The SpinCoach

A. Terry Bahill
October 28, 2015

He who loves correction loves knowledge,
but he who hates reproof is stupid. Proverbs 12:1

This document was written to help the students at the University of Arizona to model and design systems. This document was **not** intended to be a *complete* design (for obvious reasons): instead, it tried to show a few good examples of each item that might be in a typical set of design documents.

This document was developed by Bahill. He developed it iteratively in parallel with the students' designs, but a little bit earlier. He exposed his mistakes in real time. The students were supposed to watch and learn from his mistakes. Then he incorporated the best of their ideas into this document.

Document 1: The Problem Situation

System Synopsis

For the baseball batter who needs to predict the trajectory of the pitch, the SpinCoach™ is a training system that helps him to recognize the spin on the pitch and predict the ball's spin-induced movement; unlike present coaches and books, the SpinCoach shows the batter how each pitch spins and helps him to recognize this spin.

The Product and Process

When a spinning object (like a baseball) is put in a moving fluid (like air), it will experience a force that pushes it sideways. Some highly successful baseball players have written that they see this spin of the ball and use this information to predict its future trajectory. But at present, there is no system that can teach high school and college baseball and softball players to predict this spin-induced deflection of the pitch. Our customer needs such a system. The goal of this project is to design such a system. The system will be capable of displaying images of spinning balls, allowing the subject to predict the spin induced deflection and providing feedback to facilitate learning.

Our task is to design and document the design of a-system that will help train baseball and softball players to pick up the spin on the ball and predict the spin's effect on the ball's deflection. Most baseball pitchers use one of two grips: the two-seam grip or the four-seam grip. The ball's appearance to the batter is different for the two-seam and the four-seam fastballs. In Bahill's laboratory (room 257 of Engineering building) we skewered baseballs on bolts in the two-seam and four-seam orientations. The bolts were chucked in electric drills and were rotated at 1200 rpm. The speed of rotation was measured with a stroboscope. Pictures and videos are available at: http://www.sie.arizona.edu/sysengr/baseball/index.html. The equipment in this laboratory is available for this project.

We have considered the following alternative architecture concepts:

- Status quo
  - Player participates in a real ball game
  - Player participates in batting practice on a real ball field
- Two-wheel pitching machine mounted on home plate.
- Videos of spinning balls

- Computer simulations of spinning balls
- A helmet mounted system to help the batter in real-time. The input for this system could be multiple cameras mounted at various places in the stadium or a camera mounted on the fielder's helmet.
- Lectures in a classroom with slides, equations and videos
- Documents and videos on the SpinCoach web site
- Three-dimensional virtual reality CAVE.
- The player will interact with the system through a Nintendo Wii remote.

We have considered the following alternative *process* concepts.

- Design and build everything in-house
- Subcontract the design
- Rent facilities like a three-dimensional virtual reality CAVE in CCIT at the UofA or Bob Kenyon's laboratory at the University of Illinois at Chicago.
- Sell our ideas to Microsoft or to game manufactures like Sony or Nintendo, or baseball equipment manufacturers such as Batta Baseball Machines (www.batabaseball.com) or Hoover Fence Co. (www.hooverfence.com).

In previous years, students have used the following alternative teaming process methods.

- Have face-to-face meetings of the whole team at least once a week.
- Have a teleconference once a week.
- Use D2L
- Use Microsoft NetMeeting
- Do most communication with e-mail
- Create a website specifically for this project

The SpinCoach is a product of Bahill Intelligent Computer Systems™ (BICS™). Its purpose is to help baseball and softball batters to recognize the spin on the pitch and predict the ball's spin-induced movement.

BICS will design and offer for sale several versions of the SpinCoach: professional baseball (major and minor leagues), National Collegiate Athletic Association (NCAA) softball, NCAA baseball, High School (and Babe Ruth League, American Legion, etc.) baseball, Bobby Sox softball, fast pitch softball recreation leagues and Little League. This document is for the NCAA baseball version.

The **key decision** in this project is whether to design the first version for baseball or softball. For baseball, the knowledge and drilled balls are already available. For softball, this would have to come from Mike Candrea and he is a bit busy now with the Olympic team.

Stakeholders include BICS, Buyer, Player, managers and victims. Victims include the opposing pitchers, Barry Seiller the creator of Vizual Edge™, and, until they find out about it, non-UofA Pac-10 schools.

Constraint. The English system of units shall be used for pitch speeds and all dimensions of balls, bats, bases and the field. Spin rate shall be given in revolutions per second. SI units shall be used for all aspects of modeling, design and manufacture of the SpinCoach.

Your deliverables, the eight Wymorian design documents, are due December 6, 2006. For a complex project, this set of documents could be described as an analysis model (by software managers) or as a proposal (by people with DoD experience).

Concept of Operations (ConOps)

A softball or baseball player in professional, college, high school or Little League organizations will request to use the SpinCoach. Administrative stuff will be done. The player will start the system. The player can request a particular type of pitch or a sequence of pitches. For baseball, pitch types are fastball, curveball and slider; we do not have a changeup or a knuckleball. For softball, pitch types include fastball, riseball, curveball, screwball and drop. The system will present images of the desired pitch for between 0.2 and 5 seconds. During (or immediately after) the display of each pitch the player will indicate the predicted direction of the spin induced motion. If the player is correct, he or she will be given positive reinforcement, else negative reinforcement. The system will track each player's progress and issue reports.

Operational Concept Description (OCD)

The Concept of Operations (ConOps) describes the mission of an enterprise and its component systems. It gives a broad outline describing a series of operations that the system will go through. It states what the system must do and how it fit into the enterprise. It may include business process models. It is usually supplied by the government or the customer. The Operational Concept Description (OCD) is a lower-level description of how an individual system is to be used. It is usually written by the contractor. It might be Document 6. It is a Contract Deliverable Requirements List (CDRL) validation item. It contains text, uses cases and diagrams. An abbreviated ConOps for the SpinCoach is given above. The OCD for the SpinCoach would be a few dozen pages. It might contain details like the following.

The player will login onto the system and enter a player profile and session preference information. The player will request a single pitch or a sequence of pitches and a display duration for each pitch. The system alerts the player with an auditory or a visual cue that a pitch is about to come. One second later the system displays video images of a spinning baseball or softball for the selected DisplayDuration. The player predicts the direction of spin-induced deflection of the pitch and indicates this direction using a pointing device (mouse, trackball or joy stick). The player will have a window of time (the response window) in which to indicate his or her predicted direction of spin-induced deflection. This response window starts at the beginning of the pitch display. If the player responds within the response window, then the system records the player's response in the player's ResponseDataTable and provides positive or negative feedback to the player. Otherwise, it informs the player that he or she has missed the window of opportunity. In the sequence of pitches mode of operation, a certain number (typically ten) pitches will be displayed in a session. The pitches will be separated by a fixed delay period, typically 15 second. When a session is over, the player's data will be stored to be used in advice and reports. The systems will track each player's progress and issue reports

Legal values for PitchType are {four-seam fastball, two-seam fastball, curveball, slider}, legal values for DisplayDuration are = {0.2, 0.4, 0.6, 0.8, 1.0, 2.0, 5.0} seconds, Legal values for ResponseWindow are = {0.2, 0.4, 0.6, 0.8, 1.0, 2.0, 5.0} seconds. Default values are PitchType = four-seam fastball, DisplayDuration = five seconds and ResponseWindow = DisplayDuration.

A fastball spins at 20 revolutions per second (rps). So the Nyquist sampling frequency is 40 Hz. This means that to prevent aliasing we must take at least 40 images per second. Aliasing is what

makes wagon wheels appear to be rotating backwards in old Western movies. The video images on Bahill's web site were taken at 60 Hz. The fastball video was ten seconds, the slider was five seconds. Modern digital cameras are precise. So if we tell one to take ten seconds of images, we can trust that it will take ten seconds of images. So to test the display speed, we should time how long it takes the computer to present one of these videos. Now the curveball spins at up to 33 rps. So a 60 Hz camera is not fast enough. However, a camera is available that shoots at 100 Hz.

When the pitcher releases a baseball, the ball is 54 feet from the batter's eye and it subtends 0.26 degrees of arc. When it is two thirds of the way to the contact point, when the swing must begin, the ball subtends 0.77 degrees of arc. Perhaps our displays should have a changing image size.

Document 2: Customer Requirements

Input Ports and Output Ports

An enterprise level input port will accept items of photos and videos taken of balls spinning on drills in Bahill's laboratory, room 257 of the Engineering Building.

At the operation level, we have two input ports. The first input port, ReceivePlayerInformation, is an interface through which the Player enters answers to the system's questions, such as,

Do you want to see baseballs or softballs?
Do you want to bat right-handed or left-handed?
What is your batting average?
How long do you want the ball to be displayed?

This port could be implemented with a keyboard or a speech recognition system.

The second input port, ReceivePlayerResponse, is an interface that the Player uses to enter his or her predictions of the ball movement. The Player predicts the direction of spin-induced deflection and indicates this direction using a numeric keypad or a pointing device. For the keypad implementation, the down arrow (2 key) indicates that the Player thinks the ball will drop more than that due to gravity. The right arrow (6 key) indicates that the Player expects the ball to curve to the right, away from a right-handed batter and into a left-handed batter. The PgDn (3) key means the player expects the ball to curve to the right and drop, like a typical slider or curveball thrown by a right-handed pitcher. For the pointing device (mouse, trackball or joy stick), the Player merely points in the predicted direction of spin-induced defection.

The outputs of the system are (1) images of spinning baseballs or softballs, (2) a warning to the Player that a pitch is coming, (3) feedback to the Player about the correctness of his response (4) advice to the Player, (5) results of built-in self-test (BiST) and (6) data files. The following data files must be maintained: Login account names and passwords, PlayerDataTable, PlayerProfile, SessionPreferences, and ResponseDataTable.

Output Ports		
Name	Information that crosses the port	Hardware that the port could be allocated to.

ImageDisplay Port	Images of spinning baseballs or softballs	Computer monitor
AlertSignal Port	An alert to the Player that a pitch is coming,	Auditory or visual
GiveFeedback Port	Feedback to the Player about the correctness of the response	Monitor or speech synthesizer
Advice Port	Advice to the Player	Monitor, a file or a printout
BiST Port	Results of built-in self-test (BiST).	Probably the monitor
DataFiles Port	Data files	CD ROM driver, USB port, LAN port

Customer requirements

All requirements that are mentioned in the use case model are being put in the customer requirements document.

CuR1: Bahill Intelligent Computer Systems (BICS) shall advertise the sale of the SpinCoach. Derived: FR1-1.

CuR2: BICS shall be capable of accepting orders and payment. Derived: FR1-2.

CuR3: BICS shall be capable of shipping orders. Derived: FR1-3.

CuR4: The system shall implement login and logout. Derived: FR2-2.

These first four requirements are requirements on the process. They would be in the Business Model. The next requirements are requirements on the product. They belong in the System Model.

CuR5: The system shall gather and store the player's profile and session preferences. Derived: FR2-3.

CuR6: The system shall gather and store the player's cumulative responses Derived: FR2-4.

Model Mapping Rule: Documents 2 and 6 use the phrase "player's cumulative responses." In documents 3, 4 and 7 this phrase will be replaced with the class name PlayerDataTable. Requirements are supposed to say what not how. They should leave the details for the designer. Therefore, in these requirements I tried to not specify specific names for classes that certainly will be used. But after a while, I found out that I had classes with specific names. Therefore, to avoid confusion I came back and added these specific names into the requirements in Document 3, but not here in Document 2. The particular example here is the entity class named PlayerDataTable.

CuR7: The system shall have the capability of computing random numbers. Derived: FR3-1.

CuR8: The system shall present video images of various pitches on a display. Derived: FR3-3.

CuR9: The system shall be capable of timing DisplayDuration. Derived: FR3-4.

CuR10: The system shall accept player predictions about the direction of spin-induced deflections using a numeric keypad or a pointing device. Derived: FR3-5.

CuR11: The system shall provide feedback to the player. Derived: FR3-6 and NFPR3-1.

CuR12: The system shall store the player's session information in the player data table. Derived: FR3-10.

Potential future capabilities

Future versions will have multiple spin rates and multiple spin axes for all pitches.

BICS expects to sell annual upgrades.

The SpinCoach could be adapted for cricket and tennis.

Bahill holds U.S. patent number 5,118,102 for the Bat Chooser™ a system that computes the Ideal Bat Weight™ for individual baseball and softball batters. Marketing should try to incorporate licenses for this patent into the SpinCoach package.

