Coaching Action Research Excellence

Web Browser Programming for Best Practices in Coaching Data Collection.

by: Mark D. Mann, Ph.D.

2nd Edition

Introduction

Your Best Data Miner?

This book has been written to provide a Foundational Text for Coaching Action Research with detailed methods of collecting action research data using current best practices in web based data acquisition. The use of the ELM programming language is described with tutorials on collecting ones own data to evaluate tactical and technical coaching goals executed by and with ones athletes. One of the essential questions of action research is to determine whether or not the core values the coach has professed are actually being carried out in their professional life.

You are Your Best Data Miner

As a Coach of Junior Club and Collegiate Women's Volleyball for the past 25 years, I have asked myself many questions regarding the most important skills that can be taught and learned in the game of volleyball, and how the learning of such skills translates to winning points. It is true that there are sophisticated tools such as DataVolley that help Coaches answer those questions. But, such tools usually require funding and staffing that many programs at the Jr. High School and High School Levels simply do not have at their disposal. When I asked myself the question, "What skill in the sport of volleyball is the most directly correlated to winning points?" I found an interesting line of research related to the question whose answer was the particular skill of Serve Receive Passing. A 2010 Study on Skill Importance in Women's Volleyball by Fellingham, et.al. found

that Serve Receive Passing was indeed the skill that was most determinant for points scoring.

Importance Scores for the volleyball logistic regression analysis	
Skill	**Importance Score**
Pass	8.49
Float Serve	8.22
Jump Serve	7.61
Set Distance	6.36
Digs	5.26

Fellingham Study demonstrating the importance of Passing in Volleyball

A tactical rationale for why the pass ranked highest in order of importance for point scoring was sensible. That is because, the more options a team had with the ball in the setters hands, the more hitters would be a threat to the opponent, and the match ups for attacking could more likely be one on one, or better yet, one on zero. This makes for an optimal scoring opportunity for the hitter, and thus, the team.

Beyond the work of Fellingham, et.al., I was intrigued by the results of the Fellingham study and wanted to conduct research for myself. A most important question to me was not just to know the value of skill of passing to point scoring but particularly, the value of serve receive passing accuracy as a determinant of point outcome. I was interested in this from the perspective of NCAA Division 1 Volleyball, but also from the perspective of Internationally competitive sitting volleyball teams that competed in the Paralympic Games every four years. In this book, I will share the results of these research questions, and also share the means by which the data was collected. For the purposes of the data collection process, I created a rather

simple app that could be used to track serve receive passing scores (on a scale of 0 to 3) for winning and losing points in NCAA Division 1 Competition, and in Paralympic Volleyball Competition. Serving as a volunteer assistant coach at a division 1 NCAA school from 2008-2016, I have found the opportunity to take the very basic premise that good serve receive passing significantly increased point scoring probability, and put it to the test. I was also blessed with the opportunity to be given a large sample of serve receive passing data from the USA Paralympic Women's Sitting Volleyball Team in an international competition so that I could also analyze the relationship between serve receive passing and point scoring for this specific volleyball playing population.

2013: Action Research Project NCAA D1 Volleyball Quantitative Observations

In 2013, over the course of four matches, data was collected for Division I Volleyball Serve Receive Passing in live conference play. using a homemade app called PasStat The results of the study demonstrated that there was a strong distinction identified between passing scores when points were won (2.75), and passing scores when points were lost (2.09). Passing scores were statistically significantly higher on points won (2.75) vs. points lost (2.09) where T Score, P-Value and Cohen's D respectively were: T=11.72, the p-value was < .0001, & Cohen's D measuring effect size was =1.29.

Mann, 2013- Action Research Project		
NCAA Division I	**Winning Point Passing**	**Losing Point Passing**
Pass Attempts	**188**	111
Pass Points	**517**	232

Mann, 2013- Action Research Project		
NCAA Division I	Winning Point Passing	Losing Point Passing
Pass Average (Mean)	2.75	2.09
SD	0.49	0.51

Passing Scores Recorded on PasStat for Winning and Losing Points

Volleyball Coaches who put into practice an emphasis on serve receive passing accuracy, because of findings such as these that place a high value on serve receive passing accuracy, will benefit from conducting drills and game scenarios in practice that improve these skills.

2014: Action Research Project USA Sitting Volleyball Paralympians- Quantitative Observations

In 2014, over the course of six matches, data were collected for Four Paralympic Women's Sitting Volleyball Teams' Serve Receive Passing Accuracy on winning and losing points during six matches. The USA, Russian, Chinese, and Brazilian Teams Serve Receive Passing Data was used in this study. Using DataVolley to collect the passing data, the results of the study demonstrated that there was a strong distinction identified between serve receive passing scores when points were won (2.63), and serve receive passing scores when points were lost (1.68) where T Score, P-Value and Cohen's D respectively were: T=6.79, the p-value was < .00001, & Cohen's D measuring effect size was =2.77. Measuring effect size in this study was important as total team serve receive passing means were used on a

per match basis over the course of the four team round robin tourney.

Mann, 2014- Action Research Project		
Paralympians	**Winning Point Passing**	**Losing Point Passing**
Pass Average (Mean)	**2.63**	1.68
SD	**0.38**	0.30

Passing Scores Recorded with DataVolley for Winning and Losing Points

A conclusion that could be drawn regarding the results of the serve receive passing data for winning and losing points is that improving serve receive passing accuracy increases point scoring probability for both NCAA Division 1 and Paralympic Sitting Volleyball Teams. Volleyball Coaches who put into practice an emphasis on serve receive passing accuracy, in response to findings such as these, will benefit from conducting drills and devising game scenarios in practice that improve these skills.

Qualitative Observations

Of course, engaging athletes in meaningful practice sessions where quality of performance matters to them is vital to success. The coaches of the USA Paralympic Team and the NCAA Division 1 team that were studied were quite successful when it came to engaging their athletes. Observations of the NCAA team and USA Paralympic Team for whom the 2013 and 2014 Action Research Projects were conducted, and how the coaches of those teams taught serve receive passing to their athletes could be summarized as follows:

1. Serve Receive Passing was not taught as a “drill” in isolation from other skills of the game. Instead, serve receive was most often the part of a serve/pass/set/hit situation.

2. Serve Receive was conducted in high volume/high repetition game like drills such as 3 vs. 3, 2 vs. 2 half court, and fast knock out kinds of situations.

3. Questioning was used to engage critical thinking and strategy making for each type of serve receive pass situation that they were dealing with.

