

Applying Scrum to Manage a Senior Capstone Project

Dr. Zesheng Chen, Indiana University Purdue University, Fort Wayne

Dr. Chen is an assistant professor in the Department of Computer Science at Indiana University - Purdue University Fort Wayne. He received his M.S. and Ph.D. degrees from the School of Electrical and Computer Engineering at the Georgia Institute of Technology in 2005 and 2007. He also holds B.E. and M.E. degrees from the Department of Electronic Engineering at Shanghai Jiao Tong University, Shanghai, China in 1998 and 2001, respectively. He worked as an assistant professor in the Department of Electrical and Computer Engineering at Florida International University from 2007 to 2009. He moved to Fort Wayne in 2009 and worked as a limited term lecturer in the Department of Engineering and in the Department of Computer Science from 2009 to 2015 at Indiana University - Purdue University Fort Wayne. He also worked as a software engineer at TransWorks from 2012 to 2015.

Applying Scrum to Manage a Senior Capstone Project

Abstract

Software project management is essential for the success of a software engineering project. In this paper, we describe in detail how we applied the Scrum methodology to manage a senior capstone project that started from Summer 2015 to the end of Spring 2016 at Indiana University-Purdue University Fort Wayne (IPFW). Specifically, we point out that the Scrum principles, such as working closely with customers, using incremental development with online tools such as Pivotal Tracker and GitHub, and running weekly Scrum, contribute significantly to the success of this project. The completed final software product was put into public use and received positive comments from satisfied customers. The students in the project also gave excellent feedback on the guidance of the faculty advisor and their design experience.

Introduction

A senior capstone project is an excellent way for computer science (CS) students to work on a real product and acquire real-world experience. From Summer 2015 to the end of Spring semester in 2016, our senior design team built a Web application for the Department of Audio Reading Service at the Allen County Public Library (ACPL) in Fort Wayne, Indiana. The senior design project is based on two courses in two consecutive semesters: CS 460 (Senior Capstone Project I) in the Fall semester and CS 465 (Senior Capstone Project II) in the Spring semester. Our students used the Summer time in 2015 to meet with ACPL to collect requirements.

The goal of this senior capstone project is to design a Web application to be used by ACPL to collect the information of library users with disabilities and assign audio radios to them. Such users may have stroke, Parkinson's, and arthritis, which make it difficult to hold a book or turn pages. Other users may be any individual who is unable to read conventional printed materials because of blindness, low-vision, or a literacy issue. The project needs to provide an easy-to-access interface for users to input their data such as access to service, service type, demographics, and customer satisfaction. Moreover, the project should deliver Web functionalities that allow ACPL to store, retrieve, and analyze users' data (*e.g.*, users' profile and other survey data) through Web browsers and mobile devices. The designed Web application is expected to help ACPL better serve users with disabilities.

The system architecture of our designed Web application is illustrated in Figure 1. A user or an organization can apply for the audio service by filling out an online form. Once the form is submitted, the data are stored in the database. An administrator at ACPL can log into the system through the Web interface and manage the system. The functionalities for the administrator interface include searching, reading, creating, and updating users' and radios' profile, as well as assigning audio radio(s) to a user or an organization.

The main challenges to implement such an architecture lie in two aspects:

