Concordia University
Department of Computer Science
and Software Engineering

Touch For Food

Management

SOEN 490
Capstone Project
Fall 2012 – Winter 2013

CloudNine

Josh Hum (Team Leader)	9583157
Katrina Anderson	9106251
Cristian Asenjo	9280014
Christian Daher	9599673
Cynthia Donato	9353852
Mikhail Levkovsky	9583165
Patrick Modafferi	9401377
Ryan Nasr	9605614
Matthew Tam	9675701

Table of Contents

List of Figures

List of Tables

Touch For Food

Management

Version 11.23

Revision History

Date	Rev.	Description	Author(s)
2012-11-03	1.0	Document Creation	Josh Hum
2012-11-04	1.1	Update doc	Josh Hum
2012-12-03	2.2	Added Iteration 2 retrospective	Josh Hum
2012-12-17	3.3	Made correction to Section 3.3	Josh Hum
2012-12-30	4.4	Reducing redundancy	Katrina Anderson
2013-01-08	5.5	Iteration 3 Plan and Report	Josh Hum
2013-01-10	5.6	Iteration 4 Plan and Report	Josh Hum
2013-01-11	5.7	Iteration 5 Plan added and formatting fixing	Katrina Anderson
2013-01-15	5.8	Fixed user story number error	Josh Hum
2013-01-15	5.9	Iteration 0 Report	Josh Hum
2013-01-18	6.10	Iteration 6 Plan	Josh Hum
2013-02-01	6.11	Iteration 6 Report, Analysis, and Retrospective	Josh Hum
2013-02-01	7.12	Iteration 7 Plan	Josh Hum
2013-02-12	7.13	Reviewed Document for submission	Cynthia Donato
2013-02-12	7.14	Iteration 7 Report, Analysis, and Retrospective	Josh Hum
2013-02-18	8.15	Iteration 8 Plan	Josh Hum
2013-02-26	8.16	Iteration 8 Report, Analysis, and Retrospective	Josh Hum
2013-03-01	9.17	Iteration 9 Plan	Josh Hum
2013-03-17	9.18	Iteration 9 Report, Analysis, and Retrospective	Josh Hum
2013-03-18	9.19	Iteration 10 Plan	Josh Hum
2013-03-30	10.20	Iteration 10 Report, Analysis, and Retrospective	Josh Hum
2013-03-31	10.21	Reviewed document	Cristian Asenjo
2013-04-05	11.22	Iteration 11 Plan, Report, Analysis, and Retrospective	Josh Hum
2013-04-05	11.23	Final Project Report	Josh Hum

1 List of User Stories

The following is the list of user stories in the backlog. Stories will be chosen to work on at the beginning of each sprint based on the results of the AHP graph as well as if they are blockers or not.

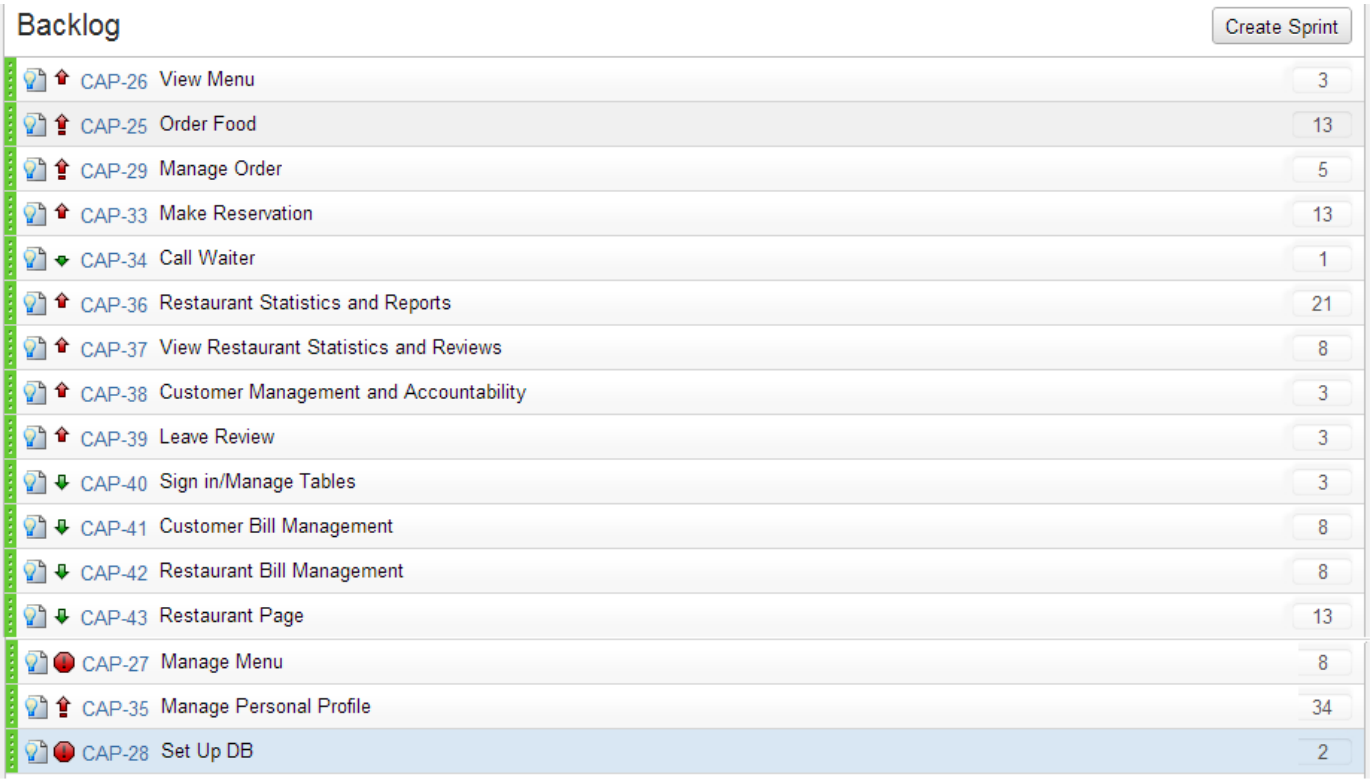

Figure 1-1 Backlog Stories

2 AHP

The AHP method was used to determine priority of user stories. The following graph shows the results of building AHP cost-value matrices. High priority stories are between the y-axis and the first diagonal line. Medium priority stories fall between the two diagonal lines and low priority stories come between the second diagonal line and the x-axis.

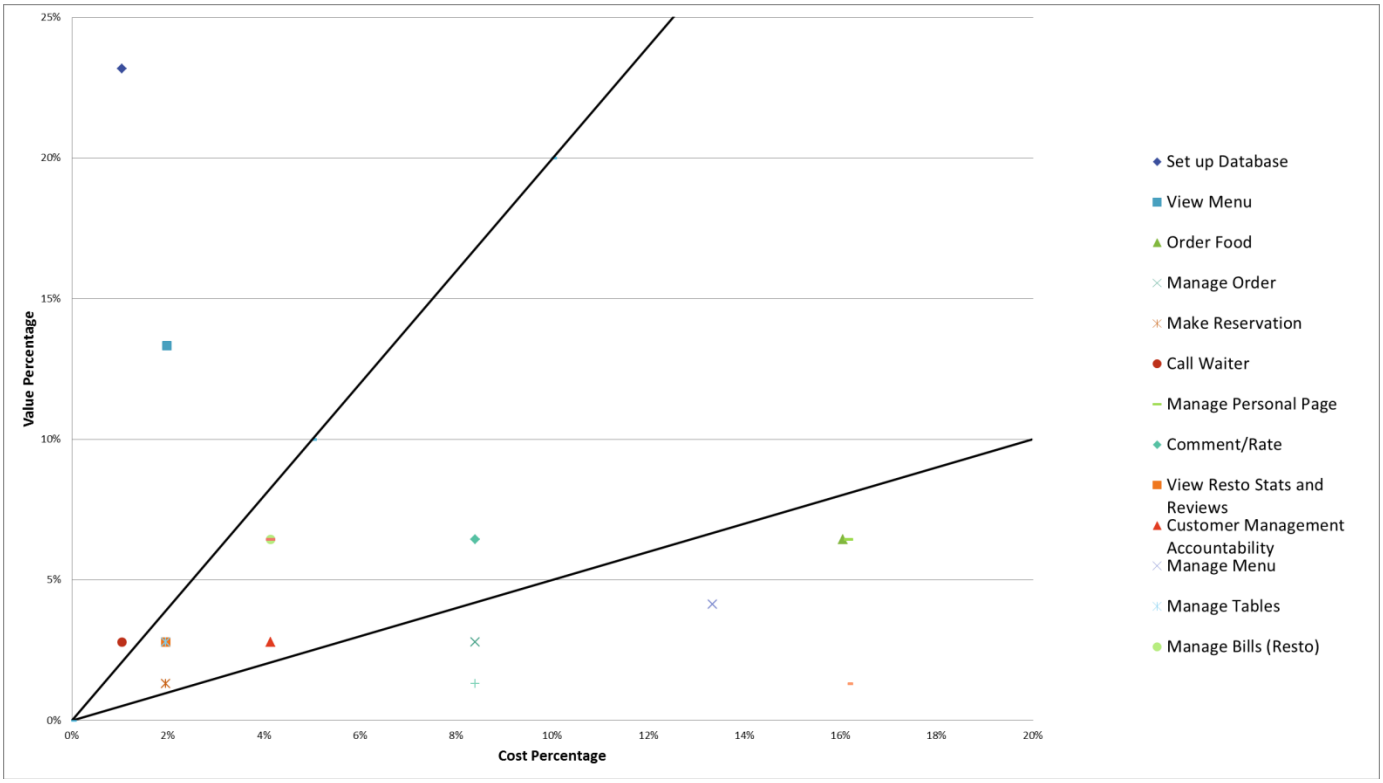

Figure 2-1 AHP Graph

3 Code Quality Goals

We will attempt to meet the following goals with the combination of our code and the Code Metrics Viewer Visual Studio plugin [**1**]. The plugin outputs a report with the following metrics displayed. Average is defined on the report with a yellow background, while an optimal result is shown with a green background. Below average is shown with a red background. Our goals are shown in the table below.

Table 3-1 Code Quality Goals

Metric	Level
Cyclomatic Complexity (CC)	Average (or lower)
Maintainability Index (MI)	Average (or higher)
Class Coupling (CCP)	Average (or lower)
Lines of Code (LOC)	Average (or lower)
Depth of Inheritance (DOI)	Average (or lower)

4 Iteration 0 Report

Iteration 0 was spent doing research, setup, and preparation for development. Research was conducted for configuration management, code hosting and versioning, server and client side technologies, mobile technologies, and databases. We also made decisions on local machine standards for both documenting and code development. Time was also spent setting up our local machines.

4.1 Person-Hour Work Log

The following work log shows hours worked during Iteration 0.

Start Date: 3/Sep/12 End Date: 21/Oct/12 [Change] (UNREGISTERED)				Christian Daher	Cristian Asenjo	Cynthia Donato	Josh Hum	Katrina Anderson	Matthew Tam	Mikhail Levkovsky	Patrick Modafferi	Ryan Nasr	Total
Issue			Total	15.583h	27.133h	23.25h	31.917h	23.333h	15h	19.217h	37.083h	7h	199.517h
CAP-8	Documentation - Proposal		3h	0h	0h	0.75h	0h	0.75h	0h	0h	1.5h	0h	3h
CAP-9	Documentation - Vision Document		6.25h	1.5h	0h	1.083h	0h	1.917h	0h	1.25h	0.5h	0h	6.25h
CAP-10	Documentation - SRS		10h	1h	3.667h	0h	2h	0.583h	0.75h	0h	2h	0h	10h
CAP-11	Documentation - SAD		16.417h	2.5h	2.5h	2h	0h	1.667h	0.75h	3h	4h	0h	16.417h
CAP-12	Documentation - User Manual		0.583h	0h	0h	0h	0h	0.583h	0h	0h	0h	0h	0.583h
CAP-14	Correspondence		25.3h	1.083h	2.05h	1.917h	10.917h	3.583h	1.5h	0.167h	4.083h	0h	25.3h
CAP-18	Meetings		102.55h	9h	15.25h	14h	17h	11.5h	12h	8.8h	8h	7h	102.55h
CAP-23	Setup		19.333h	0.5h	1.333h	0.5h	1h	2h	0h	1h	13h	0h	19.333h
CAP-27	Manage Menu		8h	0h	0h	0h	0h	0h	0h	4h	4h	0h	8h
CAP-28	Set Up DB		5h	0h	0h	3h	1h	0h	0h	1h	0h	0h	5h
CAP-44	Documentation - Test Plan		0.75h	0h	0h	0h	0h	0.75h	0h	0h	0h	0h	0.75h
CAP-45	Set up Visual Studio skeleton project		2.333h	0h	2.333h	0h	0h	0h	0h	0h	0h	0h	2.333h

Figure 4-1 Person-Hour Work Log

5 Iteration 1 Plan

Iteration 1 is the first iteration where coding development began. The period of time before Iteration 1 was used for planning, meetings, setup and configuration of each team member's development environment.

5.1 Planned Activities

The two stories planned for Iteration 1 are CAP-35 – Manage Personal Profile and CAP-27 – Manage Menu. We chose to start with them because they provide a foundation for other user stories to be developed.

Table 5-1 Planned Activities

User Story ID	Total Story Points	Related Use Cases
CAP-35, CAP-27, CAP-28	44.00	UC1.1, UC2.5

5.2 Person-Hour Estimation

The Expert Judgement Method was used to calculate person-hour estimations. Estimations were made for the worst case, most likely case, and best case. The worst case and best case are +30% and -30%. The expected case was then calculated with the formula:

Expected case = 1/6([worst case] + 4[most likely case] + [best case])

Table 5-2 Person-Hour Estimation

User Story ID	Worst Case	Most Likely Case	Best Case	Expected Case
US35	110	90	75	90.8
US27	30	20	15	20.8
US28	10	7	3	6.8
Total (ph)	**150**	**117**	**93**	**118.4**
Velocity (ph/day)	**10.7**	**8.4**	**6.6**	**8.5**
Velocity (ph/team member/day)	**1.2**	**0.9**	**0.7**	**0.9**

6 Iteration 1 Report

6.1 Person-Hour Work Log

The following table shows the person-hour work log for each team member and activity during this iteration. This table was generated from the JIRA management system.

Start Date: **22/Oct/12** End Date: **4/Nov/12** [Change] (UNREGISTERED)

Issue			Total	Christian Daher	Cristian Asenjo	Cynthia Donato	Josh Hum	Katrina Anderson	Matthew Tam	Mikhail Levkovsky	Patrick Modafferi	Ryan Nasr	Total
			6.5h	**12.783h**	**19.667h**	**32h**	**11.75h**	**4.75h**	**6.833h**	**11.5h**	**0.092h**	**105.875h**
CAP-8	Documentation		**0.5h**	0h	0h	0h	0h	0h	0h	0.5h	0h	0h	**0.5h**
CAP-11	CAP-8 SAD		**3.5h**	0h	0h	0h	0h	0h	0h	1h	2.5h	0h	**3.5h**
CAP-15	CAP-14 Email		**13.333h**	0h	0.083h	1.25h	6.5h	4h	0.5h	0h	1h	0h	**13.333h**
CAP-16	CAP-14 Phone		**3.417h**	0h	0h	0h	2.5h	0.917h	0h	0h	0h	0h	**3.417h**
CAP-17	CAP-14 Google Groups		**0.667h**	0h	0.083h	0h	0h	0h	0h	0.583h	0h	0h	**0.667h**
CAP-18	Meetings		**1h**	0h	0h	1h	0h	0h	0h	0h	0h	0h	**1h**
CAP-19	CAP-18 Weekly Group Meeting		**6.75h**	0h	0h	1h	1h	0h	2.75h	0h	2h	0h	**6.75h**
CAP-23	Setup		**7.5h**	0h	0.5h	3h	0h	0h	0h	0h	4h	0h	**7.5h**
CAP-24	CAP-8 Proposal		**0.5h**	0h	0h	0h	0.5h	0h	0h	0h	0h	0h	**0.5h**
CAP-26	View Menu		**4.867h**	0h	4.867h	0h	0h	0h	0h	0h	0h	0h	**4.867h**
CAP-27	Manage Menu		**0.05h**	0h	0h	0h	0h	0h	0h	0h	0h	0.05h	**0.05h**
CAP-44	CAP-8 Test Plan		**14.042h**	4.5h	0h	1h	1.75h	3h	1h	0.75h	2h	0.042h	**14.042h**
CAP-47	CAP-27 Menu Editor Main Page		**2h**	0h	0h	0h	0h	0h	0h	2h	0h	0h	**2h**
CAP-53	DB structure for menu, item and category is faulty.		**2h**	0h	0h	0h	0h	0h	0h	2h	0h	0h	**2h**
CAP-56	CAP-18 Other (GoogleHangouts/Mini Team Meetings/Emergency Meetings/SCRUM)		**9.333h**	2h	1.75h	1h	1.75h	2.833h	0h	0h	0h	0h	**9.333h**
CAP-57	CAP-8 Management Docs		**18h**	0h	0h	0h	17h	1h	0h	0h	0h	0h	**18h**
CAP-59	CAP-35 SAD 4.3 Login Page Use Case		**0.5h**	0h	0h	0h	0h	0h	0.5h	0h	0h	0h	**0.5h**
CAP-60	CAP-35 Customer Settings Page Programming		**7.5h**	0h	0h	7.5h	0h	0h	0h	0h	0h	0h	**7.5h**
CAP-61	CAP-35 SAD 4.3 Customer Settings Page Use Case		**0.417h**	0h	0h	0.417h	0h	0h	0h	0h	0h	0h	**0.417h**
CAP-62	CAP-35 Customer Sign-up Page Programming		**5h**	0h	5h	0h	0h	0h	0h	0h	0h	0h	**5h**
CAP-63	CAP-35 SAD 4.3 Customer Sign-up Use Case		**0.5h**	0h	0.5h	0h	0h	0h	0h	0h	0h	0h	**0.5h**
CAP-64	CAP-35 Customer Profile Page View Programming		**0.5h**	0h	0h	0.5h	0h	0h	0h	0h	0h	0h	**0.5h**
CAP-65	CAP-35 SAD 5.3 Class Diagram		**3h**	0h	0h	3h	0h	0h	0h	0h	0h	0h	**3h**
CAP-66	CAP-35 SAD 4.1 & 4.2 Update		**1h**	0h	0h	0h	1h	0h	0h	0h	0h	0h	**1h**

Figure 6-1 Person-Hour Work Log

6.2 Cumulative Velocities vs. Time

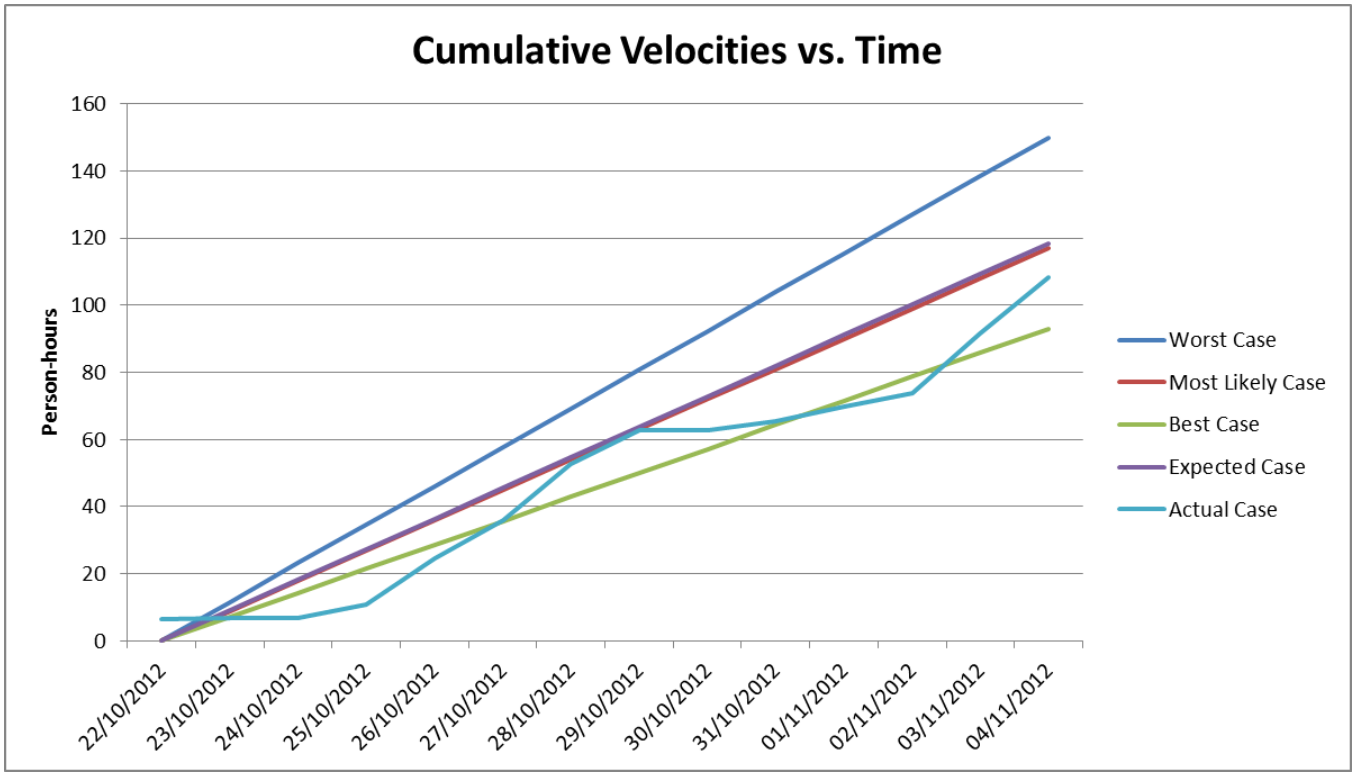

Figure 6-2 Cumulative Velocities vs. Time

6.3 Cumulative Flow Diagram

The following diagram shows the three stories and how the work progressed. Blue tasks have not been started, purple tasks are in progress and green represents a completed story. This diagram was generated in JIRA.

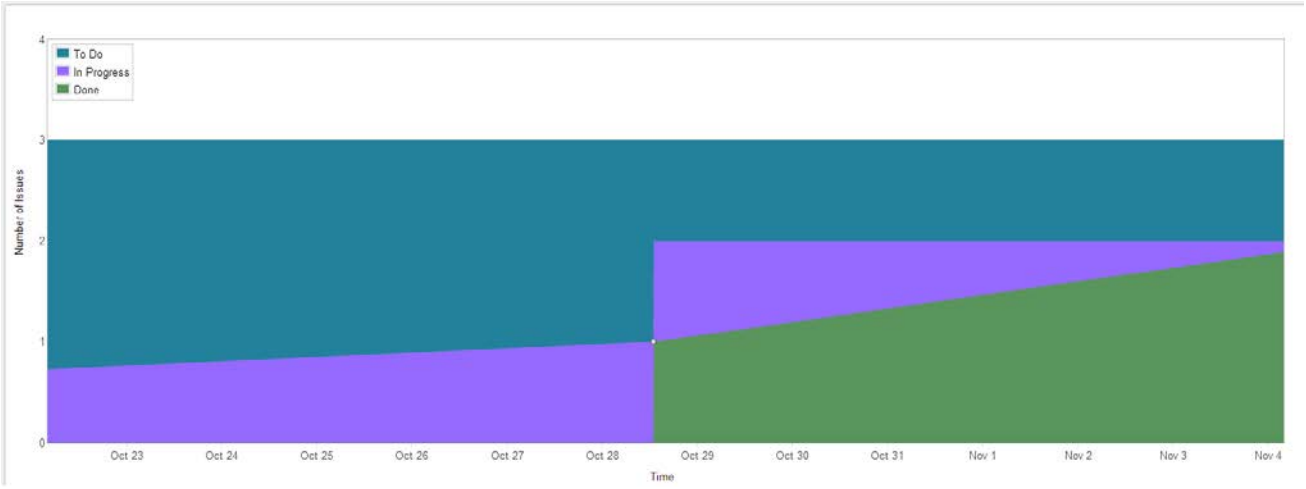

Figure 6-3 Cumulative Flow Diagram

6.4 Measurement Report

6.4.1 Code Quality Analysis

The following report was generated after analyzing the code. All our minimum goals were met while some were exceeded.

Analysis tool used: Code Metrics Viewer
Found at: http://visualstudiogallery.msdn.microsoft.com/9f35524b-a784-4dbc-bd7b-6babd7a5a3b3
Version: 1.5.3
Last updated: 2/5/2012

	Scope	Hiearchy	Maintainability Index	Cyclomatic Complexity	Depth of Inheritance	Class Coupling	Lines of Code
2	ModuleScope	TouchForFood.dll	88	286	58	3	413
3	NamespaceScope	TouchForFood	90	4	9	2	7
4	TypeScope	MvcApplication	90	4	9	2	7
5	MemberScope	Application_Start() : void	80	1	3		3
6	MemberScope	MvcApplication()	100	1	1		1
7	MemberScope	RegisterGlobalFilters(GlobalFilterCollection) : void	94	1	2		1
8	MemberScope	RegisterRoutes(RouteCollection) : void	82	1	3		2
9	NamespaceScope	TouchForFood.Controllers	78	76	36	3	187
10	TypeScope	CategoryController	76	12	12	3	33
21	TypeScope	HomeController	96	2	2	3	3
24	TypeScope	Menu_CategoryController	69	20	29	3	42
35	TypeScope	MenuController	73	16	30	3	38
46	TypeScope	RestaurantController	76	12	12	3	33
57	TypeScope	UserController	75	14	15	3	38
58	MemberScope	Create() : ActionResult	89	1	2		2
59	MemberScope	Create(user) : ActionResult	69	2	8		6
60	MemberScope	Delete(int) : ActionResult	77	1	5		3
61	MemberScope	DeleteConfirmed(int) : ActionResult	71	1	8		5
62	MemberScope	Details(int) : ViewResult	77	1	5		3
63	MemberScope	Dispose(bool) : void	87	1	3		2
64	MemberScope	Edit(int) : ActionResult	77	1	5		3
65	MemberScope	Edit(user, HttpPostedFileBase) : ActionResult	59	4	11		11
66	MemberScope	Index() : ViewResult	83	1	5		2
67	MemberScope	UserController()	92	1	2		1
68	NamespaceScope	TouchForFood.Models	92	206	21	2	219
69	TypeScope	category	92	9	4	1	11
79	TypeScope	item	92	19	6	1	21
99	TypeScope	menu	93	11	5	1	12
111	TypeScope	menu_category	93	13	5	1	14
125	TypeScope	menu_item	93	13	4	1	13
139	TypeScope	order	92	25	8	1	27

Figure 6-4 Code Analysis report

6.5 Retrospective

In Iteration 1, we completed US27 and US28. This amounts to 10 out of the planned 44 story points. Therefore, it was not a successful iteration. The following table shows the completed user stories:

Table 6-1 Completed User Stories

User Story ID	User Story	Story Points
CAP-27	As a restaurant manager, I would like to create, update, delete and view customized menus.	8
CAP-28	As a developer, I would like to set up the database in order to be able to begin building the system.	2

Although the story US35 could not be completed, it was partially done and progress was made. However, for the next sprint, we will adjust the amount of story points we take on based on our current velocity.

Positives:

- Finished the critical story of setting up the database

Negatives:

- Did not complete all the user stories planned
- Could have better communication
- Missed out on certain management metrics

105.9 person-hours were spent working in Iteration 1. This was close to the 118 person-hour estimate. However, it is below half of the initial budgeted person-hours scheduled. We attribute this to two causes. First, this iteration was started while the developers were right in the middle of midterms. We expect to accomplish much more and spend more time working in the following iteration due to less time constraints. Secondly, the team is not yet fully accustomed to the use of the management system. Thus the work times logged are much lower than the actual work times. There seem to be quite a few data entry errors as well. We will look to correct this in the following iteration.

7 Iteration 2 Plan

7.1 Planned Activities

The stories CAP-26 – View Menu and CAP-35 – Manage Personal Profile were initially planned for Sprint 2. However, early in the sprint, bugs were found in CAP-27 – Manage Menu that were blockers, so the story was reopened and added to the sprint.

Table 5-7-1 Planned Activities

User Story ID	Total Story Points	Related Use Cases
CAP-35, CAP-27, CAP-26	45.00	UC1.1, UC3.1, UC4.1, UC4.2, UC4.3

7.2 Person-Hour Estimation

The Expert Judgement Method was used to calculate person-hour estimations.

Table 5-7-2 Person-Hour Estimation

User Story ID	Worst Case	Most Likely Case	Best Case	Expected Case
CAP-35	110	90	75	90.8
CAP-27	30	20	15	20.8
CAP-26	30	20	15	20.8
Total (ph)	**170**	**130**	**105**	**132.4**
Velocity (ph/day)	**12.1**	**15.5**	**7.5**	**9.5**
Velocity (ph/team member/day)	**1.3**	**1.7**	**0.8**	**1.1**

8 Iteration 2 Report

8.1 Person-Hour Work Log

The following table shows the person-hour work log for each team member and activity during this iteration. This table was generated from the JIRA management system. Individual schedules varied so there is a large deviation between hours worked. These hours reflect time spent on all tasks.

Start Date: 6/Nov/12 End Date: 21/Nov/12 [Change] (UNREGISTERED)				Christian Daher	Cristian Asenjo	Cynthia Donato	Josh Hum	Katrina Anderson	Matthew Tam	Mikhail Levkovsky	Patrick Modafferi	Ryan Nasr	Total
Issue			Total	10h	16.917h	7.75h	9.417h	18.917h	7.417h	16.583h	23.25h	8.5h	118.75h
CAP-8	Documentation		1h	0h	0h	0h	0h	1h	0h	0h	0h	0h	1h
CAP-14	Correspondence		10.417h	0h	0.083h	3.25h	0.75h	2.25h	1h	0.083h	3h	0h	10.417h
CAP-18	Meetings		22h	6h	1h	0.5h	2.667h	0.667h	2.167h	2h	1.5h	5.5h	22h
CAP-23	Setup		33h	2h	0h	0h	6h	15h	0h	0h	9h	1h	33h
CAP-26	View Menu		4h	2h	0h	0h	0h	0h	0h	0h	0h	2h	4h
CAP-27	Manage Menu		24h	0h	0h	0h	0h	0h	0.25h	14h	9.75h	0h	24h
CAP-35	Manage Personal Profile		13.5h	0h	5.5h	4h	0h	0h	4h	0h	0h	0h	13.5h
CAP-37	View Restaurant Statistics and Reviews		2.333h	0h	2.333h	0h	0h	0h	0h	0h	0h	0h	2.333h
CAP-39	Leave Review		3h	0h	3h	0h	0h	0h	0h	0h	0h	0h	3h
CAP-77	Restaurants are not connected to any user		0.5h	0h	0h	0h	0h	0h	0h	0.5h	0h	0h	0.5h
CAP-82	Duplicate usernames can be created. Database has to be changed to restrict usernames to be unique.		3h	0h	3h	0h	0h	0h	0h	0h	0h	0h	3h
CAP-83	Confirm password entry isn't working after moving to a well-partial'd class		2h	0h	2h	0h	0h	0h	0h	0h	0h	0h	2h

Figure 8-1 Person-Hour Work Log

8.2 Hour Burndown Chart

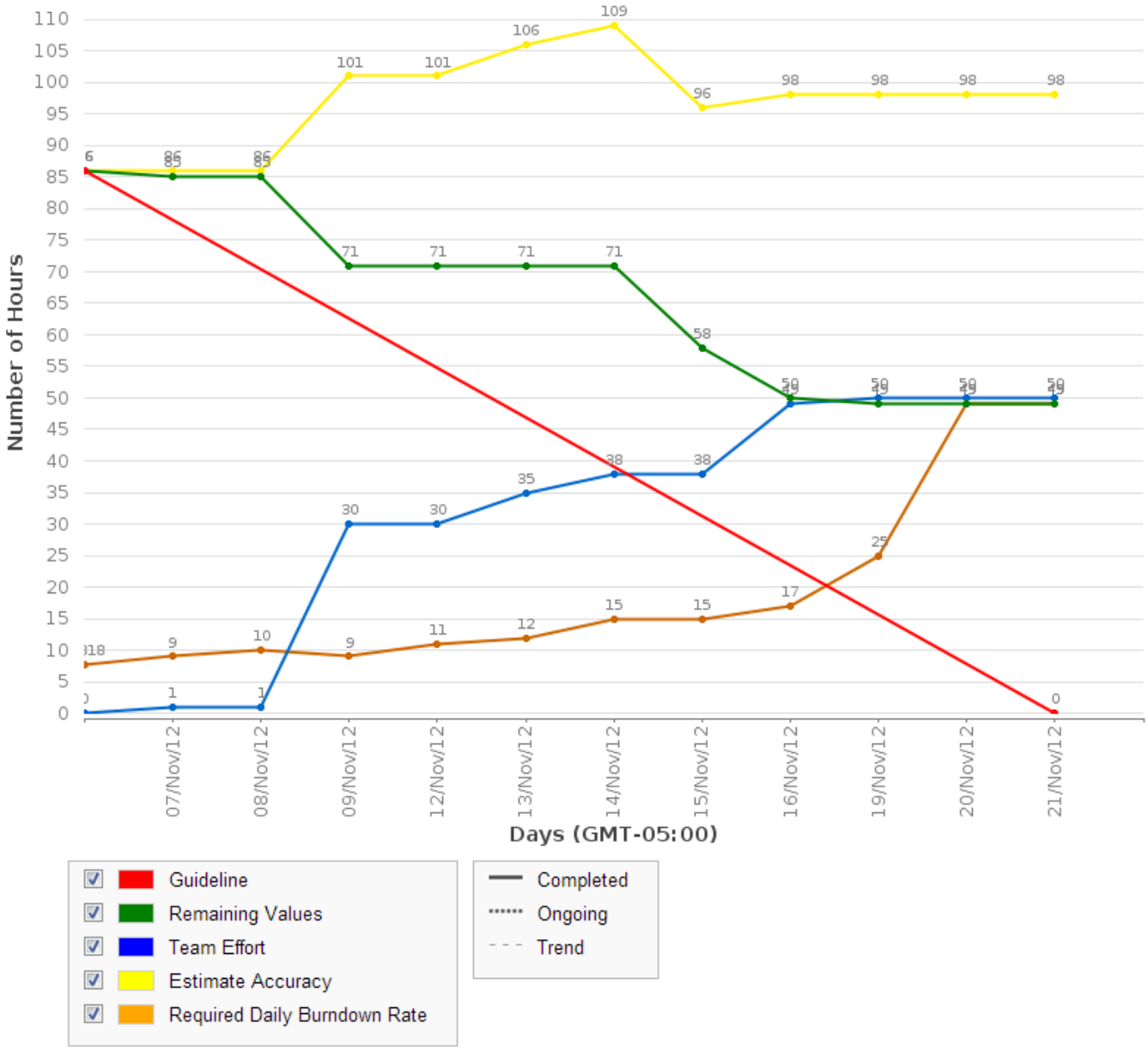

Figure 8-2 Hour Burndown Chart

As seen in the previous chart, the team was not able to burndown everything that was assigned initially in this sprint. We initially estimated 85 hours were needed to complete all the programming tasks. Although the team put 118 hours of work into this iteration, only 50 were spent working on coding tasks as seen by the blue line. The remaining tasks represented by the green line show that there are 49 hours left before all stories can be burned down. We attribute this discrepancy between our total original estimate (85 hours) and the actual work needed (50 + 49 = 99 hours) to overly optimistic estimates. For example, some tasks took much longer than planned so the burndown was much slower than desired. The following image is an example of one such task.

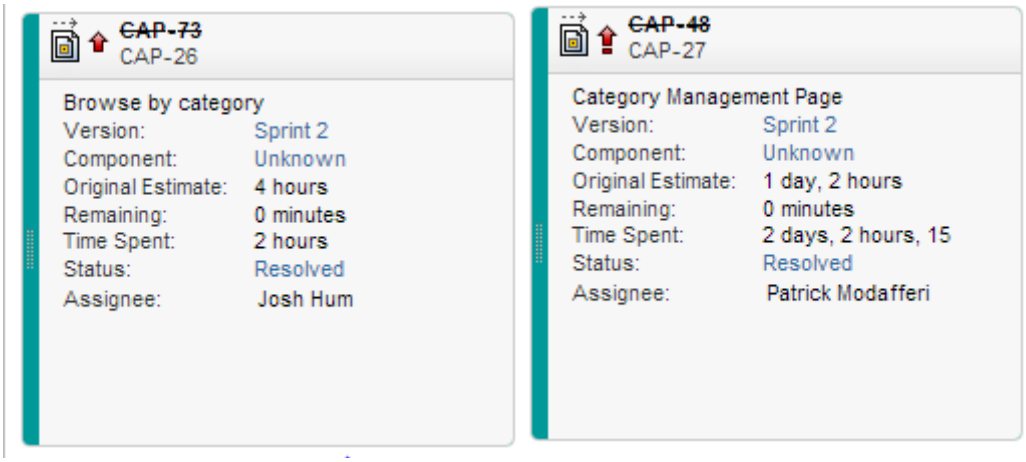

Figure 8-3 Overestimation of CAP-73 and Underestimation of CAP-48

As shown in the previous two figures, there were both overestimations and underestimations made during our sprint planning. However, the discrepancy between the underestimations and the actual time needed were greater than the overestimations which caused the team to fall behind schedule. There were multiple tasks which were underestimated and only two that were overestimated.

8.3 Issue Burndown Chart

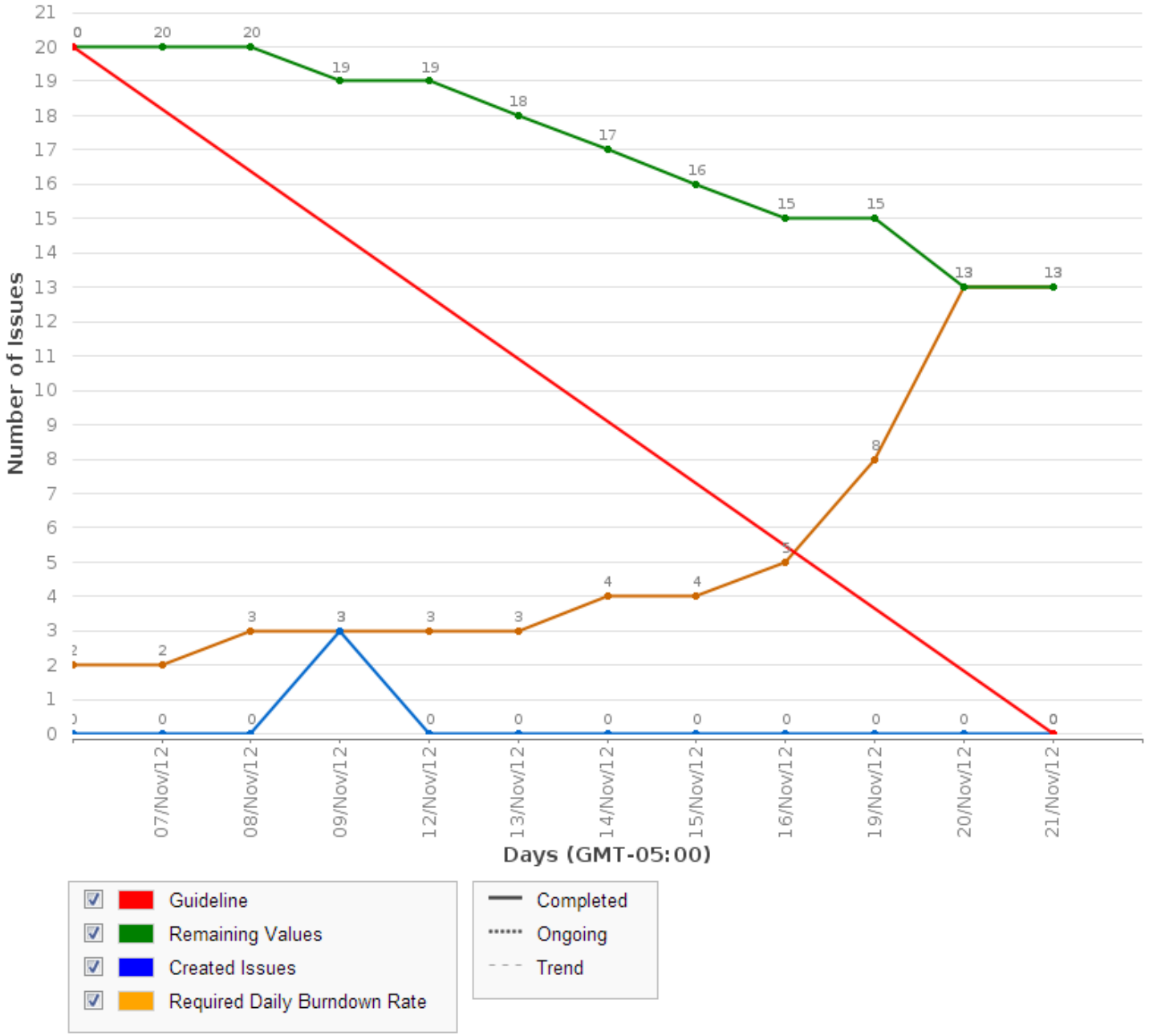

Figure 8-4 Issue Burndown Chart

As seen in the issue burndown chart, 3 tasks were created during the iteration. This was because we overlooked some sub-tasks that were part of the sprint and needed to be added immediately. We will be working to eliminate such mistakes in the future with more detailed planning.

8.4 Cumulative Velocities vs. Time

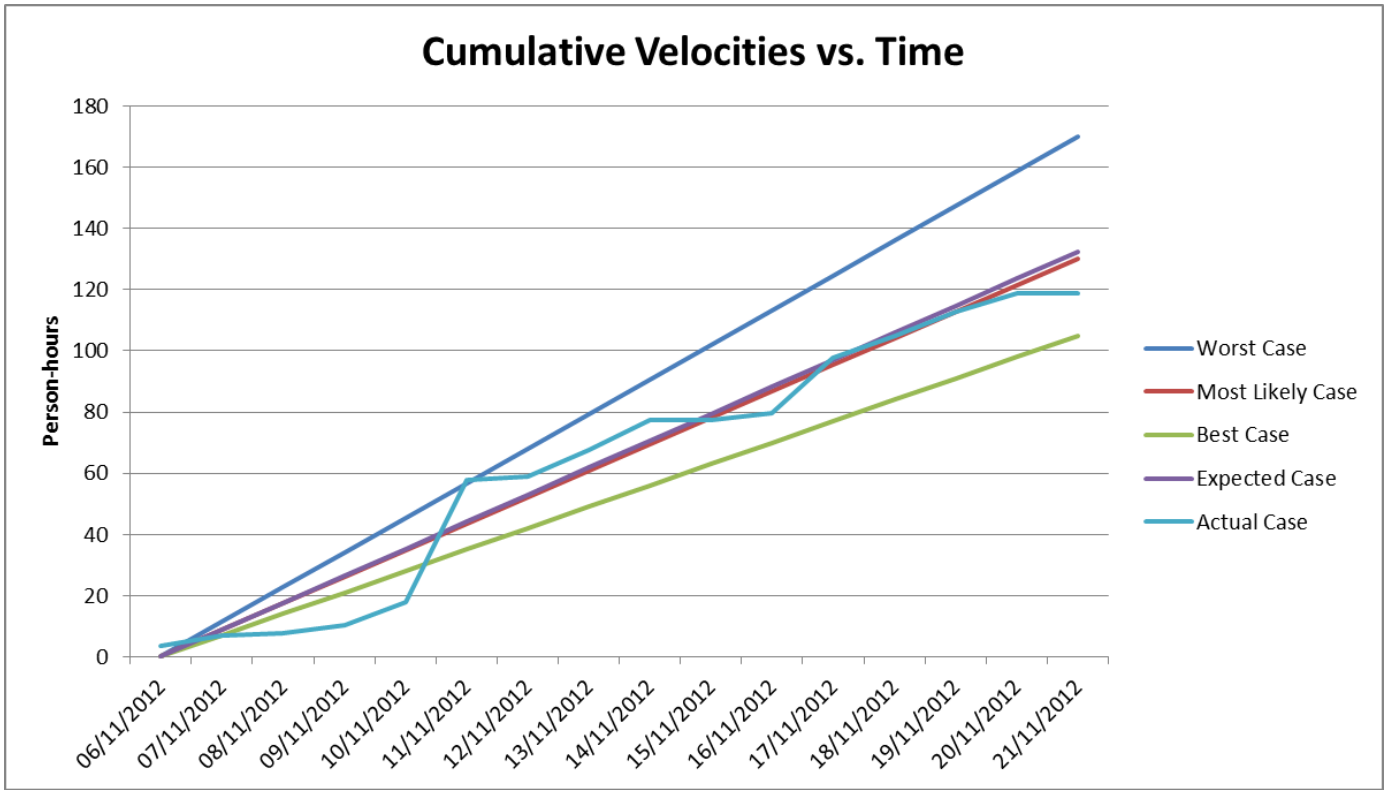

Figure 8-5 Cumulative Velocities vs. Time

8.5 Cumulative Flow Diagram

The following diagram shows the three stories and how the work progressed. Blue tasks have not been started, purple tasks are in progress and green represents a completed story. This diagram was generated in JIRA.

We attribute the increase in blue areas to inaccurate initial time estimations. Therefore, as we increased our time estimations to complete a story, the blue area also increased. The purple area stays relatively the same over the sprint showing that the work in progress stayed relatively constant.

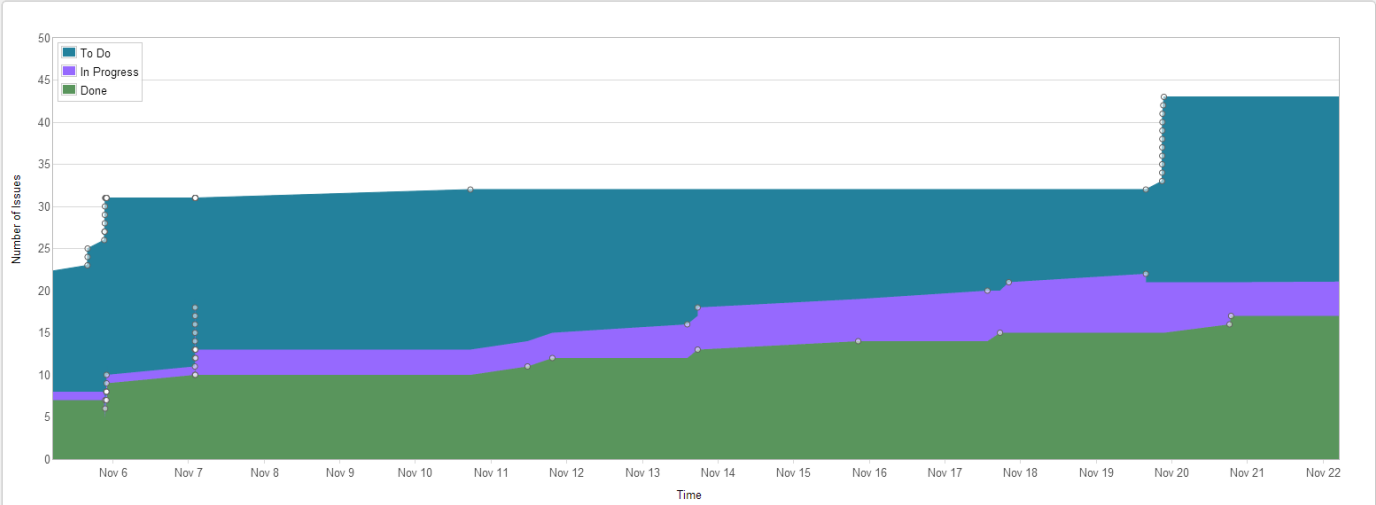

Figure 8-6 Cumulative Flow Diagram

8.6 Measurement Report

8.6.1 Code Quality Analysis

The following report was generated after analyzing the code. Only one method did not meet code quality expectations by exceeding the expected number of lines of code. This will be corrected in future iterations.

