The Role of Software Entrepreneurship in Computer Science Curriculum

Kevin Daimi and Nassif Rayess
College of Engineering and Science
University of Detroit Mercy, Detroit, Michigan, 48221
{daimikj, rayesna}@udmercy.edu

ABSTRACT

Future undergraduate Computer Science and Software Engineering curricula need to be revitalized. This revitalization is based on promoting and practicing computational thinking. Computational thinking could seed entrepreneurship if we provide our computer science and software engineering students with entrepreneurial skills and discover potential entrepreneurs among them. Entrepreneurship develops new computing markets, introduces new software and hardware technologies, and creates employment. Therefore, it is demanding to enhance our computer science and software engineering programs with entrepreneurship. In this paper, we aim to design a course in Software Entrepreneurship. Course rationale, description, objectives, outcomes, and assessment will be introduced. In addition, the characteristics of suitable case studies and projects for such a course will be highlighted.

Keywords
Entrepreneurship, Computational Thinking, Entrepreneurship Thinking, Innovation, Software Entrepreneurship, Course Design, Course Outcomes, Course Assessment

1. INTRODUCTION

Entrepreneurship is embodied in a skill set that is conducive to identifying opportunities and creating successful ventures to make the most of such opportunities. Opportunities are prerequisite to successful entrepreneurship ventures and the ability to identify and assess opportunities is a prerequisite skill to an entrepreneur. Basically, if one is unable to spot a market/business opportunity, one simply keeps on walking unaware of having missed anything. Also, if one is unable to properly assess the value of an opportunity, one is apt to miss a great prospect or, even worse, to embark on a doomed adventure. The importance of opportunity is discussed amply in the literature [1-2]. From an entrepreneurship education standpoint, educators can “sow the seeds” of Entrepreneurship by training the students to identify opportunities.

Researchers have identified different types of entrepreneurship, depending on the context and the kinds of opportunities that are available. The type of entrepreneurship that is commonly known to the general public is commercial entrepreneurship, and involves identifying market opportunities and exploiting competitive advantages for the purpose of profit. The types of businesses generated by commercial entrepreneurs span the entire range from single person small businesses to vast multinational corporations. There are two other types of entrepreneurship that are somewhat pertinent to this discussion. These are corporate, public and social entrepreneurship.

Corporate entrepreneurial activities are defined as the firm’s sum of innovation, endeavor, and strategic renewal [3]. The question of why some firms engage in corporate entrepreneurship more than others is a very active subject of research [4]. Regardless, overall corporate entrepreneurship is highly pertinent to the entrepreneurial spirit of the employees. The ability to function as an entrepreneur within a corporation is known as Intrapreneurship and is a very important consideration for educators given that the vast majority of the students (even Entrepreneurship students) will go on to work in large corporations.
Public entrepreneurship is also investigated in the literature [5]. Public entrepreneurs are not concerned with creating products and services, but aim to slowly reinvent their organizations in the hope of improving government effectiveness and efficiency. Many students seek careers in public service and the effects of Entrepreneurship education cannot be dismissed.

Social entrepreneurship is defined as “an innovative, social value creating activity that can occur within or across the nonprofit, business, or government sectors” [6]. Sahlman [7] set up a model to depict the crucial elements of commercial entrepreneurship, thus laying the foundation for developing a framework for social entrepreneurship. This model exemplified the interrelationship among its four components: people, context, deal, and opportunity. The primary difference between commercial and social entrepreneurship is based primarily on the goals and not the process. Commercial Entrepreneurship seeks to maximize profits while Social Entrepreneurship is focused on maximizing social benefits while being satisfied with economic subsistence. Non Governmental Organizations (NGOs) internationally and non-profit organizations in the US are the primary models here. Social Entrepreneurship has gained a great deal of publicity with the success of the microcredit industry and the bestowing of the Nobel Peace Prize on the champions of that industry, Mohammad Yunus and Grameen Bank.

