Spring 4-28-2017

The Agile Approach in Pharmaceutical Software Development

Priyal Bhargava
University of Connecticut - Storrs, priyal.bhargava@gmail.com

THE AGILE APPROACH IN PHARMACEUTICAL SOFTWARE DEVELOPMENT

Priyal Bhargava

Department of Biomedical Engineering

University of Connecticut, Storrs, CT 06269

28 April 2017

Acknowledgements and Thank You

Dr. Kevin Brown

Metrum Research Group

UConn Honors Program

TABLE OF CONTENTS

Abstract

The Agile development model is increasingly being adopted by the pharmaceutical industry for software development. Currently, there are several software projects under development by pharmaceutical companies that are using the Agile process. Other methodologies such as the Waterfall process, a traditional approach to project management, have evolved over time and been compared against the Agile method. Also there are conflicts of interest which should be taken into consideration when in a pharmaceutical setting such as abiding by FDA regulations. In addition, Agile's sub-divisional approaches add to the versatility of how this iterative approach is applied in the pharmaceutical industry. The client of our senior design project, Metrum Research Group, is new to this process. Our senior design project focuses on developing a web scraper for parsing and compiling drug development pipeline data from an array of pharmaceutical sources, all by using the Agile approach. Software development, using Agile, in the pharmaceutical industry for drug development is very unique and has many components that are transforming the biomedical industry as a whole.

Keywords – Agile, Waterfall, Pharmaceutical Industry, Software Development

Introduction

What is the Agile methodology? The Agile methodology is an interactive technique used for the developmental progress of a project [1]. Agile is a project management system where sprints or iterations are utilized to complete allocated tasks and assignments. Sprints or iterations are short spans of time, usually two weeks, where a team meets up and discusses the cycle of a project. In these meetings, tasks are assigned according to a proper timeline. With the Agile methodology, team work plays a significant role in the success of a project and there are frequent interactions between the client and the developers.

Agile has been gaining recognition in the pharmaceutical industry for a variety of reasons. Waterfall is a more traditional and linear process that has been often compared to Agile for projects. Nevertheless, there are several positive elements for Agile when comparing projects that utilize Waterfall. Utilizing Agile in the pharmaceutical sector not only adds to associated documentation for software development in a highly regulated industry, but also leads to successful projects in terms of time, budget, techniques, and scope flexibility.

This paper is going to go through the history behind Agile and how it evolved, other software development methodologies, and approaches under the umbrella of Agile. It will also analyze certain tradeoffs within Agile, how it is currently used in the real world, and how it is being applied to our senior design project. Agile is currently implemented in the pharmaceutical industry for software development in various ways and there are numerous advantages to using this iterative methodology.

Theory and History Behind Agile

The software development methodology faced issues at the start of the 1990s. The phrase, "the application development crisis", represents the harsh conditions at the time [2]. During that period of time, projects would be terminated due to the variability of business requirements and specifications which was difficult to keep up with. There was a broad gap between the business and software development approach in flexibility of a project. For instance if a business's needs changed rapidly, it would be difficult financially or timewise to adapt to those alterations or fluctuations in the project. In other words, changes made during a project's course led to complications further down the line. As a result, projects would be withdrawn or not completed in spite of a project's initial scope being reached.

From these challenging circumstances, leaders from different industries began to think about alternatives to solve the dilemmas. For example Jon Kern, from the aerospace industry, sought to create an improved system for software advancement [2]. He compared it to waterfall and further methods of development present during the era. It turns out that he was one of the seventeen pioneers of the Agile methodology which dates back to February 2001.

The story of how *Manifesto for Agile Software Development* emerged connects back to a ski resort [3]. There was a group of seventeen people in one of Utah's ski resorts, Lodge at Snowbird. Kern, Kent Beck, Ward Cunningham, Arie van Bennekum, and others were amongst the group. The group was called "The Agile Alliance" and some of the members were the founders of developmental techniques such as Extreme Programming [4].

Here, this group discussed how to expedite the software production process. They came up with some advantages to their methodology. One of benefits is that users would obtain a software's business uses at a quicker rate [2]. Another advantage is that the feedback received by

the development team is faster. In other words, the team can consistently gather significant information on the project's overall course throughout the developmental process. These two characteristics became fundamental for the Agile methodology. For instance not only can the developers gain feedback from the business, but also users would be able to receive features of the software in a quicker fashion. This leads to the twelve underlying principles of the Agile Manifesto, created by the founders:

1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
2. Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
4. Business people and developers must work together daily throughout the project.
5. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
6. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
7. Working software is the primary measure of progress.
8. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
9. Continuous attention to technical excellence and good design enhances agility.
10. Simplicity--the art of maximizing the amount of work not done--is essential.
11. The best architectures, requirements, and designs emerge from self-organizing teams.

12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly [5].

These principles are the pillars of the foundation of what Agile is and its expectations. It was developed as a standard for individuals to follow and serves as the foundation of the Agile Manifesto.