Analysis tool used:	Code Metrics Viewer
Found at:	http://visualstudiogallery.msdn.microsoft.com/9f35524b-a784-4dbc-bd7b-6babd7a5a3b3
Version:	1.5.3
Last updated:	2/5/2012

Scope	Hiearchy	Maintainability Index	Cyclomatic Complexity	Depth of Inheritance	Class Coupling	Lines of Code
ModuleScope	TouchForFood.Tests.dll	72	13	20	1	29
ModuleScope	TouchForFood.dll	85	470	95	3	728
NamespaceScope	TouchForFood	81	4	9	2	9
TypeScope	MvcApplication	81	4	9	2	9
MemberScope	Application_Start() : void	80	1	3		3
MemberScope	MvcApplication()	100	1	1		1
MemberScope	RegisterGlobalFilters(GlobalFilterCollection) : void	94	1	2		1
MemberScope	RegisterRoutes(RouteCollection) : void	71	1	3		4
NamespaceScope	TouchForFood.Attributes	84	2	5	3	4
TypeScope	AjaxAttribute	84	2	5	3	4
MemberScope	AjaxAttribute(bool)	87	1	1		2
MemberScope	IsValidForRequest(ControllerContext, MethodInfo) : bo	83	1	4		2
NamespaceScope	TouchForFood.Controllers	71	148	60	3	369
TypeScope	CategoryController	76	13	16	3	36
TypeScope	FriendshipController	69	20	29	3	42
TypeScope	HomeController	80	3	9	3	4
TypeScope	Menu_CategoryController	69	22	30	3	50
TypeScope	MenuController	70	20	35	3	53
TypeScope	OrderController	68	24	31	3	46
TypeScope	RestaurantController	76	14	15	3	35
TypeScope	ReviewController	68	8	24	3	21
TypeScope	UserController	66	24	38	3	82
MemberScope	Create() : ActionResult	74	2	9		3
MemberScope	Create(user) : ActionResult	52	5	16		19
MemberScope	Delete(int) : ActionResult	77	1	5		3
MemberScope	DeleteConfirmed(int) : ActionResult	71	1	8		5
MemberScope	Details(int) : ViewResult	77	1	5		3
MemberScope	Dispose(bool) : void	87	1	3		2
MemberScope	Edit(int) : ActionResult	77	1	5		3
MemberScope	Edit(user, HttpPostedFileBase) : ActionResult	59	4	11		11
MemberScope	Index() : ViewResult	73	2	12		3
MemberScope	Index() : ViewResult	73	2	12		3
MemberScope	LogOff() : ActionResult	82	1	3		3
MemberScope	LogOn() : ViewResult	87	1	2		2
MemberScope	LogOn(string, string) : ActionResult	50	3	16		24
MemberScope	UserController()	92	1	2		1
NamespaceScope	TouchForFood.Models	93	308	32	2	331
TypeScope	category	92	9	4	1	11
TypeScope	CategoryFilterVM	92	6	4	1	10
TypeScope	friendship	93	11	2	1	11
TypeScope	item	92	19	6	1	21
TypeScope	menu	93	11	5	1	12
TypeScope	menu_category	93	13	5	1	14
TypeScope	menu_item	93	13	4	1	13
TypeScope	order	92	25	8	1	27
TypeScope	order_item	93	13	3	1	13
TypeScope	order_status	92	17	5	1	18
TypeScope	restaurant	92	21	9	1	25
TypeScope	restaurant_user	93	11	3	1	11
TypeScope	RestaurantMetadata	94	9	1	1	9
TypeScope	review	93	21	6	1	21
TypeScope	ReviewMetadata	93	11	5	1	11
TypeScope	sysdiagram	93	11	1	1	11
TypeScope	touch_for_foodEntities	92	32	19	2	32
TypeScope	user	91	27	7	1	32
TypeScope	UserMetadata	93	15	5	1	15
TypeScope	waiter	93	13	5	1	14
MemberScope	first_name.get() : string	98	1	0		1
MemberScope	first_name.set(string) : void	95	1	0		1
MemberScope	id.get() : int	98	1	0		1
MemberScope	id.set(int) : void	95	1	0		1
MemberScope	last_name.get() : string	98	1	0		1
MemberScope	last_name.set(string) : void	95	1	0		1

439	MemberScope	last_name.set(string) : void	95	1	0		1
440	MemberScope	orders.get() : ICollection<order>	98	1	2		1
441	MemberScope	orders.set(ICollection<order>) : void	95	1	2		1
442	MemberScope	restaurant.get() : restaurant	98	1	1		1
443	MemberScope	restaurant.set(restaurant) : void	95	1	1		1
444	MemberScope	resto_id.get() : int?	98	1	1		1
445	MemberScope	resto_id.set(int?) : void	95	1	1		1
446	MemberScope	waiter()	87	1	2		2
447	NamespaceScope	TouchForFood.Util.Category	70	8	12	1	15
448	TypeScope	CategoryUtil	70	8	12	1	15
449	MemberScope	CategoryUtil()	100	1	0		1
450	MemberScope	CategoryUtil()	94	1	1		1
451	MemberScope	filterListByMenu(menu) : IList<category>	58	6	12		13

8.7 Retrospective

In Iteration 2, we found that we were overly ambitious. We found that our time estimates were too optimistic which caused us to burn down tasks slower than planned. There were also still issues in learning how to use JIRA and setting up the coding environment. Being in the second sprint, some of these issues are understandable and accounted for in our best case-worst case styled estimates. However, corrective action must be taken immediately to avoid escalating these problems or repeating them further in the process. In this section we will see what the velocity was, what the team did well and did less well, along with what corrective measures we will take to improve our workflow and efficiency.

Our velocity in person-hours for this sprint was 50 person-hours. In terms of story points, our velocity was 0. If we look at cumulative velocity, we must add the velocity from Sprint 1 as well which was 2 story points, giving an average velocity of 1 story point per sprint. Although this figure seems alarming, we must consider that the Manage Personal Profile system is a story worth 34 points which we have started working on and spent over 24 person hours on already. Furthermore, the View Menu and Manage Menu user story are very close to completion. Ultimately, due to the fact that fractions of a user story can't be considered when measuring this type of velocity, the actual story point number didn't reflect perfectly the work that is done and what is left to be done.

8.7.1 What was learned

The following positive and negative points were gathered from this sprint.

Positives:

- We committed more person-hours than previously
- Our processes and procedures became more defined and clear to the team
- Team members slowly becoming more familiar with the technologies in place
- The technologies chosen for the coding environment and framework are powerful and useful

Negatives:

- Did not complete any user stories planned
- The learning curve on setup, JIRA, and coding slowed down general progress
- The times logged are still inaccurate
- The estimates for the time required for each task was generally incorrect
- Many charts and graphs based on story points are misleading

8.7.2 Changes in methodology

From this sprint a couple of changes in methodology were made. These changes come from the process of refining and perfecting our workflow to match the agile methodology and adapt it to our current project.

First of all, tasks were reorganized in JIRA, and tools were made available to all developers so they may access the specific task and log work under the correct sections. This effort is an attempt to reduce time needed to log one's time in JIRA and encourage a more involved and accurate timesheet in the end.

Another flaw that was exposed was the lack of planning in the start of the sprint. Many of the tasks in the sprint had their estimates completely wrong and some issues and tasks were not accounted for from the start. This is an unacceptable occurrence and in order to reduce the chances of this occurring again, more thorough sprint planning meetings will be conducted were design for the user stories is discussed in more depth, thus giving the team a better idea of what needs to be done and how complex each task really is. As a result of this meeting, all the tasks should be clearly labelled, estimated and accompanied by a description of what is expected to be done for this task to be resolved.

Finally, there is a more technical adjustment made on the JIRA side. It was noticed that the new GreenHopper tools only displayed graphs based on story points which, as discussed earlier, were not completely representative. The scrum leader then switched the sprint planning and reporting to work with JIRA's "Classic" functionality which allows more flexibility such as issues burndown and more importantly an hourly burndown chart to track the team's progress against an estimated guideline.

9 Iteration 3 Plan

9.1 Planned Activities

This sprint was scheduled to span four weeks instead of the usual two weeks because the CloudNine team members had responsibilities to complete final projects and final exams. Documentation defects were assigned in this sprint as were the stories: CAP-26 – View Menu and CAP-27 – Manage Menu. There were no expectations to get much work done within the first two weeks as many team members were overwhelmed with other responsibilities. After the first two weeks, CAP-35 – Manage Personal Profile was removed from the sprint as we realized it shouldn't have been there in the first place. Thus, the following two stories were planned for this sprint.

Table 9-1 Planned Stories

User Story ID	Total Story Points	Related Use Cases
CAP-27, CAP-26	11.00	UC1.1, UC3.1

The following defects were also planned. There are no story points associated to these defects.

Table 9-2 Planned Defects

Defect ID	Hours Estimated
CAP-88, CAP-91, CAP-92, CAP-93, CAP-94, CAP-95, CAP-96, CAP-97, CAP-98, CAP-99, CAP-100	15.25 hours

9.2 Person-Hour Estimation

The Expert Judgement Method was used to calculate person-hour estimations. Estimations were made for the worst case, most likely case, and best case. The worst case and best case are +30% and -30% of the most likely case. The expected case was then calculated with the formula:

Expected case = 1/6([worst case] + 4[most likely case] + [best case])

The numbers are extremely low because these hours only take planned stories into account. Overall however, there were many defects assigned in this sprint that were estimated to take 15.25 hours to complete. Although in time the sprint took place over four weeks, it was planned to be done in the final two weeks so all estimates are based on a fourteen day time period.

Table 9-3 Person-Hour Estimation

User Story ID	Worst Case	Most Likely Case	Best Case	Expected Case
CAP-27	6.5	5	3.5	5
CAP-26	8.8	6.75	4.7	6.75
Total (ph)	**15.3**	**11.75**	**8.2**	**11.75**
Velocity (ph/day)	**1.09**	**0.84**	**0.59**	**0.84**
Velocity (ph/team member/day)	**0.12**	**0.09**	**0.07**	**0.09**

10 Iteration 3 Report

10.1 Person-Hour Work Log

The following table shows the person-hour work log for each team member and activity during this iteration. This table was generated from the JIRA management system. As expected, there were very few hours worked this sprint due to other academic and work responsibilities. These hours reflect time spent on all tasks.

Start Date: **21/Nov/12** End Date: **17/Dec/12** [Change] (UNREGISTERED)			Christian Daher	Cristian Asenjo	Cynthia Donato	Josh Hum	Katrina Anderson	Matthew Tam	Mikhail Levkovsky	Patrick Modafferi	Ryan Nasr	Total
Issue		Total	11h	7.083h	7.517h	10.75h	11.25h	7.25h	5.5h	9.25h	12.5h	82.1h
CAP-8	Documentation - Proposal	1h	0h	0h	0h	0h	0h	0h	0h	1h	0h	1h
CAP-10	Documentation - SRS	3h	0h	2h	0h	0h	0h	1h	0h	0h	0h	3h
CAP-14	Correspondence	7h	2h	0h	0h	0.75h	3.25h	0h	0.25h	0.75h	0h	7h
CAP-18	Meetings	36h	3h	3h	4h	6.5h	4.5h	5.25h	4.75h	5h	0h	36h
CAP-23	Setup	5.5h	0h	0h	0h	0h	0h	0h	0h	0h	5.5h	5.5h
CAP-26	View Menu	8h	4h	0h	0h	0h	0h	0h	0h	0h	4h	8h
CAP-27	Manage Menu	3h	1.5h	0h	0h	0h	0h	0h	0h	0h	1.5h	3h
CAP-54	Decimals can't be used when creating a price for an item	0.167h	0.167h	0h	0h	0h	0h	0h	0h	0h	0h	0.167h
CAP-55	Database IDs need to be configured with auto-increment	0.083h	0.083h	0h	0h	0h	0h	0h	0h	0h	0h	0.083h
CAP-57	Documentation - Management	3h	0h	0h	0h	3h	0h	0h	0h	0h	0h	3h
CAP-85	Creating a review should use the last visit (i.e. the order) you created	0.5h	0h	0h	0h	0h	0h	0h	0h	0h	0.5h	0.5h
CAP-86	Latest reviews still don't display properly after adding new restaurant reviews	1h	0h	0h	0h	0h	0h	0h	0h	0h	1h	1h
CAP-88	Page 6-7-8-16	1h	0h	0h	0h	0h	0h	1h	0h	0h	0h	1h
CAP-91	Page 17	1h	0h	1h	0h	0h	0h	0h	0h	0h	0h	1h
CAP-92	Page 18	0.5h	0h	0h	0h	0h	0h	0h	0h	0.5h	0h	0.5h
CAP-93	Page 9	1.25h	0h	0.75h	0h	0h	0h	0h	0h	0.5h	0h	1.25h
CAP-94	Page 20	0.5h	0h	0h	0h	0h	0h	0h	0h	0.5h	0h	0.5h
CAP-95	24-27-28-32	1h	0h	0h	0h	0h	0h	0h	0h	1h	0h	1h
CAP-96	Traceability Page 40 something	3.517h	0h	0h	3.517h	0h	0h	0h	0h	0h	0h	3.517h
CAP-97	Page 57-61-67-73	0.5h	0h	0h	0h	0h	0h	0h	0.5h	0h	0h	0.5h
CAP-98	Page 76-80	0.5h	0h	0h	0h	0.5h	0h	0h	0h	0h	0h	0.5h
CAP-99	Page 28	0.333h	0h	0.333h	0h	0h	0h	0h	0h	0h	0h	0.333h
CAP-100	Eliminate Redundancy	3h	0h	0h	0h	0h	3h	0h	0h	0h	0h	3h
CAP-101	Vision Doc Section 4.3 Missing Information	0.25h	0.25h	0h	0h	0h	0h	0h	0h	0h	0h	0.25h
CAP-102	Test Plan Incorrect Dates	0.5h	0h	0h	0h	0h	0.5h	0h	0h	0h	0h	0.5h

Figure 10-1 Person-Hour Work Log

10.2 Hour Burndown Chart

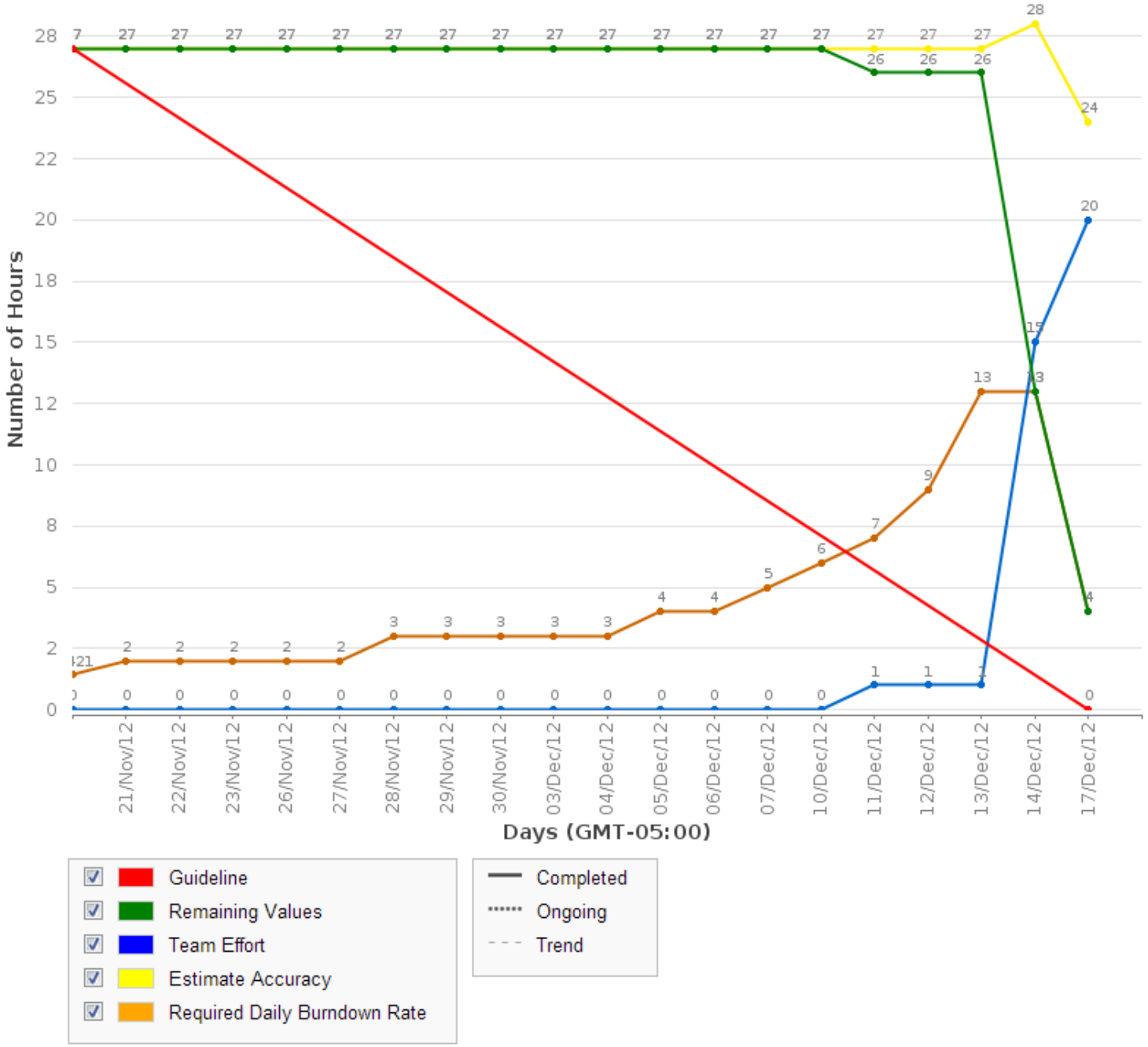

Figure 10-2 Hour Burndown Chart

In total, between planned stories and defects, there were 27 hours estimated to working. As seen in the above graph, most of the work was either done last minute or logged last minute. Both stories were completed and all defects were completed except for two. Overall, we met our goals for our stories and for most of our defects. However, we were not satisfied with our overall time management because we felt that we did not plan enough and therefore felt no pressure to get more work done.

10.3 Issue Burndown Chart

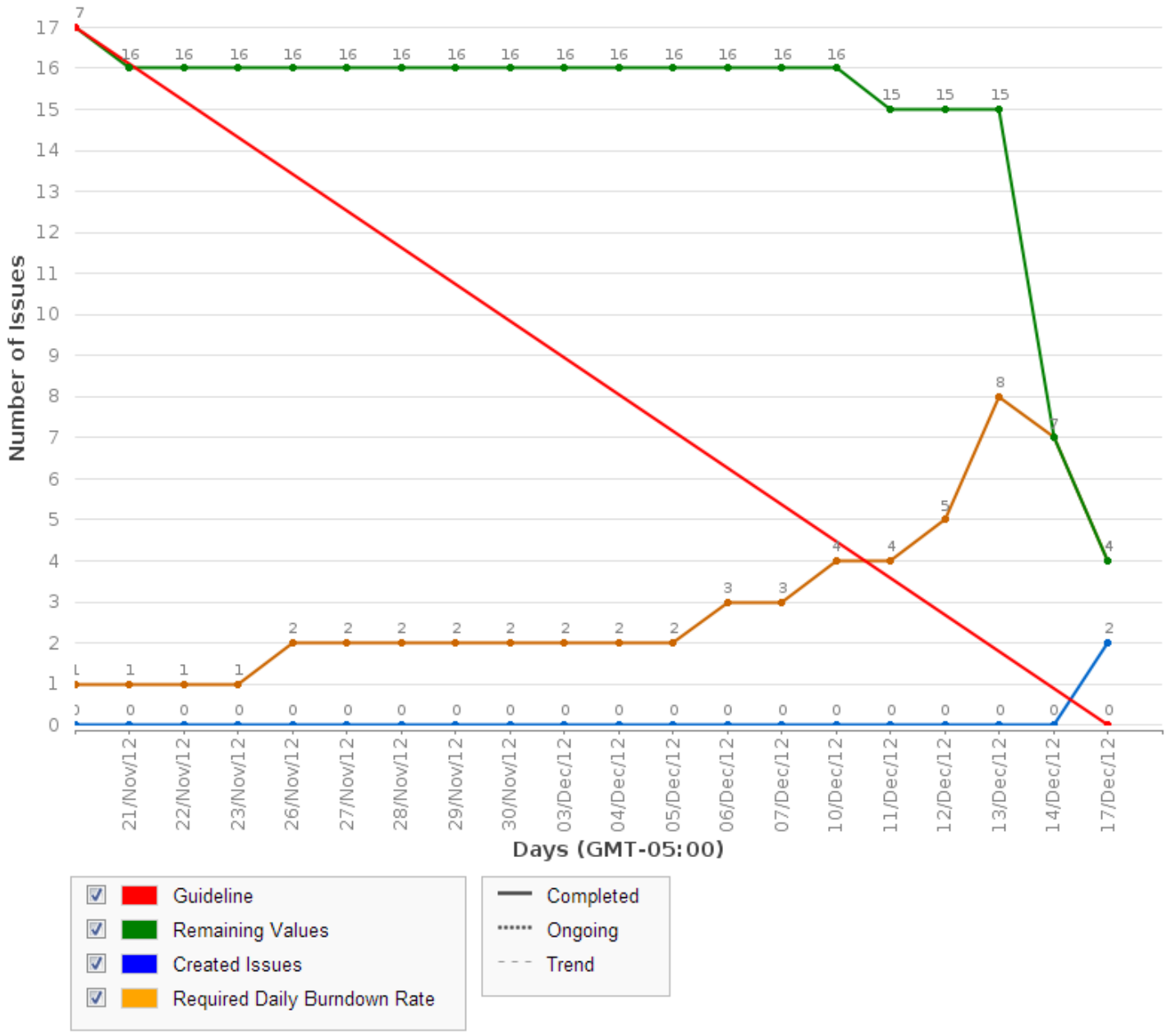

Figure 10-3 Issue Burndown Chart

As seen in the issue burndown chart, most of the issues (both stories and defects) were completed. The two defects that were not completed were CAP-98 and CAP-100 which we realized would take more time and planning than expected.

10.4 Cumulative Flow Diagram

The following diagram shows the cumulative flow of the project so far based on the number of issues completed, in progress, and to do. Although we are completing issues, we are also finding bugs and adding them to the backlog which explains the increase of issues to do at the end of the sprint.

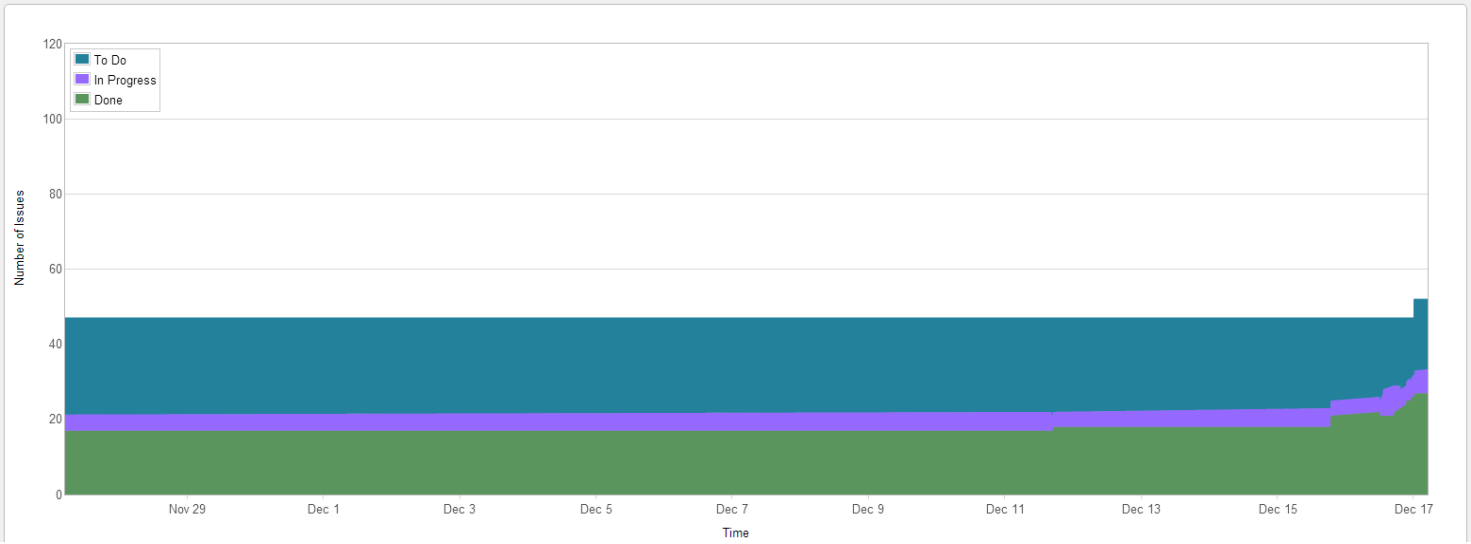

Figure 10-4 Cumulative Flow Diagram

10.5 Measurement Report

10.5.1 Code Quality Analysis

The following report was generated after analyzing the code. Only one method did not meet code quality expectations by exceeding the expected number of lines of code. This was found in an earlier iteration and has not been yet corrected.

Analysis tool used: Code Metrics Viewer
Found at: http://visualstudiogallery.msdn.microsoft.com/9f35524b-a784-4dbc-bd7b-6babd7a5a3b3
Version: 1.5.3
Last updated: 2/5/2012

	Scope	Hiearchy	Maintainability Index	Cyclomatic Complexity	Depth of Inheritance	Class Coupling	Lines of Code
2	ModuleScope	TouchForFood.Tests.dll	72	13	20	1	29
14	ModuleScope	TouchForFood.dll	85	494	95	3	787
15	NamespaceScope	TouchForFood	81	4	9	2	9
16	TypeScope	MvcApplication	81	4	9	2	9
17	MemberScope	Application_Start() : void	80	1	3		3
18	MemberScope	MvcApplication()	100	1	1		1
19	MemberScope	RegisterGlobalFilters(GlobalFilterCollection) : void	94	1	2		1
20	MemberScope	RegisterRoutes(RouteCollection) : void	71	1	3		4
21	NamespaceScope	TouchForFood.Attributes	84	2	5	3	4
22	TypeScope	AjaxAttribute	84	2	5	3	4
23	MemberScope	AjaxAttribute(bool)	87	1	1		2
24	MemberScope	IsValidForRequest(ControllerContext, MethodInfo) : bo	83	1	4		2
25	NamespaceScope	TouchForFood.Controllers	71	170	66	3	426
26	TypeScope	CategoryController	76	13	16	3	36
38	TypeScope	FriendshipController	69	20	29	3	42
49	TypeScope	HomeController	80	3	9	3	4
52	TypeScope	ItemController	74	17	29	3	41
64	TypeScope	Menu_CategoryController	69	22	30	3	50
76	TypeScope	MenuController	70	20	35	3	53
89	TypeScope	OrderController	68	24	31	3	46
100	TypeScope	RestaurantController	76	14	15	3	35
111	TypeScope	ReviewController	62	13	30	3	32
116	TypeScope	UserController	66	24	38	3	87
117	MemberScope	Create() : ActionResult	74	2	9		3
118	MemberScope	Create(user) : ActionResult	52	5	16		19
119	MemberScope	Delete(int) : ActionResult	77	1	5		3
120	MemberScope	DeleteConfirmed(int) : ActionResult	71	1	8		5
121	MemberScope	Details(int) : ViewResult	77	1	5		3
122	MemberScope	Dispose(bool) : void	87	1	3		2
123	MemberScope	Edit(int) : ActionResult	77	1	5		3
124	MemberScope	Edit(user, HttpPostedFileBase) : ActionResult	59	4	11		11

125	MemberScope	Index() : ViewResult	73	2	12		3
126	MemberScope	LogOff() : ActionResult	82	1	3		3
127	MemberScope	LogOn() : ViewResult	87	1	2		2
128	MemberScope	LogOn(string, string) : ActionResult	47	3	21		29
129	MemberScope	UserController()	92	1	2		1
130	NamespaceScope	TouchForFood.Models	93	310	32	2	333
131	TypeScope	category	92	9	4	1	11
141	TypeScope	CategoryFilterVM	92	6	4	1	10
148	TypeScope	friendship	93	11	2	1	11
160	TypeScope	item	92	19	6	1	21
180	TypeScope	menu	93	13	5	1	14
194	TypeScope	menu_category	93	13	5	1	14
208	TypeScope	menu_item	93	13	4	1	13
222	TypeScope	order	92	25	8	1	27
248	TypeScope	order_item	93	13	3	1	13
262	TypeScope	order_status	92	17	5	1	18
280	TypeScope	restaurant	92	21	9	1	25
302	TypeScope	restaurant_user	93	11	3	1	11
314	TypeScope	RestaurantMetadata	94	9	1	1	9
324	TypeScope	review	93	21	6	1	21
346	TypeScope	ReviewMetadata	93	11	5	1	11
358	TypeScope	sysdiagram	93	11	1	1	11
370	TypeScope	touch_for_foodEntities	92	32	19	2	32
403	TypeScope	user	91	27	7	1	32
431	TypeScope	UserMetadata	93	15	5	1	15
447	TypeScope	waiter	93	13	5	1	14
448	MemberScope	first_name.get() : string	98	1	0		1
449	MemberScope	first_name.set(string) : void	95	1	0		1
450	MemberScope	id.get() : int	98	1	0		1
451	MemberScope	id.set(int) : void	95	1	0		1
452	MemberScope	last_name.get() : string	98	1	0		1
453	MemberScope	last_name.set(string) : void	95	1	0		1
454	MemberScope	orders.get() : ICollection<order>	98	1	2		1
455	MemberScope	orders.set(ICollection<order>) : void	95	1	2		1
456	MemberScope	restaurant.get() : restaurant	98	1	1		1
457	MemberScope	restaurant.set(restaurant) : void	95	1	1		1
458	MemberScope	resto_id.get() : int?	98	1	1		1
459	MemberScope	resto_id.set(int?) : void	95	1	1		1
460	MemberScope	waiter()	87	1	2		2
461	NamespaceScope	TouchForFood.Util.Category	70	8	12	1	15
462	TypeScope	CategoryUtil	70	8	12	1	15
463	MemberScope	CategoryUtil()	100	1	0		1
464	MemberScope	CategoryUtil()	94	1	1		1
465	MemberScope	filterListByMenu(menu) : IList<category>	58	6	12		13

Figure 10-5 Code Quality Report

10.6 Retrospective

In Iteration 3, we underachieved. Although we met our goals, our goals were probably not set high enough. Even though this was with due reason, it was probably still not enough and we are not satisfied with our overall effort. On the other hand, however, the two stories planned were finished which makes this sprint a success.

10.6.1 Velocity

Sprint 3 velocity (story points): 11 story points
Cumulative velocity (story points): 13 story points
Average velocity (story points): 4.3 story points

Sprint 3 velocity (p-h): 20 person-hours
Cumulative velocity (p-h): 98 person-hours
Average velocity per sprint: 32.7 person-hours

Our sprint 3 velocity in story points greatly exceeded our previous velocities. However, this was due to the fact that the stories completed did not have much work left to be done. In terms of person-hours, we spent less time working on stories than in the previous two sprints.

10.6.2 Budget

Total person-hours budgeted to date: 2,484 person-hours
Total person-hours worked to date: 306.725 person-hours

We are grossly under-budget. This can be attributed to both a large over-estimation in the budget and also not enough time commitment from team members.

10.6.3 What was learned

The following positive and negative points were gathered from this sprint.

Positives:

- Completed 11 story points which is greater than any previous sprint
- Team members slowly becoming more familiar with the technologies in place

Negatives:

- Spent a lot of time on overhead and not so much on coding new stories and moving the project forward
- Did not commit enough time/effort to this sprint
- Still figuring out how to use JIRA to its full potential

11 Iteration 4 Plan

11.1 Planned Activities

The stories planned for this sprint are CAP-25 – Order Food and CAP-29 – Manage Order. They are necessary to complete before moving forward with other aspects of the application as well as comprise the most important aspects of the Touch for Food application.

Table 11-1 Planned Stories

User Story ID	Total Story Points
CAP-25, CAP-29	18.00

The following defects were also planned. There are no story points associated to these defects.

Table 11-2 Planned Defects

Defect ID	Hours Estimated
CAP-98, CAP-100, CAP-111, CAP-125, CAP-156, CAP-160, CAP-168	15.00 hours

11.2 Person-Hour Estimation

The Expert Judgement Method was used to calculate person-hour estimations. Estimations were made for the worst case, most likely case, and best case. The worst case and best case are +30% and -30% of the most likely case. The expected case was then calculated with the formula:

Expected case = 1/6([worst case] + 4[most likely case] + [best case])

Table 11-3 Person-Hour Estimation

User Story ID	Worst Case	Most Likely Case	Best Case	Expected Case
CAP-25	89.7	69	48.3	69
CAP-29	75.4	58	40.6	58
Total (ph)	**165.1**	**127**	**88.9**	**127**
Velocity (ph/day)	**11.8**	**9.1**	**6.4**	**9.1**
Velocity (ph/team member/day)	**1.3**	**1.0**	**0.7**	**1.0**

12 Iteration 4 Report

12.1 Person-Hour Work Log

The following table shows the person-hour work log for each team member and activity during this iteration. This table was generated from the JIRA management system. As planned, much more time was committed this sprint and it is reflected in this work log. These hours reflect time spent on all tasks.

Start Date: **18/Dec/12** End Date: **2/Jan/13** [Change] (UNREGISTERED)			Christian Daher	Cristian Asenjo	Cynthia Donato	Josh Hum	Katrina Anderson	Matthew Tam	Mikhail Levkovsky	Patrick Modafferi	Ryan Nasr	Total
Issue	Total	**26.5h**	**10.667h**	**7.25h**	**23h**	**38.433h**	**18.5h**	**36.85h**	**32.533h**	**16.25h**	**209.983h**
CAP-8	Documentation - Proposal	**0.017h**	0h	0h	0h	0h	0.017h	0h	0h	0h	0h	**0.017h**
CAP-10	Documentation - SRS	**0.5h**	0h	0h	0h	0h	0.5h	0h	0h	0h	0h	**0.5h**
CAP-11	Documentation - SAD	**0.917h**	0h	0h	0h	0h	0.917h	0h	0h	0h	0h	**0.917h**
CAP-14	Correspondence	**14.9h**	0h	1.633h	0.25h	0.25h	6.217h	0.5h	2.3h	2.25h	1.5h	**14.9h**
CAP-18	Meetings	**24.333h**	2h	1.75h	0h	3.75h	5.667h	2h	3.667h	5.5h	0h	**24.333h**
CAP-23	Setup	**10.75h**	0h	0h	0h	2.25h	3.25h	1h	1h	3.25h	0h	**10.75h**
CAP-25	Order Food	**81.633h**	24.25h	0h	0h	2h	9.517h	4.5h	18.617h	8h	14.75h	**81.633h**
CAP-29	Manage Order	**42.55h**	0h	7.283h	7h	11.75h	2h	10.5h	2h	2.017h	0h	**42.55h**
CAP-57	Documentation - Management	**5.017h**	0h	0h	0h	3h	0.017h	0h	0h	2h	0h	**5.017h**
CAP-98	Page 76-80	**0.017h**	0h	0h	0h	0h	0h	0h	0h	0.017h	0h	**0.017h**
CAP-100	Eliminate Redundancy	**10.333h**	0h	0h	0h	0h	10.333h	0h	0h	0h	0h	**10.333h**
CAP-111	Items getting added to menu instead of menu_items	**8.5h**	0h	0h	0h	0h	0h	0h	1h	7.5h	0h	**8.5h**
CAP-125	Ability to create a category and associate it to a menu from the menu page	**6h**	0h	0h	0h	0h	0h	0h	6h	0h	0h	**6h**
CAP-156	Cannot create new menu - Cannot implicitly convert type 'bool?' to 'bool'.	**0.267h**	0h	0h	0h	0h	0h	0h	0.267h	0h	0h	**0.267h**
CAP-160	Unable to add/remove items from a menu	**4h**	0h	0h	0h	0h	0h	0h	2h	2h	0h	**4h**
CAP-168	OrderStats Enum and Database Mismatch	**0.25h**	0.25h	0h	0h	0h	0h	0h	0h	0h	0h	**0.25h**

Figure 12-1 Person-Hour Work Log

12.2 Hour Burndown Chart

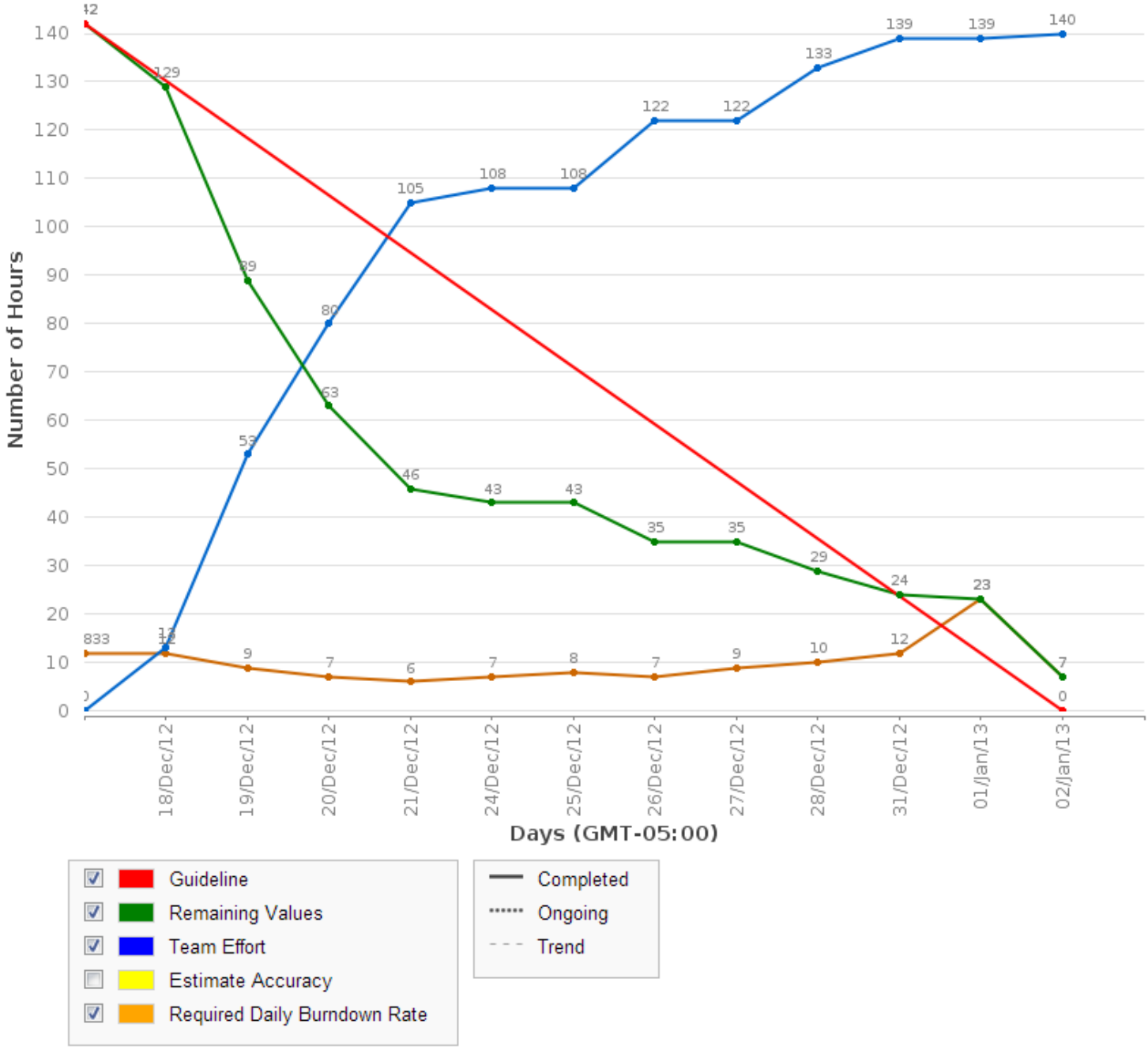

Figure 12-2 Hour Burndown Chart

For the first time, the estimated number of hours worked and the actual number of hours worked were very close. By the end of the sprint, we worked 7 hours less than initially planned. However, there was no delay in working and tasks were started at the beginning of the sprint. This is a very positive thing for the team.

12.3 Issue Burndown Chart

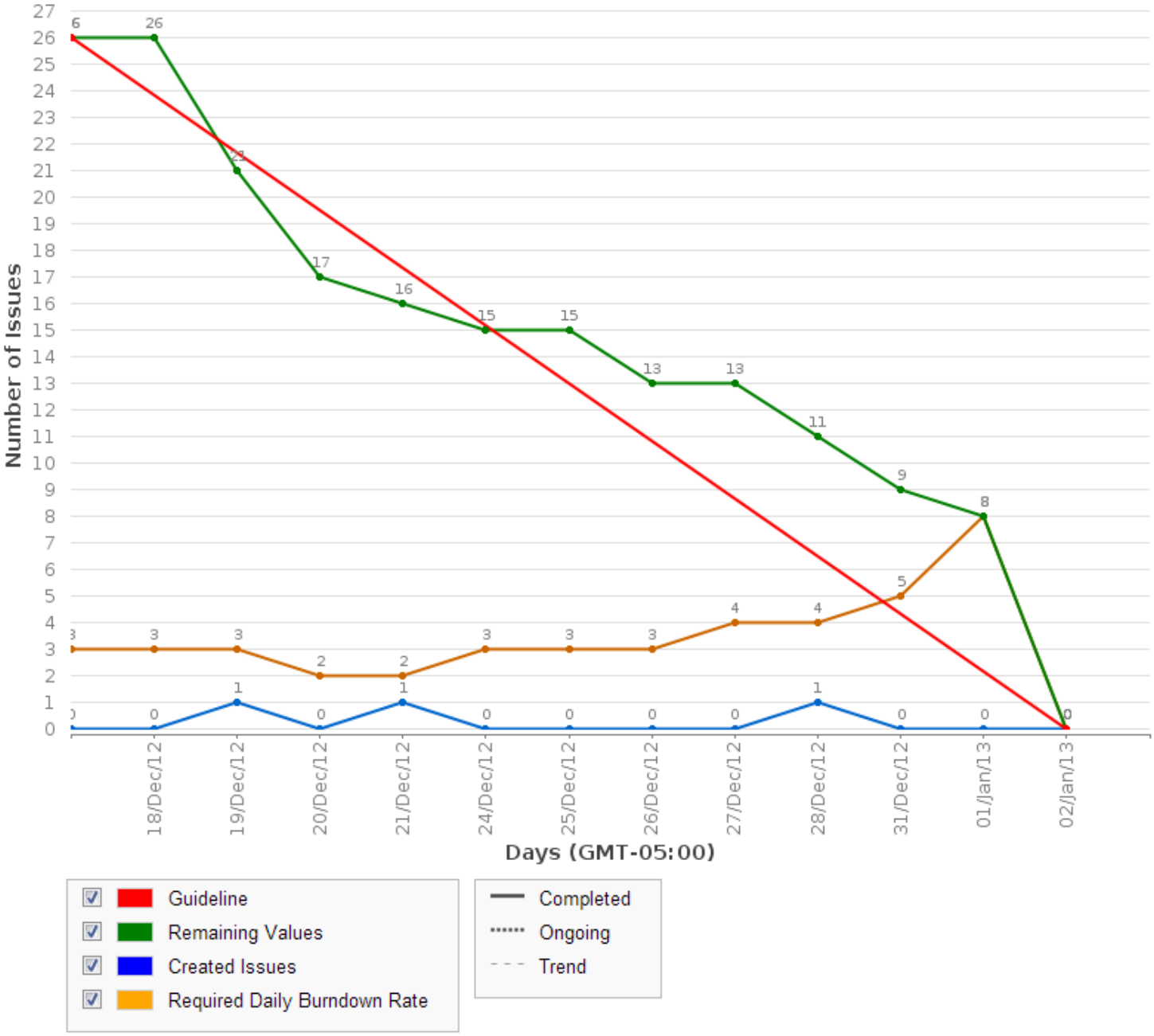

Figure 12-3 Issue Burndown Chart

As seen in the issue burndown chart, all of the issues planned were completed. Tasks were completed in a timely manner and burned down at a relatively steady rate.

12.4 Cumulative Flow Diagram

The following diagram shows the cumulative flow of the project so far based on the number of issues completed, in progress, and to do. Although we are completing issues, we are also finding bugs and adding them to the backlog which explains the increase of issues to do at the end of the sprint.

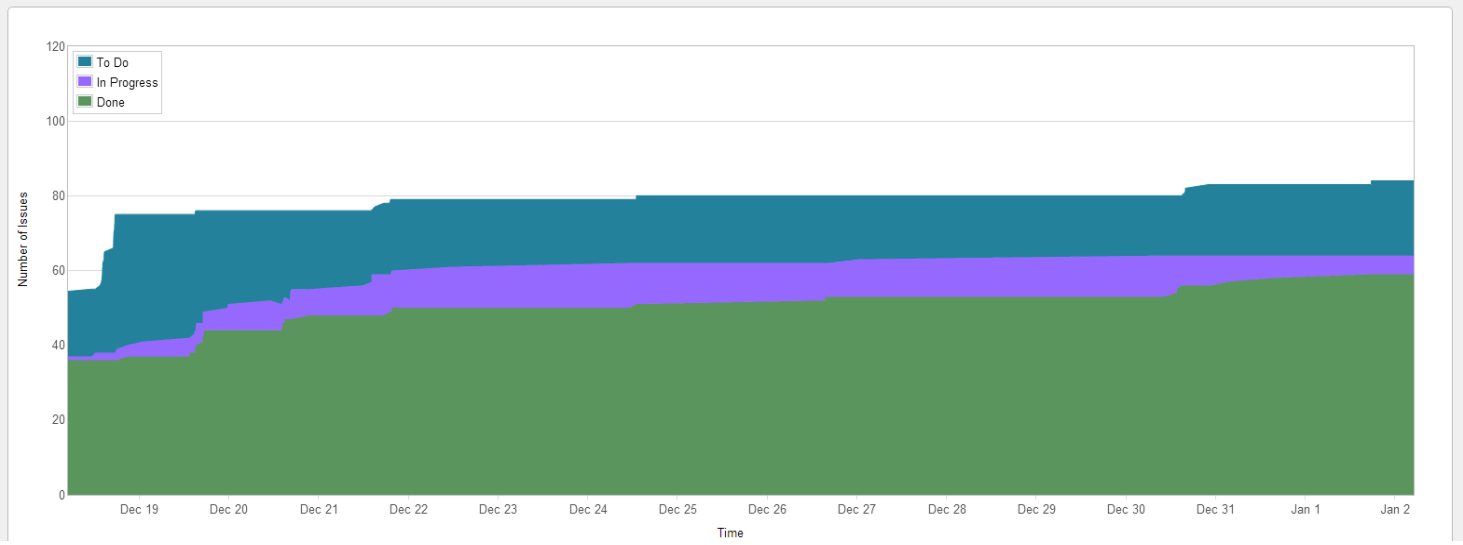

Figure 12-4 Cumulative Flow Diagram

12.5 Measurement Report

12.5.1 Code Quality Analysis

The following report was generated after analyzing the code. Three methods did not meet code quality goals by exceeding the expected number of lines of code. One of these was found in an earlier iteration and has not been yet corrected while the other two instances were introduced in this sprint. All of them have been entered as bugs and will be fixed in a future iteration.