In the last few decades, business entrepreneurship education has gone from relative obscurity to more than 2,200 courses at over 1,600 schools, 277 academic positions, 44 refereed academic journals, conventional management journals dedicating special issues to entrepreneurship, and over 100 established and funded centers [8, 9]. Moreover, a brand of entrepreneurship that is based primarily in colleges of engineering and science has taken hold in many universities. This Technical Entrepreneurship emphasizes, among other things, innovation as a mean to create competitive products and services that are technical in nature. The product of this type of education is the much sought after “high-tech startup” which is very common in technology parks that surround many major universities.

Technical Entrepreneurship is the marrying of business entrepreneurship principles with high technology work. There have been two models of how universities chose to educate engineering and science students in that field. The first involves teaming technology graduate and advanced undergraduate students with MBA students [10, 11]. Other programs aim at providing their science and technology a modicum of entrepreneurship training through a limited number of courses [12, 13].

Entrepreneurship has a collaborative nature. This characteristic would demand collaborative and interdisciplinary projects and case studies when teaching entrepreneurship. Reader et al [14] explored the structure of entrepreneurship using co-author analysis. This established a cooperative structure of the entrepreneurship literature from its research members’ point of view. The co-citation frequencies of 78 leading entrepreneurship researchers were analyzed using multivariate techniques. Exploring the intellectual structure of entrepreneurship research was carried out through cluster analysis and multidimensional scaling to identify groups of researchers with similar work. They concluded that within entrepreneurship research, strong social and collaborative ties exist.

Computational thinking paves the way for entrepreneurial thinking. For the future, we will need graduates that combine their computational thinking with entrepreneurial thinking. Our understanding of the essence of entrepreneurial thinking assists in isolating experiences related to decisive profound beliefs inherent to our mental models of entrepreneurship. Krueger [15] stated that “Cognitive developmental psychology and constructivism offer possibilities for the future of entrepreneurial cognition research to explore.”

The above mentioned literature will inevitably have its impact on shaping the design of our Software Entrepreneurship course. In what follows, we will complement this with education-related literature.

Rohde et al [17] and Klamma et al [18] concentrated on establishing communities of practice among students and start-up companies. To this extent, they designed the course “High Tech Entrepreneurship and New Media” in an effort to maintain the process of social capital–building between entrepreneurs and students. They provided projects based on the experience of local start-up companies. They concluded that having start-up entrepreneurs involved in the group projects would simulate the market-oriented perspective within the course.

A project, “Managing Innovation in the Digital Economy,” to augment the undergraduate and
graduate information management and information systems curricula with entrepreneurial ideas was introduced by Foster et al [19]. They conjectured that “differences in levels of prior knowledge of business studies and in cultural background can impact on students’ acquisition of domain knowledge and intellectual and information research skills during collaborative development of a business plan.”

The vast majority of computer science and software engineering programs do not offer entrepreneurial courses. Rusu et al [20] proposed a framework to allow computer science and software engineering students the opportunity of experiencing and igniting the spirit of entrepreneurship early in the program. Their framework included the following phases; identifying potential enterprising, assigning students to entrepreneurial mentors, practicing entrepreneurial skills within a software engineering course, and having students develop their own entrepreneurial ideas.

Kussmaul [21] proposed a team project within their Systems Design and Implementation course to emphasize software entrepreneurship. In addition to the main goal of working in a significant software projects and exercising software engineering and project management, students will also be exposed to ideas and concepts of entrepreneurship within the Systems Design and Implementation course.

In this paper, we introduce the design of an entrepreneurship course for computer science and software engineering. Software has great potential for entrepreneurship. Details of course rationale, description, objectives, outcomes, projects, case studies, and assessment will be emphasized.

