is done. How assessment tools are developed and used. Describe both direct and indirect measurement tools development and deployment. How to close the loop in assessment? How to create curriculum map? How to define Key Performance Indicators (KPIs) and how to benchmark them with internal and external similar programs? We dig deep to the course level and describe measurements and assessment tools at their levels. We finally test the tool to show how the system is used to track weak areas and direct users' focus to these areas. In some cases the tool is tested to see how the system might recommend remedies for improvement.

The purpose is to select the best approach that can handle the management of this E-Accreditation Web-Based system, efficiently steer both the development team and the process itself, and deal effectively with project's risks. The result is a table of scores for each of the three agile methodologies.

**3.1 RUP/OpenUP**

Rational Unified Process (RUP) is a heavyweight process framework. It is phase-oriented, incremental, and iterative. RUP is designed for high-complexity, high-quality systems. RUP methods are grouped into core workflows (or "disciplines"- See figure 7). Key principles behind RUP are: adapt the process, balance stakeholder priorities, demonstrate value iteratively, encourage abstraction, and focus continuously on quality.

RUP Phase/ System Requirements	Suitability	Comments
1-Inception	Poor	-Assumptions may change as the E-Accreditation project progresses
2-Elaboration	Poor	-Developing suitable system outweighs measuring price -Internet largely defines

		system architecture
3-Construction	Good	-Web Based Languages
4-Transition	Good	-In some cases easier because distribution is automatic
5-Short development cycles	Poor	-Large overhead
6-Changing requirements	Poor	-Planned Based
7-Fixed deadlines, Flexible content	Poor	-Difficult to control
8-Parallel development	Poor	-No
9-Reuse	Good	-Due to global planning
10-Integration	Poor	-Difficult to control
11-Adapting to flexibility level	Good	-Due to heavyweight

Table 1: RUP's Suitability for E-Accreditation System

Table 1 shows the suitability of RUP for the E-Accreditation Web-Based system. Most of the poor performance is due to the heavy weight nature of RUP. OpenUp [3] is the Agile version of RUP (see figure 6). Both OpenUp and RUP are tailor-able and they both run under the Eclipse Process Framework (EPF). OpenUp is free while RUP is not. Agile RUP is a previous agile version of RUP before OpenUp but both of them are complete and well defined. OpenUp has only 17 work products while RUP has 76 (AKA Artifacts). Figure 6 shows the 3 layers of OpenUp: Stakeholder focus- Project lifecycle, Demo-able or shippable build team focus- iteration lifecycle, and micro-increment –personal focus cycle.

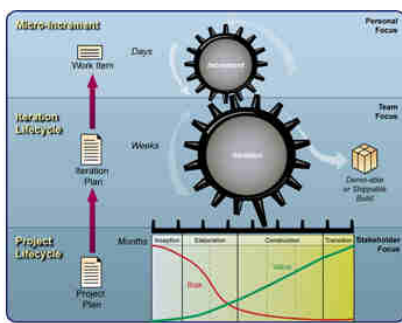


Fig. 6 OpenUp three Layers



Fig 7. Four Phases of RUP/OpenUP



Fig. 8 OpenUP Client Interface I

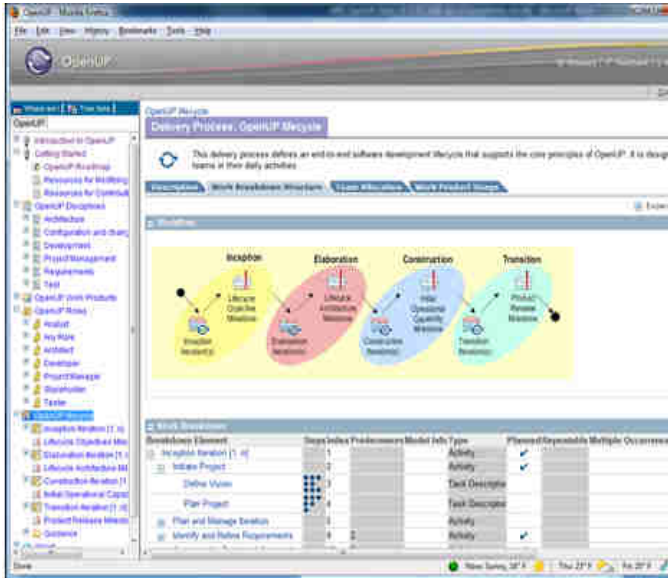


Fig. 9 OpenUp Client Interface II

An OpenUp Client [3] was downloaded from their site, installed and fed with work item list from the IS E-Accreditation project. A sample list of the work items, phases, team member roles, inputs, outputs can be found in figures 8 and 9 above.

