The Educational Review, USA, 2021, 5(12), 478-489
https://www.hillpublisher.com/journals/er/
ISSN Online: 2575-7946
ISSN Print: 2575-7938

Fostering Creativity, Innovation and Problem-Solving Skills by Incorporating Design Thinking in an Introductory IT Course: Students' and Faculty Perceptions

Maha Al-Yahya[1,*], Henda Chorfi Ouertani[1], Sahar Bayoumi[1,2]

[1]Department of Information Technology, College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia.
[2]Department of Information Technology, Institute of Graduate Studies & Research, Alexandria University, Alexandria, Egypt.

How to cite this paper: Maha Al-Yahya, Henda Chorfi Ouertani, Sahar Bayoumi. (2021). Fostering Creativity, Innovation and Problem-Solving Skills by Incorporating Design Thinking in an Introductory IT Course: Students' and Faculty Perceptions. *The Educational Review, USA*, 5(12), 478-489.
DOI: 10.26855/er.2021.12.004

Received: November 10, 2021
Accepted: December 5, 2021
Published: December 29, 2021

Corresponding author: Maha Al-Yahya, Department of Information Technology, College of Computer and Information Sciences, King Saud University, Riyadh, Saudi Arabia.
Email: malyahya@ksu.edu.sa

Abstract

The IT job market demands graduates equipped with skills necessary for the new digital era. These skills include complex problem solving, critical thinking, and creativity. To provide the students with the necessary experience and the opportunities to practice these skills to be ready for the job market, we present the design and evaluation of an introductory course in the IT Bachelor program at King Saud University. The course introduces students to the IT discipline and provides an opportunity to experience problem solving and engage in activities which foster creativity and innovation using design thinking methods. In this paper, we present the design of the course, its content and structure, and an evaluation of the course over the period of two semesters. Student and faculty surveys were used to collect the data, and quantitative and qualitative methods were used for data analysis and evaluation. The results of this study show that students enjoyed the course and were satisfied with the development of their knowledge and problem-solving skills using design thinking. Faculty also indicated that the course has helped students build solutions targeting real world problems, and enhanced their problem solving and creative thinking skills.

Keywords

Design thinking, IT curriculum, problem solving, creativity, innovation, computational thinking

1. Introduction

According to the World Economic Forum "Future of Jobs"[1], the forecasts for the job labor market from 2020 to 2025 show that the top skills in demand include critical thinking and analysis, problem solving, as well as creativity and originality. These skills are essential elements in any introductory computer programming course (Tasneem, 2012). One approach to promote students' creative and critical thinking skills is by incorporating design thinking into the curricu-

[1] https://www3.weforum.org/docs/WEF_Future_of_Jobs_2020.pdf.

lum (Clemente et al., 2016), As for the Saudi Job market, a recent report by the City and Guilds Group[2] on building the pipeline talent in Saudi Arabia indicates that there is a mismatch between the outputs of higher education and the needs of the Saudi job market. Therefore, it is vital that universities in Saudi Arabia target this gap and continuously review their curriculum in light of the needs of the job market. Therefore, to ensure that IT graduates are ready for the job market, the Information Technology department at King Saud University has reviewed its curriculum (Al-Baity et al., 2018) and has introduced courses that foster the development of the essential skills required by the job market such as problem solving, critical thinking, creativity, innovation, communication, and entrepreneurship. The course has been offered for students enrolled in the IT program the first time in the first semester (Fall 2018), and in the second semester (Spring 2019). The objective of this research is to evaluate the course content and structure, and to assess the development of students' skills necessary for the job market in Saudi Arabia, problem solving and design thinking.

Traditionally, programming courses offered early in the curriculum do not have enough time to introduce key skills of critical thinking, creative problem solving, and design thinking, since the emphasis in these courses is on developing core programming knowledge and technical concepts. Therefore, to prepare market-ready graduates we need to create learners who can problem solve using various methods which include critical thinking, computational thinking and design thinking. As part of the ongoing process of review and development of the bachelor's in information technology curriculum at KSU, a new 3-credit hour course titled "IT 210: Information Technology Fundamentals" has been introduced in the program. This course aims to equip students who are new to the discipline with the basic knowledge and skills required to succeed in an IT program. The course was proposed in the curriculum review process to achieve three main objectives: (1) Encourage creative thinking and innovation using the design thinking methodology, as these are key skills required in the job market (2) Enhance complex problem-solving skills, and (3) Introduce students to the IT discipline.

In this paper, we present the design of a new introductory course in the Bachelor of IT program at the College of Computer and Information Sciences in King Saud University (KSU). We also present the perceptions of students and faculty on the course design and structure, and how it has enabled the development of student problem solving and creative thinking skills. The remainder of the paper is structured as follows: Section 2 introduces the topic of design thinking, and Section 3 reviews key literature reporting its implementation in the classroom. Section 4 presents the course design and how design thinking was incorporated in the course. Section 5 presents the discussion, and finally Section 6 presents the conclusions of the study.

