Journal of Engineering Education Transformations,
Volume No. 37, January 2024 Special Issue, eISSN 2394-1707

Integrated Project Development through Combined Theory and Practices of Core Courses focusing on Software Development Skills: Integrated Learning Framework

Dr. Padmashree Desai[1], Dr. P.G Sunitha Hiremath[2]
[1,2] School of Computer Science & Engineering, KLE Technological University, Hubballi-580031, India
[1]padmashri@kletech.ac.in
[2]pgshirmath@kletech.ac.in

***Abstract*—The National Education Policy promotes moving from the conventional content-heavy and memorization learning practice towards holistic learning/integrated learning. It imparts a creative and multidisciplinary curriculum that focuses equally on curriculum and assessment. All educational establishments assess students using written examinations, quizzes, seminars, term paper writing, and course projects. A semester typically includes 4-5 courses, and students must earn credits for these courses by scoring a good Cumulative Grade Point Average (CGPA). As course projects offer depth knowledge/holistic learning/lifelong learning of a course for the student, many courses include course projects as one of the activities in the course. If all the courses are intended to include course projects as a mandatory pedagogy, it will be difficult for students to acquire in-depth knowledge and required skills while also dealing with stress. So we are proposing an integrated learning framework by applying the theory and practices of two core courses- Software Engineering and Web Technologies to develop a web application. This integrated learning focuses on developing software development and software testing skills in computer science for undergraduate students pursuing a bachelor of engineering degree. This framework alleviated the pressure on students during placement and created job opportunities in software development.**

The framework consists of three important phases- The first phase includes the identification of the problem as a need for customers, writing requirements and analyzing the same. Students apply modular design principles and break down the codebase into distinct modules. This technique enhances code organization, reusability, and maintainability. The second phase focused on developing the front end by harnessing the power of Angular, a leading web framework, to craft a sleek and interactive user interface. The backend is built using Node.js, which serves as the foundation, enabling the software system to cater to high-performance server environments.

These modules communicated seamlessly through well-defined APIs, facilitating the integration of various components within the application, ultimately delivering a seamless and responsive user experience. An industry expert conducted a workshop on Angular and React to get hands-on experience. The third phase focused on software testing using appropriate testing tools such as Selenium, Jmeter, TestComplete and Appium to test the web application. A software testing workshop was conducted for students by industry experts to expose the students to designing test cases, test plans, and testing strategies. The hands-on experience on testing tools was provided during the workshop. Faculty reviews are conducted on each phase, and rubrics-based assessment is done on each phase.

Approximately sixty teams created web-based applications for real-world scenarios. Positive aspects of the framework in feedback indicated that more than 87% of the students agreed that they could apply Software engineering principles and practices such as requirements management. modular design and testing in web applications. Also, more than 85% of students acquire skills from code-to-web design mastery by developing web applications in Angular Node.js and backend implementation. This framework helped to improve teamwork, presentation and communication skills. Confidence in software development improved to a greater extent. The design and implementation of the framework met the stated outcome of the courses. The student's academic performance improved by 10% compared to the previous year when students were not involved in the integrated project development.

Keywords— Framework, learning, practices, project, skills, technology, testing, web

***JEET Category*—Technology Enhanced Learning**

.

This paper was submitted for review on Sept 10, 2023. It was accepted on Nov, 15, 2023. Corresponding author: Padmashree Desai, Department, K.L.E. Technological University, Karnataka India. Address: Hubblii-580031 (e-mail: padmashri@kletech.ac.in).