Software Development Methodologies Timeline

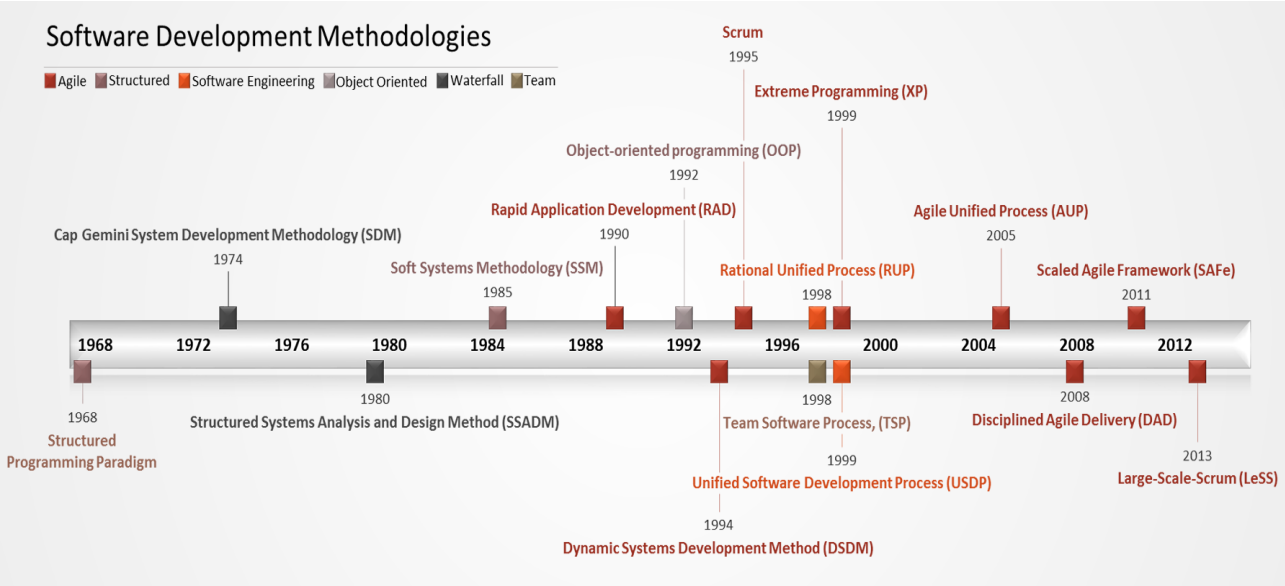

Figure 1. Software Development Methodologies [6]

This figure displays the overall timeline of different software approaches. It starts off from the late 1960s and goes until recent times. It is divided into various sections marked on the legend: Agile, Structured, Software Engineering, Object Oriented, Waterfall, and Team. From the timeline, it shows that Structured Programming Paradigm initiated the approaches. Waterfall emerged in mid-1970s. The focus of this paper is Agile which began early 1990s and continues on until present day. Some of the approaches that will be discussed in detail later consist of the Dynamic Systems Development Method, United Software Development Process, Scrum, and

other agile components. Each have different focus areas and are distinct from one another. Overall, this figure represents the history of software development from start to current day and when they emerged.

Methods of Development Under Agile

As explained, Agile is a distinctive methodology that is widely used in different industries for project management and development. Agile encompasses several components such as iterations, testing, and planning which is outlined in the diagram below:

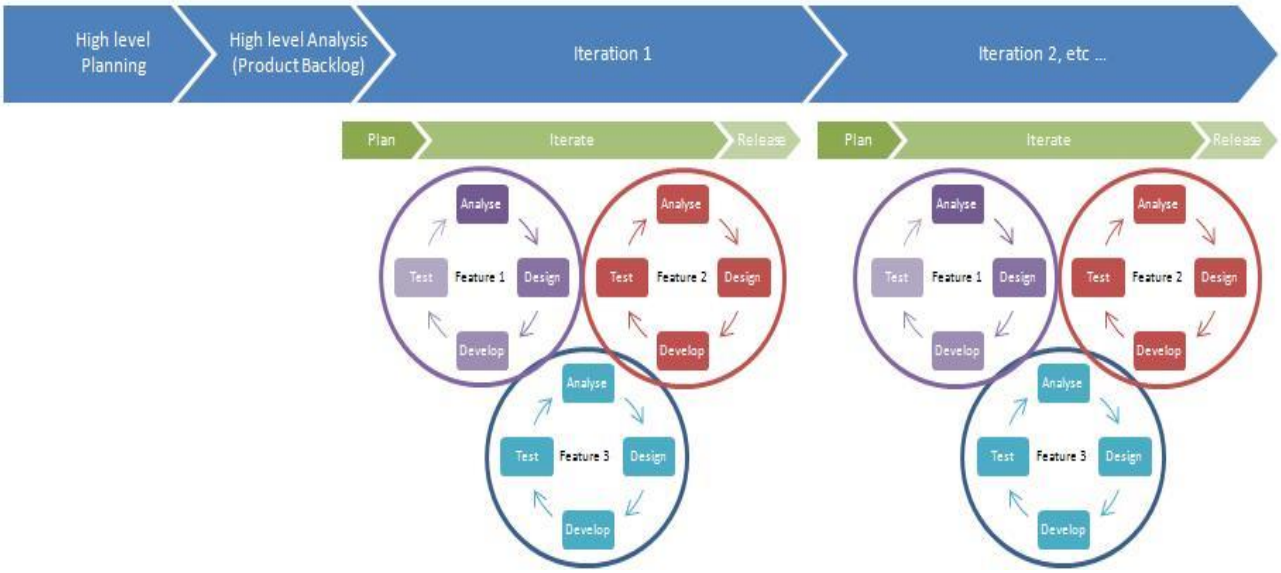

Figure 2. General Workflow of Agile [7]

Figure 2 shows a basic overview of how Agile flows. From the diagram, a project generally starts off with a significant amount of planning. After there a backlog is created to list out the tasks and assignments to be completed during a certain amount of iterations or sprints throughout the course. Each sprint is around two weeks where development and testing occurs. Team members all work on different tasks during an iteration. There also may be some

intersection between tasks from different groups. For example, two features in the same sprint could have similar components and therefore demonstrate some overlay.

Delving deeper in the Agile methodology, there are several components that fall under Agile. The different methods include Scrum, Dynamic Systems Development Model, Kanban, Feature-Driven Development, Lean, Extreme Programming, and Crystal [8]. All of these Agile approaches vary in different ways. Scrum and DSDM will be expanded upon and explained in greater detail further in the paper.

