

10 Steps to Starting a Successful FPA Program

By Wayne Wild, CFPS

The Problem

“The average software project is late and expensive. Software projects estimated at 1 year and \$1 million will be delivered in 2.2 years and cost \$1.9 million. After all of that, it will only have 61% of the originally specified functionality.” (Standish, 2002)

StoneHenge was off to a great start in 2007. Our largest software development project in quite some time had started in February. We were creating a new Order Fulfillment application for a large international client. This project was sized using our traditional method of estimating (a couple late nights over pizza, breaking down the work elements). We assigned a team of 6 people to work 5 or 6 months utilizing the waterfall method of development, to create the new application. After participating in the project estimation process, Mike Fletcher, StoneHenge President, realized that we needed a better method to size and scope our projects. He invited Lonnie Franks, Vice President of Corporate Quality, for Lighthouse Technologies, Inc. to conduct an off-site 3 day Function Point Analysis training session. The FPA training was conducted in April 2007 and it included key members of the Order Fulfillment Project Team as well as other technical and support staff. During the April FPA training session we counted the FP's associated with the original Order Fulfillment use cases and estimated the project to be 386 FP's. By May 2007 project scope had increased significantly due to new functionality requested by the client. Our Project Manager submitted a change order to lengthen the timeline and project cost by 50%. This change order caught the Client by surprise. Since we did not use FPA to originally estimate the size of the project we had very little documentation that clearly identified what was in our original project estimate and what was not. The client insisted that some of these newly discovered requirements had been mentioned in the early scoping and requirement gathering meetings and therefore, in their opinion, they were not new. Four of our project team members, over a weekend, conducted a FP recount and the size of the project had actually increased not by 50% but by 260% to 1,002 FP's. In July, StoneHenge promoted a current team member, Matt Peters (who was part of the group who participated in the FP recount), to the technical lead role. Under his direction we divided the remaining functionality into eight iterations. Stonehenge then doubled the size of the original development team and they worked extended hours until we delivered all code associated with the 260% increase in scope, within 150% of the original time line. After code delivery, our Client ran into systems integration testing problems due to communications issues regarding the data model, backend systems and interfaces to other applications. As a result the project never implemented. From a financial and customer relations standpoint the project was a loss. Both StoneHenge and the client terminated employees that were members of the original project team.

Does any of this sound familiar? If industry statistics are correct, chances are you too have experienced a similar result at some point in your career.

The Solution

“A clever person solves problems. A wise person avoids them.” (Albert Einstein)

I participated in Lonnie’s FPA101 class in April 2007 and then again in May 2007, per his suggestion, I purchased and read “Function Point Analysis – Measurement Practices for Successful Software Projects” (Garmus Herron, 2001). They state, **“Function point analysis is usually successful in organizations in which senior management believes that measurement is necessary to improve and to be competitive.”** This book along with the function point training and mentoring from Lonnie, sold me on the value of function point analysis to successfully size projects and to manage them. In June, I joined the International Function Point User’s Group (IFPUG) and read the Counting Practice Manual and ordered two Case Study books to use to practice counting FPs and I started my pursuit of becoming a Certified Function Point Specialist (CFPS). Even though I was not a member of the Order Fulfillment project team, mentioned above, I became very alarmed when the four team members’ recount showed the project size had grown by 260%. So in August 2007, after witnessing the impact the Order Fulfillment project had on our organization, I proposed that we utilize the remainder of the project as a FPA proof of concept project to StoneHenge management. We would use function point analysis to size the last four iterations of the project and then we would track our actual development time and defects associated with these iterations and compare our performance to industry standards. The goal of the project was to gain some firsthand experience with this tool so we would be willing and able to use it in the future. The project was approved.