The development team has been experimenting with the above for both managing and developing the IS E-Accreditation project. The purpose is to investigate the suitability of the method for the target IS E-Accreditation project. The OpenUp Client tool was used to analyze project's cycles, and investigate release management and how different versions are launched and deployed. In this context we make use of the chosen tool to describe in terms to tasks: how "Learning Outcomes" are defined at both the program and course levels. How assessment of these outcomes is done. How assessment tools are developed and used. Describe both direct and indirect measurement tools development and deployment. How to close the loop in assessment? How to create curriculum map? How to define Key Performance Indicators (KPIs) and how to benchmark them with internal and external similar programs? We dig deep to the course level and describe measurements and assessment tools at their levels. We finally test the tool to show how the system is used to track weak areas and direct users' focus to these areas. In some cases the tool is tested to see how the system might recommend remedies for improvement.

The purpose was to evaluate the OpenUP approach on how to handle the management of the E-Accreditation Web-Based system, efficiently steer both the development

team and the process itself, and deal effectively with project's risks. The result is a table of scores for each KPI for the suitability. Figures 6-9 show the IS E-Accreditation team work experience with OpenUp. Table 1 scores of RUP would improve slightly with OpenUp, specially with items 3,4,5,6,8,and 10.

3.2 SCRUM

As for the SCRUM software engineering development and project management model figures 10 and 11 show the model structure. A SCRUM-based tool called "Exo-Soft Scrum" [5] has been tested for support and logging of all the activities of the team- see figures 12 and 13. SCRUM is an agile model this is light weight and has proven its capabilities lately. Table 2 shows the ratings of the E-Accreditation system suitability if we use SCRUM as a software development and project management methodology.