Self Motivated Athletes

In this chapter, we will diverge slightly from the emphasis on data collection to discuss the underlying purpose of data collection in coaching. To create self-motivated athletes, coaches must cultivate the characteristics of competence, autonomy, and cohesiveness. These aims are best achieved using the insights offered by the Teaching Games for Understanding (TGfU) approach to coaching. The deployment of TGfU from the elite level down to the recreational level will be successful if coaches have a clear plan to implement TGfU techniques and methodologies. Strategies such as employing a large percentage of time to small sided games where decision making and tactically challenging situations are prevalent is critical for success. Follow up with the use of questioning techniques are also important to help the athlete feel a sense of competence on the court in his or her sport. From the basis of tactical competence, the athlete will have greater opportunity for autonomy on the court or in the field of play. Coaches, nevertheless, must have training in rethinking the benefits that athlete autonomy can bring to their own chasing practice. Another aspect of cultivating self-motivated athletes involves the role of data collection that the coach can gather as part of their coaching practice. In this book, we will highlight the use of web-based technologies to facilitate and make easier the process of coach data collection. Coaches will be shown how easy it is to collect real time data on TGfU theory based games and game-like drills. The apps developed and presented here are able to run offline or online in a web browser on a laptop computer, tablet, or smartphone.

For a coach, I believe that the most important action research question is that of player motivation. Using the theory of self-determination as a foundation for understanding healthy motivation, and the motivational status of our athletes, we are able to pose important questions about where along the continuum of motivation (from least self determined to highly self determined) our athletes may be at any given time of the season. A website called http://www.wellnessoft.org/ has been created to demonstrate the technology used to collect passing data through a web app called PasStat. The literature on the subject of Sport Motivation discusses that the environment that the coach creates can play a key role in the contextual motivation of the athlete. The coach that creates an environment where there is: 1) a sense of autonomy, 2) a sense of increased competence, and 3) a sense of cohesiveness is most likely to see improvement for his or her athletes.

Autonomy

The Coach that asks meaningful questions to their athletes and enriches critical thinking and decision making capacities among their athletes will be cultivating player autonomy. Autonomy is visible when players are making their own quality tactical and technical decisions on the court/ field of play, and off the court as well. The autonomous athlete feels like their voice is heard. TeamPolls and TeamGoals were apps developed for the purpose of increasing athlete autonomy within the context of developing leadership and athlete autonomy in Team Sports.

Competence

The coach who wants to emphasize team competence will be able to create drills, with input from their athletes, that are characterized by "lots of touches" and "game like" conditions. Instead of just going through the motions of "drills," athletes will gain confidence because they are playing in game like conditions and are becoming more familiar with the myriad of options and situations that can occur in games in their sport.

Cohesiveness

With regard to performance feedback using live sport data collection, the app that we have built for the purpose of assisting with team cohesiveness is called PasStat. When one thinks of PasStat, one should think about the concept of Specialized Player Focus. The process of Tracking a Statistic that isn't as readily kept in volleyball- First ball contact- Volleyball Passing Statistics- the level of team cohesiveness is likely to increase. The reason: keeping such a stat elevates the importance of first ball "ball control" for the team. PasStat keeps track of as many passers at a time as one would want to keep track of, and the app updates Individual and Team Passing Averages in real time. One is also able to save the datafile for later review. PasStat increases team COHESIVENESS by highlighting appreciation for the ball control specialists on the team. What Stat Could you track to help your team be more cohesive? PasStat has been successfully used in the sport of volleyball. Because most sport statistics record “on/with the ball” activities, in order to motivate athletes whose usual position is away or off the ball, recording statistics which keep track of the “unsung heroes” or “unheralded

skills “ of a sport, can be highly motivational for those individuals, and for the team as a whole.

Why Use Technology in Coaching?:

A Practical Case for Winning.

Beyond the noble goal of increasing motivation, improving the organization and creativity of practices, technology embedded in tools such as PasStat, TeamPolls, TeamGoals, and TeamChat, also serve a practical function. In my opinion, technology that can be used to help a coach look at questions that are on the cutting edge of tactical and technical decision making helps the coach know how much time and energy he or she should dedicate to particular phases of the game. Live data collection using PasStat gave me several real time advantages in terms of observing and reacting to first ball tactical trends (by our team, or the opponent) in the game of volleyball. Knowing who the other team was serving to in certain rotations, and being able to know specifically how "our" team was performing in each situation, was a key to helping our team to gain an edge on the first ball in competitions, and thus, have a better chance to win points, win matches, and win championships.

NCAA Female Sport Champions in Mann's Action Research Studies		
University	**Year**	**Championship**
ORU Basketball	2007	Summit League Champion
ORU Basketball	2008	Summit League Champion
TU Volleyball	2010	C-USA Champion
TU Volleyball	2011	C-USA Champion
TU Volleyball	2012	C-USA Champion

NCAA Female Sport Champions in Mann's Action Research Studies		
University	**Year**	**Championship**
USA Sitting Volleyball	2016	Paralympic Gold Medal

NCAA Conference Championships & Paralympic Medals Won

2007-2008: Use of Forecasting Models for Goal Setting

Motivated Players need to be pushed towards high goals and high standards. At ORU from 2006-2008, where I served as a volunteer assistant coach, and chaplain for the team, I was able to share with key players certain individual accomplishments that they could achieve if they were to be consistent and methodical in their day to day preparation and approach to the game. The Goal Setting Spreadsheet was an effective tool that helped one younger burgeoning leader of the team in particular to see that she could in fact, be one of the best players to ever play at that particular university. As a result, in part, of achievable goal setting, that stretched her to reach for the highest performance, her team won two conference championships, and she set the school all time record for rebounds and also scored over 2000 points in her four years at the university. The stretch goals for a key player can be spelled out with spreadsheet calculations that make once seemingly lofty goals appear realistically achievable. Therein lies the power of forecasting models applied to coaching student athletes.

2009-2010: Coach Feedback During Time-Outs

The years 2009-2010 were years where this researcher became very aware of the importance of Coach Feedback as it related to athlete performance. The purpose of this line of research was to understand the relationship of coach feedback during time-outs to the performance of 16-18 year old volleyball players in competitive match play situations. The systematic observation of coach feedback during 89 time-outs was recorded using the Coach Time-Out Observation Instrument (CTOOI). Out of the 879 feedback statements that were made during the 89 time-outs, the CTOOI categorized coach feedback for technical feedback (with an internal or an external focus), tactical feedback (referring to our team or the opponent), and psychological feedback (as either encouraging or discouraging remarks). Data from the Game Performance Assessment Instrument (GPAI) were collected for the "quarterback of the volleyball team:" the setter. Data were collected to evaluate setter performance for the four rallies before the time-out and the four rallies immediately after the time-out. The GPAI measured setter positioning, decision making, and skill execution. The multiple regression analysis did not show any feedback strategy to be significant for the entire group of setters in terms of performance improvement. However, as the literature on coach feedback had suggested, when the setters were divided into groups of higher and lower skilled setters, significance was found for certain coach feedback types in each group of setters. For higher skilled setters, significant improvement in setter performance ($p = .03$) came from feedback that was tactically oriented towards the opponent in combination with technical internal feedback. For lower

skilled setters, setter decision-making was improved significantly (p= .05) by time-out feedback characterized by psychologically encouraging over and above discouraging remarks that were made during the time-out.

2010-2014: PasStat History

The years 2010-2012 were years where as a sport coaching researcher, I was truly blessed to be around one of the finest athletes in NCAA Division 1 women's volleyball. The dominating outside hitter at the midwestern university where I worked made the coaches job rather simple. Pass the volleyball well, and we give our team a great chance to win points, win matches, and win Championships.