With Kanban, there is constant deliverables from developers. Collaboration within the team is highly valued with this method. Additionally, there is more flexibility in regards to sudden alterations and priorities can be adjusted easily during the project. There is a tool called a Kanban chart which lists out tasks according to status [9]. An example of a Kanban chart is shown below:

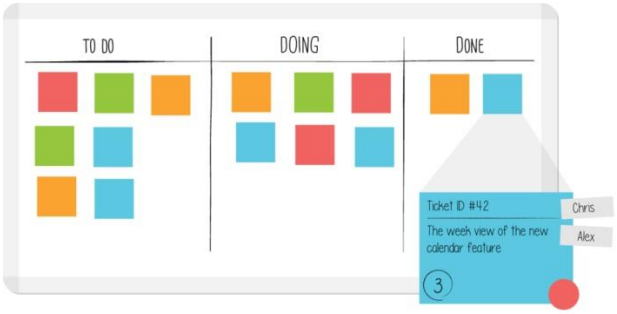

Figure 3. Kanban Chart [10]

This Kanban chart displays the status of different tasks. There are different assignments that need to be done, are in progress, and are completed. Furthermore, having this chart gives team the flexibility to change items around freely if change is needed. This chart gives a representation of the development of certain tasks for projects.

Feature-Driven Development (FDD) is utilizes features which are comparable to deliverables to the business [8]. The sprints are quicker and are based on the overall features.

There can be members that handle certain features. Deliverables are more obtainable in this method as well. FDD is usually more beneficial to teams with more members. In addition, FDD consists of smaller sprint cycles due to feature scheduling.

Lean is a unique process that has several characteristics. This method highly focuses on the consumer [8]. Some qualities of Lean include decreasing excess or surplus in regards to development and fast responses and communication from users. Lean also makes sure that members are efficient constantly throughout the project. In other words, individuals should utilize their means effectively. Team motivation is highly encouraged, too.

Extreme Programming (XP) focuses the software development component the most. More specifically, the overall standard of the software is highly valued through this process [8]. There is significant consumer or user communication, frequent testing, and constant iterations. Testing is a huge part in this process. For instance tests are conducted in regards to both sides, the consumer and development team. The scheduling and preparation is steady to ensure deliverables. Furthermore, there are “user stories” which represent the small tasks and these are ranked or prioritized throughout the course of the project. Overall, XP concentrates on the quality of the software for user satisfaction.

Lastly, the Crystal process is very flexible and highly adjustable [8]. Some qualities of this methodology include continuous deliverables and consumer interaction. Additionally, the framework is based on how the team communicates with each other. For example, the team considers ways in which the project can be better and adjusts with the changes.

Agile vs. Waterfall

The Waterfall methodology is a traditional approach to Agile. It is a more traditional practice to development. It emerged before Agile and has several distinct characteristics that make it unique from Agile.

Waterfall is a step by step project management process that gathers all client requirements and specifications upfront before development. In other words, questions and feedback to and from the client are very frequent in the beginning of the project. The project scope is solidified at that time, as well. Therefore, there is less flexibility or adaptability with this method. Waterfall is useful for big-scale projects where the main objective is certain so no changes occur with the business [11]. The framework is structured distinctively as shown with the diagram below:

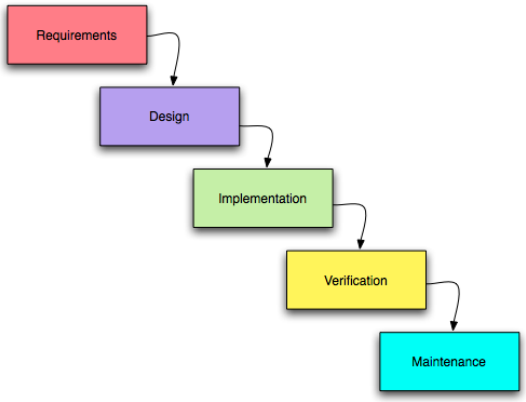

Figure 4. Waterfall General Flow [11]

This figure demonstrates the basic flow of the waterfall method. It begins by collecting requirements, designing prototypes, testing and applying them, and solidifying one design. After the design has been selected, there may be several bugs and defects found. The next step would be to fix them and finally bring the product out to production. Afterwards, the steps are the same for both Agile and Waterfall – to create a possible training guide and put it out into the market.

Overall from the diagram, it shows that the traditional Waterfall method has linearity with its steps.

There are notable negative aspects with Waterfall when compared to Agile. For example with Waterfall, time and cost of a project may be limited [12]. If development costs have to be cut or time shortens, the quality of a product would be lower. With Agile, the budget can be monitored throughout the iteration and can be tracked more efficiently. In addition, there is more risk with the waterfall method [12]. For instance, timing is a significant factor in the success of a project. Since the testing portion of development occurs near the finish, it leaves more room for error and risk. Testing occurs constantly through the development of a project with Agile, thus making it less risky. Also the Waterfall method is not as flexible to change as is Agile [12]. With Agile, changes that occur during the sprints are more adaptable than during the process of Waterfall. Once the requirements and specifications are recognized, it is more difficult to alter the deliverable at a later point in the methodology.

Documentation is also an important difference between Waterfall and Agile. With Waterfall, the main aims, requirements, and specifications are written out in the beginning. In other words, the deliverable's requirements are noted from the start. Agile is iterative, meaning that there are sprints of about two weeks during the course of the project. At each sprint, there is documentation stating what has been done and what needs to be completed in the next cycle [13]. A sample outline of a sprint document is displayed below:

Last Sprint Completed Tasks	To Do Tasks for This Sprint
Task 1	Task 3
Task 2	Task 4

Table 1. Sample Sprint Document

This template shows the basic overview of a sprint document. Members list out their completed task and tasks to do in the current iteration. A backlog is also another way to check if the tasks in the sprint are going according to plan. A backlog is a list of tasks for the entire project. Sprints are based on the backlog assignments, which are usually mapped out in the beginning of the project [13].

