Ten Years of the Agile Software Factory for Software Engineering Education and Training

Robert C. Green II and Joseph T. Chao

Department of Computer Science
Bowling Green State University
Bowling Green, OH
{greenr; jchao}@bgsu.edu

Abstract—The Agile Software Factory (ASF) was created in the Department of Computer Science at Bowling Green State University nearly ten years ago to promote the use of agile methods, improve student education through service-learning opportunities, and to facilitate community engagement by handling the many software requests that regularly arrive in the department. In pursuit of these goals, the ASF has had many successes and faced challenges in a variety of areas. This paper describes the history of the ASF along with the multiple successes and challenges that it has faced over the years. Future plans to continue fulfilling the educational mission of the ASF while widening its impact are also discussed, including the creation of a student mentorship program, promoting software engineering research, and sponsoring software engineering related events.

Keywords—agile software development; software factory; software engineering; service learning; real-world projects

I. INTRODUCTION

Nearly ten years ago, the Department of Computer Science (CS) at Bowling Green State University (BGSU) had a pressing need: Requests for software development from real-world clients, community members, and researchers across campus kept arriving in the department, but could only be pursued on occasion due to time and resource constraints. To handle these requests, the Agile Software Factory (ASF) was created with a $21,000 startup fund from the Agile Alliance 1) To serve as a clearing house for such projects, 2) To act as a venue to bolster student experience, practice, and education through service learning, and 3) To encourage Agile practices in software development and engineering [1]. Since that time, the ASF has participated in over 40 software development projects, employed over 40 students, and evolved a new funding model that is based on corporate sponsorships.

Fundamentally, the software factory model seeks to integrate the academic and industrial sectors in order to improve the educational experience of students while simultaneously providing benefits to universities and business [2] [3] [4]. In fact, there a variety of software factories currently in existence in both Europe and the US that have successfully undertaken multiple collaborative projects [5]. For example, a group in the University of Helsinki created a software factory to support education, research, and entrepreneurship while directly integrating the unit as part of the Master's level capstone project [6] [7]. In addition, others have also used the software factory model alongside the classroom to perform research on aspects of agile development, such as self-organized learning [8] and the impacts of motivation [9] on student learning. Related to the software factory model, many works have discussed the experience of integrating real-world clients/projects into various courses in their curriculum [10]. Others have varied on this theme by having students develop projects to aid other students in learning computer science while using real-world methods and tools [11].

The remainder of this review will focus on the experience of creating, maintaining, and growing the ASF at the Department of Computer Science at BGSU by discussing the general operating model of the ASF, reflecting on various accomplishments and challenges experienced, and concluding with a discussion of the road ahead.

II. OPERATING MODEL

The ASF follows a clean operating model that combines personnel, projects, and funding. In terms of personnel, the Director of the ASF, a service role within the department, who takes on all aspects of management and is responsible for obtaining projects, funding, and managing workers. The Director of the ASF is also responsible for submitting an annual report to the Faculty of the CS Department detailing all activities from the previous year. Reporting directly to the Director of the ASF are multiple student employees who are paid at an hourly rate and work a fixed weekly schedule of (typically) no more than 20 hours per week. These students come from a variety of backgrounds – both CS and non-CS – and acquire their positions through a standard interview process. There have been no regular employees other than students.

To maintain an effective level of personnel, obtaining funding is a key aspect of the ASF and a core responsibility of its Director. Historically, this funding has come from a variety of sources – paid projects, grant funding, donations, etc. – that began with a $21,000 grant from the Agile Alliance. As maintaining continuous funding has become a challenge, this issue will be discussed in detail in Section IV. It should be noted that the ASF does not market the service directly to for-profit corporations, and does not compete with other software business or developers.

In addition to personnel and funding, fundamental to the operation of the ASF is having multiple, targeted projects on which to work. Projects developed in the ASF have historically come from a variety of sources including requests from non-

profit organizations, private individuals or corporations, and the faculty, staff and students at BGSU. Each project request goes through an evaluation and negotiation process that is driven by the Director of ASF. The result of this process is one of the following outcomes:

1. *Paying Projects.* Paying projects involve project definition, estimation, quoting, and project contracts that come from private entities. Throughout the years, the ASF was contracted to develop both proof-of-concept and prototypes of various projects, and to do software jobs related to web development, data analysis, tool development, or grant funding. A key example of this is the BGSU Campus App, which was developed and maintained by the ASF for multiple years, until it was outsourced in the summer of 2016. Note that this is the only option for a for-profit entity's project to be accepted. These projects provide agile software development opportunities for students while, at the same time, satisfying the needs and requirements of the ASF's internal and external clients.