Analysis tool used:	Code Metrics Viewer
Found at:	http://visualstudiogallery.msdn.microsoft.com/9f35524b-a784-4dbc-bd7b-6babd7a5a3b3
Version:	1.5.3
Last updated:	2/5/2012

	Scope	Hiearchy	Maintainability Index	Cyclomatic Complexity	Depth of Inheritance	Class Coupling	Lines of Code
2	ModuleScope	TouchForFood.dll	83	673	108	3	
3	NamespaceScope	TouchForFood	81	4	9	2	9
4	TypeScope	MvcApplication	81	4	9	2	9
5	MemberScope	Application_Start() : void	80	1	3		3
6	MemberScope	MvcApplication()	100	1	1		1
7	MemberScope	RegisterGlobalFilters(GlobalFilterCollection) : void	94	1	2		1
8	MemberScope	RegisterRoutes(RouteCollection) : void	71	1	3		4
9	NamespaceScope	TouchForFood.Attributes	84	2	5	3	4
10	TypeScope	AjaxAttribute	84	2	5	3	4
11	MemberScope	AjaxAttribute(bool)	87	1	1		2
12	MemberScope	IsValidForRequest(ControllerContext, MethodInfo) : bool	83	1	4		2
13	NamespaceScope	TouchForFood.Controllers	69	272	80	3	708
14	TypeScope	CategoryController	72	17	19	3	54
28	TypeScope	FriendshipController	69	20	29	3	42
39	TypeScope	HomeController	80	3	9	3	4
42	TypeScope	ItemController	70	22	36	3	63
57	TypeScope	Menu_CategoryController	67	26	33	3	65
71	TypeScope	Menu_ItemController	69	22	30	3	50
83	TypeScope	MenuController	70	20	35	3	53
96	TypeScope	Order_ItemController	66	30	43	3	66
109	TypeScope	OrderController	61	61	50	3	157
129	TypeScope	RestaurantController	76	14	15	3	35
140	TypeScope	ReviewController	62	13	30	3	32
145	TypeScope	UserController	66	24	38	3	87
146	MemberScope	Create() : ActionResult	74	2	9		3
147	MemberScope	Create(user) : ActionResult	52	5	16		19
148	MemberScope	Delete(int) : ActionResult	77	1	5		3
149	MemberScope	DeleteConfirmed(int) : ActionResult	71	1	8		5
150	MemberScope	Details(int) : ViewResult	77	1	5		3
151	MemberScope	Dispose(bool) : void	87	1	3		2
152	MemberScope	Edit(int) : ActionResult	77	1	5		3

153	MemberScope	Edit(user, HttpPostedFileBase) : ActionResult	59	4	11		11
154	MemberScope	Index() : ViewResult	73	2	12		3
155	MemberScope	LogOff() : ActionResult	82	1	3		3
156	MemberScope	LogOn() : ViewResult	87	1	2		2
157	MemberScope	LogOn(string, string) : ActionResult	47	3	21		29
158	MemberScope	UserController()	92	1	2		1
159	NamespaceScope	TouchForFood.Models	92	338	36	2	397
160	TypeScope	category	92	9	4	1	11
170	TypeScope	CategoryFilterVM	83	9	6	1	16
179	TypeScope	friendship	93	11	2	1	11
191	TypeScope	item	92	17	5	1	18
209	TypeScope	ItemFilterVM	83	9	6	1	16
218	TypeScope	menu	93	13	5	1	14
232	TypeScope	menu_category	93	13	5	1	14
246	TypeScope	menu_item	93	15	7	1	16
262	TypeScope	MenuItemStatusHelper	100	0	0	1	0
263	TypeScope	MenuItemStatusHelper.MenuItemStatusEnum	100	0	0	1	0
264	TypeScope	order	89	29	11	1	36
291	TypeScope	order_item	92	20	5	1	22
312	TypeScope	OrderItemMetadata	100	1	0	1	1
314	TypeScope	OrderMetadata	100	1	0	1	1
316	TypeScope	OrderStatusHelper	66	24	7	1	44
322	TypeScope	OrderStatusHelper.OrderItemStatusEnum	100	0	0	1	0
323	TypeScope	OrderStatusHelper.OrderStatusEnum	100	0	0	1	0
324	TypeScope	restaurant	92	21	9	1	25
346	TypeScope	restaurant_user	93	11	3	1	11
358	TypeScope	RestaurantMetadata	94	9	1	1	9
368	TypeScope	review	93	21	6	1	21
390	TypeScope	ReviewMetadata	93	11	5	1	11
402	TypeScope	sysdiagram	93	11	1	1	11
414	TypeScope	touch_for_foodEntities	92	28	17	2	28
443	TypeScope	user	91	27	7	1	32
471	TypeScope	UserMetadata	93	15	5	1	15
487	TypeScope	waiter	93	13	5	1	14
488	MemberScope	first_name.get() : string	98	1	0		1
489	MemberScope	first_name.set(string) : void	95	1	0		1
490	MemberScope	id.get() : int	98	1	0		1
491	MemberScope	id.set(int) : void	95	1	0		1
492	MemberScope	last_name.get() : string	98	1	0		1
493	MemberScope	last_name.set(string) : void	95	1	0		1
494	MemberScope	orders.get() : ICollection<order>	98	1	2		1
495	MemberScope	orders.set(ICollection<order>) : void	95	1	2		1
496	MemberScope	restaurant.get() : restaurant	98	1	1		1
497	MemberScope	restaurant.set(restaurant) : void	95	1	1		1
498	MemberScope	resto_id.get() : int?	98	1	1		1
499	MemberScope	resto_id.set(int?) : void	95	1	1		1
500	MemberScope	waiter()	87	1	2		2
501	NamespaceScope	TouchForFood.Util.Category	70	8	12	1	15
502	TypeScope	CategoryUtil	70	8	12	1	15
503	MemberScope	CategoryUtil()	100	1	0		1
504	MemberScope	CategoryUtil()	94	1	1		1
505	MemberScope	filterListByMenu(menu) : IList<category>	58	6	12		13
506	NamespaceScope	TouchForFood.Util.Item	69	10	14	1	16
507	TypeScope	ItemUtil	69	10	14	1	16
508	MemberScope	filterListByItem(menu_category) : IList<item>	56	8	14		14
509	MemberScope	ItemUtil()	100	1	0		1
510	MemberScope	ItemUtil()	94	1	1		1
511	NamespaceScope	TouchForFood.Util.Order	80	16	16	1	34
512	TypeScope	OrderStatusUtil	95	2	1	1	2
515	TypeScope	OrderUtil	66	14	16	1	32
516	MemberScope	filterItem(menu_item) : item	77	1	4		3
517	MemberScope	filterMenuItem(order_item) : menu_item	74	1	5		4
518	MemberScope	mergeExistingOrderToDb(order) : void	48	10	14		23
519	MemberScope	OrderUtil()	100	1	0		1

520	MemberScope	OrderUtil()	94	1	1		1
521	NamespaceScope	TouchForFood.Util.Session	57	5	15	1	27
522	TypeScope	SessionUtil	57	5	15	1	27
523	MemberScope	getOpenOrder(user) : order	48	4	15		26
524	MemberScope	SessionUtil()	100	1	0		1
525	NamespaceScope	TouchForFood.Util.User	77	4	8	1	9
526	TypeScope	UserUtil	77	4	8	1	9
527	MemberScope	getAuthenticatedUser(HttpRequestBase) : user	66	2	8		7
528	MemberScope	UserUtil()	100	1	0		1
529	MemberScope	UserUtil()	94	1	1		1
530	NamespaceScope	TouchForFood.ViewModels	79	14	11	1	25
531	TypeScope	OrderItemVM	76	1	2	1	4
533	TypeScope	OrderVM	82	13	9	1	21
534	MemberScope	addItem(OrderItemVM) : void	94	1	2		1
535	MemberScope	getItemById(int) : OrderItemVM	71	4	5		5
536	MemberScope	orderItemVMs.get() : IList<OrderItemVM>	98	1	2		1
537	MemberScope	orderItemVMs.set(IList<OrderItemVM>) : void	95	1	2		1
538	MemberScope	OrderVM()	80	1	3		3
539	MemberScope	OrderVM(order)	80	1	3		3
540	MemberScope	OrderVM(order, IList<OrderItemVM>)	81	1	3		3
541	MemberScope	removeById(int) : void	78	2	2		3
542	MemberScope	removeItem(int) : void	94	1	2		1

Figure 12-5 Code Quality Report

12.6 Retrospective

Iteration 4 was a success because we completed all the stories planned for this sprint. The project started gaining momentum as both the stories completed were critical. There was a greater effort put forth by the entire team and this produced immediate results. All other issues planned were also completed.

12.6.1 Velocity

Sprint 4 velocity (story points): 18 story points
Cumulative velocity (story points): 31 story points
Average velocity (story points): 7.75 story points

Sprint 4 velocity (p-h): 124.2 person-hours
Cumulative velocity (p-h): 222.2 person-hours
Average velocity per sprint: 55.6 person-hours

In sprint 4, the team's velocity in story points increased once again compared to previous sprints. We attribute this to simply more time being spent on coding than previously. The team's velocity in person-hours also increased. It is clear that as we dedicate more person-hours to the stories planned, the results will follow.

12.6.2 Budget

Total person-hours budgeted to date: 2,898 person-hours
Total person-hours worked: 516.705 person-hours

Person-hours budgeted per sprint: 414 person-hours
Person-hours worked in Sprint 4: 209.98 person-hours

Although we are still under budget, the effort produced in this sprint exceeded previous sprints. There was an over-estimation in the budget so it is unlikely to ever match the budget, but an effort closer to 300 person-hours will probably be necessary in future sprints.

12.6.3 What was learned

The following positive and negative points were gathered from this sprint.

Positives:

- Completed 18 story points which is greater than any previous sprint
- Greater team effort and contribution overall
- Critical stories were completed
- More organized configuration management
- Team members more familiar with the technologies in place

Negatives:

- Still need to catch up to get back on schedule

13 Iteration 5 Plan

In iteration 5, the plan is to make up for a lack of quality or thoroughness in past sprints. The story point count is low so that the team can focus on catching up on testing, documenting and amending defects. In addition, the user interface will be implemented throughout the application during this sprint.

13.1 Planned Activities

Tables featured in this section of the document were created with JIRA [**2**].

13.1.1 Tasks

Table 13-1 Iteration 5 Planned Tasks

Key	Issue Type	Priority	Assignee	Original Estimate (hours)
CAP-164	Task	Major	Katrina Anderson	0.00
CAP-165	Sub-task	Major	Katrina Anderson	10.00
CAP-166	Sub-task	Major	Katrina Anderson	10.00
CAP-172	Sub-task	Major	Katrina Anderson	1.00
CAP-57	Task	Major	*Unassigned*	0.00
CAP-173	Sub-task	Minor	Mikhail Levkovsky	1.00
CAP-174	Sub-task	Minor	Mikhail Levkovsky	1.00
CAP-175	Sub-task	Minor	Mikhail Levkovsky	1.00
CAP-176	Sub-task	Minor	Mikhail Levkovsky	1.00
CAP-177	Sub-task	Major	Patrick Modafferi	1.00
CAP-178	Sub-task	Minor	Patrick Modafferi	1.00
CAP-180	Sub-task	Minor	Josh Hum	1.00
CAP-181	Sub-task	Minor	Josh Hum	1.00
CAP-182	Sub-task	Minor	Katrina Anderson	1.00
CAP-183	Sub-task	Minor	Josh Hum	1.50
CAP-184	Sub-task	Minor	Josh Hum	1.50
CAP-185	Sub-task	Minor	Josh Hum	1.50
CAP-186	Sub-task	Minor	Josh Hum	2.00
CAP-187	Sub-task	Minor	Josh Hum	2.00
CAP-188	Sub-task	Minor	Josh Hum	2.00
CAP-189	Sub-task	Minor	Josh Hum	5.00
CAP-11	Task	Major	*Unassigned*	0.00
CAP-215	Sub-task	Minor	Katrina Anderson	2.00
CAP-162	Task	Major	Josh Hum, Patrick Modafferi & Matt Tam	0.00
CAP-197	Sub-task (CAP-162)	Major	Josh Hum	2.00
CAP-202	Sub-task (CAP-162)	Major	Patrick Modafferi	6.00
CAP-208	Sub-task (CAP-162)	Major	Patrick Modafferi	5.00
CAP-209	Sub-task (CAP-162)	Major	Josh Hum, Patrick Modafferi & Matt Tam	16.00
CAP-210	Sub-task (CAP-162)	Major	Josh Hum, Patrick Modafferi & Matt Tam	16.00

CAP-211	Sub-task (CAP-162)	Major	Patrick Modafferi	4.00
CAP-212	Sub-task (CAP-162)	Major	Patrick Modafferi	18.00
CAP-213	Sub-task (CAP-162)	Major	Matthew Tam	16.00
CAP-214	Sub-task (CAP-162)	Major	Josh Hum, Patrick Modafferi & Matt Tam	8.00
CAP-216	Sub-task (CAP-162)	Major	Josh Hum, Patrick Modafferi & Matt Tam	10.00
CAP-217	Sub-task (CAP-162)	Major	Josh Hum	4.00
CAP-218	Sub-task (CAP-162)	Major	Josh Hum	6.00
CAP-230	Sub-task (CAP-162)	Major	Josh Hum	4.00
CAP-231	Sub-task (CAP-162)	Major	*Unassigned*	5.00
CAP-198	Task	Major	Mikhail Levkovsky, Katrina Anderson & Cristian Asenjo	0.00
CAP-199	Sub-task (CAP-198)	Major	Katrina Anderson	3.00
CAP-200	Sub-task (CAP-198)	Major	Katrina Anderson	3.00
CAP-201	Sub-task (CAP-198)	Major	Mikhail Levkovsky	3.00
CAP-203	Sub-task (CAP-198)	Major	Cristian Asenjo	16.00
CAP-204	Sub-task (CAP-198)	Major	Mikhail Levkovsky	3.00
CAP-205	Sub-task (CAP-198)	Major	Mikhail Levkovsky	5.00
CAP-206	Sub-task (CAP-198)	Major	Katrina Anderson	6.00
CAP-207	Sub-task (CAP-198)	Major	Mikhail Levkovsky	1.00
			TOTAL	**207.50**

13.1.2 Stories

In this iteration, two stories are planned to be resolved, for a total of six story points.

Table 13-2 Iteration 5 Planned Stories

Key	Issue Type	Priority	Assignee	Story Points	Original Estimate (hours)
CAP-40	Story	Critical	Cristian Asenjo & Cynthia Donato	3.00	0.00
CAP-163	Sub-task (CAP-40)	Major	Cristian Asenjo & Cynthia Donato		16.00
CAP-194	Sub-task (CAP-40)	Major	Cristian Asenjo & Cynthia Donato		4.00
CAP-196	Sub-task (CAP-40)	Major	Cristian Asenjo & Cynthia Donato		12.00
CAP-192	Story	Blocker	Christian Daher & Ryan Nasr	3.00	0.00
CAP-219	Sub-task (CAP-192)	Major	Ryan Nasr		8.00
CAP-220	Sub-task (CAP-192)	Major	Ryan Nasr		8.00
CAP-221	Sub-task (CAP-192)	Major	Christian Daher & Ryan Nasr		20.00
CAP-222	Sub-task (CAP-192)	Major	Ryan Nasr		6.00
CAP-224	Sub-task (CAP-192)	Major	Christian Daher & Ryan Nasr		4.00
CAP-225	Sub-task (CAP-192)	Major	Christian Daher & Ryan Nasr		16.00
CAP-244	Sub-task (CAP-192)	Major	Christian Daher & Ryan Nasr		2.00
CAP-245	Sub-task (CAP-192)	Major	Christian Daher & Ryan Nasr		2.00
			TOTAL	**6.00**	**98.00**

13.1.3 Defects

In this iteration, twelve defects are planned to be resolved. Blocker or Critical category defects can be added to the iteration at any time.

Table 13-3 Iteration 5 Planned Defects

Key	Issue Type	Priority	Assignee	Reporter	Original Estimate (hours)
CAP-85	Bug	Critical	Ryan Nasr	Cristian Asenjo	0.50
CAP-108	Bug	Minor	Christian Daher	Christian Daher	4.00
CAP-109	Bug	Major	Mikhail Levkovsky	Christian Daher	4.00
CAP-128	Bug	Trivial	Mikhail Levkovsky	Patrick Modafferi	2.00
CAP-158	Bug	Minor	Mikhail Levkovsky	Katrina Anderson	3.00
CAP-159	Bug	Major	Ryan Nasr	Matthew Tam	0.50
CAP-161	Bug	Critical	Patrick Modafferi	Cynthia Donato	0.17
CAP-169	Bug	Major	Cynthia Donato	Katrina Anderson	5.00
CAP-170	Bug	Trivial	Katrina Anderson	Katrina Anderson	0.25
CAP-190	Bug	Major	Ryan Nasr	Patrick Modafferi	4.00
CAP-191	Bug	Minor	Cristian Asenjo	Cristian Asenjo	4.00
CAP-193	Bug	Minor	Patrick Modafferi	Patrick Modafferi	6.00
CAP-226	Sub-task (CAP-193)	Major	Patrick Modafferi	Patrick Modafferi	0.00
CAP-227	Sub-task (CAP-193)	Major	Patrick Modafferi	Patrick Modafferi	0.00
CAP-228	Sub-task (CAP-193)	Major	Patrick Modafferi	Patrick Modafferi	0.00
				TOTAL	**33.42**

13.2 Person-Hour Estimation

The Expert Judgement Method was used to calculate person-hour estimations.

Table 13-4 Iteration 5 Person-Hour Estimation

Description	Worst Case	Most Likely Case	Best Case	Expected Case
Tasks	269.75	207.50	145.25	207.50
Stories	127.40	98.00	68.60	98.00
Defects	43.45	33.42	23.39	33.42
Total(ph)	440.60	338.92	237.24	338.92
Velocity(ph/day)	31.47	24.21	16.95	24.21
Velocity (ph/team member/day)	3.50	2.69	1.88	2.69

14 Iteration 5 Report

14.1 Person-Hour Work Log

Start Date: 2/Jan/13 End Date: 15/Jan/13 [Change] (UNREGISTERED)				Christian Daher	Cristian Asenjo	Cynthia Donato	Josh Hum	Katrina Anderson	Matthew Tam	Mikhail Levkovsky	Patrick Modafferi	Ryan Nasr	Total
Issue			Total	21.833h	27.517h	24.767h	52.008h	38.292h	30.767h	34.45h	41.9h	17.633h	289.167h
CAP-8	Documentation - Proposal		0.017h	0h	0h	0h	0h	0.017h	0h	0h	0h	0h	0.017h
CAP-11	Documentation - SAD		2.5h	0h	0h	0h	0h	1h	0h	0h	1.5h	0h	2.5h
CAP-14	Correspondence		34.467h	1.083h	2.467h	4.583h	5.692h	9.925h	2.15h	3.067h	4.5h	1h	34.467h
CAP-18	Meetings		49.417h	2.5h	3.75h	5h	8.167h	6.917h	5.333h	9.083h	8.667h	0h	49.417h
CAP-23	Setup		13.917h	0h	0h	3.5h	0.333h	2.75h	1h	0.667h	3.167h	2.5h	13.917h
CAP-35	Manage Personal Profile		0.033h	0h	0h	0h	0h	0h	0h	0h	0.033h	0h	0.033h
CAP-40	Sign in		5.417h	0h	5.417h	0h	0h	0h	0h	0h	0h	0h	5.417h
CAP-57	Documentation - Management		25.283h	0h	0h	0h	19.75h	2.017h	0h	3.017h	0.5h	0h	25.283h
CAP-67	Add unit tests project into main TFF project		4h	0h	4h	0h	0h	0h	0h	0h	0h	0h	4h
CAP-85	Creating a review should use the last visit (i.e. the order) you created		0.333h	0h	0h	0h	0h	0h	0h	0h	0h	0.333h	0.333h
CAP-98	Page 76-80		0.017h	0h	0h	0h	0h	0h	0h	0h	0.017h	0h	0.017h
CAP-108	Activate/disactivate menu from index page of menu		2.5h	2h	0h	0h	0h	0h	0h	0h	0.5h	0h	2.5h
CAP-109	Soft Deleting a Menu_Category when there is a Menu_Item attached to it		4.933h	0h	0h	0h	0.183h	0h	0h	3.5h	1.25h	0h	4.933h
CAP-158	SAD Section 3 Invalid		2h	0h	0h	0h	0h	0h	0h	2h	0h	0h	2h
CAP-159	Database Data Type for Price and Total		0.5h	0h	0h	0h	0h	0h	0h	0h	0h	0.5h	0.5h
CAP-161	Setting Order Status to editing changes status to 6 instead of 3		0.833h	0h	0h	0.667h	0h	0h	0h	0h	0.167h	0h	0.833h
CAP-162	Look and Feel		56.333h	0h	0h	0h	17.883h	0h	22.283h	0h	16.167h	0h	56.333h
CAP-164	Documentation - Test Report		15.25h	0h	0h	0h	0h	15.25h	0h	0h	0h	0h	15.25h
CAP-168	OrderStats Enum and Database Mismatch		0.25h	0.25h	0h	0h	0h	0h	0h	0h	0h	0h	0.25h
CAP-169	Optimistic offline locks logic missing from orders		7.583h	0h	0h	7.583h	0h	0h	0h	0h	0h	0h	7.583h
CAP-170	Test Plan Section 5 Inaccurate		0.417h	0h	0h	0h	0h	0.417h	0h	0h	0h	0h	0.417h
CAP-190	Order Status Enum should no longer be in Database		4.017h	2h	0h	0h	0h	0h	0h	0h	0h	2.017h	4.017h
CAP-191	Fix Decline Order/Menu Item Status to use enum-only status rather than an enum/DB status		1.033h	0h	1.033h	0h	0h	0h	0h	0h	0h	0h	1.033h
CAP-192	Manage Tables		22.05h	13h	0h	0h	0h	0h	0h	0h	0.017h	9.033h	22.05h
CAP-193	Menu Items don't get a price set to them		6.5h	0h	0h	0h	0h	0h	0h	1.25h	5.25h	0h	6.5h
CAP-198	Unit Testing Retroactive		22.517h	1h	10.667h	0h	0h	0h	0h	9.85h	0h	1h	22.517h
CAP-229	Soft Deleting Menu when categories or items are associated to it		2.517h	0h	0h	0.5h	0h	0h	0h	2.017h	0h	0h	2.517h
CAP-252	build break - order does not have a field table number		0.517h	0h	0h	0.267h	0h	0h	0h	0h	0h	0.25h	0.517h
CAP-253	Order status enum runtime error in drop down list display		1h	0h	0h	0h	0h	0h	0h	0h	0h	1h	1h
CAP-259	Edit Order has a compilation error		0.167h	0h	0h	0h	0h	0h	0h	0h	0.167h	0h	0.167h
CAP-262	User edit not working		2.6h	0h	0.1h	2.5h	0h	0h	0h	0h	0h	0h	2.6h
CAP-275	Should not be able to decline an order that is in edit		0.25h	0h	0.083h	0.167h	0h	0h	0h	0h	0h	0h	0.25h

Figure 14-1 Person-Hour Work Log

14.2 Hour Burndown Chart

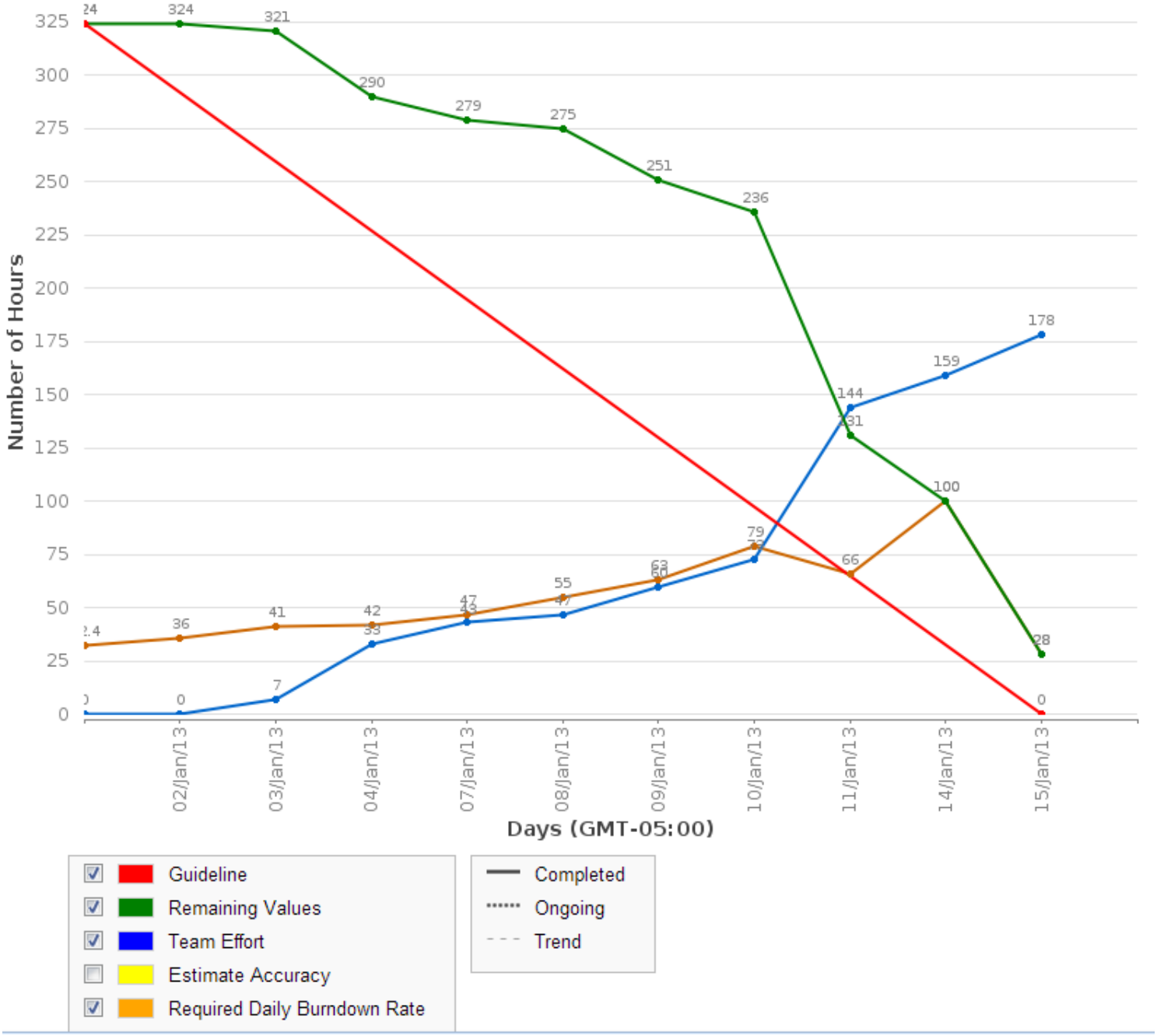

Figure 14-2 Hour Burndown Chart

Almost 300 hours were spent working in Iteration 5. The effort was steady as can be seen by this graph with quite a few hours being worked from the beginning of the sprint.

14.3 Issue Burndown Chart

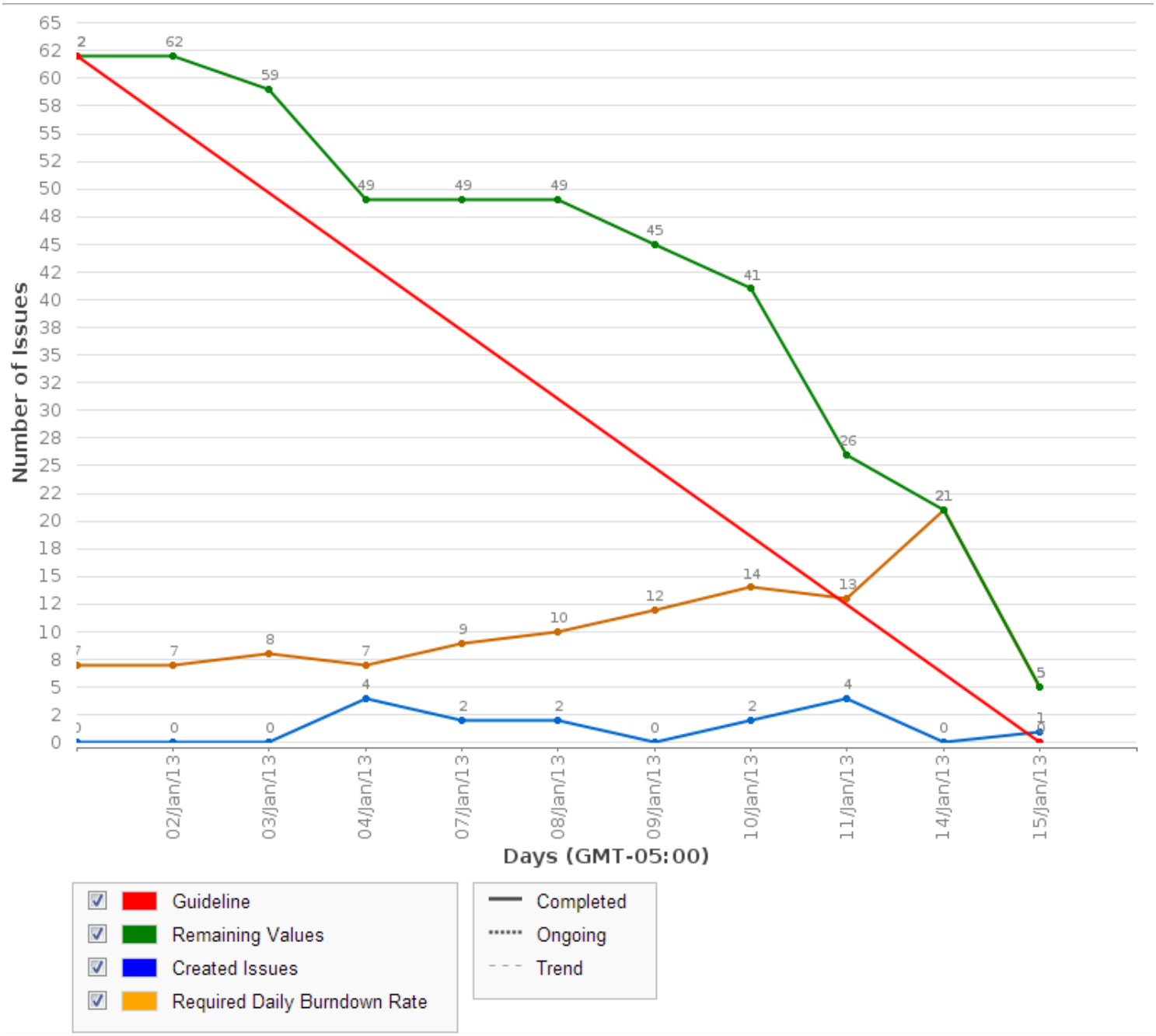

Figure 14-3 Issue Burndown Chart

As seen in the above chart, issues were burned down in correspondence with the amount of effort put in. Again, almost all the issues planned in this sprint were completed. CAP-40 – Sign In was pushed to the next sprint because it was not yet completed.

14.4 Cumulative Flow Diagram

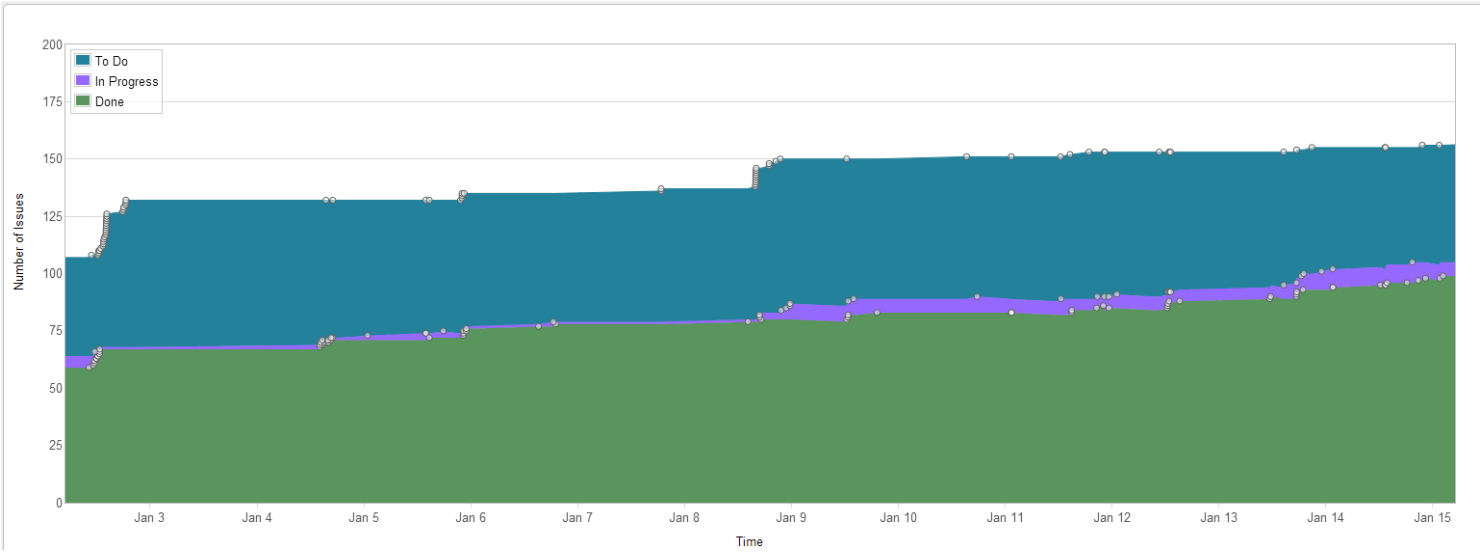

Figure 14-4 Cumulative Flow Diagram

Although new bugs are found and added as issues, our progress is steady in completing issues. Overall, almost 100 issues have been completed, leaving 50 more issues to complete until completing the project. Of course, we expect the number of remaining issues to increase as more bugs are added or new tasks defined.

14.5 Measurement Report

14.5.1 Code Quality Analysis

The following report was generated after analyzing the code. Four methods did not meet code quality goals by exceeding the expected number of lines of code. Three of these were found in an earlier iteration and had not been corrected yet while the new instances were introduced in this sprint. All of them have been entered as bugs and will be fixed in a future iteration.

Analysis tool used:	Code Metrics Viewer
Found at:	http://visualstudiogallery.msdn.microsoft.com/9f35524b-a784-4dbc-bd7b-6babd7a5a3b3
Version:	1.5.3
Last updated:	2/5/2012

	Scope	Hiearchy	Maintainability Index	Cyclomatic Complexity	Depth of Inheritance	Class Coupling	Lines of Code
2	ModuleScope	TouchForFood.Tests.dll	72	115	84	2	408
103	ModuleScope	TouchForFood.dll	82	870	149	4	
104	NamespaceScope	TouchForFood	81	4	9	2	9
105	TypeScope	MvcApplication	81	4	9	2	9
106	MemberScope	Application_Start() : void	80	1	3		3
107	MemberScope	MvcApplication()	100	1	1		1
108	MemberScope	RegisterGlobalFilters(GlobalFilterCollection) : void	94	1	2		1
109	MemberScope	RegisterRoutes(RouteCollection) : void	71	1	3		4
110	NamespaceScope	TouchForFood.Attributes	84	2	5	3	4
111	TypeScope	AjaxAttribute	84	2	5	3	4
112	MemberScope	AjaxAttribute(bool)	87	1	1		2
113	MemberScope	IsValidForRequest(ControllerContext, MethodInfo) : bool	83	1	4		2
114	NamespaceScope	TouchForFood.Controllers	68	341	95	3	891
115	TypeScope	CategoryController	71	19	28	3	60
129	TypeScope	FriendshipController	69	20	29	3	42
140	TypeScope	HomeController	80	3	9	3	4
143	TypeScope	ItemController	70	25	39	3	74
159	TypeScope	Menu_CategoryController	67	37	41	3	93
176	TypeScope	Menu_ItemController	67	24	32	3	56
188	TypeScope	MenuController	67	33	45	3	81
203	TypeScope	Order_ItemController	64	32	45	3	74
216	TypeScope	OrderController	60	71	49	3	184
237	TypeScope	RestaurantController	72	16	24	3	41
248	TypeScope	ReviewController	62	14	31	3	34
253	TypeScope	TableController	67	23	42	3	55
265	TypeScope	UserController	64	24	43	3	93
266	MemberScope	Create() : ActionResult	74	2	9		3
267	MemberScope	Create(user) : ActionResult	51	5	20		21
268	MemberScope	Delete(int) : ActionResult	77	1	5		3
269	MemberScope	DeleteConfirmed(int) : ActionResult	71	1	8		5
270	MemberScope	Details(int) : ViewResult	77	1	5		3

271	MemberScope	Dispose(bool) : void	87	1	3		2
272	MemberScope	Edit(int) : ActionResult	77	1	6		3
273	MemberScope	Edit(user, HttpPostedFileBase) : ActionResult	59	4	12		11
274	MemberScope	Index() : ViewResult	73	2	13		3
275	MemberScope	LogOff() : ActionResult	76	1	5		4
276	MemberScope	LogOn() : ViewResult	87	1	2		2
277	MemberScope	LogOn(string, string) : ActionResult	46	3	24		32
278	MemberScope	UserController()	92	1	2		1
279	NamespaceScope	TouchForFood.Exceptions	97	2	1	2	2
280	TypeScope	ItemActiveException	97	2	1	2	2
281	MemberScope	ItemActiveException()	100	1	1		1
282	MemberScope	ItemActiveException(string)	98	1	1		1
283	NamespaceScope	TouchForFood.Mappers	73	14	7	1	17
284	TypeScope	MenuCategoryOM	77	5	6	1	7
287	TypeScope	MenuOM	69	9	7	1	10
288	MemberScope	delete(menu) : void	61	8	7		9
289	MemberScope	MenuOM()	100	1	0		1
290	NamespaceScope	TouchForFood.Models	92	409	45	2	497
291	TypeScope	category	92	11	4	1	13
303	TypeScope	CategoryFilterVM	83	9	6	1	16
312	TypeScope	friendship	93	11	2	1	11
324	TypeScope	item	92	19	5	1	20
344	TypeScope	ItemFilterVM	80	18	16	1	34
357	TypeScope	menu	92	17	5	1	18
375	TypeScope	menu_category	92	19	5	1	20
395	TypeScope	menu_item	92	21	6	1	22
417	TypeScope	MenuMetadata	94	7	4	1	7
425	TypeScope	order	88	31	12	1	52
455	TypeScope	order_item	92	20	5	1	22
476	TypeScope	OrderItemMetadata	100	1	0	1	1
478	TypeScope	OrderMetadata	100	1	0	1	1
480	TypeScope	OrderStatusHelper	66	24	7	1	44
486	TypeScope	OrderStatusHelper.OrderItemStatusEnum	100	0	0	1	0
487	TypeScope	OrderStatusHelper.OrderStatusEnum	100	0	0	1	0
488	TypeScope	restaurant	91	25	10	1	30
514	TypeScope	restaurant_user	93	11	3	1	11
526	TypeScope	RestaurantMetadata	94	9	1	1	9
536	TypeScope	review	93	21	6	1	21
558	TypeScope	ReviewMetadata	93	11	5	1	11
570	TypeScope	sysdiagram	93	11	1	1	11
582	TypeScope	table	91	14	8	1	19
597	TypeScope	TableMetadata	100	1	0	1	1
599	TypeScope	touch_for_foodEntities	92	32	19	2	32
632	TypeScope	user	91	35	9	1	40
668	TypeScope	UserMetadata	93	15	5	1	15
684	TypeScope	waiter	93	15	5	1	16
685	MemberScope	first_name.get() : string	98	1	0		1
686	MemberScope	first_name.set(string) : void	95	1	0		1
687	MemberScope	id.get() : int	98	1	0		1
688	MemberScope	id.set(int) : void	95	1	0		1
689	MemberScope	last_name.get() : string	98	1	0		1
690	MemberScope	last_name.set(string) : void	95	1	0		1
691	MemberScope	orders.get() : ICollection<order>	98	1	2		1
692	MemberScope	orders.set(ICollection<order>) : void	95	1	2		1
693	MemberScope	restaurant.get() : restaurant	98	1	1		1
694	MemberScope	restaurant.set(restaurant) : void	95	1	1		1
695	MemberScope	resto_id.get() : int?	98	1	1		1
696	MemberScope	resto_id.set(int?) : void	95	1	1		1
697	MemberScope	version.get() : int	98	1	0		1
698	MemberScope	version.set(int) : void	95	1	0		1
699	MemberScope	waiter()	87	1	2		2
700	NamespaceScope	TouchForFood.Util.Category	70	8	12	1	15
701	TypeScope	CategoryUtil	70	8	12	1	15
702	MemberScope	CategoryUtil()	100	1	0		1

703	MemberScope	CategoryUtil()	94	1	1		1
704	MemberScope	filterListByMenu(menu) : IList<category>	58	6	12		13
705	NamespaceScope	TouchForFood.Util.Html	68	10	17	1	25
706	TypeScope	HtmlDropDownExtensions	68	10	17	1	25
707	MemberScope	EnumDropDownListFor<TModel, TEnum>(this HtmlHelper<TModel>, Expression	84	1	5		2
708	MemberScope	EnumDropDownListFor<TModel, TEnum>(this HtmlHelper<TModel>, Expression	59	3	13		10
709	MemberScope	GetEnumDescription<TEnum>(TEnum) : string	68	3	4		6
710	MemberScope	GetNonNullableModelType(ModelMetadata) : Type	70	2	3		6
711	MemberScope	HtmlDropDownExtensions()	88	1	1		1
712	NamespaceScope	TouchForFood.Util.Item	63	9	15	1	17
713	TypeScope	ItemUtil	63	9	15	1	17
714	MemberScope	filterListByItem(menu_category) : IList<item>	54	8	15		16
715	MemberScope	ItemUtil()	100	1	0		1
716	NamespaceScope	TouchForFood.Util.Order	82	18	19	1	43
717	TypeScope	OrderStatusUtil	95	2	1	1	2
720	TypeScope	OrderUtil	68	16	19	1	41
721	MemberScope	filterItem(menu_item) : item	77	1	4		3
722	MemberScope	filterMenuItem(order_item) : menu_item	74	1	5		4
723	MemberScope	filterTable(order) : table	77	1	5		3
724	MemberScope	filterUser(order) : user	77	1	5		3
725	MemberScope	filterWaiter(order) : waiter	77	1	5		3
726	MemberScope	mergeExistingOrderToDb(order) : void	48	9	13		23
727	MemberScope	OrderUtil()	100	1	0		1
728	MemberScope	OrderUtil()	94	1	1		1
729	NamespaceScope	TouchForFood.Util.Security	79	20	19	4	73
730	TypeScope	AES	67	14	11	1	62
740	TypeScope	CustomAuthorizationAttribute	70	6	7	4	11
743	TypeScope	SiteRoles	100	0	1	1	0
744	NamespaceScope	TouchForFood.Util.Session	56	5	15	1	27
745	TypeScope	SessionUtil	56	5	15	1	27
746	MemberScope	getOpenOrder(user) : order	47	4	15		26
747	MemberScope	SessionUtil()	100	1	0		1
748	NamespaceScope	TouchForFood.Util.User	77	4	8	1	10
749	TypeScope	UserUtil	77	4	8	1	10
750	MemberScope	getAuthenticatedUser(HttpRequestBase) : user	64	2	8		8
751	MemberScope	UserUtil()	100	1	0		1
752	MemberScope	UserUtil()	94	1	1		1
753	NamespaceScope	TouchForFood.ViewModels	76	24	13	1	36
754	TypeScope	OrderItemVM	76	1	2	1	4
756	TypeScope	OrderVM	76	23	11	1	32
757	MemberScope	addItem(OrderItemVM) : void	94	1	2		1
758	MemberScope	checkObjects() : void	64	8	3		7
759	MemberScope	getItemById(int) : OrderItemVM	71	4	5		5
760	MemberScope	orderItemVMs.get() : IList<OrderItemVM>	98	1	2		1
761	MemberScope	orderItemVMs.set(IList<OrderItemVM>) : void	95	1	2		1
762	MemberScope	OrderVM()	80	1	3		3
763	MemberScope	OrderVM(order)	73	2	3		5
764	MemberScope	OrderVM(order, IList<OrderItemVM>)	74	2	3		5
765	MemberScope	removeById(int) : void	78	2	2		3
766	MemberScope	removeItem(int) : void	94	1	2		1

Figure 14-5 Code Quality Report

14.6 Retrospective

In iteration 5, we completed 3 out of a planned 6 story points. Although this decreases our story point velocity, many other important issues were addressed and completed. As planned, we focused this sprint on testing, defects and missing documentation. The incomplete story was moved to the next sprint for completion.

14.6.1 Velocity

Sprint 5 velocity (story points): 3 story points
Cumulative velocity (story points): 34 story points
Average velocity (story points): 6.8 story points

Sprint 5 velocity (p-h): 289.17 person-hours
Cumulative velocity (p-h):805.872 person-hours
Average velocity per sprint:161.17 person-hours

It should be noted that velocity in person-hours was being calculated as just counting tasks with story points associated to them. From now on, it will be calculated to include all person-hours worked.

14.6.2 Budget

Total person-hours budgeted to date: 2,898 person-hours
Total person-hours worked to date: 805.872 person-hours

Person-hours budgeted per sprint: 414 person-hours
Person-hours worked in Sprint 5: 289.17 person-hours

The number of hours worked has increased each sprint and this was no exception. With our person-hour effort reaching almost 300 hours this iteration, many issues were taken care of and rapid progress is being made in terms of catching up to where we need to be in the project.

14.6.3 What was learned

The following positive and negative points were gathered from this sprint.

Positives:

- Completed 3 story points
- Greater team effort and contribution overall
- More organized and structured meetings
- Better processes established
- Many lingering issues (documents, defects, etc.) were dealt with

Negatives:

- Still need to catch up to get back on schedule

15 Iteration 6 Plan

In iteration 6, the goal is to continue working on increasing quality of our work while also moving the project forward. New user stories will be worked on while we will have one person dedicated to fixing bugs and one other person doing QA. To build off our last iteration, all new pages will include the UI theme that was implemented in sprint 5.

15.1 Planned Activities

Tables featured in this section of the document were created with JIRA [**2**].

Table 15-1 Planned Activities

Key	Issue Type	Priority	Assignee	Original Estimate (hours)
CAP-26	Story	Major	Josh Hum	
CAP-42	Story	Major	Ryan Nasr	
CAP-72	Sub-task	Major	Josh Hum	24
CAP-110	Task	Major	Unassigned	0
CAP-128	Bug	Trivial	Christian Daher	2
CAP-130	Sub-task	Major	Cynthia Donato	2
CAP-131	Sub-task	Major	Cristian Asenjo	4
CAP-134	Sub-task	Major	Mikhail Levkovsky	2
CAP-135	Sub-task	Major	Christian Daher	1
CAP-136	Sub-task	Major	Matthew Tam	4
CAP-137	Sub-task	Major	Katrina Anderson	3
CAP-139	Sub-task	Major	Ryan Nasr	3
CAP-141	Sub-task	Major	Josh Hum	3
CAP-142	Sub-task	Major	Patrick Modafferi	0.25
CAP-143	Sub-task	Major	Patrick Modafferi	0.25
CAP-144	Sub-task	Major	Cristian Asenjo	0.5
CAP-145	Sub-task	Major	Cynthia Donato	0.5
CAP-147	Sub-task	Major	Patrick Modafferi	0.5
CAP-148	Sub-task	Major	Matthew Tam	0.5
CAP-149	Sub-task	Major	Katrina Anderson	0.5
CAP-150	Sub-task	Major	Josh Hum	0.5
CAP-151	Sub-task	Major	Mikhail Levkovsky	0.5
CAP-152	Sub-task	Major	Christian Daher	0.5
CAP-153	Sub-task	Major	Ryan Nasr	0.083333333
CAP-154	Sub-task	Major	Ryan Nasr	0.083333333
CAP-155	Sub-task	Major	Katrina Anderson	0.5
CAP-169	Bug	Major	Cynthia Donato	5
CAP-194	Sub-task	Major	Unassigned	4
CAP-196	Sub-task	Major	Cristian Asenjo	12

CAP-198	Task	Major	Matthew Tam	0
CAP-199	Sub-task	Major	Matthew Tam	3
CAP-205	Sub-task	Major	Matthew Tam	5
CAP-206	Sub-task	Major	Matthew Tam	6
CAP-234	Sub-task	Minor	Unassigned	0
CAP-235	Bug	Critical	Christian Daher	2
CAP-236	Bug	Critical	Christian Daher	2
CAP-237	Bug	Critical	Christian Daher	2
CAP-238	Bug	Critical	Christian Daher	2
CAP-239	Bug	Critical	Christian Daher	2
CAP-240	Bug	Critical	Christian Daher	2
CAP-241	Bug	Critical	Christian Daher	2
CAP-242	Bug	Critical	Christian Daher	2
CAP-243	Bug	Critical	Christian Daher	2
CAP-246	Bug	Critical	Christian Daher	1
CAP-247	Task	Minor	Unassigned	0
CAP-250	Bug	Minor	Christian Daher	8
CAP-251	Bug	Minor	Unassigned	0
CAP-255	Improvement	Minor	Mikhail Levkovsky	4
CAP-261	Bug	Major	Ryan Nasr	8
CAP-263	Sub-task	Minor	Josh Hum	2
CAP-265	Sub-task	Major	Matthew Tam	10
CAP-267	Sub-task	Major	Patrick Modafferi	3
CAP-268	Sub-task	Major	Mikhail Levkovsky	3
CAP-269	Sub-task	Major	Christian Daher	3
CAP-270	Sub-task	Major	Cynthia Donato	3
CAP-271	Sub-task	Major	Ryan Nasr	3
CAP-272	Sub-task	Major	Cristian Asenjo	3
CAP-273	Sub-task	Major	Katrina Anderson	3
CAP-274	Bug	Major	Ryan Nasr	8
CAP-276	Bug	Minor	Cristian Asenjo	3
CAP-277	Bug	Critical	Christian Daher	3
CAP-278	Sub-task	Major	Josh Hum	16
CAP-281	Sub-task	Major	Mikhail Levkovsky	2
CAP-282	Sub-task	Major	Cynthia Donato	8
CAP-283	Sub-task	Major	Mikhail Levkovsky	1
CAP-284	Sub-task	Major	Katrina Anderson	8
CAP-285	Sub-task	Major	Katrina Anderson	2
CAP-286	Sub-task	Major	Cynthia Donato	2
CAP-287	Sub-task	Major	Cynthia Donato	2
CAP-288	Sub-task	Major	Unassigned	2

CAP-289	Sub-task	Major	Mikhail Levkovsky	2
CAP-290	Sub-task	Major	Mikhail Levkovsky	4
CAP-291	Sub-task	Major	Katrina Anderson	5
CAP-292	Sub-task	Major	Katrina Anderson	2
CAP-293	Sub-task	Major	Unassigned	3
CAP-294	Sub-task	Minor	Josh Hum	4
CAP-295	Sub-task	Minor	Mikhail Levkovsky	0.5
CAP-296	Sub-task	Minor	Josh Hum	0.75
CAP-297	Sub-task	Minor	Josh Hum	2
CAP-298	Sub-task	Minor	Josh Hum	2
CAP-299	Improvement	Major	Patrick Modafferi	0
CAP-300	Sub-task	Major	Ryan Nasr	5
CAP-301	Sub-task	Major	Ryan Nasr	8
CAP-302	Sub-task	Major	Christian Daher	4
CAP-303	Sub-task	Major	Ryan Nasr	12
CAP-304	Sub-task	Major	Ryan Nasr	6
CAP-305	Sub-task	Major	Unassigned	8
CAP-306	Sub-task	Major	Unassigned	8
CAP-307	Sub-task	Major	Unassigned	4
CAP-308	Bug	Major	Ryan Nasr	
			TOTAL	**297.62**

15.2 Person-Hour Estimation

The Expert Judgement Method was used to calculate person-hour estimations.