The history of the program, PasStat, is one which developed during those years of a slow, methodical, and steady growth from an offline standalone spreadsheet (with about 25 sheets) to an online web application with helpful features, and lots of room for growth. PasStat began as a a server side web app that used SQL to store and query form data on a remote server, and return results to the end user (coach). As such, PasStat has been an official "APP" at the Firefox Marketplace since Mid-2012. PasStat was successfully used to emphasize benchmarks for the team to achieve regarding serve receive passing efficiency. The passers on the team knew, with the kind of offensive firepower that we had, if we passed at an average of 2.4 (on a three point scale), that our chances of winning were excellent. We also used PasStat to set and measure goals of limiting our opponents to an overall serve receive passing score of a 2.0 or below in their service reception opportunities. For the 2011

undefeated 20-0 record in conference, in that historic season, every single conference match was characterized by these high standards of quality performance of our passers and servers being met.

Sometime in 2012, after being inspired with the idea to to post PasStat to the Firefox marketplace as a SQL based app, the there were problems with it's deployment to high school and jr. high volleyball coaches. It became evident that a different kind of approach was needed. In a nutshell, there were too many coaches out there who did not have a reliable internet connection from their gymnasiums to be able to use PasStat with confidence. Thus, my own technology research led me to the idea of building PasStat as an offline app that did not need an internet connection to function properly. What this required was for all data to be stored on the local Computer, Phone, or Tablet, and not require online connectivity. Thankfully, HTML5, Local Storage, and the Elm Programming language were coming of age. This made the creation of Offline Web Apps a real possibility for "novice" programmers, and coach/educators like myself. PasStat works on or offline, and saves all data on local storage so that the data is persistent from one viewing to the next, and the user can close the web browser and reopen it and find that the data from previous sessions is still there, is reloadable, and savable. In the chapter entitled Future Directions 2, more detail will be given on how PasStat was built with Elm.

2014-2016: Inspiration and Disability.

Because coaches are vital to the success of an athletes' motivational level in the sport they coach, the significance of motivation in a competitive sports environment should be explored in order for coaches to be able to coach more effectively in their sport. When certain aspects of motivation are utilized, not only is an athlete able to have a higher level of motivation, the coach is more able to direct their athletes to incorporate motivational strategies that will assist them over the long term in their sport careers. By learning the motivation level in the context of Paralympic Athletes, coaches will be able to appropriate transforming instructional methods and apply coaching strategies that can increase intrinsic motivation, and improve athletic performance. Through statistical analysis, we were able to collect data and report to the coaching staff the motivational profile of the Paralympic Athletes from the USA Paralympic Sitting Volleyball Team and compare Paralympian scores to the SMS-II motivational scores of the typical NCAA athlete as conducted in a study by Bean [1] in 2014. The USA Paralympic Sitting Volleyball Team, prior to the 2016 Summer Paralympic Games in Brazil had lost in the gold medal match to China in the three previous Paralympics (2004, 2008, and 2012). The research that we conducted in 2015 was used to help gauge the USA team's motivational level prior to entering the 2016 Paralympic Competition. By all indications, the USA Team in 2015 was highly motivated, and in fact, they did win the Gold Medal Match at the Paralympic Games in Brazil in 2016.

.

Paralympic Coaching Challenges

Coaching Elite Sport has it's array of challenges. Coaching Elite "Paralympic" sport has additional layers of complexity that add to the level of expertise needed to do the job well. Martin, et.al. [2] identified such aspects as: understanding the nuances of each of their athletes' unique disabilities, dealing with accessibility issues, and navigating additional travel logistics. In the area of applied sport psychology to Paralympic Coaching and Athletes, researchers [3] have shown elite athletes with a physical disability are more intrinsically motivated when they feel in control of their sport experience, feel competent in their sport, and are motivated to achieve new goals. The same is true of intrinsic motivation to experience stimulation, which is defined as doing the activity for the positive physical and emotional experiences that occur while doing the activity. Canadian national wheelchair rugby players [4] have expressed increased feelings of self-confidence and empowerment as a result of belonging to a sport community where they were no longer isolated because of their disability. Moreover, their sport experience helped them grow as individuals on and off the court. Such research identifies the importance of creating an autonomy-supportive sport environment when coaching elite athletes with a disability. Because there are few programs available for coaches of elite athletes with a disability, studies that have been qualitative observations of coaches who model such coaching behaviors are of tremendous value. In 2014, the coach of the USA Paralympic Sitting Volleyball Team was observed [5] modeling coaching behaviors that enhanced elite disability sport participants level of intrinsic motivation. The ideal coaching strategies employed by the coaching

staff, provided best practices that were used to train student coaches who work with Paralympic and Able Bodied athletes. Observed behaviors included, but were not limited to, 1) providing how to provide athletes with choices (e.g., which drills to use during practice), 2) how to give athletes opportunities to take initiative (e.g., designing a practice on their own), and 3) how to give constructive, task-oriented feedback in order to foster intrinsic motivation. The type of coaching described herein, can be characterized as that of a transformational leader [6]. Transformational leaders espouse ideals, act as role models, and show care and concern for each subordinate. They inspire their followers by formulating a vision and setting challenging goals, and stimulate them intellectually to think about old problems in innovative ways. Charbonneau, et.al. [7] successfully demonstrated that the very nature of the different components of transformational leadership will be particularly suited to enhancing intrinsic motivation, and improving athletic performance. “With its emphasis on stirring individuals to think for themselves, and to approach old problems in new ways, the intellectual-stimulation component of transformational leadership increases knowledge, learning, and understanding. Similarly, charisma raises individuals’ and groups’ expectations about what they can achieve and is likely to increase the accomplishment and task orientation component of intrinsic motivation.”

Comparing athletes with and without disability is a common practice within sport psychology research for individuals with a disability. [7] Such a comparison is often justified by the idea that athletes with a disability differ from athletes without disability in that they “have had

a major life trauma, loss, or chronic situation to which they have had to adjust." Critics would argue that the challenging life event(s) that the Paralympic Athlete has faced would cause the Paralympic athlete to score higher than the able bodied athlete on Sport Motivational tests. What is not taken into account, across the board, when studies like this have been conducted, is the fact that there is a lack of trained [3] coaches of adapted sport teams. This lack of training can bring Sport Motivation Scores down to the point of predicting athlete burnout and disinterest, and poor scores in Sport Motivation. However, when well trained coaches are working in the field of Paralympic Sport, the same (if not, more significant) benefit of implementing a transformational leadership style to able bodied athletes should occur. As research from the Banack, et.al. study has indicated [3], the same psychological processes apply to athletes with a physical disability, as they do to able bodied athletes. The key, then, it would seem, is in the coaching.

Data was collected with the USA Paralympic Sitting Volleyball Team in the Summer of 2015. All members of the team (23) were invited to voluntarily participate in the study. After signed consent forms were received from the participants, the surveys were given and the results were collected prior to a practice with paper and pencil results collected and then tabulated. The data from the NCAA Sports Teams was collected in a previous study (Bean, 2014) and used for the purpose of comparison in this study.