The proof of concept project results are as follows:

Iteration #	<u>5</u>	<u>6</u>	<u>7</u>	<u>8</u>	<u>Total</u>	
# of Use Cases	6	5	5	5	21	
FP Count	143	154	131	101	529	
Est. Dev. Hrs.	758	818	698	539	2813	
Actual Dev. Hrs.	691	519	559	909	2678	95% of the Estimate
Est. Defects	75	81	69	53	278	
Actual Defects	78	71	67	91	307	110% of the Estimate

Note: We used a project delivery rate of 14 hr/FP for C# applications and a 38% Development Phase estimate and a 0.525 Defect/FP for Coding Defects found in component testing for our estimates. These industry statistics were obtained from research in various Capers Jones and International Software Benchmarking Standards Group (ISBSG) publicly available reference materials.

Lessons Learned

“You cannot control what you do not measure.” (W. Edwards Deming)

On November 5, 2007 StoneHenge held an internal project post mortem review meeting to discuss the lessons learned. In addition to the impressive proof of concept results above, the 1,002 revised function point count from the Order Fulfillment project, when multiplied by the Value Adjustment Factor and the industry average 14 hrs/FP (for C# programming language) the estimated total project hours was 16,156, which was within 8% of our actual total effort. Below is a list of the five primary lessons learned from the Order Fulfillment project above.

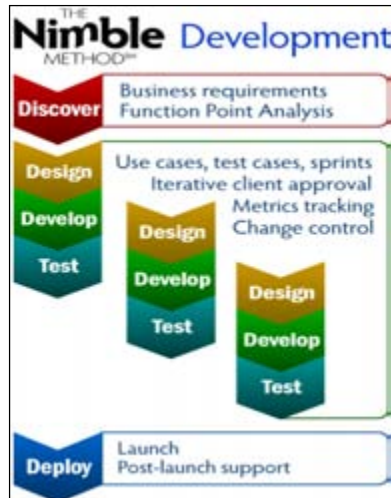
1. Establish project tracking metrics upfront, including a data gathering methodology, to track development activity and defects.
2. Develop and deliver code in usable iterations of approximately 3 weeks in duration.
3. Establish a baseline Function Point Analysis (FPA) once requirements have been defined.
4. Requirement documentation (Use Cases) needs to include a data flow diagram and user interface definitions.
5. Add an independent Scope Manager to the project team to monitor scope creep throughout the software development life cycle (SDLC).

At the conclusion of the meeting, Mike Fletcher challenged us to incorporate FPA into our software development methodology. He didn't want to bid another project without it.

The Results

“Software development is notoriously difficult. The failure rate of software projects is alarmingly high. Quality is the most important topic of software measurement because, historically, the costs of finding and fixing software defects have been the most expensive activity during both software development and maintenance. Therefore, measurement of software projects, and especially key indicators affecting quality and risk, are vital to major corporations.” (Jones, 2008)

After the post mortem meeting above, we went to work and revised our software development methodology to incorporate the use of function point analysis and to develop code in iterations to reduce defects. We call it The NimbleSM Method.



Our preferred billing approach was also changed from time and materials to charging by the function point count associated with the functionality we delivered. Our focus was not only on developing size based metrics, but it was also on reducing defects to insure success. Specifically, we changed our use case documentation process to include a data flow diagram and user interface definitions. We also started to track defects in all phases of the project starting with requirements gathering. In addition, we added an independent Scope Manager as a role on all future software development projects and I became the Metrics/Scope Manager at StoneHenge. This was somewhat of a self-fulfilling prophecy, since I had recommended this role be added to all future teams after reading “Metrics Based Project Governance” (Morris, 2004). Pam Morris’s conference paper makes a clear case for Scope Management, she said, “For IT projects it is about providing the client with the capability to monitor the project status and control the risk of the project not delivering the business value they require within the time and budget available.” She goes on to state, “The Scope Manager provides independent project status reports based on the amount of functionality delivered, versus functionality planned to be delivered, within each reporting period. This is an ‘output based’ metric for project reporting that is more meaningful to the business client than ‘input based’ metrics of budget or effort consumed.”