Description	Worst Case	Most Likely Case	Best Case	Expected Case
Activities	386.91	297.62	208.33	297.62
Total(ph)	386.91	297.62	208.33	297.62
Velocity(ph/day)	27.64	21.26	14.88	21.26
Velocity (ph/team member/day)	3.07	2.36	1.65	2.36

16 Iteration 6 Report

16.1 Person-Hour Work Log

Start Date: 16/Jan/13 End Date: 29/Jan/13 [Change] (UNREGISTERED)			Christian Daher	Cristian Asenjo	Cynthia Donato	Josh Hum	Katrina Anderson	Matthew Tam	Mikhail Levkovsky	Patrick Modafferi	Ryan Nasr	Total
Issue		Total	17.917h	20h	22.567h	54.883h	37.483h	25.15h	21.5h	21.617h	30.417h	251.533h
CAP-11	Documentation - SAD	1h	0h	0h	1h	0h	0h	0h	0h	0h	0h	1h
CAP-14	Correspondence	18.317h	0h	3.383h	1.3h	2.217h	6.833h	0.483h	1.717h	2.383h	0h	18.317h
CAP-18	Meetings	31.917h	1h	4h	1h	4.25h	3h	4.5h	4h	5.167h	5h	31.917h
CAP-23	Setup	3.667h	0h	0h	0.5h	0h	0h	0h	0h	3.167h	0h	3.667h
CAP-26	View Menu	32.083h	0h	0h	0h	32.083h	0h	0h	0h	0h	0h	32.083h
CAP-34	Call Waiter	34.75h	0h	0h	9h	0.333h	24.167h	0h	0.5h	0h	0.75h	34.75h
CAP-40	Sign in	5.65h	0h	5.65h	0h	0h	0h	0h	0h	0h	0h	5.65h
CAP-42	Restaurant Bill Management	26.117h	13h	0h	0h	0h	0h	0h	0h	0h	13.117h	26.117h
CAP-57	Documentation - Management	9.75h	0h	0h	0h	9.5h	0h	0h	0.25h	0h	0h	9.75h
CAP-110	Documentation - UIR	18.3h	0.417h	4.483h	3.033h	1.667h	0.983h	3.167h	1.7h	0.317h	2.533h	18.3h
CAP-162	Look and Feel	1h	0h	0h	0h	0h	0h	1h	0h	0h	0h	1h
CAP-164	Documentation - Test Report	11.567h	2h	1.517h	0.767h	2h	2.5h	0h	0.767h	0h	2.017h	11.567h
CAP-198	Unit Testing Retroactive	6h	0h	0h	0h	0h	0h	6h	0h	0h	0h	6h
CAP-235	FK Issue for Deleting Restaurants	0.017h	0h	0h	0h	0h	0h	0h	0.017h	0h	0h	0.017h
CAP-236	FK Issue for Deleting Menu	0.017h	0h	0h	0h	0h	0h	0h	0.017h	0h	0h	0.017h
CAP-237	FK constraint - User delete	0.017h	0h	0h	0h	0h	0h	0h	0.017h	0h	0h	0.017h
CAP-238	FK Issue for Deleting Category	0.017h	0h	0h	0h	0h	0h	0h	0.017h	0h	0h	0.017h
CAP-239	FK constraint - order delete	0.017h	0h	0h	0h	0h	0h	0h	0.017h	0h	0h	0.017h
CAP-240	FK constraint - waiter delete	0.017h	0h	0h	0h	0h	0h	0h	0.017h	0h	0h	0.017h
CAP-241	FK constraint - Review delete	0.017h	0h	0h	0h	0h	0h	0h	0.017h	0h	0h	0.017h
CAP-242	FK Issue for Deleting Menu Item	0.017h	0h	0h	0h	0h	0h	0h	0.017h	0h	0h	0.017h
CAP-243	FK constraint - Table delete	0.017h	0h	0h	0h	0h	0h	0h	0.017h	0h	0h	0.017h
CAP-246	Removing order item causes price to go below 0	1h	1h	0h	0h	0h	0h	0h	0h	0h	0h	1h
CAP-255	is_active and is_deleted field add extra complexity, factor that out	16.867h	0h	0h	5.967h	0h	0h	0h	10.9h	0h	0h	16.867h
CAP-261	Database context inconsistent	1h	0h	0h	0h	0h	0h	0h	0h	0h	1h	1h
CAP-264	User Interface Testing	8h	0h	0h	0h	0h	0h	8h	0h	0h	0h	8h
CAP-276	Custom routing has to be removed or fixed to work when logging in (CustomAuthorize is affecting it)	0.167h	0h	0.167h	0h	0h	0h	0h	0h	0h	0h	0.167h
CAP-277	Order half disappears	0.5h	0.5h	0h	0h	0h	0h	0h	0h	0h	0h	0.5h
CAP-280	Documentation - DB backup instructions	2h	0h	0h	0h	0h	0h	2h	0h	0h	0h	2h
CAP-299	Flow and Improvements	11.833h	0h	0h	0h	0h	0h	0h	1.5h	10.333h	0h	11.833h
CAP-308	The review controller should receive the order_id instead of the restaurant_id	0.083h	0h	0h	0h	0h	0h	0h	0h	0.083h	0h	0.083h
CAP-314	Code Review	8.833h	0h	0h	0h	2.833h	0h	0h	0h	0h	6h	8.833h

CAP-315	null user role		**0.8h**	0h	0.8h	0h	0h	0h	0h	0h	0h	0h	**0.8h**
CAP-319	Decline order not working		**0.167h**	0h	0h	0h	0h	0h	0h	0h	0.167h	0h	**0.167h**
CAP-339	String Localization		**0.017h**	0h	0h	0h	0h	0h	0h	0.017h	0h	0h	**0.017h**

Figure 16-1 Person-Hour Work Log

The above table shows the person-hours spent on various tasks in Sprint 6. The effort continued to be high as the team worked 251 hours in total. These hours include overhead tasks such as communication and setup.

16.2 Hour Burndown Chart

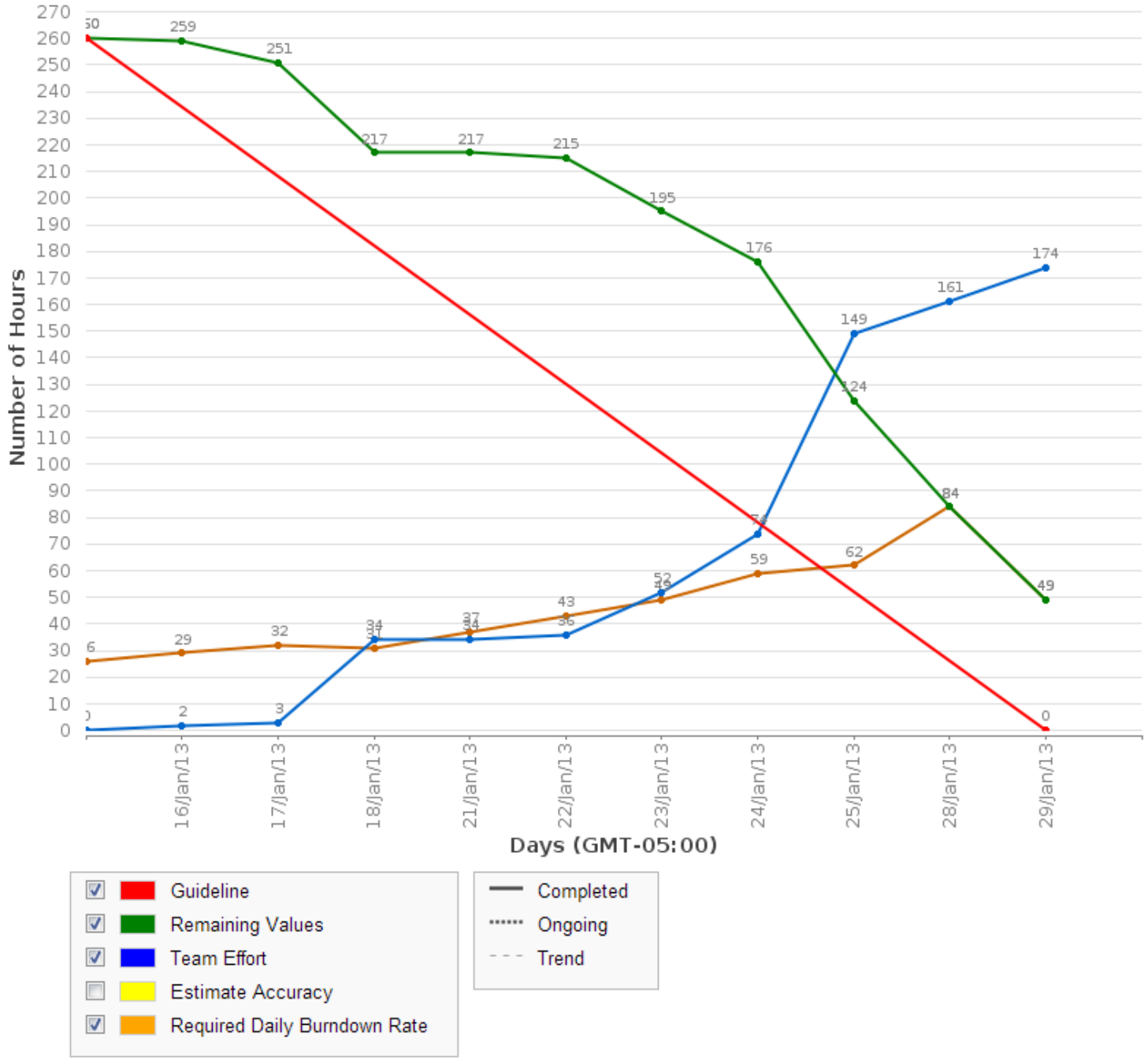

Figure 16-2 Hour Burndown Chart

Approximately 210 of the 260 estimated hours were completed. Some hours remained as a few tasks had to be carried over into Iteration 7. As seen in the chart, the team effort was pretty consistent with a little more effort being put in at the end of the sprint.

16.3 Issue Burndown Chart

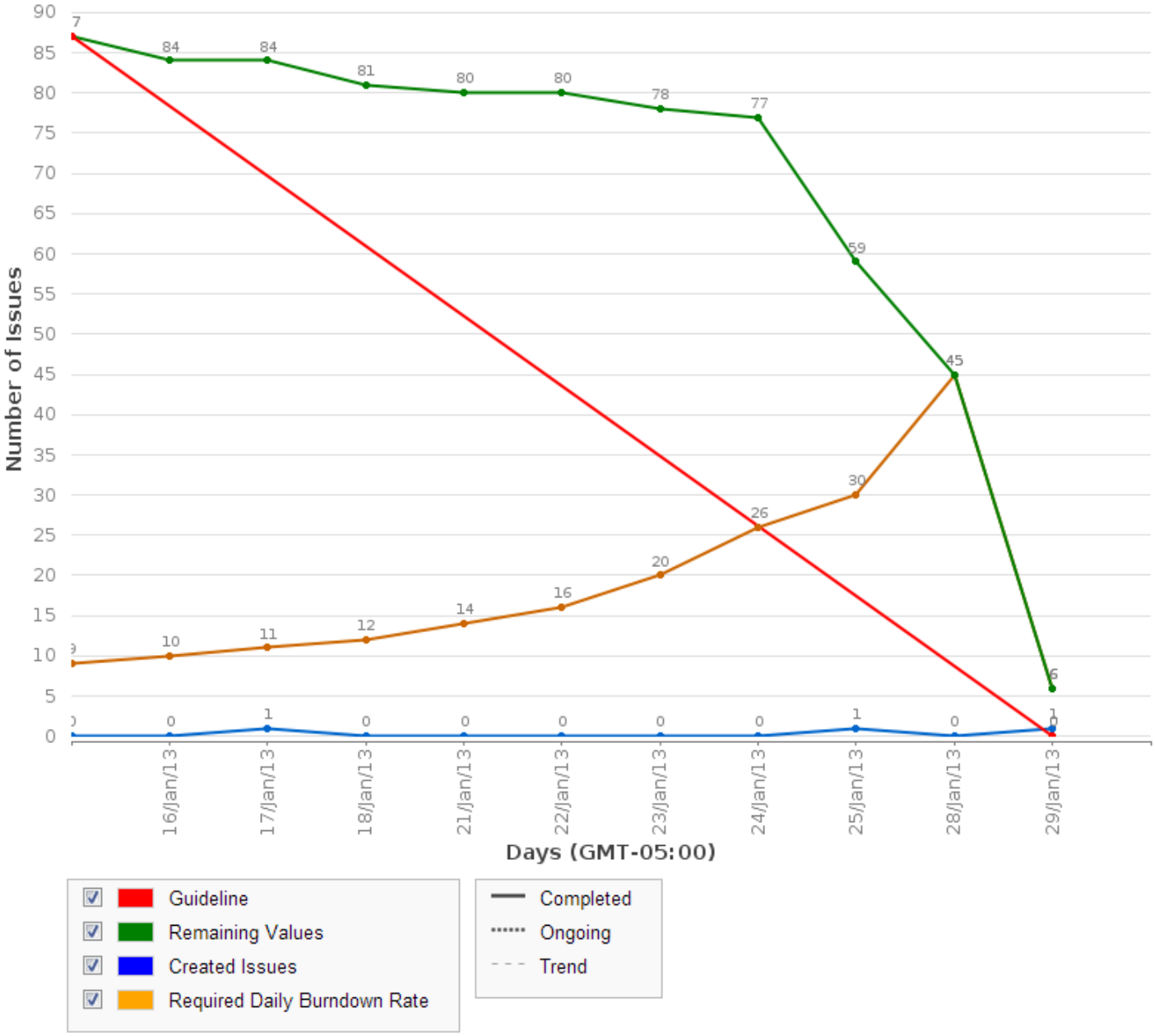

Figure 16-3 Issue Burndown Chart

Considering that 81 out of the 87 planned issues were burned down, we are happy with our progress. The two major issues not completed were CAP-350 – Search Menu and CAP-34 – Call Waiter. However, the majority of the work on them was done and they will be completed in Iteration 7. After discussing in our weekly meeting, it was noted that a few of the issues completed earlier in the sprint were not marked as resolved which is why the line is fairly flat and then shoots down at the end.

16.4 Cumulative Flow Diagram

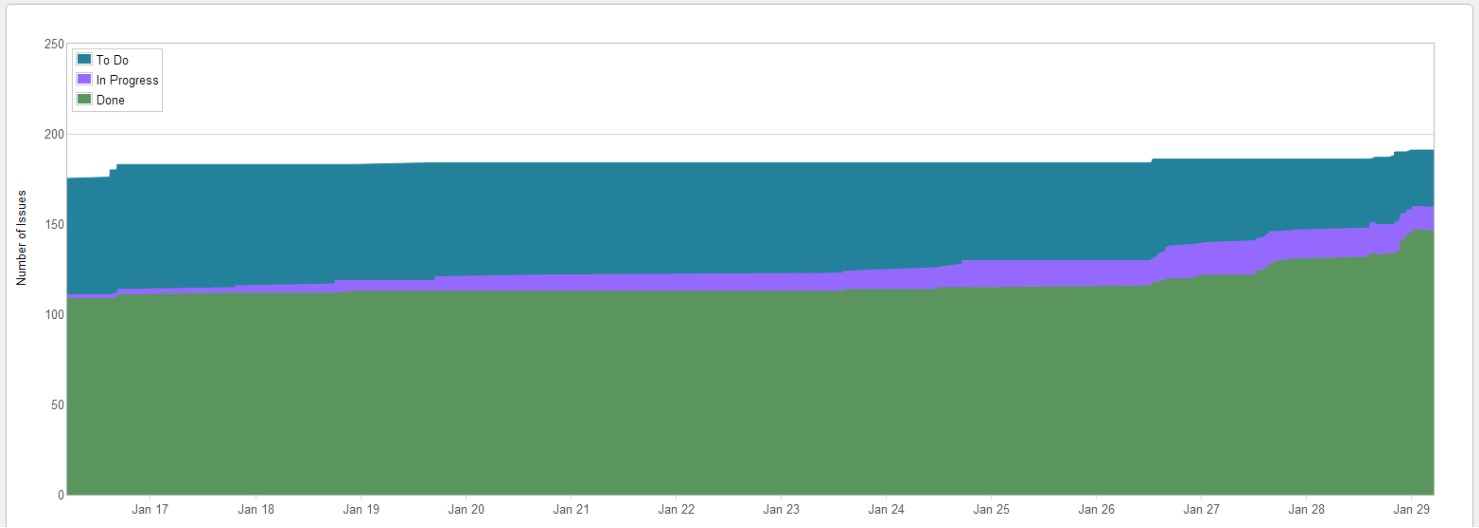

Figure 16-4 Cumulative Flow Diagram

Since each new task that we take on gets new subtasks created, there is always an increase in “to do” tasks at the end of the sprint as we start planning the next one. However, progress is continuous and the tasks that are done are starting to catch up to the tasks to do.

16.5 Measurement Report

16.5.1 Code Quality Analysis

The following report was generated after analyzing the code. Five methods did not meet code quality goals by exceeding the expected number of lines of code. Four of these were found in an earlier iteration and have not been yet corrected while the new instance was introduced in this sprint. All of them have been entered as bugs and will be fixed in a future iteration. Since it is a minor bug, the priority is not high for it.

Since our only issues have been with lines of code, we took note for the future that the following scale defines lines of code quality:

0: red
1 – 10: green
11 – 10: yellow
21+: red

Analysis tool used: Code Metrics Viewer
Found at: http://visualstudiogallery.msdn.microsoft.com/9f35524b-a784-4dbc-bd7b-6babd7a5a3b3
Version: 1.5.3
Last updated: 2/5/2012

Scope	Hiearchy	Maintainability Index	Cyclomatic Complexity	Depth of Inheritance	Class Coupling	Lines of Code
ModuleScope	TouchForFood.Tests.dll	73	185	100	2	691
ModuleScope	TouchForFood.dll	80		178	4	
NamespaceScope	TouchForFood	80	4	9	2	11
TypeScope	MvcApplication	80	4	9	2	11
MemberScope	Application_Start() : void	80	1	3		3
MemberScope	MvcApplication()	100	1	1		1
MemberScope	RegisterGlobalFilters(GlobalFilterCollection) : v	94	1	2		1
MemberScope	RegisterRoutes(RouteCollection) : void	66	1	3		6
NamespaceScope	TouchForFood.Attributes	84	2	5	3	4
TypeScope	AjaxAttribute	84	2	5	3	4
MemberScope	AjaxAttribute(bool)	87	1	1		2
MemberScope	IsValidForRequest(ControllerContext, MethodI	83	1	4		2
NamespaceScope	TouchForFood.Controllers	66	502	110	3	
TypeScope	BillController	68	29	40	3	80
TypeScope	CategoryController	71	19	29	3	60
TypeScope	FriendshipController	69	20	29	3	42
TypeScope	HomeController	74	5	15	3	6
TypeScope	ItemController	70	25	40	3	74
TypeScope	Menu_CategoryController	67	41	31	3	85
TypeScope	Menu_ItemController	69	25	34	3	59
TypeScope	MenuController	67	39	44	3	81
TypeScope	Order_ItemController	66	28	40	3	66
TypeScope	OrderController	60	80	57	3	208
TypeScope	RestaurantController	72	16	25	3	41
TypeScope	ReviewController	62	14	31	3	34
TypeScope	SearchController	56	33	19	3	75
TypeScope	ServiceRequestController	62	67	45	3	132
TypeScope	TableController	66	35	46	3	69
TypeScope	UserController	64	26	45	3	101
MemberScope	Create() : ActionResult	74	2	9		3
MemberScope	Create(user) : ActionResult	51	5	20		21

MemberScope	Delete(int) : ActionResult	77	1	6		3
MemberScope	DeleteConfirmed(int) : ActionResult	71	1	9		5
MemberScope	Details(int) : ViewResult	77	1	6		3
MemberScope	Dispose(bool) : void	87	1	3		2
MemberScope	Edit(int) : ActionResult	77	1	6		3
MemberScope	Edit(user, HttpPostedFileBase) : ActionResult	50	6	22		19
MemberScope	Index() : ViewResult	73	2	13		3
MemberScope	LogOff() : ActionResult	76	1	5		4
MemberScope	LogOn() : ViewResult	87	1	2		2
MemberScope	LogOn(string, string) : ActionResult	46	3	24		32
MemberScope	UserController()	92	1	2		1
NamespaceScope	TouchForFood.Exceptions	97	4	1	2	4
TypeScope	AssociationExistsException	97	2	1	2	2
TypeScope	ItemActiveException	97	2	1	2	2
MemberScope	ItemActiveException()	100	1	1		1
MemberScope	ItemActiveException(string)	98	1	1		1
NamespaceScope	TouchForFood.Mappers	77	159	55	2	372
TypeScope	CategoryIM	84	6	6	2	10
TypeScope	CategoryOM	70	7	13	2	22
TypeScope	ItemIM	84	6	6	2	10
TypeScope	ItemOM	77	4	8	2	10
TypeScope	MenuCategoryIM	80	6	13	2	10
TypeScope	MenuCategoryOM	65	8	15	2	30
TypeScope	MenuIM	80	6	13	2	10
TypeScope	MenuItemIM	80	6	13	2	10
TypeScope	MenuItemOM	70	8	13	2	22
TypeScope	MenuOM	65	8	15	2	30
TypeScope	OrderIM	84	6	6	2	10
TypeScope	OrderOM	65	11	14	2	31
TypeScope	RestaurantIM	81	6	13	2	10
TypeScope	RestaurantOM	71	18	24	2	40
TypeScope	ReviewIM	84	6	6	2	10
TypeScope	ReviewOM	71	7	14	2	19
TypeScope	TableIM	84	6	6	2	10
TypeScope	TableOM	75	9	12	2	23
TypeScope	UserIM	84	6	6	2	10
TypeScope	UserOM	72	8	15	2	23
TypeScope	WaiterIM	84	6	6	2	10
TypeScope	WaiterOM	76	5	10	2	12
MemberScope	clearOrder(ICollection<order>) : void	75	2	8		3
MemberScope	delete(int) : int	67	1	5		7
MemberScope	WaiterOM()	100	1	1		1
MemberScope	WaiterOM(touch_for_foodEntities)	98	1	2		1
NamespaceScope	TouchForFood.Mappers.Abstract	90	5	1	1	8
TypeScope	GenericIM	86	2	1	1	4
TypeScope	GenericOM	94	3	1	1	4
MemberScope	delete(int) : int	100	1	0		0
MemberScope	GenericOM()	86	1	1		2
MemberScope	GenericOM(touch_for_foodEntities)	87	1	1		2
NamespaceScope	TouchForFood.Models	92	453	46	2	544
TypeScope	bill	92	21	7	1	22
TypeScope	category	92	11	4	1	13
TypeScope	CategoryFilterVM	83	9	6	1	16
TypeScope	friendship	93	11	2	1	11
TypeScope	item	92	19	5	1	20
TypeScope	ItemFilterVM	80	18	16	1	34
TypeScope	menu	92	17	5	1	18
TypeScope	menu_category	92	19	5	1	20
TypeScope	menu_item	92	21	6	1	22
TypeScope	MenuMetadata	94	7	4	1	7
TypeScope	order	89	35	14	1	58
TypeScope	order_item	92	24	6	1	26
TypeScope	OrderItemMetadata	100	1	0	1	1
TypeScope	OrderMetadata	100	1	0	1	1

TypeScope	OrderStatusHelper	65	25	7	1	44
TypeScope	OrderStatusHelper.OrderItemStatusEnum	100	0	0	1	0
TypeScope	OrderStatusHelper.OrderStatusEnum	100	0	0	1	0
TypeScope	restaurant	91	27	10	1	32
TypeScope	restaurant_user	93	11	3	1	11
TypeScope	RestaurantMetadata	94	9	1	1	9
TypeScope	review	92	25	7	1	25
TypeScope	ReviewMetadata	93	11	5	1	11
TypeScope	service_request	93	15	3	1	15
TypeScope	table	91	16	9	1	22
TypeScope	TableMetadata	100	1	0	1	1
TypeScope	touch_for_foodEntities	92	34	20	2	34
TypeScope	user	91	35	9	1	40
TypeScope	UserMetadata	93	15	5	1	15
TypeScope	waiter	93	15	5	1	16
MemberScope	first_name.get() : string	98	1	0		1
MemberScope	first_name.set(string) : void	95	1	0		1
MemberScope	id.get() : int	98	1	0		1
MemberScope	id.set(int) : void	95	1	0		1
MemberScope	last_name.get() : string	98	1	0		1
MemberScope	last_name.set(string) : void	95	1	0		1
MemberScope	orders.get() : ICollection<order>	98	1	2		1
MemberScope	orders.set(ICollection<order>) : void	95	1	2		1
MemberScope	restaurant.get() : restaurant	98	1	1		1
MemberScope	restaurant.set(restaurant) : void	95	1	1		1
MemberScope	resto_id.get() : int?	98	1	1		1
MemberScope	resto_id.set(int?) : void	95	1	1		1
MemberScope	version.get() : int	98	1	0		1
MemberScope	version.set(int) : void	95	1	0		1
MemberScope	waiter()	87	1	2		2
NamespaceScope	TouchForFood.Util.Bill	74	13	9	1	15
TypeScope	BillUtil	74	13	9	1	15
MemberScope	GetTotalAfterTax(bill) : decimal	78	3	3		2
MemberScope	GetTotalBeforeTax(bill) : decimal	85	1	3		2
MemberScope	GetTPS() : decimal	84	1	1		2
MemberScope	GetTVQ() : decimal	84	1	1		2
MemberScope	Update(ref bill) : void	62	7	9		7
NamespaceScope	TouchForFood.Util.Category	70	8	12	1	15
TypeScope	CategoryUtil	70	8	12	1	15
MemberScope	CategoryUtil()	100	1	0		1
MemberScope	CategoryUtil()	94	1	1		1
MemberScope	filterListByMenu(menu) : IList<category>	58	6	12		13
NamespaceScope	TouchForFood.Util.Html	69	16	26	1	39
TypeScope	HtmlDropDownExtensions	68	10	17	1	25
TypeScope	ImageActionLinkHelper	64	1	7	1	8
TypeScope	UrlUtils	75	5	4	1	6
MemberScope	ConvertRelativeUrlToAbsoluteUrl(string) : strin	67	4	4		5
MemberScope	UrlUtils()	100	1	0		1
NamespaceScope	TouchForFood.Util.Item	63	9	15	1	17
TypeScope	ItemUtil	63	9	15	1	17
MemberScope	filterListByItem(menu_category) : IList<item>	54	8	15		16
MemberScope	ItemUtil()	100	1	0		1
NamespaceScope	TouchForFood.Util.Order	82	22	20	1	49
TypeScope	OrderStatusUtil	95	2	1	1	2
TypeScope	OrderUtil	68	20	20	1	47
MemberScope	filterItem(menu_item) : item	77	1	4		3
MemberScope	filterMenuItem(order_item) : menu_item	74	1	5		4
MemberScope	filterTable(order) : table	77	1	5		3
MemberScope	filterUser(order) : user	77	1	5		3
MemberScope	filterWaiter(order) : waiter	77	1	5		3
MemberScope	mergeExistingOrderToDb(order) : void	48	9	13		23
MemberScope	OrderUtil()	100	1	0		1
MemberScope	OrderUtil()	94	1	1		1
MemberScope	UpdatePrice(ref order) : void	65	4	11		6

NamespaceScope	TouchForFood.Util.Search	66	8	19	1	43
TypeScope	SearchService	53	3	18	1	37
TypeScope	SearchViewModelHelper	78	5	9	1	6
MemberScope	PopulateSearchViewModel(menu_item) : Sear	77	1	6		2
MemberScope	PopulateSearchViewModelList(IList<menu_iter	76	3	5		3
MemberScope	SearchViewModelHelper()	100	1	0		1
NamespaceScope	TouchForFood.Util.Security	79	20	19	4	73
TypeScope	AES	67	14	11	1	62
TypeScope	CustomAuthorizationAttribute	70	6	7	4	11
TypeScope	SiteRoles	100	0	1	1	0
NamespaceScope	TouchForFood.Util.ServiceRequest	100	1	0	1	1
TypeScope	ServiceRequestUtil	100	1	0	1	1
TypeScope	ServiceRequestUtil.ServiceRequestStatus	100	0	0	1	0
NamespaceScope	TouchForFood.Util.Session	55	5	15	1	30
TypeScope	SessionUtil	55	5	15	1	30
MemberScope	getOpenOrder(user) : order	46	4	15		29
MemberScope	SessionUtil()	100	1	0		1
NamespaceScope	TouchForFood.Util.User	77	5	11	1	12
TypeScope	UserUtil	77	5	11	1	12
MemberScope	getAuthenticatedUser(HttpRequestBase) : user	64	2	8		8
MemberScope	isUserRole(SiteRoles, HttpContext) : bool	84	1	3		2
MemberScope	UserUtil()	100	1	0		1
MemberScope	UserUtil()	94	1	1		1
NamespaceScope	TouchForFood.ViewModels	81	38	13	1	56
TypeScope	OrderItemVM	76	1	2	1	4
TypeScope	OrderVM	76	23	11	1	32
TypeScope	SearchViewModel	92	14	0	1	20
MemberScope	catName.get() : string	98	1	0		1
MemberScope	catName.set(string) : void	95	1	0		1
MemberScope	description.get() : string	98	1	0		1
MemberScope	description.set(string) : void	95	1	0		1
MemberScope	menuItemId.get() : int	98	1	0		1
MemberScope	menuItemId.set(int) : void	95	1	0		1
MemberScope	menuName.get() : string	98	1	0		1
MemberScope	menuName.set(string) : void	95	1	0		1
MemberScope	metadata.get() : string	98	1	0		1
MemberScope	metadata.set(string) : void	95	1	0		1
MemberScope	name.get() : string	98	1	0		1
MemberScope	name.set(string) : void	95	1	0		1
MemberScope	SearchViewModel()	100	1	0		1
MemberScope	SearchViewModel(int, string, string, string, stri	70	1	0		7

Figure 16-5 Code Quality Report

16.6 Retrospective

In iteration 6, we completed 11 out of a planned 15 story points. The two stories CAP-350 – Search Menu and CAP-34 – Call Waiter were added to the next sprint as they were almost but not yet complete. We also introduced the official roles of bug basher and QA. From Sprint 6 and onward, everyone in the team will take a turn being one of these two positions. There was also one person appointed to working on the overall flow of the application. Code reviews were also established. Everyone works in their own branch and code has to be reviewed before being committed back to the trunk. There are three designated code reviewers per sprint although anyone can review code and leave feedback.

See Appendix B Glossary for definitions of Bug Basher, QA, and flow fixer.

16.6.1 Velocity

Sprint 6 velocity (story points): 11 story points
Cumulative velocity (story points): 45 story points
Average velocity (story points): 7.5 story points

Sprint 6 velocity (p-h): 251.53 person-hours
Cumulative velocity (p-h):1,057.40 person-hours
Average velocity per sprint:176.23 person-hours

Our average velocity in story points increased this iteration as did our average velocity in person-hours. The amount of effort is staying consistent above 200 hours. If we continue at this rate, we should be able to complete the project on time.

16.6.2 Budget

Total person-hours budgeted to date: 3,312 person-hours
Total person-hours worked to date: 1,057.40 person-hours

Person-hours budgeted per sprint: 414 person-hours
Person-hours worked in Sprint 6: 251.53 person-hours

Since the budget was over-estimated, our new goal is to work at least 200 hours per sprint. If this goal is reached, the project should be finished on time. In Iteration 6, we met this goal.

16.6.3 What was learned

The following positive and negative points were gathered from this sprint.

Positives:

- Completed 11 story points
- New roles such as QA, Bug basher, Flow fixer, and code reviewers improved the overall quality of the project
- UIR was created
- Back on schedule to finish the project in time

Negatives:

- None

17 Iteration 7 Plan

The goal of Iteration 7 is to continue our current progress on the project while also start preparing for presentations. We will be having meetings with our stakeholder and project supervisor during the sprint. We will also hand in deliverable 2 at the end of the sprint. Thus, the focus of the sprint is to make our documentation consistent and prepare for our meetings. We will continue to finish open stories and work on new stories. We will also have people working on QA, bug bashing, fixing flow, and reviewing code.

17.1 Planned Activities

Tables featured in this section of the document were created with JIRA [**2**].

Table 17-1 Planned Activities

Key	Issue Type	Priority	Assignee	Original Estimate (hours)
CAP-128	Bug	Trivial	Mikhail Levkovsky	2
CAP-234	Sub-task	Minor	Cynthia Donato	1
CAP-247	Task	Minor	Unassigned	0
CAP-251	Bug	Minor	Unassigned	0
CAP-255	Improvement	Minor	Cynthia Donato	4
CAP-263	Sub-task	Minor	Josh Hum	2
CAP-276	Bug	Minor	Cristian Asenjo	3
CAP-294	Sub-task	Minor	Josh Hum	4
CAP-295	Sub-task	Minor	Mikhail Levkovsky	0.5
CAP-296	Sub-task	Minor	Josh Hum	0.75
CAP-297	Sub-task	Minor	Josh Hum	2
CAP-298	Sub-task	Minor	Josh Hum	2
CAP-375	Sub-task	Minor	Josh Hum	32
CAP-26	Story	Major	Josh Hum	0
CAP-42	Story	Major	Ryan Nasr	0
CAP-110	Task	Major	Unassigned	0
CAP-130	Sub-task	Major	Cynthia Donato	2
CAP-131	Sub-task	Major	Cristian Asenjo	4
CAP-134	Sub-task	Major	Mikhail Levkovsky	2
CAP-135	Sub-task	Major	Christian Daher	1
CAP-136	Sub-task	Major	Matthew Tam	4
CAP-137	Sub-task	Major	Katrina Anderson	3
CAP-139	Sub-task	Major	Ryan Nasr	3
CAP-141	Sub-task	Major	Josh Hum	3
CAP-142	Sub-task	Major	Patrick Modafferi	0.25
CAP-143	Sub-task	Major	Patrick Modafferi	0.25
CAP-144	Sub-task	Major	Cristian Asenjo	0.5

CAP-145	Sub-task	Major	Cynthia Donato	0.5
CAP-147	Sub-task	Major	Patrick Modafferi	0.5
CAP-148	Sub-task	Major	Matthew Tam	0.5
CAP-149	Sub-task	Major	Katrina Anderson	0.5
CAP-150	Sub-task	Major	Josh Hum	0.5
CAP-151	Sub-task	Major	Mikhail Levkovsky	0.5
CAP-152	Sub-task	Major	Christian Daher	0.5
CAP-153	Sub-task	Major	Ryan Nasr	0.083333333
CAP-154	Sub-task	Major	Ryan Nasr	0.083333333
CAP-155	Sub-task	Major	Katrina Anderson	0.5
CAP-169	Bug	Major	Cynthia Donato	5
CAP-194	Sub-task	Major	Cristian Asenjo	4
CAP-196	Sub-task	Major	Cristian Asenjo	12
CAP-198	Task	Major	Katrina Anderson	0
CAP-199	Sub-task	Major	Katrina Anderson	3
CAP-205	Sub-task	Major	Katrina Anderson	5
CAP-206	Sub-task	Major	Katrina Anderson	6
CAP-261	Bug	Major	Ryan Nasr	8
CAP-265	Sub-task	Major	Matthew Tam	10
CAP-267	Sub-task	Major	Patrick Modafferi	3
CAP-268	Sub-task	Major	Mikhail Levkovsky	3
CAP-269	Sub-task	Major	Christian Daher	3
CAP-270	Sub-task	Major	Cynthia Donato	3
CAP-271	Sub-task	Major	Ryan Nasr	3
CAP-272	Sub-task	Major	Cristian Asenjo	3
CAP-273	Sub-task	Major	Katrina Anderson	3
CAP-274	Bug	Major	Mikhail Levkovsky	8
CAP-278	Sub-task	Major	Josh Hum	16
CAP-281	Sub-task	Major	Mikhail Levkovsky	2
CAP-282	Sub-task	Major	Katrina Anderson	8
CAP-283	Sub-task	Major	Mikhail Levkovsky	1
CAP-284	Sub-task	Major	Katrina Anderson	8
CAP-285	Sub-task	Major	Katrina Anderson	2
CAP-286	Sub-task	Major	Cynthia Donato	2
CAP-287	Sub-task	Major	Cynthia Donato	2
CAP-288	Sub-task	Major	Ryan Nasr	2
CAP-289	Sub-task	Major	Mikhail Levkovsky	2
CAP-290	Sub-task	Major	Mikhail Levkovsky	4
CAP-291	Sub-task	Major	Katrina Anderson	5
CAP-292	Sub-task	Major	Katrina Anderson	2
CAP-293	Sub-task	Major	Cynthia Donato	3

CAP-299	Improvement	Major	Josh Hum	0
CAP-300	Sub-task	Major	Ryan Nasr	5
CAP-301	Sub-task	Major	Ryan Nasr	8
CAP-302	Sub-task	Major	Christian Daher	4
CAP-303	Sub-task	Major	Ryan Nasr	12
CAP-304	Sub-task	Major	Ryan Nasr	6
CAP-305	Sub-task	Major	Patrick Modafferi	8
CAP-306	Sub-task	Major	Patrick Modafferi	8
CAP-307	Sub-task	Major	Patrick Modafferi	4
CAP-308	Bug	Major	Ryan Nasr	0
CAP-350	Sub-task	Major	Josh Hum	24
CAP-235	Bug	Critical	Christian Daher	2
CAP-236	Bug	Critical	Christian Daher	2
CAP-237	Bug	Critical	Christian Daher	2
CAP-238	Bug	Critical	Christian Daher	2
CAP-239	Bug	Critical	Christian Daher	2
CAP-240	Bug	Critical	Christian Daher	2
CAP-241	Bug	Critical	Christian Daher	2
CAP-242	Bug	Critical	Christian Daher	2
CAP-243	Bug	Critical	Christian Daher	2
CAP-246	Bug	Critical	Christian Daher	1
CAP-277	Bug	Critical	Christian Daher	3
CAP-319	Bug	Blocker	Unassigned	2
			TOTAL	**324.42**

17.2 Person-Hour Estimation

The Expert Judgement Method was used to calculate person-hour estimations.

Table 17-2 Person-Hour Estimation

Description	Worst Case	Most Likely Case	Best Case	Expected Case
Activities	421.75	324.42	227.09	324.42
Total(ph)	421.75	324.42	227.09	324.42
Velocity(ph/day)	30.12	23.17	16.22	23.17
Velocity (ph/team member/day)	3.35	2.57	1.80	2.57

17.3 Activity-on-Node Planning

The following graph reflects the progress of the planned user stories. The numbers represent the Early Start, Duration, Early Finish, Late Start, Slack, and Late Finish in days. View Menu was reopened to do search functionality. Toppings and Sides should have been completed earlier but it was missed. Both Customer and Restaurant Bill Management will have to be slightly edited after Toppings and Sides are complete.

Figure 17-1 Activity-on-Node Diagram

18 Iteration 7 Report

18.1 Person-Hour Work Log

Start Date: 30/Jan/13 End Date: 12/Feb/13 [Change] (UNREGISTERED)				Christian Daher	Cristian Asenjo	Cynthia Donato	Josh Hum	Katrina Anderson	Matthew Tam	Mikhail Levkovsky	Patrick Modafferi	Ryan Nasr	Total
Issue			Total	11.767h	7.633h	26.883h	39.2h	28.367h	6.7h	20.067h	17.167h	13.033h	170.817h
CAP-11	Documentation - SAD	↑	13.867h	3h	0.75h	4.333h	0.5h	0.017h	2h	1.267h	1h	1h	13.867h
CAP-14	Correspondence	↑	13.1h	0h	0.25h	0.667h	3.6h	4.833h	0.75h	0.083h	1.917h	1h	13.1h
CAP-18	Meetings	↑	26.05h	1.5h	1.7h	4.033h	4h	3.2h	2.95h	4.25h	2.917h	1.5h	26.05h
CAP-23	Setup	↑	9.667h	0h	0h	0.25h	2.833h	0.167h	0h	0h	4.417h	2h	9.667h
CAP-26	View Menu	↑	13.25h	0h	0h	0h	13.25h	0h	0h	0h	0h	0h	13.25h
CAP-34	Call Waiter	↓	8.25h	0h	0h	0h	0h	8.25h	0h	0h	0h	0h	8.25h
CAP-39	Leave Review	↑	11.883h	0h	0h	0h	0h	0h	0h	5.55h	6.333h	0h	11.883h
CAP-41	Customer Bill Management	↓	14.05h	7.017h	0h	0h	0h	0h	0h	0h	0h	7.033h	14.05h
CAP-44	Documentation - Test Plan	↓	0.167h	0h	0h	0h	0h	0h	0h	0.167h	0h	0h	0.167h
CAP-57	Documentation - Management	↑	12.083h	0h	0h	0h	11.917h	0h	0h	0.167h	0h	0h	12.083h
CAP-112	Toppings and Sides	↑	1h	0h	0h	0h	0h	0h	1h	0h	0h	0h	1h
CAP-128	The View Model Model (VMM) is not reflected in the SAD	↓	0.583h	0h	0h	0h	0h	0h	0h	0h	0.583h	0h	0.583h
CAP-162	Look and Feel	↑	0.017h	0h	0h	0h	0.017h	0h	0h	0h	0h	0h	0.017h
CAP-164	Documentation - Test Report	↑	5.75h	0h	0h	0h	0h	5.75h	0h	0h	0h	0h	5.75h
CAP-169	Optimistic offline locks logic missing from orders	↑	0.667h	0h	0h	0.667h	0h	0h	0h	0h	0h	0h	0.667h
CAP-164	Documentation - Test Report	↑	5.75h	0h	0h	0h	0h	5.75h	0h	0h	0h	0h	5.75h
CAP-169	Optimistic offline locks logic missing from orders	↑	0.667h	0h	0h	0.667h	0h	0h	0h	0h	0h	0h	0.667h
CAP-255	is_active and is_deleted field add extra complexity, factor that out	↓	9.417h	0h	0h	8.667h	0h	0.75h	0h	0h	0h	0h	9.417h
CAP-264	User Interface Testing	↑	3.5h	0h	0h	0h	0h	3.5h	0h	0h	0h	0h	3.5h
CAP-299	Flow and Improvements	↑	0.833h	0h	0h	0h	0.833h	0h	0h	0h	0h	0h	0.833h
CAP-313	The order should not have a 'VIEW_STATUS' status when a user wants to view the order.	⇑	1.017h	0h	0h	0h	0h	0h	0h	1.017h	0h	0h	1.017h
CAP-314	Code Review	↑	3h	0.25h	0h	0h	1.25h	0h	0h	1h	0h	0.5h	3h
CAP-316	Cancelling an order crashes the application	↑	0.517h	0h	0h	0h	0h	0h	0h	0.517h	0h	0h	0.517h
CAP-317	The roles for 'Order/Finalize' are invalid.	↑	0.017h	0h	0h	0h	0h	0h	0h	0.017h	0h	0h	0.017h
CAP-321	When logged in as a user (resto or customer), you can access the edit page for any other user	↑	0.383h	0h	0.383h	0h	0h	0h	0h	0h	0h	0h	0.383h
CAP-322	Viewing menu details as a customer will not work when site-wide authentication is in place	↑	0.017h	0h	0.017h	0h	0h	0h	0h	0h	0h	0h	0.017h
CAP-323	Menu category details shows editing links when viewing as a normal customer	↓	0.017h	0h	0.017h	0h	0h	0h	0h	0h	0h	0h	0.017h
CAP-324	Removing an order item from an order removes it right away and doesn't wait for the finalize button to be tapped	↓	0.017h	0h	0.017h	0h	0h	0h	0h	0h	0h	0h	0.017h
CAP-342	iPhone App	↓	4.5h	0h	4.5h	0h	0h	0h	0h	0h	0h	0h	4.5h
CAP-347	create string table for localization	↓	3.517h	0h	0h	0h	0h	0h	0h	3.517h	0h	0h	3.517h
CAP-348	have finders be overloaded for is deleted and is active to be passed as variables	↑	0.517h	0h	0h	0h	0h	0h	0h	0.517h	0h	0h	0.517h
CAP-377	Review Documents for Deliverable 2 Submission	↑	11.267h	0h	0h	8.267h	1h	0h	0h	2h	0h	0h	11.267h
CAP-388	User Interface Test Cases Written Incorrectly	⇑	1.9h	0h	0h	0h	0h	1.9h	0h	0h	0h	0h	1.9h

Figure 18-1 Person-Hour Work Log

The above table shows the person-hours spent on various tasks in Sprint 7. The overall number of hours worked was lower than the past few iterations. See the iteration 7 retrospective for further explanation.

18.2 Hour Burndown Chart

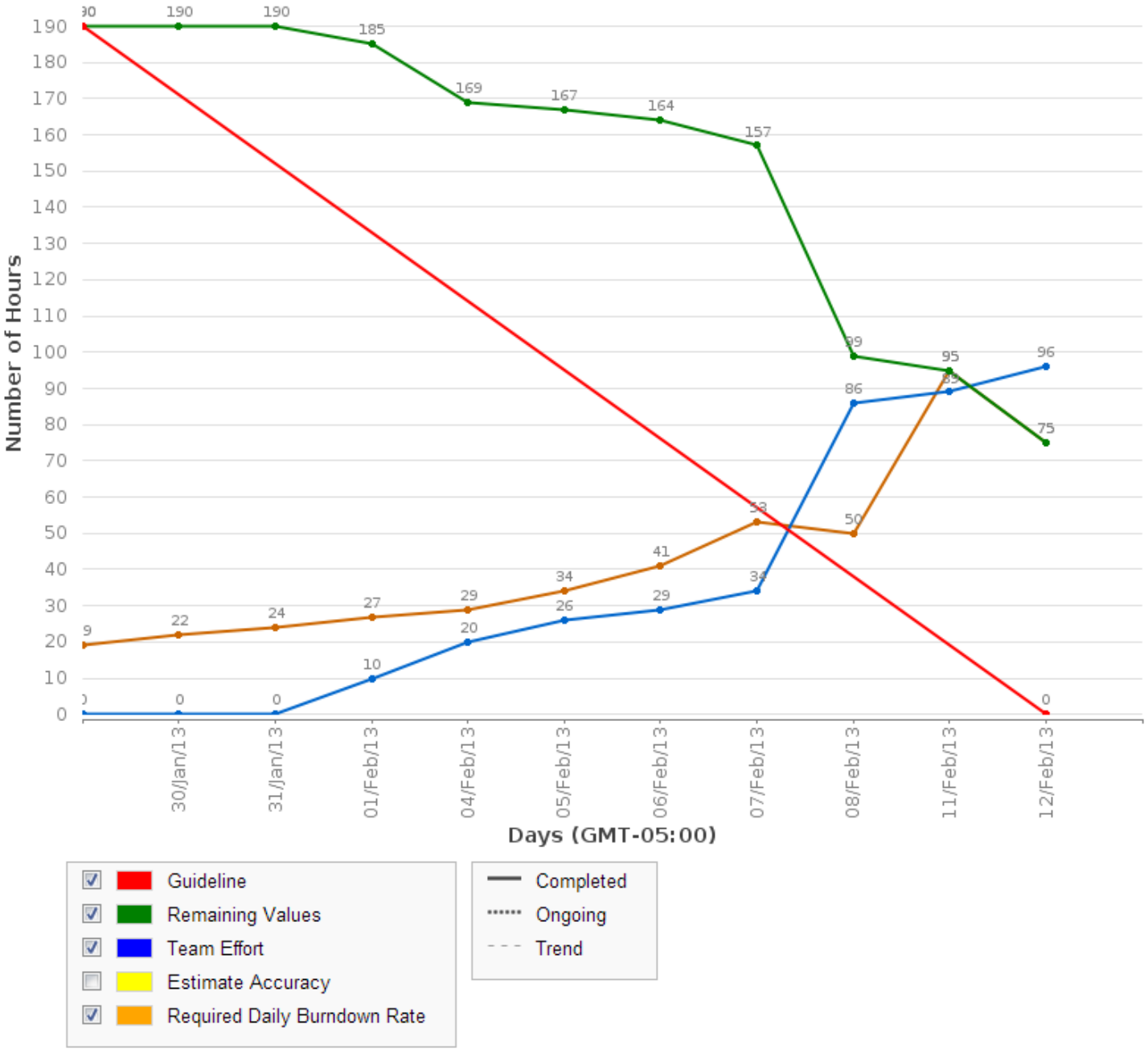

Figure 18-2 Hour Burndown Chart

Due to various reasons, there were 75 estimated hours left that were not burned down. This is not critical as essential parts of the system still got done. The chart above shows that the first week was slow in terms of effort but that more hours were worked at the end of the sprint. This was due to a SVN trunk merge that did not go well and other responsibilities preventing the team from working as much as usual.