Ubiquitous language glossary

Fastball A normal overhand throw. For professionals most will have a speeds of 85 to 95 miles per hour (mph) and backspins of 20 revolutions per second (rps), which will cause them to move to the right (from the perspective of a right-handed pitcher, for a three-quarter arm delivery) and not drop as far as expected due to gravity.

Curveball To produce this pitch, the pitcher snaps his wrist and produces top spin. For professionals it will have a speed of 70 to 80 mph and top spin of 30 to 33 rps, which will cause it to move down and to the left (from the perspective of a right-handed pitcher).

Slider This pitch is thrown like a football so that it spins lie a bullet, except that the axis or rotation is not coincident with the direction of motion, rather it is pointed up and to the left (from the perspective of a right-handed pitcher). For professionals it will have a speed of 80 to 85 mph and spin of 23 rps, which will cause it to move down and to the left, but not as far as a curveball.

Ball A pitch that does not pass through the strike zone and is not swung at by the batter. If such a pitch hits the batter, the batter shall be awarded first base. If the pitch touches the ground and bounces through the strike zone, it is a ball.

Ball A sphere 2.9 inches in diameter weighing 5.125 ounces. Balls are constructed in layers. At the center is a composition cork sphere encased in two thin layers of rubber, one black one red. On this is wound 121 yards of blue-gray wool yarn, 45 yards of white wool yarn, another 53 yards of blue-gray wool yarn, 150 yards of fine cotton yarn, and a coat of rubber cement. It is enclosed in a cowhide cover hand-stitched with 216 red cotton stitches.

Document 3: Derived Requirements

The attributes of the following requirements are listed row by row, because this is a Word document. If this were a spreadsheet, they would be listed column by column.

Functional Requirements

Identification tag (Id)	FR1-1
Name	Advertise SpinCoach
Text	BICS shall advertise the sale of the SpinCoach

Comment	This is a requirement on the process
Derived from	Doc2 CuR1 and the Sell the SpinCoach use case
Verify method	Inspection
Priority	High

Identification tag (Id)	FR1-2
Name	Sell SpinCoach
Text	BICS shall be capable of accepting orders and payment for the SpinCoach
Comment	This is a requirement on the process
Derived from	Doc2 CuR2 and the Sell the SpinCoach use case
Verify method	Inspection
Priority	Medium

Identification tag (Id)	FR1-3
Name	Ship SpinCoach
Text	BICS shall be capable of shipping the SpinCoach
Comment	This is a requirement on the process (the company and its processes) not on the product (the SpinCoach).
Derived from	Doc2 CuR3 and the Sell the SpinCoach use case
Verify method	Inspection
Priority	Low

Identification tag (Id)	FR2-1
Name	BiST
Text	The system shall have built-in self-test (BiST)
Comment	
Derived from	BICS company policy and the Learn Spin-induced Deflections use case
Verify method	Test
Priority	High

Identification tag (Id)	FR2-2
Name	Login
Text	The system shall accomplish login/logout and initialization.
Comment	This is a requirement on the process
Derived from	Doc2 CuR4 and the Learn Spin-induced Deflections use case
Verify method	Inspection
Priority	Low

Identification tag (Id)	FR2-3a

Name	Store player profile
Text	The system shall gather and store the player's profile in the PlayerProfile table.
Comment	
Derived from	Doc2 CuR5 and the Learn Spin-induced Deflections use case
Verify method	Test
Priority	Low
Model Mapping Rule	Documents 2 and 6 use the phrase "player's profile." In documents 3, 4 and 7 this phrase will be replaced with the class name PlayerProfile table

Identification tag (Id)	FR2-3b
Name	Store player preferences
Text	The system shall gather and store the player's preferences for the session in the SessionPreferences table.
Comment	
Derived from	Doc2 CuR5 and the Learn Spin-induced Deflections use case
Verify method	Test
Priority	Low
Model Mapping Rule	Documents 2 and 6 use the phrase "session preferences." In documents 3, 4 and 7 this phrase will be replaced with the class name SessionPreferences.

Identification tag (Id)	FR2-4
Name	Store information in player data table
Text	The system shall record the player's cumulative responses in the PlayerDataTable.
Comment	This table should have a maximum size and a lifetime. FR2-4 is a sibling of FR3-7
Derived from	Doc2 CuR6 and the Learn Spin-induced Deflections use case
Verify method	Test
Priority	Low
Model Mapping Rule	Documents 2 and 6 use the phrase "player's cumulative responses." In documents 3, 4 and 7 this phrase will be replaced with the class name PlayerDataTable

Identification tag (Id)	FR3-1
Name	Compute random numbers
Text	The system shall have the capability of computing random numbers
Comment	This should have a range, like 0 to 20, and a fidelity constraint.
Derived from	Doc2 CuR7 and the *Display Random Sequence* use case
Verify method	Inspection
Priority	Low

Identification tag (Id)	FR3-2
Name	Signal warning
Text	The system shall alert the player that a pitch is about to come. This could be a visual or an audible cue.
Comment	Once the implementation technique is selected, this will be quantified with size and luminance or a frequency and decibel level.
Derived from	OCD and the *Display Random Sequence* use case
Verify method	Test
Priority	Low

Identification tag (Id)	FR3-3
Name	Display video images
Text	The system shall present video images of various pitches on a display
Comment	
Derived from	Doc2 CuR8, OCD and the *Display Random Sequence* use case
Verify method	Test.
Priority	High, This is the heart of the system

Identification tag (Id)	FR3-4
Name	Time the DisplayDuration
Text	The system shall be capable of timing DisplayDuration
Comment	Legal values for DisplayDuration are = {0.2, 0.4, 0.6, 0.8, 1.0, 2.0, 5.0} seconds. The default value is five seconds.
Derived from	Doc2 CuR9, OCD and the *Display Random Sequence* use case
Verify method	Inspection
Priority	Medium

Identification tag (Id)	FR3-5
Name	Accept player predictions
Text	The system shall accept the player predictions about the direction of spin-induced deflections using a numeric keypad or a pointing device (mouse, trackball or joy stick).
Comment	
Derived from	Doc2 CuR10, OCD and the *Display Random Sequence* use case
Verify method	Test
Priority	High

Identification tag (Id)	FR3-6
Name	Provide feedback

Text	The system shall provide positive or negative feedback to the player
Comment	The system shall provide feedback to the player within 500 milliseconds of the player's response
Derived from	Doc2 CuR11, OCD and the *Display Random Sequence* use case
Verify method	Test
Priority	High

Identification tag (Id)	FR3-7
Name	Store player predictions
Text	The system shall store the player's predictions in the player's ResponseDataTable
Comment	
Derived from	OCD and the *Display Random Sequence* use case
Verify method	Test
Priority	Low

Identification tag (Id)	FR3-8
Name	Time the window of opportunity
Text	The system shall be capable of timing the window of opportunity (called the response window).
Comment	Legal values for ResponseWindow are {0.2, 0.4, 0.6, 0.8, 1.0, 2.0, 5.0} seconds. Default value for ResponseWindow is DisplayDuration. The system receives this information in the Learn Spin-induced Deflections use case.
Derived from	OCD and the *Display Random Sequence* use case
Verify method	Inspection
Priority	Low

Identification tag (Id)	FR3-9
Name	Count number of images displayed
Text	The system shall be able to count the number of video images displayed, from 0 to ten
Comment	
Derived from	OCD and the *Display Random Sequence* use case
Verify method	Inspection
Priority	Low

Identification tag (Id)	FR3-10
Name	Store session in player data table

Text	The system shall store information about the player's session in the PlayerDataTable
Comment	FR3-10 is a sibling of FR2-4
Derived from	Doc2 CuR12 and the *Display Random Sequence* use case
Verify method	Inspection
Priority	Low

Nonfunctional Performance Requirement

Id	NFPR3-1
Name	Response Time
Text	The system shall provide feedback to the player within 500 milliseconds of the player's response.
Trace to	FR3-8
Derived from	interviews with Bahill
Verify method	Test
Priority	Medium

Cost Requirement

Id	CoR1
Name	Selling price
Text	Manufacturer's recommended selling price of the SpinCoach shall not exceed $300.
Comment	This is still a soft target.
Derived from	
Verify method	Inspection
Priority	Medium

Schedule Requirement

Id	SR1
Name	Due date
Text	The documents describing the design model for this system shall be submitted on or before 3 PM, December 6, 2006.
Comment	Late projects will be assessed a penalty of 5% per day.
Derived from	Customer's statement in the SIE-454/554 syllabus
Verify method	Inspection
Priority	High

Risk Requirements

Id	RR1
Name	Check for competing patents
Text	Initiate patent search. If no other product is found, then initiate patent application. Otherwise, discuss licensing. Immediately contact our lawyer and ask if putting this document on my web site constitutes disclosure.
Derived from	Risk analysis done during SIE-554 Test 1 on October 18, 2006
Rationale	If a similar system has already been patented, then we will not be able to market the SpinCoach.

Verify method	Analysis
Priority	Medium

Validation Requirement

Id	VR1
Name	Collect validation data
Text	BICS shall collect data to prove that the SpinCoach is successfully training ball players to hit fastballs.
Derived from	BICS company policy
Rationale	We must make sure that we are training the player to perform the real-world task and not training the player to perform the simulation. We use batting average as a metric for success. It certainly is odd to have a validation requirement. Validation is a process of its own. Does it need a requirement? We are experimenting with this. It gives us the obligation to test the batters' performance. All of our other requirements are tests on our system.
Verify method	Analysis
Priority	High

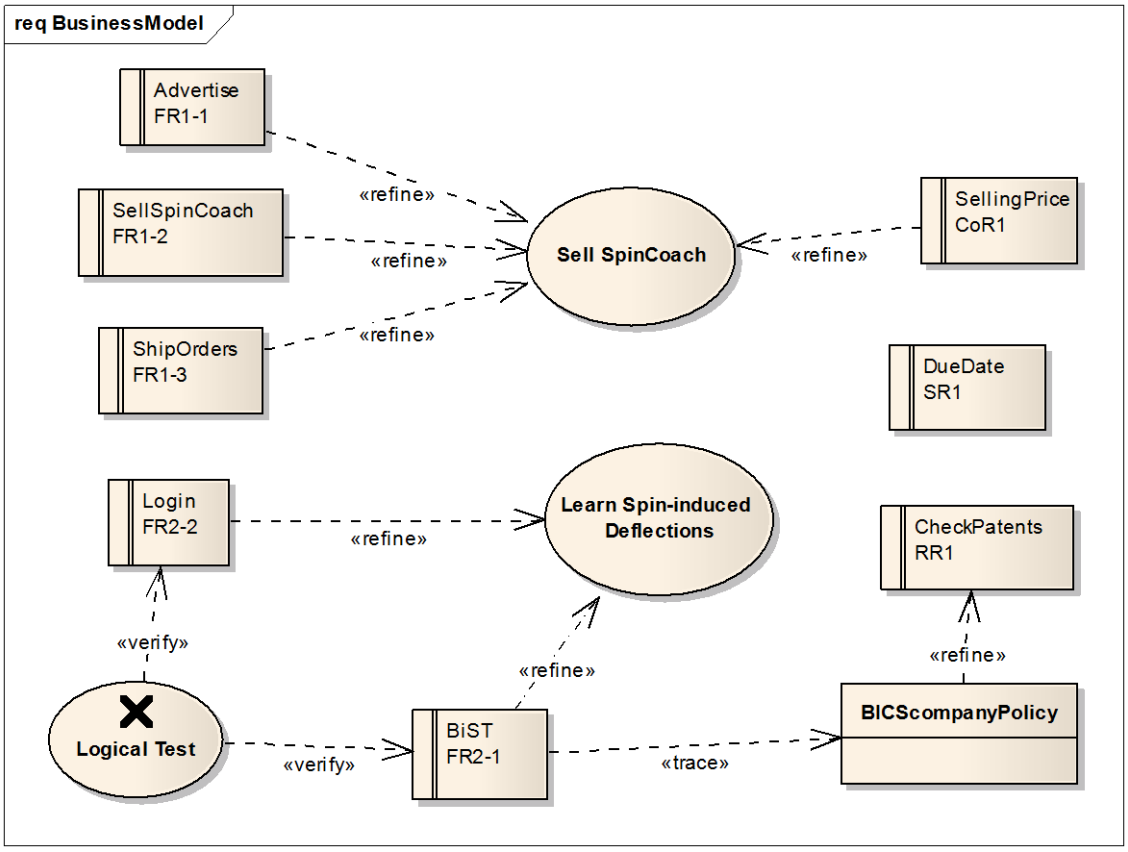

Figure 2. Requirements diagram for the Business Model.