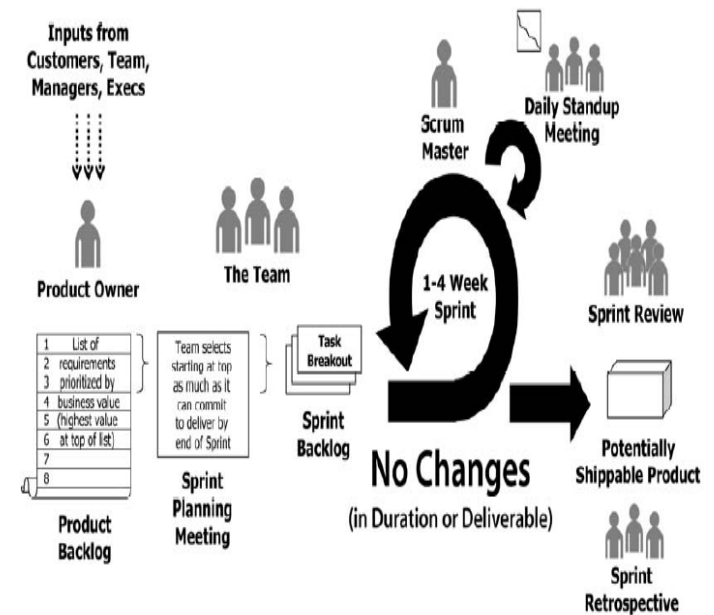


Fig. 10 SCRUM methodology Activities

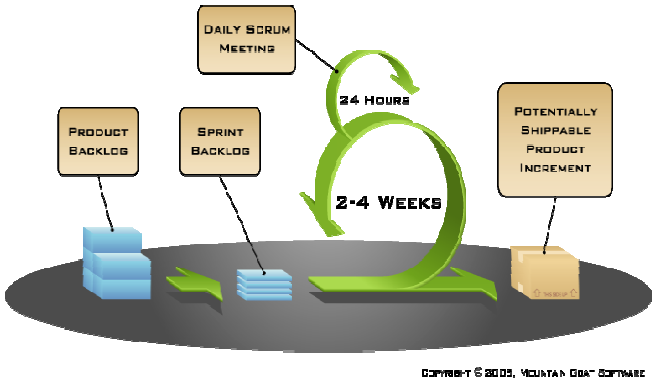


Fig. 11 SCRUM Phases and Iterations

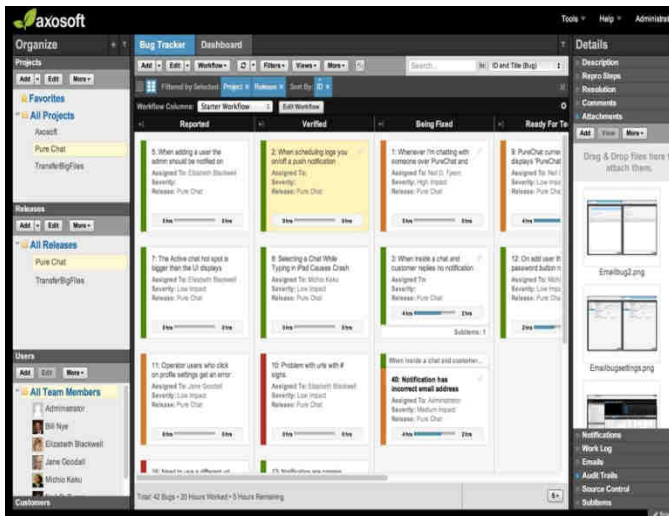


Fig. 12 Axo-Soft SCRUM Support Tool with E-Accreditation Tasks

E-Accreditation/SCRUM Requirements	Suitability	Comments
1-Compling Product Backlog	Good	-Assumptions may change as the E-Accreditation project progresses
2-Iteration Planning	Good	-Internet largely defines system architecture
3-Iterations Management	Good	-Tight control
4-Iterations Review	Good	In some cases easier because distribution is automatic
5-Short development cycles	Good	-Elaborate process
6-Changing requirements	Good	-Due to its agile nature
7-Fixed deadlines, Flexible content	Good	-We-Based
8-Parallel development	Good	-Due to its agile nature
9-Reuse	Good	-Refactoring
10-Integration	Good	-Tight control
11-Adapting to flexibility level	Good	-Due to its agile nature

Table 2: SCRUM's Suitability for E-Accreditation System

### 3.3 Extreme Programming (XP)

Extreme Programming (XP) is one of the most popular forms of agile processes. It is iterative, test-driven, more human-centric /feedback-oriented. Figure 14 shows the rapid successive releases that are produced after every iteration. Figure 15 shows the iteration view of the process. Figure 16 shows the phase view of the XP methodology with the following phases: Exploration, Planning, Iterations to Releases, Production, and Maintenance phases. . Table 3 shows the scores of the E-Accreditation system if we apply the XP methodology. The table shows that XP is a good fit when used in developing the E-Accreditation system due to the following features: short development cycles, coping with changing requirements, flexibility with respect to deadlines and contents, being able to perform parallel development. On the negative side, it does not allow for reuse and continuous integration, and does not adapt to high levels of complexity. Figure 17 shows a screenshot of the "XPlanner" tool that is commonly used to support XP teams. The tool was investigated and tested for its



Fig. 13 Axo-Soft SCRUM Tool Dashboard with E-Accreditation Progress



support to the team for developing the target E-Accreditation system.

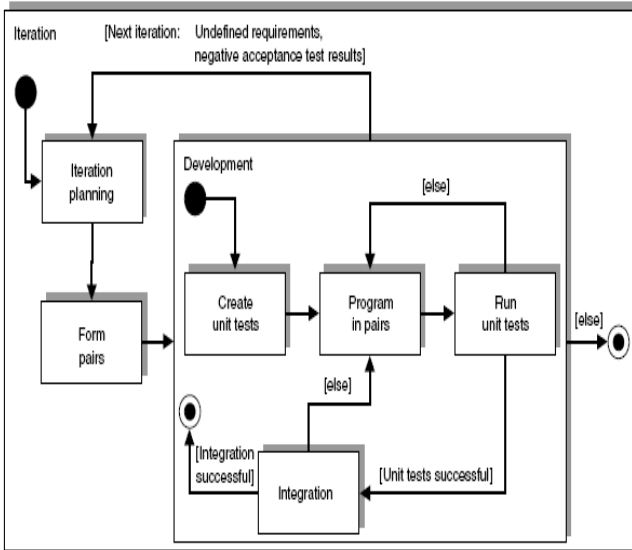


Fig. 14 XP Iteration View with Releases

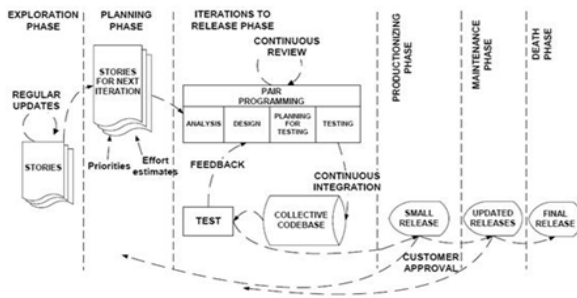


Fig. 15: XP Six Phases of Development and Maintenance

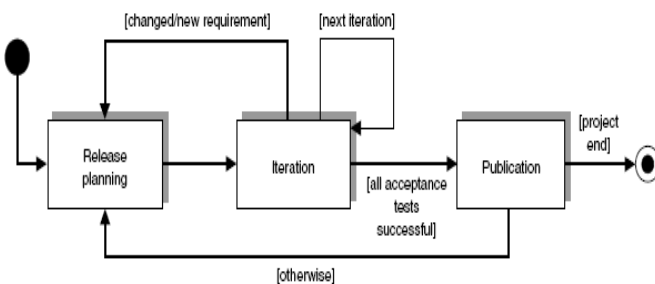


Fig. 16 XP Iterations Structure

		progresses
2-Requirements Planning	Good	-Developing suitable system outweighs measuring price -Internet largely defines system architecture
3-Iteration and Release Management	Good	-Iterations are elaborated
4-Production Management	Fair	In some cases easier because distribution is automatic
5-Short development cycles	Good	- Short iterations
6-Changing requirements	Good	-Due to its agile nature
7-Fixed deadlines, Flexible content	Poor	-No global planning
8-Parallel development	Fair	-Concurrent Versions are possible
9-Reuse	Good	-Refactoring
10-Integration	Good	-Due to its agile nature
11-Adapting to flexibility level	Good	-

