iBudget

Software Project Management Plan

Approvals:	L. Assayah	LA	date: 2/27/12
	C. Leung	CL	date: 2/27/12
	Q. Pham	QCP	date: 2/27/12
	V. Velev	VV	date: 2/27/12
	J. Reimels	JR	date: 2/27/12
	V. Dineva	VD	date: 2/27/12

Revision History

Date	Author	Version	Reason
2/12/12	V.Velev	1.0	First Draft
2/12/12	Q.Pham	1.0	Draft Schedule
2/12/12	V.Velev	1.1	Added Budget and Resource Allocation section.
2/12/12	V.Dineva	1.1	Risk Management Contribution
2/12/12	Q.Pham	1.1	Risk Management Contribution
2/13/12	Q.Pham	1.2	Add TOC and Reference Section. Modify Schedule.
2/26/12	Q.Pham	1.3	Update Schedule
2/26/12	V.Velev	1.4	Updated Process Model

Table of Contents

1. Introduction

1.1. Project overview

iBudget is planned as a personal financial management software tool that produces as an output a thorough and simple analysis of one's financial statements. The user should be able to use iBudget to easily and efficiently attain an overview of his or her finances as well as create a budget.

1.2. Project deliverables

At the end of the school semester, the team shall deliver a working software program as well as all the necessary documentation.

1.3. Evolution of the SPMP

This document shall be maintained on a weekly basis by the project leader. It is subject to configuration management by means of the SCMP.

1.4. Project deliverables

Source code for the *iBudget* project and all necessary documentation.

1.5. Acronyms

QA – quality assurance
SCMP – Software Configuration Management Plan
SPMP – Software Project Management Plan
SRS – Software Requirements Specification
SDD – Software Design Document
STP – Software Test Plan

2. Project Organization

2.1. Process Model

The project will be executed using the Unified Process. This iterative and incremental approach will consist of the following phases: Inception, Elaboration, Construction, and Transition. Over the course of the project different emphasis will be put on the process disciplines. For example, emphasis on design will be highest at the beginning of the project and decrease as the project progresses.

2.2. Organizational Structure

The team will consist of members with designated roles. The roles are team leader, the configuration management leader, the quality assurance leader, the requirements management leader, the design leader, and the implementation leader. These roles are shown in figure 2.2.

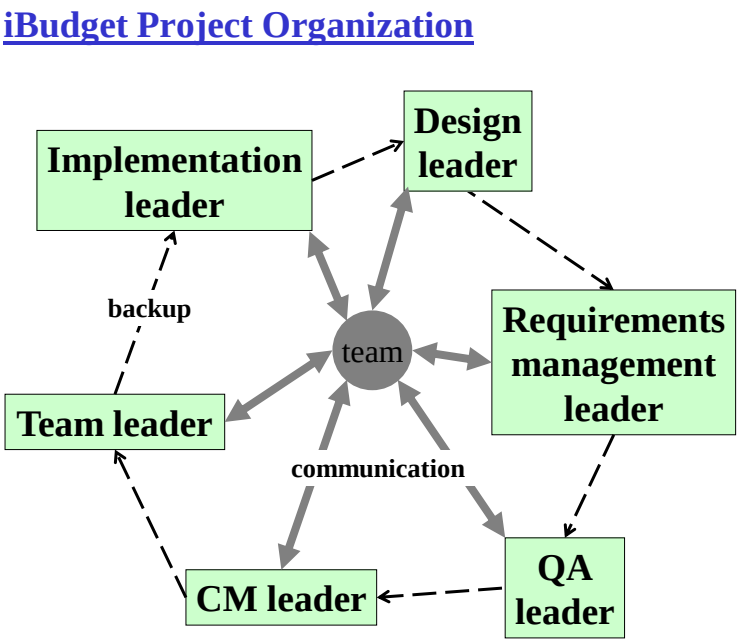

Figure 2.2: iBudget Project Organization

2.3. Project Responsibilities

The responsibilities of the participants in the project are shown in table 2.3.

Member	Team Leader	CM Leader	QA Leader	Requirements Management Leader	Design Leader	Implementation Leader
Document Responsibility	SPMP	SCMP	SQAP STP	SRS	SDD	Code Base

Table 2.3: Project Responsibilities

Being responsible for the document includes the following:

- Making sure the document is completed in time.
- Keeping the document up-to-date throughout the project life cycle

3. Management Process

3.1. Management objectives and priorities

The highest management priority shall be the completion of the project on time before the end of the school semester. This also includes that the project be on schedule. The second priority shall be meeting a degree of quality as specified by the quality parameters. The third priority shall be to include as many features as were planned throughout the project lifecycle.

3.2. Assumptions, Dependencies and Constraints

None.