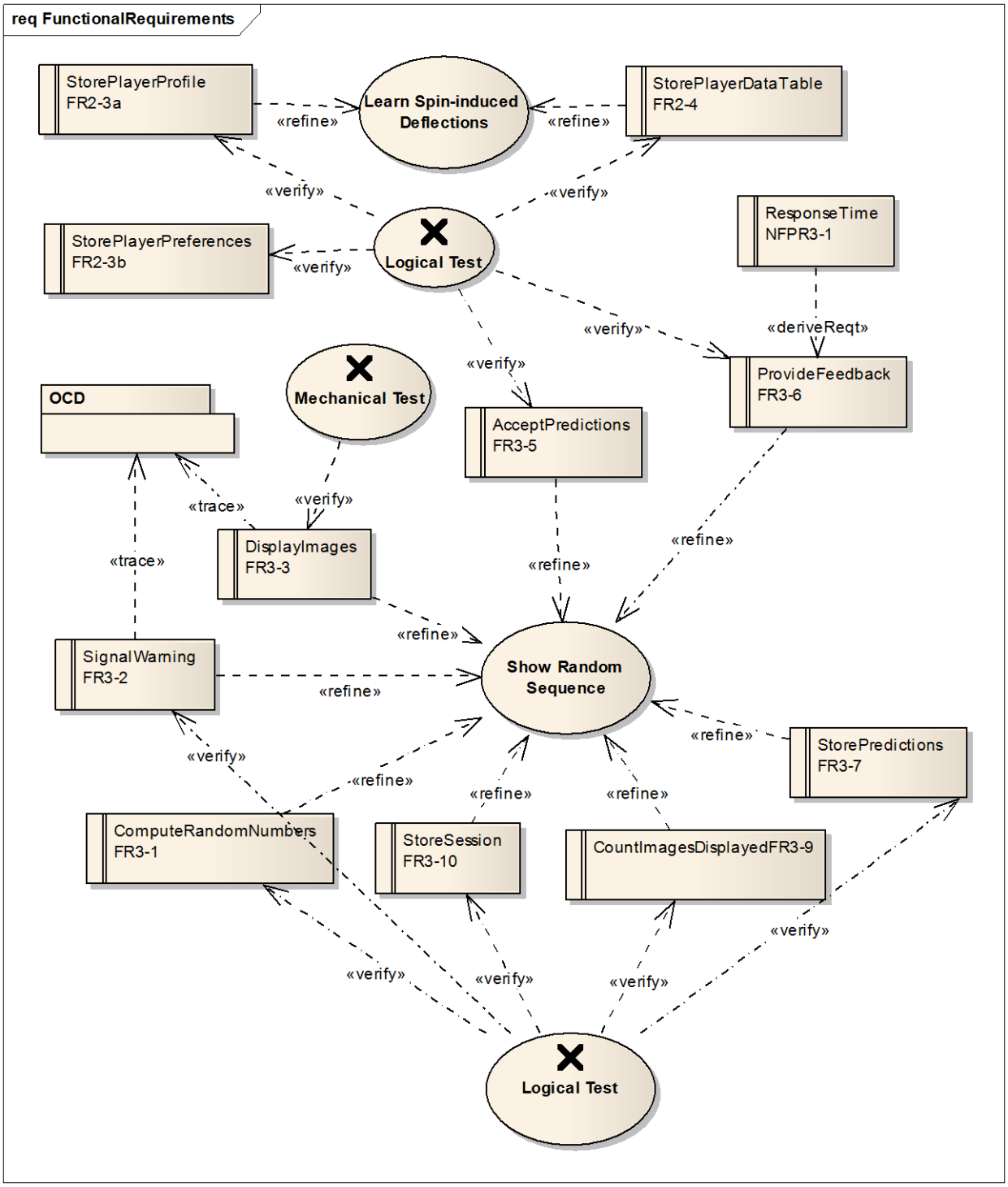

Figure 3. Requirements diagram for the functional requirements.

Document 4: System Verification and Validation

The requirements given in Document 3 are correct and consistent. After reading eleven student-team design documents in the fall of 2006, Bahill thinks that he has captured all of the necessary system requirements. Therefore, the requirements set is valid.

The equipment in Bahill's laboratory (room 257 of the Engineering Building) can satisfy most of the spinning ball requirements. Putting a sports visual training program on a CD ROM has been demonstrated by the commercial off the shelf product the Vizual Edge Performance Trainer™ (VEPT®) (http://www.vizualedge.com/). Rob Gray's laboratory at ASU displays computer simulations of pitches.

The system and the requirements are verifiable by test. The system test plan describes or implies tests for all of the requirements.

The biggest impediment to validation is proving that skills learned using the SpinCoach are transferable to the sport of baseball.

Validating the system means building the *right system*: making sure that the system does what it is supposed to do. Validation determines the correctness and completeness of the product, and ensures that the system will satisfy the actual needs of the customer.

Baseball and softball players need a system to teach them to identify fastballs. Observation of batters shows their difficulty in identifying 2-seam and 4-seam fastballs. Validation means proving that skills learned using the SpinCoach are transferable to the sport of baseball. We use batting average as a metric for success. For a high-level qualitative validation, we could ask experts such as high school and college coaches to use and evaluate the system.

After one of the alternatives is chosen as the preferred solution, then a paragraph will be added here justifying that it is indeed the best alternative.

If we chose one of the alternatives that pitches an actual baseball through the air (the status quo, a two-wheel pitching machine), then we conduct training sessions and record improvements in batting over time.

Verifying the system means building the *system right*: ensuring that the system complies with its requirements and conforms to its design.

System Test

The system shall have built-in self-tests. In addition, when it is in Test Mode, it will be subjected to mechanical and logical tests.

Mechanical Test

1. Tester puts the SpinCoach into Test mode.
2. Tester initiates Display Four-seam Fastball use case.
3. System displays images of a four-seam fastball for the default duration of five seconds. System describes verbally (with audio or written text) what visual attributes the player should attend to and how these visual attributes can be used to predict the spin-induced deflection of the ball.
4. Tester records result.
5. Tester initiates Display Two-seam Fastball use case.
6. System displays images of a two-seam fastball for the default duration of five seconds. System describes verbally (with audio or written text) what visual attributes the player should attend

to and how these visual attributes can be used to predict the spin-induced deflection of the ball

7. Tester records result.
8. Tester initiates Display Curveball use case.
9. System displays images of a curveball for the default duration of five seconds. System describes verbally (with audio or written text) what visual attributes the player should attend to and how these visual attributes can be used to predict the spin-induced deflection of the ball
10. Tester records result.
11. Tester initiates Display Slider use case.
12. System displays images of a slider for the default duration of five seconds. System describes verbally (with audio or written text) what visual attributes the player should attend to and how these visual attributes can be used to predict the spin-induced deflection of the ball.
13. Tester records result.
14. If the system passed all four of these test, then Tester certifies that SpinCoach is OK.

Logical Test

1. Tester puts the SpinCoach into Test mode.
2. Tester initiates the Learn Spin-induced Deflections use case.
3. System invokes Login/Logout use case.
4. The SpinCoach welcomes Tester to the system.
5. The SpinCoach asks Tester (1) if he or she wants to start a new profile or update an old one, (2) for experience level, (3) for type of game (softball, baseball, Little League), (4) Tester's handedness (Does the player bat right or left handed?), and (6) (if the player is currently playing in a league) current batting average.

Contents of player information menu			
Question	Legal values	Units	Default value
Update old data or start new profile?	{update, new}		new
Experience level?	1 to 100	Years	2
Type of game?	{softball, baseball, Little League}		baseball
Batting handedness?	{right, left}		right
Currently playing in a league?	{yes, no}		no
If currently playing in a league = yes, then current batting average?	0 to 1.0	Real	.200

6. Tester enters a test vector, then terminates the session and checks that the stored information is correct. Tester repeats this step for all possible input combinations, and then he or she continues with the use case.
7. Tester selects the Type of Event {single pitch, random sequence}, PitchType and DisplayDuration from a menu. As of September 2006, for a baseball SpinCoach, the menu contains PitchType = {four-seam fastball, two-seam fastball, curveball, slider}, DisplayDuration = {0.2, 0.4, 0.6, 0.8, 1.0, 2.0, 5.0} seconds.

Contents of session information menu			
Question	Legal values	Units	Default value
Same parameters as last session?	{yes, no}		No
Type of event?	{single pitch, random sequence}		random
If Type of event = single pitch, then PitchType?	{four-seam fastball, two-seam fastball, curveball, slider}		four-seam fastball
DisplayDuration?	{0.2, 0.4, 0.6, 0.8, 1.0, 2.0, 5.0}	seconds	5.0
ResponseWindow?	{0.2, 0.4, 0.6, 0.8, 1.0, 2.0, 5.0}	seconds	DisplayDuration

8. Tester enters a test vector, then terminates the session and checks that the stored information is correct. Tester repeats this step for all possible input combinations, and then he or she continues with the use case.
9. Tester enters a test vector, and then provides a correct response to the spin-induced deflection. Tester records the system response. Tester repeats this step for all possible input combinations.
10. Tester enters a test vector, and then provides an incorrect response to the spin-induced deflection. Tester records the system response. Tester repeats this step for all possible input combinations.
11. System updates and closes the PlayerDataTable.
12. Tester verifies that the information in the PlayerDataTable is correct.
13. The SpinCoach logs Tester out with the Login/Logout use case and says good-bye [exit use case].

Built-in self-tests (BiST) [FR2-1]	
The following quick self-tests are preformed at the start of the Learn Spin-induced Deflections use case.	
Test that the player has a legal copy of the SpinCoach	
Test integrity of player data files. Ensure that they can be read and written. Ensure that there is sufficient disk space for a session.	PlayerProfile SessionPreferences ResponseDataTable PlayerDataTable
Verify integrity of Spin Tester files	

The following system experiment will test the high-level system functioning.

Time	Present state	Input	Next state	BiST indicator	Related requirements
0	Performing BiST	BiST OK	Login	False	FR2-1
1	Login	LoginOK	Writing Player Profile	True	FR2-2
2	Writing Player Profile	Done	Gathering Session Information	True	FR2-3a
3	Gathering Session Information	Done	Selecting PitchType and Duration	True	FR2-3b
4	Selecting PitchType and Duration	Done	Managing Display and Interface	True	FR2-3b
5	Managing Display and Interface	Displayed 10 Pitches	Updating Whole PlayerDataTable	True	FR3-3 FR3-8 FR3-9
6	Updating PlayerDataTable & ResponseDataTable	Done	Logout	True	FR3-7 FR3-10
7	Logout	Done	Exit	True	FR2-2

Validating requirements means ensuring that the *set* of requirements is correct, complete and consistent, that a model can be created that satisfies the requirements, that a real-world solution can be built that satisfies the requirements, and that this real-world solution can be tested to prove that it satisfies the requirements. If the client has requested a perpetual motion machine, this is the place to stop the project and save the money.

The requirements given in Document 3 are correct and consistent. Eleven student teams have worked for a semester writing requirements for the SpinCoach. After reading these documents, Bahill thinks that he has captured all of the necessary system requirements. Therefore, the requirements set is correct, complete and consistent. We still need to prove that that a model can be created that satisfies the requirements, that a real-world solution can be built that satisfies the requirements, and that this real-world solution can be tested to prove that it satisfies the requirements.

Verifying requirements means proving that each requirement has been satisfied. Verification can be done by logical argument, inspection, modeling, simulation, analysis, test or demonstration**.** The verification matrix must show a one-to-one mapping between the requirements and the test plan. Its audience is Systems Engineering and the customer.

The way my Documents 2 and 3 are written, we do not have to verify any of the Document 2 requirements, because they all have derived requirements in Document 3. In this section, we are

only going to verify the system functional requirements: we are not going to verify the process requirements, the nonfunctional performance requirements, the cost requirements, the schedule requirements or the validation requirements.

Identification tag (Id)	FR2-1
Name	BiST
Text	The system shall have built-in self-test (BiST)
Verification	This requirement is tested during the system experiment of the System Tests.

Identification tag (Id)	FR2-2
Name	Login
Text	The system shall accomplish login/logout and initialization.
Verification	This requirement is tested during the system experiment and the logical test of the System Test.

Identification tag (Id)	FR2-3a
Name	Store player profile
Text	The system shall gather and store the player's profile in the PlayerProfile table.
Verification	This requirement is tested during the system experiment and the logical test of the System Test.

Identification tag (Id)	FR2-3b
Name	Store player preferences
Text	The system shall gather and store the player's preferences for the session in the SessionPreferences table.
Verification	This requirement is tested during the system experiment and the logical test of the System Test.