After implementing our new software development methodology in late 2007, through May 2009, StoneHenge has counted function points for over 20 potential projects. We have won 50% of these project bids. The average size project has been 230 adjusted function points and we have sized projects in the following industries;

- Investigative Services
- Car Rental
- Public Schools
- Industrial Products Supplier
- Energy
- Commercial Property Management

- Healthcare
- Event Promotions
- Airline
- Government Bids
- Food Distribution

Out of these projects, we have completed three C# software development projects where we were responsible for the all phases of the project up to final acceptance testing. Our internal planned project delivery rate ranged from 14 to 15.4 hrs/FP. Below is a summary of the results of these three projects.

Project results using FPA are as follows:

Sequence	<u>1st</u>	<u>2nd</u>	<u>3rd</u>	<u>Average</u>
Adjusted FP Count	103	146	197	
Planned Project Hrs.	1442	2045	3030	
Actual Project Hrs.	1793	1953	3416	
Actual Hrs/Estimated Hrs	118%	96%	113%	110%
Code Delivery vs. Plan	-2 days	On Time	+6 days	+2.67 days
Delivered Defects per FP(*)	0	.014	TBD	

(*) – The number of defects reported within the first 30 days of use of the software divided by the number of Function Points.

The results of these three projects along with the others, have led us to conclude that our estimates are accurate within +/-5%, if we have well defined requirements and a stable familiar environment. Our actual project delivery rate was high on the first project above due to our own learning curve with the new methodology and tools we were using. The second project above was in a very stable familiar environment with no learning curve. The third project above involved new tools and it was a multi-party development effort.

Remember: “The Problem”

“The average software project is late and expensive. Software projects estimated at 1 year and \$1 million will be delivered in 2.2 years and cost \$1.9 million. After all of that, it will only have 61% of the originally specified functionality.” (Standish, 2002)

FPA = Industry Problem Solved!

The Steps

“All told, Everest killed twelve men and women in the spring of 1996, the worst single-season death toll since climbers first set foot on the peak seventy-five years ago. Certainly time had as much to do with the tragedy as the weather, and ignoring the clock can’t be passed off as an act of God. Delays at the fixed lines were foreseeable and eminently preventable. Predetermined turn-around times were egregiously ignored. Extending the turn-around times may have been influenced to some degree by the rivalry between Fischer and Hall. From a business standpoint, there was tremendous pressure to be successful.” (Krakauer, 1999)

Below are the 10 steps StoneHenge used to successfully introduce Function Point Analysis into our organization and software development process. Follow the steps below and you will be able to assist your Company break the “Culture of Disappointment” that surrounds the behind the curtains traditional approach to software development project management. In addition, your Company will be prepared to prevent or mitigate a disaster(project failure event) and protect both you and your client.

10 Steps to Starting a Successful FPA Program:

1. Identify an internal sponsor

Remember, function point analysis is usually successful in organizations in which senior management believes that measurement is necessary to improve and to be competitive.

2. Find a mentor (industry expert)

This person will be able to help you and your management team through the rough spots, until you gain confidence in your own abilities. Also, joining IFPUG is another way to gain access to industry practitioners through posting your questions on their website’s bulletin board.

3. Conduct initial FPA training

A three day (preferably off-site) FPA101 training session is an effect way to introduce the concept to key representatives of multiple levels within the organization from top management down to the developers. Following this up with a general orientation session for all levels within a software development organization will create a general understanding of FPA and how using it will lead to improvement and possible changes within your software development process.

4. Select internal FPA process owners

An organization needs to have at least one person, who is passionate enough to take the initiative to learn and apply the material, as their FPA champion. Then the organization also needs to select at least one backup FP counter, to insure the continuity and availability of this core competency skill set.