   In terms of grant funding projects, they are generally related to active research that is being pursued on campus. It is not uncommon for researchers to contact the ASF before submitting a grant proposal to discuss the inclusion of software development by the ASF in the budget. A clear example of this was the *How Many?* Android Application that was used for subsidizing research. It is also possible for the ASF to develop a proof-of-concept before submission of a grant.

2. *Service-Learning Opportunities.* Often, not for profit organizations (or their members) approach the CS department or ASF with a demonstrated need. Based on an evaluation of the ASF Director and available resources, these projects will be taken on at no cost to the individual or organization.

3. *CS 4540/5540. Software Engineering Project.* Projects with a reasonable scope and the appropriate amount of challenge may become a project in this course. The course is designed as an elective course in the BS in Computer Science degree and is a required course in the newly developed BS in Software Engineering degree at BGSU. Offered on a yearly basis, the course solicits real projects from real-world clients (mainly not for profit entities). During the course, each class is divided into software development teams that apply Agile methods to develop the software project while working with the client and being supervised by the instructor. At the end of the course, all artifacts are archived and hosted within the ASF, which guarantees one semester of service to maintain the newly created software. This helps to provide a mechanism for maintaining software developed in this course.

4. *Student Projects.* When some projects are beyond the scope of the CS 4540/5540 course and may not be a good fit for development in the ASF, they may be passed on to individual undergraduate or graduate students to be completed as either independent studies or MS level projects.

5. *Rejection.* If the project cannot be tailored into a suitable class project, ASF does not have the capacity to take on a pro bono project, a student cannot be found to work on the project, or the client cannot afford the software development cost, then the project is rejected.

Once a project is brought inside the ASF for development, a generalized Agile methodology is followed that leverages modern software including git and Gitlab for version control and project management, Slack for communication, and a variety of other tools including IDEs, etc.

III. ACCOMPLISHMENTS

Over the past ten years, the ASF has achieved much in terms of its goals and priorities. Some of these have included:

1. *Classroom Support.* The ASF has provided maintenance services for software developed in the CS 4540/5540 course as well as guidance and mentoring to students currently developing projects as part of this course. This improves the educational experience in this course in two ways. First, as clients have a guaranteed support system after development of their software, making recruiting of clients/projects a less strenuous task for the instructor while allowing the project to remain educational and not solely focused on production. Second, students also have a built-in support system where they can go to other more experienced students (not just the instructor) to ask questions about the entire software development process and specific technologies.

2. *Service and Training.* Benefits to students include the addition of a service-learning component to existing software development courses and the hands-on training and experience that comes from working within the ASF.

3. *Marketability.* The ASF has helped in increasing the visibility of the CS program to potential students and outside entities.

4. *Research productivity.* The ASF has produced software, educational experiences, and training experiences that are related to at least 13 academic publications. In addition, multiple employees of the ASF have worked directly to develop research-related software for faculty members outside of the ASF. The ASF has also supported additional software systems that directly support research.

Overall, the ASF has developed and/or supported over 40 different software projects during its existence. A few of the more recent and significant projects include applications like:

1. BGSU Campus Mobile App which served as the main mobile application for BGSU for multiple years. This app was given regular financial support by BGSU ITS until the Summer of 2016;

2. KnoWare, a mobile platform designed for performing citizen science was developed with a key goal of improving water quality in the Toledo Metro area [12];

3. Virtual Token Economy, an educational app designed to motivate younger students to improved learning, was developed in conjunction with the University of Memphis;

4. How Many?, a game like app targeted at improving the mathematical ability of students to subitize at a young age, was developed as a research tool; and

5. Townie Cup, a website and golf tournament application for a local charity organization.

Considering the variety of projects undertaken by the ASF, one item of note is that there are often "repeat customers". For instance, a faculty member from the College of Technology has worked with the ASF via the CS 4540/5540 course on three separate projects over four semesters, including two research/teaching oriented projects (Sing N' Sketch and EASEL) and one non-profit effort (Our Food Farm). He continues to fund one CS undergraduate student who is continuing work on the EASEL app and previously supported an undergraduate student to continue working on the Sing N' Sketch application as an Undergraduate Research project.