Identification tag (Id)	FR2-4
Name	Store information in player data table
Text	The system shall gather and store the player's cumulative information in a PlayerDataTable
Verification	This requirement is tested during the logical test of the System Test.

Identification tag (Id)	FR3-1
Name	Compute random numbers
Text	The system shall have the capability of computing random numbers
Verification	This requirement is tested during the logical test of the System Test.

Identification tag (Id)	FR3-2
Name	Signal warning
Text	The system shall alert the player that a batted-ball is about to come. This could be a visual or an audible cue.
Verification	This requirement is tested during the mechanical test of the System Test.

Identification tag (Id)	FR3-3
Name	Display video images
Text	The system shall present video images of various pitches on a display
Verification	This requirement is tested during the mechanical test of the System Test.

Identification tag (Id)	FR3-4
Name	Time the DisplayDuration
Text	The system shall be capable of timing DisplayDuration
Verify method	Inspection

Identification tag (Id)	FR3-5
Name	Accept player predictions
Text	The system shall accept player predictions about where the batted-ball will land the using physical body movement or a pointing device (mouse, trackball or joy stick).
Verification	This requirement is tested during the logical test of the System Test.

Identification tag (Id)	FR3-6
Name	Provide feedback
Text	The system shall provide positive or negative feedback to the player n a manner that increases learning.
Verification	This requirement is tested during the logical test of the System Test.

Identification tag (Id)	FR3-7
Name	Store player predictions
Text	The system shall store the player's predictions in the player's ResponseDataTable
Verification	This requirement is tested during the system experiment and the logical test of the System Test.

Identification tag (Id)	FR3-8
Name	Time the window of opportunity
Text	The system shall be capable of timing the window of opportunity (also called the response window).
Verification	This requirement is tested during the mechanical test of the System Test.

Identification tag (Id)	FR3-9
Name	Count number of images displayed
Text	The system shall be able to count the number of video images displayed, from zero to ten
Verification	This requirement is tested during the system experiment and the logical test of the System Test.

Identification tag (Id)	FR3-10
Name	Store session in player data table
Text	The system shall store information about the player's session in the PlayerDataTable
Verification	This requirement is tested during the system experiment and the logical test of the System Test.

Document 5: Concept Exploration

Alternative architectures for the SpinCoach

1. The Status Quo. Some batters can recognize the spin on the ball and predict its spin induced movement. But they have difficulty verbalizing this capability and teaching it to others. In batting practice, we can have the pitcher announce to the batter "curve" and then throw a curveball: announce "slider" and then throw a slider. Etc. This could be done with a human pitcher or a pitching machine.

2. Computer Simulations. Images of spinning balls can be simulated and presented on a computer monitor. This is Rob Gray's system described in SpinTeacherGray.doc. I have filled out the tradeoff study as if we are going to use Gray's program, although we could or write our own program.

3. System on CD-ROM. Balls spinning on drills can be photographed and their images stored on CD-ROM disks. Such videos are on my web site. http://www.sie.arizona.edu/sysengr/baseball/index.html. These images along with the software program (application) will be transferred from the CD-ROM to the user's hard disk using a license key provided by BICS. There after the user runs the SpinCoach from his or her hard disk. The user must login for each session. The information gathered at login is used to track user performance history. System upgrades will be provided with new CD-ROMs. To facilitate history tracking, installation may have to be limited to one computer.

3b. Application Sold On-Line. A slight variant of the CD-ROM alternative is to sell the application through an on-line store with distribution via Internet downloads.

4. Interactive DVD. Balls spinning on drills can be photographed and their images stored on DVDs. Such videos are on my web site. http://www.sie.arizona.edu/sysengr/baseball/index.html. These images along with the software program (application) will be accessed by the user. There will be no user performance tracking. System upgrades will be provided with new DVDs.

5. Web-based Application. Balls spinning on drills can be photographed and their images stored on the BICS web server. Such videos are on my web site. http://www.sie.arizona.edu/sysengr/baseball/index.html. These images along with the software program (application) will be on an Internet accessible web site. This system will be based on the Apache web server with web pages written in PHP or HTML and video clips in AVI format. Access to the system will be granted by monthly subscription and login based authentication control. The user will have a profile in the system and can access this profile from any terminal connected to the Internet. The system will store user information in a database (such as MySQL). This information is used to track user performance history. System upgrades can be made on the web site at any time and will be transparent to the user.

6. Vizual Edge Addition. Add our module to the commercial off the shelf product the Vizual Edge Performance Trainer™ (VEPT®) (http://www.vizualedge.com/). http://sports.espn.go.com/ncaa/news/story?id=2481988

7. Television. Batters can watch television and wait for ESPN to show slow motion pictures of the ball and learn its spin induced movement. Al Nathan in the Physics Department at the University of Illinois has four complete major league games on his computer.

8. Put the drills and balls in a suitcase and sell it to individuals or teams.

9. Put the drills and balls in a suitcase and sell it to libraries.

10. Use a 3D CAVE in CCIT at the UofA or in Bob Kenyon's laboratory at the University of Illinois at Chicago. This is often called virtual reality.

11. Sell the SpinCoach as a service not as a product. Charge people to come into the Human Control Systems Laboratory (room 257 of the Engineering Building) and observe the spinning balls.

12. Sell the SpinCoach as a one-day seminar, delivered in a facility of the customer's choosing.

13. Sell the SpinCoach as individualized one-on-one tutoring.

14. Marketing will have to tell us if the market is large enough to consider sales in retail stores. I doubt it.

15. Manufacture a special ball containing a processor that will detect the speed, spin rate and spin axis and will then illuminate imbedded LEDs of different colors for different pitch types.

16. Make it into a video game and sell it to Nintendo, Sony or Microsoft.

Evaluation Criteria

Performance Evaluation Criteria (importance weight is 7)
Convenience is a measure of how convenient and easy it will be for the player to obtain and use the product. Subcriteria include Ease of Purchase, Portability, Number of Assistants Needed, Simplicity

of Facilities (e.g. will it be used on a baseball field, in a special laboratory like a CAVE, in a library or at home) and Number of Pitches per Minute that can be viewed. A monotonic increasing scoring shall be used. This should trace to the sales and marketing plan. Importance weight is 8.

Fidelity of Images. How realistic are the images? Are they two or three-dimensional? What is the resolution? What is the color depth? What is the update rate? Will the presentation vary depending on the processor speed or the communications bandwidth? For example, would the system degrade with a dialup telephone connection to the Internet? I don't know how to make a scoring function for this yet. This criterion traces to the Operational Concept Description (OCD) and is also related to Validation. Importance weight is 6.

Duration Control means that the player can control the duration of the image presentation. This should be a Boolean (yes or no) scoring function. This criterion traces to CuR9. Importance weight is 5.

Verification. Can we gather data that shows the player's progress in using the SpinCoach? Can we gather data that shows improvements in the player's performance in the real game? Importance weight is 10.

Feedback Time. The system shall provide positive or negative feedback to the player after each prediction. The system shall provide this feedback to the player within 500 milliseconds of the player's response. This will be a Boolean (yes or no) function. This traces to CuR11 and NFPR3-1. Importance weight is 10.

Cost Evaluation Criteria (importance weight is 4)
Product Production Cost is a measure of how much it will cost in U. S. dollars for BICS to produce one unit of the product. A monotonic decreasing scoring function shall be used (L=0, B=10, S=-0.1, U=500). Input range is 0 to 500 dollars, baseline is 10 dollars and slope is -0.1. This traces to CoR1. Importance weight is 6.

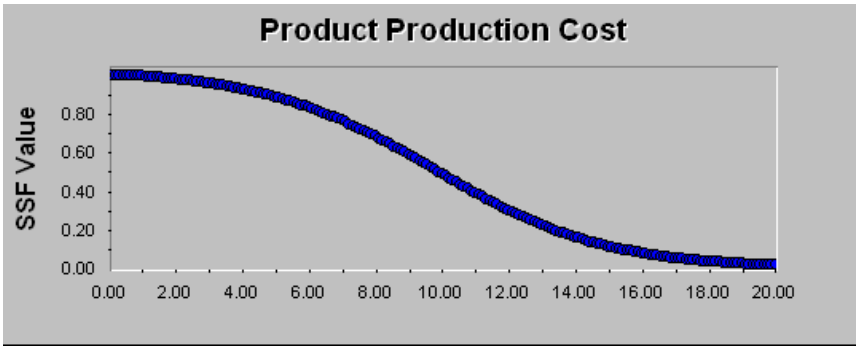

Figure 4. Product Production Cost monotonic decreasing scoring function.

The *Shipping Cost* evaluation criterion is composed of Shipping Weight, Shipping Expenses, Shipping Effort and Billing Cost per unit. A scoring function is not necessary if the subcriteria have scoring functions and normalized weights. Our target value is ten dollars. This traces to CoR1. Importance weight is 5.

Process Cost is a measure of the annual process production cost in thousands of U. S. dollars. This includes sales, marketing, advertising, development, engineering, upgrades and telephone support. A monotonic decreasing scoring function shall be used (L=0, B=200, S=-0.005, U=500). Baseline is $200,000. This traces to CoR1. Importance weight is 5.

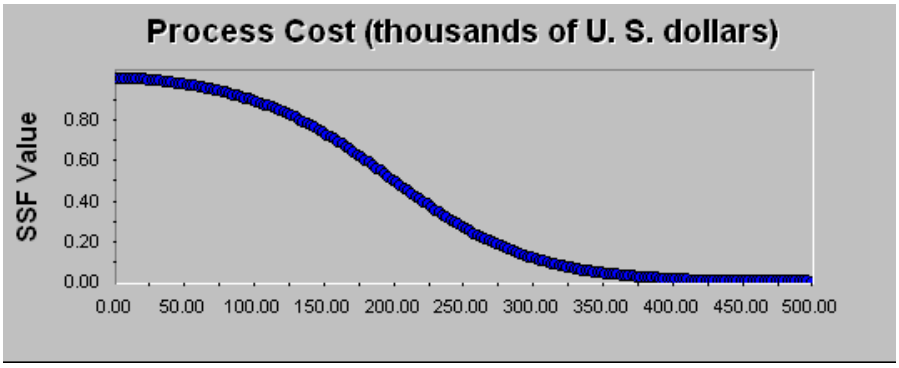

Figure 5. Process Cost monotonic decreasing scoring function.

Controllability assesses how much control BICS will have over the product. A monotonic increasing scoring shall be used. This traces to an interview with Terry Bahill on September 29, 2006. Importance weight is 4.

Schedule Evaluation Criteria (importance weight is 9)
Autonomy. Do we control the needed assets? Negotiating a contract could take months. Finding and buying equipment might take months. This traces to SR1. Importance weight is 10.

Updatability. The system shall be continually improved and updated throughout the system life cycle. Corrective maintenance such as bug fixes should be performed within weeks. Adaptive maintenance, which includes revisions necessary to allow the system to run on new or improved hardware and software, should be accomplished in a monthly time frame. Performance and functional updates will be performed yearly. This should trace to the business plan. Importance weight is 5.

Validation Evaluation Criteria (importance weight is 10)
Validation. Validity indicates the fidelity of transferring knowledge learned from the SpinCoach to the real world. We will use batting average as a metric for success. This criterion traces to BICS company policy and to requirement VR1 in Document 3. Importance weight is 10.

Other Evaluation Criteria (importance weight is 0)
I did not use *Safety* (and many other criteria that you suggested) because I did not think it would differentiate between alternatives. I did not include *Reliability*, or *Increase in Batting Average*, because these metrics would be useful only after we have an operating system. Other criteria that I did not include are *Operating Cost*, *Ability to Show Instructor* (i. e. present several minutes of video of the instructor talking to the student), *Percent of Pitches Player Identifies Correctly, Player's Perception of Effectiveness of Training,*

The tradeoff study matrix and sensitivity analysis for these alternatives and criteria is at http://www.sie.arizona.edu/sysengr/sie554/SpinCoach.xls.