Table 3: XP's Suitability for E-Accreditation System

XP/ E-Accreditation Requirements	Suitability	Comments
1- Requirement Exploration	Good	-Assumptions may change as the E-Accreditation project

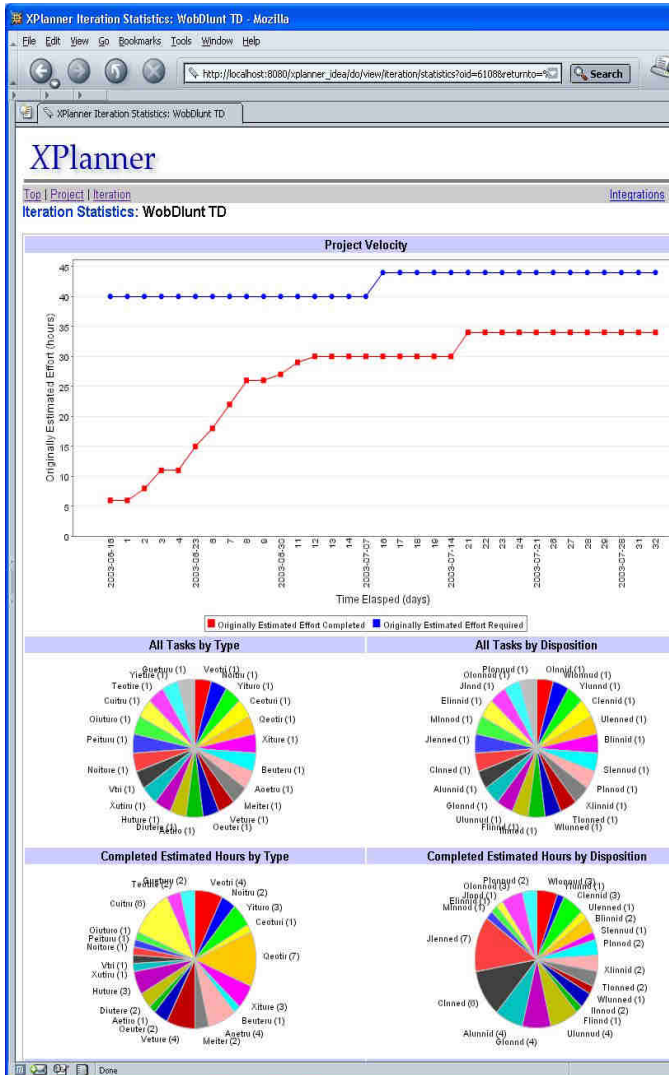


Fig. 17 Screenshot of XPlanner XP Support Tool

### 3.4 Meta Process

Web projects are different from traditional projects in many ways. For example traditional projects stress at: quality product at lowest cost whereas the web ones target usable system in shortest time. Team member size for web projects are much smaller than traditional projects. Web project duration is smaller in horizon with costs in thousands. Traditional large project lend themselves to planned-based more than agile based, while web based projects are the opposite. Traditional projects commonly use "Object Oriented" methods, while web projects rely on component-based and ad-hoc processes. Standard projects rely on rigid processes, while web projects rely on standard high usability. A mid-size web project like the current E-Accreditation system have chosen to adapt existing modern software process models that fit such

type of web applications. We consider both heavy, light weight and hybrid models. The current "Meta-Process" is a hybrid process (see figure 18). We use some target system characteristics such as number of users, size of the development team, system cost, etc. to identify the complexity of the system and based on that we chose one of the four software development processes: RUP/OpenUP, XP, SCRUM, and Hybrid that are potentially fitting the development of the E-Accreditation project. With every process we measure its suitability for the development of the proposed E-Accreditation system. We use qualitative measure for judging each requirement. Agile processes are generally preferred for Web applications, but have two main obstacles: scalability and complexity, and their high demands on team members. Handling scalability and complexity occurs over the course of several projects. The Hybrid option can help manage such obstacles. We define it as a meta-process methodology. The methodology simply monitors the characteristics of the Web application in terms of project management parameters such as cost, time, quality, human resources, complexity, etc. And if the values of these parameters are on the low side, we initiate "lightweight" pure agile processes, mid-range, we perform a "transition phase". If the values of the parameters are on the high values, we initiate "heavyweight" planned-based processes. So for low complexity we initiate agile processes, and for high complexity we initiate heavyweight processes.

