Tyler Bennett

CS 250 X2116

Dr. Walker

19 June 2021

Sprint Review and Retrospective

ChadaTech has been around for quite a few years now and up until now we have specialized in custom software design and development using the waterfall method. Today we will be reviewing a recently completed project that my team did using a Scrum-agile approach. A Scrum-agile approach is different from a waterfall method in several major ways. Within the waterfall method, the client would tell us what they wanted to do and we would start working on it. We would all work on various facets of the project and in the end compile all of our code into one and tweak anything that didn’t work properly. In the past this method has seen some success and some failures. Recently with the waterfall method, we completed a multimillion dollar project only to have the client change their minds on several rather large functions. This led to us basically restarting the project from scratch. With the Agile approach, we wanted to analyze its ability to remain flexible while meeting deadlines.

My Scrum-agile team consisted of four different components. We have the Product Owner, this person is responsible for our Product Backlog and the coordination with customers and our shareholders. They help the team prioritize the needs of the client through user stories. User stories are a way to provide the Agile team with a clear and concise client need. They are generally written as, I the <user> want to be able to do <this> so that <this> will happen. An example, As the end user, I want to be able to click on google chrome and have it open the application. The Product Owner has a great relationship with the Scrum Master.

The Scrum Master is responsible for ensuring the teams needs are being met, all obstacles are taken care of and tracks the progress of the project. The scrum master is responsible for the planning of daily scrum meetings, sprint planning, backlog refinement, sprint reviews and sprint retrospectives. Although the Scrum Master is organizing and taking responsibility, they are still a part of the team. No one individual in a Scrum-agile team is over another. We are all peers. The scrum master plays a vital role in the teams ability to remain flexible while meeting deadlines. When the clients needs change, the product owner adjust the user stories and product backlog. The Scrum Master must then pass this information onto the rest of the team and coordinate with them to reevaluate their sprints and goals. The remaining two components of a Scrum team are the developers and the testers.

The developers on a scrum team contributes more to the project than just lines of code. The developer is responsible for making sure that the user stories are being accurately created. One of the most important aspects of being a developer is the ability to communicate between the rest of the team. When working on the project and a developer finds a user story to have several possible outcomes, it is their responsibility to communicate with the product owner and the client. They must be able to accurately describe what they are doing as well as ask concise questions to help them proceed. An example of this would be sending an email to the product owner and client stating something like, “… User story number 3 asks us to create a clickable link for the users profile that opens the users profile..”. While this user story tells the developer what is needed it doesn’t tell them how they want it completed. The developer could go on to ask them if they want the profile to open in a separate window, separate tab, or to replace the current webpage.

The final component of our team are the testers. The testers are responsible for ensuring that the users needs are being met by the implementation of the user stories. In order to do this the tester must have great attention to detail as well as the ability to ask questions. The tester works closely with the developers and the client to ensure they are all working towards the same goal. The tester evaluates the product that the developer has given them. They follow the test cases step by step. A test case is similar to a very detailed set of building instructions. They tell the user to click on x and verify that y happens. Then from there they should be able to click on a, b, and c, to get another response. The tester must verify that every aspect of the product is working and that the project doesn’t produce any errors. While doing this, if they come across something that they are unsure of they must have the ability to communicate this with the client. Similar to the developers email, the tester will help ensure they are getting the most accurate information for their testing.

Now that we have established the various components of an agile team, we can look at the different benefits. During our project we found great success in our daily scrums. The daily scrums are instrumental in monitoring workflow and the development team’s success. A 15-minute morning meeting lets them know what their teammates are working on that day, what they finished the day prior and if anything is stopping them from succeeding. The daily scrum holds all of the development team accountable and this meeting is directly facilitated by the Scrum Master. Another important aspect of the agile approach is sprint planning. During sprint planning the team gathers to talk about the product backlog. The product backlog contains all of the user stories for the project. They evaluate them and decide their priority as well as assign them some kind of value. A user story that the team feels is going to take a long time may be quantified as three user stories, whereas something simple may only be one user story. Once the team has established the user stories they are going to complete during their sprint, which could be 1-4 weeks depending on the team, they assign each member their user stories. The quantifying of user stories helps the team understand what they can realistically complete during a sprint. In the beginning this may not be accurate but as the team grows and begins to learn each others abilities, this will become more accurate. Having these daily scrums and bi-weekly sprint planning meetings allowed us the flexibility and adaptability to adjust to our clients needs when they quickly asked us to change some major aspects of the project. Instead of waiting until the end of the project to provide our client with their product, we provide it to them throughout the process. This allowed them to identify any concerns they had and it made it easy for us to reassign it to a user story to correct during our next sprint.

My team faced some adverse challenges during this trial of the agile method. One of our biggest challenges was COVID. We were unable to meet in person for any of the project. This required us to really focus on our communication skills as well as find alternative ways to implement the Scrum-agile approach. Microsofts Azure Boards allows the team to drag and drop sprint planning, create work items that are essentially user stories online (you and your team are able to describe the product owners needs as well as what you need to complete the story). Azure Boards has functions for Product Backlogs that show the item type, the status, who the item is assigned too. The list is customizable making it easy to rearrange as things change and make for effective sprint planning. The user dashboard provides clear data on what is happening. It is fully customizable to allow you creativity in adding widgets for tracking progress. Although this tool is best combined with in person daily scrums and sprint planning, when tested with remote work it still played a vital role in the teams effectiveness.

To keep the conclusion of this short and sweet. I would fully recommend the agile approach over the waterfall approach every single time. The agile approach allowed our team the ability to communicate clearly, to understand each others needs throughout the process, but most importantly it allowed us the ability to adapt to our clients needs while putting out a more efficient product in a shorter amount of time.