Many of the projects developed in the ASF are under continual use by a variety of entities. Some examples of these include the Life Tracker project (currently in active research use by a professor in the Department of Psychology), a Musical Instrument Inventory system that was developed for the College of Musical Arts at BGSU, and the previously mentioned GIS application that was developed for the Center for Regional Development.

IV. CHALLENGES

In conjunction with the successes and accomplishments of the ASF over the past ten years have come a variety of challenges. Like most other organizations, funding/budget is often an ongoing concern for the ASF. Being in an academic setting also presents its own set of challenges such as hiring, training, and working with student employees. Managing client expectations under academic scheduling constraints is also challenging, particularly when aimed at the goal of fulfilling educational missions and satisfying clients simultaneously. This section will discuss some of these issues.

A. Funding Model

One key issue related to any entity like the ASF is the continual funding of personnel and equipment. Over the years, the ASF has had support and success from various funding sources including university and departmental resource support, grant funding, and project-based funding. This has allowed for the active employment of between two and nine students at any given point in time. The ASF does not receive any institutional budget but must seek out funds from projects, grants, and/or sponsorships. This adds a significant responsibility to the faculty member assigned as the Director of the ASF.

The initial start-up grant of $21,000 from the Agile Alliance and the first major project (a commercial project developing more than a hundred online learning modules for a large international bank for $120,000) laid a strong financial footing for the ASF.

In the years after this, the ASF continued to work on some paid commercial projects. However, due to the complexity of commercial contract development and intellectual property concerns, the ASF has been shying away from commercial projects and focusing on service-learning projects centered on university departments, programs, and research as well as pro-bono projects for non-profits. In recent years, the number of internally sponsored projects has decreased due to the increasing budget constraints many universities have been facing. Hence, the department created the ASF corporate membership model in 2015 after many years of preparation.

The corporate membership program was developed as a method for private entities to show their support for the ASF and the Department of Computer Science through monetary donations that benefit these same entities. The corporate memberships support many facets of the ASF, including scholarships, student wages, hardware and software for development projects, and professional training and development. Currently the ASF has seven corporate members at various support levels.

Due to the hard work of the ASF Directors and the combined generosity of sponsors, the ASF has been self-sustained financially since inception and is also able to continue pursuing its educational mission with revenue generated from various entities.

B. Project Management Concerns

In the realm of Software Engineering, project management of teams of any size can be a challenge. Managing student workers focused on multiple projects in an academic environment provides its own unique layer of complexity on top of this already challenging area. Some of the main issues experienced by the ASF over the years include:

1. *Working with Student Employees.* Student employees provide a wide variety of challenges. One such issue is a lack of skills and maturity. While more and more students come into CS programs with some level of programming skills, the majority of students are not exposed to any serious programming or software development before entering the CS program. In addition, when hiring a student in their second or third year of college, their technical skills are often not ready for professional work and they may or may not have the maturity and work ethic that are required of professional software developers.

Another issue that is unique to the academic enviroment is that all student employees are short term employees due to the nature of their degree program. Consider a typical case where a student is hired during the beginning of their junior year. After some training and mentorship, they (at the most) are able to contribute to the ASF for roughly 2-3 semesters.

A variant on the issue of short term employees is the hiring of Graduate Assistants (GAs). The ASF often receives project-based GA funding from internal project sponsors. Some programs on campus have GA budgets for projects but do not have cash to cover the software development cost within the ASF. As such, the potential client will often sponsor a GA position. The GA typically requires extensive training, works on a specific project, and leaves

after the contract period. Nevertheless, the hiring of a GA requires the same amount of management.

An additional issue is overall project management and guidance as the Director of the ASF also serves as the project manager for all ASF projects. While project management is, in itself, a full-time position, the Director also has the additional teaching, research, and service load of being a professor and/or instructor. Some of this burden is relieved as the position is considered a major service position within the CS department and includes a course release.

2. *Managing Client Expectations.* In addition to managing students, interactions with and expectations of clients must also be managed by the ASF Director. When compared to a typical software development company or freelancer, the ASF functions solely on part-time student employees, is closed between May and August due to 9-month faculty contracts and summer internships, and accomplishes work at a much slower rate as the mission of the ASF is not to make profit, but to educate students in software engineering practices through hands-on engagement. In addition, the easiest (and also a major) pathway for ASF projects, the CS 4540/5550 course, is typically only offered during one semester of the year (historically the spring) and results in software projects that are not yet at the professional level. These often leave clients with a long wait until their project begins development in the course followed by an additional summer semester of waiting for continued development.