2. Design Thinking

Design thinking is a "human-centered approach to problem-solving that help people and organizations become more innovative and creative" (Brown, 2009). The main goal of Design Thinking is to match customer needs with what technology can provide. Design thinking employs empathy, creativity and experimentation to arrive at innovative solutions, and when incorporated with software design delivers better solutions for problems (Denning, 2013). At the core of design thinking is an approach to problem solving that is based on inquiry, ideation, reflection, and modification. Using deign thinking methodology, students focus on a "bottom-up user-centered solution rather than a top-down technology focus solution" (Hardy et al., 2018).

An interesting aspect of design thinking is that it gives students the opportunity to "fail" and think of new, creative ways to solve the problem. Giving students the per-mission to test and fail is key to encourage creative thinking (Pink, 2010). Moreover, design thinking requires "a conceptual change in the way that problems are framed and solved" (Hardy et al., 2018). Instead of having one correct solution, multiple potential solutions are developed.

The Stanford design school model of design thinking[3] focuses on five key stages of the design thinking methodology:

- Empathize: "in order to gain insights which may reveal new and unexplored ways of seeing, and courses of action to follow in bringing about preferred situations for business and society. Where the designers examine as much data as they feel necessary to be able to fully contribute to the problem-solving process".
- Define: "reframing the perceived problem or challenge at hand, and gaining perspectives, which allow a more holistic look at the path towards these preferred situations. Where the problem is defined as best as possible prior to solving it".
- Ideate: "explore possibilities; divergent styles of thinking to explore as many possibilities as possible, deferring judgment and creating an open ideations space to allow for the maximum number of ideas and points of view

[2] https://www.cityandguildsgroup.com/research/building-the-talent-pipeline-in-saudi-arabia.
[3] Stanford Design School at https://dschool.stanford.edu/.

to surface. Where the designer commences creating possible solutions without examining their practicality until a large number of solutions has been proposed. Once this is done, impractical solutions are eliminated or played with until they become practical".

- Prototype: "early exploration of selected ideas, rapidly modeling potential solutions to encourage learning while doing, and allow for gaining additional insight into the viability of solutions before too much time or money has been spent. Where that idea is built and delivered as a product. Where the best ideas are simulated in some means so that their value can be explored with users".
- Test: "assess outcomes and plan improvements. Each step guides students through deeper understanding and mastery of cognitive problem-solving skills and productive habits of mind. Where the product is tested with the user in order to ensure that it solves the original problem in an effective manner".

3. Related Work

The design thinking approach has been incorporated into curriculums and courses especially in engineering and computer science to enhance students' skills in creative thinking and problem solving. A number of studies have reported using design thinking as pedagogy for educational activities involving problem solving and creative design. The majorities of studies are experience reports, which aim to enhance our understanding of design thinking use in the curriculum and provide student reflections. Design thinking can be applied in introductory programming courses as well as senior and advanced level software engineering courses, entrepreneurship courses, or independent workshops.

For introductory courses, the study presented in (Shalamova, 2016) describes an experience in integrating user experience (UX) concepts and design thinking in a technical communication course to enhance students' communication skills. The study showed that students appreciate human-centered method of documentation for technical products. In addition, students enjoyed the course, as the design thinking is a hands-on approach with activities that appeals to engineering students and provides engagement. The work presented in (Rajashekharaiah et al., 2016) describes the use of design thinking in a second year object oriented programming course to enhance students' skills in problem analysis and object oriented concepts. The authors describe the lab experience of the course in which a design first then implement later approach to software development was adopted. The authors report that students' performance was improved in all three stages of development (design, implementation, and testing). They also report that students found that the approach has helped them develop their design and analysis skills. For developing mobile applications, design thinking is introduced in the context of a mobile app development project (Gama et al., 2019). The study presents the experience of applying a framework that incorporates design thinking with challenge-based learning and lean startup methods to develop mobile apps. The aim was to encourage creative thinking and help students build solutions targeting real world problems. The study indicated that students felt motivated and were able to develop apps quickly.

For senior courses, design thinking was introduced in a capstone project within agile software development practices to enhance students' problem-solving skills (Palacin-Silva et al., 2017). In this study, design thinking was incorporated into the requirements engineering, software design, and testing. The study reported that the design thinking approach was suitable for the iterative development process of software. Moreover, they report that using design thinking methods such as personas, story boarding, journey maps, prototype and usability testing was successful in helping students think of and develop innovative solutions. In another study (Sohaib et al., 2019), the authors show how design thinking can be incorporated into eXtreme Programming (XP) exploration phase, prototyping and usability evaluation for a technological product. The study demonstrates the integration of design-thinking and XP to improve software project management while maintaining usability. Here design-thinking is used as a method for analyzing user needs in the software development project.

Design thinking has also been introduced in an entrepreneurship course for science and engineering students to help them acquire an entrepreneurial mindset (Lynch et al., 2019). During the course, students were introduced to the design-thinking framework and were asked to find new entrepreneurial opportunities for a technological service. Students indicated that the course was valuable and engaging and enhanced their skills regarding design thinking and commercialization of technology, as well as other skills such as teamwork, interpersonal communication, networking, and empathy. The study also reported that the students began to apply design thinking in real-life. Another example of the application of design thinking in an IT course is the study presented in (Lin et al., 2020). The study aimed to explore the effects of design thinking and how it can facilitate achieving the different levels of curriculum goals. The authors report that design thinking helped students' master the basic information knowledge and skills, and that students' digital works were more valuable, diverse, and creative with design thinking approach.

