|  |
| --- |
| CS-480 Senior Capstone I |
| PitchMetrics Project Management Plan |
| Version 4.0 |

|  |
| --- |
| Maschino, Jeremy  6-28-2019 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Version | Implemented By: | Revision Date | Approved by: | Approval D9te: | Reason: |
| 1.0 | **Jeremy Maschino** | **2/17/2019** | **Dr. Jacobs** | **2/18/2019** | **Project Management Plan -First Draft** |
| 2.0 | **Jeremy Maschino** | **2/24/2019** | **Dr. Jacobs** | **2/24/2019** | **Project Management Plan -Second Draft** |
| 3.0 | **Jeremy Maschino** | **2/25/2019** | **Dr. Jacobs** | **2/27/2019** | **Project Management Plan -Final Draft** |

****

**Table of Contents**

**Deliverables Page**

Project Proposal p. 3

Research, Comparison & Technical Specifications p. 4

Scope Statement p. 6

Timeline v1.0 p. 7

Cost Estimate p. 8

Work Breakdown Structure p. 9

Network Diagram p. 11

Gantt Chart p. 12

Critical Path Analysis p. 13

PERT Analysis p. 14

Technology Acceptance Model p. 15

Quality Metrics p. 16

Testing Plans p. 17

Risk Analysis p. 19

Communication Plan p. 20

Stakeholder Management Plan p. 22

Organizational Chart p. 24

Project Proposal Conclusion p. 26

**Figures & Tables Page**

Figure 1: Timeline v1.0 p. 7

Figure 2: Visual Work Breakdown Structure p. 10

Figure 3: Network Diagram p. 11

Figure 4: Gantt Chart p. 12

Figure 5: Critical Path Analysis p. 13

Figure 6: Technology Acceptance Model p. 15

Figure 7: Quality Metrics p. 16

Figure 8: Power-Interest Matrix p. 22

Figure 9: PitchMetrics Organizational Chart p. 24

Figure 10: Example Organizational Chart p. 24

Table 1: Computer Systems p. 4

Table 2: Doppler Radars p. 4

Table 3: Operating Systems p. 4

Table 4: Databases p. 5

Table 5: UIs with Software Language p. 5

Table 6: Cost Estimate p. 8

Table 7: PERT Analysis p. 14

Table 8: Test Items p. 17

Table 9: Test Environment p. 18

Table 10: Risk Analysis p. 19

Table 11: Communication Plan p. 20

Table 12: Stakeholder Management Plan p. 22

Table 13: Tech Specs p. 26

**PitchMetrics Project Proposal**

**Created by:** Jeremy Maschino **Date:** 1/13/2019

**Project Title**: PitchMetrics **Project Number:** PM1

**Project Start Date:** 1/13/2019 **Projected Finish Date:** 7/8/2019

**Project Manager:** Jeremy Maschino [maschinoj90@uiu.edu](mailto:maschinoj90@uiu.edu)

**Topic:** Within the sport of baseball, the use of analytics, metrics, and statistics has become increasingly more popular. Major League Baseball, and its member teams, have now employed the use of doppler radars to gather information on how a baseball moves. For this project, the goal is to create a system that implements a database that stores pitchers, their pitches, the date and gathers information from a Doppler Radar that tracks the flight of a ball, that is pitched, and displays relevant information to the user such as velocity (MPH), Spin Rate (RPM), a recreation of the pitch path and the location of the pitch when it enters the strike zone.

**Problem:** The problem low to medium-budget, athletic programs face is not that these systems do not exist, but that they are expensive. The TrackMan system seen in Major League Stadiums costs $30,000 (Boddy, Driveline Baseball) and the cheaper alternative, Rapsodo, costs $4,000 plus a subscription fee of $500 (Rapsodo), but implements the use of a high-speed camera, which can be less accurate. Creating a more cost-effective model would allow more low-budget athletic programs, such as the Upper Iowa University Baseball Program to afford a system that measures the variables stated in the Topic section.

**Solution:** The solution is to create the PitchMetrics system. A cloud stored database would be created to store the different pitchers, with pitches they throw, and the date they are throwing. Along with that, a Doppler Radar would be created by following MIT’s Doppler Radar lab, which describes how to build a Doppler Radar, to gather velocity of the previous pitch, and the spin rate. A graphical user interface is then used to display all the metrics gathered by user input (pitcher, date, pitch type) and the Doppler Radar (velocity, spin rate), along with a recreation of the pitch path and where the pitch crosses the strike zone since Doppler Radars produce 3D data. All information gathered would be sent to the database so that any pitch could be pulled up, regardless of when it happened.

References

Boddy, K. (2018, February 01). Rapsodo, Trackman, and Pitch Tracking Technologies – Where

We Stand. Retrieved from [https://www.drivelinebaseball.com/2016/11/rapsodotrackman](https://www.drivelinebaseball.com/2016/11/rapsodo-trackman-pitch-tracking-technologies-stand/)

[pitch-tracking-technologies-stand/](https://www.drivelinebaseball.com/2016/11/rapsodo-trackman-pitch-tracking-technologies-stand/)

Rapsodo. (n.d.). Products – Rapsodo. Retrieved from <https://rapsodo.com/shop-2>

**PitchMetrics Research, Comparison, & Tech Specs**

**Created by:** Jeremy Maschino **Date:** 1/20/2019

|  |  |  |
| --- | --- | --- |
| Computer System | HP EliteBook Folio 9740m | MacBook Air 13in |
| Cost | $0 (Already purchased) | $1400 |
| RAM | 12 GB DDR3 | 8 GB DDR3 |
| Processor | Intel Core i7 @ 2.10 GHz | Dual Core Intel Core i5 @ 1.66 GHz |
| Storage | 256 GB | 256 GB |
| OS | Windows | macOS |