The pharmaceutical industry has several ways to apply the Waterfall method. A hypothetical example would be: a company is developing a medical software application for treating the flu. In this case, the company would first consult with the client several times to ensure they understand the scope of the project and their needs. The development team would ask a series of questions to their client at the meeting and verify the specifications they would like for the project. After, several prototypes would be designed. Team members would contribute their ideas and design a rough sketch of the application and its features. After deliberation, a final design would be chosen to implement. That design would be tested and revised until the final product is up to the client's expectations. Once the team reaches this point of development, it is more difficult to change specifications if the client's needs alter. If they do change, the development team would have to start the process over which would result in lost time and money. After solidifying the product, it would move out to production and marketing. The product would have to be approved by not only the client, but also higher organizations such as the FDA. A hypothetical case like this can also be viewed from an Agile perspective as well.

Taking another look at this case through the Agile lens, the development team and client would have several meetings for planning out the overall scope of the project. After, a backlog of task would be created by the development team. This backlog would contain tasks ranging from coding to assessing user feedback. The backlog assignments would then be further broken down

into features or stories which would be allocated to certain sprints. The sprints would last two weeks each and all of the tasks in the backlog would be spread out between them. Starting with the sprints, the development team would meet up frequently to discuss progress and have daily stand up meetings. At the end of each sprint, all members of the project would come together to go over the status. If the client's need changes, the team can quickly adapt to this. For example, if the client would like an additional feature that tests for symptoms for the flu, the development team would be able to revise the next sprint or iteration to adapt to this change. After all of the iterations and feedback from the client and users, the product would be pushed to production and approval. Throughout the project, the sprint documentations and code repositories would act as documentation. The code repositories could be a website where the developers interacted with each other and uploaded code. The two approaches, Waterfall and Agile, to this hypothetical case each have different aspects to software development. Comparing and contrasting the two methodologies gives people an understanding of what each entails when deciding which approach to utilize for a certain project.

Non-Agile Software Development Methodology

Joint Application Development

Joint Application Development (JAD) is another methodology utilized in the pharmaceutical industry. JAD is a technique in which looks at or is more oriented towards the business aspect more [14]. In other words, the client is more engaged in the developmental process. With JAD, workshops or sessions with members of a project being present, consisting of the clients, coders, and consumers.

This methodology also is cost-effective in regards to collecting requirements [14]. For instance, with JAD time to gather specifications and conditions for a project is cut down, therefore, diminishing the overall cost. This is possible with the aid of the individuals involved in this project and their contribution or interaction with the team altogether. Due to the frequent connection with the business, there is a higher contentment on the client side. In addition, the total time to develop the project is quicker. With the waterfall method, the development team has to scrutinize objectives and what the client actually wants from the project whereas with JAD, there are more regular collaborations with the business [14]. Overall, this approach intends to decrease time and cost efficiently by various techniques.

A case study exemplifies this method's characteristics in a pharmaceutical setting. A Laboratory Information Management System (LIMS) was to be created by the company called Thermo LabSystems [15]. Thermo LabSystems connected with three other businesses to further their advancements: Astra Pharmaceuticals, Coca Cola Company, and Horseracing Forensic Laboratory. These corporations helped access various LIMSs and provided useful information to Thermo LabSystems.

There were several concerns that were taken into account when choosing a LIMS. For instance, consumer feedback highly mattered as well as the interaction with the producer of the system [15]. Another encounter was considering which labs favored which kind of system. The cost and time to set up are other factors that were taken into consideration. Pharmaceutical companies can have specific rules and regulations that need to be adhered to when using LIMS, as well.

Astra Hassle is a part of the Astra Group became involved with Thermo LabSystems due to the fact their system was experiencing several faults and thus was very expensive for them

[15]. A little about the company and its aims, Astra Hassle's bioanatical chemistry lab had several types of samples which were to be handled. For example according to the article, "more than 80,000 clinical, pre-clinical, and toxicological samples that it examined each year" [15]. With this number, there are various other systems that catered to these tests. From this, a LIMS was desired and the process to adapt to the information was laborious and pricey. Additionally, Astra Hassle's main focus is producing novel drugs into the market. Therefore by taking part in generating a LIMS cut into their drug formation time. The company had numerous requirements they would like the LIMS to encompass. For instance, the LIMS must abide by Good Laboratory Practice Regulations, have high adaptability and flexibility, and maintainability for a high number of years [15]. Furthermore, companies should be able to communicate with Astra Hassle. Looking at these specifications, Astra Hassle narrowed their search for a system that consists of these characteristics and figured there was not any. Thus, Astra Hassle combined with Thermo LabSystems to reduce the cost and time.

Thermo LabSystems was in the course of making a LIMS called Nautilus [15]. The company realized that several corporations wanted to take part in the process due to time constraints on their end. JAD then taken into consideration and was introduced to the other companies. With this users could provide responses to the product in a more time efficient manner.

At first, Astra Hassle decided to conduct meetings within the company employees in regards to Nautilus [15]. These individuals ranged from marketing to coders. They gathered for around four months and discussed how to incorporate other companies' requirements with theirs. After selecting the three corporations that agreed with their plan, Astra Hassle requested an Ambassador user and some Advisor users. More specifically, the role of the Ambassador user

was to dedicate 60% of their day to Astra Hassle's new LIMS whereas the Advisor users dedicated up to 15% to JAD methodology [15]. For example, the business aspect of the project was handled more by the Ambassadors. Using JAD, a website was implemented to record feedback from consumers. In the meantime, the developers would respond to these defects to ensure satisfaction.

As a result joining hands with Thermo LabSystems, Astra Hassle not only was able to obtain a LIMS with their requirements, but also able to continue to work on their subject area of drug production [15]. This aided their pharmaceutical lab by a great margin. In addition with the JAD clients, the final module catered to the necessities of different kinds of labs, such as biotechnology.