An explanation of the sensitivity analysis is at
http://www.sie.arizona.edu/sysengr/sie554/sensit12short.doc

Other Tradeoffs

Alternative names for our system

1. Spin Teacher
2. Spin Trainer
3. Baseball (softball) Improvement System
4. Curveball Trainer
5. Train for Success System
6. Spin Master
7. Pitch Predictor
8. Pitch Recognition System
9. Spin Assessment Trainer
10. SpinCoach
11. Spin Professor

Alternative target audiences

1. Professional baseball (major and minor leagues)
2. National Collegiate Athletic Association (NCAA) softball and high school softball
3. NCAA baseball
4. High School baseball (and Babe Ruth League, American Legion, etc.)
5. Little League
6. Coaches and managers at all levels

Alternative pitches to be displayed

For baseball, the following pitches are candidates:

overarm 2-seam fastball (1200 rpm)
overarm 4-seam fastball (1200 rpm)
¾ arm 2-seam fastball (1200 rpm)
¾ arm 4-seam fastball (1200 rpm)
overarm 2-seam curveball (1800 rpm)
overarm 4-seam curveball (1800 rpm)
¾ arm 2-seam curveball (1800 rpm)
¾ arm 4-seam curveball (1800 rpm)
slider (1400 rpm)

Initially all pitches will be thrown by a right-handed pitcher.
We may want similar videos for pitches thrown by left-handed pitchers.

Alternatives for creating the curveball

1. Find a motor that will spin a softball at 2000 rpm. Reverse the direction of rotation. Rotate the platform 45 degrees. Take pictures with Alex's camera, which will shoot at 100 frames per second, or a video camera.
2. Rotate the images with Photoshop and speed up the rotation rate with some as yet unknown software package.

Alternative input devices
For input port 1
1. Computer keyboard
2. Speech recognition device
For input port 2
1. Keypad of a computer keyboard
2. Independent keypad
3. Pointing device (mouse, trackball or joy stick)

Alternative image formats (This needs some work)
1. Flash media file format (*.swf) by Macromedia
2. Mpeg
3. *.AVI
4. QuickTime

Killer Trades

On September 21, 2006, Bahill used the above criteria, performed a killer trades study and eliminated many alternatives.

The surviving architectures were

1. The Status Quo
2. Computer Simulations
3. System on CD-ROM
4. Interactive DVD
5. Web-based Application
6. Vizual Edge Addition

The surviving name for our system is the SpinCoach, because athletes like coaches more than they like teachers.

The surviving target audience is NCAA and high school baseball, because Mike Candrea has not gotten back to me with softballs and knowledge.

Tradeoff Matrix and Sensitivity Analysis

My tradeoff study matrix is in this same place on my web site. It suggests that the preferred alternatives are the System on a CD-ROM and the Web-based Application. The sensitivity analysis of this tradeoff matrix (http://www.arizona.edu/sysengr/sie554/SpinCoach.xls) shows that the most important parameter is the weight for the Schedule criterion. For a class project, schedule is the most important factor. We have Schedule as an Independent Variable (SAIV). We set the finish date and adjust cost and performance to meet that schedule. The most important subcriterion weight is for Autonomy. This is reasonable because Autonomy is the schedule driver. The most important scores are the scores for Autonomy of the six alternatives. The interactions of these three parameters were also important. This caused me to go back and reevaluate these numbers.

Document 6: The Use Case Model

Use Case 1
Name: Sell the SpinCoach

Iteration: 1.2
Derived from: Business Model
Brief description: BICS wants to sell the SpinCoach to individuals and organizations.
Added value: BICS will make a profit, which it can use to buy dog food for Sasha.
Level: Summary
Scope: Baseball and softball players throughout the United States.
Primary actor: BICS
Supporting actors: Buyer and Player**:** individual baseball and softball players and organizations
Frequency: Buyer should buy an upgrade once a year.
Precondition: BICS has a stock of SpinCoaches.
Trigger: BICS places advertisements.
Main Success Scenario:

1. BICS advertises and offers the SpinCoach for sale.

2. Buyer orders and pays for the SpinCoach. This step should include company-standard order and billing use cases.

3a. BICS ships the SpinCoach. This should also include a company-standard shipping use case.

4. Player uses the SpinCoach to learn the spin induced deflection of a ball. [exit use case].

Anchored Alternate Flows:

3b. The SpinCoach is out of stock and Buyer must be notified of a delay.
Specific Requirements

Functional Requirements:

FR1-1: BICS shall advertise the sale of the SpinCoach [from step 1].

FR1-2: BICS shall be capable of accepting orders and payment [from step 2].

FR1-3: BICS shall be capable of shipping orders [from step 3].
Postcondition: Buyer has the SpinCoach and BICS has the money.
Author/owner: Terry Bahill
Last changed: October 14, 2006

Use Case 2.
Name: Learn Spin-induced Deflections
Iteration: 2.3
Derived from: Concept of operations
Brief description: Player uses the SpinCoach and learns to predict the spin-induced deflection of a ball.
Added value: Player will be better able to predict the trajectory of the ball and consequently should have a higher batting average.
Level: User goal
Scope: One season for a college or high school baseball or softball player
Primary actor: Player
Supporting actors: This depends on the alternative that is chosen for implementation.
Frequency: Once a day 20 times per year
Precondition: Player has a legal copy of the SpinCoach
Trigger: Player starts up the SpinCoach.
Main Success Scenario:

1. System performs built-in self-test.

2. **Include** Login/Logout use case.

Note: The Login/Logout use case is a company-standard inclusion use case that will not be described in this documentation. Patterns for Login/Logout use cases are contained in Övergaard and Palmkvist (2005).

3. The SpinCoach welcomes Player to the system.

4. The SpinCoach asks Player (1) if he or she wants to start a new profile or update an old one, (2) for experience level, (3) for type of game (softball, baseball, Little League), (4) Player's handedness (Does the player bat right or left handed?), and (6) (if Player is currently playing in a league) current batting average.

Contents of PlayerProfile menu. This is an informational table. It is not a part of the use case.			
Question	Legal values	Units	Default value
Update old data or start new profile?	{update, new}		new
Experience level?	1 to 100	Years	2
Type of game?	{softball, baseball, Little League}		baseball
Batting handedness?	{right, left}		right
Currently playing in a league?	{yes, no}		no
If currently playing in a league = yes, then current batting average?	0 to 1.0	real	

5a. The Player enters player profile information.

6a. The Player chooses session preferences for Type of Event, PitchType and DisplayDuration from a menu.

Contents of Session Preferences menu. This is an informational table. It is not a part of the use case.			
Question	Legal values	Units	Default value
Same parameters as last session?	{yes, no}		no
Type of event?	{single pitch, random sequence}		random
If Type of Event = single pitch, then PitchType?	{four-seam fastball, two-seam fastball, curveball, slider}		four-seam fastball
DisplayDuration?	{0.2, 0.4, 0.6, 0.8, 1.0, 2.0, 5.0}	seconds	5.0
Response window?	{0.2, 0.4, 0.6, 0.8, 1.0, 2.0, 5.0}	seconds	DisplayDuration

7. System displays a pitch or a sequence of pitches.

8. System updates and closes the player data table.

9. The SpinCoach logs the Player out with the **include** Login/Logout use case and says good-bye [exit use case].

Anchored Alternate Flow:
5b. Tester must be able to interrupt the flow and check the player profile stored data
6b. Tester must be able to interrupt the flow and check the session preferences stored data.

Unanchored Alternate Flow:
This use case can be halted at any time if the Player presses Ctrl-C.
Postcondition: The system is reset and is waiting for a new player.

Extension points:		
Name	Location	Condition
Display Specified Images	Where the System displays a pitch or a sequence of pitches	This use case can invoke many different extension use cases depending on the Type of Event and the PitchType.

Specific Requirements

Functional Requirements:

FR2-1: The system shall have built-in self-test [from step 1 and BICS company policy].

FR2-2: The system shall accomplish login and logout. [from steps 2 and 9].

FR2-3: The system shall gather and store the player's profile and session preferences [from steps 4, 5 and 6].

FR2-4: The system shall gather and store the player's cumulative information in a player data table [from step 8].

Author/owner: Terry Bahill
Last changed: October 13, 2006
Note that once this use case is written, debugged and tested, it will never have to be altered or tested again, even after more pitch types are added. That is why we used extension use cases.

Use Case 3. This is an abstract extending use case.
Name: *Display Random Sequence*
Iteration: 2.3
Derived from: Conversations with Terry Bahill, President of BICS
Brief Description: The flow of this use case is inserted into the Learn Spin-induced Deflections use case at the Display Specified Images extension point if Event Type = random sequence. At the end of the display, this subflow ends and the use case instance continues according to the Learn Spin-induced Deflections use case at the Display Specified Images extension point. This concrete extending use case can be called by its extended use case (Learn Spin-induced Deflections) or Tester can initiate the use case.
Added value: The Player will see images of many pitch types in a game like situation.
Level: Subfunction
Scope: This use case displays a sequence of 10 pitches. The number of pitches to be displayed could be an input parameter.
Frequency: Many times per session

Precondition: This use case is invoked from the Learn Spin-induced Deflections use case at the Display Specified Images extension point if Event Type = random sequence.

Main Success Scenario:

1. The System uses a random number generator to choose a PitchType.
2. The System alerts the Player that a pitch is about to come.
3. One second later the System displays the selected PitchType for DisplayDuration.
4. The Player predicts the direction of spin-induced deflection and indicates this direction using a numeric keypad or a pointing device (mouse, trackball or joy stick).

Note: It is incorrect to write a use case at the level of individual keystrokes. So I will not. But here I will describe what the user's input will look like. The down arrow (2 key) indicates that the Player thinks the ball will drop more than that due to gravity. The right arrow (6 key) indicates that the Player expects the ball to curve to the right, away from a right-handed batter and into a left-handed batter. The PgDn (3) key means the Player expects the ball to curve to the right and drop, like a typical slider or curveball thrown by a right-handed pitcher.

5a. The Player will have a Window of Time in which to indicate his or her predicted direction of spin-induced deflection. This response window starts at the beginning of the pitch display. If the player responds within the response window, then the system records the player's response in the player's response data table and provides positive or negative feedback to the Player [repeat at step 1, for ten pitches].

6. The System updates the player's data table [exit use case].

Alternate Flow:

5b. System informs the Player that he or she has missed the window of opportunity [repeat at step 1, for ten pitches].

Postcondition: Nothing is being displayed.

Extension points:		
Name	Location	Condition
Display Selected Pitch	System displays selected pitch	[PitchType = two-seam fastball]
	System displays selected pitch	[PitchType = four-seam fastball]
	System displays selected pitch	[PitchType = curveball]
	System displays selected pitch	[PitchType = slider]

Specific Requirements

Functional Requirements:

FR3-1: The system shall compute random numbers [from step 1].

FR3-2: The system shall alert the Player that a pitch is about to come [from step 2]. This could be a visual or an audible cue.

FR3-3: The system shall present video images of various pitches on a display [from step 3].

FR3-4: The system shall be capable of timing DisplayDuration **[**from step 3].

FR3-5: The system shall accept the Player predictions about the direction of spin-induced deflections using a numeric keypad or a pointing device [from step 4].

FR3-6: The system shall provide positive or negative feedback to the Player [from step 5].

FR3-7: The system shall store the Player predictions in the player's response data table [from step 5].

FR3-8: The system shall be capable of timing the response window **[**from step 5].

FR3-9: The system shall be able to count the number of video images displayed, from 0 to ten **[**from step 5].

FR3-10: The system shall store the Player's session in the player data table [from step 6].

Nonfunctional Requirements:

NFPR3-1: The system shall provide feedback to the player within 500 milliseconds of the player's response [from interviews with Bahill].

Author/owner: Terry Bahill

Last changed: October 13, 2006

Note: The *Display Random Sequence* use case inherits values for Primary Actor from the Learn Spin-induced Deflections use case. The other slots are consistent, but more specific.

Use Case 4. This is a concrete extending use case.

Name: Display Four-seam Fastball

Iteration: 1.2

Derived from: Conversations with Terry Bahill, President of BICS

Brief Description: The flow of this use case is inserted into the Learn Spin-induced Deflections use case at the Display Specified Images extension point if PitchType = four-seam fastball or into the *Display Random Sequence* use case at the Display Selected Pitch extension point if PitchType = four-seam fastball. At the end of the display, this subflow ends and the use case instance continues at the extension point of the extended use case. This concrete extending use case can be called by either of its extended use cases **(**Learn Spin-induced Deflections or *Display Random Sequence*) or Tester can initiate the use case. Note: It would be better to make this extension use case an alternate flow of the *Display Random Sequence* or Learn Spin-induced Deflections use cases, but I want it to be separately callable by Tester.

Added value: The Player can learn the spin-induced motion of a four-seam fastball.

Level: Subfunction

Scope: This use case displays a single pitch.

Primary actor: Player or Tester

Frequency: Many times per session

Precondition: This use case is invoked from the Learn Spin-induced Deflections use case at the Display Specified Images extension point if PitchType = four-seam fastball, or from the *Display Random Sequence* use case at the Display Selected Pitch extension point if PitchType = four-seam fastball. It can also be invoked by Tester.