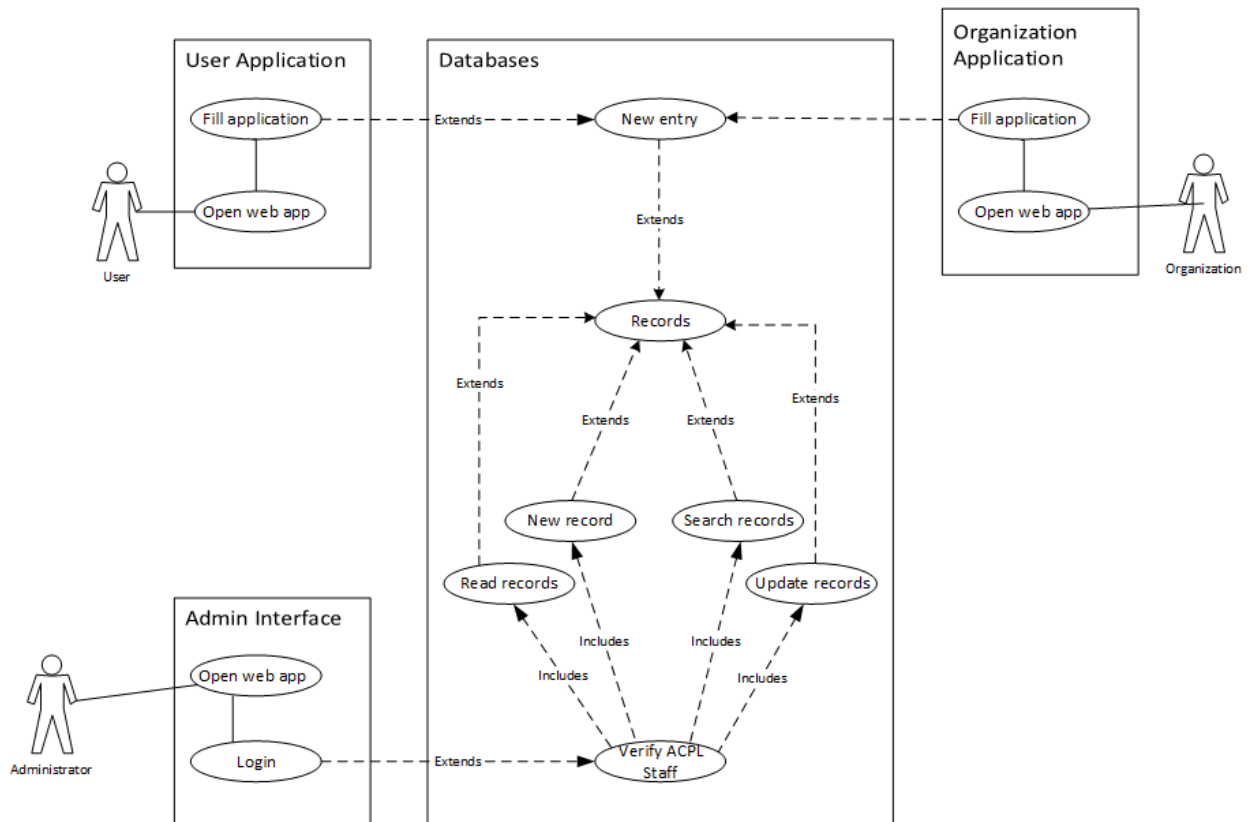


Figure 1. System Structure of Our Web Application

- (1) Ease of use. The information for both the user's interface and the admin's interface must be displayed in a user friendly manner. Especially, the user's interface should be accessible to people with visual or reading impairment. Moreover, the functionalities must be beneficial and practical for ACPL.
- (2) Eliminating errors. This project will not be continued by other students in the future. Therefore, the software system must be as bug-free as possible.

To overcome these challenges, we apply the Scrum methodology in our project. Moreover, we consider Americans with Disabilities Act (ADA) compliance standards [1] [2] and ISO/IEC/IEEE International Standard 23026-2015 [3] in order to make our designed Website more accessible by people with disabilities.

The remainder of this paper is structured as follows. We first discuss the issues with the traditional project management method and introduce the Scrum methodology. We then show the details of how we apply the Scrum principles to our senior capstone project. We further describe the evaluation and outcomes of the project. Finally, we discuss the related work and conclude this paper.

Issues with the Traditional Project Management Method

The waterfall approach is the traditional project management method. During Fall 2015 and Spring 2016, there were a total of 10 senior design projects in the same class sequence. Most of

these projects used the waterfall approach or the variation of it. Following the waterfall project management, students go through the stages of collecting requirements, designing the system, implementing, and testing. Students must completely finish one stage before moving to the next stage.

However, CS senior capstone project teams typically face several challenges, if applying the waterfall method to manage the project:

- Misalignment between the software project and customers' requirements, i.e., what students build is not what customers want.
- Late or delayed development of the project, i.e., students procrastinate to develop and test the product.
- Uneven distribution of workloads, i.e., some students spend much less time and effort than their teammates in the project.

As shown by the students' final project presentations, these issues did occur among the teams using the waterfall approach.

Scrum

Our capstone project applied the Scrum methodology. Different from the traditional waterfall management, Scrum does not focus on the details of the up-front plan and does not attempt to closely follow a predetermined plan. Instead, Scrum is an Agile approach for developing innovative products and can adapt quickly to the changes of users' requirements. The manifesto for Agile software development states: "We are uncovering better ways of developing software by doing it and helping others do it. Through this work we have come to value:

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more." [4]

Currently, Scrum is a widely adopted industry standard for building business software.