An independent-samples t-test was conducted to compare motivation scores for USA Paralympic Athletes and NCAA Division 1 Athletes on Six Key Types of Motivation as defined by the Self-Determination Theory of Motivation. The data was collected using

the 7-point likert scale of the Sport Motivation Scale-II (SMS-II).

External Motivation Scores for the Sport Motivation Scale- II				
	N	Mean	SD	SE
Paralympians	23	1.75	1.01	0.25
NCAA Athletes	111	3.20	1.46	0.13
	t	df	p	
T-Test & P Value	t= 5.72	df= 132	p=0.0001	where p<.05

Figure 2

The athletes motivation types are categorized as either having a positive or negative impact on long term motivation. In a positive light, higher scores on the Intrinsic, Integrated, and identified Motivation Scales are seen as positively effecting athlete motivation. In a negative light, Higher scores on the Introjected, External, and Amotivation types are seen as negatively effecting athlete motivation.

Amotivation Scores for the Sport Motivation Scale II				
	N	Mean	SD	SE
Paralympians	23	1.39	0.90	0.25
NCAA Athletes	112	2.30	1.39	0.13
	t	df	p	
T-Test & P Value	t= 3.98	df= 133	p=0.0001	where p<.05

Figure 3

2.2 The Sport Motivation Scale II

Self Determination Theory has a "multi-dimensional measurement tool used to assess sport motivation." This evaluation instrument is called the Sports Motivation Scale (SMS). The SMS was created in 1995, and was redeveloped in 2013. The revised version of the SMS is the Sports Motivation Scale II (SMS-II). This SMS-II has been tested for validity and reliability, and has been deployed in professional and amateur sport coaching research. With it's questions designed in line with the key aspects of Self Determination theory, it is an appropriate theoretical framework to understand and promote ideal motivation in sport [8]. The SMS-II was designed to be more efficient and asks fewer questions than the original SMS. The SMS has 28 questions while the SMS-II has 18 questions. The participant answers questions on a likert scale (1 to 7) and can complete the survey in under 15 minutes. The motivation of the participant is evaluated, based on the fact that the respondents' questions are coded to correspond with a specific form of motivation regulation. We have found the use of this scale to be non intrusive and taken by participants quite willingly. Thus, the designers of the instrument have met the goal of making the SMS-II easier to administer than the SMS-1, which took a bit longer for participants to complete.

RESULTS

There was a significant difference on the scores for the USA Paralympic Athletes (M=6.03, SD=1.24) and the NCAA Athletes

(M=5.3, SD=1.39) on the Intrinsic Motivation Scores from the Sport Motivation Scale- II; t (132)=2.47, p = 0.015. (see Fig.1)

Intrinsic Motivation Scores for the Sport Motivation Scale II				
	N	Mean	SD	SE
Paralympians	23	6.03	1.24	0.25
NCAA Athletes	111	5.31	1.39	0.13
	t	df	p	
T-Test & P Value	t= 2.47	df= 132	p=0.015	where p<.05

Figure 1

Also significant were the differences in scores on the External Motivation, and Amotivation Scales. Because these types of motivation are not productive or conducive to continued participation, productivity, and/or improvement, the scores on these measures are better when lower. The results of the External Motivation Scale for the USA Paralympic Athletes (M=1.75, SD=1.01) and the NCAA Athletes (M=3.20, SD=1.46) on the External Motivation Scores from the Sport Motivation Scale- II were; t (131)=-5.72, p = 0.0000001. (see Fig.2)

The results for the Amotivation scale for the USA Paralympic Athletes (M=1.39, SD=0.90) and the NCAA Athletes (M=2.3, SD=1.39) on the External Motivation Scores from the Sport Motivation Scale- II were; t (134)=-3.98, p = 0.0001. (see Fig.3) were also statistically significant.

High Motivation Levels of Paralympic Athletes

The results of the study suggest that, given a coach who practices transformational leadership techniques [5], Paralympic Athletes will demonstrate a high motivation level on scores on the Sport Motivation Scale-II. These scores, based upon their comparison to the study by Bean [1] of NCAA Athletes, was at a significantly higher motivational level than traditional NCAA Athletes on the Intrinsic Motivation Score, and significantly lower on the negative factors of External Motivation and Amotivation scores of the SMS-II. The conclusion that can be drawn, is therefore, that the Paralympians given quality transformational coaching, will be relatively autonomous in their pursuit of their athletic goals. Specifically, our results suggest that USA Paralympians were highly motivated to perform and participate in competitive athletics at a motivational level that is significant in terms of both positive motivation (Intrinsic Motivation) and that scores are significantly low in terms of negative motivation (External Motivation and Amotivation) when contrasted with the motivational levels of traditional NCAA Athletes. Further study will be conducted to verify these preliminary findings. It would also be interesting to determine if transformational coaches can also predict the level of motivation in their athletes during different stages of the competitive season, and in the case of the USA Women's Paralympic Sitting Volleyball Team, as they enter a new phase of training for the 2020 Paralympic Games in Japan.

Future Directions 1:
Simulations.

Having begun the foray into creating simulation software for the individual sport of downhill skiing, it can be stated that progress in this area of software development will take time, but, is well worth the effort. The Downhill Skiier app developed using the elm programming language is one that shows tremendous promise. First, because Elm is a relatively easy programming language to learn, Elm code can be modified and shaped to ones intended purpose. Although creating 2 dimensional (2D) software with elm was rather pedestrian in terms of an individual sport such as skiing, creating simulation software for team sports will involve a deeper level of programming skill. This development will involve a process of trial and error, product testing, and software modification, all of which are well suited for the Elm programming language. The hope for my work in the sport of Volleyball, will be to begin with rotation animations, and then go into defensive and offensive simulations in 2D. Since Elm is moving into the 3D space rather quickly, it will also be possible to create 3D simulations at a later date.

Past Research in using Elm to modify 2D game software was successfully carried out by converting a game called Elm Maze into Elm Downhill Skiier (see code examples in Appendix #1)

Individual and Team Sport Software Simulation Time Line		
Software	**Year**	**Status**
Downhill Skiier	2016	Completed

Individual and Team Sport Software Simulation Time Line		
Software	**Year**	**Status**
2D Volleyball Rotation Simulations	2017	in process
2D Volleyball Defensive and Offensive Simulations	2017-2018	Future Project
3D Volleyball Defensive and Offensive Simulations	2018-2019	Future Project to follow 2D simulation software development

Dr. Mann's Individual and Team Sport Simulation Software Timeline

The development of Individual Sport Simulation Software: Downhill Skiier

The "Digital Native" is mentioned in Kiili's work [1], as the 21st century student athlete/learner who has been raised in a different environment to that of the pre-21st century learner. The pervasive technology that surrounds most of their lives, for example: smart phones (with all of their features), laptops, tablets, and/or games consoles, has created a new generation of learners that reason differently than their pre-21st century counterparts (the Digital Immigrants) and therefore need to be educated differently. As learners and consumers of information, Digital Natives are more naturally inclined to learn from web blogs and social media, rather than by reading books. In this context, then, computer games can be seen as being a more natural means of acquiring and processing information to them.