With Nautilus to make sure the system restores itself, a comprehensive document with all alterations throughout the project was logged [15]. After passing the guidelines of the JAD clients, training guides were created. Conferences and discussions continued to occur every two weeks between the developers and JAD clients. Here they updated the systems. With JAD, clients were informed about changes by email. The clients involved in this project gained a sense of how to better their methodologies in the lab. Nautilus's performance with the partners was adapted to each requirement. Here JAD expected for communication and honesty with all member to have a successful project. In other words, companies should be willing to disclose relevant information with collaborators. Overall, the JAD process worked successfully in terms of offering a deliverable to the clients [15]. The LIMS adapted to the specifications of each business and provided consumer satisfaction with the JAD framework based on partnership.

Conflicts in Agile

There are several conflicts which may occur when using the Agile approach in the pharmaceutical industry. One difficultly that pharmaceutical software projects encounter is abiding by the guidelines of certain organizations [16]. For example, organizations like the FDA, EMA, and ISPE can control the approval of a product going into market. In addition, companies also have to adhere to specific laws such as HIPPA if their project correlates with it. Furthermore, these guidelines may be altering constantly with time which may negatively impact the team [16]. For instance, a rule regarding the approval of a team's product may have changed from the time they started development to finish. This can cause a significant amount of time and cost lost. Additionally, the development team must have sufficient amount of documentation to support their product. Projects with lower than expectation documentation may not be able to be approved by higher agencies.

Implementing the project within the time and budget given by the client can also present a challenge in the pharmaceutical software development area. For example, if the development team realizes later on in the project that they will need more resources to accommodate the needs of the client, it can be problematic. In addition, if unexpected hindrances occur during the course of development such as a technical delay, it can push the entire project back, leading to dissatisfaction on the user side. Furthermore, not adhering to the sprint task and completing certain tasks on time can lead to more delays in the project.

Another conflict of interest is remaining Agile throughout the course of development [16]. Some teams may be used to the traditional method of development and keeping members on track can be a difficult process. Ensuring the team is following Agile protocol is an important

aspect, especially in the pharmaceutical industry where documentation and regulations are strict and meticulous.

Real World Applications

Project 1: Scrum

Scrum is a methodology under Agile that is used for project management [14]. Scrum is an iterative process that uses sprints or iterations throughout the project course. Teams meet on a sprint by sprint basis to discuss tasks and assignments for a time period [17]. From the start, these tasks are listed out in the iterations and take into account priority. There are also standup meetings on an everyday basis in which the development team describes the status of their tasks and future steps. Communication is significant in scrum and the team meets frequently during the project development [14].

The scrum master is an individual who helps the team overcome obstacles which may hinder the overarching progress of the project [14]. In other words, he or she organizes the team's iteration goals and helps with team productivity. The project course is a team effort where there is no head controlling the assignments. Another role in scrum is the product owner (PO) who is more on the business side and knows what the final product should entail [17].

Sprints are usually characterized with a two week time period. The team meets in the beginning of an iteration and decides on the amount of tasks each person can complete. This list, called a backlog, is then recorded [17]. Features, which are above tasks, are developed and tested at each iteration. The regular standup meetings occur for about fifteen minutes during sprints. There is another sprint meeting at the close of the two week time period where the PO can input their thoughts about the iteration and the team can review the overall status and next steps in the

project [17]. The goals of scrum include enhancing quality, time management, and change management.

An example of effectively utilizing scrum in the pharmaceutical industry can be demonstrated. A study was conducted to analyze the effects of using scrum to create a system to manage inventory for a enterprise resource planning division in a pharmaceutical company [18]. It was also compared to the more traditional process of waterfall. The study goes over the scrum process with the company and identifies shortcomings with the traditional techniques.

The pharmaceutical company studied was Farma Y which is concentrated in Brazil and specializes in generic drug advertising and manufacture [18]. According to the article, the generic drugs are divided by therapy: “antibiotics, antidiabetics, antidepressants, antihypertensives, antihistamines, antilipemics, antifungal, anti-infectives, analgesics, anti-inflammatories, muscle relaxants, and glucocorticoids” [18]. Additionally, Brazil’s strict drug control regulations related to the amount of drug and discontinuation date geared this project’s main goal.

The primary aim of the project was to create an inventory element which would regulate drug expiration dates [18]. This would therefore minimize the shortfall of drugs that have a narrow span of expiry and decrease waste. This project was named “XYZ Project”. The team decided to use the scrum method to implement it. The plan was to split the components into sprints that lasted for 20 days per iteration. Each sprint had different objectives. For example, sprint number one covered extracting numerical information about the drugs with a narrow survival span and sprint number two applied filters to the information [18]. The second iteration was further divided due to time expenditure on various aspects. Additionally, sprint number three focused more on the cost of different components.

As discussed previously, there are a few roles that are filled with scrum. In this case, there is a scrum master, PO, and the development team. The scrum master's task was to make sure the team adhered to the principles and regulations of the methodology, as well as furthering efficiency with the project. Furthermore, the scrum master enhanced productivity within the team and removed any obstacles that could sway the course of development. In addition, the scrum master is the middle man, connecting the client and the developers. In the XYZ project, the scrum master's role was to lead the IT development group and motivate members to reach their goals in sprints recorded in the backlog [18]. The PO in the XYZ project handled the costs and business side. The PO would connect with the scrum master in regards to the backlog of tasks to ensure the project was according to schedule. Lastly, the team had several kinds of members apart of different divisions. For example, the team consisted of people who specialized in finance, coders, and consultants. In the XYZ project, the team members all worked in large groups at a table with no boundaries [18]. It was an open space for all to converse. This allowed for the most communication and efficiency. For instance, coders who know a one language can promptly aid others due to being in the same vicinity.