Apart from being part of the curriculum, some programs introduce design thinking to their students in workshops. For

example, the study presented in (Pham et al., 2018) describes the experience of introducing design thinking to help students generate ideas and inspire creativity for a mobile app development project.

Another use case for design thinking within the educational curriculum is to support in the structuring of courses. The study presented in (McKilligan et al., 2017) describes the use of design thinking as a method to restructure the teaching and learning experiences in a electrical and computer engineering department. The study investigated how the design thinking process can be used to develop better pedagogical approaches for engineering education.

4. The IT 210 Course Design

Students at the College of computer and information sciences at KSU complete the first year of the program as a "Common Year" with students from other scientific majors. They then specialize in Information Technology in the first semester of the 2nd year, in which they register in the IT 210 course. The first offering of this course was opened in the first semester of 2018. The total number of students enrolled in the course for the first semester is 129 students, and for the second semester is 40 students. A total of six instructors and lab assistants delivered the course. The first semester offering usually has a large number of students since it is a normal cohort. The second semester offering usually has fewer students enrolled, since the majority are transfer students. The IT 210 course is divided into three blocks: (1) Introduction to IT, (2) Problem Solving, and (3) Design Thinking. Each block is given over a period of 4 weeks during the semester. The blocks cover the following topics:

- Introduction to IT: introduces IT as a discipline, its relevance and impact on the world. It presents the basic concepts of information technology and issues related to ethics, security and privacy. It also provides an overview of computer architecture including hardware, software and data representation.
- Problem Solving: covers critical thinking, logic, and analytical methods for solving problems. This part of the course allows students to experience and practice visual methods for representing solutions as well as block programming. It covers levels of detail (abstraction); steps required to solve a problem (algorithms); breaking a large problem into a set of smaller problems (decomposition); recognizing that a new problem can be similar or related to other problems (pattern recognition); learning that a solution can be used to solve other problems (generalization). Students are also exposed to the pillars of computational thinking (CT) namely: decomposition, pattern recognition, abstraction, and algorithm design. The focus is mainly on how to use CT to solve problems rather than on coding and implementing the solution. It also presents algorithmic-oriented activities including the fundamental control flow structures in programming: sequential, selection, and repetition offer much greater benefits for students.
- Design Thinking: introduces creative problem solving using the design thinking approach. During this block, students are given the opportunity to apply the design thinking process on a real-world project, build a prototype for the solution using MIT App Inventor, and evaluate the prototype with real users.

Students apply the design thinking methodology through a course project, which is team-based project performed within teams of five students. Technology today is associated with on-line, social, and increasingly mobile solutions, therefore the project for this course was chosen as a mobile application. Moreover, the projects were chosen by students to solve real world problems. Giving the students the chance to work on a re-al-world problem enhances students' entrepreneurial skills. Moreover, true passion comes from a sense of purpose, which drives innovation. Real world problems give students that sense of purpose, as they need to know that their learning has deeper meaning such as helping others, that sense of purpose will sustain them through failures and keep them eager to learn.

Students were introduced to the MIT App Inventor [4] and the Scratch programming environment (Maloney et al., 2010) for the development of the projects. Both tools are suitable for beginners in programing since they provide a block-based visual programming environment. Students apply the design thinking stages to prototype a mobile app using the MIT App Inventor or an educational program using scratch. MIT App Inventor and Scratch are categorized as visual and block-based programming environments, these environments have been used in introductory programing courses to motivate students and facilitate the comprehension of computational thinking and problem solving (Vinayakumar et al., 2018).

5. Course Evaluation and Results

To evaluate the course, data was collected from course instructors, lab assistants and students who were involved in

[4] MIT App Inventor http://appinventor.mit.edu/explore/.

the course. Quantitative and qualitative methods for data analysis and evaluation were used in this study.

At the end of each course offering (Fall and Spring semesters), course instructors were asked to give their feedback on the course content and structure. A total of six instructors and lab assistants were involved in the study. In addition, a survey was distributed to the students in both course offerings. A total of 69 students responded to the survey (40% first semester, 45% second semester). The objective of the survey was to measure the general level of satisfaction of students with the course (the structure and content), their understanding of the IT discipline, and the development of their skills in problem solving and design thinking. A number of open-ended questions were also provided to get feedback from the students on their general perceptions and improvements for the course.

In addition, to assess student acquired skills, feedback from instructors teaching another relevant course in the IT program, the User Experience Design (UX Design) course. Students usually enroll in this course in the third year, after the IT 210 course. Therefore, the UX Design course instructors were also interviewed to see if the IT 210 introductory course has had an impact on the development of their skills in creative problem solving and design thinking.

5.1. Students Feedback

Students were asked on their overall satisfaction of the quality of the course, and the development of their knowledge and problem-solving skills. They were asked to rate the course on the following scale (excellent, very good, good, fair, poor), the results for both semesters are shown in Figure 1-A. The students were also asked on the level of satisfaction with the course, in terms of knowledge gained and problem-solving skills developed, the results are shown in Figure 1-B.

