Medical Informatics Project Plan

Tyler J. Latshaw

Southern New Hampshire University

TABLE OF CONTENTS

Medical Informatics Project Plan

An industry leader in health services and reporting, Medical Informatics recently initiated a multi-month-long product development project. The project, which kicked off in early May of 2017, is focused on expanding its clinical data warehouse to accept data from third-party feeds and onsite data to improve group analytics and reporting. The project has a firm deadline of December 21, 2017, which is currently in jeopardy due to several factors, including the project manager leaving the project partway through, the team's lack of understanding of the programming language, changing requirements, and changing priorities from other stakeholders the team is dependent upon. This conflict resolution plan will review the current team dynamics and make suggestions for strengthening team communication and project leadership.

Conflict Resolution Plan

Whether it is directly malicious or more passive, conflict is an inevitable component and byproduct of any project team. Conflict comes in many forms and can range from small disagreements on handling something to much larger arguments between individuals. Combatting conflict from the beginning is crucial for the project's success as a careful balance needs to remain for the team to work cohesively with each other. To do so, teams should adopt a conflict resolution plan. This conflict resolution plan will review the current team dynamics and make suggestions for strengthening team communication and project leadership.

Team Dynamics

Heading up the entire project for the first two of the three modules is the project manager, John Current. John is a seasoned professional, having been with the company for 23 years, and is well-versed in both the project's technical side and the business needs. In addition to John, the core technical team is comprised of a business analyst, two ETL developers, two report

developers, an architect, and a QA. The team is also joined by a subject matter expert, a sponsor, a business owner, and an IS manager. As needed, the team will also reach out for support from other departments such as legal and user testing and the software vendor for technical support. For this project's purposes, all team members report directly to the project manager overseeing all the efforts.

For perspective, it can be helpful to consider the team in terms of where it lies in the five-stage development model. This model describes a team's development and consists of the five stages of forming, storming, norming, performing, and adjourning (Larson & Gray, 2018). The group started in the forming stage, where they were getting to know each other, and progressed into the storming stage, where they began to encounter conflicts and become somewhat less effective. While the team has mostly entered the normative phase where productivity increases, they have not reached the performing stage yet where effectiveness is maximized.

Compared to other projects of a similar nature, this team has a distinct advantage. All the developers are experienced in the company, and with all of the systems they will be using. Right from the beginning, the team could dive right in and begin making changes instead of spending time becoming acclimated with the systems. The project will need to be created using MS Visual Studio, the first major conflict for the team, considering they are only familiar with Java-based programming. To combat this, the developers all attended training courses for the new language and environment.

Another primary conflict source includes the project manager leaving the project after the team completed the second module. While the team is strong enough to withstand losing the project manager from a self-management point of view, the issue that will occur is that the team will now need to absorb the work that John was completing or take the time to onboard a new

project manager to the project. The project manager was also there to help mitigate any conflict that arose from time to time. Since the team did not clearly assign the tasks at the beginning of the project, the team has seen multiple instances where the architect may have one opinion on the database structure. Still, the database administrator may have a differing opinion. Without the project manager, it is harder for the team to combat issues like this.

Additionally, it was mentioned that there are changing user requirements for the project. While the project's general scope was made clear initially, none of the specifics were available as they were not yet identified from the business. This has led to recurring issues such as scope creep as the users keep tacking on additional requirements or changing existing ones that affect the team. The developers have been more than accommodating, but this will delay some part of the timeline.

Lastly, it was made clear that other departments were not involved with the project initially; instead, they were included later as needed. The problem with this is that the departments could not plan for this specific project adequately and have other deadlines that need to be met. For example, the project requires policy document wording for release, but they might not have it ready in time. If they had known about the project earlier, they could have allocated resources better.

Conflict Resolution Leadership

Identifying conflict is only part of the battle to improve a team as the team needs to resolve disputes to succeed collectively. While conflict resolution will mainly come from the leadership in a top-down approach, it is ultimately up to every team member to employ resolution strategies. As the project leader, the project manager will need to adapt their leadership style for the best results for the team. Common leadership styles include guiding,

coaching, structuring, and directing, all of which are situational (Larson & Gray, 2018). The project manager does not need to embody one for the entire project duration and can use a combination of the styles. In the current stage of the project, the most effective would likely be directing. While coaching could be useful as the team is expected to become overwhelmed, the group mainly needs someone to step in and give orders until the project is back on track.

Given that the team is also running into overlapping roles and responsibilities, it would be prudent for the project manager to help everyone identify their roles and skills needed for it. While existing workflow will clearly define most of the positions already, the project manager should specifically look at any "gray areas" of overlap and explain who handles what. At the least, they should have a protocol in place to determine which role will be responsible for making any decisions that need to be made when there is overlap. In terms of skills, all soft skills will likely be the same across all of the roles. In addition to common skills like communication and professionalism, specific focus areas for soft skills include Patrick Lencioni's hungry, humble, and smart virtues. Lencioni's theory states that any team member embodying all of these traits will be an ideal team player (Lencioni, 2016). Hard skills will be unique to the position. The project manager should have identified these attributes before the project beginning but had not in this case. To be most effective, any role definitions and skills required should be made available for everyone to review and learn from when working with that teammate.

Lastly, as conflict arises, each team member needs to identify how to resolve it. For this, the team will adopt one of the primary conflict resolution strategies of assertiveness, accommodation, avoidance, compromise, or collaboration. As with most conflict resolution models, all of these are entirely situational, and there is no right or wrong for which strategy to employ. As a rule of thumb, the team will be encouraged to use the technique of collaboration as

it is the maximum blend of assertiveness and cooperativeness while still completing the task at hand (Larson & Gray, 2018).

Motivation and Confidence

Going forward, emphasis should be placed upon motivating and instilling confidence in the employees in order to maximize the team's production. To do this, the project manager should follow five best practices, all of which are proven to be useful from various data points. The five best practices include recognizing a job well done, allowing employees autonomy, including the employees when setting goals, facilitating meaningful relationships, and rethinking how reviews are conducted (Stobierski, 2019). At the root of all of these best practices is the idea of empowering employees to strive to do better and perform higher. When team members hear that they did a good job or a respectfully told they did not meet an expectation, they are more receptive. Additionally, when they are included directly in the goal-setting process that will directly affect them or are given the ability to manage themselves more, they will likely feel more valued and want to help the team succeed as much as they want to succeed (Stobierski, 2019). It is important to note that this process of motivating the group is a constant process and will need to be evaluated for effectiveness and modified as required by the project manager.

Project Evaluation

As previously mentioned, this project is at a pivotal moment as it can take a downward turn or chart a new course to success. Losing a project manager in addition to other hindrances has stunted the team's potential, but they remain optimistic and willing to put in the work needed. The full details of careful evaluation of the project's current state are as follows.

Status Evaluation

As noted in the case study, the project formally kicked off on May 5, 2017, with additional monitoring and planning from the information systems manager and business owner dating back to February 14, 2017. With this, the project has a strict deadline and launch date of December 21, 2017. The project cannot change this date as the marketing team has completed promotional materials referencing this specific date. There is also additional pressure from other business units that need this project to go live as soon as possible.