18.3 Issue Burndown Chart

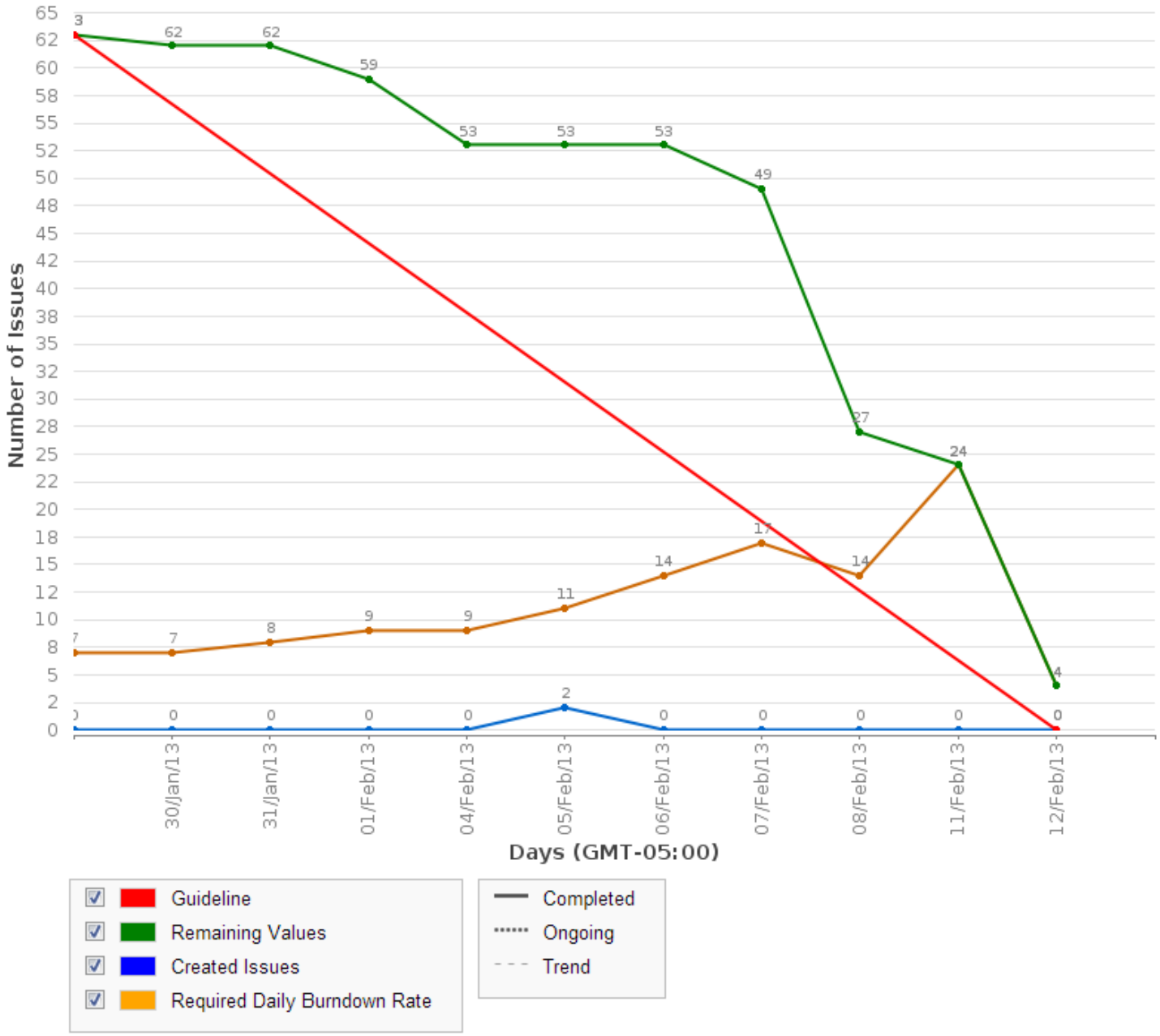

Figure 18-3 Issue Burndown Chart

Most issues were completed. However, two stories were not completed. These two stories are CAP-39 Leave Review and CAP-112 Toppings and Sides. They are scheduled to be pushed and completed in sprint 8.

18.4 Cumulative Flow Diagram

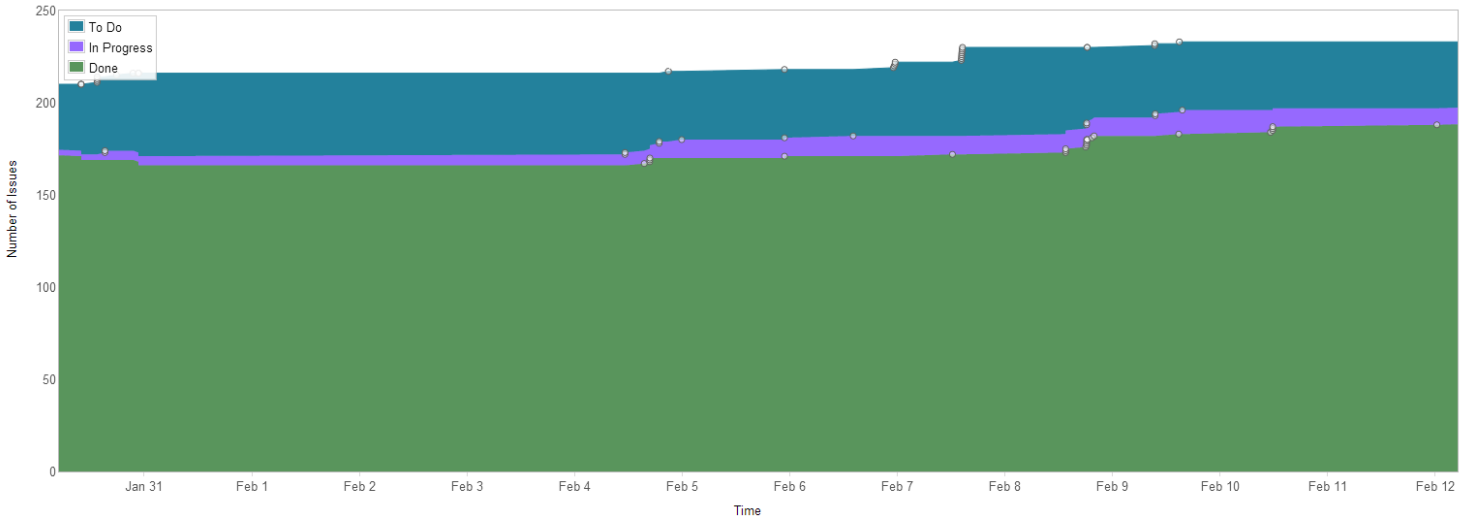

Figure 18-4 Cumulative Flow Diagram

The figure above shows that there has been a lot of progress since the beginning of the project. There is still work to do but many of the larger issues are getting cleared out quickly.

18.5 Measurement Report

18.5.1 Code Quality Analysis

The following report was generated after analyzing the code. Five methods did not meet code quality goals by exceeding the expected number of lines of code. All five were found in previous iterations. They have been entered as bugs and will be fixed in a future iteration. Since it is a minor bug, the priority is not high for it which is why they occur repeatedly in the code quality analysis report.

Since our only issues have been with lines of code, we took note for the future that the following scale defines lines of code quality:

0: red
1 – 10: green
11 – 10: yellow
21+: red

Analysis tool used: Code Metrics Viewer
Found at: http://visualstudiogallery.msdn.microsoft.com/9f35524b-a784-4dbc-bd7b-6babd7a5a3b3
Version: 1.5.3
Last updated: 2/5/2012

Scope	Hiearchy	Maintainability Index	Cyclomatic Complexity	Depth of Inheritance	Class Coupling	Lines of Code
ModuleScope	TouchForFood.dll	80		192	4	
NamespaceScope	TouchForFood	82	4	10	2	9
TypeScope	MvcApplication	82	4	10	2	9
MemberScope	Application_Start() : void	73	1	4		5
MemberScope	MvcApplication()	100	1	1		1
MemberScope	RegisterGlobalFilters(GlobalFilterCollection) : void	94	1	2		1
MemberScope	RegisterRoutes(RouteCollection) : void	82	1	3		2
NamespaceScope	TouchForFood.Attributes	84	2	5	3	4
TypeScope	AjaxAttribute	84	2	5	3	4
MemberScope	AjaxAttribute(bool)	87	1	1		2
MemberScope	IsValidForRequest(ControllerContext, MethodInfo) : bool	83	1	4		2
NamespaceScope	TouchForFood.Controllers	66	516	127	3	
TypeScope	BillController	68	29	40	3	80
TypeScope	CategoryController	71	23	31	3	57
TypeScope	FriendshipController	69	20	29	3	42
TypeScope	HomeController	74	5	15	3	6
TypeScope	ItemController	70	27	36	3	72
TypeScope	Menu_CategoryController	67	41	31	3	85
TypeScope	Menu_ItemController	69	31	25	3	58
TypeScope	MenuController	67	39	39	3	82
TypeScope	Order_ItemController	66	28	40	3	66
TypeScope	OrderController	59	85	62	3	207
TypeScope	RestaurantController	71	21	25	3	40
TypeScope	ReviewController	59	15	35	3	40
TypeScope	SearchController	55	12	25	3	27
TypeScope	ServiceRequestController	64	73	45	3	111
TypeScope	TableController	66	37	50	3	71
TypeScope	UserController	64	30	43	3	96
MemberScope	Create() : ActionResult	74	2	9		3
MemberScope	Create(user) : ActionResult	51	5	20		21
MemberScope	Delete(int) : ActionResult	84	1	4		2

MemberScope	DeleteConfirmed(int) : ActionResult	68	3	14		5
MemberScope	Details(int) : ViewResult	84	1	4		2
MemberScope	Dispose(bool) : void	87	1	3		2
MemberScope	Edit(int) : ActionResult	84	1	4		2
MemberScope	Edit(user, HttpPostedFileBase) : ActionResult	54	8	18		14
MemberScope	Index() : ViewResult	74	2	11		3
MemberScope	LogOff() : ActionResult	76	1	5		4
MemberScope	LogOn() : ViewResult	87	1	2		2
MemberScope	LogOn(string, string) : ActionResult	46	3	24		32
MemberScope	UserController()	75	1	4		4
NamespaceScope	TouchForFood.Exceptions	97	4	1	2	4
TypeScope	AssociationExistsException	97	2	1	2	2
TypeScope	ItemActiveException	97	2	1	2	2
MemberScope	ItemActiveException()	100	1	1		1
MemberScope	ItemActiveException(string)	98	1	1		1
NamespaceScope	TouchForFood.Mappers	75	174	58	2	459
TypeScope	CategoryIM	84	6	6	2	10
TypeScope	CategoryOM	70	7	13	2	22
TypeScope	ItemIM	84	6	6	2	10
TypeScope	ItemOM	77	4	8	2	10
TypeScope	MenuCategoryIM	73	6	14	2	18
TypeScope	MenuCategoryOM	65	8	15	2	30
TypeScope	MenuIM	66	8	17	2	35
TypeScope	MenuItemIM	73	6	14	2	18
TypeScope	MenuItemOM	70	8	13	2	22
TypeScope	MenuOM	65	8	15	2	30
TypeScope	OrderIM	84	6	6	2	10
TypeScope	OrderOM	65	11	14	2	31
TypeScope	RestaurantIM	77	6	14	2	15
TypeScope	RestaurantOM	71	18	24	2	40
TypeScope	ReviewIM	84	6	6	2	10
TypeScope	ReviewOM	71	7	14	2	19
TypeScope	ServiceRequestIM	73	8	15	2	23
TypeScope	ServiceRequestOM	73	5	9	2	18
TypeScope	TableIM	84	6	6	2	10
TypeScope	TableOM	75	9	12	2	23
TypeScope	UserIM	84	6	6	2	10
TypeScope	UserOM	72	8	15	2	23
TypeScope	WaiterIM	84	6	6	2	10
TypeScope	WaiterOM	76	5	10	2	12
MemberScope	clearOrder(ICollection<order>) : void	75	2	8		3
MemberScope	delete(int) : int	67	1	5		7
MemberScope	WaiterOM()	100	1	1		1
MemberScope	WaiterOM(touch_for_foodEntities)	98	1	2		1
NamespaceScope	TouchForFood.Mappers.Abstract	90	5	1	1	8
TypeScope	GenericIM	86	2	1	1	4
TypeScope	GenericOM	94	3	1	1	4
MemberScope	delete(int) : int	100	1	0		0
MemberScope	GenericOM()	86	1	1		2
MemberScope	GenericOM(touch_for_foodEntities)	87	1	1		2
NamespaceScope	TouchForFood.Models	92	453	46	2	544
TypeScope	bill	92	21	7	1	22
TypeScope	category	92	11	4	1	13
TypeScope	CategoryFilterVM	83	9	6	1	16
TypeScope	friendship	93	11	2	1	11
TypeScope	item	92	19	5	1	20
TypeScope	ItemFilterVM	80	18	16	1	34
TypeScope	menu	92	17	5	1	18
TypeScope	menu_category	92	19	5	1	20
TypeScope	menu_item	92	21	6	1	22
TypeScope	MenuMetadata	94	7	4	1	7
TypeScope	order	89	33	13	1	55
TypeScope	order_item	92	26	9	1	29
TypeScope	OrderItemMetadata	100	1	0	1	1

TypeScope	OrderMetadata	100	1	0	1	1
TypeScope	OrderStatusHelper	65	25	7	1	44
TypeScope	OrderStatusHelper.OrderItemStatusEnum	100	0	0	1	0
TypeScope	OrderStatusHelper.OrderStatusEnum	100	0	0	1	0
TypeScope	restaurant	91	27	10	1	32
TypeScope	restaurant_user	93	11	3	1	11
TypeScope	RestaurantMetadata	94	9	1	1	9
TypeScope	review	92	25	7	1	25
TypeScope	ReviewMetadata	93	11	5	1	11
TypeScope	service_request	93	15	3	1	15
TypeScope	table	91	16	9	1	22
TypeScope	TableMetadata	100	1	0	1	1
TypeScope	touch_for_foodEntities	92	34	20	2	34
TypeScope	user	91	35	9	1	40
TypeScope	UserMetadata	93	15	5	1	15
TypeScope	waiter	93	15	5	1	16
MemberScope	first_name.get() : string	98	1	0		1
MemberScope	first_name.set(string) : void	95	1	0		1
MemberScope	id.get() : int	98	1	0		1
MemberScope	id.set(int) : void	95	1	0		1
MemberScope	last_name.get() : string	98	1	0		1
MemberScope	last_name.set(string) : void	95	1	0		1
MemberScope	orders.get() : ICollection<order>	98	1	2		1
MemberScope	orders.set(ICollection<order>) : void	95	1	2		1
MemberScope	restaurant.get() : restaurant	98	1	1		1
MemberScope	restaurant.set(restaurant) : void	95	1	1		1
MemberScope	resto_id.get() : int?	98	1	1		1
MemberScope	resto_id.set(int?) : void	95	1	1		1
MemberScope	version.get() : int	98	1	0		1
MemberScope	version.set(int) : void	95	1	0		1
MemberScope	waiter()	87	1	2		2
NamespaceScope	TouchForFood.Util.Bill	74	13	9	1	15
TypeScope	BillUtil	74	13	9	1	15
MemberScope	GetTotalAfterTax(bill) : decimal	78	3	3		2
MemberScope	GetTotalBeforeTax(bill) : decimal	85	1	3		2
MemberScope	GetTPS() : decimal	84	1	1		2
MemberScope	GetTVQ() : decimal	84	1	1		2
MemberScope	Update(ref bill) : void	62	7	9		7
NamespaceScope	TouchForFood.Util.Category	70	8	12	1	15
TypeScope	CategoryUtil	70	8	12	1	15
MemberScope	CategoryUtil()	100	1	0		1
MemberScope	CategoryUtil()	94	1	1		1
MemberScope	filterListByMenu(menu) : IList<category>	58	6	12		13
NamespaceScope	TouchForFood.Util.Html	69	16	26	1	39
TypeScope	HtmlDropDownExtensions	68	10	17	1	25
TypeScope	ImageActionLinkHelper	64	1	7	1	8
TypeScope	UrlUtils	75	5	4	1	6
MemberScope	ConvertRelativeUrlToAbsoluteUrl(string) : string	67	4	4		5
MemberScope	UrlUtils()	100	1	0		1
NamespaceScope	TouchForFood.Util.Item	63	9	15	1	17
TypeScope	ItemUtil	63	9	15	1	17
MemberScope	filterListByItem(menu_category) : IList<item>	54	8	15		16
MemberScope	ItemUtil()	100	1	0		1
NamespaceScope	TouchForFood.Util.Order	82	22	20	1	49
TypeScope	OrderStatusUtil	95	2	1	1	2
TypeScope	OrderUtil	68	20	20	1	47
MemberScope	filterItem(menu_item) : item	77	1	4		3
MemberScope	filterMenuItem(order_item) : menu_item	74	1	5		4
MemberScope	filterTable(order) : table	77	1	5		3
MemberScope	filterUser(order) : user	77	1	5		3
MemberScope	filterWaiter(order) : waiter	77	1	5		3
MemberScope	mergeExistingOrderToDb(order) : void	48	9	13		23
MemberScope	OrderUtil()	100	1	0		1
MemberScope	OrderUtil()	94	1	1		1

MemberScope	UpdatePrice(ref order) : void	65	4	11		6
NamespaceScope	TouchForFood.Util.Search	63	31	34	1	114
TypeScope	SearchService	51	3	18	1	45
TypeScope	SearchUtil	60	23	19	1	63
TypeScope	SearchViewModelHelper	78	5	9	1	6
MemberScope	PopulateSearchViewModel(menu_item) : SearchViewModel	76	1	6		2
MemberScope	PopulateSearchViewModelList(IList<menu_item>) : IList<Search	76	3	5		3
MemberScope	SearchViewModelHelper()	100	1	0		1
NamespaceScope	TouchForFood.Util.Security	77	21	22	4	78
TypeScope	AES	67	14	11	1	62
TypeScope	CustomAuthorizationAttribute	64	7	10	4	16
TypeScope	SiteRoles	100	0	1	1	0
NamespaceScope	TouchForFood.Util.ServiceRequest	100	1	0	1	1
TypeScope	ServiceRequestUtil	100	1	0	1	1
TypeScope	ServiceRequestUtil.ServiceRequestStatus	100	0	0	1	0
NamespaceScope	TouchForFood.Util.Session	55	5	15	1	30
TypeScope	SessionUtil	55	5	15	1	30
MemberScope	getOpenOrder(user) : order	46	4	15		29
MemberScope	SessionUtil()	100	1	0		1
NamespaceScope	TouchForFood.Util.User	75	7	23	1	19
TypeScope	UserUtil	75	7	23	1	19
MemberScope	getAuthenticatedUser(HttpContextBase) : user	64	2	14		7
MemberScope	getAuthenticatedUser(HttpRequestBase) : user	64	2	8		8
MemberScope	isUserRole(SiteRoles, HttpContext) : bool	84	1	3		2
MemberScope	UserUtil()	100	1	0		1
MemberScope	UserUtil()	94	1	1		1
NamespaceScope	TouchForFood.ViewModels	81	40	14	1	59
TypeScope	OrderItemVM	76	1	2	1	4
TypeScope	OrderVM	76	23	11	1	32
TypeScope	SearchViewModel	91	16	1	1	23
MemberScope	catName.get() : string	98	1	0		1
MemberScope	catName.set(string) : void	95	1	0		1
MemberScope	description.get() : string	98	1	0		1
MemberScope	description.set(string) : void	95	1	0		1
MemberScope	menuItemId.get() : int	98	1	0		1
MemberScope	menuItemId.set(int) : void	95	1	0		1
MemberScope	menuName.get() : string	98	1	0		1
MemberScope	menuName.set(string) : void	95	1	0		1
MemberScope	metadata.get() : string	98	1	0		1
MemberScope	metadata.set(string) : void	95	1	0		1
MemberScope	name.get() : string	98	1	0		1
MemberScope	name.set(string) : void	95	1	0		1
MemberScope	price.get() : decimal	98	1	1		1
MemberScope	price.set(decimal) : void	95	1	1		1
MemberScope	SearchViewModel()	100	1	0		1
MemberScope	SearchViewModel(int, string, string, string, string, string, decima	68	1	1		8

Figure 18-5 Code Quality Report

18.6 Retrospective

In iteration 7, we completed 11 out of a planned 22 story points. The two stories CAP-39 – Leave Review and CAP-112 – Topping and Sides were added to the next sprint as they were not yet complete.

There was less time spent working this iteration than in the previous two. This can be attributed to a few things. There was a problem with the SVN when we merged a branch with the trunk. It took a few days to solve and code could not be worked on during that time. Also, other classes took time away from the TFF project. Quizzes and assignments happened to fall on the last day of this sprint which made it difficult to get a lot of work done. One team member was also out of the country for most of the iteration.

The team also had two meetings during iteration 7, one with Dr. Grogono and one with Dr. Constantinides. The meeting with Dr. Grogono was to introduce him to the project and get his feedback as he is our new stakeholder since Dr. Ormandjieva left. Everything went well and he seemed satisfied with the progress thus far. There were no changes to the requirements.

All things considered, the team achieved a satisfactory amount of work this iteration. We did not complete everything planned, but we are looking to complete all the rest of the user stories by the end of iteration 8. This is possible because the team has a one week break from school during iteration 8. Thus, we will make up the time lost in iteration 7 and more during iteration 8.

18.6.1 Velocity

Sprint 7 velocity (story points): 11 story points
Cumulative velocity (story points): 56 story points
Average velocity (story points): 8 story points

Sprint 7 velocity (p-h): 170.82 person-hours
Cumulative velocity (p-h):1,228.22 person-hours
Average velocity per sprint:175.46 person-hours

Our average velocity in story points increased this iteration. Our average velocity in person-hours dropped by one hour. Overall, we're expecting our velocity to increase next sprint as we have a one week break from school. This should give us much more time to commit to the project and complete it on time.

18.6.2 Budget

Total person-hours budgeted to date: 3,726 person-hours
Total person-hours worked to date: 1,228.22 person-hours

Person-hours budgeted per sprint: 414 person-hours
Person-hours worked in Sprint 7: 170.82 person-hours

We did not meet our goal of working at least 200 person-hours per sprint. However, we were missing one person because they were out of the country. Other school and work responsibilities also led to us not meeting our budget. In iteration 8, the team is planning to make up for that lost time.

18.6.3 What was learned

The following positive and negative points were gathered from this sprint.

Positives:

- Completed 11 story points
- Roles such as QA, Bug basher, Flow fixer, and code reviewers continued to improve the quality of the project
- Still on schedule to finish the project in time
- Communication and meetings are going very well
- Met with our stakeholder and project supervisor

Negatives:

- Did not meet our goal in person-hours worked
- Did not finish the planned stories

19 Iteration 8 Plan

The goal of Iteration 8 is to finish all stories left in the project. We are aware that there are a lot of stories to do, but we have a week without classes this sprint so some of the team members will be able to work more than in previous sprints. The reason we want to finish this sprint is to leave the last two sprints for fixing defects and focusing on testing and quality assurance. There is no official “flow fixer” this sprint as most people have higher priority tasks assigned to them.

19.1 Planned Activities

Tables featured in this section of the document were created with JIRA [**2**].

Table 19-1 Planned Activities

Key	Issue Type	Priority	Assignee	Original Estimate (hours)
CAP-390	Bug	Trivial	Ryan Nasr	0.5
CAP-394	Improvement	Trivial	Josh Hum	0.083333333
CAP-400	Improvement	Trivial	Josh Hum	0.083333333
CAP-342	Task	Minor	Cristian Asenjo	24
CAP-345	Sub-task	Minor	Cristian Asenjo	4
CAP-386	Bug	Minor	Unassigned	0
CAP-391	Bug	Minor	Ryan Nasr	0.5
CAP-392	Bug	Minor	Josh Hum	0.5
CAP-393	Bug	Minor	Josh Hum	0.5
CAP-405	Improvement	Minor	Unassigned	4
CAP-414	Bug	Minor	Ryan Nasr	0
CAP-418	Sub-task	Minor	Unassigned	8
CAP-449	Sub-task	Minor	Josh Hum	3
CAP-450	Sub-task	Minor	Josh Hum	2
CAP-451	Sub-task	Minor	Josh Hum	2
CAP-452	Sub-task	Minor	Josh Hum	0.5
CAP-453	Sub-task	Minor	Mikhail Levkovsky	0.5
CAP-36	Story	Major	Cynthia Donato	0
CAP-39	Story	Major	Mikhail Levkovsky	0
CAP-112	Story	Major	Matthew Tam	0
CAP-169	Bug	Major	Cynthia Donato	5
CAP-198	Task	Major	Katrina Anderson	0
CAP-199	Sub-task	Major	Katrina Anderson	3
CAP-200	Sub-task	Major	Katrina Anderson	5
CAP-201	Sub-task	Major	Katrina Anderson	7
CAP-204	Sub-task	Major	Katrina Anderson	5
CAP-205	Sub-task	Major	Katrina Anderson	5
CAP-206	Sub-task	Major	Katrina Anderson	9
CAP-207	Sub-task	Major	Katrina Anderson	4

CAP-258	Sub-task	Major	Katrina Anderson	5
CAP-267	Sub-task	Major	Patrick Modafferi	3
CAP-279	Story	Major	Patrick Modafferi	0
CAP-343	Sub-task	Major	Cristian Asenjo	5
CAP-344	Sub-task	Major	Cristian Asenjo	10
CAP-352	Sub-task	Major	Mikhail Levkovsky	0.5
CAP-354	Sub-task	Major	Patrick Modafferi	16
CAP-355	Sub-task	Major	Patrick Modafferi	1
CAP-356	Sub-task	Major	Mikhail Levkovsky	6
CAP-357	Sub-task	Major	Patrick Modafferi	6
CAP-362	Sub-task	Major	Cynthia Donato	0.5
CAP-387	Bug	Major	Ryan Nasr	3
CAP-402	Bug	Major	Ryan Nasr	1
CAP-403	Bug	Major	Ryan Nasr	2
CAP-404	Bug	Major	Ryan Nasr	1
CAP-406	Sub-task	Major	Katrina Anderson	0.75
CAP-407	Sub-task	Major	Josh Hum	6
CAP-408	Task	Major	Unassigned	0
CAP-409	Sub-task	Major	Christian Daher	2
CAP-410	Sub-task	Major	Christian Daher	2
CAP-411	Sub-task	Major	Christian Daher	2
CAP-412	Sub-task	Major	Christian Daher	2
CAP-413	Bug	Major	Ryan Nasr	1
CAP-415	Bug	Major	Ryan Nasr	2
CAP-416	Sub-task	Major	Patrick Modafferi	16
CAP-417	Sub-task	Major	Mikhail Levkovsky	8
CAP-419	Sub-task	Major	Cynthia Donato	2
CAP-420	Sub-task	Major	Cynthia Donato	2
CAP-421	Sub-task	Major	Cynthia Donato	2
CAP-422	Sub-task	Major	Cynthia Donato	2
CAP-424	Sub-task	Major	Unassigned	0.5
CAP-425	Sub-task	Major	Unassigned	2
CAP-428	Sub-task	Major	Cynthia Donato	0.5
CAP-429	Sub-task	Major	Cynthia Donato	0.5
CAP-430	Sub-task	Major	Cynthia Donato	0.5
CAP-432	Sub-task	Major	Unassigned	12
CAP-433	Sub-task	Major	Matthew Tam	32
CAP-434	Sub-task	Major	Unassigned	4
CAP-435	Sub-task	Major	Unassigned	3
CAP-436	Sub-task	Major	Unassigned	3
CAP-437	Sub-task	Major	Unassigned	3

CAP-438	Sub-task	Major	Cynthia Donato	0.5
CAP-439	Sub-task	Major	Cynthia Donato	0.5
CAP-440	Sub-task	Major	Josh Hum	4
CAP-441	Sub-task	Major	Katrina Anderson	3
CAP-442	Sub-task	Major	Cynthia Donato	1
CAP-443	Sub-task	Major	Cynthia Donato	0.5
CAP-444	Sub-task	Major	Josh Hum	3
CAP-445	Sub-task	Major	Josh Hum	3
CAP-446	Sub-task	Major	Katrina Anderson	3
CAP-447	Sub-task	Major	Josh Hum	10
CAP-448	Sub-task	Major	Josh Hum	8
CAP-454	Sub-task	Major	Katrina Anderson	1
CAP-455	Sub-task	Major	Unassigned	0.5
CAP-456	Sub-task	Major	Katrina Anderson	3
CAP-457	Sub-task	Major	Patrick Modafferi	3
			TOTAL	**307.92**

19.2 Person-Hour Estimation

The Expert Judgement Method was used to calculate person-hour estimations.

Table 19-2 Person-Hour Estimation

Description	Worst Case	Most Likely Case	Best Case	Expected Case
Activities	400.30	307.92	215.54	307.92
Total(ph)	400.30	307.92	215.54	307.92
Velocity(ph/day)	28.59	21.99	15.40	21.99
Velocity (ph/team member/day)	3.18	2.44	1.71	2.44

19.3 Activity-on-Node Planning

The following graph reflects the progress of the planned user stories. The numbers represent the Early Start, Duration, Early Finish, Late Start, Slack, and Late Finish in days. All remaining stories are currently in progress. The goal is to complete all of them this sprint so that the final two sprints can be spent on fixing defects and ensuring a high quality product.

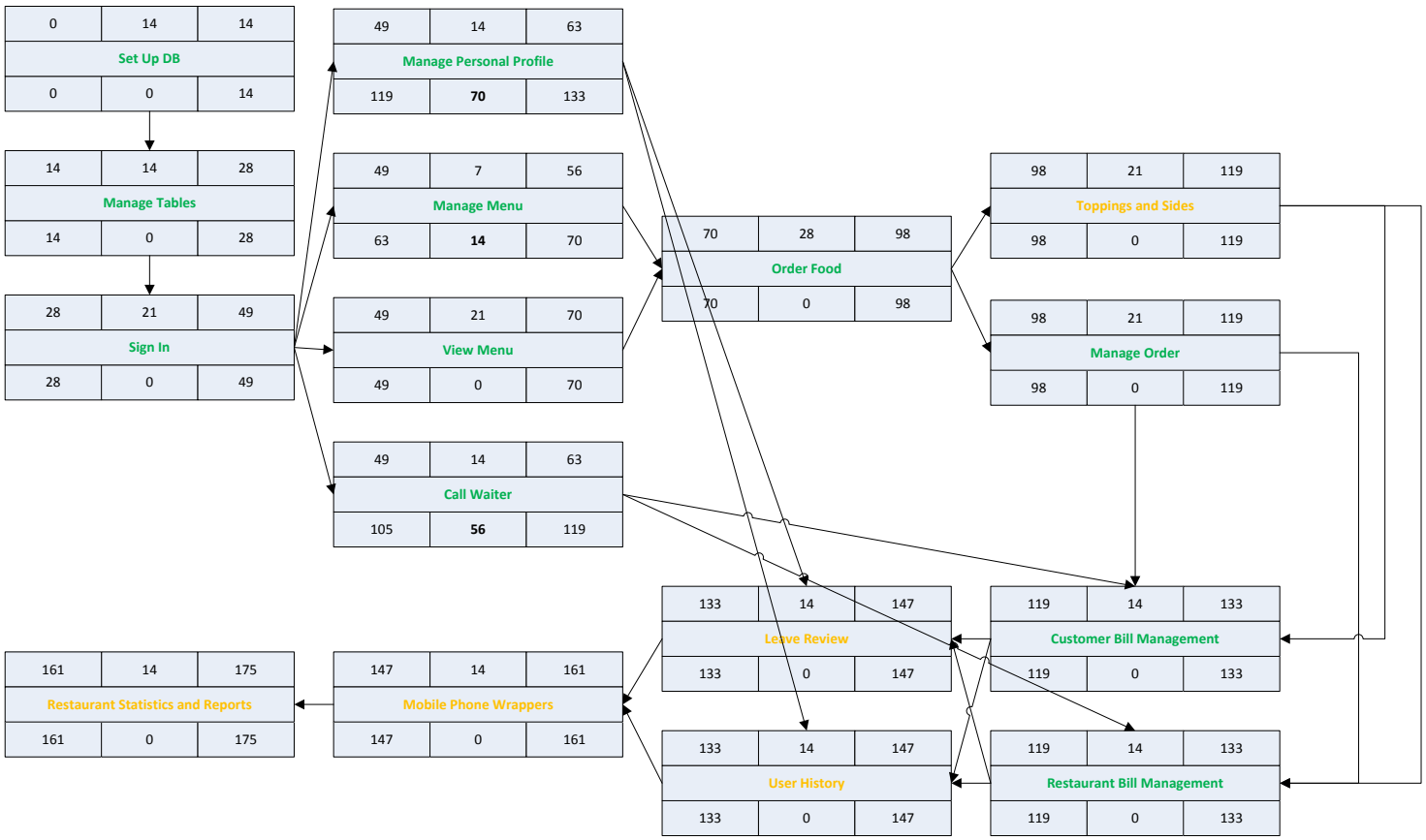

Figure 19-1 Activity-On-Node Diagram

20 Iteration 8 Report

20.1 Person-Hour Work Log

Start Date: 13/Feb/13 End Date: 26/Feb/13 [Change] (UNREGISTERED)			Christian Daher	Cristian Asenjo	Cynthia Donato	Josh Hum	Katrina Anderson	Matthew Tam	Mikhail Levkovsky	Patrick Modafferi	Ryan Nasr	Total
Issue	Total	8.017h	16.317h	32.867h	39.95h	35.8h	29.717h	24.617h	35.983h	20.1h	243.367h
CAP-11	Documentation - SAD	3.667h	0h	0h	0h	3.167h	0h	0h	0.5h	0h	0h	3.667h
CAP-14	Correspondence	12.75h	0h	2.133h	1.167h	1.35h	4.433h	0h	0h	3.667h	0h	12.75h
CAP-18	Meetings	24.333h	3h	4.417h	1.617h	4.033h	3.617h	1.617h	2.367h	3.667h	0h	24.333h
CAP-23	Setup	18.583h	0h	0h	0.333h	3.75h	6.167h	0h	0h	4.333h	4h	18.583h
CAP-26	View Menu	5.5h	0h	0h	0h	5.5h	0h	0h	0h	0h	0h	5.5h
CAP-36	Restaurant Statistics and Reports	20.25h	0h	0h	20.25h	0h	0h	0h	0h	0h	0h	20.25h
CAP-39	Leave Review	29.5h	0h	0h	0h	0h	0h	0h	20h	9.5h	0h	29.5h
CAP-44	Documentation - Test Plan	0.5h	0h	0h	0h	0h	0.5h	0h	0h	0h	0h	0.5h
CAP-57	Documentation - Management	5.033h	0h	0h	0h	4.783h	0h	0h	0.25h	0h	0h	5.033h
CAP-112	Toppings and Sides	28.1h	0h	0h	0h	0h	0h	28.1h	0h	0h	0h	28.1h
CAP-164	Documentation - Test Report	8.783h	0h	0h	0h	4.783h	0h	0h	0h	4h	0h	8.783h
CAP-169	Optimistic offline locks logic missing from orders	1h	0h	0h	1h	0h	0h	0h	0h	0h	0h	1h
CAP-198	Unit Testing Retroactive	21.083h	0h	0h	0h	0h	21.083h	0h	0h	0h	0h	21.083h
CAP-255	is_active and is_deleted field add extra complexity, factor that out	8.5h	0h	0h	8.5h	0h	0h	0h	0h	0h	0h	8.5h
CAP-264	User Interface Testing	5.083h	0h	0h	0h	5.083h	0h	0h	0h	0h	0h	5.083h
CAP-279	User History	10.533h	0h	0h	0h	0h	0h	0h	0h	10.533h	0h	10.533h
CAP-299	Flow and Improvements	6.083h	0h	0h	0h	6.083h	0h	0h	0h	0h	0h	6.083h
CAP-314	Code Review	2.917h	0h	0h	0h	0.917h	0h	0h	0h	0h	2h	2.917h
CAP-342	iPhone App	9.767h	0h	9.767h	0h	0h	0h	0h	0h	0h	0h	9.767h
CAP-387	Restaurant User Not Associated To Restos They Create	1.517h	0h	0h	0h	0h	0h	0h	0h	0h	1.517h	1.517h
CAP-391	Manage Bill Option Visibility	0.517h	0h	0h	0h	0h	0h	0h	0h	0h	0.517h	0.517h
CAP-392	Menu Item Create Form Missing Cancel Option	0.167h	0h	0h	0h	0.167h	0h	0h	0h	0h	0h	0.167h
CAP-393	Menu Category Create Form Missing Cancel Option	0.167h	0h	0h	0h	0.167h	0h	0h	0h	0h	0h	0.167h
CAP-394	UIR - Change figure 2.3	0.083h	0h	0h	0h	0.083h	0h	0h	0h	0h	0h	0.083h
CAP-400	UIR 2.4.2 update image	0.083h	0h	0h	0h	0.083h	0h	0h	0h	0h	0h	0.083h
CAP-403	OM should contain creation logic	2.017h	0h	0h	0h	0h	0h	0h	0h	0h	2.017h	2.017h
CAP-404	Rejected items should not appear in the order or int he bill management	1.017h	0h	0h	0h	0h	0h	0h	0h	0h	1.017h	1.017h
CAP-408	Android Wrapper	10.033h	5.017h	0h	0h	0h	0h	0h	0h	0.017h	5h	10.033h
CAP-413	Bill TVQ TPS and TOTAL rounding	1h	0h	0h	0h	0h	0h	0h	0h	0h	1h	1h
CAP-414	UserUtil methods for getAuthenticatedUser should be updated to use the mappers	1.017h	0h	0h	0h	0h	0h	0h	0h	0h	1.017h	1.017h
CAP-415	blank images should not be displayed	2.017h	0h	0h	0h	0h	0h	0h	0h	0h	2.017h	2.017h
CAP-464	Trunk is broken due to db change	0.267h	0h	0h	0h	0h	0h	0h	0h	0.267h	0h	0.267h
CAP-510	Review Rating should always be a number	0.5h	0h	0h	0h	0h	0h	0h	0.5h	0h	0h	0.5h
CAP-511	Fix flicker on Star Ratings	1h	0h	0h	0h	0h	0h	0h	1h	0h	0h	1h

Figure 20-1 Person-Hour Work Log

We are satisfied by the effort that we put into Iteration 8. The hours worked went up to 250 hours from 170 hours in Iteration 7. Team members will able to work more hours on the project since we had one week off school. Obviously, other courses had to be worked on as well so it wasn't a full week dedicated to the project, but it was more time than usual.

20.2 Hour Burndown Chart

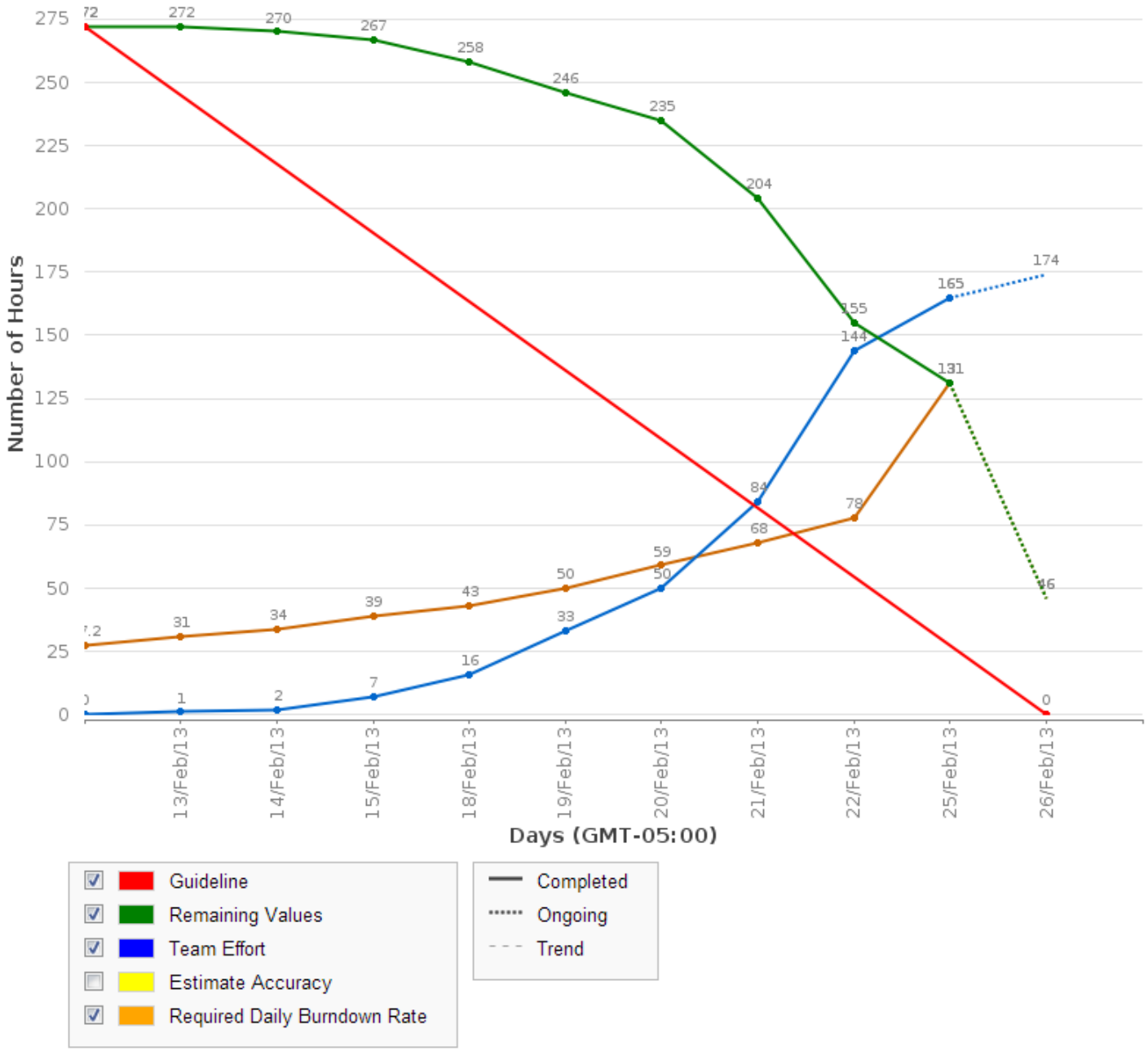

Figure 20-2 Hour Burndown Chart

There are 131 estimated hours that are not burned down. This can be attributed to the fact that although most of the coding functionality tasks were completed, some documentation and unit testing tasks were not and will have to be pushed till next sprint. The hours for those incomplete tasks are thus reflected here. 243 hours were worked this sprint in total and 272 hours were estimated. With the 131 hours remaining, this proves that 102 hours were spent working on overhead, other non-scheduled tasks, or our tasks took more time than estimated.

20.3 Issue Burndown Chart

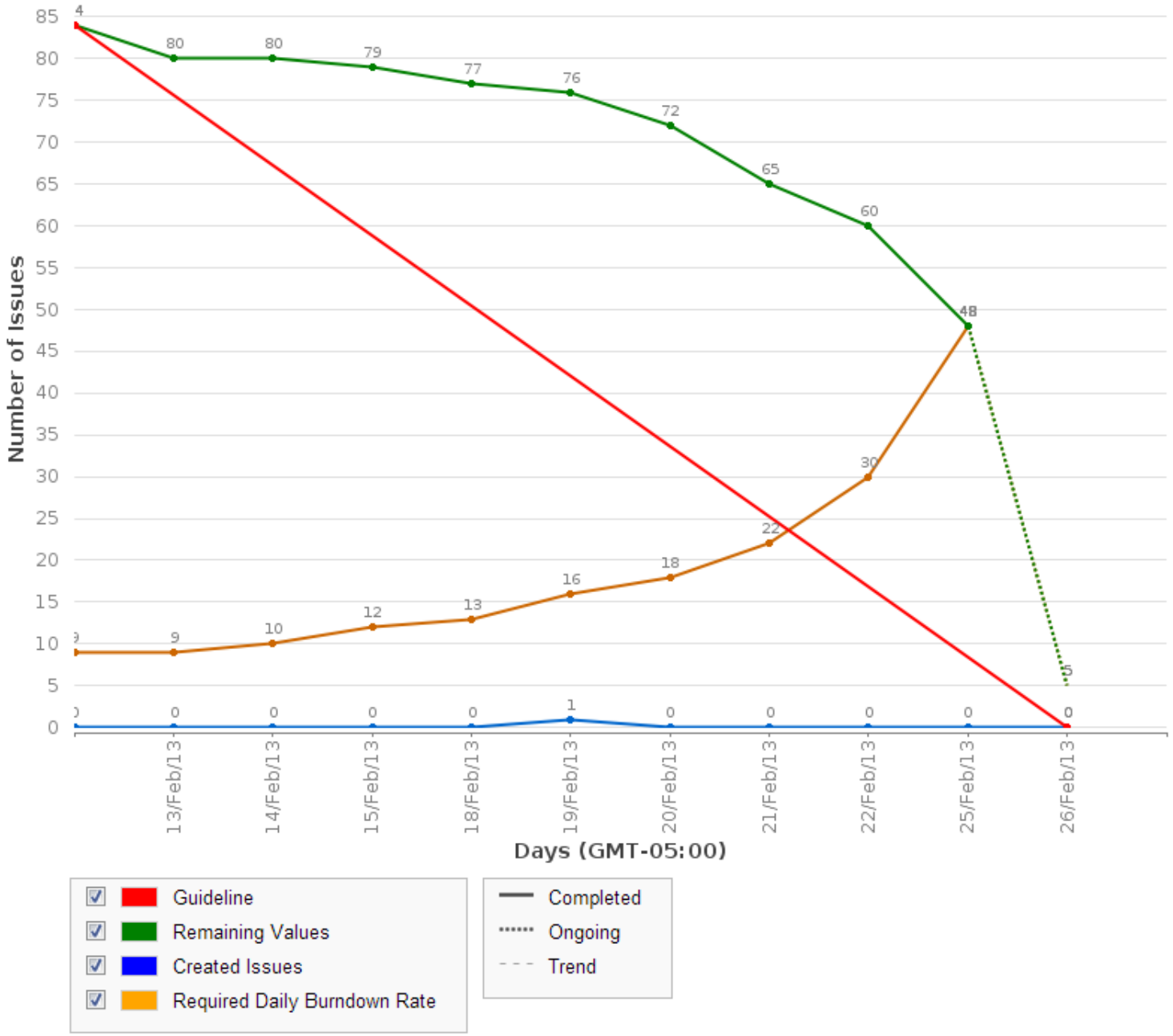

Figure 20-3 Issue Burndown Chart

48 out of the 84 tasks were not completed. However, as mentioned previously, the main functionality of the planned stories is done. The 48 incomplete tasks reflect documentation or unit testing which will be completed next sprint. Although the goal was to complete all the planned issues, we are still satisfied with the amount and quality of work completed.

20.4 Cumulative Flow Diagram

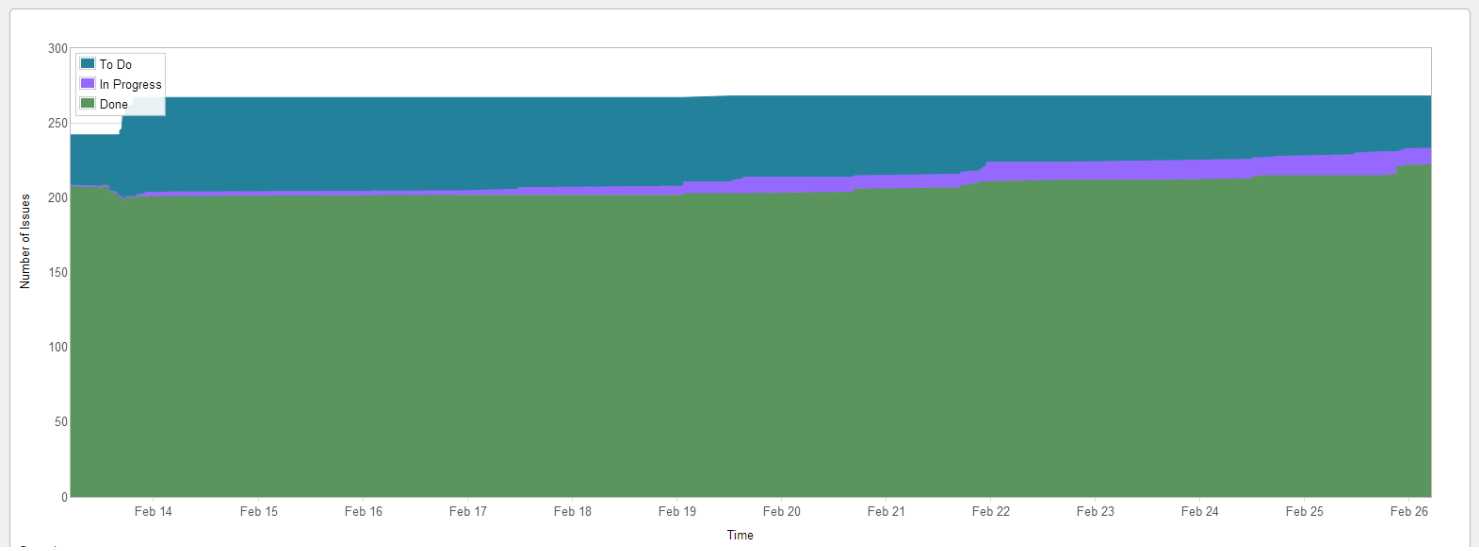

Figure 20-4 Cumulative Flow Diagram

The figure above reflects the steady progress that our team is making. With only two sprints left, we plan to close out all remaining planned tasks in order of priority. If a couple low priority tasks are left incomplete, it is not that critical.

20.5 Measurement Report

20.5.1 Code Quality Analysis

The following report was generated after analyzing the code. Five methods did not meet code quality goals by exceeding the expected number of lines of code. All five were found in previous iterations. They have been entered as bugs and will be fixed in a future iteration. Since it is a minor bug, the priority is not high for it which is why they occur repeatedly in the code quality analysis report.