I. INTRODUCTION

THE incredible progress in technology has directed to the application of software systems in our daily lives. Software engineering is an active domain that aims to create good-quality automated software systems through organized project practices custom-made to user-centred design principles. Software Engineering (SE) is one of the important core courses for computer science undergraduate students. According to industry inputs and job requirements, this course mainly cultivates students' abilities to master software engineering principles and methods for software project development. Students will immediately apply skills acquired through this course in the industry. Practical applications of the course benefit students. Developing a course project applying SE principles and techniques provides practical exposure to them. Similarly, Web Technologies (WT) is another course where students learn the implementation of frontend and backend and connect them for Web Technology standards. Practical knowledge in this course is essential during job interviews and

career opportunities. Statistics from the Bureau of Labor Statistics (BLS) in the U.S. show there will be about 2,20,000 job opportunities for web developers in 2022. In the same year, nearly 2.2 million more people worked as software developers, as per BLS. Directly or indirectly almost every society employs web developers and software engineers. All employees benefited financially from these job opportunities.

Nevertheless, people in these job roles normally collaborate and cooperate instead of working strictly from a software vs. web developer viewpoint. Usually, emphasis and duties can be interleaved. They can visualize alike big-picture areas despite different job opportunities. In minor organizations, one individual may handle both roles as a whole.

Software and web developers use almost similar programming and coding languages. They also often apply similar skills—communication, creativity, analyzing and problem-solving. Many times the boundary line between these two roles is negligible.

As per the study, a particular software engineer can develop a quality web application if he knows software engineering principles, different models for requirements gathering, designing aspects and testing tools that can be used during project development. A web application can suit customer requirements. If a developer knows technically which framework, backend and frontend can be used, then the quality of the developed web application will have a better user experience.

These facts enlightened us to design an integrated course project where students apply the theory and practices of SE and WT standards. Web application development is considered a part of continuous internal evaluation (CIE). It is stretched over a semester, monitored, and assessed through reviews. We designed a course delivery plan by following active learning pedagogy. Three reviews are designed focusing on important aspects of software development. Team formation is carried out with a maximum of members.

The proposed integrated learning framework includes the following phases:

Problem identification

Students are Identifying a real-time scenario by meeting customers. The project team identifies, analyses, and collects requirements from customers. Students identify need statements from customers who need software for their day-to-day lives, such as doctors, shop owners or K.L.E Technology event coordinators/associations.

Modular design and implementation

SE and WT standards and techniques are adopted to design and implement the project. It includes frontend and backend design integration using Angular Node.js and React.

Test automation

The web application is tested using appropriate testing tools such as Selenium, Jmeter, TestComplete and Appium. Test automation allows the students to validate the system as per system requirements. SE and WT faculty are involved in all phases of the development process and assessment with regular reviews.

A. Organization of the paper

Other sections of the paper are ordered as follows: Section II discusses the existing approaches to teaching learning Software Engineering. Section III discusses the integrated course project learning framework, planning the design of assessment parameters, problem identification, module design and implementation and testing of web application phase-wise. Section IV describes one case study implemented toward completing a web application, Section V validates the results and Section VI provides conclusion.

II. RELATED WORK

A literature survey on integrated project development using SE and WT involved examining research papers, articles, and books that explore the integration of software engineering practices with web technologies to enhance the development of projects is carried out. Below, the authours provide an overview of some key themes and relevant sources in this area.

(Ahmed et al., 2018; Alok et al., 2018 and hu et al., 2017) discuss skill development in freshman engineers and innovations in software engineering as a practical course to provide hands-on experience in software development. They brought innovative practice in the "Software Engineering" course.

Yehong Han (2016) recommended a task-driven approach for the laboratory course in software engineering. Students must hand in their software engineering project at the end of the course. The instructor performs the acceptance test to assess the student's project. This procedure provides a gradual increase of topic comprehension, analytical, and project execution skills.

Naredla et at., (2019) discuss an innovative approach to teaching software engineering that provides students with practical hands-on experience. They provided an intuitive lab environment using open-source, affordable software for students to learn, understand, and appreciate programming design concepts.

Alok et at., 2018 and Anuradha 2019) discuss the teaching-learning process and pedagogies that can help in practical knowledge. Web applications often benefit from established software engineering principles such as requirement analysis, design, and testing.