According to the text, the project started well. The developers could begin coding full force given their prior experience and pre-existing familiarity with the systems and work on similar projects. From the time between the project kickoff to August 5, 2017, the team saw only a few minor issues due to the lack of user requirements in the requirements document. They resolved all the problems quickly between the developers and the users. As a result of this, it was reported that the project was not moving as fast as initially anticipated.

In the time leading up to October 5, 2017, the project's progress became much more delayed. As a result, the project sponsor became very concerned about the deadline, and the team needed to begin working overtime to catch up. Additionally, other departments raised concerns about meeting the deadline, given that they were not made aware of the project earlier on. For example, the legal team is needed for writing up documentation but was not made aware of this in advance. They would have been able to plan for this task but are now scrambling to get it completed in time. Other impediments to the project include a host of larger bugs that the developers saw in the code. All of these bugs need to be fixed before release, along with finishing the code in general. In addition to this, the project manager left the project about two

months before release and left the team drowning in complications, which include overlapping roles and no leader to make decisions for the team.

Beyond this, the project had seen several managerial issues before the project manager's departure. According to a progress report from the project manager before he left, the project is expected to be significantly over budget. Additionally, based on the project resource sheet, a report developer and a project manager are both overallocated to the project as resources. From a project sponsor standpoint, the project manager had been unable to schedule a one-on-one meeting with the sponsor, and she was also not at the team meeting in May. This could highlight a potential breakdown in communication.

Project Analysis

According to the case study, the original project documentation did not include specific requirements for user needs and expectations; instead, it contained only general goals. This contributed to significant scope creep that, in turn, contributed to delaying the project. Additionally, the performance of the system was initially prolonged. The team determined that the developers' code was very inefficient since they were new to the environment and language. Going back and fixing the performance is necessary but was not factored into the original scope, along with several bugs already mentioned.

Baseline

A baseline consists of three parts in project management - the schedule, cost, and scope (Larson & Gray, 2018). Project baselines are a crucial reference point that allows project managers and other stakeholders to determine a projected track. If too much variance occurs from the baseline, the project may be off track. Managers can use baselines to determine

potential lack of resources, delays, issues in reporting, or project management issues (Wilson, 2020).

Scope creep can and will affect the overall baseline of the project. One example is the gun registration system in Canada. Due to changing requirements and lack of information before the project started, the system's cost went from $100 million to over $1 billion (Widman, 2008). Management can assume that this project's baseline will be affected negatively unless the project leadership makes significant changes to stop the negative trend.

Project Costs

According to a project report from the project manager, the project baseline was budgeted at $441,650. The actual cost to date was $230,892, leaving only $210,758 for the project's remainder. The new estimated cost of the project is $529,980 for a calculated overage of 20%. This is a significant variance from the original project estimation and will likely cause investigation and review from company leadership. Additionally, it is reported that a total cost of ownership fee schedule has not yet been created to date. This document is essential in extrapolating the system's price over time to determine the actual cost.

Forecasting

For the sake of clarity, forecasts about this project will be broken into different scenarios as different actions and assumptions will likely lead to different outcomes. When forecasting the future of this project, the author will make a few key assumptions. Assumptions that guide the predictions include that resources are stretched to capacity as is and the company can allocate no more than one full resource to the project, the time to replace the project manager is significant, and even if they are replaced, it will take a while to onboard them, and the company needs to minimize excess cost at the project's expense. Given the current project state, the primary

assumption of adding a project manager to the team or not will be the primary differentiation between the scenarios. Additional assumptions and predictions will be based on the author's prior project management experience as a business analyst.

Schedule Variance

In a likely scenario, the project manager will be replaced for the project within a few weeks to a month after the current project manager, John Current, leaves. Assuming they are onboarded quickly, they will likely need to make a few tough decisions that will strain the team but will create a new standard. The team will narrowly meet the required deadline, but not without gaining an extra resource, nailing down specific requirements with the users, and continuing to work long hours. The new dynamic will test the team but will strengthen the team for future projects.

In an alternate scenario, the project manager is not replaced promptly or is not replaced at all. This will continue to cause the team to spiral, and they will not meet the deadline, or they will but with missing essential requirements. This will likely cause severe issues for the developers and hurt their communication in the future.

Cost Variance

Meeting the project's baseline cost given the current 20% variance is extremely unlikely, considering the project tends to be about 20% more costly than anticipated. In the scenario where a new project manager is assigned, they likely will need to add an additional resource or account for overtime. In both cases, the cost variance will likely increase even more than 20%. In the scenario where no project manager is assigned, the team will only meet the baseline cost if the team makes severe sacrifices to the project requirements, decreasing the project's scope. Simply

put, the team will not meet the deadline if they are not given a project manager and still need to meet the original baseline.

Estimates at Completion

Predicting the actual overrun for the project will be a challenge given the number of variables that may affect the outcome. If a new project manager is added to the team, they may break the trendline closer to the baseline, or it may continue to increase further away, adding more cost to the project. If no project manager is added, the project's overrun could metastasize exponentially. Averaging the factors out, it is likely that the project overrun at completion will be in the range of 20 to 30%. Based on experience, the project manager will need to sacrifice cost to maintain quality while meeting requirements and the deadline.

Impact of the Past

Based on the presented information, the lack of project management was the single most considerable impact that contributed to the project's current state and will result in failure if not corrected immediately. Other contributing factors include lack of requirements and proper documentation at the beginning of the project, inadequate notification of the project requirements to other business units involved, and improper delineation of roles and responsibilities. These issues have been documented as examples of setbacks or additional costs to the project that they could have avoided. While each could be viewed as isolated examples, each falls under the purview of the project manager.

Corrective Actions

First and foremost, the prior project manager, John Current, will need to be immediately replaced, or the project will likely not be completed on time. The new project manager's supervisor will need to have a meeting with the entire project team. The supervisor will use this

meeting to notify the group of structural changes and new expectations. It will need to be made clear what each role is responsible for. With this, a clearly defined conflict resolution protocol needs to be implemented for the team to quickly work through any conflict that arises from time to time. Also, a shared resource space will be made available to the entire team. This area will house the finalized requirements and any shared information such as the new roles and responsibilities and the conflict resolution protocol.

The project manager will add all target dates and milestones to a single location. They will be readily available to all team members, including the sponsor and business owner, for complete transparency. Assuming the current date of October 5, 2017, a new project manager will need to be assigned no later than the following Friday, October 13. The meeting introducing the new project manager and relaying new expectations will need to be held no later than the following Monday, October 16.

Immediately following the team meeting, the project manager will meet with the users and finalize the requirements. They will clarify to the users that the requirements cannot change for the minimum viable product unless in extenuating circumstances where a piece of logic was not accounted for and is necessary for proper use. The project manager can add any additional requirements in additional phases. By the end of the first week with the new project manager, no later than October 20, they will need to make available the documents mentioned above and create a shared space for all of them. It is expected that the new project manager will be fully onboarded and acclimated to the project by October 27, about two weeks after they are added to the project.