**Computer System:**

*Table 1 Computer Systems*

The choice of computer system is the HP EliteBook Folio 9740m. Mainly, because it is already purchased, so no extra cost for the computer system, and is personalized to me already, so there is no learning-curve associated with purchasing and learning a new laptop.

|  |  |  |
| --- | --- | --- |
| Doppler Radar | MIT’s Doppler Radar Lab | Garmin GMR Fantom 18 Radar |
| Cost | $400 Estimated | $2,000 |
| Pre-Fabricated? | No | Yes |

**Doppler Radar:**

*Table 2 Doppler Radars*

The choice of a doppler radar is the MIT’s Doppler Radar. MIT provides a lab that gives instructions on how to create your own doppler radar at home. Not only is it cheaper, but it allows myself to write code specifically for the purpose of being used for PitchMetrics as I will assemble it myself and write the code for it as well.

|  |  |  |
| --- | --- | --- |
| Operating System | Windows 10 | Mojave 10 |
| Cost | $0 (already installed) | Free with Mac device |
| Support | All computer systems besides Apple (Mac) devices | Apple (Mac) devices |

**Operating System:**

*Table 3 Operating Systems*

The choice of Operating system is Windows 10. It is already pre-installed on my computer system that this project will be using. It provides more support for other, non-Apple devices, and has more support by applications that Apple does.

|  |  |  |
| --- | --- | --- |
| Database | MySQL | Access |
| Cost | $0 | Included with MS Office |
| OS Support | Windows, macOS, Linux | Windows |
| Partitioning | Composite, Range | No partitioning support |
| Better Performance | x |  |
| Security | Configured with SSL support | Username/Password |
| Number of users | Large, very scalable. | Small |

**Database:**

*Table 4 Databases*

The choice of database is MySQL. Cost and OS Support were not factors in deciding, but it was good to see those. The choice to choose SQL came down to its performance and ability to hold large amounts of data and the connectivity to that data it provides.

|  |  |  |
| --- | --- | --- |
| Programming Language | ASP.NET w/ C# | Java |
| Cost | $0 | $0 |
| Ecosystem | Microsoft Platforms | Open-source ecosystem |
| Database Connectivity | C# .NET Framework | JDBC |
| Interfaces | x | x |
| Type-Safe | x | x |
| Garbage Collection | x | x |
| Single Inheritance | x | x |
| Polymorphism | Invoke “virtual” keyword | Enabled by default |
| Support for Delegates | Serve as methods | Must find workaround |
| When to use? | .NET Framework  (Windows) | Linux or other non-Microsoft platforms |

**User/Web**

**Interface**

**W/ DB &**

**Software**

**App. Language:**

*Table 5 UIs with Software Language*

The choice of programming language is ASP.NET with C#. This allows me to make an interface that is connected to the web, as well as draw from a database of pitch & pitcher information. I chose ASP.NET and C# due to the .NET framework that is used on Windows machines, and because I have more knowledge in ASP.NET and C# than I do Java.

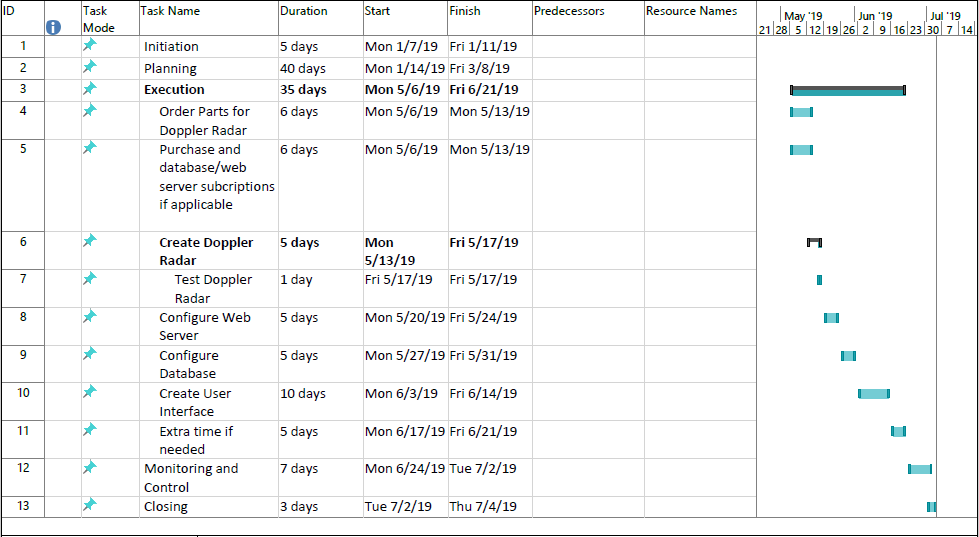
**PitchMetrics Scope Statement v1.0**

**Created by:** Jeremy Maschino **Date:** 1/28/2019

|  |
| --- |
| Project Justification:  Low to medium-budget, athletic programs cannot afford pitch tracking systems like high-budget college teams, semi-professional, or MLB teams. The TrackMan system seen in Major League Stadiums costs $30,000 and the cheaper alternative, Rapsodo, costs $4,000 plus a subscription fee of $500, but implements the use of a high-speed camera, which can be less accurate. The goal of the PitchMetrics team is to bring a more cost affordable, pitch tracking system. |
| Product Characteristics and Requirements:   1. User interface to gather pitch information and display pitch data to user 2. Working server with database that contains pitch information, connected with interface 3. Accurate Doppler Radar 4. Software that interprets Doppler Radar information. |
| Summary of Project Deliverables  Product-related deliverables: research reports, design documents, software code, hardware.   1. Working Server 2. Fully functional application for end-users 3. Connected and functioning database 4. Collected data reports |
| Project Success Criteria:  The user interface will depend on how successful the implementation of the Doppler Radar is. A good doppler radar will provide accurate information of which we can derive a pitch track, velocity, and spin rate. This information will then be displayed, through the user interface, to the end-user. The database and web server must be able to handle large amounts of entries and activity. Before a final PitchMetrics product is released, it must be thoroughly tested. |
| Summary:  Because there is a need for a cheaper alternative for pitch tracking, the idea of PitchMetrics was derived. PitchMetrics will include a user interface to first gather pitch information from the user, and then take in pitch information from the Doppler, and then display a pitch track, velocity, and spin rate to the end user. This information will all be stored in a database for end-users to recall for a summary of their bullpen. PitchMetrics will deliver a working server, fully functionals application for end-users, a connected and functioning database, and collected data reports. The project depends on a successful implementation of the Doppler Radar as that is where the bulk of the pitch information comes from. The database must also be able to handle large amounts of activity. |

