

Life Cycle Plan (LCP)

Pic Share App (RA)

Team 1

Logan Chang – Project Manager, Developer

Jiashun Zhou – Developer

Yew Hua Chua – Developer

Haihao Lu – Developer

Yunwei Yuan – Developer

Yao Wang – Developer

Weili Zhang – Developer

Joseph Mouawad – Developer

Nicholas Funk – Developer, Tester

27 April 2016

Version History

Date	Author	Version	Changes made	Rationale
02/22/16	Logan Chang, Yao Wang	1.0	Added Sections 1-6 for RDC Package	LCP was first started during the RDC Phase.
02/23/16	Nick Funk	1.1	Updated Team Experience	TBD in previous version
04/27/16	Logan Chang	1.2	Updated Section 6	Added CCD Results.

Table of Contents

Version History	ii
Table of Contents	iii
Table of Tables	v
Table of Figures.....	vi
1. Introduction	1
1.1 Purpose of the LCP	1
1.2 Status of the LCP	1
1.3 Assumptions.....	1
2. Milestones and Products.....	2
2.1 Overall Strategy	2
2.2 Project Deliverables	3
3. Responsibilities	5
3.1 Project-specific stakeholder’s responsibilities.....	5
3.2 Responsibilities by Phase.....	5
3.3 Skills	7
4. Approach.....	8
4.1 Monitoring and Control	8
4.2 Methods, Tools and Facilities.....	8
5. Resources	9
6. Iteration Plan.....	9
6.1 Plan.....	9
6.1.1 Capabilities to be implemented	9
6.1.2 Capabilities to be tested.....	10
6.1.3 Capabilities not to be tested	10
6.1.4 CCD Preparation Plans.....	10
6.2 Iteration Assessment.....	11
6.2.1 CCD Capabilities Implemented, Tested, and Results	11
6.2.2 Core Capabilities Drive-Through Results	12
6.2.3 Re-CCD Capabilities Implemented, Tested, and Results	12

6.2.4 Retry Core Capabilities Drive-Through Results	13
6.3 Adherence to Plan	14
6.3.1 CCD Adherence	14
6.3.2 ReCCD Adherence.....	14

Table of Tables

Table 1: Artifacts Deliverables in Re-Baselined Foundations Phase3
Table 2: Artifacts Deliverables in Development Phase.....3
Table 3: Artifact Deliverables in Transition Readiness Phase.....4

Table of Figures

No table of figures entries found.

1. Introduction

1.1 Purpose of the LCP

The purpose of the LCP is to provide an overview of the overall plan for the project's development as well as serve as a record of the progress over the life cycle of the project.

1.2 Status of the LCP

The status of the LCP is currently at the Rebaselined Development Commitment Package version 1.0. The LCP was first created during the second semester of the class due to using the Resilient Agile during the first semester.

1.3 Assumptions

- The duration of the project is 24 weeks with 12 weeks in Fall 2015 and 12 weeks in Spring 2016.
- There are 9 team members, 7 on-campus and 2 DEN students. Two new students have replaced two students from Fall 2015 in Spring 2016.
- No additional staff turnover will occur.

2. Milestones and Products

2.1 Overall Strategy

Initially, the overall strategy for our project was to follow the Resilient Agile process, as that was the process assigned to us during the first semester of the class. Through the process, we have designed and developed Robustness Diagrams, Use Case Scenarios, Sequence Diagrams, and a Class Diagram based on a core set of product requirement and use cases defined by and agreed upon by the client. During the second semester, we are using an agile approach to develop the mobile app and backend for the project. The following phases only cover the second semester.

Re-Baselined Foundations Phase

Duration: 1/19/16 to 2/12/16

Concept: During the rebaseline phase, the team acquired two new team members that were brought up to speed for the product. The team continued development of the main product based upon the prototype developed during the previous semester. The team also adjusted priorities based on the news that the backend solution Parse was shutting down to account for developing a custom backend server based on the Open Source Parse Server code.

Deliverables: Rebaseline Development Commitment Package

Milestone: Rebaseline Development Commitment Review

Strategy: One Incremental Commitment Cycle

Development Phase

Duration: 2/13/16 to 3/25/16

Concept: During the development phase, the team focused on developing all the core MVP features of the app as well as a custom Parse backend server solution.