Different papers talk about integrating Agile and DevOps methodologies to streamline web development. User-centric design principles are vital in web development. APIs play a crucial role in connecting web applications with external services. Many case studies and best practice guides showcase successful integrated project development using software engineering and web technologies (Aluvala et al,. 2015).

Hiremath and Shetty(2015) described how to attain focused outcomes for Data mining and Web technologies courses through the Integrated Learning Framework. Desai et al.,(2015) discussed the flipped class as a pedagogy tool to teach a computer science course. Desai et al., (2016) provided an activity-based teaching-learning method for software engineering.

Desai et al., (2020) narrated how to measure students' performance and teachers' involvement using learning

analytics and problem-based learning. Hiramath et al.,(2021) provide an innovative approach for monitoring the progress of projects and assessment during Covid-19 at the engineering undergraduate Level.

III. Framework for Integrated Project Development

The integrated learning framework for SE and WT core courses is depicted in Fig. 1. The expectations and timeline schedule to complete the integrated course project are announced in the first week of the semester.

The integrated course project spreads over 13 work weeks of the semester. Three reviews are planned for the semester's 3rd, 8th, and 13th week, and the anticipated deliverables are clearly stated. All the reviews are assessed based on rubrics with a team of faculty members. It consists of three reviews: (i) Review-1, which comprises a presentation on the problem description, domain knowledge, and requirements elicitation (function and nonfunctional requirements) use cases. It carries a convinced weightage of marks (ii) Review-2, focus on module design, front-end user interfaces as per user experience and database design carries marks. (iii) Review-3 includes the final project demo, including requirements validation by writing appropriate test cases and testing plans using the tool. In between each review, one-to-one discussion is carried out with the team members.

The course on Software Engineering includes chapters on Requirements engineering, Agile software development, System modelling, Architectural design, Software testing, and introduction to DevOps. It is a 3-credit course which includes 40 hours of teaching.

The course on Web Technologies Laboratory includes chapters on Introduction to HTML basics, JavaScript, RESTful API using NodeJS and Express, Angular and React. It is a two-credit laboratory course that spans 36 hours of teaching and experiments.

Coordinating the theory and practices of two courses in this process was difficult as both courses run simultaneously for the students, and the course contents differ. In the first two weeks of the SE course, teachers covered topics on basic software engineering principles and requirements engineering in their classes. WT course teachers covered Introduction to HTML Basics JavaScript in laboratories. The objective and importance of the integrated project development are discussed with students. The total weightage given by both the courses is 40%. Details of rubrics and weightage for each review are explained to students as shown in Table 1.

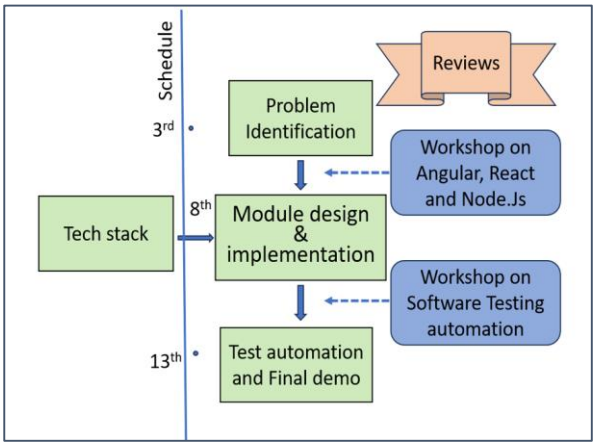

Fig. 1. Integrated project development framework

A. *Problem Identification*

The main objective of this phase is to identify a real-time problem scenario where students can interact with customers to identify issues, challenges and needs for software development. It also helps to identify the customer data and user interfaces they need to design. Team formation and procedure to identify the problem and documentation needed during requirement collection are discussed. Weightage for the review-1 is made clear to students. Brainstorming sessions are conducted by faculty during lab sessions.