2. COURSE RATIONALE

The main motivation behind developing this course stemmed from the fact that future computer science and software engineering courses will be based on computational thinking. If computational thinking is pursued, entrepreneurship will be the direct beneficiary of the created ideas and approaches. Other factors that have driven our course design are:

- Developing new software markets and renovating current markets. When a computer science/software engineering graduate (entrepreneur) establishes a new software market, the economic situation and the national income will be improved. In addition, the new software markets will have their impact in improving our curriculum to meet the requirements of the new/renovated market.
- Introducing new software technologies and products. As every aspect of life relies on software, new software technologies and products will further improve the way industry, government, health, and education do business. New technologies and software products are necessary for our nation to continue to lead the IT revolution worldwide. This is one side of the coin. The other side deals with the impact of these technologies and products in improving and modernizing the teaching of both computer science and software engineering. They will force us to review and revise the courses we offer, and either improve some courses or develop new courses to produce graduates ready to apply these technologies and products.
- Creating further computing employment. New software market, technologies, and products will supply more computing jobs, and thus, increasing the demand for computer science and software engineering graduates.
- Boosting the interest in computer science and software engineering programs. Successful young software entrepreneurs can be great examples for high school students to imitate. If software entrepreneurship is encouraged and enhanced, both the success of the entrepreneurs and the computing jobs they create will help in attracting students for both programs.
- Minimize the impact of offshoring on the software business and the national economy. Entrepreneurs are not scared of offshoring as they have their own companies. In addition, they create many jobs for their colleagues. Through time, our graduates will not be put off by offshoring.
- Adding further healthy competition in the software industry. Healthy software competition drives the market effectively. This competition will result in new and enhanced software market, technologies, and products. In addition, this healthy competition will certainly intensify innovations.
• Encouraging more research in software entrepreneurship. Research in entrepreneurship has its roots in business schools. The vast majority of published papers are from these schools. We anticipate that when our students are equipped with entrepreneurial thinking and skills, software entrepreneurship research will improve.

• Developing entrepreneurial bias among graduates. When students are introduced to software entrepreneurship, the probability of graduating more software entrepreneurs increases. We refer to this increase as “entrepreneurial bias.”

• Increasing the number of software leaders. Software entrepreneurs are armed with enthusiastic vision. They strive to be positive, courteous, persistent, and enthusiastic. They are innovative and highly credible people who have the ability, the desire, the energy, and the vision to take their business beyond its current level of achievements. Furthermore, they accept the challenge and risks of starting a new business and create the conditions and atmosphere to achieve their goals.

3. COURSE DESCRIPTION

The entrepreneurship course presents the essential concepts, methods, techniques, theories, models, and practice of entrepreneurship, which are vital for computer science and software engineering. To this end, this course will articulate the following:

- Software entrepreneurial process
- Leadership characteristics and styles
- Principles of software business ownership
- Entrepreneurial software marketing
- Software business communication and negotiation techniques
- Feasibility analysis
- Accounting Management for software business
- Small software business management
- Funding approaches technique
- Entrepreneurial financing
- Software market opportunity identification
- Software business law
- Demand forecasting
- Software product discovery
- Software business plan development

- Software entrepreneurship risk management
- Rules and regulations
- Computational and entrepreneurial thinking
- Software Innovation
- Software product development
- Social entrepreneurship
- Commercial entrepreneurship
- Public entrepreneurship
- Corporate entrepreneurship
- Software entrepreneurship case studies and project

As entrepreneurship is characterized by collaboration and interdisciplinary, the course will promote collaborative and interdisciplinary teams.

4. COURSE OBJECTIVES

The software entrepreneurship course aims to introduce the concepts, techniques, standards, skills, theories models, methods, and practice necessary for either starting a new software business or innovating an existing one. In addition to course lectures, students will carryout collaborative team projects, work on case studies, and be exposed to entrepreneurship research. Furthermore students will attend talks presented by software entrepreneurs, and watch videos on various aspects of entrepreneurship. The following are the objectives of the course:

1. To provide students with concepts, theories, techniques, models, and methods of entrepreneurship.
2. To promote entrepreneurial thinking and innovational thinking within the field of software.
3. Present the techniques and process of identifying new software business opportunity.
4. To study the approach taken in analyzing and managing risk.
5. To introduce various types of entrepreneurship.
6. To present the concepts, techniques and theories of marketing, accounting, financial analysis, and planning as related to software.