5. Invest in on-going FP training

In my experience, the FP Champion should be spending at least 33% of their time counting FPs, training, Scope Managing or in FPA research, in order to keep their skills sharp. They should also be encouraged to become a Certified Function Point Specialist (CFPS), especially now that the exam is available online for \$250. The backup/additional FP counter(s) should also be spending at least 10% of their time counting FPs or in training, as well. There is a fairly significant upfront training investment (80 to 120 hours to read the IFPUG Counting Practice Manual and complete a few practice counts), that will be lost in a few months of inactivity, so these roles need to have an on-going commitment from both the organization and the employee. If you don’t use your FP counting skills, you lose them. All counters should be certified/qualified. Qualifying a

counter may be done internally by setting an accuracy goal and testing your counters to insure they meet your expectations. Below is a minimum FPA budget for our operation.

FPA Annual Budget:

a. FPA Personnel (.43 x \$104,000/yr per Empl.)	\$44,720
b. Backup Counter Training Review (80hr/yr)	\$4,160
c. Corporate IFPUG Membership	\$700
d. Organizational Training (15 people, 3 days)	\$21,200
e. Reference/Training Materials	<u>\$2,500</u>
Total =	<u>\$73,280/yr</u>

6. Create your FPA toolkit

- For under \$5,000 you should be able to buy a couple seats of a good FP counting software package. Benefits include; reduced errors, functional modeling, productivity attributes by component, multiple counts throughout the SDLC and professional FPA reports that can be easily added to any client proposal.
- I set up a SharePoint site for our FPA toolkit that serves as a knowledge repository and includes training materials, reference materials and the documentation of our methodology. This allows the results of my research to be easily shared with others within the organization. Additional key references that I have used but have not mentioned include; www.ISBSG.org, www.TotalMetrics.com and www.DavidConsultingGroup.com. I used some of these sites to download practice CFPS exams, which greatly assisted my preparation for the exam.

7. Identify a proof of concept project

This is a very effective way to solidify the support for your FPA program and to move it to the next level, which is to incorporate it into your Company's software development methodology. It also gives you some experience applying the tool before presenting it to a client.

8. Measure your project delivery rate

It is important to track your time and defects during the proof of concept project so you may develop your own project delivery rate statistics and quality statistics, before you start to quote a \$/FP software development price to a client. Our FP project client pricing rates can range from \$1,000 to \$2,000 per FP due to the complexity of the project, the deliverables, the project's priority level, the environment, and the programming language involved. We had to develop extra time codes that allowed us to track hours charged by project by iteration by role during our proof of concept project. This trial method then became our primary method of reporting project time. The developers didn't even complain about the extra time reporting codes, because they knew that the better we were able to estimate the size of a project the less likely they will be asked to work 60 hour weeks to make up for a bad estimate. Finally, compare your project delivery rate and defects/FP performance to industry data. The www.ISBSG.org website is a great source.

9. Incorporate FPA into your project approach

- Incorporating FPA into our software development methodology changed our sales and marketing approach, project estimating, project pricing, project proposals, requirements gathering and documentation, project time reporting, project planning (our FPA estimates fed into our MS Project Plan and they were also used to divide a project into iterations), defect tracking and project performance tracking (see below).
 - o Scope Traceability Matrix – I created a spreadsheet that lists all of the functional components in each FP count in separate columns throughout the SDLC. This tool is a very effective way to document and communicate scope creep with a client. See Attachment A for an example of this tool.
 - o Project Delivery Rate (PDR) Statistics – I also created a weekly project status report that contains the following; Project Manager comments, comparison of the FP estimate to actual reported project effort, comparison of the planned component delivery dates to their actual completion dates and a comparison of planned defects to actual reported defects. This tool contains objective output based metrics which allows the project and management team know exactly how the project is progressing. It can also identify issues much sooner, which gives you more time to take corrective action. See Attachment B for an example of this tool.
- Whether leading a software development effort or a group of mountain climbers, it is critical that an impartial review of the facts periodically occur in order to be successful. For software development, we believe that is best achieved through the use of a Scope Manager who has objective project metrics available to support his/hers conclusions and recommendations. This person should always consult the Project Manager with their project assessments prior to presenting them to the rest of the management or project team. The Project Manager is still in charge of the project's day to day direction and may be aware of facts unknown to the Scope Manager. However, if the Scope Manager disagrees with the Project Manager's assessment, then they must present their view, as well. For this approach to be effective, it is also important that an organization create an environment where opposing views and open debate of the facts are welcome. Creating this type of an environment can only occur with visible executive management support.