Since our only issues have been with lines of code, we took note for the future that the following scale defines lines of code quality:

0: red
1 – 10: green
11 – 10: yellow
21+: red

Analysis tool used: Code Metrics Viewer
Found at: http://visualstudiogallery.msdn.microsoft.com/9f35524b-a784-4dbc-bd7b-6babd7a5a3b3
Version: 1.5.3
Last updated: 2/5/2012

Scope	Hiearchy	Maintainability Index	Cyclomatic Complexity	Depth of Inheritance	Class Coupling	Lines of Code
ModuleScope	TouchForFood.dll	80		197	4	
NamespaceScope	TouchForFood	82	4	10	2	9
TypeScope	MvcApplication	82	4	10	2	9
MemberScope	Application_Start() : void	73	1	4		5
MemberScope	MvcApplication()	100	1	1		1
MemberScope	RegisterGlobalFilters(GlobalFilterCollection) : void	94	1	2		1
MemberScope	RegisterRoutes(RouteCollection) : void	82	1	3		2
NamespaceScope	TouchForFood.Attributes	84	2	5	3	4
TypeScope	AjaxAttribute	84	2	5	3	4
MemberScope	AjaxAttribute(bool)	87	1	1		2
MemberScope	IsValidForRequest(ControllerContext, MethodInfo) : bool	83	1	4		2
NamespaceScope	TouchForFood.Controllers	66	529	131	3	
TypeScope	BillController	68	30	41	3	81
TypeScope	CategoryController	71	23	31	3	57
TypeScope	FriendshipController	69	20	29	3	42
TypeScope	HomeController	74	5	15	3	6
TypeScope	ItemController	70	27	36	3	72
TypeScope	Menu_CategoryController	67	41	30	3	83
TypeScope	Menu_ItemController	69	31	25	3	58
TypeScope	MenuController	67	39	39	3	82
TypeScope	Order_ItemController	66	30	41	3	66
TypeScope	OrderController	59	85	59	3	200
TypeScope	RestaurantController	75	19	21	3	38
TypeScope	ReviewController	60	15	34	3	39
TypeScope	SearchController	55	12	25	3	27
TypeScope	ServiceRequestController	64	73	45	3	111
TypeScope	TableController	64	48	54	3	82
TypeScope	UserController	64	31	44	3	98
MemberScope	Create() : ActionResult	74	2	9		3
MemberScope	Create(user) : ActionResult	51	5	20		21
MemberScope	Delete(int) : ActionResult	84	1	4		2

211	MemberScope	DeleteConfirmed(int) : ActionResult	68	3	14		5
212	MemberScope	Details(int) : ViewResult	84	1	4		2
213	MemberScope	Dispose(bool) : void	87	1	3		2
214	MemberScope	Edit(int) : ActionResult	73	2	6		4
215	MemberScope	Edit(user, HttpPostedFileBase) : ActionResult	54	8	18		14
216	MemberScope	Index() : ViewResult	74	2	11		3
217	MemberScope	LogOff() : ActionResult	76	1	5		4
218	MemberScope	LogOn() : ViewResult	87	1	2		2
219	MemberScope	LogOn(string, string) : ActionResult	46	3	24		32
220	MemberScope	UserController()	75	1	4		4
221	NamespaceScope	TouchForFood.Exceptions	97	4	1	2	4
222	TypeScope	AssociationExistsException	97	2	1	2	2
225	TypeScope	ItemActiveException	97	2	1	2	2
226	MemberScope	ItemActiveException()	100	1	1		1
227	MemberScope	ItemActiveException(string)	98	1	1		1
228	NamespaceScope	TouchForFood.Mappers	75	205	63	2	540
229	TypeScope	BillIM	83	8	7	2	14
235	TypeScope	BillOM	72	8	15	2	23
241	TypeScope	CategoryIM	84	6	6	2	10
246	TypeScope	CategoryOM	70	7	13	2	22
251	TypeScope	ItemIM	84	6	6	2	10
256	TypeScope	ItemOM	77	4	8	2	10
260	TypeScope	MenuCategoryIM	73	6	14	2	18
265	TypeScope	MenuCategoryOM	65	8	15	2	30
270	TypeScope	MenuIM	66	8	17	2	35
276	TypeScope	MenuItemIM	73	6	14	2	18
281	TypeScope	MenuItemOM	70	8	13	2	22
286	TypeScope	MenuOM	65	8	15	2	30
291	TypeScope	Order_ItemIM	83	8	7	2	14
297	TypeScope	Order_ItemOM	70	5	10	2	20
302	TypeScope	OrderIM	84	6	6	2	10
307	TypeScope	OrderOM	63	13	17	2	41
	TypeScope	RestaurantIM	77	6	14	2	15
	TypeScope	RestaurantOM	71	18	24	2	40
	TypeScope	ReviewIM	84	6	6	2	10
	TypeScope	ReviewOM	71	7	14	2	19
	TypeScope	ServiceRequestIM	73	8	15	2	23
	TypeScope	ServiceRequestOM	73	5	9	2	18
	TypeScope	TableIM	84	6	6	2	10
	TypeScope	TableOM	75	9	12	2	23
	TypeScope	UserIM	84	6	6	2	10
	TypeScope	UserOM	72	8	15	2	23
	TypeScope	WaiterIM	84	6	6	2	10
	TypeScope	WaiterOM	76	5	10	2	12
	MemberScope	clearOrder(ICollection<order>) : void	75	2	8		3
	MemberScope	delete(int) : int	67	1	5		7
	MemberScope	WaiterOM()	100	1	1		1
	MemberScope	WaiterOM(touch_for_foodEntities)	98	1	2		1
	NamespaceScope	TouchForFood.Mappers.Abstract	90	5	1	1	8
	TypeScope	GenericIM	86	2	1	1	4
	TypeScope	GenericOM	94	3	1	1	4
	MemberScope	delete(int) : int	100	1	0		0
	MemberScope	GenericOM()	86	1	1		2
	MemberScope	GenericOM(touch_for_foodEntities)	87	1	1		2
	NamespaceScope	TouchForFood.Models	92	497	48	2	592
	TypeScope	bill	92	23	7	1	24
	TypeScope	category	92	11	4	1	13
	TypeScope	CategoryFilterVM	83	9	6	1	16
	TypeScope	friendship	93	11	2	1	11
	TypeScope	item	92	19	5	1	20
	TypeScope	ItemFilterVM	80	18	16	1	34
	TypeScope	menu	92	17	5	1	18
	TypeScope	menu_category	92	21	6	1	23
	TypeScope	menu_item	92	21	6	1	22

TypeScope	MenuMetadata	94	7	4	1	7
TypeScope	order	89	35	14	1	58
TypeScope	order_item	92	30	10	1	33
TypeScope	OrderItemMetadata	100	1	0	1	1
TypeScope	OrderMetadata	100	1	0	1	1
TypeScope	OrderStatusHelper	65	25	7	1	44
TypeScope	OrderStatusHelper.OrderItemStatusEnum	100	0	0	1	0
TypeScope	OrderStatusHelper.OrderStatusEnum	100	0	0	1	0
TypeScope	restaurant	91	27	10	1	32
TypeScope	restaurant_user	93	11	3	1	11
TypeScope	RestaurantMetadata	94	9	1	1	9
TypeScope	review	92	23	9	1	24
TypeScope	review_order_item	93	17	4	1	17
TypeScope	ReviewMetadata	94	7	3	1	7
TypeScope	service_request	93	15	3	1	15
TypeScope	side	92	19	6	1	20
TypeScope	table	91	16	9	1	22
TypeScope	TableMetadata	100	1	0	1	1
TypeScope	touch_for_foodEntities	92	38	22	2	38
TypeScope	user	91	35	9	1	40
TypeScope	UserMetadata	93	15	5	1	15
TypeScope	waiter	93	15	5	1	16
MemberScope	first_name.get() : string	98	1	0		1
MemberScope	first_name.set(string) : void	95	1	0		1
MemberScope	id.get() : int	98	1	0		1
MemberScope	id.set(int) : void	95	1	0		1
MemberScope	last_name.get() : string	98	1	0		1
MemberScope	last_name.set(string) : void	95	1	0		1
MemberScope	orders.get() : ICollection<order>	98	1	2		1
MemberScope	orders.set(ICollection<order>) : void	95	1	2		1
MemberScope	restaurant.get() : restaurant	98	1	1		1
MemberScope	restaurant.set(restaurant) : void	95	1	1		1
MemberScope	resto_id.get() : int?	98	1	1		1
MemberScope	resto_id.set(int?) : void	95	1	1		1
MemberScope	version.get() : int	98	1	0		1
MemberScope	version.set(int) : void	95	1	0		1
MemberScope	waiter()	87	1	2		2
NamespaceScope	TouchForFood.Util.Bill	74	13	9	1	15
TypeScope	BillUtil	74	13	9	1	15
MemberScope	GetTotalAfterTax(bill) : decimal	78	3	3		2
MemberScope	GetTotalBeforeTax(bill) : decimal	85	1	3		2
MemberScope	GetTPS() : decimal	84	1	1		2
MemberScope	GetTVQ() : decimal	84	1	1		2
MemberScope	Update(ref bill) : void	62	7	9		7
NamespaceScope	TouchForFood.Util.Category	70	8	12	1	15
TypeScope	CategoryUtil	70	8	12	1	15
MemberScope	CategoryUtil()	100	1	0		1
MemberScope	CategoryUtil()	94	1	1		1
MemberScope	filterListByMenu(menu) : IList<category>	58	6	12		13
NamespaceScope	TouchForFood.Util.Html	69	16	26	1	39
TypeScope	HtmlDropDownExtensions	68	10	17	1	25
TypeScope	ImageActionLinkHelper	64	1	7	1	8
TypeScope	UrlUtils	75	5	4	1	6
MemberScope	ConvertRelativeUrlToAbsoluteUrl(string) : string	67	4	4		5
MemberScope	UrlUtils()	100	1	0		1
NamespaceScope	TouchForFood.Util.Item	63	9	15	1	17
TypeScope	ItemUtil	63	9	15	1	17
MemberScope	filterListByItem(menu_category) : IList<item>	54	8	15		16
MemberScope	ItemUtil()	100	1	0		1
NamespaceScope	TouchForFood.Util.Order	82	22	20	1	49
TypeScope	OrderStatusUtil	95	2	1	1	2
TypeScope	OrderUtil	68	20	20	1	47
MemberScope	filterItem(menu_item) : item	77	1	4		3
MemberScope	filterMenuItem(order_item) : menu_item	74	1	5		4

MemberScope	filterTable(order) : table	77	1	5		3
MemberScope	filterUser(order) : user	77	1	5		3
MemberScope	filterWaiter(order) : waiter	77	1	5		3
MemberScope	mergeExistingOrderToDb(order) : void	48	9	13		23
MemberScope	OrderUtil()	100	1	0		1
MemberScope	OrderUtil()	94	1	1		1
MemberScope	UpdatePrice(ref order) : void	65	4	11		6
NamespaceScope	TouchForFood.Util.Search	63	31	34	1	114
TypeScope	SearchService	51	3	18	1	45
TypeScope	SearchUtil	60	23	19	1	63
TypeScope	SearchViewModelHelper	78	5	9	1	6
MemberScope	PopulateSearchViewModel(menu_item) : SearchViewModel	76	1	6		2
MemberScope	PopulateSearchViewModelList(IList<menu_item>) : IList<Sea	76	3	5		3
MemberScope	SearchViewModelHelper()	100	1	0		1
NamespaceScope	TouchForFood.Util.Security	77	21	22	4	78
TypeScope	AES	67	14	11	1	62
TypeScope	CustomAuthorizationAttribute	64	7	10	4	16
TypeScope	SiteRoles	100	0	1	1	0
NamespaceScope	TouchForFood.Util.ServiceRequest	100	1	0	1	1
TypeScope	ServiceRequestUtil	100	1	0	1	1
TypeScope	ServiceRequestUtil.ServiceRequestStatus	100	0	0	1	0
NamespaceScope	TouchForFood.Util.Session	55	5	15	1	30
TypeScope	SessionUtil	55	5	15	1	30
MemberScope	getOpenOrder(user) : order	46	4	15		29
MemberScope	SessionUtil()	100	1	0		1
NamespaceScope	TouchForFood.Util.User	75	7	23	1	19
TypeScope	UserUtil	75	7	23	1	19
MemberScope	getAuthenticatedUser(HttpContextBase) : user	64	2	14		7
MemberScope	getAuthenticatedUser(HttpRequestBase) : user	64	2	8		8
MemberScope	isUserRole(SiteRoles, HttpContext) : bool	84	1	3		2
MemberScope	UserUtil()	100	1	0		1
MemberScope	UserUtil()	94	1	1		1
NamespaceScope	TouchForFood.ViewModels	81	40	14	1	59
TypeScope	OrderItemVM	76	1	2	1	4
TypeScope	OrderVM	76	23	11	1	32
TypeScope	SearchViewModel	91	16	1	1	23
MemberScope	catName.get() : string	98	1	0		1
MemberScope	catName.set(string) : void	95	1	0		1
MemberScope	description.get() : string	98	1	0		1
MemberScope	description.set(string) : void	95	1	0		1
MemberScope	menuItemId.get() : int	98	1	0		1
MemberScope	menuItemId.set(int) : void	95	1	0		1
MemberScope	menuName.get() : string	98	1	0		1
MemberScope	menuName.set(string) : void	95	1	0		1
MemberScope	metadata.get() : string	98	1	0		1
MemberScope	metadata.set(string) : void	95	1	0		1
MemberScope	name.get() : string	98	1	0		1
MemberScope	name.set(string) : void	95	1	0		1
MemberScope	price.get() : decimal	98	1	1		1
MemberScope	price.set(decimal) : void	95	1	1		1
MemberScope	SearchViewModel()	100	1	0		1
MemberScope	SearchViewModel(int, string, string, string, string, string, deci	68	1	1		8

Figure 20-5 Code Quality Report

20.6 Retrospective

In Iteration 8, we completed 3 out of a planned 40 user story points. This seems alarming at first glance. However, the story with the most points assigned to it was CAP-36 – Restaurant Statistics and Reports. We did not implement as much functionality as was originally planned, thus it was not really worth 21 story points. The functionality for CAP-36 was implemented this sprint; unit testing and docs for the story must still be completed. CAP-112 – Toppings and Sides and CAP-279 – User History were almost completed as well; unit testing must be finished for those stories as well.

The goal of this sprint was to finish implementing all desired functionality for the project to give us time to work on bugs, documentation, and quality during the last two sprints. Although the stories planned were not completed, the implementation part of the stores was completed. Since only unit testing and documentation is left for those stories, we are considering Iteration 8 a successful iteration.

One of our team members ran into computer issues this sprint where her laptop crashed which forced her to re-setup the entire environment and project. This slowed the progress on some of the tasks. There were a couple merge issues as well. However, the team worked hard and accomplished the main goals of this sprint. The main implementation of the project is done and we will now look forward to fixing defects, testing, and documentation.

20.6.1 Velocity

Sprint 8 velocity (story points): 3 story points
Cumulative velocity (story points): 65 story points
Average velocity (story points): 8.1 story points

Sprint 8 velocity (p-h): 243.37 person-hours
Cumulative velocity (p-h): 1,471.59 person-hours
Average velocity per sprint: 183.95 person-hours

Our person-hour velocity went up this iteration as we worked almost 250 hours. This is a considerable increase (up 80 hours from last sprint). We are hoping to stay around 200 person-hours per sprint in the last two iterations. The average story point velocity also increased but that is because of a calculation where our cumulative story points were wrong. It is now correct.

20.6.2 Budget

Total person-hours budgeted to date: 4,140 person-hours
Total person-hours worked to date: 1,471.59 person-hours

Person-hours budgeted per sprint: 414 person-hours
Person-hours worked in Sprint 8: 243.37 person-hours

In Iteration 8, we made up for the less time spent working in Iteration 7 (170 person-hours). We are still significantly under budget but we are meeting our adjusted goal to be around 200 hours per sprint.

20.6.3 What was learned

The following positive and negative points were gathered from this sprint.

Positives:

- On schedule to finish the project in time
- Communication and meetings are going very well
- Everyone is doing their part and working as a team
- Met our goal in person-hours worked

Negatives:

- Did not completely finish the planned stories

21 Iteration 9 Plan

The goal of Iteration 9 is twofold. In the first week, we will focus on documentation (Requirements, Analysis, and Architecture) while in the second week we will work on bugs and testing. There are quite a few bugs that need to be addressed before our presentation so we will have at least half the team working on bugs. The other half will complete unfinished stories and do testing.

21.1 Planned Activities

Tables featured in this section of the document were created with JIRA [**2**].

Table 21-1 Planned Activities

Key	Issue Type	Priority	Assignee	Original Estimate (hours)
CAP-390	Bug	Trivial	Ryan Nasr	0.5
CAP-486	Bug	Trivial	Josh Hum	0.333333333
CAP-488	Bug	Trivial	Mikhail Levkovsky	0.166666667
CAP-491	Bug	Trivial	Cristian Asenjo	1.5
CAP-508	Bug	Trivial	Christian Daher	1
CAP-342	Task	Minor	Cristian Asenjo	24
CAP-345	Sub-task	Minor	Cristian Asenjo	4
CAP-405	Improvement	Minor	Matthew Tam	4
CAP-460	Bug	Minor	Ryan Nasr	0.5
CAP-462	Bug	Minor	Ryan Nasr	1
CAP-481	Bug	Minor	Mikhail Levkovsky	3
CAP-487	Bug	Minor	Patrick Modafferi	0.5
CAP-489	Bug	Minor	Mikhail Levkovsky	0.25
CAP-490	Bug	Minor	Cristian Asenjo	4
CAP-492	Bug	Minor	Josh Hum	0
CAP-493	Bug	Minor	Josh Hum	1.5
CAP-509	Bug	Minor	Christian Daher	2
CAP-513	Bug	Minor	Cristian Asenjo	2
CAP-520	Sub-task	Minor	Josh Hum	2
CAP-521	Sub-task	Minor	Josh Hum	2
CAP-522	Sub-task	Minor	Josh Hum	2
CAP-523	Sub-task	Minor	Josh Hum	0.5
CAP-524	Sub-task	Minor	Mikhail Levkovsky	0.5
CAP-525	Sub-task	Minor	Josh Hum	2
CAP-526	Sub-task	Minor	Josh Hum	2
CAP-36	Story	Major	Cynthia Donato	0
CAP-112	Story	Major	Matthew Tam	0
CAP-198	Task	Major	Katrina Anderson	0
CAP-199	Sub-task	Major	Katrina Anderson	3
CAP-204	Sub-task	Major	Katrina Anderson	5

CAP-205	Sub-task	Major	Katrina Anderson	5
CAP-206	Sub-task	Major	Katrina Anderson	9
CAP-207	Sub-task	Major	Katrina Anderson	4
CAP-258	Sub-task	Major	Katrina Anderson	5
CAP-344	Sub-task	Major	Cristian Asenjo	10
CAP-402	Bug	Major	Ryan Nasr	1
CAP-413	Bug	Major	Mikhail Levkovsky	1
CAP-422	Sub-task	Major	Cynthia Donato	2
CAP-429	Sub-task	Major	Cynthia Donato	3
CAP-432	Sub-task	Major	Matthew Tam	12
CAP-433	Sub-task	Major	Matthew Tam	32
CAP-434	Sub-task	Major	Matthew Tam	4
CAP-435	Sub-task	Major	Matthew Tam	3
CAP-438	Sub-task	Major	Cynthia Donato	0.5
CAP-439	Sub-task	Major	Cynthia Donato	0.5
CAP-442	Sub-task	Major	Cynthia Donato	1
CAP-446	Sub-task	Major	Katrina Anderson	3
CAP-456	Sub-task	Major	Katrina Anderson	3
CAP-459	Bug	Major	Ryan Nasr	3
CAP-461	Bug	Major	Ryan Nasr	3
CAP-463	Bug	Major	Patrick Modafferi	1
CAP-466	Sub-task	Major	Katrina Anderson	0.5
CAP-467	Sub-task	Major	Katrina Anderson	0.5
CAP-468	Sub-task	Major	Katrina Anderson	0.5
CAP-469	Sub-task	Major	Mikhail Levkovsky	2
CAP-470	Sub-task	Major	Christian Daher	8
CAP-471	Sub-task	Major	Mikhail Levkovsky	1
CAP-472	Sub-task	Major	Patrick Modafferi	0.5
CAP-473	Sub-task	Major	Ryan Nasr	1
CAP-474	Sub-task	Major	Josh Hum	4
CAP-475	Sub-task	Major	Josh Hum	3
CAP-476	Sub-task	Major	Josh Hum	1
CAP-477	Sub-task	Major	Patrick Modafferi	0.5
CAP-478	Sub-task	Major	Matthew Tam	1.5
CAP-480	Bug	Major	Ryan Nasr	3
CAP-484	Bug	Major	Patrick Modafferi	2
CAP-485	Sub-task	Major	Katrina Anderson	3
CAP-494	Bug	Major	Josh Hum	1.5
CAP-495	Bug	Major	Josh Hum	1.5
CAP-496	Bug	Major	Josh Hum	1
CAP-497	Bug	Major	Josh Hum	1.5

CAP-498	Bug	Major	Josh Hum	0.166666667
CAP-499	Bug	Major	Josh Hum	1.5
CAP-500	Bug	Major	Patrick Modafferi	2
CAP-501	Bug	Major	Cristian Asenjo	5
CAP-502	Bug	Major	Ryan Nasr	2
CAP-503	Bug	Major	Christian Daher	2
CAP-504	Bug	Major	Cristian Asenjo	2
CAP-505	Bug	Major	Cristian Asenjo	3
CAP-506	Bug	Major	Christian Daher	2
CAP-507	Bug	Major	Christian Daher	2
CAP-510	Bug	Major	Mikhail Levkovsky	0
CAP-511	Bug	Major	Mikhail Levkovsky	1
CAP-512	Bug	Major	Cristian Asenjo	4
CAP-514	Sub-task	Major	Patrick Modafferi	7
CAP-515	Sub-task	Major	Mikhail Levkovsky	6
CAP-516	Bug	Major	Patrick Modafferi	4
CAP-519	Sub-task	Major	Mikhail Levkovsky	10
CAP-527	Sub-task	Major	Josh Hum	10
CAP-528	Sub-task	Major	Katrina Anderson	1
CAP-529	Sub-task	Major	Cristian Asenjo	10
CAP-530	Sub-task	Major	Katrina Anderson	3
CAP-531	Sub-task	Major	Katrina Anderson	3
CAP-532	Sub-task	Major	Katrina Anderson	3
CAP-533	Sub-task	Major	Katrina Anderson	3
CAP-534	Sub-task	Major	Katrina Anderson	15
CAP-535	Sub-task	Major	Katrina Anderson	10
CAP-536	Bug	Major	Katrina Anderson	1
CAP-539	Sub-task	Major	Cynthia Donato	2
CAP-540	Sub-task	Major	Cynthia Donato	3
CAP-541	Sub-task	Major	Cynthia Donato	0.5
CAP-542	Sub-task	Major	Cynthia Donato	0.5
CAP-543	Sub-task	Major	Cynthia Donato	5
CAP-544	Sub-task	Major	Cynthia Donato	1
CAP-558	Bug	Major	Mikhail Levkovsky	2
CAP-546	Bug	Critical	Cristian Asenjo	1
CAP-553	Bug	Critical	Katrina Anderson	6
CAP-517	Bug	Blocker	Christian Daher	4
CAP-518	Bug	Blocker	Christian Daher	4
CAP-569	Bug	Blocker	Katrina Anderson	2
			TOTAL	**356.42**

21.2 Person-Hour Estimation

The Expert Judgement Method was used to calculate person-hour estimations.

Table 21-2 Person-Hour Estimation

Description	Worst Case	Most Likely Case	Best Case	Expected Case
Activities	463.35	356.42	249.49	356.42
Total(ph)	463.35	356.42	249.49	356.42
Velocity(ph/day)	33.10	25.46	17.82	25.46
Velocity (ph/team member/day)	3.68	2.83	1.98	2.83

21.3 Activity-on-Node Planning

The following graph reflects the progress of the planned user stories. The numbers represent the Early Start, Duration, Early Finish, Late Start, Slack, and Late Finish in days. All remaining stories are currently in progress.

Although a few tasks are still left to complete, they involve just unit testing and documentation. The functionality will not change. By sprint 10, all stories should be complete. We are still on track to complete in time for our presentation.

Figure 21-1 Activity on Node Diagram

22 Iteration 9 Report

22.1 Person-Hour Work Log

Start Date: 27/Feb/13 End Date: 12/Mar/13 [Change] (UNREGISTERED)				Christian Daher	Cristian Asenjo	Cynthia Donato	Josh Hum	Katrina Anderson	Matthew Tam	Mikhail Levkovsky	Patrick Modafferi	Ryan Nasr	Total
Issue			Total	14.017h	19.917h	19.183h	46.5h	73.95h	17.4h	18.783h	24.55h	19.833h	254.133h
CAP-9	Documentation - Vision Document	↑	0.5h	0h	0h	0h	0h	0h	0h	0h	0.5h	0h	0.5h
CAP-11	Documentation - SAD	↑	43.15h	0.017h	3h	1h	19.333h	3.7h	2h	8.35h	3.75h	2h	43.15h
CAP-14	Correspondence	↑	19.15h	0h	0.633h	2.167h	3.733h	7.267h	0.167h	1.017h	4.167h	0h	19.15h
CAP-18	Meetings	↑	23.167h	0h	1.667h	1.667h	5h	4h	3.167h	4.167h	3.5h	0h	23.167h
CAP-23	Setup	↑	7.833h	0h	0h	0h	1.833h	0.5h	0h	0h	2h	3.5h	7.833h
CAP-36	Restaurant Statistics and Reports	↑	5h	0h	0h	4.75h	0h	0h	0h	0.25h	0h	0h	5h
CAP-44	Documentation - Test Plan	↓	0.7h	0h	0h	0h	0h	0.7h	0h	0h	0h	0h	0.7h
CAP-57	Documentation - Management	↑	6.167h	0h	0h	0h	6h	0h	0h	0.167h	0h	0h	6.167h
CAP-112	Toppings and Sides	↑	12.05h	0h	0h	0h	0h	0h	12.05h	0h	0h	0h	12.05h
CAP-162	Look and Feel	↑	5.117h	0h	0h	0h	0h	5.117h	0h	0h	0h	0h	5.117h
CAP-164	Documentation - Test Report	↑	9.6h	0h	0h	9.6h	0h	0h	0h	0h	0h	0h	9.6h
CAP-198	Unit Testing Retroactive	↑	49.5h	0h	0h	0h	0h	49.5h	0h	0h	0h	0h	49.5h
CAP-314	Code Review	↑	0.2h	0h	0h	0h	0.2h	0h	0h	0h	0h	0h	0.2h
CAP-342	iPhone App	↓	7.633h	0h	7.633h	0h	0h	0h	0h	0h	0h	0h	7.633h
CAP-390	Empty Menu Category Names	↔	1h	0h	0h	0h	0h	0h	0h	0h	0h	1h	1h
CAP-402	OM which do soft deletes should update version	↑	1h	0h	0h	0h	0h	0h	0h	1h	0h	0h	1h
CAP-390	Empty Menu Category Names	↔	1h	0h	0h	0h	0h	0h	0h	0h	0h	1h	1h
CAP-402	OM which do soft deletes should update version	↑	1h	0h	0h	0h	0h	0h	0h	1h	0h	0h	1h
CAP-405	use global resource folder for strings	↓	0.767h	0h	0h	0h	0h	0.417h	0.017h	0.333h	0h	0h	0.767h
CAP-413	Bill TVQ TPS and TOTAL rounding	↑	0.25h	0h	0h	0h	0h	0h	0h	0.25h	0h	0h	0.25h
CAP-460	logic in view that doesn't belong	↓	1h	0h	0h	0h	0h	0h	0h	0h	0h	1h	1h
CAP-461	Menu Activation Cascade to MenuCategory and MenuItem	↑	1.75h	0h	0h	0h	0h	0h	0h	0h	1.75h	0h	1.75h
CAP-462	duplicate files	↓	1h	0h	0h	0h	0h	0h	0h	0h	0h	1h	1h
CAP-463	Order/Details line 64 errors	↑	0.25h	0h	0h	0h	0h	0h	0h	0h	0.25h	0h	0.25h
CAP-480	Show nothing to a customer until they have a table	↑	3h	0h	0h	0h	0h	0h	0h	0h	0h	3h	3h
CAP-486	Remove Consloe.WriteLine(e) From Controllers	↔	0.083h	0h	0h	0h	0.083h	0h	0h	0h	0h	0h	0.083h
CAP-487	Mapper Missing In Item Controller Edit	↓	1h	0h	0h	0h	0h	0h	0h	0h	0h	1h	1h
CAP-488	Menu Category Controller	↔	0.167h	0h	0h	0h	0h	0h	0h	0.167h	0h	0h	0.167h
CAP-489	Menu_CategoryController getDB method	↓	0.083h	0h	0h	0h	0h	0h	0h	0.083h	0h	0h	0.083h
CAP-490	When a developer or administrator create a user, they should be able to choose what role that user should have	↓	0.083h	0h	0h	0h	0.083h	0h	0h	0h	0h	0h	0.083h
CAP-491	Passwords should be encrypted	↔	1.717h	0h	1.717h	0h	0h	0h	0h	0h	0h	0h	1.717h
CAP-493	Administrators cannot see menus at all. They should have privileges to manage menus.	↓	0.083h	0h	0h	0h	0h	0h	0h	0h	0.083h	0h	0.083h
CAP-494	menu_item and menu_category is_active column must be set to true by default	↑	0.5h	0h	0h	0h	0h	0h	0h	0h	0.5h	0h	0.5h

CAP-495	Restaurant and administrator search does not work. They should not have to be tapped into a table. Just use their own resto id.		**1.167h**	0h	0h	0h	1.167h	0h	0h	0h	0h	0h	**1.167h**
CAP-496	Correct main menu to show what each user role should see.		**0.083h**	0h	0h	0h	0.083h	0h	0h	0h	0h	0h	**0.083h**
CAP-497	Restaurant should only see its own menus, categories, items, tables and orders		**3h**	0h	0h	0h	0h	0h	0h	0h	3h	0h	**3h**
CAP-498	The box to create an item says category at the top. Should be changed to Item.		**1.667h**	0h	0h	0h	1.667h	0h	0h	0h	0h	0h	**1.667h**
CAP-500	Remove Item from category does not work. Remove category from menu does not work.		**2.75h**	0h	0h	0h	0h	0h	0h	0h	2.75h	0h	**2.75h**
CAP-501	When a restaurant user is created, they need to be assigned a restaurant to manage		**7.083h**	0h	0h	0h	7.083h	0h	0h	0h	0h	0h	**7.083h**
CAP-502	Customer order does not show that it has been declined after the restaurant declines it		**2h**	0h	0h	0h	0h	0h	0h	0h	0h	2h	**2h**
CAP-503	Must handle a user trying to add to an order if they have already finalized		**3h**	3h	0h	0h	0h	0h	0h	0h	0h	0h	**3h**
CAP-504	Customer should be able to make service request without having to login again		**0.017h**	0h	0.017h	0h	0h	0h	0h	0h	0h	0h	**0.017h**
CAP-505	Password must be case sensitive		**1.717h**	0h	1.717h	0h	0h	0h	0h	0h	0h	0h	**1.717h**
CAP-506	The resto should only be able to create tables at his own restaurant		**2h**	2h	0h	0h	0h	0h	0h	0h	0h	0h	**2h**
CAP-507	System crashes on empty create new table form. Should display error message		**2h**	2h	0h	0h	0h	0h	0h	0h	0h	0h	**2h**
CAP-508	Tables should not be created/edited without names.		**1h**	1h	0h	0h	0h	0h	0h	0h	0h	0h	**1h**
CAP-509	Table should not show on the list after being deleted.		**0.167h**	0h	0h	0h	0.167h	0h	0h	0h	0h	0h	**0.167h**
CAP-511	Fix flicker on Star Ratings		**0.5h**	0h	0h	0h	0h	0h	0h	0.5h	0h	0h	**0.5h**
CAP-512	[iOS] Can't scan QR code if you cancel out from the camera view		**1.5h**	0h	1.5h	0h	0h	0h	0h	0h	0h	0h	**1.5h**
CAP-513	[iOS] Pinch-to-zoom has to be disabled in the TFF tab		**1.017h**	0h	1.017h	0h	0h	0h	0h	0h	0h	0h	**1.017h**
CAP-516	Adding to Order from Suggestions causes some error when finalizing an order		**0.667h**	0h	0h	0h	0h	0h	0h	0h	0.667h	0h	**0.667h**
CAP-517	Manage Order Status - Set to delivered upon serving the food to a client		**3h**	3h	0h	0h	0h	0h	0h	0h	0h	0h	**3h**
CAP-518	Manage Order Status - Set to complete once the user has paid		**3h**	3h	0h	0h	0h	0h	0h	0h	0h	0h	**3h**
CAP-536	Menu_CategoryControllerTest - AddItemTest Fails		**0.417h**	0h	0h	0h	0h	0.417h	0h	0h	0h	0h	**0.417h**
CAP-545	Review and Review_Order Unit Tests Fail		**0.667h**	0h	0h	0h	0h	0.333h	0h	0h	0.333h	0h	**0.667h**
CAP-546	Restaurant Controller Create Exception Handling and Error Msg Needed		**1.017h**	0h	1.017h	0h	0h	0h	0h	0h	0h	0h	**1.017h**
CAP-548	Restaurant Controller Edit(restaurant) Exception Msg		**1h**	0h	0h	0h	0h	0h	0h	0h	0h	1h	**1h**
CAP-549	Restaurant Controller DeleteConfirmed Exception Message		**0.5h**	0h	0h	0h	0h	0h	0h	0h	0h	0.5h	**0.5h**
CAP-554	Table Controller Exception Handling & Error Message Missing From Create(table)		**0.5h**	0h	0h	0h	0h	0h	0h	0h	0h	0.5h	**0.5h**
CAP-556	Table Controller DeleteConfirmed Exception Msg		**0.5h**	0h	0h	0h	0h	0h	0h	0h	0h	0.5h	**0.5h**
CAP-557	Editing a menu category makes it inactive		**0.167h**	0h	0h	0h	0h	0h	0h	0h	0.167h	0h	**0.167h**
CAP-558	Missing Test Cases		**0.5h**	0h	0h	0h	0h	0h	0h	0.5h	0h	0h	**0.5h**
CAP-561	User Controller Create(user) Missing Exception Handling and Error Messages		**1h**	0h	0h	0h	0h	0h	0h	0h	0h	1h	**1h**
CAP-564	User Controller Edit(user) method Exception Handling Issues & Missing Error Msgs		**0.5h**	0h	0h	0h	0h	0h	0h	0h	0h	0.5h	**0.5h**
CAP-566	User Controller DeleteConfirmed Exception Message		**0.333h**	0h	0h	0h	0h	0h	0h	0h	0h	0.333h	**0.333h**
CAP-569	Resource Files Don't Work When Unit Testing		**1.75h**	0h	0h	0h	0h	1.5h	0h	0.25h	0h	0h	**1.75h**
CAP-573	Edit Category does not reflect the name change		**1.5h**	0h	0h	0h	0h	0h	0h	1.5h	0h	0h	**1.5h**

CAP-575	Error Message should display, if we try removing an active Menu Category (same for menu item)		**1h**	0h	0h	0h	0h	0h	0h	0h	0h	1h	**1h**
CAP-576	MenuCategoryOM Delete Method Doesn't Take Care of Side Association in Cascading Delete.		**0.017h**	0h	0h	0h	0.017h	0h	0h	0h	0h	0h	**0.017h**
CAP-577	MenuCategoryOM Delete Method Doesn't Check For Active Menu Status As Indicated In Comment.		**0.083h**	0h	0h	0h	0h	0h	0h	0h	0.083h	0h	**0.083h**
CAP-579	User Menu should use IM and OM to display only active categories and items		**0.333h**	0h	0h	0h	0h	0h	0h	0h	0.333h	0h	**0.333h**
CAP-580	Test Suggestion system against larger data set		**0.5h**	0h	0h	0h	0h	0h	0h	0h	0.5h	0h	**0.5h**
CAP-583	when menu item is soft deleted do not delete the order items		**0.25h**	0h	0h	0h	0h	0h	0h	0.25h	0h	0h	**0.25h**
CAP-586	Item Controller Tests Fail		**0.5h**	0h	0h	0h	0h	0.5h	0h	0h	0h	0h	**0.5h**
CAP-592	Customer role sees Table link in navigation bad		**0.017h**	0h	0h	0h	0.017h	0h	0h	0h	0h	0h	**0.017h**
CAP-594	Admin should have option of user role when creating a user		**0.017h**	0h	0h	0h	0.017h	0h	0h	0h	0h	0h	**0.017h**
CAP-595	No way to associate restaurant user to a resto		**0.017h**	0h	0h	0h	0h	0h	0h	0h	0.017h	0h	**0.017h**
CAP-598	Admin User should have developer rights		**0.017h**	0h	0h	0h	0.017h	0h	0h	0h	0h	0h	**0.017h**
CAP-599	Missing Menu Link in Developer View		**0.017h**	0h	0h	0h	0h	0h	0h	0h	0.017h	0h	**0.017h**
CAP-610	Customer cannot update profile		**0.017h**	0h	0h	0h	0h	0h	0h	0h	0.017h	0h	**0.017h**
CAP-615	Customer user cannot view suggestions		**0.167h**	0h	0h	0h	0h	0h	0h	0h	0.167h	0h	**0.167h**

Figure 22-1 Person-Hour Work Log

Our goal to work over 200 hours per sprint was once again met. Every team member put in the required hours to complete their tasks. We are satisfied with our progress and the project is advancing as desired towards completion.

22.2 Hour Burndown Chart

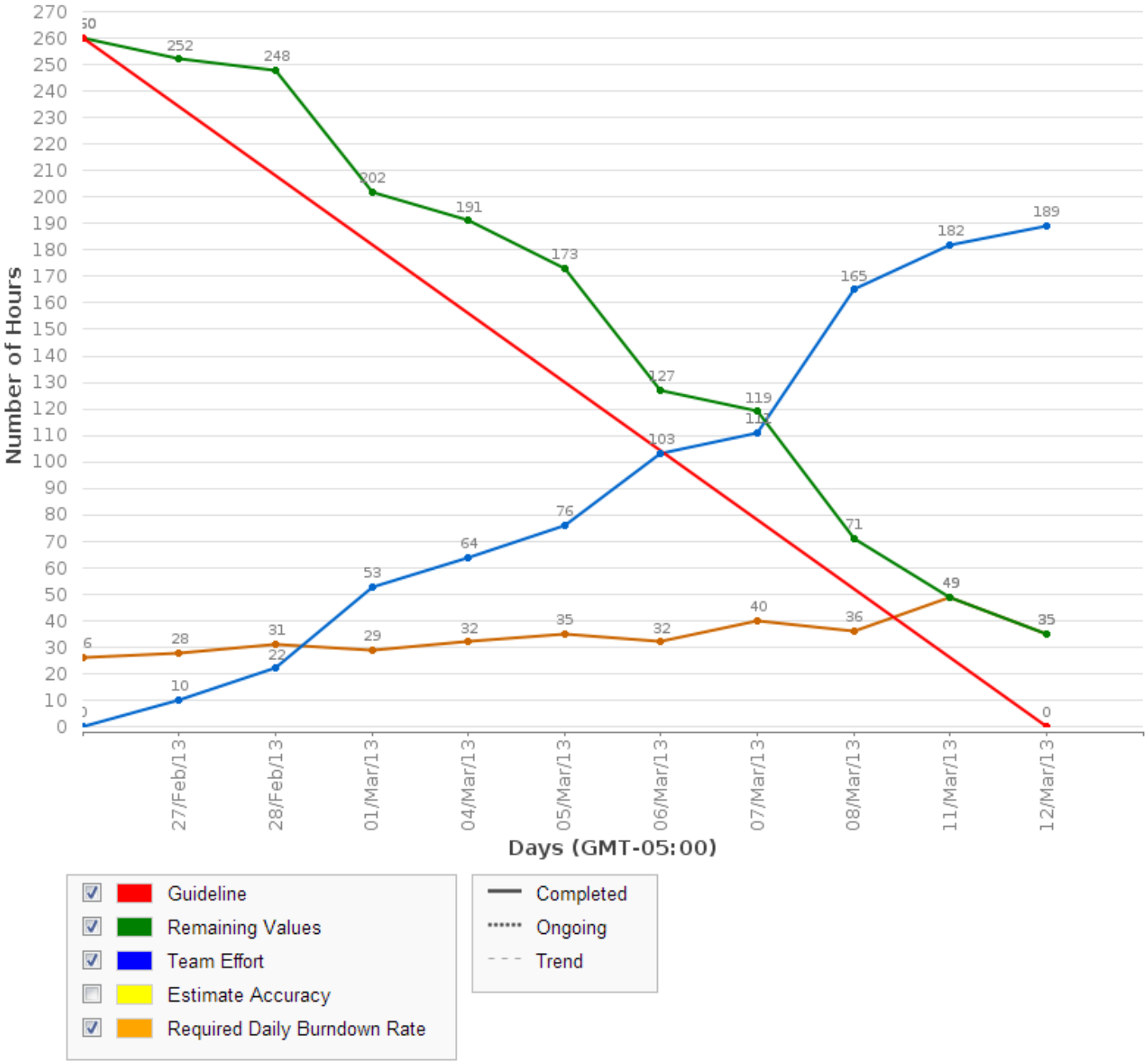

Figure 22-2 Hour Burndown Chart

Out of 260 estimated hours on all tasks, only 35 were not burned down. The 35 incomplete hours can be attributed to a few tasks that were not completed. The chart shows that there was a consistent burndown as team members worked continuously throughout the sprint.

22.3 Issue Burndown Chart

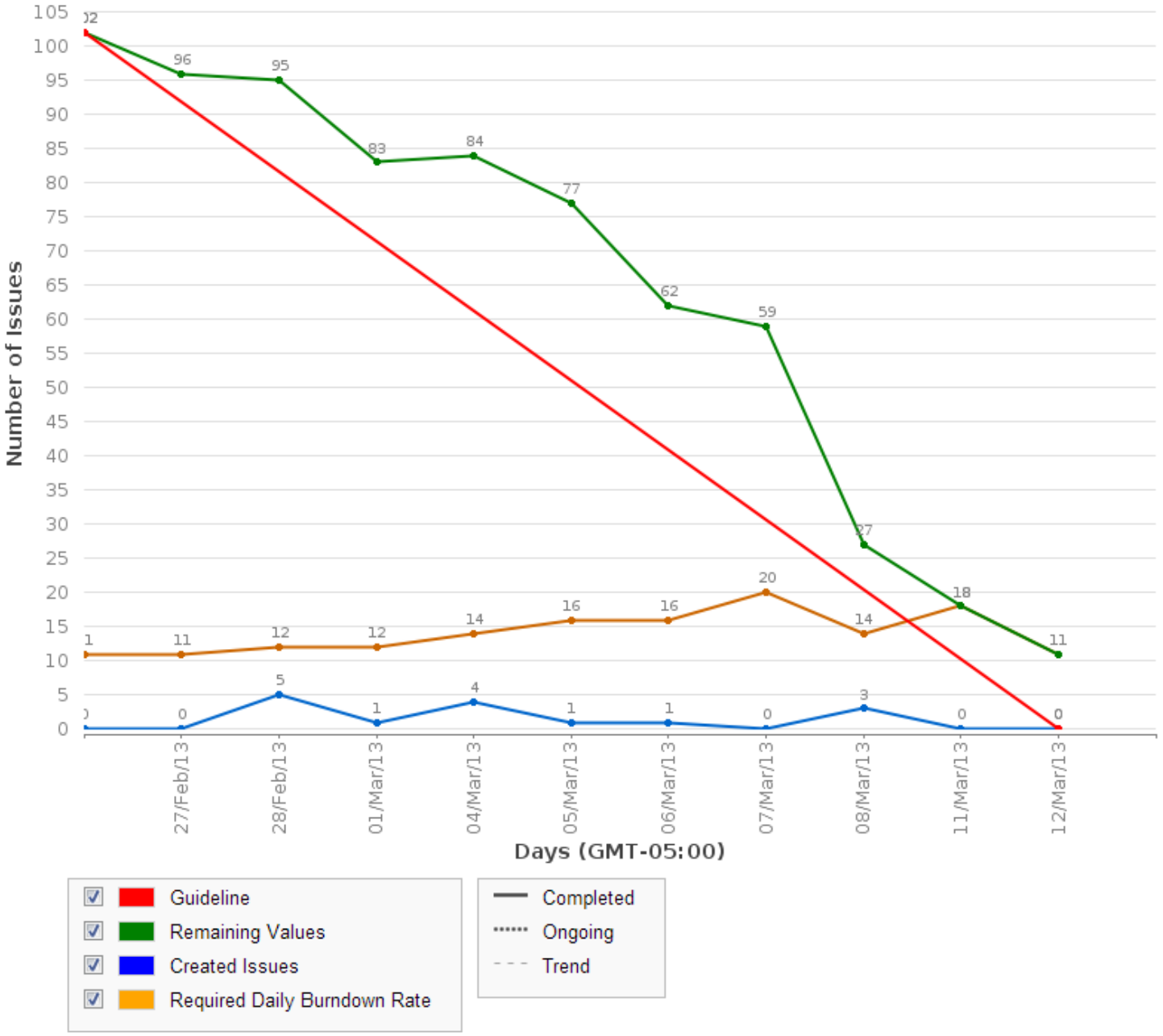

Figure 22-3 Issue Burndown Chart

91 out of the 102 planned tasks were completed this sprint. This is excellent and we are quite happy with our progress. With the upcoming sprint as the last planned sprint to work on coding tasks, we are well on our way to completion of the project.

22.4 Cumulative Flow Diagram

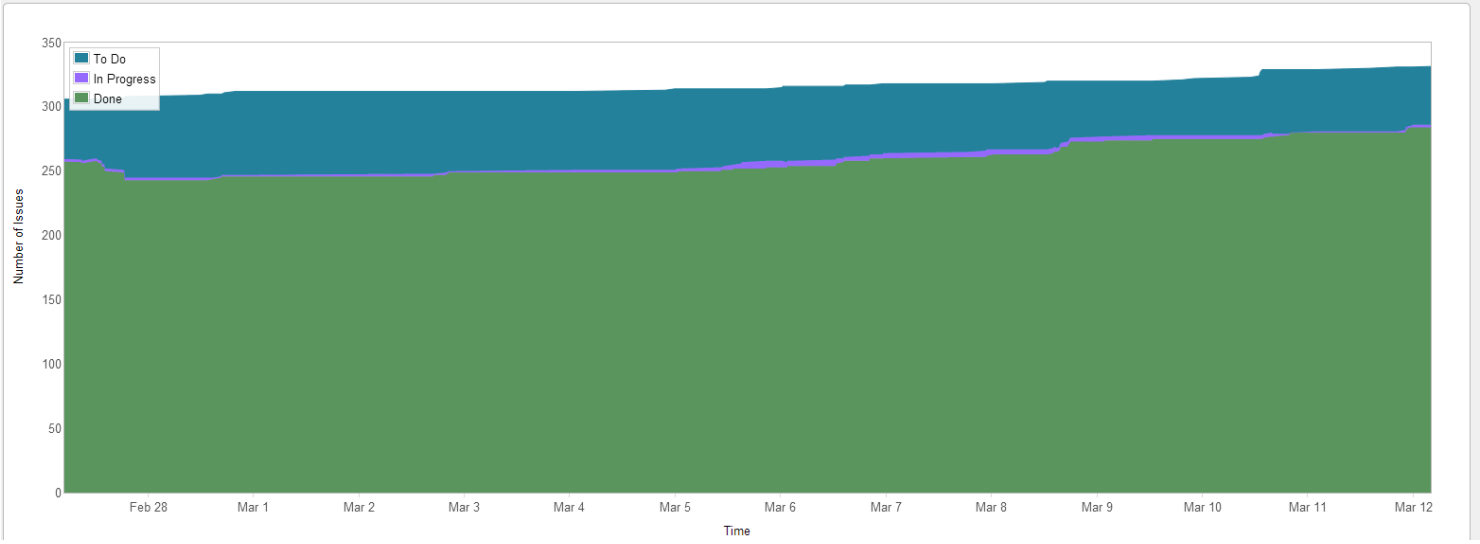

Figure 22-4 Cumulative Flow Diagram

The figure above reflects the steady progress that our team is making. With one full sprint left and having completed around 280 tasks, we are looking to close out the majority of the remaining tasks next sprint.

22.5 Measurement Report

22.5.1 Code Quality Analysis

The following report was generated after analyzing the code. Thirteen methods did not meet code quality goals by exceeding the expected number of lines of code or having none. They have been entered as bugs and will be fixed in a future iteration. Since it is a minor bug, the priority is not high for it which is why they occur repeatedly in the code quality analysis report. Given that there is only one sprint left, we will look to fix these issues in the next sprint or in the half sprint following it.