Main Success Scenario:

1. System displays images of a four-seam fastball for a duration passed in the DisplayDuration parameter. Default duration is five seconds. System describes verbally (with audio or written text) what visual attributes the Player should attend to and how these visual attributes can be used to predict the spin-induced deflection of the ball. [exit use case]

Postcondition: Nothing is being displayed.
Author/owner: Terry Bahill
Last changed: September 11, 2006

Use Case 5. This is a concrete extending use case.
Name: Display Two-seam Fastball
Iteration: 1.2
Derived from: Conversations with Terry Bahill, President of BICS
Brief Description: The flow of this use case is inserted into the Learn Spin-induced Deflections use case at the Display Specified Images extension point if PitchType = two-seam fastball, or into the *Display Random Sequence* use case at the Display Selected Pitch extension point if PitchType = two-seam fastball. At the end of the display, this subflow ends and the use case instance continues at the extension point of the extended use case. This concrete extending use case can be called by either of its extended use cases **(**Learn Spin-induced Deflections or *Display Random Sequence*) or Tester can initiate the use case.
Added value: The Player can learn the spin-induced motion of a two-seam fastball.
Level: Subfunction
Scope: This use case displays a single pitch.
Primary actor: Player or Tester
Frequency: Many times per session
Precondition: This use case is invoked from the Learn Spin-induced Deflections use case at the Display Specified Images extension point if PitchType = two-seam fastball, or from the *Display Random Sequence* use case at the Display Selected Pitch extension point if PitchType = two-seam fastball. It can also be invoked by Tester.
Main Success Scenario:

1. System displays images of a two-seam fastball for a duration passed in the DisplayDuration parameter. Default duration is five seconds. System describes verbally (with audio or written text) what visual attributes the Player should attend to and how these visual attributes can be used to predict the spin-induced deflection of the ball [exit use case].

Postcondition: Nothing is being displayed.
Author/owner: Terry Bahill
Last changed: September 11, 2006

Use Case 6. This is a concrete extending use case.
Name: Display Curveball
Iteration: 1.2
Derived from: Conversations with Terry Bahill, President of BICS
Brief Description: The flow of this use case is inserted into the Learn Spin-induced Deflections use case at the Display Specified Images extension point if PitchType = curveball, or into the *Display Random Sequence* use case at the Display Selected Pitch extension point if PitchType = curveball. At the end of the display, this subflow ends and the use case instance continues at the extension point of the extended use case. This concrete extending use case can be called by either of its extended use cases **(**Learn Spin-induced Deflections or *Display Random Sequence*) or Tester can initiate the use case.
Added value: The Player can learn the spin-induced motion of a curveball.
Level: Subfunction

Scope: This use case displays a single pitch.
Primary actor: Player or Tester
Frequency: Many times per session
Precondition: This use case is invoked from the Learn Spin-induced Deflections use case at the Display Specified Images extension point if PitchType = curveball, or from the *Display Random Sequence* use case at the Display Selected Pitch extension point if PitchType = curveball. It can also be invoked by Tester.
Main Success Scenario:

1. System displays images of a curveball for a duration passed in the DisplayDuration parameter. Default duration is five seconds. System describes verbally (with audio or written text) what visual attributes the Player should attend to and how these visual attributes can be used to predict the spin-induced deflection of the ball [exit use case].

Postcondition: Nothing is being displayed.
Author/owner: Terry Bahill
Last changed: September 11, 2006

Use Case 7. This is a concrete extending use case.
Name: Display Slider
Iteration: 1.2
Derived from: Conversations with Terry Bahill, President of BICS
Brief Description: The flow of this use case is inserted into the Learn Spin-induced Deflections use case at the Display Specified Images extension point if PitchType = slider, or into the *Display Random Sequence* use case at the Display Selected Pitch extension point if PitchType = slider. At the end of the display, this subflow ends and the use case instance continues at the extension point of the extended use case. This concrete extending use case can be called by either of its extended use cases **(**Learn Spin-induced Deflections or *Display Random Sequence*) or Tester can initiate the use case.
Added value: The Player can learn the spin-induced motion of a slider.
Level: Subfunction
Scope: This use case displays a single pitch.
Primary actor: Player or Tester
Frequency: Many times per session
Precondition: This use case is invoked from the Learn Spin-induced Deflections use case at the Display Specified Images extension point if PitchType = slider, or from the *Display Random Sequence* use case at the Display Selected Pitch extension point if PitchType = slider. It can also be invoked by Tester.
Main Success Scenario:

1. System displays images of a slider for a duration passed in the DisplayDuration parameter. Default duration is five seconds. System describes verbally (with audio or written text) what visual attributes the Player should attend to and how these visual attributes can be used to predict the spin-induced deflection of the ball [exit use case].

Postcondition: Nothing is being displayed.
Author/owner: Terry Bahill
Last changed: September 11, 2006

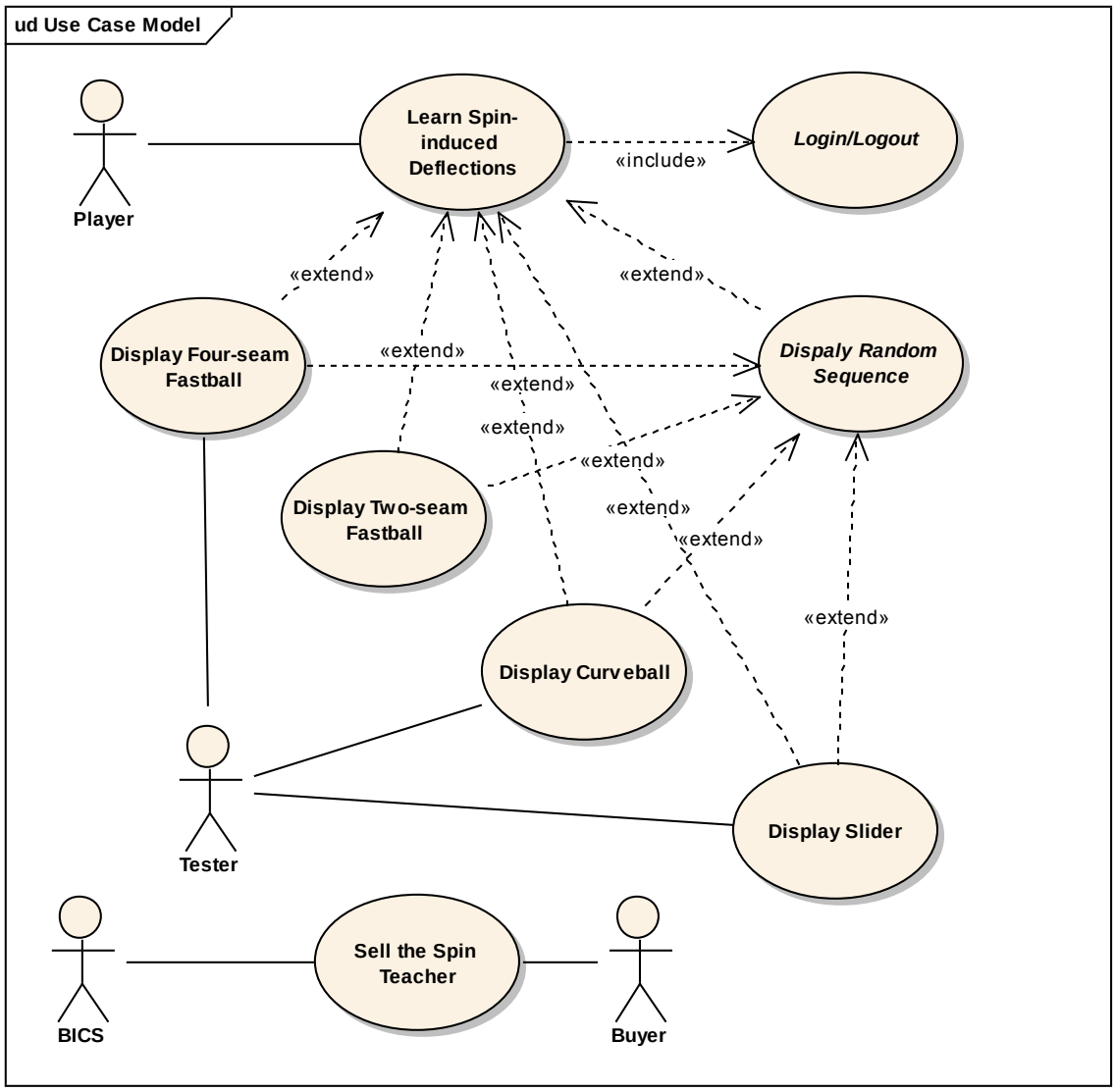

Figure 6. Use case diagram for the SpinCoach.

Document 7: The Design Model

The following classes were obtained by (1) thinking about concepts in the problem domain, (2) underlining the nouns in the use case and (3) considering the stereotypes of interfaces (boundary classes), data storage needs (entity classes) and controller classes.

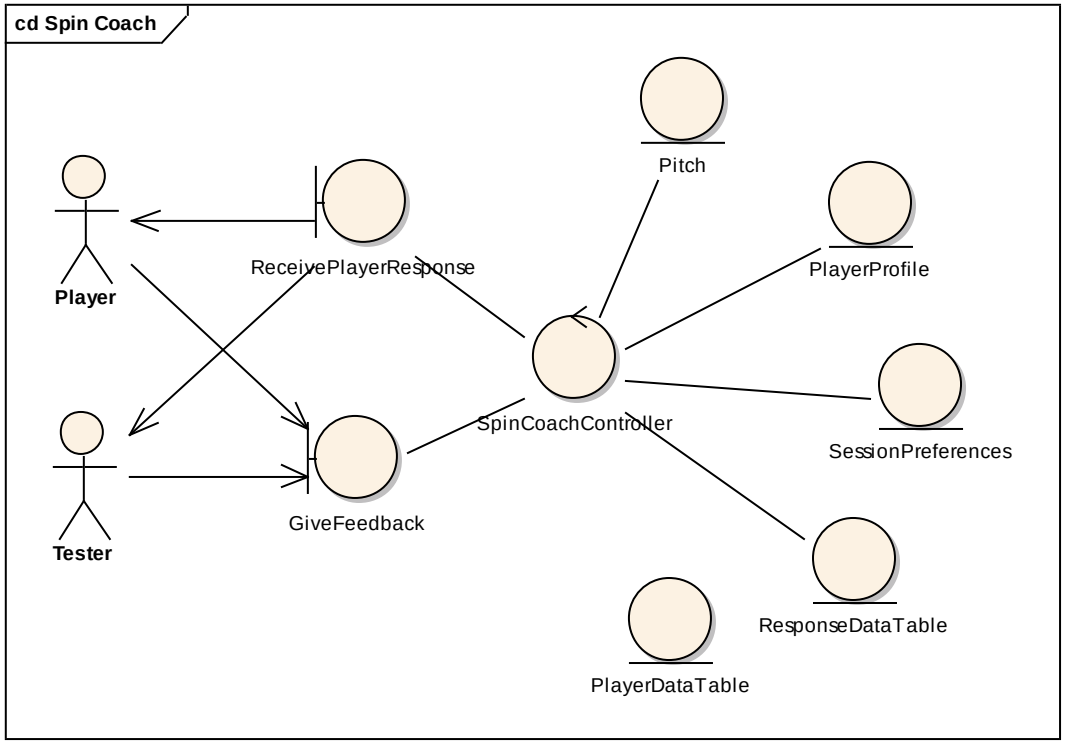

Figure 7. Class diagram for the SpinCoach.