10. Identify prospects

If the industry statistics are true, there should be a lot of prospects that have firsthand experience with software development project failure(s) and/or poor results. Our experience has proved this to be correct. FPA provides a software development company a competitive edge and allows you to differentiate your services. Successfully promoting the use of FPA will likely involve some additional education for your potential client. One way we have addressed this, is by giving our prospect a copy of "Function Point Analysis – Measurement Practices for Success Software Projects" by David Garmus and David Herron, and asking them to read the first 32 pages to gain a better understanding of FPA and how it is able to help them be successful.

Bibliography

Garmus Herron, D. D. (2001). *Function Point Analysis-Measurement Practices for Successful Software Projects*. Addison-Wesley.

Jones, C. (2008). *Applied Software Measurement-Global Analysis of Productivity and Quality*. McGraw-Hill.

Krakauer, J. (1999). *Into Thin Air - A Personal Account of the Mount Everest Disaster*. New York: Anchor Books.

Morris, P. (2004). Metrics Based Project Governance. IWSM Software Measurement Conference.

Standish. (2002). *Survey*.



Wayne Wild, CFPS, is a Senior Consultant/Metrics Manager at StoneHenge Partners, Inc. He received a BS degree in Industrial Engineering from Kansas State University in 1980. He has over 20 years of manufacturing and operations experience including implementing ERP systems, BCP, business performance management and value stream mapping. Since August 2007, Wayne's primary focus has been on software development project requirements gathering, sizing projects using function point analysis, and scope management. He has developed project tracking tools that utilize function point analysis to evaluate an active project's delivery progress and scope. Wayne has counted more than twenty projects over a wide range of industries and applications. The largest was 1,055 adjusted function points. FPA is the primary tool that has allowed StoneHenge Partners to repeatedly deliver successful software development projects that average within 5% of the originally estimated overall effort.