By November 1, the new project manager should have dictated the new standards to the team and explained the new expectations. Once shared, the team will begin working overtime

immediately at the rate of no more than 8 hours extra a week to get back on schedule. At the end of November, the team will reevaluate the current standing and make more severe course corrections as needed. The project will be delivered in its entirety on December 21, 2017, as initially expected.

To ensure the long-term success of the project, a few necessary proactive measures need to be taken. The new project manager will need to meet with the sponsor monthly after release to ensure that the new system meets expectations or gather any new requirements for changes. In addition to this, the team will need to continue to maintain shared resources for the system's life properly. As maintenance occurs, future developers will have accurate resources available for development. Lastly, the team will collectively partake in a blameless “lessons learned” activity to document pitfalls of the project so that the next project they work on will be more successful.

Project Charter

Central to the entire project, the project charter is a formal document created at the beginning of the project to list out the objectives, goals, benefits, and overview of the project, among other aspects. The project manager creating the charter will also use the document to highlight some of the project's strategies and plan for promoting proper communication. Further, the charter will be a living document because it will stick with the team through its duration and be used as a guidebook and reference point.

Project Objectives

This project is focused on developing a new product that will support the business's goals and its consumers. The project will expand the current clinical data warehouse databases to consume data streams from external entities such as onsite clinic data and wellness vendors. The system will incorporate data downstream with employer group analytics and other reporting

processes. Once implemented, the new upgrades to the system will produce activity reports on the company's products.

From a business perspective, Medical Informatics is a market leader in the health services sector, a very competitive environment. The company’s goal is to provide users with beneficial information, such as what this project will produce while keeping its competitive edge in the industry. This project needs to be seen through fruition quickly as competitors can beat the company to the market, opening the door for a significant loss in opportunity.

The project has a hard deadline for launch on December 21, 2017. By nature of existing marketing materials and company goals, the team cannot alter this launch date, and the project will need to be ready for release on time. According to the project documentation, the project kicked off on May 5, 2017, and was off to a slow but steady start. The team was met with setbacks but powered through. However, the longer the project progressed, the further behind the group became. This became a cause for concern from the sponsor. Two months before the release, the project manager stepped down, leaving the team without a leader.

At this point, corrective action will need to be taken immediately to salvage the project and meet the required deadline while still meeting all project objectives. To accomplish this, additional project objectives will be added, including adding a developer resource to the team, immediately adding a project manager to the team or utilizing an interim project manager from another team, and creating a communication plan that the team can use to work through any conflict they are met with efficiently.

Project Strategy

At present, there are several existing problems the team is battling, including lack of a project manager, changing user requirements, and outdated information, as well as an internal

conflict between members of the team. By far, the largest issue seen is the lack of a project manager. The team started the project with a manager named John Current, but he left after the second of three modules due to personal reasons. Since then, the team has been without a leader, which has caused other downstream issues.

The team has frequently seen several outdated documents regarding the user requirements and the fact that the users have changed their minds on specific requirements numerous times. While the team has been accommodating, this has caused significant delays and additional strain on the team. Further, there have been a few instances of internal miscommunication and conflict where individual team members may have differing opinions on development solutions for the database. Because there is no protocol for reporting issues like this, additional stress is being added to the team and its product.

Strengths and Weaknesses

In terms of strengths, the team generally works very well despite all being from different departments. Each of the members is more seasoned and has prior experience working on similar development projects. While he was a team member, John, the project manager, came with more than 20 years of experience working for the company and vast knowledge of the systems being developed. However, the team did have some significant weaknesses. Most notably, the team did not have any prior knowledge of any MS-oriented languages before. The entire project was to be completed in MS Visual Studio, which required a two-week delay while the team went through training. Additionally, the lack of knowledge and understanding of the software resulted in several support tickets with Microsoft, adding more delays.

Opportunities and Threats

This project in itself is a significant opportunity for Medical Informatics. While they are a dominating force in the industry, any competitor that makes a considerable enhancement like this will inevitably make a large gain. It is documented that this project is being pushed quickly through development to gain a more considerable edge. From a threat standpoint, the project team's biggest threat is needing to rely on external groups or entities. As previously mentioned, some tasks, including legal documentation, will need to be completed by other departments out of their control. Additionally, the team is heavily relying on support from the software vendor. Depending on the issue, the project may be blocked by needing help from these other teams.

Solutions for Success

Given that the project needs to be launched in two months, making immediate changes is essential if the team meets the deadline. While multiple solutions exist, the proposed best solution would include hiring a project manager, adding another developer to the team, creating a communications plan, and adjusting the project timeline and breakdown.

Adding a project manager and developer to the team is crucial for success. The team is in desperate need of a leader to help mitigate conflict and make decisions on behalf of the group. Without adding a new project manager, other team members will need to absorb the responsibilities, adding additional strain. Additionally, another developer is required, given the current development rate and how far the team is behind. However, this may not be feasible, given where the team is with the current budget being overallocated.

Creating a communications plan will mitigate conflict and help ensure that all stakeholders are on the same page and all of the developers have access to the latest documentation. In terms of giving the team extra time for the unforeseen, the project timeline

needs to be adjusted. According to the project plan, there are a few instances where there are days between tasks where the team could complete additional work. For example, if a part of the project is in user acceptance testing, the developer could be working on something else until it passes or fails. Additionally, while it is not ideal, the team could handle quality assurance and acceptance testing concurrently or even while the developer is still coding. Instead of testing all at once at the end, they could try smaller pieces at a time when they become ready.

Project Payback

According to Larson and Gray, a payback period is "the time it takes to pay back the project investment" (Larson and Gray). Until a project is fully paid back, they will be operating at a net loss from the project's expenditures. Once payback is achieved, they will begin to make money from the project. Based on the information documented in the executive project status summary, the team's budget includes $277,988 in approved capital expenditures and $163,662 in operating expenses for a combined total of $441,650. It is impossible to calculate a proper payback period given the lack of information regarding how much they anticipate charging customers for the product and how many customers will sign up. Still, it can be estimated based on rough figures. For example, if they plan to charge $10,000, they will need 45 contracts, where $2,500 would result in 177 contracts. If the company can expect to onboard 50 contracts a month, payback could be just under a month or as much as three months.

Selection Model

There are many selection models to reference when determining a project's value and whether to pursue it. Two common examples include the payback period mentioned above and the net present value (NPV), which finds a minimum return rate that is considered acceptable (Larson and Gray). For this project, either model would be sufficient, but the payback period

model would make more sense. The net present value model, while useful, is a firmer delineation between "go or no go." In contrast, the payback period provides more space for extrapolating scenarios and factoring in unforeseen circumstances like the ones seen on the project team. This would be especially useful if the company has not settled on a price yet for customers as managers would easily calculate the time until payback using different estimates for the number of contracts and various prices they are considering.

Project Communications Planning

Determining who the stakeholders are for the project is the first step in effectively communicating project requirements and information. For this project, the core group of stakeholders includes the project team comprising a business analyst, two ETL developers, two report developers, a software architect, a QA engineer, a DBA, a subject matter expert, the information systems manager, the business owner, and the sponsor. The new project manager and developer can be added to this list if and when they are added to the team. Additional stakeholders with interest in the project include other departments that will be assisting with various parts of the project, the user group that will own the tool, as well as the consumers who will be benefiting from the system and buying into it. Each of these groups will contribute to the project in a meaningful way except for the consumers, as they will only be involved after the launch.