Teams identified real-world problem scenarios related to KLE Technological University, such as event management, faculty profile management, department undergraduate graduate committee management, gymkhana, hostel and clinic support at KLE Tech. Also, students approached small clinics, shops and entrepreneurs to identify problems for software development. In this phase, students work towards understanding the problem domain. This has led to the students' development of communication and analytical skills. The assessment weightage allocated for this phase was 20%, which maps to Course Outcome CO2 and Programme Outcome PO3, as displayed in Table 1.

The following guidelines are defined and notified to student teams before the start of project development.

1. Pair-wise roles need to be assigned in a team. These two pairs will exchange roles and work towards task completion for a specific timeline.
2. Each pair will work on following specific tasks like
   - Analyst (requirement collection & elicitation)
   - Designer (Architecture and GUI designer - HTML+CSS)
   - Backend designer (Database design)
   - Developer (Implementation)
   - Tester
3. team pairs will change roles after one cycle or review.
4. The team leader can assign roles and tasks after discussions.
5. The team must meet with the client to collect, analyse,

and finalize the requirements.

6. Prepare minutes of meetings for every meeting conducted.

B. Modular design and implementation

The student team works towards understanding the solution domain and providing solutions in appropriate architectural patterns, class diagrams, and sequence diagrams for object-oriented design. Database design is done by writing an ER diagram.

In the context of the modular design and implementation, we focused on the following points:

- Modular Design: The project employed modular design principles, breaking the codebase into distinct modules. This technique enhances code organization, reusability, and maintainability.
- Frontend Framework (Angular): Utilizing a frontend framework like Angular is a technique to streamline the development of user interfaces, promoting code consistency and providing a rich user experience.
- Backend Environment (Node.js): Node.js, known for its event-driven and non-blocking I/O, was chosen as the backend technology, enabling a responsive and scalable server infrastructure.
- NoSQL Database (MongoDB): The selection of MongoDB as a NoSQL database is a technique to handle data efficiently, especially when dealing with unstructured or semi-structured data, providing scalability and flexibility.
- API Integration: Creating well-defined APIs to facilitate communication between modules is a core technique in software engineering. This approach allows different parts of the system to interact seamlessly.
- Scalability: The choice of technologies like Node.js and MongoDB also suggests a focus on scalability, a key consideration in software engineering for handling growing user loads.
- Maintainability: The emphasis on modularity, code organization, and clear APIs contributes to the overall maintainability of the software, making it easier to update and extend in the future.

These techniques demonstrate a comprehensive approach to integrated project development by applying SE and WT, ensuring the development of a robust, scalable, and maintainable web application. Industry experts train students on Angular, React, and Node.js through the 2-days workshop. Students acquire practical knowledge of frameworks so that students apply web technology standards easily in web development.

We provided standard guidelines to be followed during implementation:

- All functionalities should use the camelCasing convention for naming.
- Variable names should be meaningful.
- Documentation of code: The code has to be appropriately commented on along with the required test scenarios.
- Handle all the test conditions for exceptional cases like connection establishment, internet service failure, slow response, etc.
- The full stack framework, either MEAN or MERN, can be used for implementation.
- If any other full-stack framework is used, kindly report to the course teacher for approval.

Students presented a design solution following the above guidelines. Review 2 is conducted by faculty to assess the design per the parameters in Table 1. The assessment weightage for this phase was 20% with mapping to Course Outcome CO4 and CO5 mapping to PO14 and PO5, respectively, as shown in Table 1.

C. Testing and Demo of the web application

We conducted a day workshop on "Software Testing Automation" by an industry expert. Students are given hands-on experience writing test cases and plans using standard templates. Industry experts also demonstrated the testing tool selenium for automation.

IV. CASE STUDY

This section describes a web application developed by a student team using different phases. Students identified a problem for a clinic where a doctor manually does all tasks for patient management. A team of students decided to develop a web application for Dhanwantri Clinic to automate the activities of patient registration, appointments, scheduling and diagnosing.