**PitchMetrics Timeline v1.0**

**Created by:** Jeremy Maschino **Date:** 1/28/2019



*Figure 1 Timeline*

*v1.0*

The main components of the PitchMetrics system is the Doppler Radar, web server, database, and the user interface. First, I gave myself a week for all the stuff I need to purchase to make it to me. Then I gave myself a week to create the doppler radar and test it. I gave myself another week to configure the web server. After that I gave myself another week to configure the database. Following that I gave myself two weeks to create the user interface. This will implement doppler radar, the database and web server. I also gave myself a week of extra time, as well as I did not factor in weekends to the project as I will also use that as extra time to work on a component as needed. I gave myself 7 days plus a weekend for monitoring and controlling to test PitchMetrics in a live setting (pitcher bullpens) and make any necessary tweaks. Finally, I gave myself 3 days to prepare closing documentation.

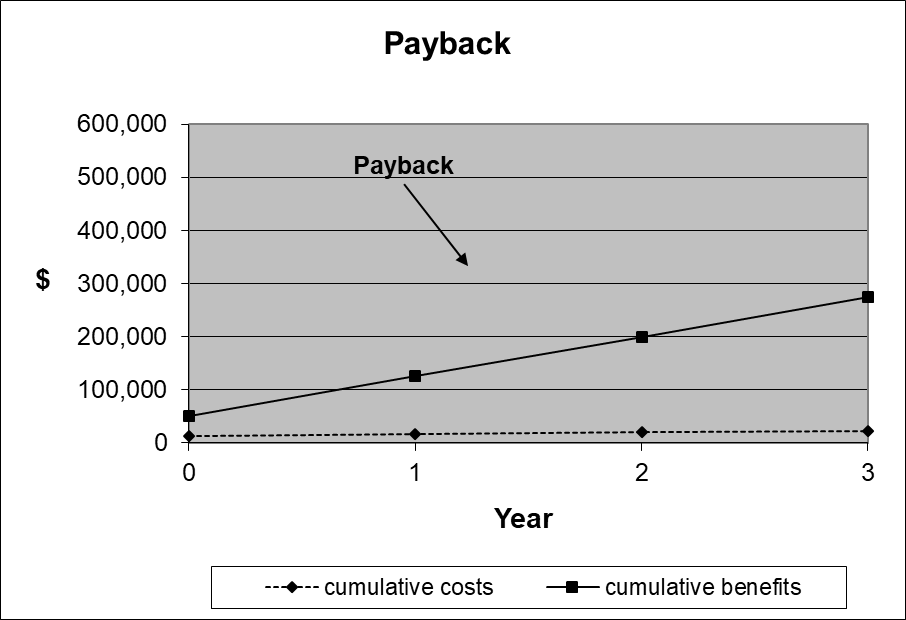
**PitchMetrics Cost Estimate**

**Created by:** Jeremy Maschino **Date:** 2/3/2019

*Table 6 Cost Estimate*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | # Units/Hrs. | Cost/Unit/Hr. | Subtotals | WBS Level 1 Totals | % of Total |
| WBS Items |  |  |  |  |  |
| Project Manager (Jeremy Maschino) | 224 | $35 | $7,840 | **$7,840** | **59%** |
| **1. Hardware** |  |  |  | **$1,400** | **10%** |
| 1.1 Laptop | 1 | $1,000 | $1,000 |  |  |
| 1.2 Doppler Radar | 1 | $400 | $400 |  |  |
| **2. Software** |  |  |  | **$4,000** | **30%** |
| 3.1 PitchMetrics Software Development | 200 | $20 | $4,000 |  |  |
| **3. Testing** |  |  |  | **$100** | **1%** |
| 3.1 Live Testing | 10 | $10 | $100 |  |  |
|  |  |  |  |  |  |
| **Total project cost estimate** |  |  |  | **$13,340** |  |

The cost estimate breaks down the time consuming, and cost affecting portions of the PitchMetrics Project. The laptop was already purchased but costed $1,000. The Doppler Radar is estimated to cost $400 to purchase the parts and assemble. Software development will be done by myself only. As well as live testing. The overall estimate to create this project is $13,340.

Along with the cost estimate we can then think about maintenance costs. Such as database maintenance, server maintenance, hardware maintenance, and software maintenance/updates. Since these will all be done by myself, I will add no cost to that as well.

Since the cheapest system like PitchMetrics goes for around $4,000. I believe we could market PitchMetrics at $1,500. For a $1,100 profit per system. Of course, if I were to expand the project to commercially produce these units. A revision of the estimate would be needed as we would hire people to do the system maintenance.