Channels of Communication

To further elicit proper communication across the teams, the project manager will use multiple communication channels, including in-person meetings, video conference calls, chat systems, and phone calls. Each channel has its merit and may be better in one situation compared to another. For example, a chat message is much quicker than an in-person meeting, but some

communication may be better transmitted in person. Additionally, the project manager will be responsible for scheduling various recurring meetings for the stakeholders to make sure they are accountable for their work. An example of the regular team meetings is as follows.

Stakeholder	Method of Communication	Scheduled Meetings	Frequency
Business Analyst	Video Chat	Backlog Review	Weekly
	Video Chat	UAT Touchpoint	Every Other Week
ETL Developers	In-Person	Standup	Daily
	Chat	Status Report	Weekly
	In-Person	Retrospective	Every Other Week
Report Developers	In-Person	Standup	Daily
	Chat	Status Report	Weekly
	In-Person	Retrospective	Every Other Week
Software Architect	Video Chat	Backlog Review	Weekly
	Video Chat	Sprint Planning	Every Other Week
QA Engineer	In-Person	Standup	Daily
	Chat	Status Report	Weekly
Database Administrator	In-Person	Standup	Daily
Subject Matter Expert	Video Chat	Backlog Review	Weekly
	Video Chat	Sprint Planning	Every Other Week
Information Systems Manager	Phone Call	Project Update	Monthly
	Video Chat	Sprint Planning	Every Other Week
Business Owner	Phone Call	Project Update	Monthly
Project Sponsor	In-Person	UAT Touchpoint	Every Other Week
	Phone Call	Project Update	Monthly

Figure 1 This chart shows the primary meetings that the project stakeholders will be involved in and expected to attend. Listed with this are the meeting frequency and channel of communication.

Distribution and Collection

Since meetings only account for part of the communication in a team, it is vital to identify how any additional information will be collected and distributed. Information regarding the project's status and requirements was produced and distributed in three primary ways: progress reports, meeting minutes, and miscellaneous documents such as user requirements. According to the project documentation, progress reports were created by the project manager on a two-week cycle, and meeting minutes were sometimes made for team meetings but were not created for all meetings.

Moving forward, the project manager will split progress updates into two parts, a large update that they will distribute to all stakeholders every Monday and a smaller internal update distributed every Thursday to just the primary project team. Having reports more frequently will alleviate some concerns over communication and bridge the gap between the current cycle where a lot could have happened in the two weeks' time. Meeting minutes will be written for every meeting from now on and will be distributed the same day to all stakeholders, so there is only a limited delay for anyone unable to attend the meeting. Any other information such as changing user requirements will be sent out immediately to the stakeholders if it will affect their work or as seen fit by the project manager otherwise.

Given that each stakeholder has a different interest in the project, the information they require will be somewhat different. As mentioned, progress reports will continue to be sent to all stakeholders for total transparency, with the smaller reports that are more technical only going to the development team. In particular, the development team will need to be made aware of any information concerning change of scope, change of timeline or user requirements, and items' status in acceptance testing. The other departments assisting with the project will only need to be made aware of tasks they need to help with rather than other parts. Lastly, external stakeholders, including the users, will need to be mindful of impediments to the project or whenever deliverables from the team are ready.

Information being delivered regularly is an excellent step toward success, but it is virtually useless if stakeholders cannot access it. Moving forward, the project manager will locate any information being shared in a central repository such as a wiki that all users will be able to access simply by logging in. Since it is digital, they can share files instantly, and users can be notified when new files are added. The team should utilize a system that allows for

custom tags and groups to be set so that the project manager can dictate easily what stakeholders get notified depending on the file contents. As information is updated, such as a change in requirements, the files should be archived and replaced with a new version to remain available for reference, but the users will have the latest information. All other information being shared should be communicated through a team group chat such as Microsoft Teams.

The project manager should always send information being collected for the team. By having the information only distributed by this person, stakeholders can trust that the information being sent is accurate and up to date. If the project manager is unavailable, the team should send a notification to a secondary team member, such as the business analyst, who can fill in for the project manager as necessary so that critical information is always distributed.

Project Planning and Controls

Essential for the project's overall success, the project manager will need to document the project's plan and controls that the team will use to keep the project on track. These, like the charter, will be able to be referenced later in the project to ensure that they are staying within the scope and within what was approved.

Project Scope Planning

This project's general scope is to upgrade the clinical data warehouse to receive data feeds from wellness vendors and onsite clinic data. This data will be used for new employer group analytics, reporting processes, and activity reports. The current tools will display wellness vendor requirements and generate SQL Server Reporting Service (SSRS) reports from user-entered parameters. The system can display information to users in three file-formats, including Excel 97-2003 (XLS), Portable Document Format (PDF), or Comma Separated Value (CSV).

The headers in the reports will be "user-friendly" and will not display the direct database column name.

All development for this project will be completed using the MS Visual Studio language and a SQL environment. The system will conduct reporting on an SSRS deployment server. In terms of software requirements, the new reports will use SQL Server 2008 R2 and Teradata 14. Development software will include MS Visual Studio 2008 Shell, SQL Server Business Intelligence Development Studio, and SQL Server Management Studio.

Below is a sample requirements traceability matrix that the project manager could use to ensure that requirements are met. It is assumed that this is only a subset of needs; the actual project requirements will vary and will be more extensive based on additional factors such as unlisted user requirements, architecture, design, and existing database structure.

Req. ID	Requirement	Phase	Test Case ID
1	The database will receive input from wellness vendors	V1	TC-01
2	The database will transform data input into employer group aggregate data	V1	TC-02
3	That database will produce activity reports	V1	TC-03
	…		
37	Users can generate custom reports using SQL Server Reporting Services	V2	TC-37
38	Reports can be viewed in XLS, PDF, or CSV file formats	V2	TC-38

Figure 2 This requirements traceability matrix shows a sample of different requirements that a project manager might have for this project. The matrix lists the requirement and defining ID, the phase the team will complete it, and the associated test case ID.

Based on the project's Microsoft Project file, below is the existing work breakdown structure. The work breakdown structure will provide the entire team a hierarchical view of the project as it highlights all of the tasks grouped by similar features. Along with the tasks, each will give the users an idea of how long the job will take to complete and the start and end date for each.