Applying Scrum to Our Senior Capstone Project

We applied the Scrum principles to our senior capstone project. Specifically, the following three principles were used:

(1) Working Closely with Customers

We worked closely with ACPL and set up monthly meetings to allow ACPL to review what we had built and provide us timely feedback. That is, we used one month as the cycle time of a Sprint [5]. We chose one month, instead of the more popular two weeks, due to the availability

of ACPL managers. During the monthly meetings, we showed the features that we had built in the past month and received feedback on whether we are on the right track. For example, during one meeting, we demonstrated the application form for individuals. After reviewing the form, ACPL managers realized that they also needed another form for organizations, which is not in the original requirements. If we were using the waterfall approach and ACPL pointed this out during the testing stage, it would be too late to add the implementation. Moreover, during the monthly meetings ACPL managers proposed the next feature(s) that they would like to see. In such a way, our students can implement most important items first. Since we met customers often and obtained timely feedback, we avoided the issue of the misalignment between the software project and customers' requirements. As a result, our Scrum approach can adapt and respond quickly to the changes of requirements.

(2) Incremental Development

We adopted the idea of incremental development and attempted to implement and test an increment each month (or Sprint). Specifically, we applied the Pivotal Tracker tool [6] to trace each increment, as shown in Figure 2.

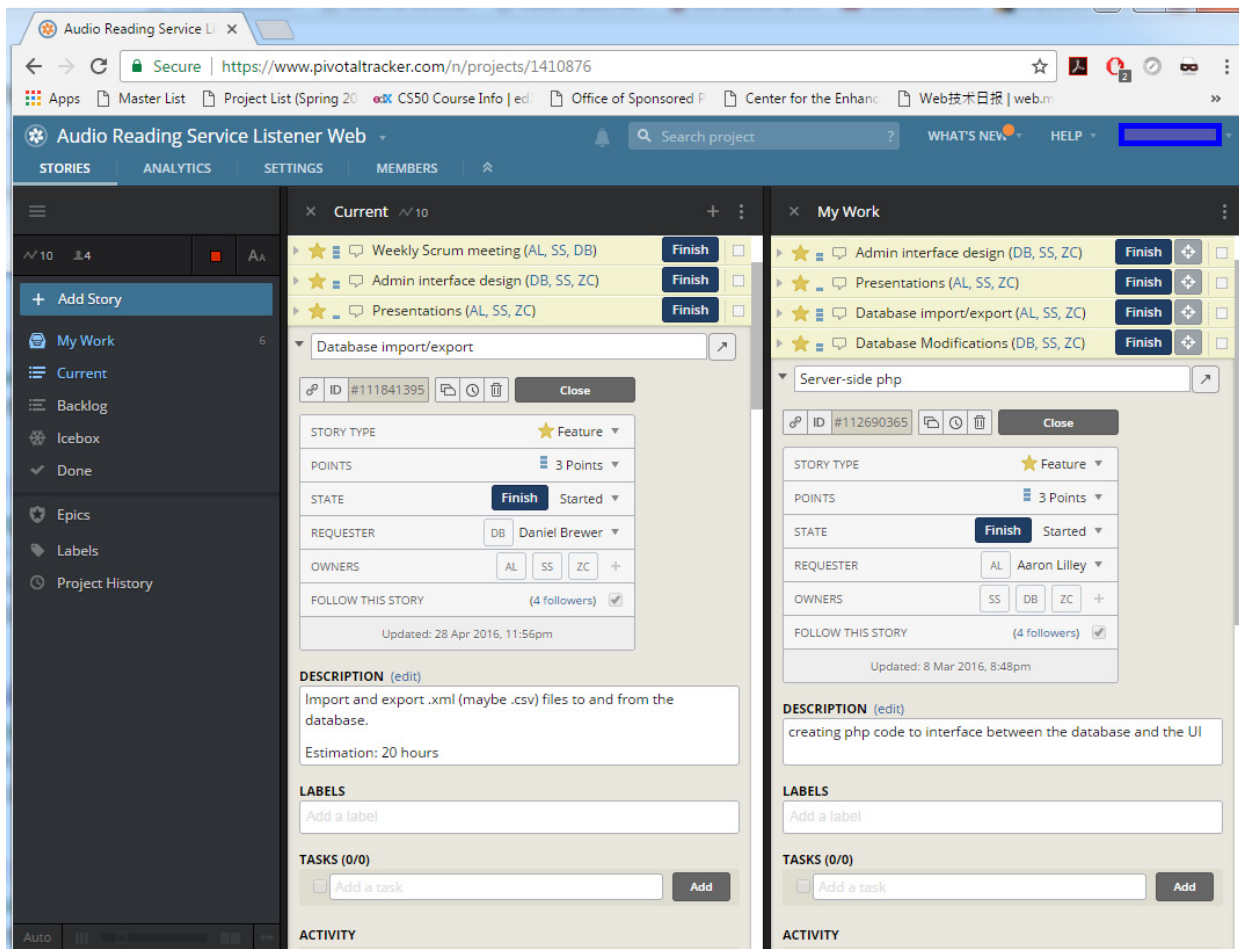


Figure 2. Screenshot of the Pivotal Tracker System

A user story was created and estimated for the size in the Pivotal Tracker system. If the story was picked by a student, he or she would start the story and log the time spent on this story. We found that such a tool is very convenient for viewing the progress of the project, assigning tasks to students, and collecting information about the project (such as how many hours that a student has spent on a task). Since every month students needed to implement an increment, we avoided the issue of the late or delayed development of the project. Moreover, because we implemented the most important items first, those items that were not so critical to the entire project were worked on towards the end of the project.

Moreover, we used Git [7] and GitHub [8] for software version control. As shown in Figure 3, we pushed the working code to GitHub, which can be found at [9]. From GitHub, the faculty advisor of the project can view who committed the code and how much code a student has committed. When the ACPL customers found a bug, we also used GitHub to understand how the bug was created and why. If needed, we can roll the project back to the previous working version from GitHub. In our project, Git and GitHub have been proven to be great tools for controlling software versions and organizing the code among multiple developers.