The research then dove into the backlog of assignments. This was regulated by the PO in the XYZ Project [18]. A sample of the backlog is shown below:

Product Backlog – XYZ Project					
ID	**User Case**	**Relevance**	**Estim.**	**How to Demonstrate**	**Remarks**
1	Know the expiration date of the medicine batches in stock	200	5	Create specific screen for in stock materials, by batch expiration date.	Use, if possible, view by the inquire screen of current inventory, through customizing.
2	Correct inconsitencies in the expiration dates of the medicine batches in stock	180	3	Create manual update mechanism, with function segregation, to update inconsistente dates. Additionally, the possibility to black manual batch entry, at the end of the production line.	Review access profiles and add new functionality, only for Quality Manager.
3	Do not allow expiration date changes for medicine transferred among warehouses	160	2	Keep, during material transfer, expiration date of the original batch.	Review stock transfer transactions, and check associated profiles.
4	Validate the expiration date of medicine batches in stock	140	5	Complementary to ID #2. There must be a process of "auditing" of in stock materials, with a record of changes in the validity of the materials dates.	New program to be created, with restricted and controlled access execution.

Figure 5. Sample Backlog [18]

The backlog portrayed the overarching with the assignment represented as "User Case", the priority, the amount of time it would take to complete it, how the deliverable is exhibited, and notes about the task. For example, the first backlog item would take around 5 days to finish for the XYZ Project. Since each iteration was 20 days, the tasks were divided among that time period properly. During this project, a Backlog Grooming meeting was held prior to the iteration which could revise the tasks accordingly.

In the XYZ Project, the scrum master gave great importance to Daily Meetings [18]. Daily Meetings are usually done in the beginning or close of the day. Furthermore, they help with project improvement and efficiency. For this project specifically, these standup meetings occurred in the morning for about fifteen minutes. During the standups, the prior day's work, obstacles, and present day's work were presented to the scrum master. The scrum master would review the issues if there were any. Some examples of shortcomings include the team not being well informed about the scrum method, need for interaction with IT units and PO, and the team inclining towards waterfall methodology [18].

For this project, there is a discussion with the PO, user, scrum master, and team in which they demonstrate their work so far at the finish of an iteration [18]. A problem that arose in the beginning iterations was that the user wanted to alter the outcomes. As a result, the iterations were divided and more time was added to the project. More specifically, an extra 40 days was provided [18].

A Sprint Retrospective meeting was held within the team members only [18]. Here, they went through what they can do better in future iterations and progressive steps they have taken which made components of the XYZ Project successful. At the finish of an iteration, Value Engineering was implemented for the XYZ Project. This approach analyzed the aim of a deliverable and to ensure the overall project objective was being met. This would, in turn, provide users with approval and maintain cost efficiency [18].

Release Sprints took place at the completion of sprint one, three, and five in the XYZ Project [18]. For sprint one, the release analyzed the drug batches with a narrow expiration span by checking the dates on the drugs. Incorrect dates were fixed and drugs that were transferred or in process were not altered. For the second and third iteration, the release looked at the applied filters such as by drug class and date [18]. The last release examined the total cost of the batches. At the end of the XYZ Project, documentation was made and it moves into production.

In conclusion of the XYZ Project, the final count was 100 days and $145,000 in all to complete the project [18]. After the first couple of iterations, the overall process became more productive and useful. There were several obstacles to overcome in the beginning, but towards the end, the approach became straightforward. The XYZ Project would have come out to be $291,000 and an additional three month to finish if a more traditional method was implemented [18]. This study demonstrates the scrum agile approach a pharmaceutical company takes to

complete a project. Here, a inventory system was generated to help control drug expiration time [18]. The users were satisfied with the outcome and praised the timely manner. The time and cost were clearly cut using the scrum approach compared to the classical one. The methods, such as everyday standups, Retrospective meetings, and releases, the pharmaceutical company used all made the project successful. The notion of sprints or iterations aided with productivity and time management, ensuring deliverables at a steady rate. The XYZ Project exhibited effective communication between the scrum master, PO, and team as well. Overall, the scope of the project was highly valued and the software system was delivered in an efficient manner.

Project 2: Dynamic Systems Development Model

The Dynamic Systems Development Model (DSDM) is a unique methodology under the umbrella of Agile. It was created in 1994 in London and stemmed from another approach under Agile called Rapid Application Development (RAD), which immerged in 1990 [14]. The viewpoint of DSDM is that the aims of the project should be definite and planned. In addition, clients should receive advantages or products in a timely and cost-effective manner throughout the course of the project development [19]. In other words, this methodology entails responsive measures, providing benefits at a fast pace. Another characteristic of the DSDM process is customer participation which betters the quality of the products. Furthermore, it is iterative and modifications during development can be adjusted to [14]. Overall, there is active participation with the investors and distribution of products within budget and time throughout.

Cardiff University and Napp Pharmaceuticals demonstrates an example of using DSDM in the pharmaceutical industry [21]. A pain management website was to be created for individuals in the university, medical specialists, and people in the public attracted to the subject.

This website would offer advice in regard to handling and treating pain. The team decided on using DSDM and adhering to eight values that represent this process.

First, DSDM uses the MoSCoW prioritization method which helps prioritize tasks and specifications in a project [20]. The letters represent Must Have, Should Have, Could Have, Won't have this time and symbolizes a negative consequence of not satisfying a project aim. This team followed this focus by prioritizing tasks and providing significant deliverables to the client.

Another ideology of DSDM the team followed is satisfying requirements in a timely manner [21]. They conducted workshops once a month where they discussed completed tasks and upcoming assignments to ensure timely deliverables. Collaboration is an additional DSDM value the team focused on [21]. In order to comply with this objective, the team set up face to face and phone meetings often to drive the team and unite members. The notion of quality is highly revered for the Cardiff team. The group went through the requirement samples frequently and re-evaluated them utilizing the MoSCoW methodology to make sure the quality was up to par.