Literature on successful game development has indicated that repetition (where students take time to master a level of difficulty until they attain that level) comes without thinking about it, and that most game players will attempt the game

until they attain the next level of the game. As Gee [2] has contended, this process is fundamental to learning, as gamers will reflect on what they have discovered in the process of advancing a level in a game. Having reflected, the player will develop a hypothesis on how the “game world" operates. With this hypothesis in mind, the player “re-probes" the game world in order to test whether the hypothesis is correct or not. In the event of the hypothesis being incorrect (i.e. the player fails to make progress or complete the current level), the player will then start from the beginning and re-probe, reflect and then re-develop/re-test their hypothesis. Gee also argues that the “probe, reflect, hypothesize, re-probe" process will also lead the player/learner to engage in critical thinking.

Another main argument for the educational use of computer or smartphone games is that they generate high levels of engagement and therefore lead to interested and motivated learners. Motivational theory has long considered the keys to successful learning involve engagement, and that motivated learners have greater involvement in the process of education. In this context, student athletes as game players are willing to spend hours of their time in order to master a certain game feature, a level within the game, or the game itself. A successful computer game involves a mixture of interactivity, motivation to complete the game, and a challenge. The level of challenge should be evenly balanced so that the game is not too hard to complete, nor too easy. [1]. Game mechanics are also important to participant motivation. The aspects of game mechanics include the player being able to identify the goals/

rewards/penalties of the game, the level of challenge given to the player and the various mechanisms employed (such as: racing against the clock) that make the game enjoyable. A person who is intrinsically motivated will chose to engage in an activity because they find it inherently interesting and seek to participate due to gaining enjoyment from the activity. On the other hand, extrinsic motivation is considered external to the individual. That is, the individual will participate because they feel obliged or required to do so. What is vital is the integration of "flow" and really getting into a game and gaining benefit from the game that, "teaches everything it has to offer before the player stops playing" [3].

Flow Theory

"Flow Theory" is a set of conditions, which if experienced at the same time, can cause the learner to enter into a flow state of mind. When learners are in a Flow state, they will be intensely engaged in their current activity, to the exclusion of everything else. This proposed level of engagement is very appealing within the education sector and is regarded by Gee [2] as one of the most persuasive reasons for the use of games within the education system. Two important characteristics of the flow state are total immersion and time loss. Total immersion means that the learner should be so engaged in the current activity that the outside world, and fear of failure, fades into the background. Time loss refers to the fact that the activity would be so engaging that the learner loses track of time, and in that context, time has flown by. The ideal balance is represented by a Flow Channel whose rules are, "as a

learner's skill increases, the level of challenge will also need to be increased" so that the learner will remain in the flow of the activity.[4]

The Programming Paradigm

As an attempt to engage 21st century student athletes with a computer game and learning environment that would be engaging and cultivate the flow experience highlighted in the previous section of this chapter, I set out to create an experimental website called http://chirp.life/. The chirp.life website was created for 21 century student athletes to play simulation games and be engaged by the games and be motivated to play them until mastery was achieved. The game would need to be playable via the web on a computer.

The programming language that was chosen to develop the game is a language called ELM. [5] The language can be found at http://www.elm-lang.org/. Originally, the researcher had thought of using a purely JavaScript implementation, but found that limited programming experience made it difficult to go into the source code and modify the games in JavaScript in such a way that was needed or possible. Without the ability to customize the game, and the students experience of the game that they were playing, to fit the flow criteria benchmarks of engagement and enjoyment of the game, the search had to continue until ELM was found. One of the great things about ELM are the tutorials that are provided to help the novice programmer get started on building games locally on their

computers, and then be able view the source code, and edit and modify that code as they see fit. Also, because ELM in an open source programming language, most ELM programmers will share their code on repositories such as GitHub where one can download all of the source code of a particular game, study it, and learn a great deal.

Prior to fully knowing the value of the ELM games being studied in terms of engaging student athletes, there were four ELM games that were ported by this researcher to the chirp.life website. ELM-Maze, ELM- Asteroids, ELM-Snake, and ELM-Tetris Game (called Flatris). thesegames were popular with visitors and students who had gone to the site, but not necessarily sport related at the onset. The ELM-Maze game was the one that was targeted to be transformed into an individual sport simulation game. The value of the upgrade from ELM-Maze into a new game called ELM-Downhill Skier from a student enjoyment and engagement perspective was the independent variable of a study conducted for the purposes of this chapter. The researcher surveyed a group of 48 student athletes using the eGameFlow survey after Playing the ELM-Maze game, and then again, days later, after the improvements to the game were made, and ELM-Downhill Skier was put in ELM-Maze's place. A T-Test was calculated to record the possibility of a significant difference on the EGameFlow scores from before to after the improvements to the game were implemented. The modified game consists of the game player being able to adeptly manipulate the skier down the five distinct mountain runs in as short a period of time as possible.

All of the runs on the downhill course were modified and were designed with the idea of progressing in difficulty from one stage of the game to another with intermittent breaks between the most difficult sections.

In terms of selection of the survey to be used for quantitative data analysis, EGameFlow [6] was selected because of it's goodness of fit with the measures of Flow and the goals of the improvement in the ELM maze game and it's transformation into ELM-downhill skier. The EGameFlow scale is a scale that measures the experience offered by E-learning games, and helps the game designer to understand the strengths and weaknesses of the game from the learner's point of view. EGameFlow consists of a number of questions in eight areas, presented in Likert-type scales. To answer the questions or statements, the respondents have to express their degree of agreement or disagreement on a scale of 1 to 7 where the higher number indicates agreement and the lower indicates disagreement. The eight areas measured by the EGameFlow Survey Instrument are: Concentration, Goal Clarity, Feedback, Challenge, Autonomy, Immersion, Social Interaction, and Knowledge Improvement.

Results of Elm-Maze to Downhill Skiier Modifications

The results of the study showed a significant improvement in student athlete scores on the EGameFlow Survey. These improvements were based on student responses after playing

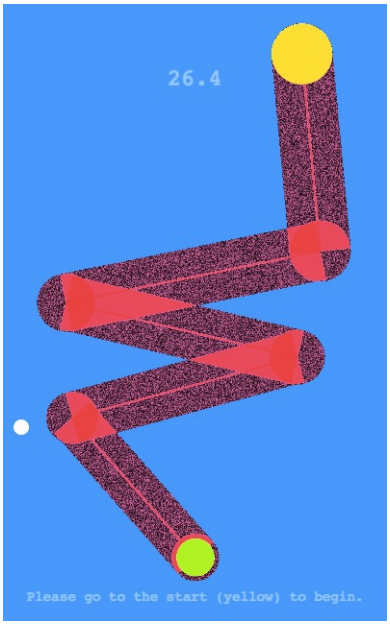
the ELM-Maze Game and then, thirty days later, the ELM-Downhill Skier Game. The mean scores from the eight categories of EGameFlow showed seven of the eight categories gaining marked improvement. (See Fig. 1.) Autonomy was the only category where the mean score was lower on ELM-Downhill Skier than ELM- Maze. It is possible that creating more courses for ELM-Downhill Skier and options for the game player in the ELM-Downhill Skier game will improve scores in the Autonomy category of the EGameFlow survey.