The student team developed the "Dhanwantri Clinic" web application to fulfill the requirement of Dr. Pradeep Agnihotri, an Ayurvedic doctor, to have a digital solution for recording patient information and medical history. The primary goal was to create a user-friendly and secure platform that enables the client to manage patient records efficiently. The following images in fig. 2. shows where students meet the customer Dr. Pradeep Agnihotri for collecting requirements, discussion with faculty, and application demo, respectively.

Fig. 2. Interaction with the customer, faculty, and project demo to the customer.

The "Dhanwantri Clinic" web application incorporates the following functionalities:

- User Authentication: Secure registration and login functionality for the doctor.
- Patient Management: The ability to add, view, update, and delete patient records. This includes personal details, medical history, treatment plans, and

prescriptions.

- Medical History: A comprehensive section to record and track the medical history of each patient, including past diagnoses, treatments, and progress.
- Responsive Design: Ensuring the application is accessible and usable on various devices, such as desktops, tablets, and smartphones.
- Data Security: Implementing security measures to protect patient data and ensuring compliance with privacy regulations.
- User Interface Design: Utilizing Bootstrap to create an intuitive and visually appealing user interface.

Technology Stack

The following technologies are used in the development of the "Dhanwantri Clinic" Web Application for patient management:

a Frontend: React.js for building dynamic user interfaces

b Backend: Node.js and Express for handling server-side logic

c Database: MongoDB for storing patient data and related information

d Styling: Bootstrap for responsive and attractive UI design.

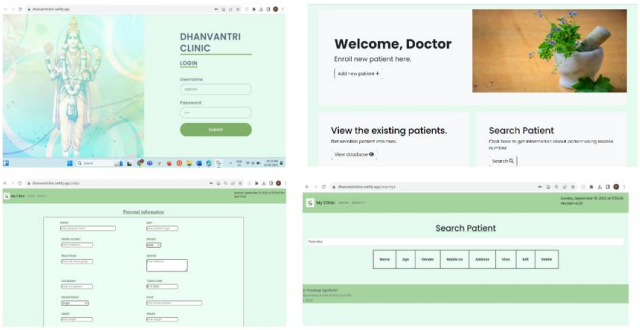

Fig. 3. Sample web pages developed by the student team for the Dhanwantri Clinic web application

Outcome of integrated project development as learnings:

The team successfully delivered the "Dhanwantri Clinic" web application for patient management, providing the following deliverables.

- Developed a user-friendly interface that simplifies patient information management
- Implemented robust authentication to ensure data privacy and security
- Designed a scalable system architecture using the MERN stack
- Provided a comprehensive platform for recording patient medical history and review facility

V. Results and Discussion

This section discusses the quantitative and qualitative assessments done to understand the impact of this integrated project development, focusing on developing students' software development skills.

Feedback is taken from a team on a Likert scale after completing the web application. When we observed the responses, no single team disagreed with the integrated learning approach followed.

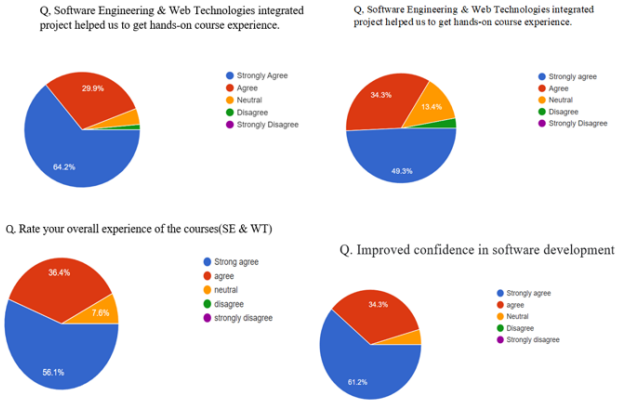

Fig.4. Students' feedback statistics for sample questions

Qualitative assessment was collected by customers from whom requirements are collected, and a web application was developed. Dr. Agnihotri was satisfied with the functionalities provided in the web application. The Dhanwanti Clinic web application is available at the following link: https://dhanvantriclinic.netlify.app/. Also, Dr. Sunitha was satisfied with the web application developed for Departmental undergraduate committee management activities.