WBS ID	Task Name	Duration	Start	Finish
1	Wellness Vendor Project	232 days	Tue 2/14/17	Sun 12/31/17

1.1	Initiation	24 days	Thu 5/4/17	Tue 6/6/17
1.1.1	Budget dollar service request	5 days	Thu 5/4/17	Wed 5/10/17
1.1.2	Project Charter	2 days	Sat 6/3/17	Tue 6/6/17
1.1.3	Obtain Approvals	3 days	Fri 6/2/17	Tue 6/6/17
1.2	Planning	41 days	Sun 6/11/17	Mon 8/7/17
1.2.1	Project Plan	8 days	Sat 7/15/17	Fri 8/4/17
1.2.2	Resource allocation approvals	4 days	Sun 6/11/17	Mon 8/7/17
1.2.3	Communication	3 days	Sat 7/29/17	Wed 8/2/17
1.2.3.1	Develop Communication Plan	0 days	Sat 7/29/17	Sat 7/29/17
1.2.3.2	Review/Approve Communication Plan	3 days	Mon 7/31/17	Wed 8/2/17
1.3	Project Kickoff	1 day	Fri 5/5/17	Fri 5/5/17
1.4	Execution	119 days	Thu 7/13/17	Fri 12/22/17
1.4.1	Vendor Wellness DB Tables	55 days	Thu 8/10/17	Tue 10/24/17
1.4.1.1	Create Tables	48 days	Thu 8/10/17	Fri 10/13/17
1.4.1.1.1	Create/Modify vendor database tables	41 days	Thu 8/10/17	Wed 10/4/17
1.4.1.1.2	Promote tables to QA	2 days	Thu 10/12/17	Fri 10/13/17
1.4.1.1.3	Validate tables	2 days	Thu 10/5/17	Fri 10/6/17
1.4.1.1.4	UAT	2 days	Thu 10/5/17	Fri 10/6/17
1.4.1.2	Promote Tables to Production	27 days	Sat 9/16/17	Tue 10/24/17
1.4.1.2.1	Create production ticket SR#	3 days	Sat 9/16/17	Wed 9/20/17
1.4.1.2.2	Deploy Tables	1 day	Thu 10/19/17	Thu 10/19/17
1.4.1.2.3	Validate Tables	3 days	Fri 10/20/17	Tue 10/24/17
1.4.2	Final Code Review-Module 1	27 days	Thu 9/7/17	Fri 10/13/17
1.4.2.1	Module 1 code review	21 days	Thu 9/7/17	Thu 10/5/17
1.4.2.2	Validate data accuracy	5 days	Mon 10/9/17	Fri 10/13/17
1.4.3	Final Code Review-Module 2	26 days	Thu 9/7/17	Thu 10/12/17
1.4.3.1	Module 2 code review	21 days	Thu 9/7/17	Thu 10/5/17
1.4.3.2	Validate data accuracy	4 days	Mon 10/9/17	Thu 10/12/17
1.4.4	Final Code Review-Module 3	33 days	Thu 10/5/17	Fri 11/17/17
1.4.4.1	Module 3 code review	12 days	Thu 10/5/17	Fri 10/20/17
1.4.4.2	Validate data accuracy	4 days	Tue 11/14/17	Fri 11/17/17
1.4.5	Wellness Vendor Reporting	119 days	Thu 7/13/17	Fri 12/22/17
1.4.5.1	Requirements	108 days	Thu 7/13/17	Thu 12/7/17
1.4.5.1.1	Define Business Rules	32 days	Thu 7/13/17	Thu 12/7/17
1.4.5.1.2	Data Mapping	22 days	Thu 9/28/17	Thu 11/30/17
1.4.5.1.3	Report mockups	8 days	Tue 10/17/17	Wed 11/29/17
1.4.5.1.4	Documentation and Current Mockup Layouts	18 days	Thu 10/19/17	Fri 12/1/17
1.4.5.2	Requirements Approved	2 days	Sat 9/2/17	Mon 9/4/17
1.4.5.3	Report Development	52 days	Fri 10/13/17	Fri 12/22/17
1.4.5.3.1	Report file layout data mapping	16 days	Tue 10/31/17	Wed 12/6/17
1.4.5.3.2	Report UAT	8 days	Wed 12/13/17	Fri 12/22/17
1.4.5.3.3	Reports Approved by Business Owner/Vendor	2 days	Tue 12/5/17	Wed 12/6/17

1.4.5.3.4	QA	50 days	Fri 10/13/17	Wed 12/20/17
1.4.5.3.4.1	Test Case Creation	22 days	Fri 10/13/17	Sat 11/11/17
1.4.5.3.4.2	Test Case Execution	28 days	Mon 11/13/17	Wed 12/20/17
1.4.5.3.4.3	Deploy Reports to Production	2 days	Fri 11/10/17	Sat 11/11/17
1.4.5.3.4.4	UAT	27 days	Sat 11/11/17	Mon 12/18/17
1.4.5.3.4.4.1	Testing Support	8 days	Sat 11/11/17	Mon 12/18/17
1.4.5.3.4.4.2	Remediation	4 days	Fri 11/17/17	Wed 11/22/17
1.4.5.3.4.4.3	Production Support	14 days	Thu 11/23/17	Tue 12/12/17
1.5	Oversight	92 days	Thu 7/13/17	Wed 11/15/17
1.5.1	Project Manager	92 days	Thu 7/13/17	Wed 11/15/17
1.5.2	Architect	92 days	Thu 7/13/17	Wed 11/15/17
1.6	Go-Live	1 day	Thu 12/21/17	Thu 12/21/17
1.6.1	Go-Live Milestone	1 day	Thu 12/21/17	Thu 12/21/17
1.6.2	Module Reports in Production- Go live	1 day	Thu 12/21/17	Thu 12/21/17
1.7	Monitor and Control	107 days	Tue 2/14/17	Thu 7/13/17
1.7.1	Support	107 days	Tue 2/14/17	Thu 7/13/17
1.7.1.1	Define Support Process	3 days	Tue 2/14/17	Thu 2/16/17
1.7.1.2	Review Support Process	3 days	Fri 2/17/17	Tue 2/21/17
1.7.1.3	Support Process Approved	0 days	Thu 7/13/17	Thu 7/13/17
1.8	Closing	4 days	Wed 12/27/17	Sun 12/31/17
1.8.1	Lessons Learned	4 days	Wed 12/27/17	Sun 12/31/17

Figure 3 This chart shows the current work breakdown structure for the project. The work breakdown structure offers all of the tasks grouped as a hierarchy, in addition to the time to complete the task and the start and end dates.

In its simplest form, a responsibility assignment matrix lists a summary of the tasks that need to be accomplished for a project and the person or roles responsible for the tasks (Larson & Gray, 2018). Below is a sample matrix listing the primary deliverables with the team member responsible for or supporting the task's delivery.

Deliverable	**BA**	**Dev & DBA**	**PM**	**Arch.**	**QA**	**SME**	**IS Mngr.**	**Sponsor**
Budget Request	S		S				R	S
Project Charter	S		R				S	S
Project Plan	S		R			S	S	
Communication Plan			R					
Vendor Wellness DB Tables		R		S	S	S		
Module 1 Code		R		S	S	S		
Module 2 Code		R		S	S	S		
Module 3 Code		R		S	S	S		
Reporting Business Rules	R					S		
Report Mockups	R			S				S
Reporting Documentation	R		S			S		

Reporting Test Cases					R			
Support Processes		R	S		S			

Figure 4 This responsibility assignment matrix is a sample of the most extensive tasks with the responsible stakeholder (marked with 'R') for seeing the task through to completion or will support (marked as 'S') the stakeholder accountable.