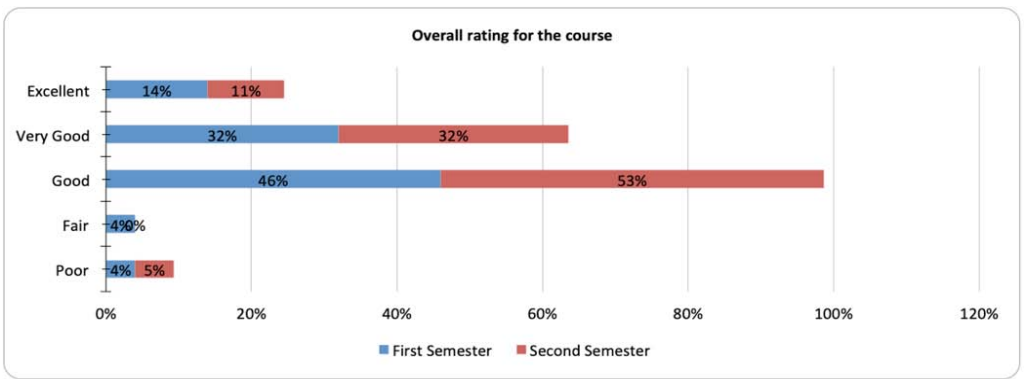

Figure 1-A. Students' overall rating of the course.

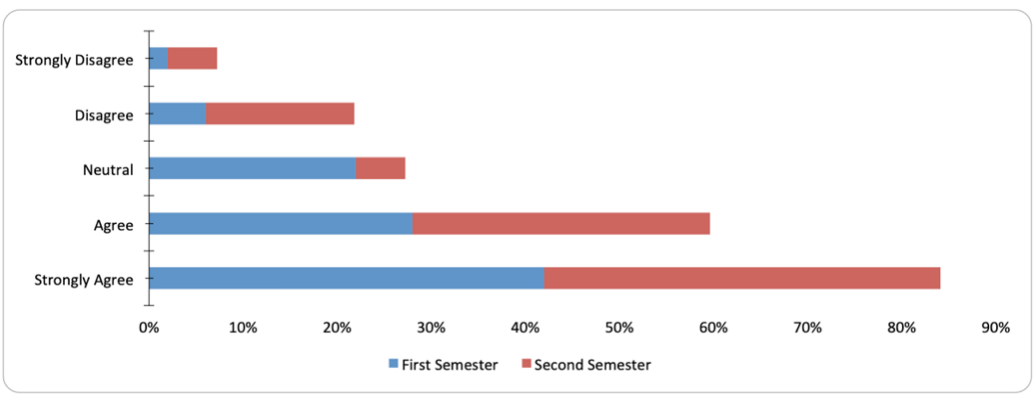

Figure 1-B. Level of Satisfaction.

5.1.1. Student learning and the development of new skills

To assess aspects of student learning, students were asked to rate their agreement on the following: (1) interest in IT, (2) interest in problem solving, and (3) interest design thinking, using a five-point agreement scale - strongly agree (SA), agree (A), neutral (N), disagree (D), strongly disagree (SD). The results for first and second semester are shown in Figure 2 left and Figure 2 right respectively.

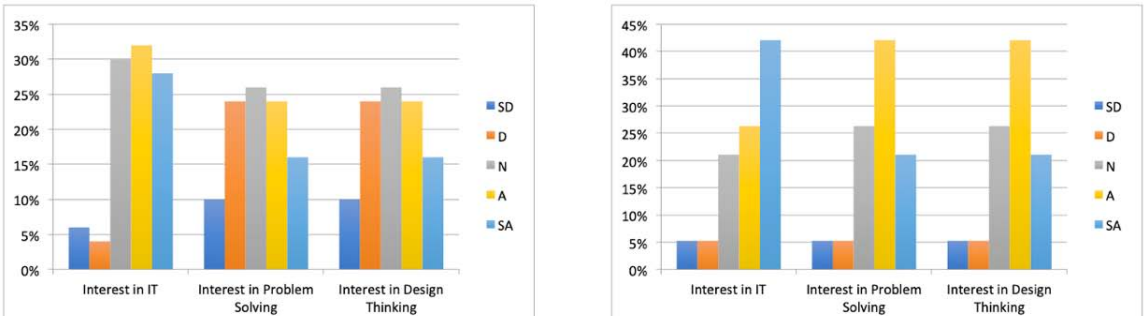

Figure 2. First Semester (left) and Second Semester (right).

They were also asked on their opinion of the course as an introduction to the IT discipline. Results are shown in Figure 3.