**PitchMetrics Work Breakdown Structure**

**Created by:** Jeremy Maschino **Date:** 2/10/2019

1. Initiation

1.1 Introduce PitchMetrics

1. Planning

2.1 Project planning documentation

2.2 Engage stakeholders

1. Execution

3.1 Order Parts for Doppler Radar

3.2 Purchase database/web server subscription if applicable

3.3 Create Doppler Radar

3.3.1 Test Doppler Radar

3.4 Configure Web Server

3.5 Configure Database

3.6 Create User Interface

1. Monitoring and Control

4.1 Test PitchMetrics

4.2 Add any changes (if needed)

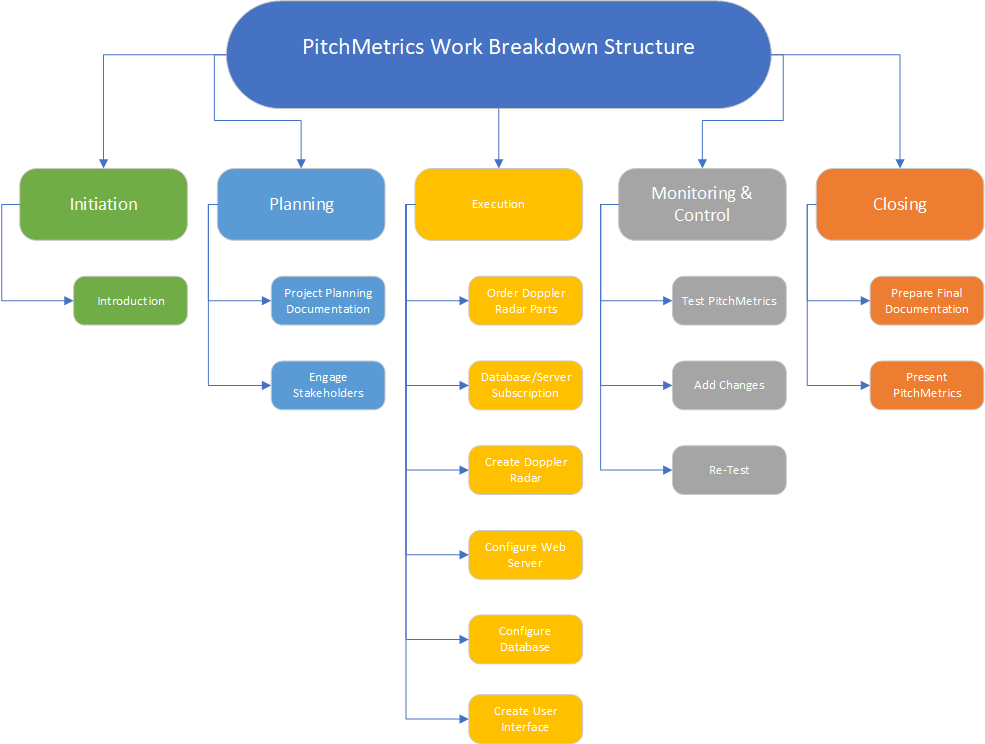
4.3 Re-Test

1. Closing

5.1 Prepare F­inal Documentation

* 1. Present PitchMetrics

**PitchMetrics Work Breakdown Structure**

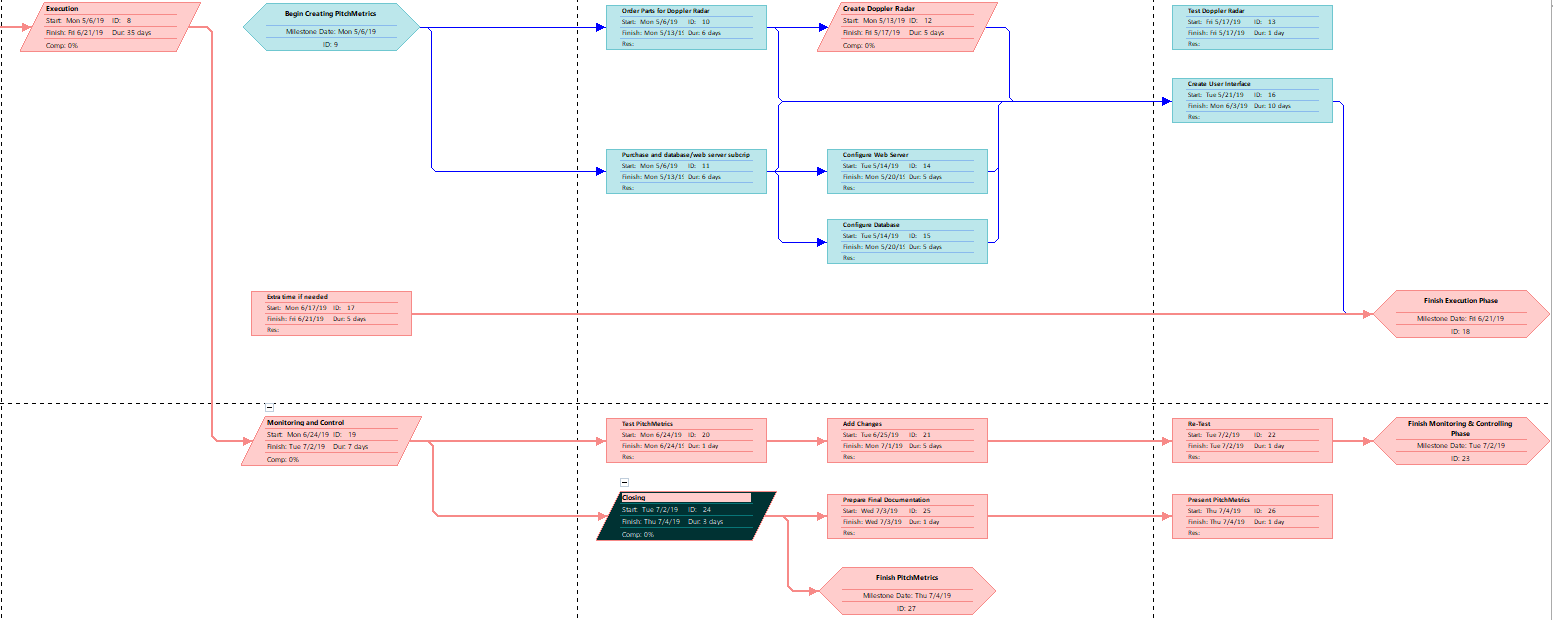
**Created by:** Jeremy Maschino **Date:** 2/10/2019