Since our only issues have been with lines of code, we took note for the future that the following scale defines lines of code quality:

0: red
1 – 10: green
11 – 10: yellow
21+: red

Analysis tool used: Code Metrics Viewer
Found at: http://visualstudiogallery.msdn.microsoft.com/9f35524b-a784-4dbc-bd7b-6babd7a5a3b3
Version: 1.5.3
Last updated: 2/5/2012

Scope	Hiearchy	Maintainability Index	Cyclomatic Complexity	Depth of Inheritance	Class Coupling	Lines of Code
ModuleScope	TouchForFood.Tests.dll	70	726	124	2	
NamespaceScope	TouchForFood.Tests	68	638	104	1	
TypeScope	BillControllerTest	64	42	54	1	137
TypeScope	CategoryControllerTest	68	30	32	1	110
TypeScope	CategoryOMTest	71	8	16	1	36
TypeScope	FriendshipControllerTest	64	37	34	1	116
TypeScope	ItemControllerTest	69	27	53	1	90
TypeScope	ItemOMTest	72	7	17	1	32
TypeScope	Menu_CategoryControllerTest	63	56	50	1	134
TypeScope	Menu_ItemControllerTest	60	53	54	1	189
TypeScope	MenuCategoryOMTest	68	9	25	1	46
TypeScope	MenuControllerTest	62	53	47	1	166
TypeScope	MenuItemOMTest	71	9	24	1	43
TypeScope	MenuOMTest	68	9	24	1	46
TypeScope	Order_ItemControllerTest	60	42	38	1	128
TypeScope	OrderControllerTest	80	10	25	1	25
TypeScope	OrderOMTest	70	15	25	1	44
TypeScope	RestaurantControllerTest	64	25	46	1	106
TypeScope	RestaurantUserControllerTest	67	21	40	1	66
TypeScope	Review_Order_ItemControllerTest	72	20	32	1	80
TypeScope	ReviewControllerTest	72	20	36	1	83
TypeScope	ReviewOMTest	76	6	14	1	22
TypeScope	ServiceRequestControllerTest	65	48	35	1	168
TypeScope	TableControllerTest	66	40	42	1	125
TypeScope	TableOMTest	75	8	17	1	25
TypeScope	UserControllerTest	65	35	41	1	156
TypeScope	WaiterOMTest	79	8	17	1	23
MemberScope	DeleteTest() : void	58	1	9		11
MemberScope	MyClassCleanup() : void	100	1	1		0
MemberScope	MyClassInitialize(TestContext) : void	94	1	3		1
MemberScope	MyTestCleanup() : void	76	1	6		4

MemberScope	MyTestInitialize() : void	73	1	6		4
MemberScope	TestContext.get() : TestContext	98	1	1		1
MemberScope	TestContext.set(TestContext) : void	95	1	1		1
MemberScope	WaiterOMTest()	100	1	0		1
NamespaceScope	TouchForFood.Tests.Classes	76	80	53	2	375
TypeScope	ContextMocks	86	14	21	1	27
TypeScope	ContextMocks.FakeSessionState	90	4	3	2	3
TypeScope	Session	62	4	19	1	29
TypeScope	TestDatabaseHelper	64	58	24	1	316
MemberScope	AddBill(order) : bill	64	1	6		9
MemberScope	AddCategory() : category	65	1	4		9
MemberScope	AddFriendship(user, user) : friendship	64	1	6		9
MemberScope	AddItem() : item	66	1	4		8
MemberScope	AddItem(category) : item	64	1	6		9
MemberScope	AddMenu(restaurant) : menu	62	1	5		11
MemberScope	AddMenuCategory(category, menu) : menu_category	63	1	6		10
MemberScope	AddMenuItem(item, menu_category) : menu_item	61	1	7		11
MemberScope	AddOrder(table) : order	63	1	7		10
MemberScope	AddOrder(table, waiter) : order	61	1	8		11
MemberScope	AddOrderItem(order, bill, menu_item) : order_item	61	1	8		11
MemberScope	AddOrderItem(order, menu_item) : order_item	62	1	7		10
MemberScope	AddRestaurant() : restaurant	61	1	4		12
MemberScope	AddRestaurantUser(user, restaurant) : restaurant_use	64	1	7		9
MemberScope	AddReview(restaurant, order, user) : review	60	1	9		12
MemberScope	AddReviewOrderItem(review, order_item, string, int)	60	1	8		12
MemberScope	AddServiceRequest(table) : service_request	59	1	7		13
MemberScope	AddSide(menu_category) : side	59	1	7		13
MemberScope	AddTable(restaurant) : table	64	1	6		9
MemberScope	AddUser(string, table, int) : user	56	1	6		16
MemberScope	AddWaiter(restaurant) : waiter	61	1	6		11
MemberScope	RemoveBill(bill) : void	69	2	4		5
MemberScope	RemoveCategory(category) : void	69	2	4		5
MemberScope	RemoveFriendship(friendship) : void	69	2	4		5
MemberScope	RemoveItem(item) : void	69	2	4		5
MemberScope	RemoveMenu(menu) : void	70	2	4		5
MemberScope	RemoveMenuCategory(menu_category) : void	69	2	4		5
MemberScope	RemoveMenuItem(menu_item) : void	69	2	4		5
MemberScope	RemoveOrder(order) : void	70	2	4		5
MemberScope	RemoveOrderItem(order_item) : void	69	2	4		5
MemberScope	RemoveRestaurant(restaurant) : void	69	2	4		5
MemberScope	RemoveRestaurantUser(restaurant_user) : void	69	2	4		5
MemberScope	RemoveReview(review) : void	69	2	4		5
MemberScope	RemoveReviewOrderItem(review_order_item) : void	69	2	4		5
MemberScope	RemoveServiceRequest(service_request) : void	69	2	4		5
MemberScope	RemoveSide(side) : void	69	2	4		5
MemberScope	RemoveTable(table) : void	69	2	4		5
MemberScope	RemoveUser(user) : void	69	2	4		5
MemberScope	RemoveWaiter(waiter) : void	69	2	4		5
MemberScope	TestDatabaseHelper()	100	1	0		1
NamespaceScope	TouchForFood.Tests.Controllers	79	8	22	1	16
TypeScope	HomeControllerTest	79	8	22	1	16
MemberScope	HomeControllerTest()	100	1	0		1
MemberScope	Index() : void	75	3	12		2
MemberScope	MyClassCleanup() : void	80	1	5		3
MemberScope	MyClassInitialize(TestContext) : void	64	1	9		8
MemberScope	TestContext.get() : TestContext	98	1	1		1
MemberScope	TestContext.set(TestContext) : void	95	1	1		1
ModuleScope	TouchForFood.dll	79		218	4	
NamespaceScope	TouchForFood	76	4	10	2	12
TypeScope	MvcApplication	76	4	10	2	12
MemberScope	Application_Start() : void	73	1	4		5
MemberScope	MvcApplication()	100	1	1		1
MemberScope	RegisterGlobalFilters(GlobalFilterCollection) : void	94	1	2		1
MemberScope	RegisterRoutes(RouteCollection) : void	69	1	3		5

NamespaceScope	TouchForFood.Attributes	84	2	5	3	4
TypeScope	AjaxAttribute	84	2	5	3	4
MemberScope	AjaxAttribute(bool)	87	1	1		2
MemberScope	IsValidForRequest(ControllerContext, MethodInfo) : b	83	1	4		2
NamespaceScope	TouchForFood.Controllers	66	760	146	3	
TypeScope	BillController	65	38	44	3	113
TypeScope	CategoryController	68	35	34	3	72
TypeScope	FriendshipController	69	23	29	3	45
TypeScope	HomeController	74	5	15	3	6
TypeScope	ItemController	70	32	35	3	74
TypeScope	Menu_CategoryController	67	45	31	3	88
TypeScope	Menu_ItemController	69	46	32	3	68
TypeScope	MenuController	67	45	37	3	90
TypeScope	Order_ItemController	66	33	40	3	76
TypeScope	OrderController	58	98	60	3	217
TypeScope	RestaurantController	74	21	21	3	39
TypeScope	RestaurantUserController	68	17	31	3	30
TypeScope	Review_Order_ItemController	69	29	23	3	56
TypeScope	ReviewController	60	39	45	3	94
TypeScope	SearchController	56	19	28	3	36
TypeScope	ServiceRequestController	61	102	48	3	140
TypeScope	SideController	67	31	32	3	65
TypeScope	TableController	62	57	58	3	93
TypeScope	UserController	63	45	58	3	126
MemberScope	Create() : ActionResult	74	2	9		3
MemberScope	Create(user) : ActionResult	49	7	25		22
MemberScope	Delete(int) : ActionResult	84	1	4		2
MemberScope	DeleteConfirmed(int) : ActionResult	68	3	13		5
MemberScope	Details(int) : ViewResult	84	1	4		2
MemberScope	Dispose(bool) : void	87	1	3		2
MemberScope	Edit(int) : ActionResult	69	3	8		5
MemberScope	Edit(user, HttpPostedFileBase) : ActionResult	54	8	17		13
MemberScope	GetAllReviews(user) : PartialViewResult	64	5	13		7
MemberScope	GetRestaurants() : List<restaurant>	86	1	4		2
MemberScope	HandleDbUpdateException(DbUpdateException) : voi	65	3	4		7
MemberScope	Index() : ViewResult	74	2	11		3
MemberScope	LogOff() : ActionResult	76	1	5		4
MemberScope	LogOn() : ViewResult	87	1	2		2
MemberScope	LogOn(string, string) : ActionResult	45	3	25		33
MemberScope	SuggestItems() : ViewResult	60	2	11		10
MemberScope	UserController()	75	1	4		4
NamespaceScope	TouchForFood.Exceptions	97	4	1	2	4
TypeScope	AssociationExistsException	97	2	1	2	2
TypeScope	ItemActiveException	97	2	1	2	2
MemberScope	ItemActiveException()	100	1	1		1
MemberScope	ItemActiveException(string)	98	1	1		1
NamespaceScope	TouchForFood.Mappers	75	252	66	2	709
TypeScope	BillIM	83	8	7	2	14
TypeScope	BillOM	72	8	15	2	23
TypeScope	CategoryIM	84	6	6	2	10
TypeScope	CategoryOM	70	8	13	2	25
TypeScope	ItemIM	84	6	6	2	10
TypeScope	ItemOM	73	7	12	2	23
TypeScope	MenuCategoryIM	68	8	14	2	30
TypeScope	MenuCategoryOM	65	9	15	2	38
TypeScope	MenuIM	64	12	24	2	48
TypeScope	MenuItemIM	68	8	14	2	30
TypeScope	MenuItemOM	73	7	9	2	24
TypeScope	MenuOM	65	9	15	2	38
TypeScope	Order_ItemIM	83	8	7	2	14
TypeScope	Order_ItemOM	70	5	12	2	20
TypeScope	OrderIM	84	6	6	2	10
TypeScope	OrderOM	65	14	17	2	46
TypeScope	RestaurantIM	77	6	14	2	15

TypeScope	RestaurantOM	68	21	25	2	53
TypeScope	Review_Order_ItemIM	82	6	14	2	10
TypeScope	Review_Order_ItemOM	82	4	5	2	9
TypeScope	ReviewIM	84	6	6	2	10
TypeScope	ReviewOM	72	9	15	2	25
TypeScope	ServiceRequestIM	73	8	15	2	23
TypeScope	ServiceRequestOM	73	6	9	2	21
TypeScope	SideIM	73	6	14	2	18
TypeScope	SideOM	69	8	13	2	23
TypeScope	TableIM	84	6	6	2	10
TypeScope	TableOM	72	11	12	2	28
TypeScope	UserIM	84	6	6	2	10
TypeScope	UserOM	72	8	12	2	26
TypeScope	WaiterIM	84	6	6	2	10
TypeScope	WaiterOM	76	6	10	2	15
MemberScope	clearOrder(ICollection<order>) : void	75	2	8		3
MemberScope	Create(waiter) : bool	79	1	5		3
MemberScope	delete(int) : int	67	1	5		7
MemberScope	WaiterOM()	100	1	1		1
MemberScope	WaiterOM(touch_for_foodEntities)	98	1	2		1
NamespaceScope	TouchForFood.Mappers.Abstract	90	5	1	1	8
TypeScope	GenericIM	86	2	1	1	4
TypeScope	GenericOM	94	3	1	1	4
MemberScope	delete(int) : int	100	1	0		0
MemberScope	GenericOM()	86	1	1		2
MemberScope	GenericOM(touch_for_foodEntities)	87	1	1		2
NamespaceScope	TouchForFood.Mappers.Search	68	6	16	2	18
TypeScope	SearchIM	68	6	16	2	18
MemberScope	findByUser(int) : int	55	4	16		16
MemberScope	SearchIM()	100	1	1		1
MemberScope	SearchIM(touch_for_foodEntities)	98	1	2		1
NamespaceScope	TouchForFood.Models	92	514	49	2	610
TypeScope	bill	92	23	7	1	24
TypeScope	category	92	11	4	1	13
TypeScope	CategoryFilterVM	83	9	6	1	16
TypeScope	friendship	93	11	2	1	11
TypeScope	item	92	19	5	1	20
TypeScope	ItemFilterVM	80	18	16	1	34
TypeScope	menu	92	17	5	1	18
TypeScope	menu_category	92	21	6	1	23
TypeScope	menu_item	92	21	6	1	22
TypeScope	MenuMetadata	94	7	4	1	7
TypeScope	order	88	36	15	1	62
TypeScope	order_item	92	29	10	1	30
TypeScope	OrderItemMetadata	100	1	0	1	1
TypeScope	OrderMetadata	95	3	2	1	3
TypeScope	OrderStatusHelper	64	28	7	1	47
TypeScope	OrderStatusHelper.OrderItemStatusEnum	100	0	0	1	0
TypeScope	OrderStatusHelper.OrderStatusEnum	100	0	0	1	0
TypeScope	restaurant	91	27	10	1	32
TypeScope	restaurant_user	93	11	3	1	11
TypeScope	RestaurantMetadata	94	9	1	1	9
TypeScope	RestaurantUserMetadata	94	5	2	1	5
TypeScope	review	92	23	9	1	24
TypeScope	review_order_item	93	17	5	1	17
TypeScope	ReviewMetadata	94	7	3	1	7
TypeScope	ReviewOrderItemMetadata	94	5	3	1	5
TypeScope	service_request	93	15	3	1	15
TypeScope	side	92	19	6	1	20
TypeScope	table	91	16	9	1	22
TypeScope	TableMetadata	100	1	1	1	1
TypeScope	touch_for_foodEntities	92	38	22	2	38
TypeScope	user	91	35	9	1	40
TypeScope	UserMetadata	93	17	5	1	17

TypeScope	waiter	93	15	5	1	16
MemberScope	first_name.get() : string	98	1	0		1
MemberScope	first_name.set(string) : void	95	1	0		1
MemberScope	id.get() : int	98	1	0		1
MemberScope	id.set(int) : void	95	1	0		1
MemberScope	last_name.get() : string	98	1	0		1
MemberScope	last_name.set(string) : void	95	1	0		1
MemberScope	orders.get() : ICollection<order>	98	1	2		1
MemberScope	orders.set(ICollection<order>) : void	95	1	2		1
MemberScope	restaurant.get() : restaurant	98	1	1		1
MemberScope	restaurant.set(restaurant) : void	95	1	1		1
MemberScope	resto_id.get() : int?	98	1	1		1
MemberScope	resto_id.set(int?) : void	95	1	1		1
MemberScope	version.get() : int	98	1	0		1
MemberScope	version.set(int) : void	95	1	0		1
MemberScope	waiter()	87	1	2		2
NamespaceScope	TouchForFood.Util	71	6	1	1	10
TypeScope	StringUtilities	71	6	1	1	10
MemberScope	ExceptBlanks(string) : string	63	5	1		9
MemberScope	StringUtilities()	100	1	0		1
NamespaceScope	TouchForFood.Util.Bill	73	18	10	1	22
TypeScope	BillUtil	73	18	10	1	22
MemberScope	CheckFullyPaid(bill) : bool	65	5	8		7
MemberScope	GetTotalAfterTax(bill) : decimal	78	3	3		2
MemberScope	GetTotalBeforeTax(bill) : decimal	85	1	3		2
MemberScope	GetTPS() : decimal	84	1	1		2
MemberScope	GetTVQ() : decimal	84	1	1		2
MemberScope	Update(ref bill) : void	62	7	9		7
NamespaceScope	TouchForFood.Util.Category	55	8	22	1	33
TypeScope	CategoryUtil	55	8	22	1	33
MemberScope	CategoryUtil()	100	1	0		1
MemberScope	filterListByMenu(menu, touch_for_foodEntities) : IList	45	7	22		32
NamespaceScope	TouchForFood.Util.Html	69	16	26	1	39
TypeScope	HtmlDropDownExtensions	68	10	17	1	25
TypeScope	ImageActionLinkHelper	64	1	7	1	8
TypeScope	UrlUtils	75	5	4	1	6
MemberScope	ConvertRelativeUrlToAbsoluteUrl(string) : string	67	4	4		5
MemberScope	UrlUtils()	100	1	0		1
NamespaceScope	TouchForFood.Util.Item	52	10	25	1	38
TypeScope	ItemUtil	52	10	25	1	38
MemberScope	filterListByItem(menu_category) : IList<item>	43	9	25		37
MemberScope	ItemUtil()	100	1	0		1
NamespaceScope	TouchForFood.Util.Order	82	25	21	1	53
TypeScope	OrderStatusUtil	95	2	1	1	2
TypeScope	OrderUtil	68	23	21	1	51
MemberScope	filterItem(menu_item) : item	77	1	4		3
MemberScope	filterMenuItem(order_item) : menu_item	74	1	5		4
MemberScope	filterSide(order_item) : side	77	1	5		3
MemberScope	filterTable(order) : table	77	1	5		3
MemberScope	filterUser(order) : user	77	1	5		3
MemberScope	filterWaiter(order) : waiter	77	1	5		3
MemberScope	mergeExistingOrderToDb(order) : void	48	9	13		23
MemberScope	OrderUtil()	100	1	0		1
MemberScope	OrderUtil()	94	1	1		1
MemberScope	UpdatePrice(ref order) : void	63	6	11		7
NamespaceScope	TouchForFood.Util.Review	75	24	12	1	56
TypeScope	Rating	68	11	5	1	23
TypeScope	Rating.ReviewRatings	100	0	1	1	0
TypeScope	TextParser	58	13	8	1	33
MemberScope	ParseReviewText(List<review_order_item>, string) : v	45	10	7		30
MemberScope	SetDate(List<review_order_item>) : void	80	2	3		2
MemberScope	TextParser()	100	1	0		1
NamespaceScope	TouchForFood.Util.Search	63	31	34	1	114
TypeScope	SearchService	51	3	18	1	45

TypeScope	SearchUtil	60	23	19	1	63
TypeScope	SearchViewModelHelper	78	5	9	1	6
MemberScope	PopulateSearchViewModel(menu_item) : SearchView	76	1	6		2
MemberScope	PopulateSearchViewModelList(IList<menu_item>) : IL	76	3	5		3
MemberScope	SearchViewModelHelper()	100	1	0		1
NamespaceScope	TouchForFood.Util.Security	77	21	22	4	78
TypeScope	AES	67	14	11	1	62
TypeScope	CustomAuthorizationAttribute	64	7	10	4	16
TypeScope	SiteRoles	100	0	1	1	0
NamespaceScope	TouchForFood.Util.ServiceRequest	100	1	0	1	1
TypeScope	ServiceRequestUtil	100	1	0	1	1
TypeScope	ServiceRequestUtil.ServiceRequestStatus	100	0	0	1	0
NamespaceScope	TouchForFood.Util.Session	54	5	15	1	33
TypeScope	SessionUtil	54	5	15	1	33
MemberScope	getOpenOrder(user) : order	45	4	15		32
MemberScope	SessionUtil()	100	1	0		1
NamespaceScope	TouchForFood.Util.Table	64	4	10	1	8
TypeScope	TableUtil	64	4	10	1	8
MemberScope	GetRestaurants(HttpRequestBase) : List<restaurant>	64	4	10		8
NamespaceScope	TouchForFood.Util.User	58	39	43	1	114
TypeScope	UserUtil	58	39	43	1	114
MemberScope	BuildMetaScores(user) : Dictionary<string, int[]>	47	12	16		26
MemberScope	CalculateItemScores(Dictionary<string, int[]>, List<me	48	7	7		25
MemberScope	ConfidenceValue(int, int) : float	75	1	0		4
MemberScope	getAuthenticatedUser(HttpContextBase) : user	64	2	14		7
MemberScope	getAuthenticatedUser(HttpRequestBase) : user	63	2	8		9
MemberScope	GetSuggestions(user, restaurant) : List<KeyValuePair<	45	9	20		32
MemberScope	isUserRole(SiteRoles, HttpContext) : bool	84	1	3		2
MemberScope	SortByScore(Dictionary<menu_item, int>) : List<KeyV	67	3	7		6
MemberScope	UserUtil()	100	1	0		1
MemberScope	UserUtil()	84	1	1		2
NamespaceScope	TouchForFood.ViewModels	83	80	25	3	123
TypeScope	OrderItemVM	69	1	4	1	6
TypeScope	OrderVM	76	23	11	1	32
TypeScope	ReviewVM	89	17	10	1	23
TypeScope	SearchViewModel	91	16	1	1	23
TypeScope	ServiceRequestVM	94	5	3	3	5
TypeScope	SideFilterVM	80	18	17	1	34
MemberScope	AddSide(side) : void	94	1	2		1
MemberScope	FirstOrDefault() : side	71	2	3		5
MemberScope	GetRegularPrice(int) : decimal?	55	6	11		15
MemberScope	menu_cat.get() : menu_category	98	1	1		1
MemberScope	menu_cat.set(menu_category) : void	95	1	1		1
MemberScope	new_price.get() : double	98	1	0		1
MemberScope	new_price.set(double) : void	95	1	0		1
MemberScope	SideFilterVM()	80	1	3		3
MemberScope	SideFilterVM()	94	1	1		1
MemberScope	SideFilterVM(menu_category, IList<side>)	82	1	3		3
MemberScope	sides.get() : IList<side>	98	1	2		1
MemberScope	sides.set(IList<side>) : void	95	1	2		1

Figure 22-5 Code Quality Report

22.6 Retrospective

In Iteration 9, we closed out all remaining unit tests and stories. The goal of the sprint was to work on defects. Almost all reported bugs for the iteration were completed and closed. There are now no more stories to finish.

There were no major issues this sprint. Everything went smoothly. We no longer had a designated role for a bug basher as most people were assigned bugs as tasks. We continued to have a QA role and multiple team members reviewed code. We decided that if you are working on bugs, you can work in trunk instead of a branch since most bugs were small and needed to be completed quickly.

22.6.1 Velocity

Sprint 9 velocity (story points): 37 story points
Cumulative velocity (story points): 128 story points
Average velocity (story points): 14.2 story points

Sprint 9 velocity (p-h): 254.13 person-hours
Cumulative velocity (p-h): 1,725.72 person-hours
Average velocity per sprint: 191.75 person-hours

Our average story point velocity finished at 14.2 story points per sprint. Since all stories are complete, this will not change anymore until the end of the project. Our average person-hour velocity went up again this sprint as we worked over 250 hours.

22.6.2 Budget

Total person-hours budgeted to date: 4,554 person-hours
Total person-hours worked to date: 1,725.72 person-hours

Person-hours budgeted per sprint: 414 person-hours
Person-hours worked in Sprint 8: 254.13 person-hours

In Iteration 9, we continued to work over 200 hours per sprint as we worked 254 hours. We are satisfied with where we are in terms of hours worked as all team members continue to work hard.

22.6.3 What was learned

The following positive and negative points were gathered from this sprint.

Positives:

- On schedule to finish the project in time
- Communication and meetings are going very well
- Everyone is doing their part and working as a team
- Met our goal in person-hours worked

Negatives:

- None

23 Iteration 10 Plan

The goal of Iteration 10 is to close out all defect tasks. Defects relate mostly to code although there are a few documentation defects. Preparation for the project demos will also begin in iteration 10. There are a few planned improvements as well. Tasks will be completed in order of priority.

We are aware that some team members will have less time to commit during this sprint as many other classes are requiring time over the next two weeks. However, this should not be an issue since all stories have already been completed. The estimated number of hours for the planned activities should be less than 200.

23.1 Planned Activities

Tables featured in this section of the document were created with JIRA [**2**].

Table 23-1 Planned Activities

Key	Issue Type	Priority	Assignee	Original Estimate (hours)
CAP-547	Sub-task	Trivial	Unassigned	0.166666667
CAP-559	Sub-task	Trivial	Unassigned	0.166666667
CAP-560	Sub-task	Trivial	Unassigned	0.166666667
CAP-562	Sub-task	Trivial	Unassigned	0.166666667
CAP-563	Sub-task	Trivial	Unassigned	0.166666667
CAP-565	Sub-task	Trivial	Unassigned	0.166666667
CAP-570	Sub-task	Trivial	Patrick Modafferi	1
CAP-572	Sub-task	Trivial	Unassigned	0.083333333
CAP-582	Bug	Trivial	Mikhail Levkovsky	0.083333333
CAP-584	Bug	Trivial	Ryan Nasr	0.083333333
CAP-594	Bug	Trivial	Unassigned	1
CAP-598	Bug	Trivial	Unassigned	2
CAP-619	Improvement	Trivial	Katrina Anderson	0.25
CAP-621	Improvement	Trivial	Katrina Anderson	0.5
CAP-405	Improvement	Minor	Matthew Tam	4
CAP-567	Bug	Minor	Cynthia Donato	1
CAP-573	Bug	Minor	Mikhail Levkovsky	1
CAP-610	Bug	Minor	Unassigned	0.166666667
CAP-611	Bug	Minor	Unassigned	0.166666667
CAP-613	Bug	Minor	Josh Hum	1
CAP-614	Bug	Minor	Ryan Nasr	1
CAP-616	Bug	Minor	Cristian Asenjo	0.5
CAP-622	Task	Minor	Josh Hum	0
CAP-623	Sub-task	Minor	Josh Hum	8
CAP-624	Sub-task	Minor	Josh Hum	2
CAP-625	Sub-task	Minor	Mikhail Levkovsky	0.5
CAP-626	Sub-task	Minor	Josh Hum	1.5

CAP-627	Sub-task	Minor	Josh Hum	1.5
CAP-628	Sub-task	Minor	Josh Hum	0.5
CAP-629	Sub-task	Minor	Josh Hum	2
CAP-630	Sub-task	Minor	Josh Hum	2
CAP-636	Bug	Minor	Christian Daher	0.5
CAP-645	Bug	Minor	Cristian Asenjo	0.083333333
CAP-112	Story	Major	Matthew Tam	0
CAP-162	Task	Major	Josh Hum	0
CAP-206	Sub-task	Major	Katrina Anderson	9
CAP-433	Sub-task	Major	Matthew Tam	32
CAP-482	Sub-task	Major	Unassigned	0
CAP-545	Bug	Major	Patrick Modafferi	1
CAP-555	Bug	Major	Cynthia Donato	0.25
CAP-574	Sub-task	Major	Cristian Asenjo	6
CAP-576	Bug	Major	Unassigned	1
CAP-577	Bug	Major	Unassigned	1
CAP-578	Sub-task	Major	Unassigned	0.083333333
CAP-588	Bug	Major	Mikhail Levkovsky	0.25
CAP-589	Bug	Major	Mikhail Levkovsky	0.166666667
CAP-590	Bug	Major	Unassigned	1
CAP-591	Bug	Major	Cynthia Donato	0.5
CAP-592	Bug	Major	Unassigned	0.083333333
CAP-595	Bug	Major	Unassigned	1
CAP-602	Bug	Major	Josh Hum	1
CAP-603	Sub-task	Major	Ryan Nasr	1
CAP-605	Bug	Major	Matthew Tam	0.5
CAP-606	Sub-task	Major	Ryan Nasr	0.5
CAP-608	Bug	Major	Josh Hum	0.5
CAP-609	Bug	Major	Christian Daher	1
CAP-617	Bug	Major	Patrick Modafferi	4
CAP-618	Bug	Major	Josh Hum	4
CAP-620	Sub-task	Major	Josh Hum	0.5
CAP-631	Bug	Major	Unassigned	1
CAP-632	Sub-task	Major	Patrick Modafferi	0
CAP-633	Sub-task	Major	Patrick Modafferi	0
CAP-634	Sub-task	Major	Patrick Modafferi	0
CAP-635	Sub-task	Major	Patrick Modafferi	0
CAP-637	Bug	Major	Christian Daher	1
CAP-639	Bug	Major	Josh Hum	2
CAP-640	Bug	Major	Mikhail Levkovsky	2
CAP-641	Sub-task	Major	Mikhail Levkovsky	0.5

CAP-643	Bug	Major	Christian Daher	0
CAP-644	Sub-task	Major	Patrick Modafferi	1
CAP-646	Sub-task	Major	Unassigned	0
CAP-647	Bug	Major	Cynthia Donato	0
CAP-648	Sub-task	Major	Josh Hum	0.5
CAP-552	Bug	Critical	Cynthia Donato	15
CAP-568	Bug	Critical	Cristian Asenjo	2
CAP-579	Bug	Critical	Josh Hum	4
CAP-599	Bug	Critical	Unassigned	0.166666667
CAP-607	Bug	Critical	Christian Daher	1
CAP-615	Bug	Critical	Unassigned	2
CAP-642	Bug	Critical	Mikhail Levkovsky	0
CAP-651	Bug	Critical	Christian Daher	0
			TOTAL	**131.92**

23.2 Person-Hour Estimation

The Expert Judgement Method was used to calculate person-hour estimations.

Table 23-2 Person-Hour Estimation

Description	Worst Case	Most Likely Case	Best Case	Expected Case
Activities	171.50	131.92	92.34	131.92
Total(ph)	171.50	131.92	92.34	131.92
Velocity(ph/day)	12.25	9.42	6.60	9.42
Velocity (ph/team member/day)	1.36	1.05	0.73	1.05

23.3 Activity-on-Node Planning

The following graph reflects the progress of the planned user stories. The numbers represent the Early Start, Duration, Early Finish, Late Start, Slack, and Late Finish in days. All remaining stories are currently in progress.

All stories were completed in Sprint 9. The activity on node diagram now reflects that all stories are complete.

Figure 23-1 Activity on Node Diagram

24 Iteration 10 Report

24.1 Person-Hour Work Log

Start Date: 13/Mar/13 End Date: 26/Mar/13 [Change] (UNREGISTERED)				Christian Daher	Cristian Asenjo	Cynthia Donato	Josh Hum	Katrina Anderson	Matthew Tam	Mikhail Levkovsky	Patrick Modafferi	Ryan Nasr	Total
Issue			Total	6.5h	10.667h	13.383h	25.8h	5.017h	11.2h	9.7h	15.283h	9.5h	107.05h
CAP-14	Correspondence		5.617h	0h	0h	0.5h	2.3h	0.8h	0h	0.267h	1.75h	0h	5.617h
CAP-18	Meetings		14.133h	0h	2.8h	2.8h	2.8h	2.3h	2.633h	0.8h	0h	0h	14.133h
CAP-23	Setup		5.083h	0h	0h	0h	1.083h	0h	0h	0h	0.5h	3.5h	5.083h
CAP-57	Documentation - Management		3.083h	0h	0h	0h	3.083h	0h	0h	0h	0h	0h	3.083h
CAP-110	Documentation - UIR		1h	0h	0h	0h	0h	1h	0h	0h	0h	0h	1h
CAP-112	Toppings and Sides		4.033h	0h	0h	0h	0h	0h	4.033h	0h	0h	0h	4.033h
CAP-162	Look and Feel		10.3h	0h	4.783h	0h	0.417h	0h	0h	0.083h	1.017h	4h	10.3h
CAP-198	Unit Testing Retroactive		0.017h	0h	0h	0h	0.017h	0h	0h	0h	0h	0h	0.017h
CAP-247	Prevent Duplicate Menus		0.017h	0h	0h	0h	0.017h	0h	0h	0h	0h	0h	0.017h
CAP-248	Prevent Duplicate Categories		0.017h	0h	0h	0h	0.017h	0h	0h	0h	0h	0h	0.017h
CAP-249	Prevent Duplicate Menu Item		0.017h	0h	0h	0h	0.017h	0h	0h	0h	0h	0h	0.017h
CAP-314	Code Review		0.417h	0h	0h	0h	0.25h	0.167h	0h	0h	0h	0h	0.417h
CAP-405	use global resource folder for strings		4.033h	0h	0h	0h	0h	0h	4.033h	0h	0h	0h	4.033h
CAP-484	Order with no Order Items		2h	2h	0h	0h	0h	0h	0h	0h	0h	0h	2h
CAP-499	Need an edit link/button on the item details page		0.017h	0h	0h	0h	0h	0h	0h	0h	0.017h	0h	0.017h
CAP-545	Review and Review_Order Unit Tests Fail		1h	0h	0h	0h	0h	0h	0h	0h	1h	0h	1h
CAP-552	Delete Methods In All Controllers Do Not Support RollBack + Lock Missing (Versions)		5.5h	0h	0h	5.5h	0h	0h	0h	0h	0h	0h	5.5h
CAP-555	Table Controller Missing OM & Exception Handling In Edit(table)		0.083h	0h	0h	0.083h	0h	0h	0h	0h	0h	0h	0.083h
CAP-568	No way to edit your profile from customer standpoint		2.517h	0h	2.517h	0h	0h	0h	0h	0h	0h	0h	2.517h
CAP-571	Category Edit page broken links		1.75h	0h	0h	0h	1.75h	0h	0h	0h	0h	0h	1.75h
CAP-579	User Menu should use IM and OM to display only active categories and items		1.5h	0h	0h	0h	1.5h	0h	0h	0h	0h	0h	1.5h
CAP-582	Menu_ItemController Create Method Add Proper Error Message		0.117h	0h	0h	0h	0h	0h	0h	0.117h	0h	0h	0.117h
CAP-584	Menu_ItemController Edit(menu_item) Exception e generic message & Missing Message for invalid models.		0.5h	0h	0h	0h	0h	0h	0h	0h	0h	0.5h	0.5h
CAP-588	Order_ItemController No Exception Handling in DeleteConfirmed Method.		0.25h	0h	0h	0h	0h	0h	0h	0.25h	0h	0h	0.25h
CAP-589	TableController Create() throws exception if user does not have a restaurant_user association		0.167h	0h	0h	0h	0h	0h	0h	0.167h	0h	0h	0.167h
CAP-591	Create user gives null version		0.5h	0h	0h	0.5h	0h	0h	0h	0h	0h	0h	0.5h
CAP-602	Cannot edit a menu item as described in test case		0.667h	0h	0h	0h	0.667h	0h	0h	0h	0h	0h	0.667h
CAP-605	Hitting Place Order(add item to order) does not close the popup screen		0.5h	0h	0h	0h	0h	0h	0.5h	0h	0h	0h	0.5h
CAP-607	Accept order item sets other order items status to PENDING		1h	1h	0h	0h	0h	0h	0h	0h	0h	0h	1h
CAP-608	Need to format the confirmation for cancel service request		0.117h	0h	0h	0h	0.117h	0h	0h	0h	0h	0h	0.117h
CAP-609	if there are no items to be added to a bill , you can create an empty one		1h	1h	0h	0h	0h	0h	0h	0h	0h	0h	1h
CAP-614	Inconsistency in create/update table		1.5h	0h	0h	0h	0h	0h	0h	0h	0h	1.5h	1.5h

CAP-616	Editing a password must hash the password		**0.517h**	0h	0.517h	0h	0h	0h	0h	0h	0h	0h	0h	0h	**0.517h**
CAP-617	Should not see tables, service requests from other restaurants (remove category and item links)		**2.5h**	0h	0h	0h	0h	0h	0h	0h	2.5h	0h	**2.5h**
CAP-618	Order from search does not work		**0.75h**	0h	0h	0h	0.75h	0h	0h	0h	0h	0h	**0.75h**
CAP-619	Add Success Messages To Service Request.		**0.25h**	0h	0h	0h	0h	0.25h	0h	0h	0h	0h	**0.25h**
CAP-621	Add Check In Create Prevent User From Filling out form if service request is already created.		**0.5h**	0h	0h	0h	0h	0.5h	0h	0h	0h	0h	**0.5h**
CAP-622	Presentation		**23.017h**	0h	0h	0h	10.017h	0h	0h	6.5h	6.5h	0h	**23.017h**
CAP-636	Cancelling an order sends the user (tested with customer role) to the login page.		**0.5h**	0.5h	0h	0h	0h	0h	0h	0h	0h	0h	**0.5h**
CAP-637	When ordering an item, on the user's order page, the date display is messed up		**1h**	1h	0h	0h	0h	0h	0h	0h	0h	0h	**1h**
CAP-639	Make links for menu items and sides to popup their details pages. View is from a resto user.		**1h**	0h	0h	0h	1h	0h	0h	0h	0h	0h	**1h**
CAP-640	Ultimate Burger bug		**1h**	0h	0h	0h	0h	0h	0h	1h	0h	0h	**1h**
CAP-642	Null reviews being created when we don't select a star on the review page		**1.017h**	0h	0h	0h	0h	0h	0h	0.517h	0.5h	0h	**1.017h**
CAP-643	Can't leave a note on an order item even tho the field is show on user and resto side		**1h**	1h	0h	0h	0h	0h	0h	0h	0h	0h	**1h**
CAP-645	user Delete confirmation shows password		**0.05h**	0h	0.05h	0h	0h	0h	0h	0h	0h	0h	**0.05h**
CAP-647	Adding report stuff to suggestions		**4.5h**	0h	0h	3h	0h	0h	0h	0h	1.5h	0h	**4.5h**
CAP-649	Review all version error messages they are not all working properly		**1h**	0h	0h	1h	0h	0h	0h	0h	0h	0h	**1h**

Figure 24-1 Person-Hour Work Log

With 107 hours worked in Iteration 10, we came close to the original estimate for the planned activities. Although the number of hours is around half of what the team has been working per sprint over the past few sprints, this was expected and we are satisfied with the team effort.

24.2 Hour Burndown Chart

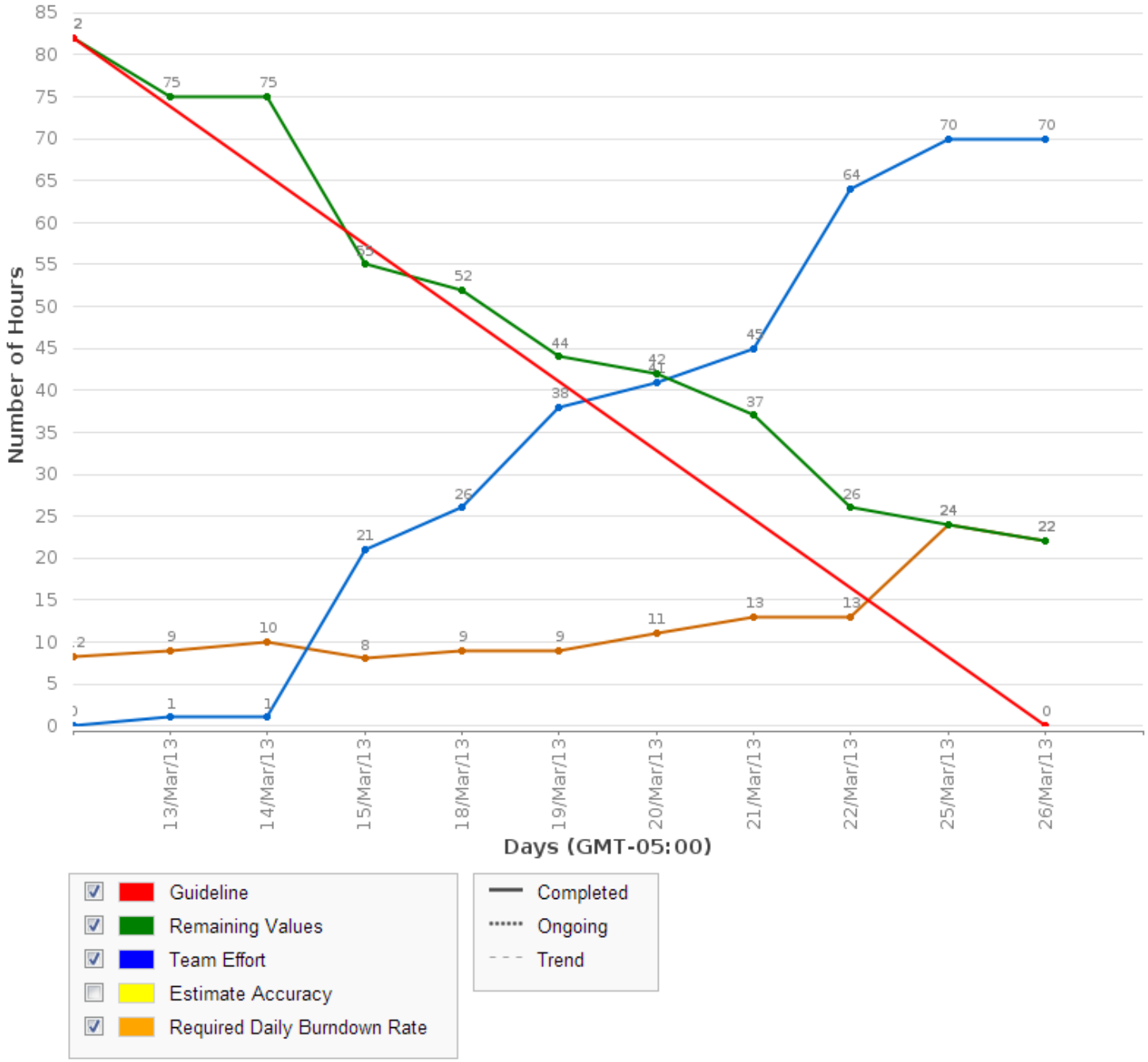

Figure 24-2 Hour Burndown Chart

The above chart shows the consistent work by team members throughout the sprint. There were 22 hours that were originally estimated that were not burned down. We attribute this mostly to faulty time logging. Some tasks that were completed did not get estimated down to 0 on completion. There were also a few incomplete tasks at the end of this sprint.

24.3 Issue Burndown Chart

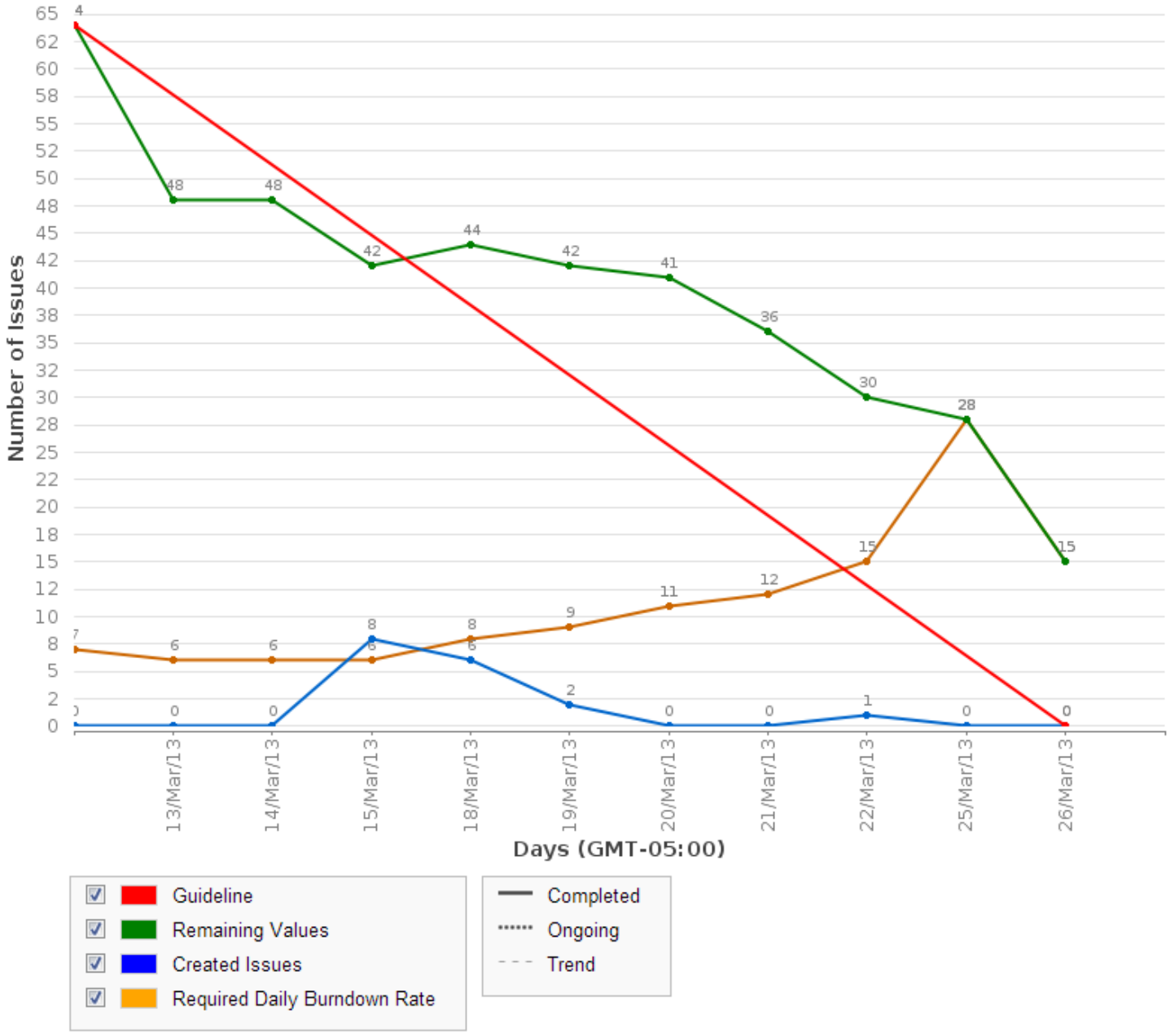

Figure 24-3 Issue Burndown Chart

15 issues were not completed at the end of the sprint. These are mostly documentation tasks and will be taken care of next sprint. There should be a little bit more time next sprint to work on these tasks.

24.4 Cumulative Flow Diagram

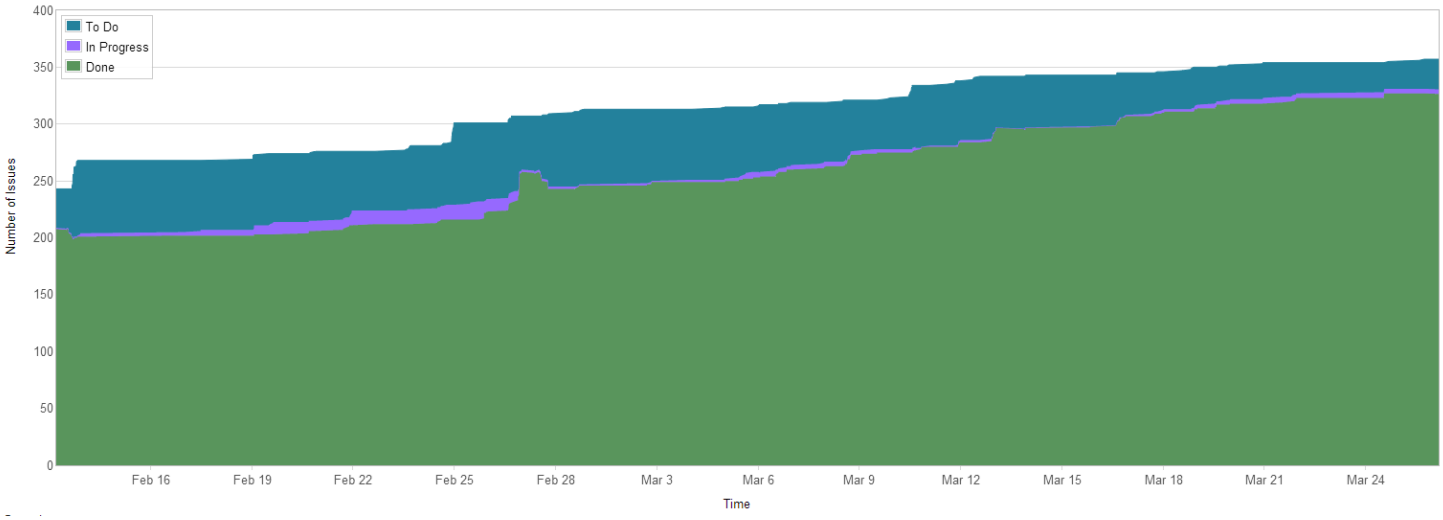

Figure 24-4 Cumulative Flow Diagram

For the first time, the above figure shows how the project is coming to an end. The number of to do tasks are significantly less and we can see how significant progress has been made this sprint.

24.5 Measurement Report

24.5.1 Code Quality Analysis

The following report was generated after analyzing the code. Thirteen methods did not meet code quality goals by exceeding the expected number of lines of code or having none. They have been entered as bugs and will be fixed in a future iteration. Since it is a minor bug, the priority is not high for it which is why they occur repeatedly in the code quality analysis report.

We are NOT planning on fixing this bug as there are many other tasks with higher priorities. The lines of code is a very minor defect and we must focus as a team on completing higher priority issues.