3. *Leadership and management change.* As neither the University nor the Department of CS has the budget for hiring a full-time director for the ASF, the Director of the ASF is a service assignment that is given to a willing faculty member in the department. This means that it is a volunteer position requiring a considerable time commitment. In recent years, the position has come with a single course release each year. As previously mentioned, the position can easily be a full-time job, which leads to difficulty in finding a faculty member who is both capable and willing to serve in the position. This leads to changes of leadership/management that can be both beneficial and detrimental.

4. *Fulfilling Educational Mission.* Recently, the CS Department raised the question that, with the significant amount of time and effort put into the ASF, has it produced sufficient student and/or program impact to make it worthwhile? Some argue that with 2 to 9 student employees at any given time (in a department of 300+ students), the student impact seems limited. Others argue that the ASF provides a valuable experience for those involved and a unique opportunity for student recruitment and marketing. While there is truth in both of these arguments, the discussion is ongoing. One commonly agreeable item is that the ASF should seek to increase educational impacts to make the effort more worthwhile.

V. LESSONS LEARNED

Undertaking an endeavor like the ASF is a significant learning experience, both for faculty and students. Based on the many successes and challenges previously discussed, much advice may be given to anyone attempting to form a similar organization. Some of this guidance includes:

1. *Give the Director of ASF ample time to supervise and manage the organization.* Faculty members are typically very busy under a high burden of teaching and research. Adding an additional service role, even when it comes with a course release, is a significant burden as such a position could easily be a full-time position. Alternatively, hiring a part-time director for the ASF should also be considered.

2. *Avoid industrial projects, but seek industrial partnerships.* An organization like the ASF should not compete with professional software companies both because the focus is on student learning and because the development pace and quality is lower than the typical professional grade. Instead, find ways to allow corporate entities to contribute funds while gaining some benefit such as recognition, access to career fairs, and/or additional/early exposure to students.

3. *Focus on the non-profit and research sectors.* Following from the previous point, entities of this type should be centered on student experiences while serving the community. Providing services to non-profit organizations is fulfilling for student employees and of great benefit to the community. Developing research software enables undergraduate research and helps students gain interest in post-baccalaureate studies. This also allows the entity to participate in grant seeking with the potential of obtaining funding. However, this approach will require identifying other funding sources.

4. *Plan for transition.* In entities like the ASF, the transition of students every one to two years is guaranteed. This requires a type of transitioning program to pass on institutional memory as well as development processes and practices. As the Director will also change with some regularity, a similar process should be developed.

5. *Software development infrastructure is a key to quality.* One of the key reasons that the ASF has succeeded is that infrastructure – computing equipment, servers, hosting, and office space – has regularly been provided through University support. Without such support and regular infrastructure, the ASF would continually struggle to maintain projects and hosting. In short, build a relationship or agreement with university technology services in order to ensure the stability of infrastructure for the long haul.

VI. THE ROAD AHEAD

While the past ten years have shown success, there are still questions to be answered and progress to be made with regards to the ASF. Currently, there are several plans in development which are discussed in this section.

A. Mentorship Program

The ASF plans to launch a mentorship program which will involve ASF student employees (mentors), CS students (mentees), and CS faculty. The program objective is to widen the impact of the ASF by offering more students additional learning opportunities while providing them with the real-world, hands-on experience needed and valued by today's employers. The program will utilize a one-on-one approach and will be open to all current BGSU CS students meeting the program requirements. The hope is that the program will allow mentors to "give back" to the ASF program and to their fellow students, learn coaching skills, encourage the sharing of knowledge which increases sense of self-worth and confidence, strengthen interpersonal relationship skills, increase personal satisfaction with the job, and foster leadership skills. Mentees will also benefit from increased self-confidence, improved interpersonal relationship skills, and opportunities for networking and experiential learning.

B. Promote graduate and undergraduate research and scholarship in software engineering.

Becoming a national model for a premier learning community is the vision of BGSU, and it is the plan for the department to become a leader in Software Engineering education. Stimulation of active student participation in meaningful research and scholarship is critical to this mission. The ASF plans to expand its function as a software project clearing house for the community to also include projects centered on research and scholarship. In becoming a research project clearing house, the ASF will create a database of research projects that can match student, faculty, and project sponsors in a collaborative manner. The ASF also plans to offer workshops in promoting research and scholarship, facilitate research colloquia in stimulating intellectual creativities, and host events in encouraging collaboration.