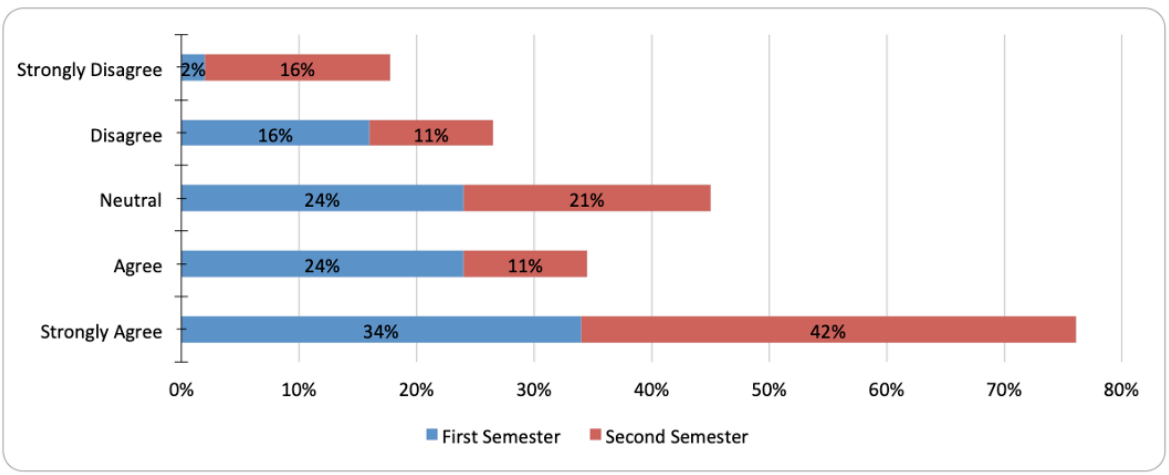

Figure 3. How helpful was the course as an introduction to IT.

They were also asked on their learning experience, and how much they rate their achievement of the course outcomes, and the level of experience in the course subject matter. The results are shown in Figure 4.

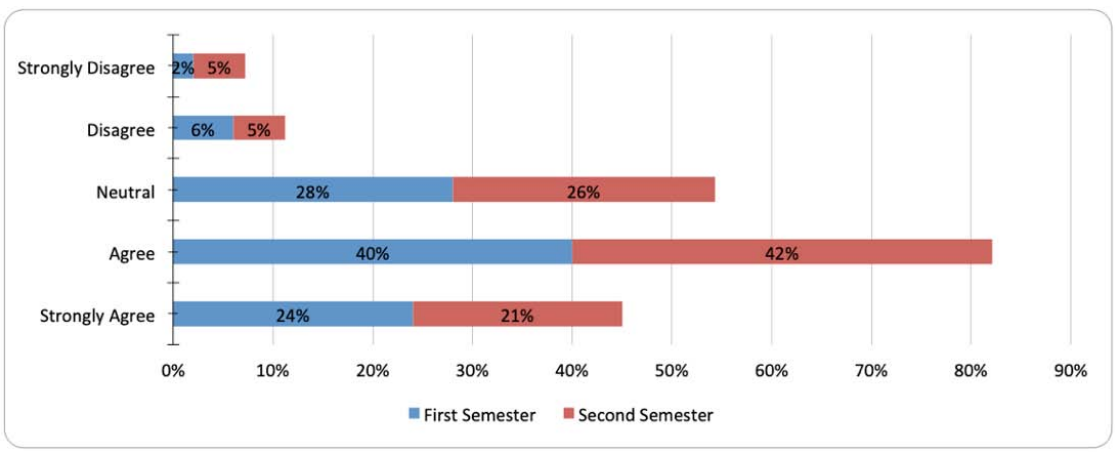

Figure 4. Level of experience with course subject matter.

Students were asked on their opinion if the course has supported the development of new skills, and a separate statement asked how the course has developed their problem-solving skills. Results are shown in Figure 5.

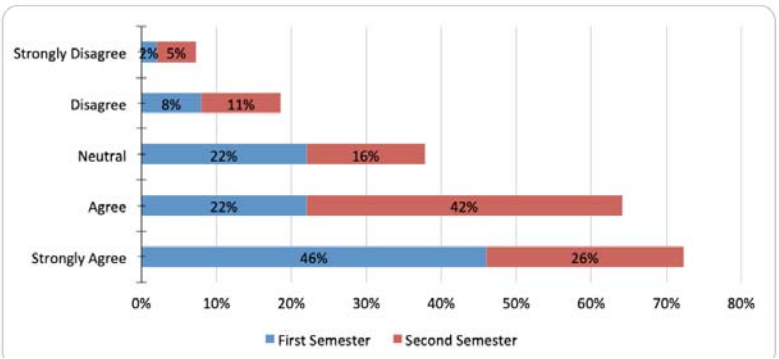

Figure 5. The course has helped me develop my problem-solving skills.

5.1.2. Student Feedback on Course Design

Students were asked to select the topics they enjoyed learning about in the course from a list of course topics, which were the most interesting topics in this course. The distribution of topic popularity with percentages is presented in Figure 6.

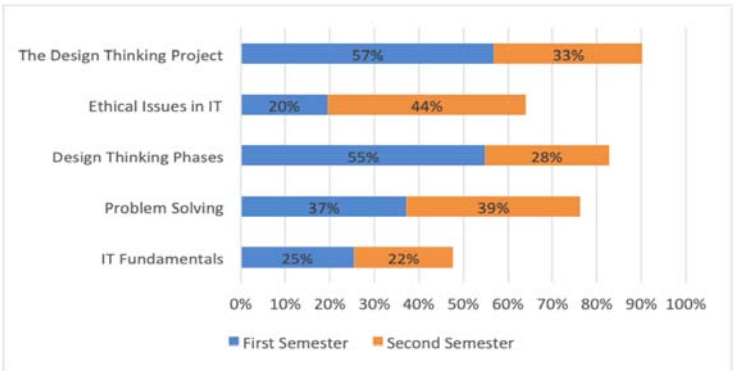

Figure 6. Which were the most interesting topics in the course.