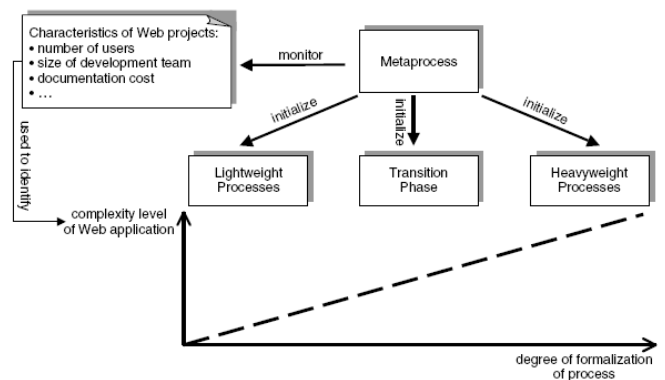


Fig. 18 The Meta Process Strategy for Web-Based Applications

## IV. SOFTWARE DEVELOPMENT PROCESS DECISIONS

After studying the three above candidates approaches; a SCRUM based model has been adapted, and "Exo-Soft

SCRUM" tool was chosen for team and project's support. Figures 19-22 show screenshots from the tool. All team members were aware of the SCRUM model and had used it before this project. The tool "Exo Soft-Scrum" has been used to support and log all the activities of the team. Promotion policies for developers, testers, quality assurance, and pre-production/production have been set by the SCRUM master and system stakeholders. Examples of some of these policies are: - For developers to promote their codes from the development environment to the testing environment they should have an error-free compiled code that has been tested at least using 20 test cases. Another policy for quality assurance team members to promote their codes to pre-production/production environment they should have fulfilled a complete test plan that satisfies a test completeness criteria. At least 300 test cases should have been passed successfully before promotion of a new release. Releases are sent for deployment for end-users (e.g faculties) at their production environments.