Consulting with the each other with face-to-face meetings was a significant part to ensure the team knew the requirements for the website beforehand [21]. For instance, the team held discussions where they talked about the aims and technology they will apply to the project prior to development. With DSDM, comprehending the specifications of the project is crucial to safeguard any risks further down the line. Another focus of DSDM the team implemented was iterative progress [21]. In this case, the team created time boxes where certain requirements were assigned to each. In this way, the group could analyze each step thoroughly and know what to expect for future iterations.

The seventh value the team applied was communication [21]. Communication is key in DSDM. From frequent workshops to meetings, the team was able to receive the status of the project and updates efficiently. Throughout the course of the project both the technical and business sides were on the same page about the project advancement. For example, some approaches they took to communicate properly with each other were having notes for meetings, an event schedule, face-to-face discussions, and further steps actions.

Lastly, a sense of control of the project was an aspect the team followed [21]. For example, there was a site for the project where the status of each project goal was portrayed. In this way, all members could see the progress of the project and have confidence that the projects was going according to plan.

Overall, the team implements the DSDM methodology efficiently to reach their project goals. They understood and implemented the core values this approach encompasses and delivered efficiently. This method was used in the pharmaceutical industry to create a website for pain management. The website was created successfully, undergoing these eight valuable focuses of DSDM [21].

Senior Design Project Application

Our team's senior design project was to create a software which would scrape pharmaceutical websites for drug development information, specifically in the pipeline section. With this our client, Metrum research Group, would like us to extract the drug name, drug phase, mechanism of action, and the condition it is treating. Metrum is a biomedical company that specializes in modeling and simulation in the drug development sector [22]. The goal or

objective of the project was to use an Agile approach to parse websites with different formats such as HTML and PDF.

For this project in terms of Agile, our team started by planning and discussing the scope of the project with the client and scrum master. Learning about the company values and researching more about project as a whole, a backlog for the entire course was created. A sample of this backlog is shown below:

	Backlog	156.67d	Sun 1/15/17	Mon 4/10/17		
23	Research R Terminology & Syntax	7 days	Sun 1/15/17	Sun 1/22/17	Backlog	Not Started
24	Research SQL Terminology & Syntax	7 days	Sun 1/15/17	Sun 1/22/17	Backlog	Not Started
25	Create list of base PDF and JPEG websites	1 day	Mon 1/23/17	Tue 1/24/17	Backlog	Not Started
26	Create algorithms for pdf extraction using Python PDFQuery	9 days	Wed 1/25/17	Fri 2/3/17	Backlog	Not Started
27	Create algorithms for jpeg extraction using Python PyTesseract	9 days	Wed 1/25/17	Fri 2/3/17	Backlog	Not Started
28	Testing Phase for HTML (PDF)	2 days	Fri 2/3/17	Sun 2/5/17	Backlog	Not Started
29	Testing Phase for HTML (JPEG)	2 days	Fri 2/3/17	Sun 2/5/17	Backlog	Not Started
30	Design Interface for HTML subunits	4 days	Mon 2/6/17	Fri 2/10/17	Backlog	Not Started
31	Test Interphase algorithm and fix bugs	2 days	Fri 2/10/17	Sun 2/12/17	Backlog	Not Started
32	Create base SQL list	1 day	Mon 2/13/17	Tue 2/14/17	Backlog	Not Started

Figure 6. Project Backlog

This backlog began from the start of the project to the end and listed out all of the tasks with durations.

After creating this document and discussing the logistics with the client, the iterations began. Our team split up the tasks into five sprints. During the sprints, tasks were divided out among the team members. The team would meet frequently to discuss the status of the project. These meetings would resemble standups where each member talks about their progress, what they are working on currently, and if any issues are present. This would help our team with future iterations if changes were to occur. For the testing, GitHub was used as a code repository and Slack was used as a communication platform. These tools provided more documentation for the client.

In addition for each sprint, proper pre-sprint and post-sprint documentation was recorded. A sample of this document is displayed below:

Drug Pipeline Data Aggregation
Sprint 4: HTML/PDF Error Handling & R Introduction

March 26, 2017

Tasks completed in previous sprint:
Research R Terminology & Syntax
R Preliminary Setup
List of Mechanisms
Final Testing Phase for HTML
Testing Phase for PDF
Google Search Algorithm
Collaborate with Metrum to discuss desired output format in R

Tasks to be completed in this sprint:
Build of wrappers
Error handling/Fix Bugs
Manifest Algorithm
Test Interphase algorithm and fix bugs
Design determining algorithm
Testing phase for determining algorithm
Create Python – R Handshake
Test handshake capabilities
Create final output pathway
Develop R Visualizations
Test R code

Figure 7. Sample Sprint Document

This gave a basic overview of what tasks were to be completed in a certain iteration and what has been finished. Furthermore, the status of the project was reviewed with the scrum master on a weekly basis. During these meetings, issues with development and timing were discussed. In terms of development in the sprints, defects and errors were improved upon during the time period. For example, our team had a tough time parsing websites with the pipeline section in PDF. Therefore, HTML websites were emphasized and worked on more. Our team discussed these issues with the client to receive their feedback.

After completing the five sprints, our software was wrapped up. From this whole Agile process, time was allocated properly within the team. The sprints provided structure in terms of scheduling and helped keep the team on track. Furthermore, the sprints helped the team members visualize what tasks are remaining and how much time to spend on them. Since we did not use any money for this project, cost was not a hindrance. Time, however, was significant because the project had to be completed by the end of the year. In addition, the documentation gave not only

the team, but also the client an important record of how this project ran. The sprint documents are helpful for regulation purposes and the GitHub repository for documentation is a beneficial method to track changes.

From the project, there are several concepts the team can improve on in terms of Agile project management. For instance, feedback from the users or client is an area that can always be augmented. Having more iteration and testing with the user may be helpful for future software development projects. Another improvement is managing the time for meetings more efficiently. Timing was a difficultly when scheduling meetings so improvement in this area could help communication for future projects. More documentation is also an area that can be worked on. Providing more documentation not only provides the client with more information for system approval, but also increases the validity of the project as a whole.