*Figure 2 Visual WBS*

PitchMetrics Initiation and Planning will be done first. Followed by that we have the Execution Phase, which includes ordering parts for Doppler Radar, purchasing database/web server subscription, creating Doppler Radar, configuring web server, configuring database, and creating the user interface. Then the Monitoring and Control phase will start. That will include testing the system, adding any changes, and re-testing. Then the Closing phase will start, which includes preparing the final documentation and presenting PitchMetrics

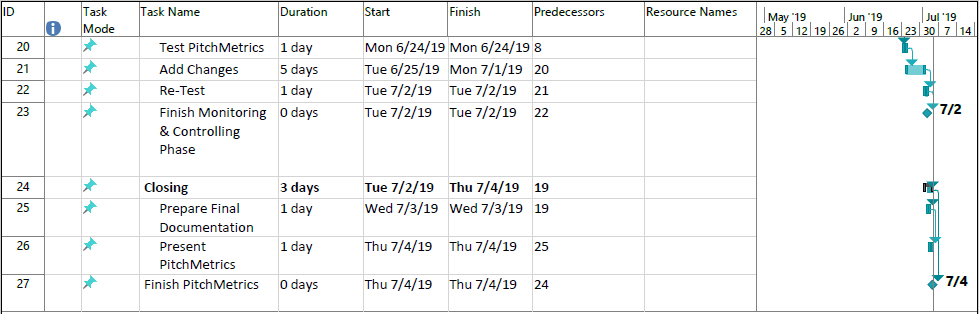
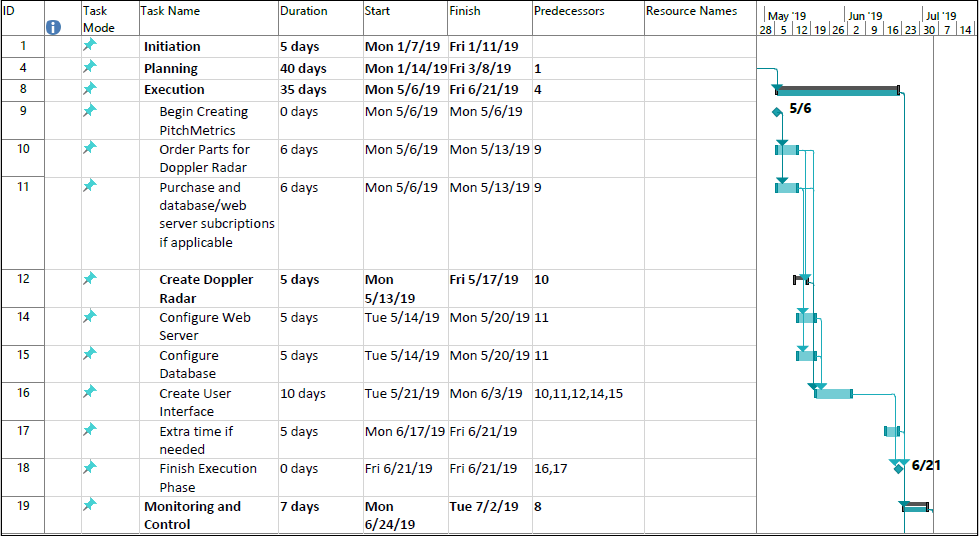
**PitchMetrics Network Diagram**

**Created by:** Jeremy Maschino **Date:** 2/10/2019



*Figure 3 Network Diagram*

**PitchMetrics Gantt Chart**

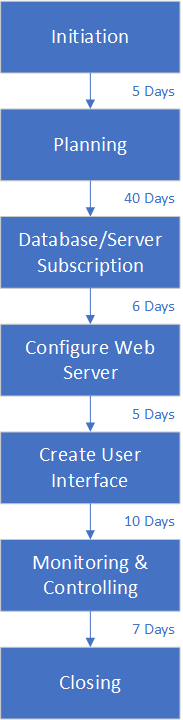
**Created by:** Jeremy Maschino **Date:** 2/10/2019

*Figure 4 Gantt Chart*

The Network Diagram and Gantt Chart can be viewed in conjunction. Some notable changes from the first Gantt chart includes the addition of milestones, and predecessors/dependencies. See the preliminary timeline for more information.

**PitchMetrics Critical Path Analysis**

**Created by:** Jeremy Maschino **Date:** 2/10/2019

The critical path is critical in showing us the path that PitchMetrics schedule depends on to stay on time. The critical path demonstrates the longest path for creation of the project. PitchMetrics critical path includes initiation (5 days), planning (40 Days), database/server subscription (6 days), configure web server (5 days), create user interface (10 days), monitoring & controlling (7 days), and closing (days).

*Figure 5 Critical Path Analysis*

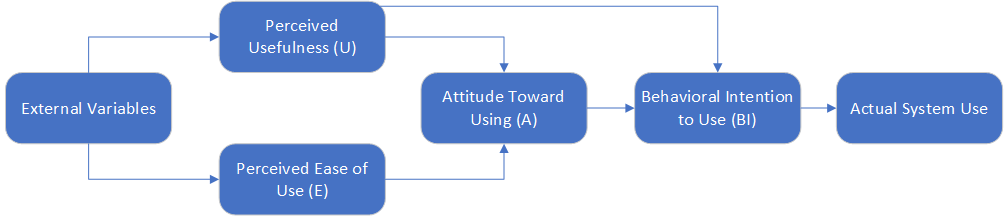
**PitchMetrics PERT Analysis**

**Created by:** Jeremy Maschino **Date:** 2/10/2019

*Table 7 PERT Analysis*

The PERT Analysis gives us an estimation on how long PitchMetrics should take given a optimistic outlook, a most likely outlook, and pessimistic outlook. Since the initiation and planning is almost over, I did not include their sub-tasks. After that, the longest task we see is the creation of the user interface. I believe it should take between 8-12 days. But most likely take 10 days. Also for extra time, I give myself 7 days if we are optimistic, to 3 days for a pessimistic view.