Since our only issues have been with lines of code, we took note for the future that the following scale defines lines of code quality:

0: red
1 – 10: green
11 – 10: yellow
21+: red

Analysis tool used: Code Metrics Viewer
Found at: http://visualstudiogallery.msdn.microsoft.com/9f35524b-a784-4dbc-bd7b-6babd7a5a3b3
Version: 1.5.3
Last updated: 2/5/2012

Scope	Hiearchy	Maintainability Index	Cyclomatic Complexity	Depth of Inheritance	Class Coupling	Lines of Code
ModuleScope	TouchForFood.Tests.dll	70	754	126	2	
NamespaceScope	TouchForFood.Tests	68	666	106	1	
TypeScope	BillControllerTest	64	42	54	1	137
TypeScope	CategoryControllerTest	70	30	33	1	97
TypeScope	CategoryOMTest	71	8	16	1	36
TypeScope	FriendshipControllerTest	64	37	34	1	116
TypeScope	ItemControllerTest	69	27	53	1	90
TypeScope	ItemOMTest	72	7	17	1	32
TypeScope	Menu_CategoryControllerTest	63	56	50	1	134
TypeScope	Menu_ItemControllerTest	60	53	54	1	189
TypeScope	MenuCategoryOMTest	68	9	25	1	46
TypeScope	MenuControllerTest	62	53	47	1	166
TypeScope	MenuItemOMTest	71	9	24	1	43
TypeScope	MenuOMTest	68	9	24	1	46
TypeScope	Order_ItemControllerTest	60	42	38	1	128
TypeScope	OrderControllerTest	80	10	25	1	25
TypeScope	OrderOMTest	70	15	25	1	44
TypeScope	RestaurantControllerTest	64	25	46	1	106
TypeScope	RestaurantUserControllerTest	67	21	40	1	66
TypeScope	Review_Order_ItemControllerTest	72	20	32	1	80
TypeScope	ReviewControllerTest	71	19	35	1	78
TypeScope	ReviewOMTest	76	6	14	1	22
TypeScope	ServiceRequestControllerTest	65	48	36	1	168
TypeScope	SideControllerTest	64	29	52	1	146
TypeScope	TableControllerTest	64	40	49	1	126
TypeScope	TableOMTest	75	8	17	1	25
TypeScope	UserControllerTest	65	35	41	1	156
TypeScope	WaiterOMTest	79	8	17	1	23
MemberScope	DeleteTest() : void	58	1	9		11
MemberScope	MyClassCleanup() : void	100	1	1		0
MemberScope	MyClassInitialize(TestContext) : void	94	1	3		1

MemberScope	MyTestCleanup() : void	76	1	6		4
MemberScope	MyTestInitialize() : void	73	1	6		4
MemberScope	TestContext.get() : TestContext	98	1	1		1
MemberScope	TestContext.set(TestContext) : void	95	1	1		1
MemberScope	WaiterOMTest()	100	1	0		1
NamespaceScope	TouchForFood.Tests.Classes	76	80	54	2	377
TypeScope	ContextMocks	86	14	21	1	27
TypeScope	ContextMocks.FakeSessionState	90	4	3	2	3
TypeScope	Session	62	4	19	1	29
TypeScope	TestDatabaseHelper	64	58	25	1	318
MemberScope	AddBill(order) : bill	64	1	6		9
MemberScope	AddCategory() : category	65	1	4		9
MemberScope	AddFriendship(user, user) : friendship	64	1	6		9
MemberScope	AddItem() : item	66	1	4		8
MemberScope	AddItem(category) : item	64	1	6		9
MemberScope	AddMenu(restaurant) : menu	62	1	5		11
MemberScope	AddMenuCategory(category, menu) : menu_category	63	1	6		10
MemberScope	AddMenuItem(item, menu_category) : menu_item	61	1	7		11
MemberScope	AddOrder(table) : order	63	1	7		10
MemberScope	AddOrder(table, waiter) : order	61	1	8		11
MemberScope	AddOrderItem(order, bill, menu_item) : order_item	61	1	8		11
MemberScope	AddOrderItem(order, menu_item) : order_item	62	1	7		10
MemberScope	AddRestaurant() : restaurant	61	1	4		12
MemberScope	AddRestaurantUser(user, restaurant) : restaurant_user	64	1	7		9
MemberScope	AddReview(restaurant, order, user) : review	60	1	9		12
MemberScope	AddReviewOrderItem(review, order_item, string, int) : re	60	1	8		12
MemberScope	AddServiceRequest(table) : service_request	59	1	7		13
MemberScope	AddSide(menu_category) : side	59	1	7		13
MemberScope	AddTable(restaurant) : table	64	1	6		9
MemberScope	AddUser(string, table, int) : user	55	1	7		18
MemberScope	AddWaiter(restaurant) : waiter	61	1	6		11
MemberScope	RemoveBill(bill) : void	69	2	4		5
MemberScope	RemoveCategory(category) : void	69	2	4		5
MemberScope	RemoveFriendship(friendship) : void	69	2	4		5
MemberScope	RemoveItem(item) : void	69	2	4		5
MemberScope	RemoveMenu(menu) : void	70	2	4		5
MemberScope	RemoveMenuCategory(menu_category) : void	69	2	4		5
MemberScope	RemoveMenuItem(menu_item) : void	69	2	4		5
MemberScope	RemoveOrder(order) : void	70	2	4		5
MemberScope	RemoveOrderItem(order_item) : void	69	2	4		5
MemberScope	RemoveRestaurant(restaurant) : void	69	2	4		5
MemberScope	RemoveRestaurantUser(restaurant_user) : void	69	2	4		5
MemberScope	RemoveReview(review) : void	69	2	4		5
MemberScope	RemoveReviewOrderItem(review_order_item) : void	69	2	4		5
MemberScope	RemoveServiceRequest(service_request) : void	69	2	4		5
MemberScope	RemoveSide(side) : void	69	2	4		5
MemberScope	RemoveTable(table) : void	69	2	4		5
MemberScope	RemoveUser(user) : void	69	2	4		5
MemberScope	RemoveWaiter(waiter) : void	69	2	4		5
MemberScope	TestDatabaseHelper()	100	1	0		1
NamespaceScope	TouchForFood.Tests.Controllers	79	8	22	1	16
TypeScope	HomeControllerTest	79	8	22	1	16
MemberScope	HomeControllerTest()	100	1	0		1
MemberScope	Index() : void	75	3	12		2
MemberScope	MyClassCleanup() : void	80	1	5		3
MemberScope	MyClassInitialize(TestContext) : void	64	1	9		8
MemberScope	TestContext.get() : TestContext	98	1	1		1
MemberScope	TestContext.set(TestContext) : void	95	1	1		1
ModuleScope	TouchForFood.dll	78		228	4	
NamespaceScope	TouchForFood	76	4	10	2	12
TypeScope	MvcApplication	76	4	10	2	12
MemberScope	Application_Start() : void	73	1	4		5
MemberScope	MvcApplication()	100	1	1		1
MemberScope	RegisterGlobalFilters(GlobalFilterCollection) : void	94	1	2		1

MemberScope	RegisterRoutes(RouteCollection) : void	69	1	3		5
NamespaceScope	TouchForFood.Attributes	84	2	5	3	4
TypeScope	AjaxAttribute	84	2	5	3	4
MemberScope	AjaxAttribute(bool)	87	1	1		2
MemberScope	IsValidForRequest(ControllerContext, MethodInfo) : bool	83	1	4		2
NamespaceScope	TouchForFood.Controllers	66	861	155	3	
TypeScope	BillController	65	39	44	3	115
TypeScope	CategoryController	68	35	34	3	72
TypeScope	FriendshipController	69	23	29	3	45
TypeScope	HomeController	74	5	15	3	6
TypeScope	ItemController	70	34	35	3	77
TypeScope	Menu_CategoryController	67	45	33	3	89
TypeScope	Menu_ItemController	68	48	32	3	71
TypeScope	MenuController	66	46	39	3	93
TypeScope	Order_ItemController	64	41	41	3	88
TypeScope	OrderController	58	100	60	3	221
TypeScope	ReportsController	73	17	29	3	27
TypeScope	RestaurantController	71	24	22	3	44
TypeScope	RestaurantUserController	68	17	31	3	30
TypeScope	Review_Order_ItemController	69	29	23	3	56
TypeScope	ReviewController	60	39	45	3	94
TypeScope	SearchController	56	19	28	3	36
TypeScope	ServiceRequestController	58	122	51	3	171
TypeScope	SideController	67	41	32	3	90
TypeScope	TableController	60	73	57	3	115
TypeScope	UserController	61	64	69	3	167
MemberScope	Create() : ActionResult	74	2	9		3
MemberScope	Create(user) : ActionResult	49	7	25		22
MemberScope	Delete(int) : ActionResult	84	1	4		2
MemberScope	DeleteConfirmed(int) : ActionResult	68	3	13		5
MemberScope	Details(int) : ViewResult	84	1	4		2
MemberScope	Dispose(bool) : void	87	1	3		2
MemberScope	Edit(int) : ActionResult	69	3	8		5
MemberScope	Edit(user, HttpPostedFileBase) : ActionResult	47	19	25		20
MemberScope	GetAllReviews(user) : PartialViewResult	64	5	13		7
MemberScope	GetRestaurants() : List<restaurant>	86	1	4		2
MemberScope	HandleDbUpdateException(DbUpdateException) : void	65	3	4		7
MemberScope	Index() : ViewResult	74	2	11		3
MemberScope	LogOff() : ActionResult	76	1	5		4
MemberScope	LogOn() : ViewResult	87	1	2		2
MemberScope	LogOn(string, string) : ActionResult	45	3	25		33
MemberScope	PopularItems() : ViewResult	53	4	21		17
MemberScope	PopularItemsByRating(menu) : ViewResult	53	4	23		17
MemberScope	SuggestItems() : ViewResult	60	2	11		10
MemberScope	UserController()	75	1	4		4
NamespaceScope	TouchForFood.Exceptions	97	4	1	2	4
TypeScope	AssociationExistsException	97	2	1	2	2
TypeScope	ItemActiveException	97	2	1	2	2
MemberScope	ItemActiveException()	100	1	1		1
MemberScope	ItemActiveException(string)	98	1	1		1
NamespaceScope	TouchForFood.Mappers	75	266	70	2	746
TypeScope	BillIM	83	8	7	2	14
TypeScope	BillOM	72	8	15	2	23
TypeScope	CategoryIM	84	6	6	2	10
TypeScope	CategoryOM	70	8	13	2	25
TypeScope	ItemIM	84	6	6	2	10
TypeScope	ItemOM	73	7	12	2	23
TypeScope	MenuCategoryIM	68	8	14	2	30
TypeScope	MenuCategoryOM	65	9	15	2	38
TypeScope	MenuIM	64	12	24	2	48
TypeScope	MenuItemIM	68	8	14	2	30
TypeScope	MenuItemOM	73	7	9	2	24
TypeScope	MenuOM	65	9	15	2	38
TypeScope	Order_ItemIM	83	8	7	2	14

TypeScope	Order_ItemOM	70	5	12	2	20
TypeScope	OrderIM	84	6	6	2	10
TypeScope	OrderOM	65	14	17	2	46
TypeScope	ReportsIM	74	7	13	2	19
TypeScope	RestaurantIM	77	6	14	2	15
TypeScope	RestaurantOM	68	21	25	2	53
TypeScope	Review_Order_ItemIM	82	6	14	2	10
TypeScope	Review_Order_ItemOM	81	5	6	2	11
TypeScope	ReviewIM	84	6	6	2	10
TypeScope	ReviewOM	72	9	15	2	25
TypeScope	ServiceRequestIM	73	9	15	2	23
TypeScope	ServiceRequestOM	73	6	9	2	21
TypeScope	SideIM	73	6	14	2	18
TypeScope	SideOM	67	12	13	2	36
TypeScope	TableIM	84	6	6	2	10
TypeScope	TableOM	75	10	12	2	26
TypeScope	UserIM	84	6	6	2	10
TypeScope	UserOM	69	10	13	2	31
TypeScope	WaiterIM	84	6	6	2	10
TypeScope	WaiterOM	76	6	10	2	15
MemberScope	clearOrder(ICollection<order>) : void	75	2	8		3
MemberScope	Create(waiter) : bool	79	1	5		3
MemberScope	delete(int) : int	67	1	5		7
MemberScope	WaiterOM()	100	1	1		1
MemberScope	WaiterOM(touch_for_foodEntities)	98	1	2		1
NamespaceScope	TouchForFood.Mappers.Abstract	90	5	1	1	8
TypeScope	GenericIM	86	2	1	1	4
TypeScope	GenericOM	94	3	1	1	4
MemberScope	delete(int) : int	100	1	0		0
MemberScope	GenericOM()	86	1	1		2
MemberScope	GenericOM(touch_for_foodEntities)	87	1	1		2
NamespaceScope	TouchForFood.Mappers.Search	68	6	16	2	18
TypeScope	SearchIM	68	6	16	2	18
MemberScope	findByUser(int) : int	55	4	16		16
MemberScope	SearchIM()	100	1	1		1
MemberScope	SearchIM(touch_for_foodEntities)	98	1	2		1
NamespaceScope	TouchForFood.Models	92	514	49	2	623
TypeScope	bill	92	23	7	1	25
TypeScope	category	92	11	4	1	14
TypeScope	CategoryFilterVM	83	9	6	1	16
TypeScope	friendship	93	11	2	1	11
TypeScope	item	92	19	5	1	21
TypeScope	ItemFilterVM	80	18	16	1	34
TypeScope	menu	92	17	5	1	19
TypeScope	menu_category	92	21	6	1	24
TypeScope	menu_item	92	21	6	1	23
TypeScope	MenuMetadata	94	7	4	1	7
TypeScope	order	88	36	15	1	63
TypeScope	order_item	92	29	10	1	31
TypeScope	OrderItemMetadata	100	1	0	1	1
TypeScope	OrderMetadata	95	3	2	1	3
TypeScope	OrderStatusHelper	64	28	7	1	47
TypeScope	OrderStatusHelper.OrderItemStatusEnum	100	0	0	1	0
TypeScope	OrderStatusHelper.OrderStatusEnum	100	0	0	1	0
TypeScope	restaurant	91	27	10	1	33
TypeScope	restaurant_user	93	11	3	1	11
TypeScope	RestaurantMetadata	94	9	1	1	9
TypeScope	RestaurantUserMetadata	94	5	2	1	5
TypeScope	review	92	23	9	1	24
TypeScope	review_order_item	93	17	5	1	17
TypeScope	ReviewMetadata	94	7	3	1	7
TypeScope	ReviewOrderItemMetadata	94	5	3	1	5
TypeScope	service_request	93	15	3	1	16
TypeScope	side	92	19	6	1	21

TypeScope	table	91	16	9	1	22
TypeScope	TableMetadata	100	1	1	1	1
TypeScope	touch_for_foodEntities	92	38	22	2	38
TypeScope	user	91	35	9	1	41
TypeScope	UserMetadata	93	17	5	1	17
TypeScope	waiter	92	15	5	1	17
MemberScope	first_name.get() : string	98	1	0		1
MemberScope	first_name.set(string) : void	95	1	0		1
MemberScope	id.get() : int	98	1	0		1
MemberScope	id.set(int) : void	95	1	0		1
MemberScope	last_name.get() : string	98	1	0		1
MemberScope	last_name.set(string) : void	95	1	0		1
MemberScope	orders.get() : ICollection<order>	98	1	2		1
MemberScope	orders.set(ICollection<order>) : void	95	1	2		1
MemberScope	restaurant.get() : restaurant	98	1	1		1
MemberScope	restaurant.set(restaurant) : void	95	1	1		1
MemberScope	resto_id.get() : int?	98	1	1		1
MemberScope	resto_id.set(int?) : void	95	1	1		1
MemberScope	version.get() : int	98	1	0		1
MemberScope	version.set(int) : void	95	1	0		1
MemberScope	waiter()	81	1	2		3
NamespaceScope	TouchForFood.Util	71	6	1	1	10
TypeScope	StringUtilities	71	6	1	1	10
MemberScope	ExceptBlanks(string) : string	63	5	1		9
MemberScope	StringUtilities()	100	1	0		1
NamespaceScope	TouchForFood.Util.Bill	70	31	10	1	36
TypeScope	BillUtil	70	31	10	1	36
MemberScope	CheckFullyPaid(bill) : bool	65	5	8		7
MemberScope	CheckItemsRemaining(order) : bool	65	6	7		7
MemberScope	CheckProcessing(order) : bool	65	7	7		7
MemberScope	GetTotalAfterTax(bill) : decimal	78	3	3		2
MemberScope	GetTotalBeforeTax(bill) : decimal	85	1	3		2
MemberScope	GetTPS() : decimal	84	1	1		2
MemberScope	GetTVQ() : decimal	84	1	1		2
MemberScope	Update(ref bill) : void	62	7	9		7
NamespaceScope	TouchForFood.Util.Category	55	8	22	1	32
TypeScope	CategoryUtil	55	8	22	1	32
MemberScope	CategoryUtil()	100	1	0		1
MemberScope	filterListByMenu(menu, touch_for_foodEntities) : IList<ca	46	7	22		31
NamespaceScope	TouchForFood.Util.Html	69	16	26	1	39
TypeScope	HtmlDropDownExtensions	68	10	17	1	25
TypeScope	ImageActionLinkHelper	64	1	7	1	8
TypeScope	UrlUtils	75	5	4	1	6
MemberScope	ConvertRelativeUrlToAbsoluteUrl(string) : string	67	4	4		5
MemberScope	UrlUtils()	100	1	0		1
NamespaceScope	TouchForFood.Util.Item	55	15	28	1	47
TypeScope	ItemUtil	55	15	28	1	47
MemberScope	filterListByItem(menu_category, touch_for_foodEntities)	44	9	25		36
MemberScope	getAverageRating(menu_item) : double	60	5	11		10
MemberScope	ItemUtil()	100	1	0		1
NamespaceScope	TouchForFood.Util.Order	82	25	21	1	53
TypeScope	OrderStatusUtil	95	2	1	1	2
TypeScope	OrderUtil	68	23	21	1	51
MemberScope	filterItem(menu_item) : item	77	1	4		3
MemberScope	filterMenuItem(order_item) : menu_item	74	1	5		4
MemberScope	filterSide(order_item) : side	77	1	5		3
MemberScope	filterTable(order) : table	77	1	5		3
MemberScope	filterUser(order) : user	77	1	5		3
MemberScope	filterWaiter(order) : waiter	77	1	5		3
MemberScope	mergeExistingOrderToDb(order) : void	48	9	13		23
MemberScope	OrderUtil()	100	1	0		1
MemberScope	OrderUtil()	94	1	1		1
MemberScope	UpdatePrice(ref order) : void	63	6	11		7
NamespaceScope	TouchForFood.Util.Review	75	24	12	1	56

TypeScope	Rating	68	11	5	1	23
TypeScope	Rating.ReviewRatings	100	0	1	1	0
TypeScope	TextParser	58	13	8	1	33
MemberScope	ParseReviewText(List<review_order_item>, string) : void	45	10	7		30
MemberScope	SetDate(List<review_order_item>) : void	80	2	3		2
MemberScope	TextParser()	100	1	0		1
NamespaceScope	TouchForFood.Util.Search	63	31	34	1	114
TypeScope	SearchService	51	3	18	1	45
TypeScope	SearchUtil	60	23	19	1	63
TypeScope	SearchViewModelHelper	78	5	9	1	6
MemberScope	PopulateSearchViewModel(menu_item) : SearchViewMo	76	1	6		2
MemberScope	PopulateSearchViewModelList(IList<menu_item>) : IList<	76	3	5		3
MemberScope	SearchViewModelHelper()	100	1	0		1
NamespaceScope	TouchForFood.Util.Security	77	21	22	4	78
TypeScope	AES	67	14	11	1	62
TypeScope	CustomAuthorizationAttribute	64	7	10	4	16
TypeScope	SiteRoles	100	0	1	1	0
NamespaceScope	TouchForFood.Util.ServiceRequest	100	1	0	1	1
TypeScope	ServiceRequestUtil	100	1	0	1	1
TypeScope	ServiceRequestUtil.ServiceRequestStatus	100	0	0	1	0
NamespaceScope	TouchForFood.Util.Session	54	5	15	1	33
TypeScope	SessionUtil	54	5	15	1	33
MemberScope	getOpenOrder(user) : order	45	4	15		32
MemberScope	SessionUtil()	100	1	0		1
NamespaceScope	TouchForFood.Util.Side	58	4	19	1	24
TypeScope	SideUtil	58	4	19	1	24
MemberScope	FilterListBySide(menu_category) : IList<side>	50	3	19		23
MemberScope	SideUtil()	100	1	0		1
NamespaceScope	TouchForFood.Util.Table	64	4	10	1	8
TypeScope	TableUtil	64	4	10	1	8
MemberScope	GetRestaurants(HttpRequestBase) : List<restaurant>	64	4	10		8
NamespaceScope	TouchForFood.Util.User	58	39	43	1	114

TypeScope	UserUtil	58	39	43	1	114
MemberScope	BuildMetaScores(user) : Dictionary<string, int[]>	47	12	16		26
MemberScope	CalculateItemScores(Dictionary<string, int[]>, List<menu_	48	7	7		25
MemberScope	ConfidenceValue(int, int) : float	75	1	0		4
MemberScope	getAuthenticatedUser(HttpContextBase) : user	64	2	14		7
MemberScope	getAuthenticatedUser(HttpRequestBase) : user	63	2	8		9
MemberScope	GetSuggestions(user, restaurant) : List<KeyValuePair<mer	45	9	20		32
MemberScope	isUserRole(SiteRoles, HttpContext) : bool	84	1	3		2
MemberScope	SortByScore(Dictionary<menu_item, int>) : List<KeyValuel	67	3	7		6
MemberScope	UserUtil()	100	1	0		1
MemberScope	UserUtil()	84	1	1		2
NamespaceScope	TouchForFood.ViewModels	85	112	26	3	169
TypeScope	MostPopularDishViewModel	92	12	0	1	17
TypeScope	OrderItemVM	69	1	4	1	6
TypeScope	OrderVM	76	24	12	1	34
TypeScope	ReviewVM	89	17	10	1	23
TypeScope	SearchViewModel	91	16	1	1	23
TypeScope	ServiceRequestVM	94	5	3	3	5
TypeScope	SideFilterVM	80	18	17	1	34
TypeScope	WaiterStatsViewModel	91	19	2	3	27
MemberScope	completedOrders.get() : int	98	1	0		1
MemberScope	completedOrders.set(int) : void	95	1	0		1
MemberScope	endDate.get() : DateTime	98	1	1		1
MemberScope	endDate.set(DateTime) : void	95	1	1		1
MemberScope	restoId.get() : int	98	1	0		1
MemberScope	restoId.set(int) : void	95	1	0		1
MemberScope	restoName.get() : string	98	1	0		1
MemberScope	restoName.set(string) : void	95	1	0		1
MemberScope	startDate.get() : DateTime	98	1	1		1
MemberScope	startDate.set(DateTime) : void	95	1	1		1
MemberScope	waiterFirstName.get() : string	98	1	0		1
MemberScope	waiterFirstName.set(string) : void	95	1	0		1
MemberScope	waiterId.get() : int	98	1	0		1
MemberScope	waiterId.set(int) : void	95	1	0		1
MemberScope	waiterLastName.get() : string	98	1	0		1
MemberScope	waiterLastName.set(string) : void	95	1	0		1
MemberScope	WaiterStatsViewModel()	100	1	1		1
MemberScope	WaiterStatsViewModel(DateTime, DateTime)	82	1	2		3
MemberScope	WaiterStatsViewModel(int, string, int, string, string, int)	70	1	1		7

Figure 24-5 Code Quality Report

24.6 Retrospective

In Iteration 10, we finished all the planned defect fixes. We had a few documentation and improvement tasks leftover that we will carry over into Iteration 11.

There were no time consuming defects. There were a lot of little defects and all the changes made a significant difference in the project. Some team members were very busy with other classes and responsibilities but that didn't slow our progress down. We are right on track with almost all coding tasks complete.

24.6.1 Velocity

Story point velocity will not be changed this sprint as there were no more stories planned. All tasks are bugs, tasks, or improvements.

Cumulative velocity (story points): 128 story points
Average velocity (story points): 14.2 story points

Sprint 10 velocity (p-h): 107.92 person-hours
Cumulative velocity (p-h): 1,833.64 person-hours
Average velocity per sprint: 183.36 person-hours

Our average person-hour velocity went down as we did not work many hours this sprint. We expect it to go slightly back up in Iteration 11.

24.6.2 Budget

Total person-hours budgeted to date: 4,968 person-hours
Total person-hours worked to date: 1,833.64 person-hours

Person-hours budgeted per sprint: 414 person-hours
Person-hours worked in Sprint 10:107.92 person-hours

In Iteration 10, we dropped even more under budget. However, this was expected this iteration.

24.6.3 What was learned

The following positive and negative points were gathered from this sprint.

Positives:

- On schedule to finish the project in time
- Communication and meetings are going very well
- Everyone is doing their part and working as a team

Negatives:

- None

25 Iteration 11 Plan

The goal of Iteration 11 is to prepare for the Capstone demos and finish up any few remaining bugs. Most of the work will occur in the first week of the sprint as the demo day for the project is scheduled for the day after the halfway point of the sprint.

25.1 Planned Activities

There were not many planned activities this sprint. The main tasks were:

- Prepare for the demo (buy supplies, get posters ready, etc.)
- Review docs
- Finish up a handful of improvements/bugs
- Prepare for presentation

25.2 Sprint Schedule

Because this is the last sprint and this document must be handed in before our last day, we will explain what the schedule will be over the course of the sprint.

Table 25-1 Sprint Tasklist

Task	Due Date
Complete Documents	March 30, Midnight
Complete Review of Documents	March 31, Midnight
Complete Coding Tasks (Code Freeze)	March 31, Midnight
Complete Testing for Demo	April 2, Midnight
Capstone Demo	April 4, 13:30 – 16:30
Hand in Documents	April 5, Midnight
Capstone Presentation	April 9, 11:30 – 13:00

26 Iteration 11 Report

26.1 Person-Hour Work Log

Start Date: 27/Mar/13 End Date: 9/Apr/13 [Change] (UNREGISTERED)			Christian Daher	Cristian Asenjo	Cynthia Donato	Josh Hum	Katrina Anderson	Matthew Tam	Mikhail Levkovsky	Patrick Modafferi	Ryan Nasr	Total
Issue		Total	3h	8.567h	2.75h	26.917h	10.833h	5.083h	2.933h	13h	12.5h	85.583h
CAP-8	Documentation - Proposal	0.017h	0h	0h	0h	0.017h	0h	0h	0h	0h	0h	0.017h
CAP-9	Documentation - Requirements Document	3.05h	0h	0h	0h	0.017h	3.033h	0h	0h	0h	0h	3.05h
CAP-10	Documentation - Analysis	0.517h	0h	0h	0.5h	0.017h	0h	0h	0h	0h	0h	0.517h
CAP-11	Documentation - Architecture	2.267h	0h	0h	0h	0.017h	0.25h	2h	0h	0h	0h	2.267h
CAP-12	Documentation - User Manual	7.183h	0h	6.683h	0h	0.5h	0h	0h	0h	0h	0h	7.183h
CAP-14	Correspondence	2.683h	0h	0.167h	0h	2.133h	0.383h	0h	0h	0h	0h	2.683h
CAP-18	Meetings	10.1h	0h	0h	0h	0.1h	2h	2h	0h	4h	2h	10.1h
CAP-23	Setup	0.267h	0h	0h	0h	0.017h	0h	0h	0h	0.25h	0h	0.267h
CAP-35	Manage Personal Profile	0.017h	0h	0h	0h	0.017h	0h	0h	0h	0h	0h	0.017h
CAP-36	Restaurant Statistics and Reports	0.033h	0h	0h	0h	0.033h	0h	0h	0h	0h	0h	0.033h
CAP-44	Documentation - Test Plan	0.517h	0h	0h	0h	0.017h	0h	0h	0.5h	0h	0h	0.517h
CAP-57	Documentation - Management	6.017h	0h	1h	0h	4.917h	0h	0h	0.1h	0h	0h	6.017h
CAP-110	Documentation - UIR	3.8h	0h	0.2h	0.25h	0.683h	0.667h	1.083h	0.417h	0h	0.5h	3.8h
CAP-157	Documentation - Traceability	0.517h	0h	0h	0.5h	0.017h	0h	0h	0h	0h	0h	0.517h
CAP-162	Look and Feel	0.267h	0h	0h	0h	0.267h	0h	0h	0h	0h	0h	0.267h
CAP-162	Look and Feel	0.267h	0h	0h	0h	0.267h	0h	0h	0h	0h	0h	0.267h
CAP-164	Documentation - Test Report	2.183h	0h	0h	0.5h	0.683h	0h	0h	0.25h	0.75h	0h	2.183h
CAP-198	Unit Testing Retroactive	3h	0h	0h	0h	3h	0h	0h	0h	0h	0h	3h
CAP-256	Improve user feedback by using ViewBag/Session	0.017h	0h	0h	0h	0.017h	0h	0h	0h	0h	0h	0.017h
CAP-264	User Interface Testing	0.783h	0h	0h	0h	0.033h	0h	0h	0.75h	0h	0h	0.783h
CAP-309	TODOs	0.017h	0h	0h	0h	0.017h	0h	0h	0h	0h	0h	0.017h
CAP-314	Code Review	0.017h	0h	0h	0h	0.017h	0h	0h	0h	0h	0h	0.017h
CAP-459	not enough null checks	1h	0h	0h	0h	0h	0h	0h	0h	0h	1h	1h
CAP-465	Reasearch Normal Form	0.017h	0h	0h	0h	0.017h	0h	0h	0h	0h	0h	0.017h
CAP-567	User Controller DeleteConfirmed Fails If User Is Related To Order or Restaurant_User Entities	0.333h	0h	0h	0.333h	0h	0h	0h	0h	0h	0h	0.333h
CAP-593	Place Order Test Case (TC25.1) Needs to be updated to reflect Toppings and Sides	0.333h	0h	0h	0h	0.333h	0h	0h	0h	0h	0h	0.333h
CAP-597	TC26.2 is wrong and needs to be updated	0.167h	0h	0h	0h	0.167h	0h	0h	0h	0h	0h	0.167h
CAP-600	Cannot create Menu as admin - Update TC	0.25h	0h	0h	0h	0.25h	0h	0h	0h	0h	0h	0.25h
CAP-601	TC27.1 fails as an Admin user	0.083h	0h	0h	0h	0.083h	0h	0h	0h	0h	0h	0.083h
CAP-604	Cancel order removes order from Orders list - Update the Test Case	0.083h	0h	0h	0h	0.083h	0h	0h	0h	0h	0h	0.083h
CAP-612	Customer Manage Bill Test Case fix TC41.1	0.083h	0h	0h	0h	0.083h	0h	0h	0h	0h	0h	0.083h
CAP-613	Restaurant user delete a bill Test Case Update (click garbage)	0.017h	0h	0h	0h	0.017h	0h	0h	0h	0h	0h	0.017h

CAP-622	Presentation	↓	**26.3h**	0h	0h	0h	10.8h	4.5h	0h	0h	7h	4h	**26.3h**
CAP-631	Prepare NFC tags for demo	↑	**0.017h**	0h	0h	0h	0.017h	0h	0h	0h	0h	0h	**0.017h**
CAP-649	Review all version error messages they are not all working properly	↑	**0.333h**	0h	0h	0.333h	0h	0h	0h	0h	0h	0h	**0.333h**
CAP-650	Can't edit bill as ooder user. you get a version error	↑	**0.333h**	0h	0h	0.333h	0h	0h	0h	0h	0h	0h	**0.333h**
CAP-651	Javascript does not seem to work in TFF Android app. Works in	⇑	**1h**	1h	0h	0h	0h	0h	0h	0h	0h	0h	**1h**
CAP-652	cannot create a second bill	↑	**2h**	2h	0h	0h	0h	0h	0h	0h	0h	0h	**2h**
CAP-653	as a user i can't leave a review	↑	**0.35h**	0h	0.35h	0h	0h	0h	0h	0h	0h	0h	**0.35h**
CAP-654	Documentation - Installation Guide	↑	**5.167h**	0h	0.167h	0h	0h	0h	0h	0h	0h	5h	**5.167h**
CAP-655	upload images for profile and food	↑	**2h**	0h	0h	0h	2h	0h	0h	0h	0h	0h	**2h**
CAP-656	put minimum height on page	↑	**0.033h**	0h	0h	0h	0.033h	0h	0h	0h	0h	0h	**0.033h**
CAP-671	Line up search	↑	**0.5h**	0h	0h	0h	0.5h	0h	0h	0h	0h	0h	**0.5h**
CAP-672	Put magnifying glass for search	↑	**1h**	0h	0h	0h	0h	0h	0h	0h	1h	0h	**1h**
CAP-674	test android application	↑	**0.5h**	0h	0h	0h	0h	0h	0h	0.5h	0h	0h	**0.5h**
CAP-675	html entities show up in review	↑	**0.417h**	0h	0h	0h	0h	0h	0h	0.417h	0h	0h	**0.417h**

Figure 26-1 Person-Hour Work Log

With just 85 hours worked, the person-hour effort was the lowest it has been in the last few sprints. However, since there was really not much work left to do on the project, that was just fine. The project was completed successfully.

26.2 Hour Burndown Chart

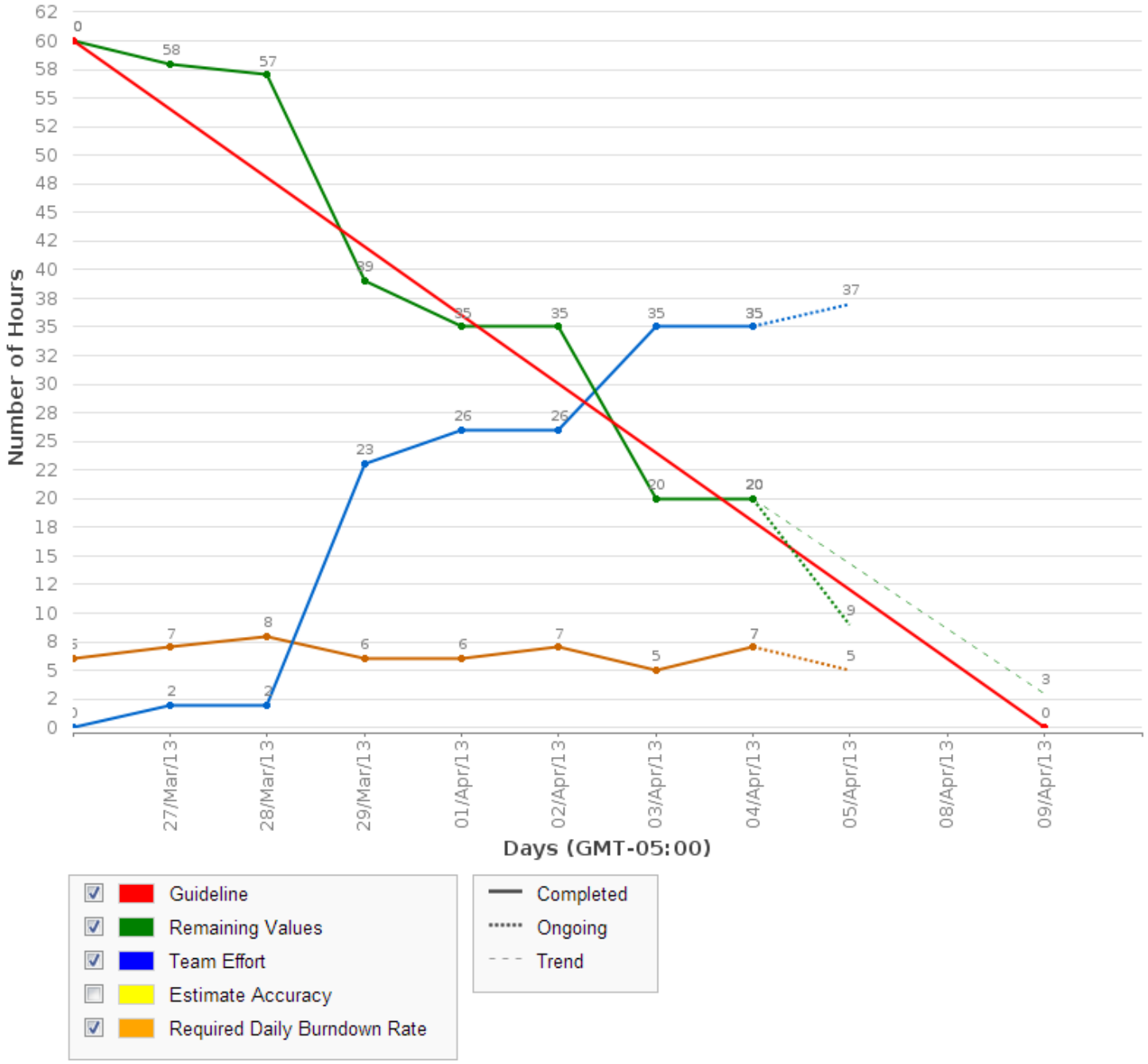

Figure 26-2 Hour Burndown Chart

The above chart is slightly incomplete as the sprint is not yet over. However, as can be seen, we are on track to burn down the rest of the estimated hours in the next few days.

26.3 Issue Burndown Chart

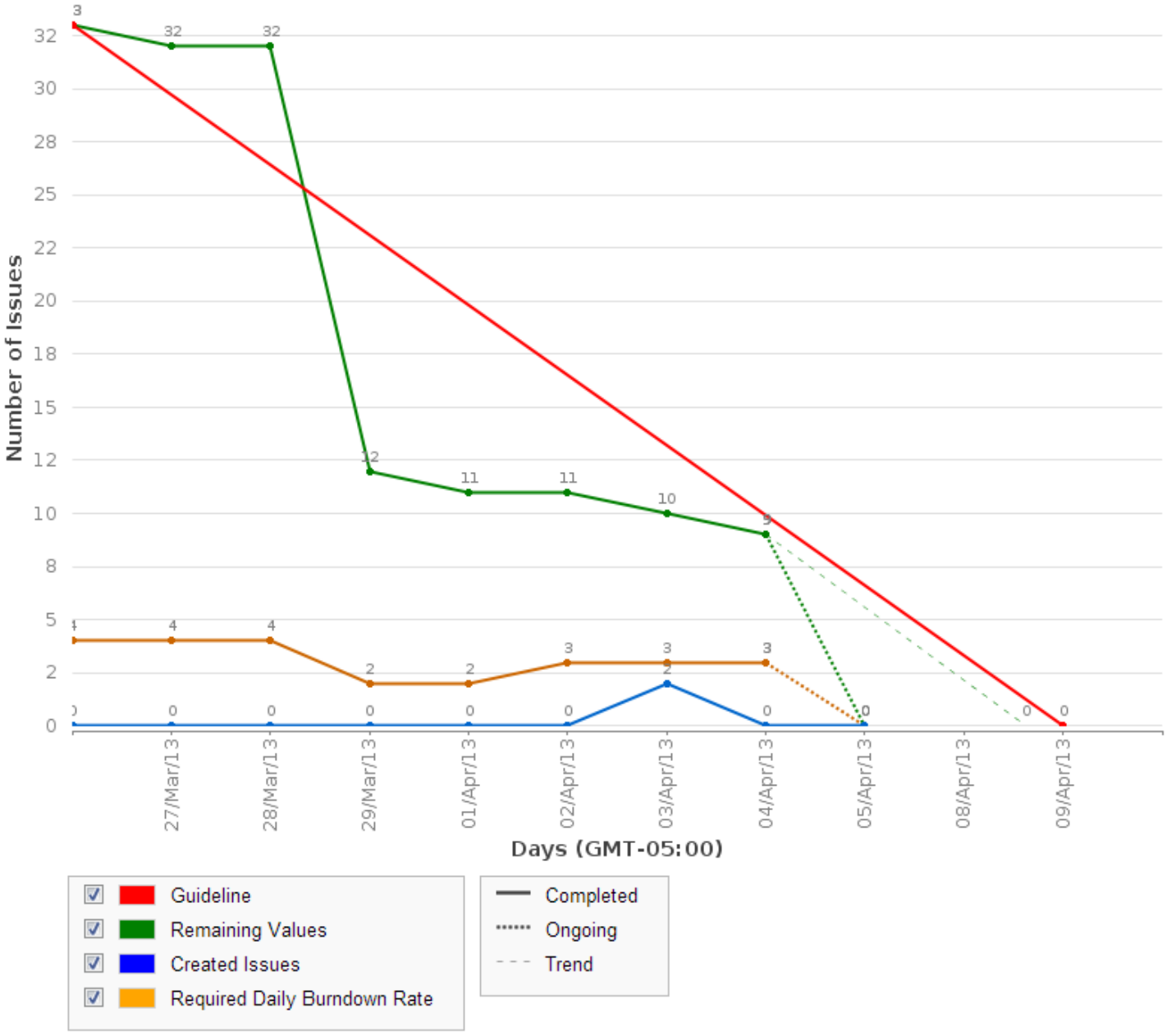

Figure 26-3 Issue Burndown Chart

All issues are completed. There are only 5 remaining to be closed out on Jira.

26.4 Cumulative Flow Diagram

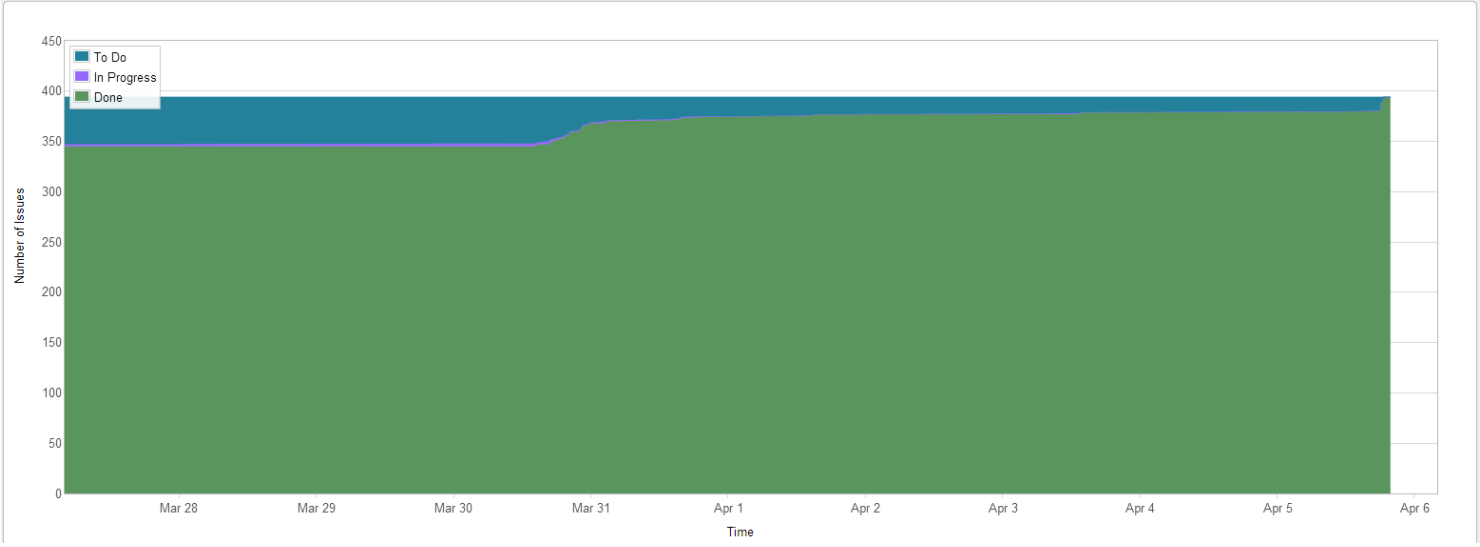

Figure 26-4 Cumulative Flow Diagram

As can be seen in the above diagram, we resolved all the issues in the project. Some issues were not completed as planned (scoped out, duplicated, ignored, etc.). Whatever the reason, all issues were closed at the end of the project.

26.5 Measurement Report

26.5.1 Code Quality Analysis

There were no major code changes during sprint 11. Thus, we did not run a new code quality report. The issue to resolve too many or too little lines of code was ignored because there was no time to complete it as other issues had higher priority.

26.6 Retrospective

In sprint 11, the team prepared for the demo and finished up outstanding documentation tasks. Unfortunately, we were not able to complete all the suggested tasks from previous submission feedback for the documents because we ran out of time and team members had to commit time to other classes. However, we completed all the doc tasks that we planned.

As for the demo, everything went smoothly and the demo was a great success. We were well prepared and did not have to scramble for anything last minute. Our posters were printed a couple weeks in advance and we arrived early to setup. We received a lot of positive feedback as well as a few suggestions from the users who came around and tried out our app. We simulated a real restaurant scenario where we allowed users to try TFF on our phones. We had a screen showing them what was going on from the restaurant point of view as they accepted orders. After ordering, the food was brought to them and their bill was closed out (this was a technicality because the food was free).

Some users had never or hardly used smartphones before. We got good feedback from them and a few suggestions on how to make the interface more intuitive.

26.6.1 Velocity

Story point velocity was not changed this sprint as there were no more stories planned. Our person-hour velocity was not very high due to team commitments to other classes. There were also less tasks to be completed this sprint.

Cumulative velocity (story points): 128 story points
Average velocity (story points): 14.2 story points

Sprint 11 velocity (p-h): 85.58 person-hours
Cumulative velocity (p-h): 2,134.89 person-hours
Average velocity per sprint: 177.91 person-hours

Our average person-hour velocity went down again. The total average velocity over the project including sprint 0 (setup and preparation) is 177.91 person-hours.

26.6.2 Budget

Total person-hours budgeted to date: 4,968 person-hours
Total person-hours worked to date: 2,134.89 person-hours

Person-hours budgeted per sprint: 414 person-hours
Person-hours worked in Sprint 11: 85.58 person-hours

Overall, we finished at around 50% of the budgeted person-hours. We hoped to be much closer to the estimated budget. However, we overestimated how long it would take to complete the project as well as how much time each team member would be able to commit to the project.

26.6.3 What was learned

The following positive and negative points were gathered from this sprint.

Positives:

- On schedule to finish the project in time
- Communication and meetings are going very well
- Everyone is doing their part and working as a team
- Very organized for the demo

Negatives:

None

27 Final Project Report

27.1 Summary

Overall this Capstone project was a great learning experience for all of us. We learned a lot. There was something new for all of us to learn whether it was working with a new coding language, framework, development method, management system, or documentation. The team worked really well together and we all pulled together to complete the project. Each team member had their strengths and they used them to the best of their abilities to help the team.

As the team became more organized, we created developer roles for each sprint. Team members would rotate to assume these roles. Some of the roles that were used throughout development included QA, Bug Basher, and Flow Fixer. Everyone took a turn to assume at least one of these three mentioned roles.

The project took a considerable effort in terms of time commitment and determination but we are very satisfied with how the project turned out.

27.2 Stakeholder

In the Fall, we had approached Dr. Ormandjieva to be our stakeholder. She accepted and I met with her a couple times. We discussed finding a restaurant stakeholder which she attempted to help us with. Unfortunately, we were unable to find a restaurant stakeholder. Since it was difficult to arrange meetings with Dr. Ormandjieva and we could not find a restaurant stakeholder, we built our needs and features ourselves from both the restaurant and customer point of view.

In the Winter semester, Dr. Ormandjieva informed us that she would not have the time to commit to being our stakeholder. Instead, Dr. Grogono accepted our request to be our stakeholder. We met once informally to talk about the project and then again with the whole group to present and demo the current progress. He did not have much feedback and seemed satisfied with the project. We wanted to meet once more before the end of the semester but did not have a chance. He attended our project demo and will come to our presentation as well.

27.3 Risk Management

Throughout the course of the project, we had to manage risk to make sure we finished the project in time. We realized at the end of December that we would probably not be able to implement the social networking aspect of the project as was originally planned. Thus, after speaking with Dr. Constantinides, we scoped out the social networking aspect of the project.

Sometimes people went on vacation, got sick, or had time commitments in other classes. These were all factors in planning each sprint and managing risk. We used an activity-on-node diagram to ensure that we were on track to finish in time. One of the things we did to manage our time was to aim to complete all the stories of the project by the end of February which would give us time to eliminate defects in the code and give us a buffer zone in case we didn’t meet that deadline. We ended finishing the stories by the second week of March which we were satisfied with.

Although not always shown in this document, rough outlines for the next few sprints till the end of the project were set prioritizing tasks in the order that they should be worked on. These were discussed in sprint planning meetings. You can see the meeting slides in /trunk/docs/Management/meeting presentations/ folder. These outlines began in the January 29.pptx slide set as we needed to manage how close we were to finishing as we were already halfway through the project.

27.4 Budget

The budget is part of risk management and has been reported on in this document in each sprint. We were very far behind in working the hours budgeted. However, the problem was that our original estimate was way too high. We couldn’t change that estimate but we continued to monitor our working hours. We set a goal to work around 200 person-hours per sprint and the timing worked out very well for us.

27.5 Conclusion

I am very proud of my team and all the hard work they put into this Capstone project. Everyone worked together to make the project a success. Although there were occasional disagreements, there were no major problems and we made it through each issue very professionally. Personally, as the leader, I can say that there were times that were stressful and taxing but it was great to work with each person in the team and it was a great overall experience.

Appendix A References

[1] Microsoft. (2012, December) Code Metrics Viewer. [Online]. http://visualstudiogallery.msdn.microsoft.com/9f35524b-a784-4dbc-bd7b-6babd7a5a3b3

[2] Atlassian. (2013, January) JIRA; Issue & Project Tracking Software. [Online]. http://www.atlassian.com/software/jira/overview

Appendix B Glossary

Term	Definition
Bug Basher	A developer role. At the beginning of each sprint, all defects in the backlog are assigned to the bug basher. It is their responsibility to fix as many defects as possible according to priority. Defects can relate to any aspect of the project but are usually code defects.
QA	A developer role. The QA person writes new UI tests for the sprint. They execute the tests during the last two days of the sprint. They are also in charge of running unit tests and writing the test report. Once usability tests start, they will also be in charge of executing each usability test.
Flow Fixer	A developer role. There are often issues where the flow between two parts of the system is not correct or multiple parts work in singularity do not work together. The flow fixer's job is to ensure that all parts of the TFF system work together as designed. Flow fixing is always a coding task.

For all other terms and definitions, please refer to the SRS document - Appendix B Glossary and Appendix C Acronyms.