5. COURSE OUTCOMES

Upon completion of the Software Entrepreneurship course, students will be able to:
1. Understand the concepts and techniques of entrepreneurship.
2. Identify a vital software business opportunity.
3. Prepare business, marketing, financial, and risk management plans for a software firm.
4. Think entrepreneurially.
5. Grasp leadership styles and techniques.
6. Perform software business negotiation techniques.
7. Carry out new software business feasibility analysis.
8. Secure funds for their software business.
9. Forecast the demand for their new technology or product.
10. Think innovatively.

6. PROJECTS AND CASE STUDIES

The projects and case studies planned for this course will contribute to promoting the entrepreneurial thinking among students. Both the projects and case studies will be team-based activities. Whenever possible, teams will be interdisciplinary. It is our desire to have teams consisting of students from business, engineering, and computer science.

All the projects will be of an applied nature. Projects will be drawn from various computer science and software engineering courses, but with one critical exception regarding the deliverables. In the traditional courses, deliverables are mainly based on algorithms, programming languages, computing tools, and design. For the entrepreneurship project, the deliverables are guided by entrepreneurial and computational thinking. We require students to use the computational skills and think innovatively to propose a software product that is characterized by practical and applied value, and their entrepreneurial skills to forecast the demand, identify a critical software business opportunity, and plan for a new software business to market the proposed product. The teams will be mentored by faculty teaching the course and an entrepreneur, preferably one of our alumni. Teams are required to prepare a feasibility study, in addition to business, marketing, financial, and risk management plans. Furthermore, teams will exercise securing funds and negotiations. For this purpose, the mentors will play the role of external organizations’ officials.

Case studies will reflect both successful and unsuccessful endeavors. For the successful one, teams are asked to specify what factors resulted in the success and suggest ways to improve the enterprise. For the unsuccessful endeavor, teams are supposed to detect what risks the enterprise failed to consider (perform risk re-analysis), specify what went wrong, and devise a rescue procedure.

7. COURSE ASSESSMENT

Our Software Entrepreneurship course will be assessed by faculty (possibly interdisciplinary faculty), and external mentors. The following techniques will be used to assess students’ performance:

- Exams: Students will be subjected to multiple-choice questions to measure their entrepreneurial and innovational thinking skills. Both the mentor and faculty will participate in devising these exams.
- Guest Speaker Talks: Upon listening to a guest speaker talk, students are required to write a report covering lessons learned, existing risks and problems, and possible improvements.
- Projects: A software-based entrepreneurial project will be provided with the goal of assessing the points mentioned in the course outcomes.
- Case Studies: Whenever possible, case studies will reflect real entrepreneurial cases. It is planned to offer two case studies per term.
- Research Paper: A number of entrepreneurial topics will be made available during the first day of class. Students are supposed to select a topic from the list of topics they feel interested in, and collect as many papers as they can (minimum five) on the topic. The papers should be taken from journals, books, and conferences. The web could also be used provided the work has been published. Instructions for writing up the paper will be provided.
- Videotapes: Students will view the processes followed by entrepreneurs and innovators and provide critical analysis of the video in a manner similar to case studies.
The weights given to the above mentioned assessing techniques are as follows:

<table>
<thead>
<tr>
<th>Technique</th>
<th>Weight</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exam</td>
<td>15%</td>
</tr>
<tr>
<td>Project</td>
<td>40%</td>
</tr>
<tr>
<td>Case Studies</td>
<td>15%</td>
</tr>
<tr>
<td>Research Paper</td>
<td>10%</td>
</tr>
<tr>
<td>Guest Speaker</td>
<td>10%</td>
</tr>
<tr>
<td>In-class Video</td>
<td>10%</td>
</tr>
</tbody>
</table>

Table 1: Assessment weights

8. CONCLUSION

It is extremely hard to break the ties between computational thinking and entrepreneurial thinking as they complement each other. Computational thinking will govern the process of innovation to produce new software products and technologies, and entrepreneurial thinking will establish the foundations for marketing these products and technologies. With this in mind, we introduced a design of a software entrepreneurship course with the goal of bridging the gap between creating products and creating marketing opportunities.

REFERENCES