**PitchMetrics Technology Acceptance Model**

**Created by:** Jeremy Maschino **Date:** 2/17/2019

*Figure 6 Technology Acceptance Model*

The technology acceptance model allows us to define the path that PitchMetrics must take to be accepted by the end-user. External variables could be defined as the current market, such as are people willing to spend the money on a system like PitchMerics. Another external variable would be the popularity of baseball. Both are uncontrollable variables, but I believe that the current market is favorable and the popularity of baseball is high enough to create PitchMetrics.

Based off other similar systems, the perceived usefulness(U) is apparent. Many of these systems are in such high demand due to their usefulness. Many people now seek out facilities where systems like PitchMetrics are used.

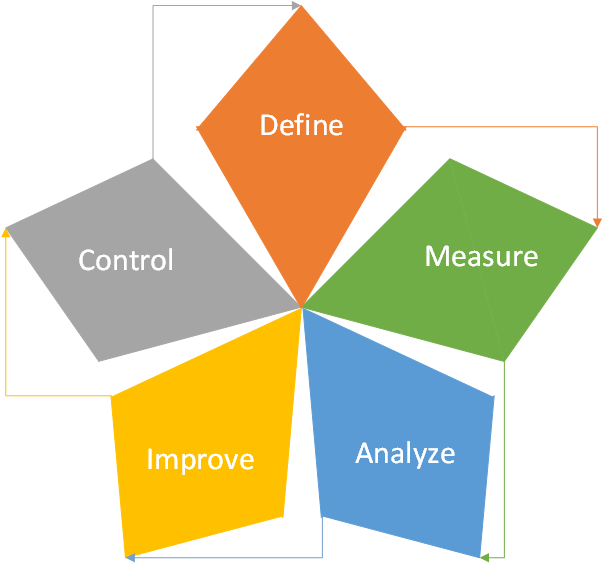
I aim for the for PitchMetrics to have a high easy of use(E) as many people that will use the system will not have much experience with an IT project. The perceived ease of use will be closely tied with how easy it is to use the UI.

Since there is perceived usefulness in like systems and I am aiming for an easy-to-use system. The attitude towards using should be positive. People who do not have experience in using IT projects generally have a better attitude towards using the project when it is easier to use.

The behavioral intention to use is well defined. It is a system that measures different variables of a pitch. Because it has been proven useful, people will use PitchMetrics for its intended purpose.

Actual system use comes when we finish the path. I believe PitchMetrics could be widely used amongst the baseball because of its ease-of-use, perceived usefulness, the attitude towards like-systems, and its behavioral intention to use.

**PitchMetrics Quality Metrics**

**Created by:** Jeremy Maschino **Date:** 2/17/2019

*Figure 7 Quality Metrics*

For Quality Metrics I followed the DMIAC five-step method for Lean Six Sigma. The first step for that is to define the problem. To define the problem, we create a Project Charter to understand the needs of the customers. Step two is to measure the problem. As in, what is the magnitude of it. To do this we set up a data collection plan and collect data and then update the Project Charter. The next step is to analyze the data to find the cause of the problem. Once the problem is identified, we must update the Project Charter again. We then improve the project by implementing and verifying the solution. We first brainstorm to improve, decide on what improvement to use and then create an implementation plan for that idea. Finally, we control by maintaining the solution. A control plan will be set in place for this to be done.

**PitchMetrics Testing Plans**

**Created by:** Jeremy Maschino **Date:** 2/10/2019

**Test Plan Identifier:** PM1

**Introduction:** The test plan is implemented to test the PitchMetrics system. The goal of the test plan is to test the Doppler radar, user-interface, database and network connectivity on how they work together. Due to testing outdoors, internet connectivity may be low, which will lead to slower response times. Also, due to outdoors testing, weather becomes a factor as well.

**References:** PitchMetrics Project Management Plan

|  |  |
| --- | --- |
| Item | Version |
| Doppler radar | 1.0 |
| User-Interface | 1.0 |
| Database/Network Connectivity | 1.0 |

**Test Items:**

*Table 8 Test Items*

**Approach:** To test PitchMetrics, it will be brought to a baseball field to be tested. It will then be set up at a predetermined distance behind home plate. Once that is completed, the system will be turned on and a pitcher will be entered into the system through the user-interface and sent to the database. The pitcher will then enter which pitch is about to be thrown, into the user-interface. The pitch is then recorded, and data will be shown back to the pitcher, and saved to the database as well. Following the completion of a bullpen, the pitcher will then test the system to call back pitches that were thrown to see data on them. Once all of this is complete, the pitcher will save the data to come back to on a later date.

**Item Pass/Fail** Passing or failing will be determined based on if the system can execute **Criteria:** the criteria defined in the approach. As this is the first test, all features need to be working before improvements and retesting.

**Suspension** Suspension of testing would result from project being behind date, **Criteria:** weather, or unable to secure a test subject. To resume testing, the project **& Resumption** must be up-to-date, the weather must be favorable (no rain), and a test-**Requirements:** subject must be re-secured.

**Test Deliverables:** 1) Test Plan 2) Test Cases 3) Defect/Enhancement Logs 4) Test Reports

|  |  |
| --- | --- |
| Hardware | Doppler radar |
| Software | User-Interface, Database Software |
| Network | Internet Connection |
| Tools | Baseball, Pitcher, Catcher, Measuring Tape |

**Test Environment:**

*Table 9 Test Environment*

**Estimate:** See PitchMetrics Cost Estimate

**Schedule:** See PitchMetrics Gantt Chart

**Staffing/Training** 1) Pitcher – Must be able to accurately thrown a ball around the average **Needs:** collegiate pitching velocity (~85 mph). 2) Catcher – Must be able to catch an average collegiate pitcher. 3) Overseer – Must have be willing to learn how to use PitchMetrics Due to the nature that gaining velocity and the ability catch a collegiate level pitch does not happen quickly, both the pitcher and catcher must be collegiate baseball players.