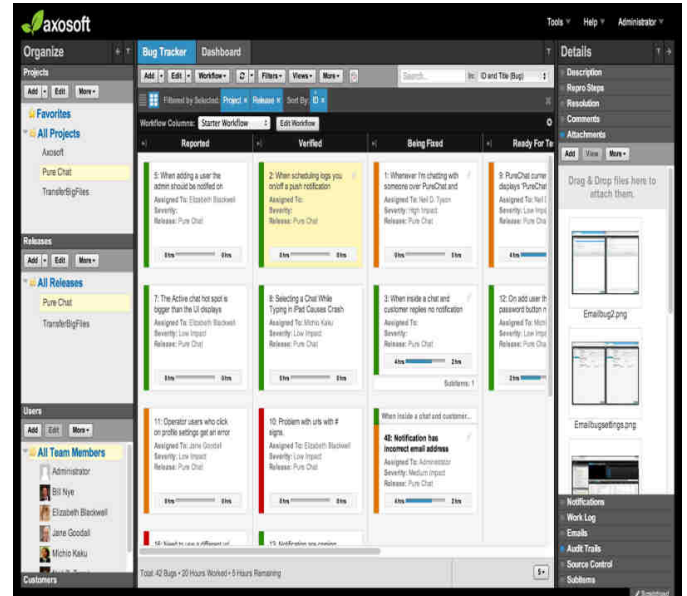


Fig. 20 Axo-Soft SCRUM Tool E-Accreditation Tasks Resource Assignment

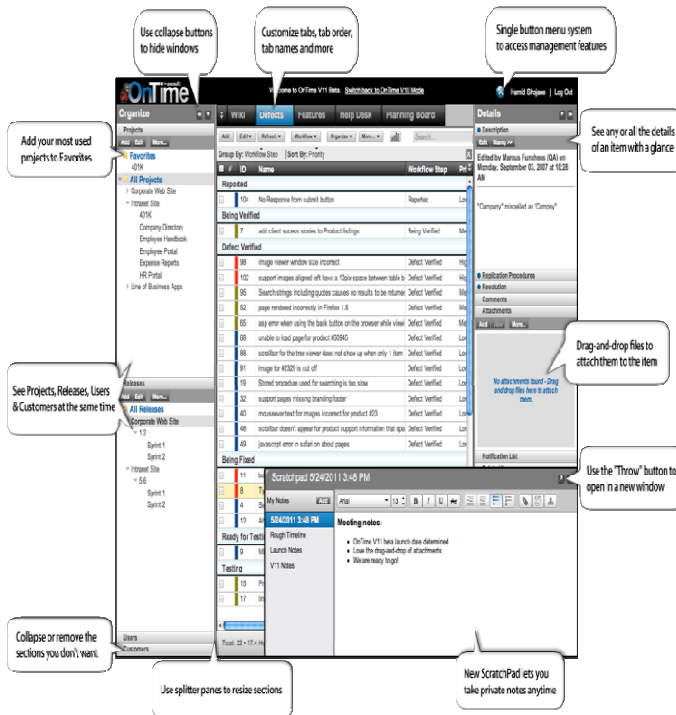


Fig. 19 Axo-Soft SCRUM Tool GUI Layout

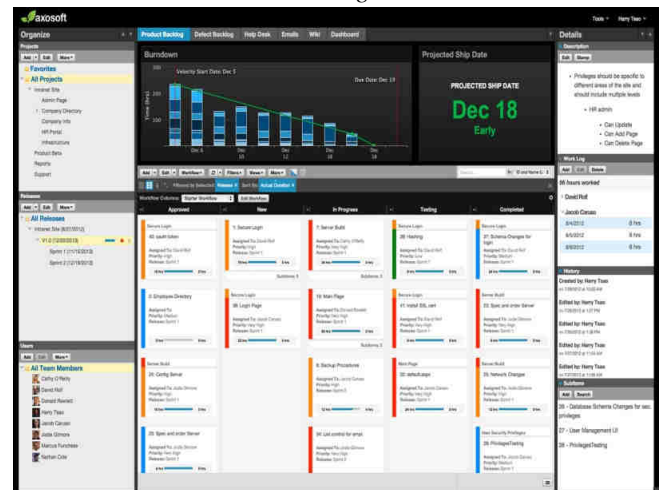


Fig. 21 Axo-Soft SCRUM Tool E-Accreditation Tasks Pipeline Progress



Fig. 22 Axo-Soft SCRUM Tool E-Accreditation Tasks Dashboards

The "Exo-SCRUM" tool has been used to store "User Stories" and track their progress from "To-Do" to "In-Progress" to "Completed". It is also used to generate "test cases" to be used for testing and quality assurance. The tool has also provided for Continuous integration, and source code management. Various testing methodologies within the tool have been used for: Unit testing, integration testing, static and dynamic analysis, risk-based testing, regression testing, acceptance testing, and maintenance testing. The above figures (19-22) show how to apply the E-Accreditation of the IS program: Define stories, tasks, show some charts from the tool. Show the dashboard of the tool. Show details to evidence that the tool and the SCRUM have been using in this project. (Mostly charts from the ExoSoft-SCRUM tool). All the above has helped in increasing the productivity of the team and improving the quality of the final product.