Attachment A – Scope Traceability Matrix

Client: XYZ Company		SH Project #: XYZ100008		Abbreviations:		EI - External Inquiry		DET - Data Element Types	
Project Name: City Sites		SH President: Mike		EO - External Output		EQ - External Inquiry		FTR - File Types Referenced	
Project Sponsor: Tony		SH Business Development Manager: Tim		EIF - External Interface File		ILF - Internal Logical File		RET - Record Element Types	
Project Owner: Brett								FP - Function Points	
								UC - Use Case	
Component Traceability Number	Functionality Description	Count #:	1	Count #:	2	Count #:	3	Count #:	
1.1.00	View Our Cars	Date:	11/9/2007	Date:	12/4/2007	Date:	12/10/2007	Date:	
1.2.01	View Our Cars	Reason:	Proposal Estimate	Reason:	Requirements	Reason:	Iteration #1	Reason:	
1.3.00	View Our Cars	Type	EIF	Type	EIF	Type	EIF	Type	
1.4.01	View Our Cars	FP's	4	FP's	6	FP's	6	FP's	
1.5.01	View Our Cars	DET	>5	DET	21	DET	21	DET	
2.1.00	View Location Map Page	FTR/RET	>50	FTR/RET	46	FTR/RET	46	FTR/RET	
2.2.00	View Location Map Page	# of	1	# of	7	# of	10	# of	
2.3.00	View Location Map Page	EIF	<51	EIF	5	EIF	5	EIF	
3.1.00	Location Things To Do	EIF	<51	EIF	4	EIF	4	EIF	
3.2.01	Location Things To Do	EIF	>19	EIF	29	EIF	29	EIF	
3.3.00	Location Things To Do	EIF	>19	EIF	12	EIF	12	EIF	
3.4.00	Location Things To Do	EIF	>19	EIF	<51	EIF	<51	EIF	
4.1.00	View Location Weather	EQ	>19	EQ	28	EQ	28	EQ	
4.2.00	View Location Weather	EIF	<20	EIF	18	EIF	18	EIF	
5.1.00	View SEO/Location Home Pg	EQ	>5	EQ	20	EQ	20	EQ	
5.2.00	View SEO/Location Home Pg	EIF	<51	EIF	7	EIF	7	EIF	
5.3.00	View SEO/Location Home Pg	EIF	<51	EIF	4	EIF	4	EIF	
6.1.00	View Other Page	EIF	>19	EIF	40	EIF	40	EIF	
		FP's	10	FP's	7	FP's	7	FP's	
		TOTAL =	71	TOTAL =	95	TOTAL =	95	TOTAL =	95
				FP Change =	24	FP Change =	0	FP Change =	0
				% Change =	33.80%	% Change =	0.00%	% Change =	0.00%
Component Change Comments									
Component Traceability Number	Date of Count	Change Approval Status	Comment Explaining the Reason for the Change						
1.2.01	12/3/2007	CO-01 Approved	SCOPE CHANGE - Sitecore DB complexity increased due to the actual number of RETs to allow the application to define templates for each page.						
1.5.01	12/3/2007	CO-01 Approved	DELETED - Didn't realize that the Secondary Nav Link is created by referencing the SitecoreDB & Locations files.						
3.2.01	12/3/2007	CO-01 Approved	CHANGE - Adding Secondary & Events wCities files decreased the complexity of the Primary wCities file because the number of RETs is less.						
3.3.00	12/3/2007	CO-01 Approved	ADD - During the requirements gathering process, Matt identified Secondary wCities file as a separate file.						
3.4.00	12/3/2007	CO-01 Approved	ADD - During the requirements gathering process, Matt identified Events wCities file as a separate file.						
5.1.00	12/3/2007	CO-01 Approved	ADDITIONAL FUNCTIONALITY REQUESTED - Layout for the SEO Home pg is now in scope. Initially this was assumed to be out of scope.						
5.2.00	12/3/2007	CO-01 Approved	ADDITIONAL FUNCTIONALITY REQUESTED - Layout for the Location Home page is now in scope. Initially this was assumed to be out of scope.						
5.3.00	12/3/2007	CO-01 Approved	ADDITIONAL FUNCTIONALITY REQUESTED - Deals is an external interface file required on SEO/Location Home pages						
6.1.00	12/3/2007	CO-01 Approved	ADD - Requirements review revealed that Free Form content is a data element within the DB therefore this is a unique external inquiry.						