**Responsibilities:** Pitcher – Throw the baseball to the catcher. Catcher – Catch the baseball. Overseer – Enter data in about pitcher.

**Risks:** See Risk Analysis

**Assumptions:** Pitcher will safely warm up before throwing. A test site will be easily secured (no shortage of baseball fields).

**Dependencies:** Doppler radar will pick up high-velocity pitches, network connectivity, working database, and user-interface.

**Approvals:** Jeremy Maschino (2/17/2019)

**PitchMetrics Risk Analysis**

**Created by:** Jeremy Maschino **Date:** 2/18/2019

*Table 10 Risk Analysis*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Negligible | Marginal | Critical | Catastrophic |
| Certain |  |  |  |  |
| Likely |  |  |  |  |
| Possible |  | Rain | Light injury (pitcher or catcher) during testing | Doppler radar isn’t accurate enough to work with high-velocity |
| Unlikely | Security Breach |  | Moderate injury (pitcher or catcher) during testing | Baseball hits PitchMetrics system |
| Rare | Preferred test site is unavailable | PitchMetrics is not deemed useful by consumers | Unable to find pitcher or catcher with listed requirements | Severe injury (pitcher or catcher) during testing |

Risks that are categorized as low priority are security breaches, preferred test site is unavailable, and PitchMetrics is not deemed useful by consumers. They are deemed low priority due to the likelihood of them happening is low and the effect of them is small. Risks that are categorized as moderate risks are rain, moderate injury to pitcher or catcher, and being unable to find a pitcher or catcher. These risks are deemed moderate due to that it is more likely they will happen, and they have more serious consequences than low priority risks. Risks that are categorized as high are light injury to pitcher or catcher, and severe injury to pitcher or catcher. These risks are deemed high due to that they are more likely to happen than the moderate risks and carry a similar to higher effect to the project. The risks that are categorized as extreme are the Doppler radar is not accurate enough to work with high-velocity pitchers and a baseball hits the PitchMetrics system. These are deemed catastrophic as they can derail the whole project.

**PitchMetrics Communication Plan**

**Created by:** Jeremy Maschino **Date:** 2/17/2019

*Table 11 Communication Plan*

|  |  |  |  |
| --- | --- | --- | --- |
| Communication | Frequency | Goal | Owner |
| Project Team |  |  |  |
| Project status report | Weekly | Review project status and discuss potential issues or delays | Project Manager |
| Project review | At milestones | Present project deliverables, gather feedback, and discuss next steps | Project Manager |
| Post-project meeting | At end of project | Assess what worked and what did not work and discuss actionable takeaways | Project Manager |
| Project Sponsor |  |  |  |
| Project status report | Weekly | Review project status and discuss potential issues or delays | Project Manager |
| Project review | At milestones | Present project deliverables, gather feedback, and discuss next steps | Project Manager |
| External Stakeholders |  |  |  |
| End User Feedback | After Testing | Discuss application usability and UI pros/cons | Project Manager |
| Post-project meeting | At end of project | Present final project and assess what worked and what did not work and discuss actionable takeaways | Project Manager |

The Project Team (myself) will provide a weekly status report to discuss project status, issues, or limitations, a project review at milestones with deliverables, and a post-project meeting, in the form of a presentation to discuss what works and what does not work. These will all be delivered by the project manager (myself). The project sponsor will receive a weekly status report from the project manager, as well as a project review upon completion of milestones, also delivered by the project manager. The external stakeholders will receive an end-user feedback to show pros and cons of the UI and a post project meeting, which will include a presentation of the final project and an assessment of the project. Both delivered by the project manager.

**PitchMetrics Stakeholder Management Plan**

**Created by:** Jeremy Maschino **Date:** 2/17/2019

*Table 12 Stakeholder Management Plan*

|  |  |  |  |
| --- | --- | --- | --- |
| Position Title | Staff Name | Power (1-5) | Interest |
| Project Sponsor | Dr. James Jacobs | 5 | 5 |
| Project Manager | Jeremy Maschino | 5 | 5 |
| UIU Baseball Head Coach | Jeremy Ische | 1 | 5 |
| Delano Athletics Baseball Club Sponsor | Ryan Hayes | 1 | 2 |
| DYBSA Representative | Dan Rassmusen | 2 | 4 |
| DYBSA Representative | Darren Knight | 1 | 5 |
| DYBSA Representative | J.T. Bruett | 3 | 3 |
| DYBSA Representative | Greg Maschino | 1 | 5 |