Project Schedule Planning

In any development project, identifying predecessors is crucial to ensuring a project is kept on track and completed logically. Predecessors have linked tasks that enforce and dictate that a particular task cannot be started until its linked counterpart is completed. For example, a database administrator cannot deploy a new table to production until the table is tested and verified for accuracy. Quality assurance cannot test the table itself until the database administrator creates it. Suppose a project team does not correctly identify predecessors before starting a project. In that case, the project will likely be delayed because of needing to backtrack through earlier tasks that were not linked.

The following tasks were identified as predecessors for the project team. It is important to note that the team only identified predecessors that would affect the project schedule, as shown below. The project contains additional tasks that would affect the critical path, but they are not direct predecessors, rather components of other milestones completed during that particular milestone. If more tasks need to be added, the project manager should check them for any potential predecessors as they very well may alter the timeline negatively.

Task ID	Task Name	Predecessors
1	Project Plan	
2	Resource allocation approvals	1
3	Develop Communication Plan	
4	Review/Approve Communication Plan	3
5	Create/Modify vendor database tables	
6	Promote tables to QA	5
7	Validate tables	6
8	UAT	7
9	Deploy Tables	

10	Validate Tables	9
11	Module 2 code review	
12	Validate data accuracy	11
13	Module 3 code review	
14	Validate data accuracy	13
15	Test Case Creation	
16	Test Case Execution	15
17	Remediation	
18	Production Support	17
19	Define Support Process	
20	Review Support Process	19

Figure 5 This chart provides a quick reference of all of the tasks with predecessors to be completed before the developer can start the task. Not all tasks are shown, only those that are a predecessor or are a successor.

Looking at the project in its current state, there are a few examples of where predecessors could be scheduled in a more meaningful manner. Most notably, there is an issue seen for the tasks under the summary task "Create Tables." Currently, validating the tables and user acceptance testing are successor tasks to promoting tables to QA. However, the predecessor is scheduled after the successors, which is not possible in practicality. Instead, the project manager should order the tasks to have the predecessor before the successors. While this will not affect the final deadline, they should update it for best practice. This example is shown below.

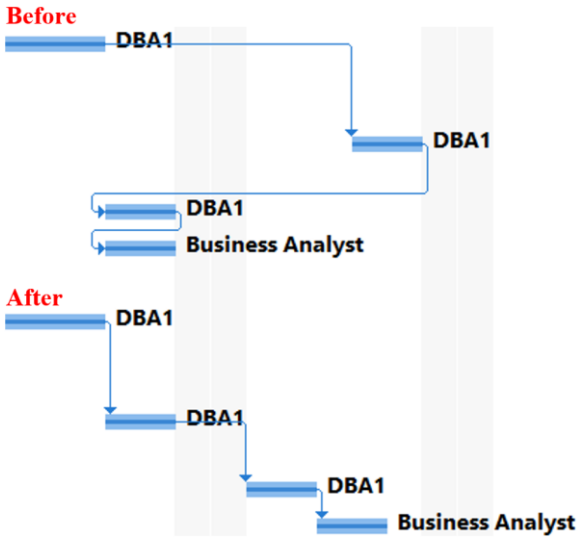

Figure 6 This screenshot shows an example of a series of successor tasks before and after changing the order of them. Before, there were successor tasks scheduled before the predecessor. After updating the order, it flows in a proper line of succession while still taking the same amount of time.

Concurrent Tasks

According to Larson and Gray, there are multiple ways to shorten a project's timeline both with and without altering the budget. Some standard methods include outsourcing, scheduling overtime, and limiting the project's scope (Larson & Gray, 2018). Given that the project is already over budget and there is a fixed deadline for completion, the desirable plan of attack would be fast-tracking or completing tasks simultaneously.

However, completing tasks concurrently may not always be desirable for every project or every task. For example, it may be feasible for one developer to be building a user interface for a web application while the database administrator is creating new database tables. Still, it would not be practical for the databases to be built simultaneously because the systems architect is determining what the structure would look like. Having these two tasks running in parallel could result in unnecessary changes for the database tables should the architect modify the information they already shared with the database administrator.

Another consideration for scheduling concurrent tasks is to not over-allocate resources. In theory, it would be desirable to develop two unique jobs to be done in tandem, but this is impossible if they are to be done by the same developer unless the duration changes. Instead, an option is reassigning one task to another developer, but there will always remain a fixed allocation of resource hours without extending into overtime.

Assuming the project can only be modified moving forward into the final milestone, a few tasks could be reorganized to be worked on simultaneously. One significant change that will be updated is to adjust the start of quality assurance (QA) to begin earlier. While it is not always desirable to test while development is actively occurring, it is possible. The QA resource would start testing existing areas of the project that are completed even if the rest of the milestone is not

finished. The team can ensure that QA will not prevent the project from being completed on time by adjusting this. Additionally, it would be possible for user acceptance testing to coincide with QA and development as long as it is clear to the users what specifically is still being worked on versus what is completed and can be tested.

Task Durations

The project itself will take an estimated 232 days to complete based on the project plan's information. Each task in the project varies in the amount of time it will take to complete, as shown earlier in Figure 3. For this project, task duration will be one of the easiest aspects to modify to reduce the project timeline. Given that some tasks take longer than others, it would only be feasible to alter upcoming tasks that take longer than a week instead of shorter ones. It would be much easier to make up a day or two on a task that takes 15 days to complete than making up a day on a three-day task.

Changes made to the task durations would include reducing module 3 code review from 12 days to 11 days, documentation and current mockup layouts from 18 days to 16 days, report file layout from 16 days to 15 days, test case creation from 22 days to 21 days, and test case execution from 28 days to 26 days. Reducing the time on these tasks will reduce strain on the project’s critical path and dependencies.

Planned vs. Actual Dates

When planning any project and creating a schedule, everything will be based on educated estimates. Still, they will always be just an estimate meaning there may be a level of guessing involved. As the project is underway, the project manager will be responsible for maintaining the project schedule. This ensures that planned dates for starting or finishing a task get converted to actual dates or when the task officially started or finished. Not only does this promote accuracy,

but it will allow estimating against the project timeline to become more accurate and verifiable (Larson & Gray, 2018). For example, if one task is planned to end on the first but does not end until the eighth, any estimates related to this task will be off by a week.

Project Cost Planning and Control

According to Investopedia, a cost-benefit analysis measures the expected project benefits related to the overall cost. The costs and benefits are somewhat subjective as they can include both tangible and intangible factors (Kenton, 2020). In theory, the benefits should always outweigh the costs to be successful.

This project's costs include the physical budgeted cost of $441,650, any additional costs such as training the developers to understand the new programming language, and future costs of the project after it is released, such as the cost to maintain it. Additional charges may include marketing and promotional materials and the time spent on the project from other departments such as the legal team. Given that some of the expenses, such as training the developers on the new language, can be transferred to future projects, this cost can be seen as a benefit but also makes calculating a “true” cost particularly imprecise to be used as a final figure.