This integrated framework is different from other integrated learning approaches where in we integrated the technology stack of the Web Technology course with the theory and principles of SE.

VI. Conclusion

The main contribution of the proposed framework is to teach integrated learning by developing a web application that combines the theory and practice of two core courses. Utilizing the MERN stack and Bootstrap allowed for a seamless user experience and efficient data handling. Software testing automation using tools helped the students test for all requirements quickly. The project not only met requirements but also exceeded the initial expectations and requirements set by the client. Approximately sixty teams created web-based applications for real-world scenarios. Positive aspects of the framework in feedback indicated that more than 87% of the students agreed that they could apply Software engineering principles and practices such as requirements management. modular design and testing in web applications. Also, more than 85% of students acquire skills from code-to-web design mastery by developing web applications in Angular Node.js and backend implementation. This framework helped to improve teamwork, presentation and communication skills. Confidence in software development for industry career improved to a greater extent. The design and implementation of the framework met the stated outcome of the courses. The student's academic performance improved by 10% compared to the

previous year when students are not involved in the integrated project development. Students and professors appreciated the participation and practices.

ACKNOWLEDGMENT

We acknowledge Prof. Manjula, Prof. Indira and Prof Neha for coordinating activities for integrated project development.

REFERENCES

Ahmed, S. M., Madhuri, G., Reddy, M. S., & Condoor, S. S. (2018). Skill development in freshmen by adopting project based learning-"introduction to engineering" course. *Journal of Engineering Education Transformations*, *2018*.

Alok, G., Anushalini, T., & Condoor, S. (2018). Effective approach towards development of idea through foundations to product design. *Journal of Engineering Education Transformations*, *31*(3), 47-52.

Wang, G., Zhou, Y., Jiang, H., Zhao, Y., & Li, X. (2016). Research on practical teaching of embedded software engineering based on flipped classroom and MOOC [J]. *Experimental Technology & Management*.

hu W H, Qian W B, Yang (2017) Teaching Practice of "Software Engineering" Course Based on KM Teaching Methodology[J*]*. *Education Teaching Forum*

Ke, G. U. (2017). Elementary Introduction to the Teaching of Software Engineering Course for Computer Science Major [J]. *Guide of Science & Education*.

Alok, G., Pothupogu, S., Reddy, M. S., & Priya, P. S. (2018, November). Trenchant pathway to bring innovation through foundations to product design in engineering education. In *2018 IEEE 6th International Conference on MOOCs, Innovation and Technology in Education (MITE)* (pp. 43-47). IEEE.

Anuradha, P. (2019). The teaching learning process. *International Journal of Advanced Science and Technology*, *28*(17), 709-714.

Aluvala, S., & Pothupogu, S. (2015). A traditional novel approach for skill enhancement of teaching-learning process in engineering education. *Journal of Engineering Education Transformations*, *28*(4), 92-95.

Han, Y. (2018, May). Teaching Research and Innovative Practice on the Course" Software Engineering". In *2018 International Conference on Advances in Social Sciences and Sustainable Development (ASSSD 2018)* (pp. 604-607). Atlantis Press.

Hiremath, P. S., & Setty, S. (2015). Integrated Learning Framework towards attaining focused outcomes. *Journal Of Engineering Education Transformations, Special*.

Kuhrmann, M., Nakatumba-Nabende, J., Pfeiffer, R. H., Tell, P., Klünder, J., Conte, T., ... & Hebig, R. (2019, May). Walking through the method zoo: does higher education really meet software industry demands?. In *2019 IEEE/ACM 41st International Conference on Software Engineering: Software Engineering Education and Training (ICSE-SEET)* (pp. 1-11). IEEE.