Impact and Future Recommendations

The Agile methodology has a significant impact on software development in the pharmaceutical industry. There are several components to Agile which make it beneficial for software development projects. Under Agile, there are various sub methods such as Scrum which can provide projects with a iterative process which in turn helps with requirement flexibility and adaptability. With Agile, sprints or iterations make having changes in the project more adjustable. In addition, the Agile method provides significant documentation standards for pharmaceutical companies. For example, FDA approval required an extensive amount of documentation for products. With Agile, the sprint documents and developer platforms provide records of project process and testing. Agile also takes into account time and budget requirements. This is very important in software development project in the pharmaceutical

industry because certain projects can only have a limited budget and time span before the product is out into production. The pharmaceutical industry is rapidly evolving and keeping up with technology and changes in this field is crucial. Agile provides a way to manage projects efficiently and provide the most user satisfaction.

For future recommendations, combining aspects from different project management methods may be beneficial. There are several positive characteristics from various approaches. For example, the sprint aspect from Agile can be joined with the DSDM's notion of the MoSCoW prioritization process. Taking strong components from different platforms can be beneficial for certain projects and can cater specifically to the needs of the client and team. Another recommendation would be to incorporate more documentation opportunities within the Agile process. Several industries, not only the pharmaceutical sector, need intense documentation for approval. Providing basis for that would help projects be more successful in the long run as well.

Conclusion

In conclusion, Agile is a distinctive approach that has widely influenced the pharmaceutical industry in a multitude of ways. Researching the different aspects of this approach has given insight on how this model can be furthered in the future and provides a core framework for real world applications. Our senior design project is a prime example of Agile in the pharmaceutical industry and displays the techniques used under the process. Methodologies have continuously progressed over time under Agile and have positively impacted the success of projects in the pharmaceutical sector.

References

[1] “What is Agile? What is Scrum?”, *cPrime,* 2017. [Online]. Available: https://www.cprime.com/resources/what-is-agile-what-is-scrum/

[2] “To agility and beyond: The history-and legacy-of agile development”, *TechBeacon*, 2017. [Online]. Available: https://techbeacon.com/agility-beyond-history%E2%80%94-legacy%E2%80%94-agile-development

[3] “The Secret History of Agile Innovation”, *Harvard Business Review,* 2017. [Online]. Available: https://hbr.org/2016/04/the-secret-history-of-agile-innovation

[4] “History: The Agile Manifesto”, *agilemanifesto.org,* 2017. [Online]. Available: http://agilemanifesto.org/history.html

[5] “Principles behind the Agile Manifesto”, *agilemanifesto.org*, 2017. [Online]. Available: http://agilemanifesto.org/principles.html

[6] “Software Development Methodologies Timeline”, *Office Timeline,* 2017. [Online]. Available: https://www.officetimeline.com/blog/software-development-methodologies-timeline

[7] “Agile Development Cycle”, *101 Ways,* 2017. [Online]. Available: http://www.allaboutagile.com/agile-development-cycle/

[8] “What Is Agile Methodology”, *VersionOne*, 2017. [Online]. Available: https://www.versionone.com/agile-101/agile-methodologies/

[9] “What is Kanban? Kanban Software Tools”, *VersionOne,* 2017. [Online]. Available: https://www.versionone.com/what-is-kanban/

[10] “What is a Kanban Board”, *leankit*, 2017. [Online]. Available: https://leankit.com/learn/kanban/kanban-board/

[11] “The Waterfall Development Methodology”, *The Smart Method*, 2017. [Online]. Available: http://learnaccessvba.com/application_development/waterfall_method.htm

[12] “Agile vs Waterfall”, *Agile In a Nutshell*, 2017. [Online]. Available: http://www.agilenutshell.com/agile_vs_waterfall

[13] “Traditional Vs Agile documentation”, *Quotium,* 2017. [Online]. Available: http://www.quotium.com/performance/traditional-vs-agile-documentation/

[14] “Software Development Methodologies”, *IT Knowledge Portal*, 2017. [Online]. Available: http://www.itinfo.am/eng/software-development-methodologies/#chapter7

[15] “LabSystems Refines LIMS Product Through Unique Joint-Development Program”,

Bioprocess Online, 2017. [Online].
Available: https://www.bioprocessonline.com/doc/labsystems-refines-lims-product-through-uniqu-0001

[16] "How the Pharmaceutical Industry and Agile Software Development Methods Conflict: A Systematic Literature Review", *IEEE Xplore*, 2017. [Online]. Available: http://ieeexplore.ieee.org/document/6976661/?reload=true

[17] "Scrum", *Mountain Goat Software*, 2017. [Online]. Available: http://www.mountaingoatsoftware.com/agile/scrum

[18] "Agile project management with Scrum: A case study of a Brazilian pharmaceutical company IT project", *ResearchGate*, 2017. [Online]. Available: https://www.researchgate.net/publication/312536118_Agile_project_management_with_Scrum_A_case_study_of_a_Brazilian_pharmaceutical_company_IT_project

[19] "What is DSDM", *Agile Business Consortium*, 2017. [Online]. Available: https://www.agilebusiness.org/what-is-dsdm

[20] "DSDM Atern Handbook (2008)", *Agile Business Consortium*, 2017. [Online]. Available: https://www.agilebusiness.org/content/moscow-prioritisation-0

[21] "Cardiff University & Napp Pharmaceuticals - A DSDM/Agile team", *Agile Business Consortium*, 2017. [Online]. Available:https://www.agilebusiness.org/pl/resources/case-studies/cardiff-university-napp-pharmaceuticals-a-dsdmagile-team

[22] Metrum Research Group, 2017. [Online]. Available: http://metrumrg.com/