3.3. Risk management

#	Risk Title	Likelihood to Occur 1-10	Impact 1-10	Retirement Cost 1-10	Priority	Retirement or Mitigation Plan	Responsible Engineer	Target Completion Date
1	Requirements Inflation	3	8	2	48	Add additional features depending on time.	All	04/02
2	Poor Productivity	7	10	10	40	Keep developers engaged and motivated.	Project Leader	04/30
3	Shortfalls in 3rd party software	4	3	6	336	Analyze software capabilities in advance.	TBD	02/30
4	Insufficient time	8	9	5	30	Follow schedule.	Project Leader	04/30
5	Unforeseen software defects	7	7	5	45	Allocate enough time for defect resolution.	All	04/16

Risk #1: Overpromising and under delivering on specified requirements will result in the failure of the project. The team has decided on a basic feature set and will add additional functionality depending on time left in the semester.

Risk#2: Time constraints and other personal/professional responsibilities can result in poor productivity. The team leader is responsible for keeping other team members engaged.

Risk#3: Functionality of 3rd party tools might not suffice. The team should analyze each tool in advance.

Risk#4: The team is required to finish the project by the end of the school semester. The team leader is responsible for making sure that all deadlines are met.

Risk#5: Software defects are inevitable. All developers should allocate enough time to debug their code and resolve any uncovered defects.

3.4. Monitoring and controlling mechanism

There will be a weekly project meeting on Mondays from 5PM to 6PM. The team will discuss all issues pertaining to the project at that time. Further discussion will take place via email or telephone. In case the need arises for addition meetings, the team will schedule one or more on a week-by-week basis depending on the availability of all team members.

Each team member's progress will be tracked by the team leader. Each member is responsible to submitting a weekly report to the team leader, indicating the tasks he or she has worked on throughout the week as well as the tasks planned for the following week. The team member must also specify an estimated time needed to complete planned tasks as well as the actual time spent on already completed tasks.

3.5. Staffing Plan

Team member roles are specified as in Table 3.5.

Name	Team Leader	CM Leader	QA Leader	Requirement Management Leader	Design Leader	Implementation Leader
Vladimir Velev	X					
Jonathan Reimels		X				
Vanya Dineva			X			
Laurene Assayah				X		

Churk Leung					X	
Quan Pham						X

Table 3.5: iBudget Staffing Plan

4. Technical Process

The SRS will describe the technical process in detail. This section describes aspect of the process which is not explicitly stated in the SRS.

4.1. Methods, tools and techniques

The iBudget project will use Apache server, MySQL, and will be implemented in PHP.

4.2. Software Documentation

Refer to the SQAP.

5. Resources and schedule

5.1. Budget and resource allocation

Estimate before beginning requirement analysis.

Estimate LOC:

FP = (∑UFP_i)*GCF
UFP_i = 3 (EI) + 4 (EO) + 3 (EIN) + 7 (ILF) + 5 (ELF) = 22
GCF = 0.65 + 0.01 * 42 = 1.07
FP = 23.54

LOC = 23.54 * (600 LOC/FP) = 14 KLOC

COCOMOI:

Effort Applied (E) = 3.0 * (14)^1.12 = 57 man-months
Development Time (D) = 2.5 * (57)^0.38 = 11 months
People required (P) = 57/11 = 6