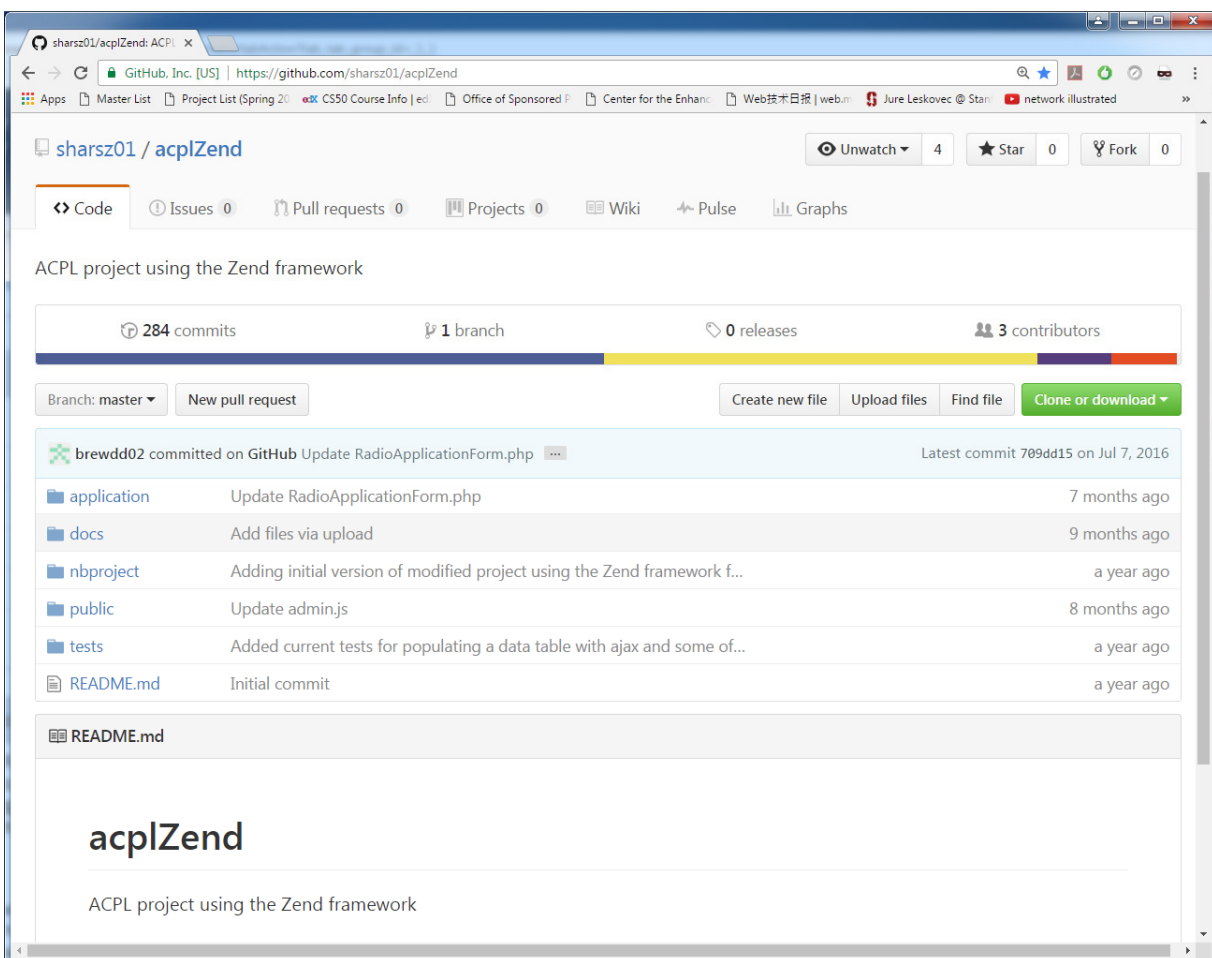


Figure 3. GitHub Repository [9]

(3) Weekly Scrum

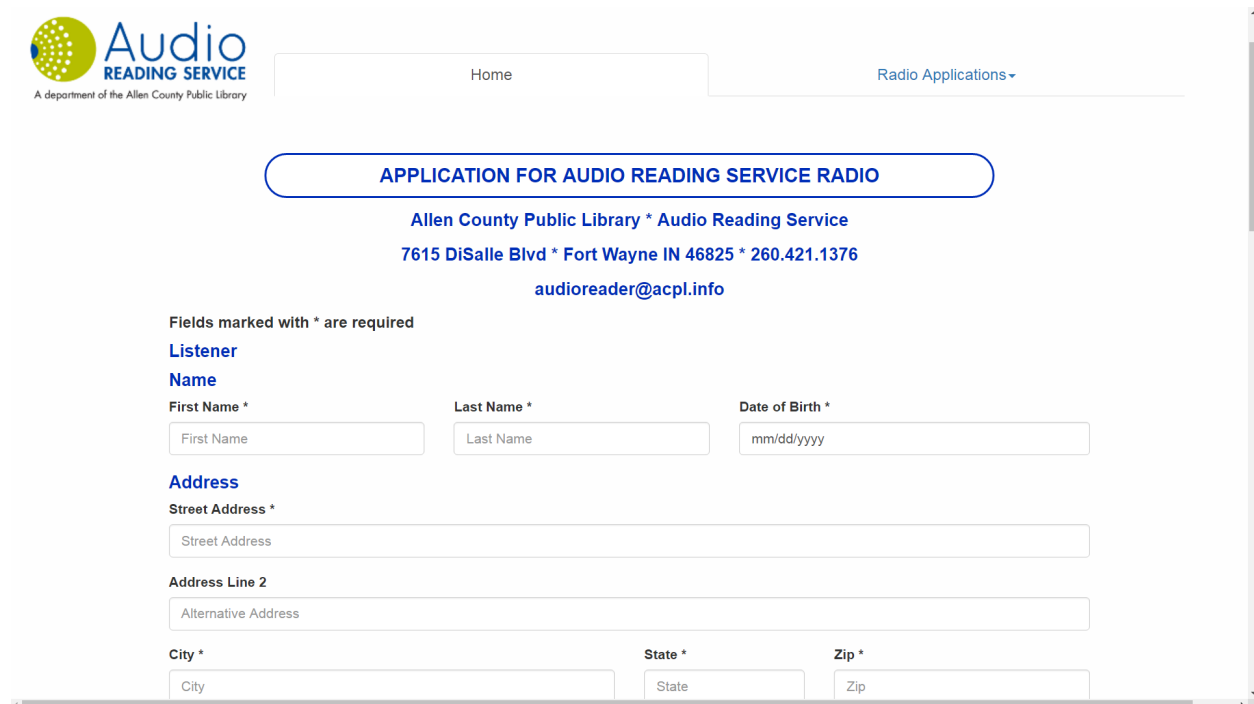
Under the spirit of daily Scrum [5], the faculty advisor met students weekly to discuss the progress of the project and assign new tasks. Specifically, during weekly Scrum, students shared their answers to three questions: “What did I accomplish since the last weekly Scrum? What do I plan to work on by the next weekly Scrum? What are the obstacles or impediments that are preventing me from making progress?” [5] After that, the advisor and students discussed how to overcome these obstacles. Since the advisor met students weekly, the advisor made sure that all students contributed to the project and avoided the issue of the uneven distribution of workloads among students.