Flores, N., Paiva, A. C., & Cruz, N. (2020). Teaching software engineering topics through pedagogical game design patterns: An empirical study. *Information*, *11*(3), 153.

Kumar, S. (2009, February). Innovative teaching of software engineering: Practical approach with labs. In *2009 22nd Conference on Software Engineering Education and Training* (pp. 284-287). IEEE.

Naredla, S., Condoor, S., & Raja Shekar, P. V. (2019). Maker faire-promoting regional innovation & entrepreneurial ecosystem by empowering students. *International Journal of Recent Technology and Engineering*, *8*(2), 5584-88.

Desai, P., & Vijayalakshmi, M. (2015). Flipped classroom: An efficient pedagogical tool to teach a course for final year computer science and engineering graduate students. *Journal of Engineering Education Transformations*, *28*(Special Issue).

Desai, P., & Joshi, G. H. (2012, July). Activity based teaching learning in software engineering-An experience. In *2012 IEEE International Conference on Engineering Education: Innovative Practices and Future Trends (AICERA)* (pp. 1-6). IEEE.

Han, Y. (2018, May). Teaching Research and Innovative Practice on the Course" Software Engineering". In *2018 International Conference on Advances in Social Sciences and Sustainable Development (ASSSD 2018)* (pp. 604-607). Atlantis Press.

Flores, N., Paiva, A. C., & Cruz, N. (2020). Teaching software engineering topics through pedagogical game design patterns: An empirical study. *Information*, *11*(3), 153.

Joshi, A., Desai, P., & Tewari, P. (2020). Learning Analytics framework for measuring students' performance and teachers' involvement through problem based learning in engineering education. *Procedia Computer Science*, *172*, 954-959.

Hiremath, P. S., Sujatha, C., & Desai, P. (2021, November). Project Progress Monitoring and Assessment at Engineering Undergraduate Level during Covid 19 Pandemic-Challenges and Solution. In *2021 World Engineering Education Forum/Global Engineering Deans Council (WEEF/GEDC)* (pp. 319-324). IEEE.

Desai, P., Meena, S. M., Giraddi, S., Desai, S., & Hanchinamani, G. (2021, November). Transformation in Course Delivery Augmented with Problem-Based Learning and Tutorial. In *2021 World Engineering Education Forum/Global Engineering Deans Council (WEEF/GEDC)* (pp. 15-22). IEEE.

Joshi, G., & Desai, P. (2016, December). Building software testing skills in undergraduate students using the spiral model approach. *In 2016 ieee eighth international conference on technology for education* (t4e) (pp. 244-245). IEEE.

TABLE1. REVIEW DETAILS AND MAPPING WITH COS AND POS

Reviews	**CO**	**PI**	**BL**	**Marks**	**Timeline and deliverables**
Review-1: Problem identification	CO2: Analyse customer reequipments and prepare software requirements (SE).	3.1.2:Elicit and document system requirements (SE).	L4	20% (SE)	3rd week Requirements specifications
Review-2: Modular design and implementation	CO3: Develop a web application for real-world problems using full stack web development framework (WT).	2.1.2: Identifying engineering systems, variables, and parameters for solving the problems (WT). 3.2.2: Generating diverse potential design solutions to meet system requirements applying design concepts, tools and techniques (WT).	L5	20%(WT)	8th week • Modules specification • Database design using MongoDB, Angular and React framework • User interfaces.
Review-3: Test automation and Final demo	CO4: Perform test planning and test execution for a given system using relevant techniques (SE). CO5: Use tools to perform software Development Life Cycle activities (SE).	14.2.2: Plan and perform testing per the test strategy (SE). 5.1.1: Identify modern IT tools for software development life cycle and system analysis activities (SE).	L4	20%(SE)	13th week • Web application • Report
	CO3: Develop a web application for real-world problems using full stack web development framework (WT).	10.1.3: Write technical short reports, formal e-mails, and reports documenting experimental or simulation methods and results (WT)		20%(WT)	