Students were also asked to rate their agreement on the course design including the block system, the project and lab work, as well as the course grades distribution. The results are presented in Table 1

Table 1. Feedback on Course Design

Item	Likert	First Semester	Second Semester
The lab course work was suitable	Strongly Disagree	6%	16%
	Disagree	16%	37%
	Neutral	30%	0%
	Agree	24%	16%
	Strongly Agree	24%	32%
The grade distribution for the course was suitable	Strongly Disagree	18%	21%
	Disagree	22%	26%
	Neutral	10%	16%
	Agree	18%	11%
	Strongly Agree	32%	26%
I like the way the course is designed as three blocks (IT, Problem Solving, Design Thinking)	Strongly Disagree	14%	5%
	Disagree	16%	16%
	Neutral	8%	16%
	Agree	10%	26%
	Strongly Agree	52%	37%

5.1.3. Student Feedback on Applying Design thinking in the course Projects

To see how the course project supported student understanding and learning of the design-thinking approach, students were asked to rate their agreement on the statement: "The project has helped me learn and understand the design thinking approach". The results are shown in Figure 7.

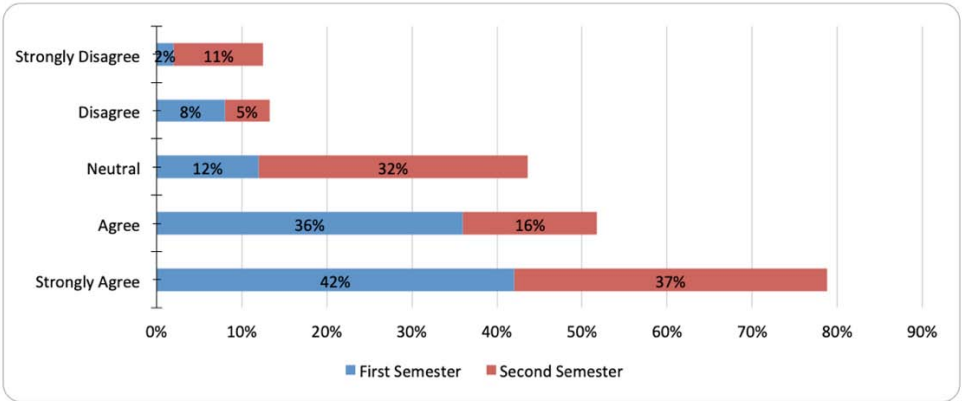

Figure 7. How helpful was the Project in Understanding Design Thinking.

In addition, students were asked the following open-ended questions to help us understand the students experience and gain insights into the project. The questions are as follows:

1. With regards to the project, to achieve maximum benefit, what would you suggest?
2. From your experience in working in a team (project, class activity), what are the problems you have experienced, if any?

Regarding the first question, some of the students' suggestions are summarized in the following points:

- Introduce design thinking early in the semester so that students can have more time to work on their projects. One student quoted: "I believe that if we had the project in the beginning of the semester, we would've done so much better because we'd have more time to think and create something better without having to think about other exams and projects".
- Adjust the grading criteria for the project, some students thought it was general and very high level and does not reflect the specific elements and approach of design thinking. They suggested that more specific grading criteria would be helpful.
- Introducing more professional tools for mobile app development in the lab for the project.
- The MIT App inventor does not support mobile app design for iOS and does not support professional interface designs.
- Some students suggested that design thinking be offered in advanced levels and not introductory level course so that they can benefit more.
- Some students thought that leaving the freedom for them to choose a problem was not very helpful since they were unsure if the problem, they chose was suitable for the design thinking project.
- Some students recommended having the project submitted in phases with feedback given before they started the next phase.

As for the second question, regarding working in a team (project, class activity), students mentioned the following issues:

- Some students mentioned they have had problems in team formation as well as communication between team members.
- Students also mentioned that it was difficult to schedule meetings given the busy schedule of students at the end of the semester.
- Students also mentioned that in a team, not every student contributed the same efforts.
- Another issue with teamwork is getting consensus from team members on decisions.
- Some students suggested they were better off doing an individual project, since they faced some team problems, such as collaboration and commitment of some team members. Some suggested giving the students the choice to work as part of a team or an individually.

5.1.4. Students Perceptions of the Course

Students were asked a number of open-ended questions to get feedback on their general perceptions on the course. They were asked: "What did you like most about this course? Student responses are summarized below:

- They enjoyed applying design thinking.
- They enjoyed the project
- The change of content and instructor each month provided excitement in the course.
- They liked the change in teaching style provided by each instructor.
- Students liked how the course presented the IT discipline.
- They enjoyed computational thinking and how to approach problem solving computationally.
- Ethics in IT and how the advancement of AI is presenting challenging ethical issues.
- The history of computing and the advancement of technology in our daily lives, smart homes, smart devices, and smart fabrics.

Students were also asked if they had any suggestions to improve this course lectures and lab. Student responses are summarized below:

- Some students though that some parts of the "Introduction to IT" block of the course was something they already have knowledge from the preparatory year, and suggested that the content be reconsidered, and suggest that more advanced content be provided.
- Some students thought that there is no connection between the three course blocks, as they seem like three separate courses in one.
- Some students mentioned that the relation between what is covered in the lecture and what is presented in the lab is not apparent.
- Some students suggest that the design thinking lectures and project be part of the lab.
- Another suggestion was to add more problem-solving examples for the computational thinking block.
- Add more interesting tools for the lab, such as tools for project management and team collaboration.
- Confusing pseudo code with java code as students are taking both courses in the same semester. Algorithms are already covered in the CSC 111 course.
- Some students suggested focusing on career paths for IT graduates and the specialties and tracks in the program.