5

5

1

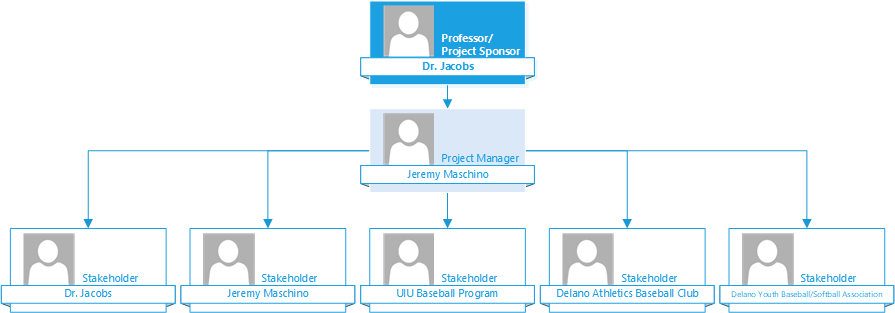
Interest

Power

*Figure 8 Power-Interest Matrix*

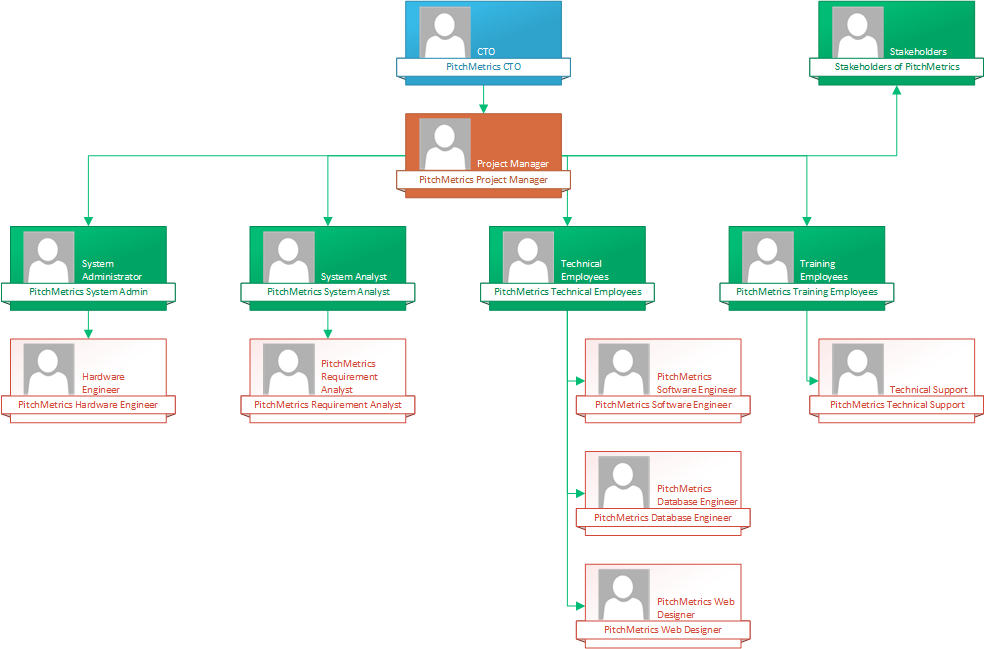
The Stakeholder Management Plan includes key stakeholders identified throughout communication with people who may be interested, their power and influence, and then a power-interest matrix. In the top-half of the chart, it includes our most powerful and influential stakeholders. The top-left most stakeholders on the matrix are the most important people. They have little interest and high power, so keeping those stakeholders engaged and happy is important. The high power, high interest people are the next most important people as they have higher power, but more interest in the project so they won’t have to be managed as much as the high power, high interest stakeholders. Then in the bottom-right corner is the low-power, low-interest stakeholders. They don’t need to be managed as much as the top-right corner as they have lower power over the project. The bottom-left hand corner includes the low-power, low-interest stakeholders. These stakeholders need to be managed the least because they have little power over the project and have little interest in it as well.

**PitchMetrics Organization Chart**

**Created by:** Jeremy Maschino **Date:** 2/24/2019

*Figure 9 PitchMetrics Organizational Chart*

The issue we have with PitchMetrics is that there is no organizational structure like we would see in a company. But, since this is a small project, one of that scale is not needed. I have placed myself as the project manager but will assume all roles that involve creating an IT project. Branching off from me is the stakeholders that I have communicated with about my project. To see a full stakeholder list, go to the Stakeholder Management Plan above. I will include an example organization chart as well to compare what positions I will fill to complete the project.



*Figure 10 Example Organizational Chart*

Above is the example Organizational Chart. When you compare the example organizational chart, to the organizational chart of the PitchMetrics project, you see that I am filling in all positions besides the CTO, and stakeholders.

**PitchMetrics Project Proposal Conclusion**

**Created by:** Jeremy Maschino **Date:** 2/24/2019

**Location:** Upper Iowa University, Fayette, IA 52142 **Project Number:** PM1 Delano, Minnesota 55328

**Context:** Within the sport of baseball, the use of analytics, metrics, and statistics has become increasingly more popular. Major League Baseball, and its member teams, have now employed the use of doppler radars to gather information on how a baseball moves. For this project, the goal is to create a system that implements a database that stores pitchers, their pitches, the date and gathers information from a Doppler Radar that tracks the flight of a ball, that is pitched, and displays relevant information to the user such as velocity (MPH), Spin Rate (RPM), a recreation of the pitch path and the location of the pitch when it enters the strike zone.

**Objective:** With the need for pitching analysis systems, PitchMetrics was born. A cloud stored database would be created to store the different pitchers, with pitches they throw, and the date they are throwing. Along with that, a Doppler Radar would be created by following MIT’s Doppler Radar lab, which describes how to build a Doppler Radar, to gather velocity of the previous pitch, and the spin rate. A graphical user interface is then used to display all the metrics gathered by user input (pitcher, date, pitch type) and the Doppler Radar (velocity, spin rate), along with a recreation of the pitch path and where the pitch crosses the strike zone since Doppler Radars produce 3D data. All information gathered would be sent to the database so that any pitch could be pulled up, regardless of when it happened.\

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Computer System | Operating System | Doppler Radar | Database | Programming Languages |
| HP Elitebook Folio 9740m | **Windows 10** | **MIT’s Doppler Radar Lab** | **MySQL** | **ASP.NET with C#** |

**Tech Specs:**

*Table 13 Tech Specs*

**Cost Estimate:** See Cost Estimate (p. 8)

**Timeline:** See Network Analysis (p. 11), Gantt Chart (p. 12), Critical Path Analysis (p. 13) and PERT Analysis (p. 14)

**Conclusion:** Given the need for a low-cost pitch analysis system, the cost to create one ($400 per Cost Estimate), and the quick timeline (8 weeks per Gantt Chart) to create a system, the PitchMetrics pitch analysis system becomes a viable project to pursue.

**Approval: x\_\_\_\_\_\_\_\_\_\_\_\_\_**\_ **(Project Sponsor) Date:\_\_\_\_\_\_\_\_\_**

**x** Jeremy Maschino **(Project Manager) Date:** 2/24/2019