Attachment B – Project Delivery Rate (PDR) Statistics

OVERALL ENROLLMENT STATUS COMMENTS:											
Project Effort/Budget Review:	Summary Comment: Verbal approval received on 12/27 to proceed with the Project. Requirements Definition started on January 7th. The SOW was signed on 2/15. The FPA was revised on 3/7/08 to include 3 RMT methods requested by Sarah. The Requirements Spec was signed by client on 3/4. Client's Test server & code became available for testing on 3/25. IT# 3 Dev. was completed on 4/15. IT #3 testing is complete with 4 issues remaining. Client agreed to accept Code Delivery on 5/6, while acknowledging the 4 remaining testing issues documented in the Quality section below.										
Project Schedule Review:	Summary Comment: Initial Development is completed. Testing is completed. Final code review is scheduled for 5/6. Pending results of Code Review & Client's response to Matt's request to address the four remaining quality issues during the Warranty period. Code was Delivered on 5/7 along with an invoice. The warranty will be extended to 180 days after Code Delivery so it will run until 11/6/08.										
Project Quality Review:	Summary Comment: All bugs have been fixed except the 12 open tickets related to one of the four issues listed in the Quality section below. We need to test some functionality after code delivery for us to complete our Testing (See Functionality Below marked as "TS" Code Status).										
Project Effort/Budget Review:											
FPA Summary:											
Date of Count - 3/28/08	Unadj.FP's	VAE	Adj.FP's	Hrs/FP	Cost %	Total Hrs					
	127	1.15	146.1	14		2044.7					
Effort Estimate by Phase:											
		FPA		Actual Report Effort by Phase (As of 5/10/08)							
		Estimated	Actual	Actual	Estimated						
Phase/Activity Code	Effort %	Total Hrs	Phase Hrs	Effort %	Total Hrs	% of Est. Hrs Remaining (*) - See Details Below					
Planning (PM)	16%	2044.7	327.2	14%	280.5	85.7%					
Requirements (RQ, UC)	19%	2044.7	388.5	23%	447.25	115.1%					
Design (DS)	7%	2044.7	143.1	7%	143.5	98.9%					
Development (DV, FF)	43%	2044.7	879.2	32%	629.25	71.6%					
Test (TS)	15%	2044.7	306.7	27%	454.5	148.2%					
	100%	2044.7	2044.7	100%	1563	95.9%					
Project Schedule Review:											
Major Project Element Planned Schedule Actual Status											
Phase 1											
Initiation	12/27-2/15	Verbal approval Received from Client to start project. SOW presented to Client on 2/5 and signed on 2/15.									
Project Planning	12/27-3/10	Project Plan updated on 3/10									
Capture Requirements	1/7-2/27	Requirements Document was signed on 3/4/08									
Produce UC's	1/21-3/10	All UC's have been approved as of 3/13/08									
Phase 2 - DEVELOPMENT SCH TO STARTED 3/3/08											
Component	Function	Function	Deliver	Status (Codes: NS-Not Started, IP-In Process, TS-Testing, Cmpl-Completed)							
Number	Parent UC Description	Functionality Description	Type	Code	Status Comment						
Enrollment	IMT Update Profile/PWD	EQ	EQ	TS	Iteration 1 (Dev Sch: 2/29 - 3/27) ... Can't Test Email						
Enrollment	IMT Enroll Member	EQ	EQ	TS	Iteration 1 (Dev Sch: 2/29 - 3/27) ... Can't Test Email						
Enrollment	IMT Reset PWD	EQ	EQ	Cmpl	Iteration 1 (Dev Sch: 2/29 - 3/27)						
Enrollment	Screen Level Help	EQ	EQ	Cmpl	Iteration 1 (Dev Sch: 2/29 - 3/27)						
Enrollment	Modify Profile Form	EI	TS	Iteration 1 (Dev Sch: 2/29 - 3/27)	Caching & CD File Issues						
Enrollment	Enrollment Form	EI	TS	Iteration 1 (Dev Sch: 2/29 - 3/27)	Nightly Processing Issue						