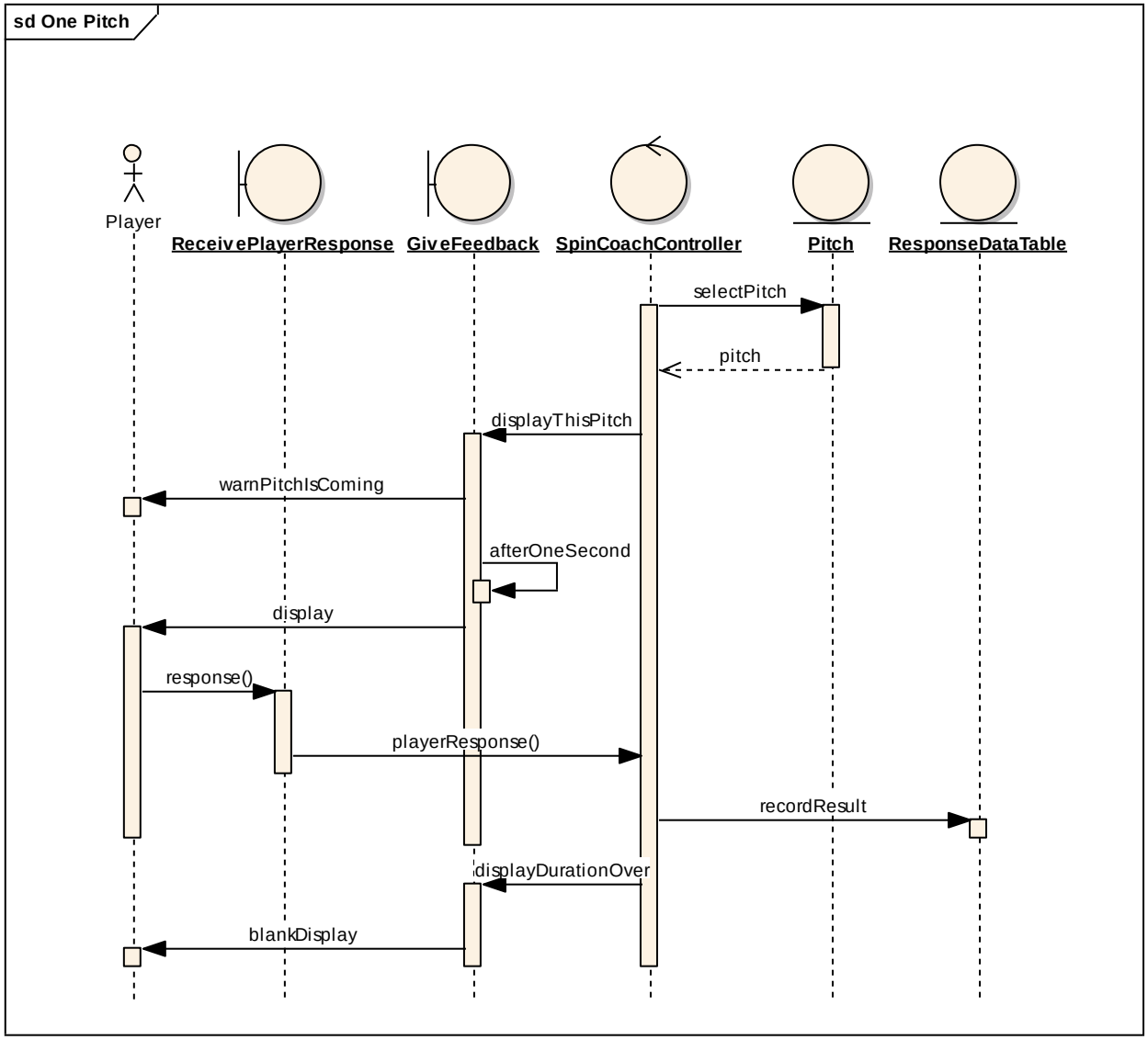

Figure 9. Sequence diagram for one pitch.

Analysis Classes

Pitch <<entity class>>
PitchType (for baseball): String = four-seam fastball {two-seam fastball, four-seam fastball, curveball, slider} DisplayDuration: Real = 5 {0.2, 0.4, 0.6, 0.8, 1.0, 2.0, 5.0} {seconds} ResponseWindow: Real = 5 {0.2, 0.4, 0.6, 0.8, 1.0, 2.0, 5.0} {seconds}

Lifetime: A Pitch exists from the moment it is created inside the SpinCoach system until the window of opportunity is over.

Player <<actor>>

name: String
catchBall()

Lifetime: A Player exists from the moment the player has entered into a relationship with BICS until the moment the Player has quit his or her relationship with BICS.

Tester <<actor>>
name: String

ReceivePlayerResponse <<boundary class>>
receivePlayerResponse()

GiveFeedback <<boundary class>>
warnPitchIsComing() displayPitchType() giveFeedback(positive, negative) blankScreen()

Alternative names for the boundary classes: Talk and Listen, Transmit and Receive, Tell and Hear, Player Response Form and Feedback Display.

PlayerProfile <<entity class >>
profile: String = new {update, new} experienceLevel: Integer = 2 {1..100} {years} typeOfGame: String = baseball {softball, baseball, Little League} battingHandedness: String = right {right, left} currentlyPlaying: String = no {yes, no} battingAverage: Real = .200 {0..1}

Lifetime: The PlayerProfile lasts for the lifetime of the Player.

SessionPreferences << entity class>>
sameParameters: String = no {yes, no} typeOfEvent: String = random {single pitch, random sequence} PitchType (for baseball): String = four-seam fastball {two-seam fastball, four-seam fastball, curveball, slider} DisplayDuration: Real = 5 {0.2, 0.4, 0.6, 0.8, 1.0, 2.0, 5.0} {seconds} ResponseWindow: Real = 5 {0.2, 0.4, 0.6, 0.8, 1.0, 2.0, 5.0} {seconds}

Lifetime: SessionPreferences exist from the moment a session has started until the moment the session has stopped.

ResponseDataTable << entity class>>
date: Date percentCurveballCorrect: Integer percentFastballCorrect: Integer percentSliderCorrect: Integer DisplayDuration: Real = 5 {0.2, 0.4, 0.6, 0.8, 1.0, 2.0, 5.0} {seconds}

Lifetime: ResponseDataTable exist from the moment a session has started until the moment the session has stopped.

PlayerDataTable << entity class>>
playerName: loginName: password: arrayOfAllSessionResults:

Lifetime: The PlayerDataTable lasts for the lifetime of the Player.

SpinCoachController <<control class>>
performBiST() loginPlayer() writePlayerProfile() gatherSessionPreferences() selectPitchTypeAndDuration() timeOneSecond() timeResponseWindow() timeDisplayDuration() gradeResponse() writePlayerDataTable() logoutPlayer()

Explanations

Functions and events	Explanation
From the class diagrams	
receivePlayerResponse()	The system must accept the Player's prediction of the direction he or she thinks the ball will move.
warnPitchIsComing()	The system must warn the Player that a pitch is coming.
displayPitchType()	The system must display a pitch of a specific type.
giveFeedback(positive, negative)	The system must tell the Player if he or she was right or wrong, perhaps the correct response will be indicated.

blankScreen()	The system must blank out the display screen.
performBiST()	The system must perform BiST.
loginPlayer()	The Player and the system must accomplish login.
writePlayerProfile()	The system must write information into the entity called PlayerProfile
gatherSessionPreferences()	The system must gather session information from the Player and store it in the entity Session Preferences.
selectPitchTypeAndDuration()	The Player selects the preferred pitch type and duration and gives this information to the system.
timeOneSecond()	The system must time a one-second interval. This is the time between the warning and the pitch.
timeResponseWindow()	The system must time an interval of time called the ResponseWindow. This is the window of opportunity during which the Player must respond.
timeDisplayDuration()	The system must time an interval of time called the DisplayDuration. This is amount of time that a pitch is displayed.
gradeResponse()	The system must determine if the Player's response is right or wrong.
writePlayerDataTable()	The system must write information into the entity PlayerDataTable
logoutPlayer()	The system must log the Player out.
Events from State Machine Diagrams	
BiST OK	The system determines that the BiSTs are OK and sends this message.
BiSTfailure	The BiST fails and the system send this message.
loginOK	The login process was successful
Displayed < 10 Pitches	Less than 10 pitches have been displayed.
Displayer 10 Pitches	Ten pitches have been displayed
OneSecondElapsed	The one-second interval between the warning and the pitch is over.
DisplayDurationOver	The specific duration to display the pitch is over.
PlayerResponse	This is the message that the Player sends to the system containing his or her prediction of the direction of the spin-induced deflection of the pitch.
WindowTimeOver	This means that the Player's window of opportunity for responding is over.

State Machine diagrams

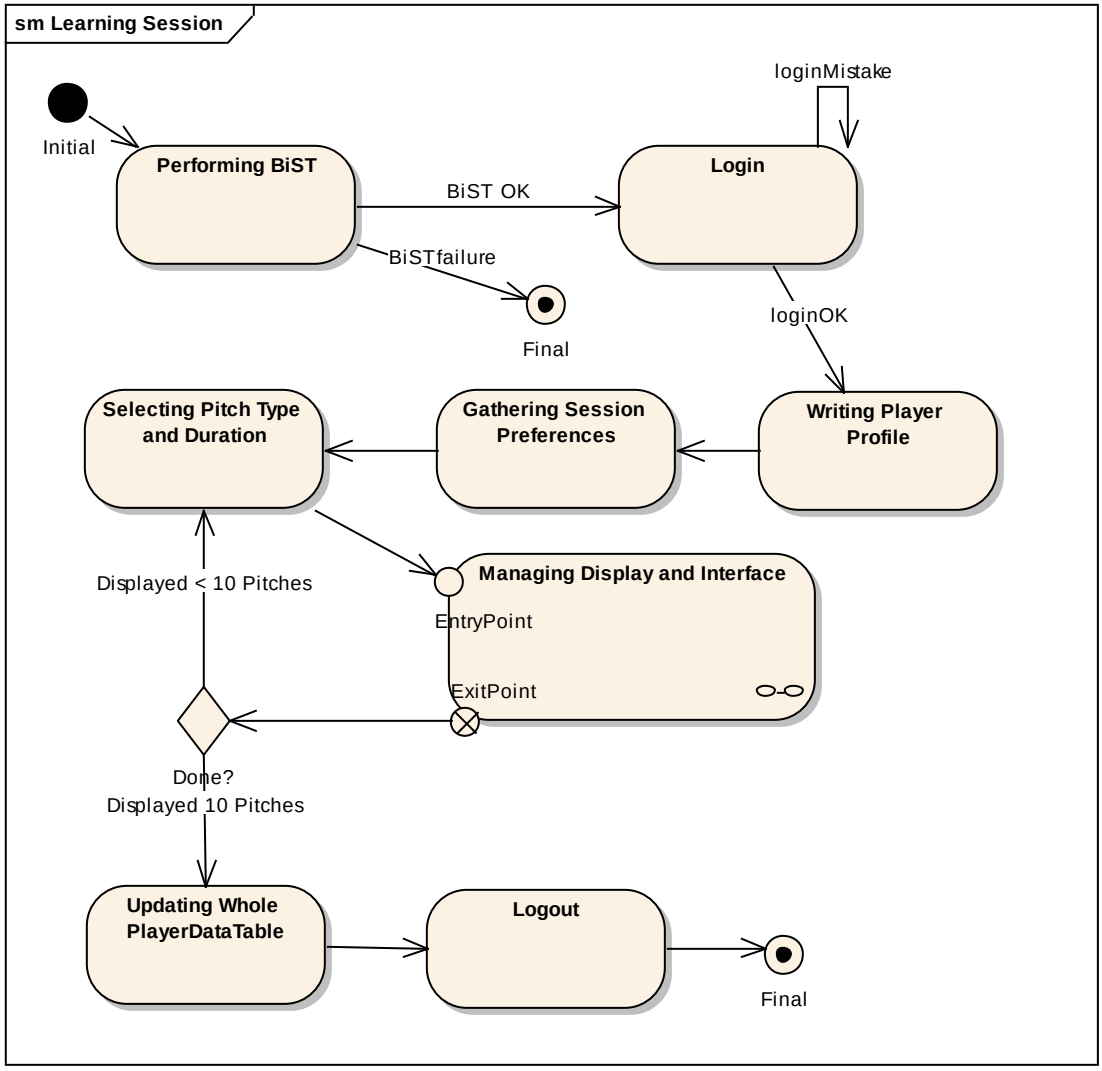

Figure 10. High-level state machine diagram.