Evaluation and Outcomes of Our Senior Capstone Project

The completed final software product was put into public use and received positive comments from satisfied customers. The students in the project also gave excellent feedback on the guidance of the faculty advisor and their design experience.

(1) Web Service Provided to ACPL

The Web application that we built has been put into public use. The radio application forms can be found at [10], as shown in Figure 4.



The screenshot shows the 'APPLICATION FOR AUDIO READING SERVICE RADIO' form. At the top left is the logo for 'Audio READING SERVICE', a department of the Allen County Public Library. The navigation bar includes 'Home' and 'Radio Applications'. The main heading is 'APPLICATION FOR AUDIO READING SERVICE RADIO'. Below this, contact information for Allen County Public Library * Audio Reading Service is provided: 7615 DiSalle Blvd * Fort Wayne IN 46825 * 260.421.1376 and audioreader@acpl.info. A note states 'Fields marked with * are required'. The form is divided into sections: 'Listener Name' with fields for 'First Name *', 'Last Name *', and 'Date of Birth *'; 'Address' with fields for 'Street Address *', 'Address Line 2', 'City *', 'State *', and 'Zip *'. Each field is a text input box.

Figure 4. Online Application Forms [10]

Moreover, Figure 5 demonstrates the structure of the admin interface used by ACPL managers to view, search, update, and delete the information of audio applicants, as well as to assign radios to them. Since this interface is used only by ACPL managers, it is not publically accessible, and ACPL managers need to login before using the functionalities.