Finally, students were asked to provide any other comments, which are summarized below:

- The change of instructors three times during the semester caused concern for some students, getting used to the instructor and her teaching and assessment style.
- However, some students saw this change in instructors and style something good since it adds excitement and enthusiasm. It is also fairer for students that one instructor teaches all course sections the same part of the course.
- Students suggested if a flavor of other IT courses could be infused in this course.
- Students suggested Linking between the three blocks of the course.

5.2. Instructors' Feedback

The IT 210 Course instructors for both semesters were asked to rate their overall experience of the course, and how well it has achieved its objectives: Introducing new students to the discipline of IT, developing students problem solving skills, and development of student's creativity using design thinking methodology. They were also asked on improvements to the course content, structure, and method of teaching. The feedback from the instructors has been analyzed and is categorized as follows:

5.2.1. Course Content

- Instructors felt that in general, the material for the course is sufficient for an introductory course in IT.
- Some instructors suggested that adding content related to the variety of data types such as media, images, sound, animations, and videos would be useful for students.
- Another suggestion from instructors was to introduce students in the lab to use tools for data processing as data are considered an important part of any IT system.
- They also indicated that students developed the skills through the informative lectures, the extensive practice, and the discussions during lectures and lab sessions. They also noted that the course project had improved their communication skills and self-confidence in this early stage of the program, as they were required to do interviews with users and present their work to their instructors and their peers in class.

- Another important observation by faculty is that students were exposed to the IT discipline and what constitutes the pillars in IT, as well as the tracks in the program, this has helped them understand the program and the major tracks (Cyber-security, Data Science, and Networks and IoT).
- One instructor highlighted that in this course students are introduced to computational thinking skills (decomposition, pattern recognition, abstraction, algorithm design), how to use the skills to solve problems. This course supports students to build logical and problem-solving skills necessary to develop algorithmic thinking. In programming courses, students usually struggle with syntax and semantics be-sides working out a problem computationally. However, with this course, students were able to acquire the necessary skills to carry out a solution regardless the implementation specificities.

5.2.2. Course Design and Structure

- Problem solving and computational thinking should be introduced earlier in the course, so that students can benefit from the learning while taking the CSC 111 Programming-1 course.
- Instructors suggested if it is possible that the IT 210 course be offered the semester prior to the CSC 111 course.
- Instructors also suggested organizing short sessions within the course on the Bachelors in IT program tracks to provide students with an understanding of major specializations in the program and how they relate to the IT discipline and opportunities in the job market for each specialization.

5.2.3. Design Thinking Project

During the first semester, the project was presented late during the semester, in week 9, and students did not have enough time to practice the complete design thinking process. Therefore, it is suggested that students be introduced to the design-thinking project early in the semester. This improvement has been implemented in the second semester, as students were introduced to the project in week 7. From this experience, instructors think it is better to introduce the design-thinking project early in the semester. A suggestion is to have the design thinking phases and project be part of the lab sessions instead of the lecture, thereby giving students more time to practice design thinking and problem solving on a real-world project.

5.2.4. Programming Tools

- MIT App Inventor, used in the first semester is a good tool for beginners in programming, however, students in the course opted to use other tools for prototyping mobile apps (interface mockups), and where given permission in the course to do so. Students found these alternatives easier to use and offered more interface design features. Instructors suggest investigating available mobile app development tools suitable for beginners to see other alternatives for MIT App inventor.
- The scratch visual programming environment was used in the second semester for the design-thinking project. It supports web development for free, and students completed the design and development of prototypes of educational web applications. However, the process for development of mobile apps using scratch needed more time, which the students did not have. Instructors suggest that maybe more time during the lab can be provided to introduce the students to mobile app development using scratch.

5.2.5. Impact on Other Courses

The Instructors teaching the User Experience Design course (a course offered after the IT 210 course) have highlighted the fact that when comparing students who completed the IT 210 course to those who have not (previous cohorts), have noticed more understanding and appreciation of the user's point of view and how approaching problem solving from a user's perspective influences the quality and functionality of the solutions they develop. One instructor indicated that a main challenge in designing an active problem-solving class is the selection of activities and exercises that permits to promote computational thinking. Designing any skill-based topic is generally effort demanding. It requires significant amount of hands-on experiences. Subsequently, there is a need to setup active classrooms (for both lectures and labs) where students are fully involved. They will spend most of their class-time on learning-by-doing. Classroom discussions would be preferable to lectures. It is important to support more peer discussion, peer teaching and peer reflection. Assignments have to be meticulously designed to require cooperative teamwork.

6. Discussion

The findings of this study show indicate that in general this course has been successful in introducing students to the IT discipline and providing them with a course that is designed to develop two of the most important skills necessary

for the IT job market of today that is problem solving and creativity. Using design thinking, students were able to practice real world problem solving using an approach that fosters creativity and innovation and targets the user needs and requirements. In general, the course has had a positive impact on students and faculty, with a good level of satisfaction. However, there is still room for improvement so that the course can better prepared students for their future careers in IT by improving their problem solving and design thinking skills.