The project's benefits include the increased benefit of having these new reports and data, improved customer or client satisfaction, potential revenue stream if the company is charging clients for the new product, and overall gains made in the industry compared to the competition. Having a higher standing in the industry cannot be directly tied to a specific dollar amount, as brand awareness alone will be a significant benefit for the company. Before the project even began, one can assume that the senior management team that decided to pursue the project carefully looked at the costs and benefits and determined that the benefits outweigh the costs in this situation.

Total Cost of Ownership

Any project needs to consider the actual costs to complete the project and all expenses related to the project even after it is completed. The total cost of ownership looks at additional project factors such as operational and maintenance costs and the cost to retire a system once its usefulness is depleted (Doig, 2015). Calculating an exact total cost of ownership for a new system or project such as this is not an exact science; instead, it is based on estimates. Part of the costs includes bug fixes that cannot be expected ahead of time. Based on an interview with an experienced project scrum master, some conservative companies can estimate double the initial cost for the total cost of ownership (W. Edwards, personal communication, January 15, 2021). Based on this, the total cost of ownership for this project could be close to $880,000.

Budget vs. Actual Costs

Immediately following the third module's completion, the budget's actual spend was $54,458 for capital expenditures and $3,825 for operating expenses or 20% and 2%, respectively. The project's total budgeted amount is $277,988 for capital and $163,662 for operating expenses totaling $441,650. However, based on the previous project manager's progress report, the project was tracking 20% over budget, meaning the actual cost at completion will be an estimated $529,980, which is significantly higher than initially approved and budgeted for.

Earned Value

A project's earned value is an excellent methodology and tool for a project manager to view the project's baseline. This reporting tracks a project's timeline based on costs. The following chart shows the planned value compared to the earned value and actual cost for the project from the start through completing the second module. Given that the planned value remains higher than the earned value, the chart shows that the project is currently behind

schedule and likely will be delayed. These values will need to be reversed to be on schedule or ahead of schedule.

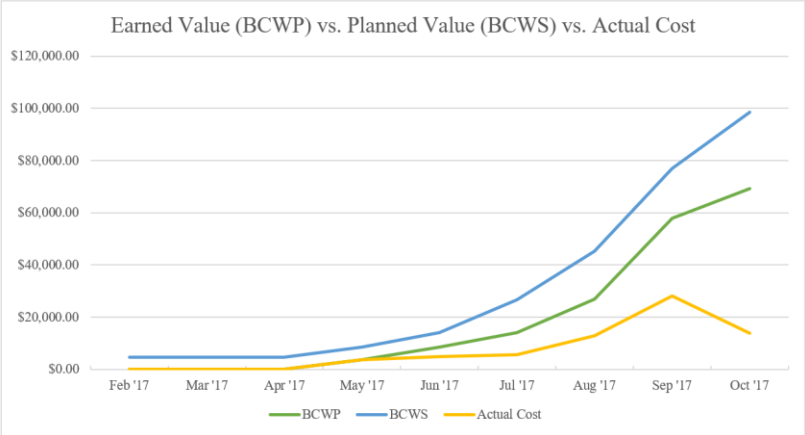

Figure 7 Produced from the project information, this chart illustrates that the project is behind schedule through the completion of the second module.

Plan to Finish the Project

As the project team is finishing up the final module, it is crucial to carefully control the project's cost and closely monitor these metrics. The project is already over budget and likely will end up this way. The main goal would be to lower the overall cost variance, where possible, instead of becoming even more over budget. Since changing anything major at this point to decrease cost will likely be very difficult, it would be beneficial to focus specifically on making sure all expenses are accounted for and are justifiable. All team members that are directly billable to the project should begin tracking their time for each task and should immediately move onto new tasks when prior ones are completed. The team will need to stick to the approved task list and allocated time as working on other unapproved tasks will increase the cost variance. If successful, the reporting should show a lower cost variance percentage and indicate the chart's earned cost higher than the planned cost. Additional metrics to watch are meeting the key performance indicators mentioned later.

Project Risk Planning

There is always some uncertainty with any project because, again, project timelines are based on estimates. This is why it is essential for project managers to closely track projects (Larson & Gray, 2018). According to a personal interview with a certified Advanced Scrum Master, there is an increased uncertainty level as projects get larger because there are more variables in play (W. Edwards, personal communication, January 15, 2021). For example, a simple project that will last two weeks will have less uncertainty than a year-long project because it is much more straightforward and easier to plan for.

This project and project team has been tasked with many unique challenges that have led to much more uncertainty than usual such as losing a project manager, coding in a new language and environment, and already being tasked with a tight deadline. The most significant uncertainty for this project can be attributed to the project schedule. As noted, the team did not break out the project into specific tasks; instead, the schedule gives a broad overview of a larger subset of tasks. This will cause the estimating to be potentially more inaccurate. Additionally, there is a level of uncertainty directly attributed to the fact the team has never worked with the programming environment. While they did have a training period before the project began, there is more of a potential for unseen hiccups along the way.

Handling uncertainty in a project is a mix of both proactive and reactive behavior from the project manager and the team (W. Edwards, personal communication, January 15, 2021). Proactively, the team can work to reduce uncertainty by ensuring the schedule remains up to date. Every time a date is adjusted from planned to actual, the uncertainty decreases as it becomes more accurate. Additionally, the team can ensure that activities are adequately tasked out into smaller tasks so that estimates can be applied based on that information.

As important as it is to be proactive, it is equally important to be reactive as things do happen. Every effort should be made to keep the schedule on track, which is why slack and padding are built into a project (Larson & Gray, 2018). If time allows, the team should counter issues with a "lessons learned" exercise for the entire team to mutually learn from the experience and prevent it in the future.

The most significant threats to this project include not having proper leadership from a project manager and the extremely tight deadline that the team is on. Any unexpected event could cause the project to be delayed at this point. The project manager can identify threats by using a risk profile or risk breakdown structure (Larson & Gray, 2018). For obvious reasons, not having a project manager on a more extended project is not ideal. Some situations could arise from this, including simply not having someone dedicated to maintaining the project schedule to ensure the team remains on track.

Additionally, the project timeline inherently is a threat to itself in this situation. This project has a strict deadline just before the end of the year and is currently tracking behind schedule. The team needs to be performing at its fullest potential, but they cannot do so if they are not working due to vacations, illness, or time off. Since the project is set to end near the holidays, there is a concern for a team member to take a few days off, which could ultimately hurt the deadline.

The threat to overcome most urgently from the major threats identified would be not having a team project manager. The team can overcome this in two separate ways. The ideal solution to this problem would be to immediately assign a project manager to the team from a different project or hire an experienced project manager who would quickly learn the project. However, since the biggest concern is that there is no one to manage the project schedule, the

other option, albeit less ideal, would be to have someone from the team such as the business analyst or IS manager be tasked with taking over this responsibility until a project manager is assigned.