(See Appendix for example code from the ELM-Downhill Skier game, and how new courses could be readily added to the code base.)

Means of EGameFlow Survey Results by Category		
Category	ELM- Maze (before)	ELM Downhill Skier (after)
Concentration	4.1	5.2
Goal Clarity	3.3	3.5
Feedback	4.5	5.2
Challenge	3.2	5.4
Autonomy	3.5	3.2
Immersion	3.6	5.4
Social Interaction	2.9	4.1
Knowledge Improvement	2.5	4.0

Regarding overall statistical significance of the EGameFlow Survey results, the p value (where a < 0.05.) for the two independent samples t-test was 0.018, where T=-2.689. Thus,

the before and after scores on the EGameFlow Survey did show a significant result.



Downhill Skier Screenshot

As the results of the study indicate, programming in ELM was a fruitful enterprise that was well received by the students athletes playing the ELM-Downhill Skier Game. The Personal Best Leaderboard indicated significant engagement that improved interaction between participants in the online game. As a faculty member at a state university, I have found that students can relate to the professor/ coach / researcher on a more engaging level as a result of the work that has been done in game development to engage them on "their turf" in the world of 21st century technology. As this study has pointed out, the possibility that creating games that are engaging for learners has a strong effect on learner concentration (4.1 to 5.2), a sense of challenge (3.2 to 5.4), and the feeling of immersion (3.6 to 5.4) in the task at hand. With an overall alpha of 0.018 it is likely that the efforts made to engage and immerse the students more in the ELM-Downhill Skier game did not happen by chance. Thus, it is a productive enterprise to engage ourselves in ways to meet our students where they are, here, in the second decade of the 21st century. Overall, such efforts can make a significant

difference in the value the learners will attribute to the games themselves, and quite possibly to their learning experience as a whole. One aside worth mentioning is that other platforms, that don't require compilation, and are pure javascript examples of game development, are listed under the howto tab at the Chirp.Life homepage. Seeing games developed in pure javascript is a good starting point, but, should lead to working with a tool like ELM where one has more control over the code that is edited, and then deployed. The EGameFlow Survey will be used to evaluate future web browser based simulation software that is being developed for 2D and 3D Volleyball Offensive and Defensive Simulations.

Future Directions 2

Advocating Elm as a Tool to improve and modify Elm Apps that have been developed for Coaching Purposes

In Future Directions 2, we will examine the creation of the PasStat 2016 program. PasStat was developed using the Elm Programming Language. http://www.elm-lang.org/ and http://www.knowthen.com/ are wonderful resources for learning how to write code in Elm. The purposes for using Elm to develop PasStat are twofold. First, the language is organized in such a way that extensive programming experience is not required to write effective code for single page applications such as PasStat. Having tried to program in web browser client side code using JavaScript, this researcher discovered that a codebase such as elm was necessary to be able to make sense of the data entry mechanisms, and the data storage capabilities that were needed to create an app that would meet the needs of research. In database terms, the all inclusive acronym for the research needs of PasStat were Create, Read, Update, and Delete or CRUD. One would think that it would be quite simple to create a CRUD application that could collect data on a volleyball passer, or a team of volleyball passers, that could run on a web browser online or offline, and provide data for in game and post game review of the passers' performance. The rather simple math in terms of calculating passers average scores as an individual and as a team were met

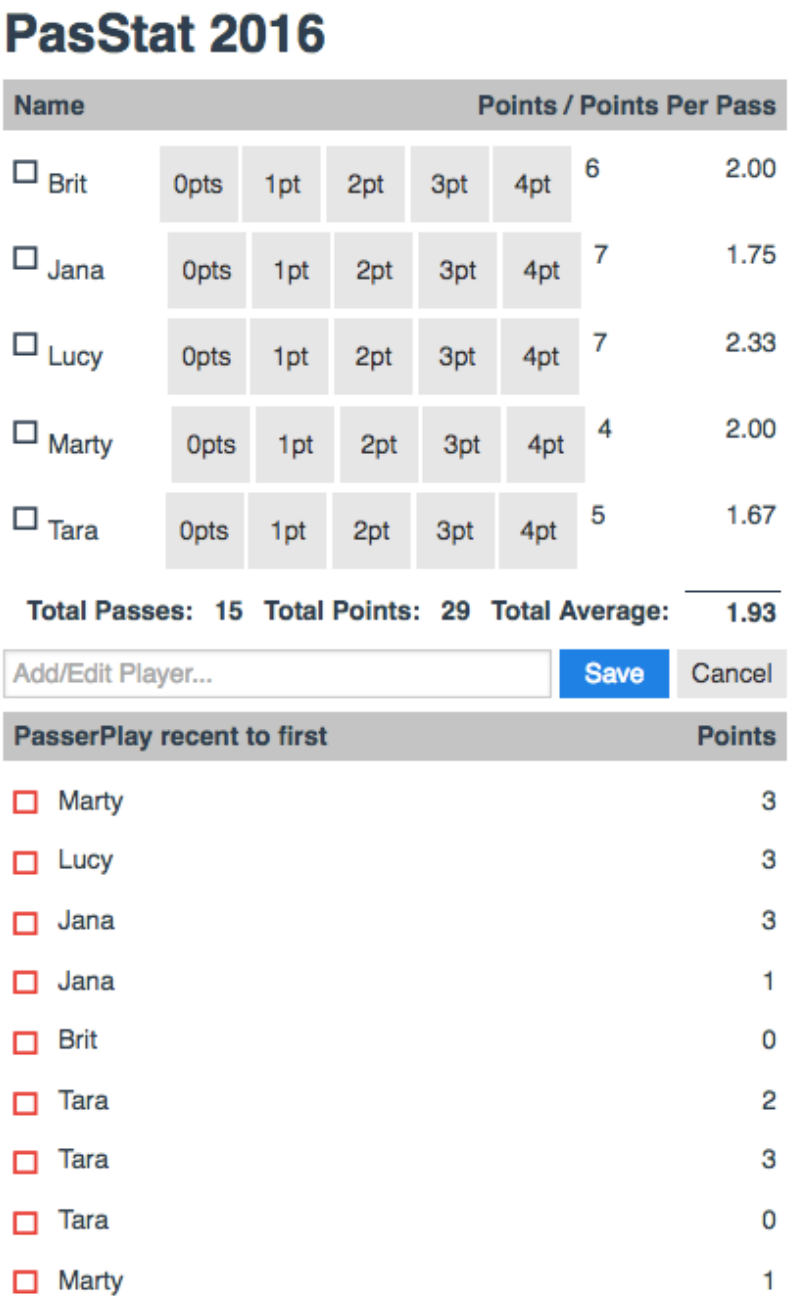
with lots of obstacles. Nevertheless, Elm had native and simple data structures within Elm itself which kept from having to learn a separate database programming language like the structured query language (SQL) to store, update, and query the data being collected on the web page for PasStat. It is likely that until and unless someone with non programming experience tries to work with Javascript and connect ones data using SQL it is difficult to appreciate ELM for the ways in which in simplifies the process of creating a tool such as PasStat. But after at least one hundred hours of looking for the means by which to create PasStat in way that it could be understood, replicated, and/or modified, this researcher can honestly say that Elm is the most elegant solution out there.