7. Conclusions

In this paper, we presented the design of an introductory course in the Bachelors in IT curriculum which supports the development of key skills required by the job market: creativity, innovation, and problem solving. An assessment of the course over two semesters has been conducted and the results indicate that the design of the course supports the achievement of its objectives and encourage the development of students' skills.

The course is designed to give students the opportunity to experience design thinking, a problem-solving methodology that fosters creativity and innovation. It also enhances their problem-solving skills and provides them with a strong foundation to succeed in the Bachelors in IT program. The project for the course is designed to give students the opportunity to experience problem solving and design thinking for a real-world problem, which provides a sense of purpose and gives learning a deeper meaning.

Acknowledgments

We would like to thank the IT 210 students and faculty involved in this study for their kind participations.

References

Al-Baity, H., Alsaeed, D. H., Bayoumi, S., Al-Twairesh, N., and Al-Khalifa, H. (2018). Rejuvenation of the IT Program at King Saud University: A Change Reflecting Local and Global IT Trends. *Proceedings of the 19th Annual SIG Conference on Information Technology Education*, 129-134. https://doi.org/10.1145/3241815.3241866.

Brown, T. (2009). *Change by Design: How Design Thinking Transforms Organizations and Inspires Innovation*. HarperBusiness.

Clemente, V., Vieira, R., and Tschimmel, K. (2016). A learning toolkit to promote creative and critical thinking in product design and development through Design Thinking. *2016 2nd International Conference of the Portuguese Society for Engineering Education (CISPEE)*, 1-6.

Denning, P. J. (2013). Design Thinking. *Commun. ACM*, *56*(12), 29-31. https://doi.org/10.1145/2535915.

Gama, K., Castor, F., Alessio, P., Neves, A., Araujo, C., Formiga, R., Soares-Neto, F., and Oliveira, H. (2019). Combining challenge-based learning and design thinking to teach mobile app development. *Proceedings - Frontiers in Education Conference, FIE*, *2018-October*. https://doi.org/10.1109/FIE.2018.8658447.

Hardy, D. L., Myers, T. S., and Sankupellay, M. (2018). Cohorts and cultures: Developing future design thinkers. *Proceedings of the 20th Australasian Computing Education Conference*.

Lin, L., Shadiev, R., Hwang, W.-Y., and Shen, S. (2020). From knowledge and skills to digital works: An application of design thinking in the information technology course. *Thinking Skills and Creativity*, *36*. https://doi.org/10.1016/j.tsc.2020.100646.

Lynch, M., Kamovich, U., Longva, K. K., and Steinert, M. (2019). Combining technology and entrepreneurial education through design thinking: Students' reflections on the learning process. *Technological Forecasting and Social Change*, 119689. https://doi.org/10.1016/j.techfore.2019.06.015.

Maloney, J., Resnick, M., Rusk, N., Silverman, B., and Eastmond, E. (2010). The scratch programming language and environment. *ACM Transactions on Computing Education (TOCE)*, 10(4), 1-15.

McKilligan, S., Fila, N., Rover, D., and Mina, M. (2017). Design thinking as a catalyst for changing teaching and learning practices in engineering. *2017 IEEE Frontiers in Education Conference (FIE)*, 1-5.

Palacin-Silva, M., Khakurel, J., Happonen, A., Hynninen, T., and Porras, J. (2017). Infusing Design Thinking into a Software Engineering Capstone Course. *2017 IEEE 30th Conference on Software Engineering Education and Training (CSEE T)*, 212-221.

Pham, Y. D., Fucci, D., and Maalej, W. (2018). A First Implementation of a Design Thinking Workshop during a Mobile App Development Course Project. *2018 IEEE/ACM International Workshop on Software Engineering Education for Millennials (SEEM)*, 56-63.

Pink, D. H. (2010). *Drive the surprising truth about what motivates us*. Canongate. http://search.ebscohost.com/login.aspx?direct=true&scope=site&db=nlebk&db=nlabk&AN=572423.

Rajashekharaiah, K. M. M., Pawar, M., Patil, M. S., Kulenavar, N., and Joshi, G. H. (2016). Design Thinking Framework to Enhance Object Oriented Design and Problem Analysis Skill in Java Programming Laboratory: An Experience. *2016 IEEE 4th International Conference on MOOCs, Innovation and Technology in Education (MITE)*, 200-205.

Shalamova, N. (2016). Blending engineering content with design thinking and UX to maximize student engagement in a technical communication class. *2016 IEEE International Professional Communication Conference (IPCC)*, 1-5.

Sohaib, O., Solanki, H., Dhaliwa, N., Hussain, W., and Asif, M. (2019). Integrating design thinking into extreme programming. *Journal of Ambient Intelligence and Humanized Computing*, *10*(6), 2485-2492. https://doi.org/10.1007/s12652-018-0932-y.

Tasneem, S. (2012). Critical Thinking in an Introductory Programming Course. *J. Comput. Sci. Coll.*, 27(6), 81-83.

Vinayakumar, R., Soman, K., and Menon, P. (2018). Alg-Design: Facilitates to Learn Algorithmic Thinking for Beginners. *2018 9th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, 1-6.