Overcoming the project timeline's strain will not be as easy in concept because it is impossible to control illness or emergency for team members. That being said, it is possible to plan for any scheduled time off. The project manager, when hired, should immediately be looking at any upcoming time off for the team. If it is longer than a few days or would be during a time when that person's work would delay a successor task from starting, a resource should be borrowed from another project team as permitted. If this is not possible, the project manager should reorganize the tasks to accommodate the schedule so being off does not delay work for another resource. Suppose individuals are off for an extended time due to unexpected illness or other related issues. In that case, the project manager should immediately request that another resource be made available to fill in. As long as everything is documented correctly, the new resource should be able to pick up the work relatively quickly until the team member can return to work.

Risk Control

After risks and threats are identified, a project manager needs to understand how impactful they could be to a team, or rather, how much they could hurt the team. A common way to accomplish this is to look at the impact and the probability. The impact of a risk is how detrimental it could become, whereas the probability is the likelihood of it occurring (Larson & Gray, 2018). For example, the probability of not having a project manager assigned causing issues poses a moderate chance of occurring and with high impact. On the other hand, a resource taking a planned day off or needing to take a day off for being sick by the end of the project

would be a high probability with a low to moderate impact per day off. After a probability and impact are identified for each risk, they can be plotted on a matrix, with one axis being the probability, the other being the impact. As risks become higher in their probability and impact, the greater the risk is to the project (Larson & Gray, 2018). In this scenario, the project manager vacancy would pose a significant risk, and resource time off would pose a moderate risk.

Contingency funding is essentially a portion of excess funding that should not be planned on but is available for unavoidable circumstances (Larson & Gray, 2018). This funding is simply additional cash reserves that can be tapped into but are usually more guarded by the project sponsor as it ultimately costs them more money. Likewise, a project will usually have a time buffer or spare time to be "acceptable" to go over the deadline. According to Larson and Gray, the more uncertain a project becomes, the more buffer is needed to cushion the project (Larson & Gray, 2018).

Pending the project was managed and planned for properly, the project likely has built-in contingency funding to handle unavoidable risks. Unfortunately, the project does not have a time buffer as the marketing materials demand a fixed, predetermined release date. However, this could potentially be built in after the fact. For example, documentation for the project systems is not a requirement for the project release from a user standpoint. In theory, the team could delay certain tasks like this until after the project deadline. This would simulate a project buffer without actually having one built-in. The project manager could simulate a contingency fund by reducing the budget by a certain amount and instead move the difference into a separate pot that would act as a contingency fund. These would put a strain on the team in the present but add additional padding for future use.

Project Quality Planning

For this project, key performance indicators will be introduced on a team or project level and an individual level to get a quantifiable look at the project and each team member. Team key performance indicators will include the overall cost variance from the budget, the number of adjustments to the project schedule, and planned value. The budget will be one of the most significant factors for the team itself, so it is necessary to include the cost variance and planned value as negative changes to those will indicate a budget problem. The number of schedule adjustments is a way for the project manager to see how often they need to adjust the schedule as changes likely indicate falling behind.

Key performance indicators for the individual team members include the percentage of on-time tasks, the number of issues found in testing, and the number of iterations needed to finish a task. While the rate of on-time tasks can indicate a developer is not meeting the schedule, the number of testing issues and the number of iterations can show a developer struggling to finish tasks adequately on the first attempt. Additional key performance indicators should be evaluated after the project is released, such as the revenue attributed to the project release as well as user satisfaction and the number of bugs reported.

Defining quality is particularly dependent on the standards of acceptability at the company already. For example, this company's standard may be much higher (or lower) than a competing company that could accomplish the same project scope. Quality should very closely mirror similar projects to add a level of integrity. The project team should never trade quality in favor of the schedule, budget, or scope; the team should change one of these other variables instead. Essentially, a team should never knowingly decrease its quality or standards to meet constraints (Steinman, 2017). For example, in this project, the budget is overrun, and they are

behind schedule. Instead of decreasing the final product's quality, either the budget will need to grow further, or additional time will need to be made up to meet the fixed deadline.

A factor in project quality is what is acceptable to the product owners or users. If a project meets the users' expectations, it can be considered quality, but if it does not serve its purpose, it would not be a quality product. Likewise, a project will need to meet the original requirements, both business and technical, to be quality.

Measuring quality for a project is crucial in ensuring that the project is successful but that it was worth the company's investment and met its expectations. Projects that are not quality run the risk of wasting company funds and resources and tarnishing reputation. Quality will be measured using metrics and key performance indicators already mentioned and through evaluations. After the project is completed, the team will also measure the quality through surveys, feedback, and interviews.

Key performance indicators will be the single largest group for measuring quality. They are typically easy to report using existing data and can be charted to map performance over time. The majority of the indicators identified above can be reported using project management software such as Microsoft Project or can be quickly calculated by the project manager. All of the data will be compiled by the project manager and stored in a central location for transparency across the team and company. They will build the data into a shared dashboard or webpage that anyone can access on demand for the most recent metrics.

The team can identify additional insight into a project's quality through evaluations. The project manager should routinely evaluate the individuals on the project to ensure they understand what is expected of them and ensure that the team members are providing feedback from their point of view. Depending on the team size, the project manager may not release

individual issues arising amongst the team. Still, if they solicit feedback from the group, it might become apparent before it causes an actual problem that affects the quality. After the project is released and completed, it is crucial to follow up with users and continue to monitor metrics to ensure the team achieved an acceptable level of quality.

Metrics such as the number of bugs reported or usage of the system can help determine this, but the team should also look at user feedback entries and survey results or even interview users to get a more in-depth insight. Collectively, management will measure these metrics and standards of quality after the project is done to deem if it was successful or not despite all of the challenges the team faced from the beginning. Without learning what was successful or unsuccessful, the team will only set themselves up for failure when they start their next project. This way, they might be able to learn from the roadblocks they encountered.

References

Doig, C. (2015, November 19). Calculating the total cost of ownership for enterprise software. CIO. https://www.cio.com/article/3005705/calculating-the-total-cost-of-ownership-for-enterprise-software.html

Edwards, W. (2021, January 15). Scope of a Project (T. Latshaw, Interviewer) [Personal communication].

Kenton, W. (2020, July 7). Cost-Benefit Analysis. Investopedia. https://www.investopedia.com/terms/c/cost-benefitanalysis.asp

Larson, E. W., & Gray, C. F. (2018). Project management the managerial process. (7th ed.). New York Mcgraw-Hill Education.

Lencioni, P. (2016). The Ideal Team Player : How to Recognize and Cultivate The Three Essential Virtues : A Leadership fable. Hoboken, Nj Jossey-Bass.

Steinman, J. (2017, May 5). Defining and measuring project quality. Control Engineering. https://www.controleng.com/articles/defining-and-measuring-project-quality/

Stobierski, T. (2019, October 7). How to Motivate Employees: 5 Data-Backed Tips for Managers. Northeastern University. https://www.northeastern.edu/graduate/blog/how-to-motivate-your-employees/

Widman, J. (2008, October 9). IT's biggest project failures -- and what we can learn from them. Computerworld. https://www.computerworld.com/article/2533563/it-s-biggest-project-failures----and-what-we-can-learn-from-them.html

Wilson, F. (2020, February 1). Project Baseline: An Overview From Agile Project Manager. NTask. https://www.ntaskmanager.com/blog/project-baseline