C. Sponsor software engineering related events

With the established infrastructure, the ASF can play a significant role in cultivating the culture of learning by sponsoring interesting and meaningful events for students. The ASF plans to work with the current ACM student chapter closely and collaboratively on expanding existing events and creating new and interesting events. The potential events could be sponsored by the ASF could include the Hour of Code, Hackathons or programming contests, and various career development events such as training/demonstration sessions/seminars. One item of significant interest is the integration of the ASF with the new Software Engineering Degree that the department will begin offering in the Fall of 2017. While the ASF has seen significant progress and success over the years, it can continue to play a major role in growing the Software Engineering program.

VII. CONCLUSION

Over ten years, the ASF has proven to be a valuable tool for enhancing Software Engineering education and training at the Department of Computer Science at BGSU while promoting software engineering research and community engagement. Along with many successes and accomplishments (continued funding, completed projects, and, most importantly, improved education experience for students), there have also been major challenges along the way. Yet, as time moves on, future plans for mentorship, training, and active engagement in the new Software Engineering degree program are laying a path to continued success for the ASF.

REFERENCES

[1] J. Chao and M. Randles, "Agile Software Factory for Student Service Learning," in *22nd Conference on Software Engineering Education and Training*, Hyderabad, India, 2009.

[2] J. Chase, E. Oakes and S. Ramsey, "Using live projects without pain: the development of the small project support center at Radford University," in *SIGCSE technical symposium on Computer science education*, Covington, Kentucky, 2007.

[3] F. Siqueira, G. Barbaran and J. Becerra, "A Software Factory for Education in Software Engineering," in *Conference on Software Engineering Education and Training*, Charleston, South Carolina, 2008.

[4] V. Cicirello, "Experiences with a real projects for real clients course on software engineering at a liberal arts institution," *Journal of Computing Sciences in Colleges*, vol. 28, no. 6, pp. 50-56, 2013.

[5] D. Taibi, V. Lenarduzzi, M. O. Ahmad, K. Liukkunen, I. Lunesu, M. Matta, F. Fagerholm, et. al., ""Free" Innovation Environments: Lessons learned from the Software Factory Initiatives," in *The Tenth International Conference on Software Engineering Advances*, Barcelona, Spain, 2015.

[6] P. Abrahamsson, P. Kettunen and F. Fagerholm, "The Set-Up of a Software Engineering Research Infrastructure in the 2010," in *International Conference on Product Focused Software*, Limerick, Ireland, 2010.

[7] F. Fagerholm, N. Oza and J. Münch, "A platform for teaching applied distributed software development: The ongoing journey of the Helsinki software factory," in *International Workshop on Collaborative Teaching of Globally Distributed Software Development*, San Francisco, California, 2013.

[8] X. Wang, I. Lunesu, J. Rikkila, M. Matta and P. Abrahamsson, "Self-organized Learning in Software Factory: Experiences and Lessons Learned," in *Proceedings of the Agile Processes in Software Engineering and Extreme Programming: 15th International Conference*, G. Cantone and M. Marchesi, Eds., Cham, Springer International Publishing, 2014, pp. 126-142.

[9] J. Urquiza-Fuentes and M. Paredes-Velasco, "Investigating the effect of realistic projects on students' motivation, the case of Human-Computer interaction course," *Computers in Human Behavior*, vol. 72, pp. 692-700, 2017.

[10] C. Murphy, S. Sheth and S. Morton, "A Two-Course Sequence of Real Projects for Real Customers," in *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education*, Seattle, Washington, 2017.

[11] V. Cicirello, "Student developed computer science educational tools as software engineering course projects," *Journal of Computing Sciences in Colleges*, vol. 32, no. 3, pp. 55-61, 2017.

[12] J. Chao, K. Parker and A. Fontana, "Developing a Social Media Interactive Learning Environment," *Issues in Informing Science and Information Technology*, vol. 8, pp. 323-334, 2011.

[13] J. Storer, J. Chao and A. Torelli, "KnoWare: A system for citizen-based environmental monitoring," *Informing Science: The International Journal of an Emerging Transdiscipline*, vol. 19, pp. 125-139, 2016.

[14] G. Duncan, C. Dowsett, A. Claessens, M. K. A. Huston, P. Klebanov, L. Pagani and e. al., "School Readiness and Later Achievement," *Developmental Psychology*, vol. 43, pp. 1428-1446, 2007.