V. SYSTEM IMPLEMENTATION AND TESTING

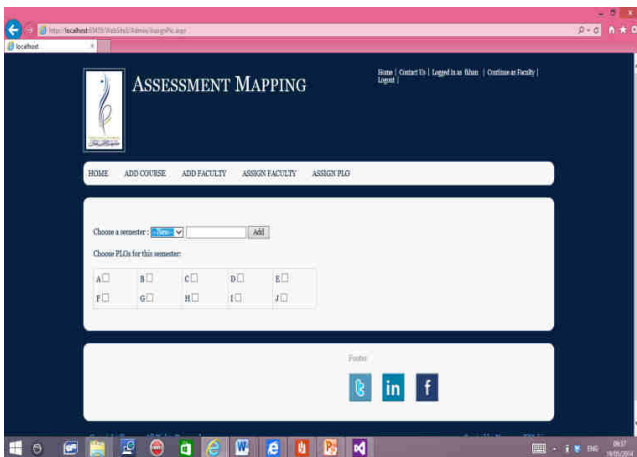


Fig. 23 Administrator can choose the PLOs to assess this semester

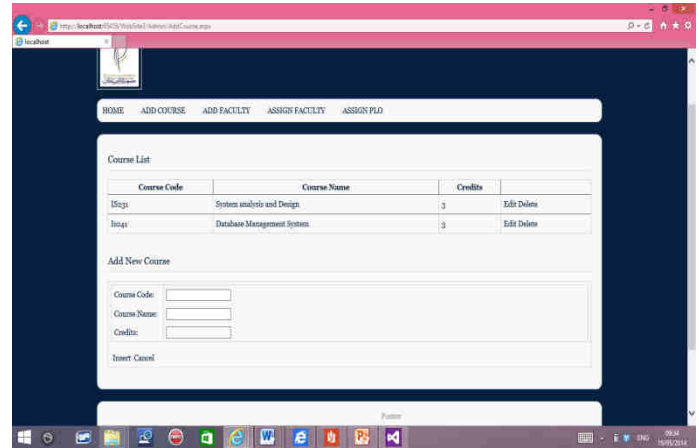


Fig. 24 Administrator can add, edit, delete courses and instructors

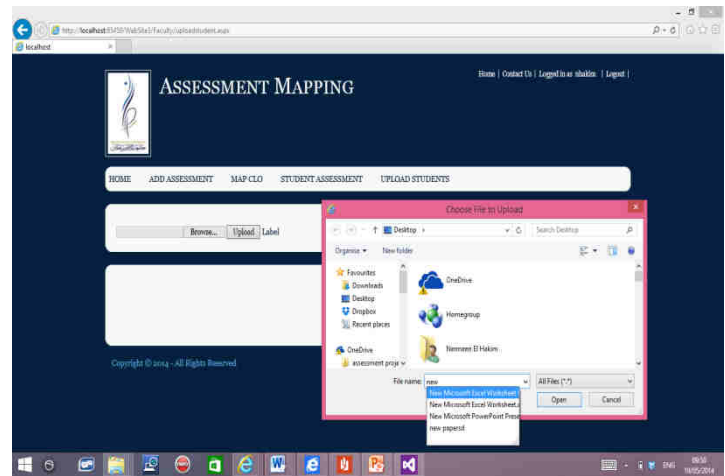


Fig. 25 Faculty has ability to load student lists from Excel

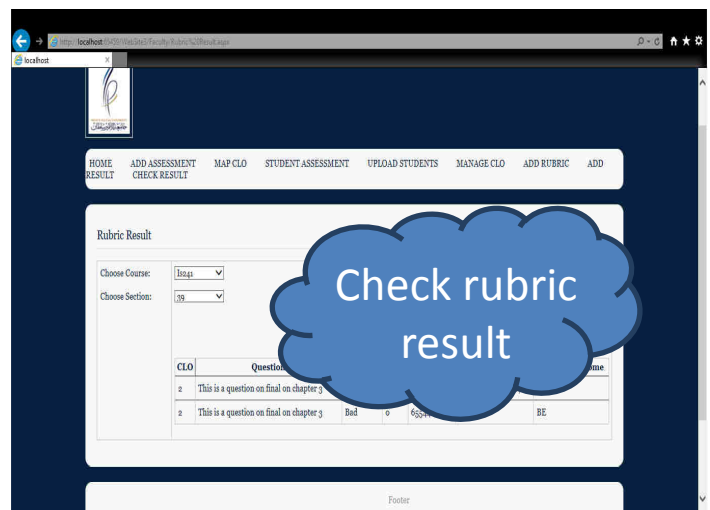


Fig. 26 Faculty is checking his Rubric Results

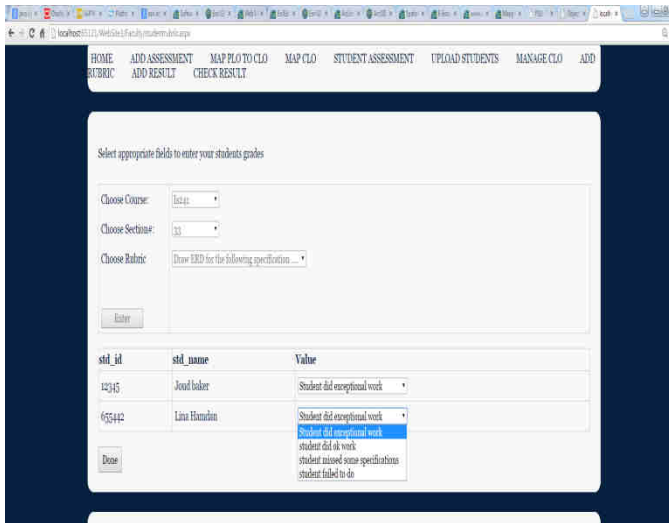


Fig. 27 Faculty add Rubric results to each student

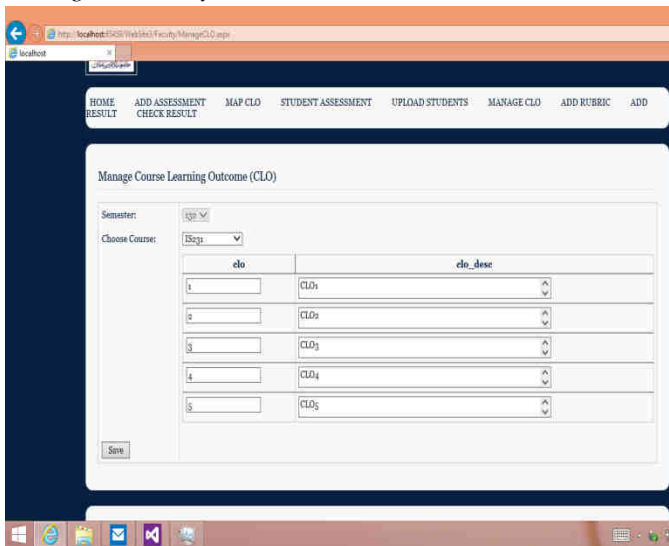


Fig 28 Faculty managing their course CLOs

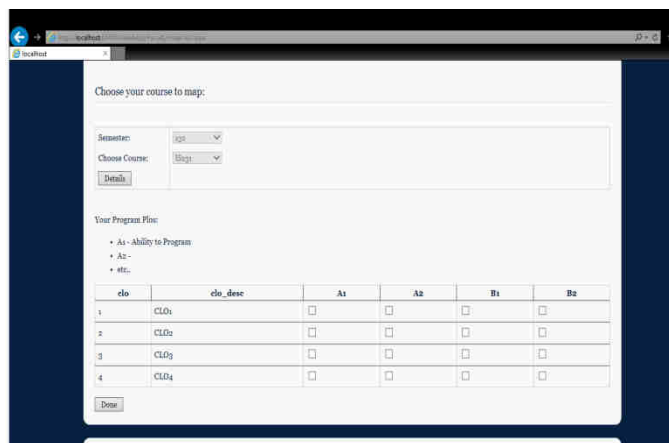


Fig. 29 Faculty can map their CLOs to the target PLOs

Figure 23-29 provide a number of screenshots from the actual system. They show how administrators, faculties, and students can upload documents (in Word, PDF, Excel formats) and/or edit text of comments, concerns, guidance, and remarks at any stage. These documents represent evidences that the end-user would view by clicking on their links. The figures also show sample assessment loop for PLO number A. The assessment loop consists of performance indicators KPI#5, and KPI#9, assessment method (Excel Rubric sheet), assessment data (rubric scores), assessment evaluation (rubric results), change made (a PDF file), and finally the closing of the loop (an action plan Excel file). The screenshots also show how administrators choose target PLO(s) for a specific semester. How they can add/edit/delete courses and faculty. How to upload student lists for specific courses. How rubrics are managed in the system. And how faculty can manage their courses' CLOs and map them to the target PLOs.

Microsoft IIS web server and Microsoft SQL database were used. ASP.NET framework was used for server side programming. As shown in the system architecture section above, four main end-users were identified: teacher, administrator, students, and accreditor/visitor. Documents and hyperlinked information are pulled from the database based on end-user credentials. SurveyMonkey [6] was used as a survey tool to run/collect/analyze surveys. Hyperlinks are used everywhere; for example a visitor/teacher/administrator can click on program learning outcomes with assessment links, KPI links, SSR links, CLO links, Assessment data links, Evaluation links, Benchmarks links, Changes links, Closing the loop links, etc. (PLO-Centric Ontology was used [2]). Each of these links takes the user to a page where material can be uploaded and/or viewed for each assessment step. Hyperlinks also lead to depository for material for NCAAA criteria not directly connected to assessment, such as students, curriculum, faculty, facilities, institution, program, courses, etc. Each of these links takes the user to a page where material related to the criterion can be uploaded and viewed. All the above pages contain links to "summary information" where auto-generated summaries are generated from the original documents

## VI. TRAINING END-USERS

All faculty members of both College of Computer and Information Science and the College of Business