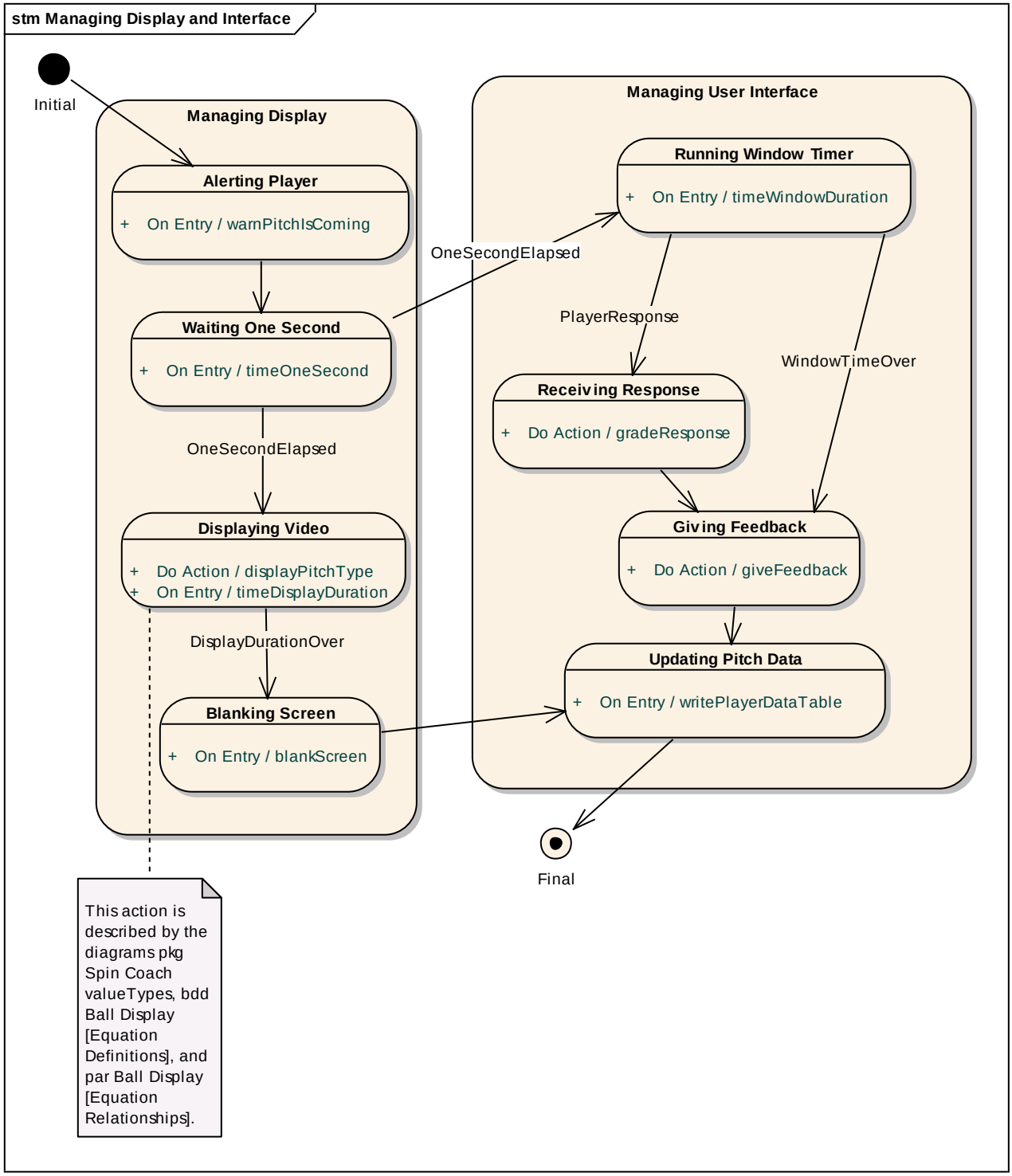

Figure 11. Lower-level state machine diagrams for the SpinCoachController of use cases 2 to 7.

Document 8: Models, Mappings and Management

Schedule

A detailed schedule for this project is given in the SIE-454/554 document "Difficulty and Value of SIE-454/554 Homework Problems."

Model mapping rules

Model mapping rules		
In docs 6 and 2, phrases like		in docs 3, 4 and 7 should be replaced with the class name
player cumulative responses	⇔	PlayerDataTable
player's cumulative responses	⇔	PlayerDataTable
player profile	⇔	PlayerProfile
player's profile	⇔	PlayerProfile
session preferences	⇔	SessionPreferences
player's responses	⇔	ResponseDataTable
verification and validation evidence	⇔	

Technical Glossary

Acronyms

BICS Bahill Intelligent Computer Systems

BiST Built-in self-tests

CDRL Contract Deliverable Requirements List

ConOps Concept of Operations

CoR Cost requirement

CuR Customer requirement

FR Functional requirement

MLB Major League Baseball

mph miles per hour

NCAA National Collegiate Athletic Association

NFPR Nonfunctional performance requirement

OCD Operational Concept Description

rps revolutions per second

RR Risk requirement

SI Systeme International d'Unites

SR Schedule requirement

TBD To be determined

Business Plan

The first version of the SpinCoach will be for NCAA softball. The second version will be for NCAA and high school baseball. The third version of the SpinCoach will be for NCAA and high school softball. The fourth version will be for Little League.

We will price the SpinCoach at $300. We expect process cost (development, engineering, patent applications, advertising and product support) to be $200,000 per year. This is primarily used to support one engineer, one technician and one secretary. Product cost (manufacture, packing and shipping) should be $10 per unit.

We estimate that there are 100 softball teams playing at the NCAA Division I level. If we could sell to half of those, we would get 50 sales at $300 each.

We estimate that there are a little under 300 baseball teams playing at the NCAA Division I level. If we could sell to half of those, we would get 150 sales at $300 each.

For the high school version, we estimate annual sales of 8,000 units. We estimate a recommended sales price of $300.

For the Little League version, we estimate annual sales of 500 units. This number is low (compared to the number of children playing Little League), because Little Leaguers are not supposed to throw curveballs: it will injure their arms. We estimate a recommended sales price of $300.

The pricing and marketing is crucial to the business success. We have not yet done a good job on this.

The SpinCoach would not be useful for slow-pitch softball.

BICS will match employee contributions to charitable organizations up to $1000 per year. Unfortunately, BICS is too small to have a maternity leave or a sabbatical leave program. Riding a motorcycle without a helmet or skydiving is an announcement by a BICS employee of a desire to terminate employment.

Risk Analysis

Failure Modes and Effects Analysis for the SpinCoach				
Failure Mode	Potential Effects	Probability	Severity	Estimated Risk
Mike Candrea is too busy with the Olympic Softball team to help us with the SpinCoach.	We design the first system for baseball	0.9	0.1	0.09

Alex and Zach both leave town before Terry takes all of the needed videos.	Terry must buy a new camera.	0.9	0.2	0.18
We cannot find drills that will spin the balls at 2,000 rpm.	Speed up the rotations with software or tolerate less fidelity.	0.7	0.4	0.28
No one in SIE-495 accepts the SpinCoach project	We don't have actual videos (in addition to what is already on Bahill's web site) for all the pitches	1	0.2	0.20
A similar system has already been patented	It will cost $5,000, but we should apply for a patent.	0.6	1	0.60
Display is dependent on processor speed.	SpinCoach performs differently on different computers. On some computers, we may have poor image fidelity. System may fail when computers become faster.	0.8	0.6	0.48

The greatest risk is that a similar system has already been patented. Therefore we should we should contact our lawyer and start a patent search.

Safety Analysis

Once the final alternative is chosen, a complete safety analysis will be done. It will include potential physical and psychological injury to the Player. Psychological injury could come in the form of unwanted disclosure of the results showing performance and progress using the SpinCoach or diminishing the player's self-esteem.

References

Bahill, A. T., Baldwin D. G. and Ramberg, J. S., Effects of altitude and atmospheric conditions on the flight of a baseball, *International Journal of Sports Science and Engineering*, http://www.worldacademicunion.com/journal/SSCI/online.htm, print ISSN 1750-9823, **3**(2):109-128, 2009.

A. T. Bahill, D. G. Baldwin, and J. Venkateswaran, Predicting a baseball's path, *American Scientist*, **93** (3) (2005) 218-225.

W. L. Chapman, A. T. Bahill and W. A. Wymore, *Engineering Modeling and Design*, CRC Press Inc., Boca Raton FL, Chapters 5 and 6, 1992.

W. L. Chapman, A. T. Bahill and W. A. Wymore, Pinewood, http://www.sie.arizona.edu/sysengr/pinewood/pinewood.pdf

J. Daniels and A. T. Bahill, The hybrid process that combines traditional requirements and use cases, *Systems Engineering*, **7**(4):303-319, 2004.

J. N. Martin, "Processes for engineering a system: an overview of the ANSI/EIA 632 standard and its heritage," *Systems Engineering*, **3**(1):1-26, 2000.

G. Övergaard and K. Palmkvist, *Use Cases: Patterns and Blueprints*, Addison-Wesley, Indianapolis, 2005.

R. G. Watts and A. T. Bahill, Keep Your Eye on the Ball: Curveballs, Knuckleballs and Fallacies of Baseball, W. H. Freeman, New York (2000).

W. A. Wymore, *Model-Based Systems Engineering*, CRC Press Inc., Boca Raton FL, Chapter 1, 1993.

A Little Bit of Physics

When a spinning object (like a baseball) is put in a moving fluid (like air), it will experience a force that pushes it sideways (Watts and Bahill, 2000). Spin-induced forces on a ball were studied by Newton (1671), Robbins (1742) and later by Magnus (published posthumously in 1853). These forces were first modeled with an equation by Kutta and Joukowski (1906). This force is usually called the Magnus force.

The direction of this Magnus force can be determined with a pair of right-hand rules: an *angular right-hand rule* and a *coordinate right-hand rule*. The spin axis of the pitch can be found by using the angular right-hand rule. If you curl the fingers of your right hand in the direction of spin, your extended thumb will point in the direction of the spin axis. This is up and to the left for a right-hander's curveball. The direction of the Magnus force can be described using the coordinate right-hand rule. Point the thumb of your right hand in the direction of the spin axis (as determined from the angular right-hand rule), and point your index finger in the direction of forward motion of the pitch. Bend your middle finger so that it is perpendicular to your index finger. Your middle finger will be pointing in the direction of the spin-induced force, down and to the left for a right-hander's curveball. The spin-induced deflection force will be in a direction stated mnemonically as **S**pin **a**xis × **D**irection = **S**pin-**i**nduced **d**eflection (SaD Sid).

The forces acting on the ball during its flight to the plate can be expressed with the following three equations. The force of gravity is downward, $F = mg$, where *m* is the mass of the ball and *g* is the gravitation constant: the magnitude of this force is the ball's weight, 0.32 lbs. The magnitude of the force opposite to the direction of flight is $F_{\text{drag}} = 0.5\rho\pi r_{\text{ball}}^2 v_{\text{ball}}^2 C_{\text{d}}$ where ρ is air mass density, v_{ball} is the ball speed and r_{ball} is the radius of the ball. If English units are to be used then ρ is in lb-s^2/ft^4, v_{ball} is in ft/s, r_{ball} is in ft, F_{drag} is in ft. For the drag coefficient, C_{d}, we use a value of 0.5. The magnitude of the spin-induced force acting perpendicular to the direction of flight is $F_{\text{perpendicular}} = 0.5\rho\pi\omega r_{\text{ball}}^3 v_{\text{ball}}$ where ω is the spin rate is in radians/s: this is often called the Magnus force.

Typical baseball and softball parameters for line drives			
	Major League Baseball	Little League	NCAA Softball
Ball	Hardball	Hardball	Softball
Ball weight (oz)	5.125	5.125	6.75
Ball weight, F_{gravity} (lb)	0.32	0.32	0.42
Ball radius (in)	1.45	1.45	1.9

Ball radius, r_{ball} (ft)	0.12	0.12	0.16
Pitch speed (mph)	85	50	65
Pitch speed, v_{ball} (ft/s)	125	73	95
Distance from front of rubber to tip of plate (ft)	60.5	46	43
Pitcher's release point: (distance from tip of plate, height), (ft)	(55.5, 6)	(42.5, 5)	(40.5, 2.5)
Bat-ball collision point: (distance from tip of plate, height), (ft)	(3, 3)	(3, 3)	(3, 3)
Air weight density, (lb/ft^3)	0.074	0.074	0.074
Air mass density, ρ (lb-s^2/ft^4)	0.0023	0.0023	0.0023
Bat	Wooden C243	Aluminum	Aluminum
Bat weight (oz)	32	23	25
Maximum bat radius (in)	1.25	1.125	1.125
Speed of sweet spot (mph)	60	45	50
Coefficient of restitution	0.54	0.60	0.52
Backspin of batted ball (rps)	10 to 70	10 to 70	10to 70
Backspin of batted ball, ω (rad/s)	63 to 440	63 to 440	63 to 440
Desired ground contact point from the plate (ft)	90 to 200	70 to 150	70 to 160

Typical values for major league pitches				
Type of pitch	Initial Speed (mph)	Spin rate (rpm)	Spin rate (revolutions per second)	Rotations between pitcher's release and the point of bat-ball contact
Fastball	85 to 95	1200	20	8
Slider	80 to 85	1400	23	10
Curveball	70 to 80	2000	33	17
Palmball (a changeup)	60 to 70	400	7	4
Knuckle ball	60 to 70	30	½	¼