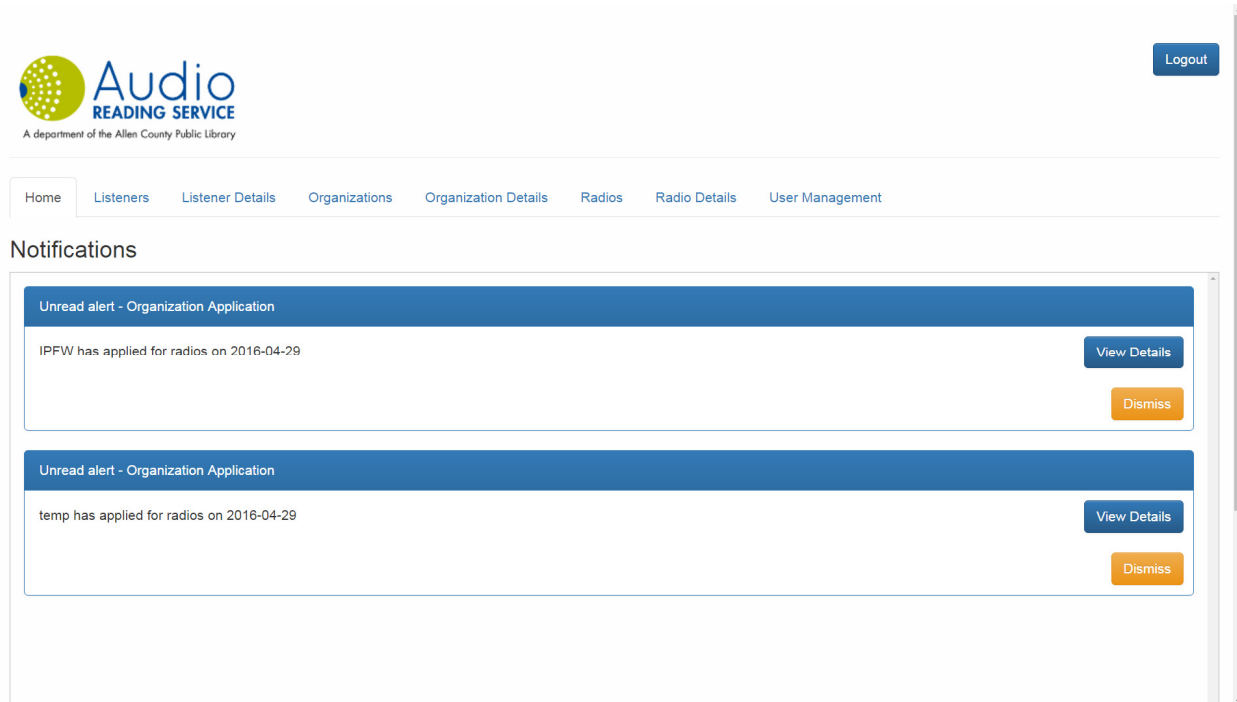


Figure 5. Structure of the Admin Interface

Upon the conclusion of project, our students documented in detail how the Website was built and how to use the functionalities. We handed over all source code and documents to the ACPL IT department, who is currently maintaining this Website.

(2) Response from ACPL Customers

ACPL managers are very satisfied about our end product. They highlighted our contributions in their newsletter: *“A new, efficient listener and radio inventory database with a streamlined online application. This was made possible because of a collaboration with IPFW’s Computer Science department; three students created this for us as their Senior Capstone project.”*

Moreover, ACPL managers commented in their emails to the faculty advisor of the project, *“We very much appreciated your help and the work that your students put into their project. It has helped our organization of listeners immensely.”*

Our senior design team was invited to ACPL Volunteer Recognition Celebration and was recognized during the meeting on June 9, 2016.

(3) Response from Students

Three students in the project responded very positively in the assessment. The following two tables show their evaluation to the faculty advisor and the class for CS 465 (Spring 2016) and CS 460 (Fall 2015). The department average evaluation score is also provided as a reference. The maximum score is 4.00.

CS 465 (Spring 2016)

#	Question	Individual Evaluation <u>Score</u>	Department Evaluation <u>Score</u>
7	Instructor's knowledge of subject	4.00	3.50
8	Instructor's ability to present	4.00	3.07
9	Instructor's enthusiasm	4.00	3.42
10	Instructor's concern individual	4.00	3.29
11	Instructor overall	4.00	3.21
12	The class content	3.67	3.20
15	Assignment related to course	4.00	3.35
16	Exams measured understanding	4.00	3.27
17	The course overall	4.00	3.22
	Average of statements 7-12 & 15-17	3.96	3.28

CS 460 (Fall 2015)

#	Question	Individual Evaluation <u>Score</u>	Department Evaluation <u>Score</u>
7	Instructor's knowledge of subject	4.00	3.53
8	Instructor's ability to present	3.67	3.12
9	Instructor's enthusiasm	4.00	3.47
10	Instructor's concern individual	4.00	3.33
11	Instructor overall	4.00	3.31
12	The class content	3.67	3.20
15	Assignment related to course	4.00	3.40
16	Exams measured understanding	4.00	3.25
17	The course overall	4.00	3.25
	Average of statements 7-12 & 15-17	3.93	3.32

Ideally, it would be better to compare the evaluation scores from senior capstone projects using the waterfall approach with scores from our project. Unfortunately, we do not have the evaluation data from other projects. However, the results in the above tables show that our evaluation scores are much higher than the department mean, which indicates that students in our project highly appreciated the Scrum approach.

Moreover, all students either agree or strongly agree that they have achieved course learning outcomes and ABET criteria through this senior design project, as shown in Figures 6 and 7.