Before diving into Elm, an important concept to understand is that Elm actually compiles the Elm code into Javascript. So, the most operational part of the code is written in the language that is native to the web browser. Thus, the code that will be documented in this book, is Elm Code that was written prior to the javascript compilation process. On the command line of Elm, compilation occurs with one rather simple command. But, in the interests of documenting the Elm Code of PasStat, that topic will not be discussed here. The Elm Code for PasStat is a modification of a basketball scoreboard app that was written by James Moore, the author of KnowThen.Com. In Moore's Elm for Beginners, one can walk through (free of charge) the basics of creating the Basketball Scorekeeper App. So, there is no need to make this a technical manual for writing

Elm Code. Instead, it is hoped that the simplicity of the language will shine through as the code for PasStat is shared here.

Elm Code in a simple single page application (SPA) for the web is contained in the main.elm file. The Main.Elm code for the mac can be written in a common text editor. For this project, Atom was the text editor of choice. Atom color codes the expressions in elm and also writes the line numbers for the code for reference purposes. (See Appendix #2: Elm Code for PasStat). Hopefully, after reviewing the code for PasStat one can make the initial assessment that the Elm Language looks more like plain english than most other languages. For a detailed walk through each line of Code, KnowThen.Com Basketball Scorekeeper can give you a line by line blow by blow walk through the code.

Once the Elm Code was written for the PasStat App, the developer, using James Moore's setup as described at KnowThen.Com, was able to compile the elm code from main.elm to a javascript file called bundle.js which is inserted into a homepage called index.html. (See Appendix #3: HTML Code for PasStat.) The final code for the index.html file when it is viewed in a web browser, looks like this. There is a text box to enter player names and passing data. At the bottom plays are displayed which if entered incorrectly can be deleted.



To save data from PasStat, at the bottom right of the Single Page Application on the webpage, there is some text that says: View PasStat JSON, Load Saved, or SAVE. When one clicks on SAVE to save the file, the PasStat program saves a Text file

(that is stored in a plain text data schema called Javascript Object Notation or JSON) that can be reloaded for future viewing and editing, or for sharing with the team at a later time. Save Mode is a feature of Elm Version 0.18, and is not supported in earlier versions of Elm. PasStat 16 is live online and can be found at: http://www.wellnessoft.org/pass16/

Future Directions 3

Advocating Elm as a Tool for Coaches to Develop and Distribute Apps of Their Own

On of the biggest concerns that coaches have in terms of sharing their "trade secrets" is with regard to the problems of security on the internet. This researcher has found a strong level of security in working with a new protocol for saving files on a decentralized public network called SafeNet. Although SafeNet is in its early adaption period, this researcher has conducted a trial port of PasStat and the data collected there for use on the SafeNet Network. Although the researcher has successfully deployed Coaching Apps for Team Blogs, and Team Chat using Google Firebase Database, one must remember that Google Hosts that database, and can ultimately control who has access to the files. Google Firebase does allow for the databases created in Firebase to be password protected. But, for the ultimate level of security, SafeNet is the best way to preserve security of one's data.

Appendices

Appendix 1: Downhill Skier

Appendix 1: ELM CODE for Downhill Skier

```
1
2  module DownHillSkier where
3  {-| A simple game by mark mann where downhill skier course is a
   modified version maze game requiring mouse or touch precision
   under time pressure.
4  The game and the measured time begins when the player moves
5  to the marked start position. It is paused if the player crashed off
   of the course. Much like in actual ski racing.
6  He then has to go back the the start of the current downhill run.
7  If the level goal is reached, the next level is started automatically.
   After finishing the last level, the summed up time inversely
   indicates the players performance. ;-)
8  -}
9  {-| The game field extends from -200 to +200 in x and y
   coordinates. -}
10 (gameWidth,gameHeight) =  (200 ,200 )
11 {-| Since the player is ball shaped, this is his main property. -}
12  playerRadius = 5
```

Appendix 1: ELM CODE for Downhill Skier

```
13 {-| Every knot consists of position and a radius, and thus is the
   same as a ball. The radius (given as a factor to the players radius
   to the constructor) Levels are designed as a path of knots. To
   provide a smooth gaming experience with fluent transitions
   between two consecutive levels, the levels should be designed
   such 46 that the last point (l) of level n equals the first point (f) of
   level n+1. -}
14 radius of f <= radius f. These levels designed by Mark Mann can
   be modified and adjusted. They unlock the potentional to
   teaching programming to gaming enthusiasts and coaches alike.
   -}
15  levels : List Level
16 levels =
17     [
18      [
19        levelKnot -70 80 4
20      ,levelKnot 50 40 3.6
21      , levelKnot -55 16 3.3
22      , levelKnot 42 -20 3.1
23      , levelKnot 0 -50 2.8
24      ]
25      ,[
26        levelKnot 0 -50 2.5
27      , levelKnot -45 -62 3.5
```

Appendix 1: ELM CODE for Downhill Skier

```
28      , levelKnot 0 70 3.4
29      ]
30      ,[
31       levelKnot 0 70 3
32      , levelKnot 50 40 2.8
33      , levelKnot -85 16 2.6
34      , levelKnot 67 -20 2.4
35      , levelKnot 0 -50 2.2
36      ]


37     , [
38      levelKnot 0 -50 2.5
39      , levelKnot -35 -75 3.5
40      , levelKnot -70 90 3.4
41      ]
42     ,[
43      levelKnot -70 80 3
44     , levelKnot 82 50 2.8
45     , levelKnot -85 16 2.6
46     , levelKnot 67 -20 2.4
47     , levelKnot -81 -60 2.1
48     , levelKnot 0 -82 2.0
49      ]
```

Appendix 1: ELM CODE for Downhill Skier

```
    ]
  -- /--------------------\
-- | view configuration |
manualText = "Guide skier to end of run (green)."
respawnText = "Please go to the start (yellow) to begin."
timeTextHeight = 7
    timeTextPosY = 95
    textHeight = 5
    textPosY = -90
-- /—end of code-----\

```

Appendix 2: ELM CODE for PasStat

Appendix 2: ELM CODE for PasStat

```
1 module Main exposing (..)
2
3 import Html exposing (..)
4 import Html.Attributes exposing (..)
5 import Html.Events exposing (..)
6 import String
7
8
9 -- model

                                                                    10
type alias Model =
    { players : List Player
    , name : String
    , playerId : Maybe Int
    , plays : List Play
    }

type alias Player =
    { id : Int
    , name : String
```

```
    , points : Int
    , shots : Int
    }
type alias Play =
    { id : Int
    , playerId : Int
    , name : String
    , points : Int
    }

initModel : Model
initModel =
    { players = [ ]
    , name = ""
    , playerId = Nothing
    , plays = [ ]
    }
-- update

type Msg
    = Edit Player
    | Score Player Int
    | Input String
```