Administration were given full access to release 0.1 then 0.2 of the system. Training sessions were organized to train all faculties at the two schools. A number of "Training Videos" were prepared for the TLC (Teaching and Learning center) to demonstrate these two Beta releases to all PSU faculty members. Selected faculty members from both sides of the PSU campus (male and female) were chosen for using the releases and provided their feedback. The system was also tested by Business faculty in a CBA mock review: Release 0.2 was used in this trial. Again feedback was collected from the end users. Work in [2] describes the methodology used to document the system structure and its subsystems for easier future maintenance and upgrade.

## VII. DEPLOYMENT

Release 0.1 has been deployed at the "Information Systems" department and all faculty members were trained on using the system to automate the process of preparing all relevant program accreditation documents, measurements, benchmarks, surveys, tests, statistics, forms, KPI, etc. They were also trained on how to establish meaningful hyperlinks, guidance, and structure among all relevant material and forms. The deployed system also provided tools for performing important assessments functions such as PLO/CLO mapping (Program Learning and Course Learning Outcomes), Rubrics direct assessment, surveys indirect assessment, trend analysis, Key Performance Indicators, etc. Faculties stated that the automated system has streamlined their assessment responsibilities such as compilation, management, analysis, tracking, benchmarking, and reporting activities, while meeting accreditation standards. It also provided desktop access for chairs, deans, and others responsible for accreditation. Students also expressed their satisfaction with the guidance component of the system. NCAAAA accreditation team has also expressed their contentment with the system. Release 0.2 has been deployed at the School of Business Administration during a "Mock Review" in preparation for the NCAAAA site visit. CBA faculties were shown how to run the system and how to take advantage of it during the rest of their preparations and during the actual NCAAAA visit.

## VIII. CONCLUSION

The choice of the software engineering process of the E-Accreditation system is presented. Management, design, development, and testing decisions are elaborated.

SCRUM agile methodology is shown to be the best suitable for the target system. Comparison of suitability with three other methodologies; RUP/OpenUp, XP, and Hybrid is presented. Agile support tools for these methodologies are also compared and investigated. SCRUM and its Exo-Soft tools are elaborated upon. Screenshots of the target system are shown along with detailed explanation of the development, management, testing, and deployment processes.

## REFERENCES

- [1] Ahmed Sameh, "Overview of the E-Accreditation Project at Prince Sultan University", Working paper
- [2] <http://www.axosoft.com/>
- [3] "SurveyMonkey," 2011. URL: <http://www.surveymonkey.com>.
- [4] Ahmed Sameh, "Two-Way Program-Level Accreditation System: Compliance and Guidance", Working paper.