Enrollment	Upgrade Form	EI	Cmpl	Iteration 1 (Dev Sch: 2/29 - 3/27)							
Enrollment	Member ID Msg	EQ	Cmpl	Iteration 1 (Dev Sch: 2/29 - 3/27)							
Enrollment	CMS	EIF	Cmpl	Iteration 1 (Dev Sch: 2/29 - 3/27)							
Enrollment	Profile Audit Log	ILF	Cmpl	Iteration 1 (Dev Sch: 2/29 - 3/27)							
Enrollment	Profile	ILF	Cmpl	Iteration 1 (Dev Sch: 2/29 - 3/27)							
Enrollment	Log In	EQ	TS	Iteration 2 (Dev Sch: 3/26 - 4/10)	Random SQL Timeout Errors on Login						
Enrollment	Lost ID Page	EQ	Cmpl	Iteration 2 (Dev Sch: 3/26 - 4/10)							
Enrollment	Select Home Page	EQ	Cmpl	Iteration 2 (Dev Sch: 3/26 - 4/10)							
Enrollment	Your Pending Reservations	EQ	Cmpl	Iteration 2 (Dev Sch: 3/26 - 4/10)							
Enrollment	Reservation Search	EQ	Cmpl	Iteration 2 (Dev Sch: 3/26 - 4/10)							
Enrollment	Your Pop-up Agreement	EQ	Cmpl	Iteration 2 (Dev Sch: 3/26 - 4/10)							
Enrollment	Corp Discount	EIF	Cmpl	Iteration 2 (Dev Sch: 3/26 - 4/10)							
Enrollment	IMT-GetPending RES	EQ	Cmpl	Iteration 2 (Dev Sch: 3/26 - 4/10)							
Enrollment	Reset Member PWD	EI	TS	Iteration 3 (Dev Sch: 4/10 - 4/25)	Caching Issue						
Enrollment	Forgot Password	EQ	Cmpl	Iteration 3 (Dev Sch: 4/10 - 4/25)							
Enrollment	Corp Enrollment Data	EQ	Cmpl	Iteration 3 (Dev Sch: 4/10 - 4/25)							
Enrollment	Email Service	EIF	TS	Iteration 3 (Dev Sch: 4/10 - 4/25)	... Can't Test Email						
Enrollment	Member Name Search	EQ	TS	Iteration 3 (Dev Sch: 4/10 - 4/25)	... Caching & Nightly Processing Issue						
Enrollment	Member ID Search	EQ	TS	Iteration 3 (Dev Sch: 4/10 - 4/25)	... Caching & Nightly Processing Issue						
Enrollment	View Member Audit	EQ	TS	Iteration 3 (Dev Sch: 4/10 - 4/25)	... Caching & Nightly Processing Issue						
Enrollment	URL Corp Data	EIF	Cmpl	Iteration 3 (Dev Sch: 4/10 - 4/25)							
Project Quality Review:											
		Est. # of Defects	Actual # of Defects	Open Ticket Summary:							
		Found	Found	# Assigned = 5	# Feedback = 4	# New/Ack = 3					
				Total = 12							
Quality Check Points:											
RTM Review & Sign-off	4.4	8.0				All Open Tickets Related to one of the following 4 Issues: 1. Caching - Handled different in different environments 2. Corp. Discount - client trying to find correct file. 3. Nightly Processing/Data Structure - Only tested once. 4. Email - Incompl awaiting client					
UC Review & Sign-off	7.4	22.0									
Design Review	61.3	5.0									
Unit Testing	163.6	8.0									
Code Review	58.4	96.0	1 Defects added since last week								
Component Testing	54.0										
Func. Tst/Doc. Review	40.9										
Total	460.1	139.0									
Hours Reported Summary (Incl Hrs Reported on Weekly Time Reports with Project # 100007 & Activity Codes)											
Employee	Week	PM	RQ	UC	DS	DV	FF	TS	Total	PR	(Non-billable Prep)
	Total =	280.5	186.75	260.5	141.5	455	174.25	454.5	1953	184.75	
(Note: Project became billable on 12/27/07)											
Rachel	1/6-1/12	3.5									
Tom	1/6-1/12		16.5								
Matt	1/6-1/12		11								
Rachel	1/13-1/19	6.5									
Tom	1/13-1/19		8	1.5							
Matt	1/13-1/19		13.5								
Trina	1/13-1/19		21.25								
Rachel	1/20-1/26	8.5		2							
Tom	1/20-1/26		3.5	25							
Matt	1/20-1/26		3	12						2.5	
Trina	1/20-1/26			14							
Wayne	1/20-1/26										17
Rachel	1/27-2/2	10.5									
Tom	1/27-2/2		8	22.5							
Matt	1/27-2/2		9		12.5						
Trina	1/27-2/2		0.5	2.5							
Wayne	1/27-2/2									20	
Bruce	1/27-2/2									6	
Rachel	2/3-2/9	14									
Tom	2/3-2/9		8	23							