Appendix 2: ELM CODE for PasStat

```
    | Save
    | Cancel
    | DeletePlay Play

update : Msg -> Model -> Model
update msg model =
    case msg of
        Input name ->
            { model | name = name }

        Cancel ->
            { model | name = "", playerId = Nothing }

        Save ->
            if (String.isEmpty model.name) then
                model
            else
                save model

        Score player points ->
            score model player points

        Edit player ->
```

Appendix 2: ELM CODE for PasStat

```
            { model | name = player.name, playerId = Just
player.id }

        DeletePlay play ->
            deletePlay model play

deletePlay : Model -> Play -> Model
deletePlay model play =
    let
        newPlays =
            List.filter (\p -> p.id /= play.id) model.plays

        newPlayers =
            List.map
                (\player ->
                    if player.id == play.playerId then
                        { player
                            | points = player.points - 1 * play.points
                            , shots = player.shots - 1
                        }
                    else
                        player
                )
```

Appendix 2: ELM CODE for PasStat

```
            model.players
    in
        { model | plays = newPlays, players = newPlayers }


score : Model -> Player -> Int -> Model
score model scorer points =
    let
        newPlayers =
            List.map
                (\player ->
                    if player.id == scorer.id then
                        { player
                            | points = player.points + points
                            , shots = player.shots + 1
                        }
                    else
                        player
                )
                model.players

        play =
```

Appendix 2: ELM CODE for PasStat

```
                    Play (List.length model.plays) scorer.id
scorer.name points
    in
        { model | players = newPlayers, plays = play ::
model.plays }

save : Model -> Model
save model =
    case model.playerId of
        Just id ->
            edit model id

        Nothing ->
            add model


edit : Model -> Int -> Model
edit model id =
    let
        newPlayers =
            List.map
                (\player ->
                    if player.id == id then
                        { player | name = model.name }
```

```
                else
                    player
            )
            model.players

        newPlays =
            List.map
            (\play ->
                if play.playerId == id then
                    { play | name = model.name }
                else
                    play
            )
            model.plays
    in
        { model
            | players = newPlayers
            , plays = newPlays
            , name = ""
            , playerId = Nothing
        }
```

Appendix 2: ELM CODE for PasStat

```
add : Model -> Model
add model =
    let
        player =
            Player (List.length model.players) model.name 0
0

        newPlayers =
            player :: model.players
    in
        { model
            | players = newPlayers
            , name = ""
        }



-- view


view : Model -> Html Msg
view model =
    div [ class "scoreboard" ]
        [ h1 [ ] [ text "PasStat 2016" ]
```

```
        , playerSection model
        , playerForm model
        , playSection model
        ]


playSection : Model -> Html Msg
playSection model =
    div [ ]
        [ playListHeader
        , playList model
        ]


playListHeader : Html Msg
playListHeader =
    header [ ]
        [ div [ ] [ text "Plays" ]
        , div [ ] [ text "Points" ]
        ]


playList : Model -> Html Msg
playList model =
```

Appendix 2: ELM CODE for PasStat

```
    model.plays
        |> List.map play
        |> ul [ ]


play : Play -> Html Msg
play play =
    li [ ]
        [ i
            [ class "remove"
            , onClick (DeletePlay play)
            ]
            [ ]
        , div [ ] [ text play.name ]
        , div [ ] [ text (toString play.points) ]
        ]


playerSection : Model -> Html Msg
playerSection model =
    div [ ]
        [ playerListHeader
        , playerList model
```

```
        , pointTotal model
        ]


playerListHeader : Html Msg
playerListHeader =
    header [ ]
        [ div [ ] [ text "Name" ]
        , div [ ] [ text "Points / Points Per Pass" ]
        ]


playerList : Model -> Html Msg
playerList model =
    -- ul [ ]
    --     (List.map player model.players)
    model.players
        |> List.sortBy .name
        |> List.map (player model.playerId)
        -- curry to include playerId for edit mode
        |>
            ul [ ]
```

Appendix 2: ELM CODE for PasStat

```
player : Maybe Int -> Player -> Html Msg
player editPlayerId player =
    li [ ]
        [ i
            [ class "edit"
            , onClick (Edit player)
            ]
            [ ]
        , p [ class (editPlayerClass editPlayerId player) ]
            -- add class for edit mode
            [ text player.name ]
        , button
            [ type_ "button"
            , onClick (Score player 0)
            ]
            [ text "0pts" ]
        , button
            [ type_ "button"
            , onClick (Score player 1)
            ]
            [ text "1pt" ]
        , button
            [ type_ "button"
```

```
            , onClick (Score player 2)
            ]
            [ text "2pt" ]
        , button
            [ type_ "button"
            , onClick (Score player 3)
            ]
            [ text "3pt" ]
        , div [ ]
            [ text (toString player.points) ]
        , div [ ]
            [ text
                (if player.shots > 0 then
                    (toFloat player.points)
                        / (toFloat player.shots)
                        |> toString
                else
                    ""
                )
            ]
        ]

pointTotal : Model -> Html Msg
```

Appendix 2: ELM CODE for PasStat

```
pointTotal model =
    let
        total =
            List.map .points model.plays
                |> List.sum
    in
        footer [ ]
            [ div [ ] [ text "Total:" ]
            , div [ ] [ text (toString total) ]
            ]


playerForm : Model -> Html Msg
playerForm model =
    Html.form [ onSubmit Save ]
        [ input
            [ type_ "text"
            , placeholder "Add/Edit Player..."
            , onInput Input
            , value model.name
            , class (editInputClass model.playerId)
              -- adds edit class for edit mode
            ]
```

Appendix 2: ELM CODE for PasStat

```
            [ ]
        , button [ type_ "submit" ] [ text "Save" ]
            , button [ type_ "button", onClick Cancel ] [ text
"Cancel" ]
        ]


editInputClass : Maybe Int -> String
editInputClass editPlayerId =
    case editPlayerId of
        Just id ->
            "edit"

        Nothing ->
            ""


editPlayerClass : Maybe Int -> Player -> String
editPlayerClass editPlayerId player =
    case editPlayerId of
        Just id ->
            if player.id == id then
                "edit"
            else
```

Appendix 2: ELM CODE for PasStat

```
            ""

        Nothing ->
            ""


main : Program Never Model Msg
main =
    Html.beginnerProgram
        { model = initModel
        , view = view
        , update = update
        }
```

Appendix 3: HTML CODE for PasStat

Appendix 3: HTML CODE for PasStat

```
<!doctype html>
<html lang="en">

<head>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>PasStat 2016</title>
  <link rel="stylesheet" href="foundation.min.css">
</head>

<body>
  <div id="app"></div>
  <script src="bundle.js"></script>
  <script>
    var app = Elm.Main.embed(document.getElementById("app"));
  </script>
</body>

</html>
```