5.2. Schedule

ID	Task Name	Start	Finish	Resource Names
1	**Scope**	**Mon 1/30/12**	**Mon 2/6/12**	
2	Define Project Scope	Mon 1/30/12	Mon 1/30/12	Churk Y. Leung
3	Define Development Tools and Hosting Environment	Mon 1/30/12	Mon 1/30/12	The iTeam
4	Define Roles and Responsibility	Mon 2/6/12	Mon 2/6/12	Vladimir Velev
5	Define Deliverables	Mon 2/6/12	Mon 2/6/12	Vladimir Velev
6	Scope Complete	Mon 2/6/12	Mon 2/6/12	
7	**Analysis/Software Requirements**	**Mon 2/13/12**	**Mon 2/27/12**	
8	Team Discussion and Software Requirements Gathering	Mon 2/20/12	Mon 2/27/12	The iTeam
9	Produce SRS with Ucase and Mockups	Mon 2/20/12	Mon 2/27/12	Vladimir Velev,Laurene
10	Team Review software specifications	Sun 2/26/12	Fri 3/16/12	The iTeam
11	Incorporate Changes into Schedule	Mon 2/13/12	Fri 3/23/12	Quan Pham
12	Develop delivery timeline	Mon 2/20/12	Fri 3/30/12	Vladimir Velev
13	Analysis complete	Mon 2/27/12	Mon 2/27/12	
14	**Design**	**Mon 2/13/12**	**Mon 3/19/12**	
15	Review preliminary software specifications	Mon 2/13/12	Mon 3/5/12	The iTeam
16	Develop functional specifications - front end	Tue 2/14/12	Mon 3/5/12	Vanya Dineva,Qua
17	Develop functional specifications - back end	Sat 2/18/12	Mon 3/19/12	Churk Y. Leung
18	Review functional specifications	Mon 2/27/12	Mon 3/26/12	The iTeam
19	Design complete	Mon 3/26/12	Mon 3/26/12	
20	**Development**	**Mon 2/27/12**	**Mon 4/2/12**	
21	Identify modular/tiered design parameters	Mon 2/27/12	Mon 3/5/12	Churk Y. Leung
22	Develop code	Mon 2/27/12	Mon 3/5/12	The iTeam
23	Developer testing (primary debugging)	Mon 3/5/12	Tue 4/10/12	The iTeam
24	Development complete	Mon 4/2/12	Mon 4/2/12	
25	**Testing**	**Mon 4/2/12**	**Mon 4/16/12**	
26	Develop test plans	Mon 4/2/12	Mon 4/9/12	Vanya Dineva,Lau
27	Review test plans	Mon 4/9/12	Mon 4/16/12	The iTeam
28	Implement Test Plans	Mon 4/9/12	Mon 4/16/12	Vanya Dineva,Lau
29	Unit testing complete	Mon 4/16/12	Mon 4/16/12	
30	**Documentation**	**Mon 2/6/12**	**Mon 4/16/12**	
31	Project Proposal	Mon 2/6/12	Mon 4/9/12	Churk Y. Leung
32	SPMP Document	Mon 2/6/12	Mon 4/9/12	Quan Pham,Vladir
33	SQAP Document	Mon 2/6/12	Mon 4/9/12	Laurene Assayah,
34	SCMP Document	Mon 2/6/12	Mon 4/9/12	Jonathan Reimels
35	Develop Project Schedule	Mon 2/6/12	Tue 4/17/12	Quan Pham,Vladir
36	Develop user manuals	Mon 2/6/12	Mon 4/23/12	The iTeam
37	Review all user documentation	Mon 4/9/12	Mon 4/30/12	The iTeam
38	Incorporate user documentation feedback	Mon 4/9/12	Mon 4/30/12	The iTeam
39	Documentation complete	Mon 4/30/12	Mon 4/30/12	
40	**Final Presentation**	**Mon 4/23/12**	**Mon 4/30/12**	
41	Determine final presentation strategy	Mon 4/23/12	Mon 4/30/12	The iTeam
42	Prepare Powerpoint Slide	Mon 4/23/12	Mon 4/30/12	Vladimir Velev
43	Deployment complete	Mon 4/30/12	Mon 4/30/12	
44	**Post Presentation Review**	**Mon 4/30/12**	**Mon 4/30/12**	
45	Document lessons learned	Mon 4/30/12	Mon 4/30/12	The iTeam
46	Create software maintenance team	Mon 4/30/12	Mon 4/30/12	The iTeam
47	Software development template complete	Mon 4/30/12	Mon 4/30/12	
48	**Deliverables**	**Mon 1/30/12**	**Mon 4/30/12**	
49	Week 1 Team Report	Mon 1/30/12	Mon 1/30/12	Vladimir Velev
50	Week 2 Team Report	Mon 2/6/12	Mon 2/6/12	Vladimir Velev
51	Week 3 Team Report	Mon 2/13/12	Mon 2/13/12	Vladimir Velev
52	Week 4 Team Report	Mon 2/20/12	Mon 2/20/12	Vladimir Velev
53	Week 5 Team Report	Mon 2/27/12	Mon 2/27/12	Vladimir Velev
54	Week 6 Team Report	Mon 3/5/12	Mon 3/5/12	Vladimir Velev
55	Week 7 Team Report	Mon 3/19/12	Mon 3/19/12	Vladimir Velev
56	Week 8 Team Report	Mon 3/26/12	Mon 3/26/12	Vladimir Velev
57	Week 9 Team Report	Mon 4/2/12	Mon 4/2/12	Vladimir Velev
58	Week 10 Team Report	Mon 4/9/12	Mon 4/9/12	Vladimir Velev
59	Week 11 Team Report	Mon 4/16/12	Mon 4/16/12	Vladimir Velev
60	Week 12 Team Report	Mon 4/23/12	Mon 4/23/12	Vladimir Velev
61	Week 13 Team Report	Mon 4/30/12	Mon 4/30/12	Vladimir Velev
62	PHASE I	Mon 2/13/12	Mon 2/13/12	The iTeam
63	PHASE II	Mon 2/27/12	Mon 2/27/12	The iTeam
64	PHASE III	Mon 3/26/12	Mon 3/26/12	The iTeam
65	FINAL PRESENTATION	Mon 4/30/12	Mon 4/30/12	The iTeam
66	Delivarables complete	Mon 4/30/12	Mon 4/30/12	

6. References

1. Software Engineering: Modern Approaches / Eric J. Braude, Michael E. Bernstein, 2nd Ed. Wiley, 2011.