Section: 23163 - Senior Capstone Project II CS 46500-07

	ABET Criteria	Strongly Agree (5)	Agree (4)	Neither Agree nor Disagree (3)	Disagree (2)	Strongly Disagree (1)	Average
I am able to apply software engineering principles and skills to a team-oriented software project.	abcdik	2	1	0	0	0	4.67
I am able to construct a software project schedule and track its progress.	aik	3	0	0	0	0	5.00
I am able to construct artifacts appropriate to demonstrate completion of each phase of the software process.	aik	1	2	0	0	0	4.33
I am able to conduct formal technical reviews.	fk	2	1	0	0	0	4.67
I am able to utilize a repository for project artifacts.	aik	3	0	0	0	0	5.00
Total		11	4	0	0	0	4.73

Figure 6. Course Learning Outcome Assessment of CS 465 (Spring 2016)

Section: 13260 - Senior Capstone Project I CS 46000-07

	ABET Criteria	Strongly Agree (5)	Agree (4)	Neither Agree nor Disagree (3)	Disagree (2)	Strongly Disagree (1)	Average
I am able to apply software engineering principles and skills to a team-oriented software project.	abcdik	2	1	0	0	0	4.67
I am able to construct a software project schedule and track its progress.	aik	2	1	0	0	0	4.67
I am able to utilize a repository for project artifacts.	aik	3	0	0	0	0	5.00
I am able to construct artifacts appropriate to demonstrate completion of each phase of the software process.	aik	3	0	0	0	0	5.00
I am able to conduct formal technical reviews.	fk	1	2	0	0	0	4.33
Total		11	4	0	0	0	4.73

Figure 7. Course Learning Outcome Assessment of CS 460 (Fall 2015)

Related Work

The Agile approach has been introduced to the software engineering course and the capstone course. For example, Reichlmayr reported on the experiences of conducting a team project in an introductory software engineering course using agile development techniques [11]. Fox and Patterson used Rails, a highly productive programming framework, to introduce Agile processes in their software engineering courses [12]. Moreover, Mahnic described the observations of the behavior of developers using Scrum for the first time in a capstone course in [13]. Different from the previous works, in this paper our senior capstone project worked with real customers and

produced a final product in public use. Furthermore, we emphasize the importance of applying the online tools, such as Pivotal Tracker and GitHub, in the Scrum process.

Conclusions

We found that the Scrum method contributed significantly to the success of our senior capstone project. By using the Scrum method, we overcame several challenges faced by the traditional waterfall approach. Moreover, ACPL is very satisfied about what we have built and delivered. The software product has been put into public use and is currently serving people with disabilities. Furthermore, the students in the project gained experience on building a real-world software product and working as an effective team. Through this experience, we believe that Scrum serves as a better project management method than waterfall for undergraduate senior capstone projects in computer science.

References

1. The Americans with Disabilities Act (ADA) compliance standards [Online]. Available: <http://www.ada.gov/> (January/2017 accessed).
2. ADA Best Practices Tool Kit for State and Local Governments Chapter 5: Website Accessibility Under Title II of the ADA [Online]. Available: <http://www.ada.gov/pcatoolkit/chap5toolkit.htm> (January/2017 accessed).
3. 23026-2015 – ISO/IEC/IEEE International Standard – Systems and software engineering – Engineering and management of websites for systems, software, and services information, DOI: 10.1109/IEEESTD.2015.7106438.
4. Manifesto for Agile Software Development [Online]. Available: <http://agilemanifesto.org/> (January/2017 accessed).
5. K. S. Rubin, *Essential Scrum: A Practical Guide to the Most Popular Agile Process*, Addison-Wesley 2012, ISBN 978-0-13-704329-3.
6. Pivotal Tracker [Online]. Available: <https://www.pivotaltracker.com> (January/2017 accessed).
7. Git [Online]. Available: <https://git-scm.com/> (January/2017 accessed).
8. GitHub [Online]. Available: <https://github.com/> (January/2017 accessed).
9. GitHub Repository for the Web Application for ACPL [Online]. Available: <https://github.com/sharsz01/acplZend> (January 2017/accessed).
10. Online Application Forms for Audio Reading Service at ACPL [Online]. Available: <https://arsdatabase.acpl.lib.in.us/RadiosApp/public/index.php/Index> (January/2017 accessed).
11. T. Reichlmayr, “The Agile Approach in an Undergraduate Software Engineering Course Project,” 33rd ASEE/IEEE Frontiers of Education Conference, November 5-8, 2003, Boulder, CO.
12. A. Fox and D. Patterson, “Viewpoint: Crossing the Software Engineering Chasm,” *Communications of the ACM*, Vol. 55 No. 5, pp 44-49, May 2012.
13. V. Mahnic, “A Capstone Course on Agile Software Development Using Scrum,” *IEEE Transactions on Education*, Vol. 55, No. 1, pp 99-106, February 2012.