Deliverables: Core Capability Report

Milestone: Core Capability Drivethrough

Strategy: One Incremental Commitment Cycle

Transition Readiness Phase

Duration: 3/26/16 to 4/15/16

Concept: During the transition phase, the team finished any outstanding issues or bugs discovered during the CCD. Also the team prepared the product to be transitioned to the client.

Deliverables: Transition Readiness Package

Milestone: Transition Readiness Review

Strategy: One Incremental Commitment Cycle

2.2 Project Deliverables

2.2.1 Re-Baselined Foundations Phase

Table 1: Artifacts Deliverables in Re-Baselined Foundations Phase

Artifact	Due date	Format	Medium
Rebaseline Development Commitment Package <ul style="list-style-type: none"> • Life Cycle Plan (LCP) • Transition Plan (TP) • Test Plan and Cases (TPC) 	2/22/2016	.doc, .pdf	Soft copy
Project Plan	Every 2 Weeks on Wednesday	.mpp	Soft copy
Bi-weekly Progress Report	Every 2 Weeks on Wednesday	.xls	Soft copy
Bi-weekly Risk & Defects	Every 2 Weeks on Wednesday	.xls	Soft copy
Jira Tickets Work Log	Every Monday	Tickets	Jira

2.2.2 Development Phase

Table 2: Artifacts Deliverables in Development Phase

Artifact	Due date	Format	Medium
Core Capability Report	3/25/2016	.doc, .pdf	Soft copy
Project Plan	Every 2 Weeks on Wednesday	.mpp	Soft copy
Bi-weekly Progress Report	Every 2 Weeks on Wednesday	.xls	Soft copy
Bi-weekly Risk & Defects	Every 2 Weeks on Wednesday	.xls	Soft copy
Jira Tickets Work Log	Every Monday	Tickets	Jira

2.2.3 Transition Readiness Phase

Table 3: Artifact Deliverables in Transition Readiness Phase

Artifact	Due date	Format	Medium
Transition Readiness Package <ul style="list-style-type: none">• RDC Package• Test Procedure and Result (TPR)• Support Plan (SP)• Regression Test Package (RTP)• User Manual (UM)• Training Materials (TM)	4/15/2016	.doc, .pdf	Soft copy
Final Project Code	4/15/2016	Web Link	Github
Project Plan	Every 2 Weeks on Wednesday	.mpp	Soft copy
Bi-weekly Progress Report	Every 2 Weeks on Wednesday	.xls	Soft copy
Bi-weekly Risk & Defects	Every 2 Weeks on Wednesday	.xls	Soft copy
Jira Tickets Work Log	Every Monday	Tickets	Jira

3. Responsibilities

3.1 Project-specific stakeholder's responsibilities

No additional project-specific stakeholders outside of the client and developers.

3.2 Responsibilities by Phase

The following table is a template for stakeholder's responsibilities in each phase.

Table 4: Stakeholder's Responsibilities in each phase

Team Member / Role	Primary / Secondary Responsibility		
	Re-Baselined Foundations	Development-Construction Iteration	Development-Transition Iteration
Name: Logan Chang <i>Product Manager, iOS Developer</i>	Primary Responsibility - Plan weekly sprints - Update biweekly progress report Secondary Responsibility - Develop iOS App - Review and merge iOS App code - Test iOS App	Primary Responsibility - Plan weekly sprints - Update biweekly progress report Secondary Responsibility - Develop iOS App - Review and merge iOS App code - Test iOS App	Primary Responsibility - Plan weekly sprints - Update biweekly progress report Secondary Responsibility - Develop iOS App - Review and merge iOS App code - Test iOS App
Name: Yew Hua Chua <i>Backend Developer</i>	Primary Responsibility - Develop backend solution and admin tool. Secondary Responsibility - Parse Alternatives Research	Primary Responsibility - Develop backend solution and admin tool. Secondary Responsibility - Test backend solution and admin tool.	Primary Responsibility - Develop backend solution and admin tool. Secondary Responsibility - Test backend solution and admin tool.
Name: Haihao Lu <i>Backend Developer</i>	Primary Responsibility - Develop backend solution and admin tool. Secondary Responsibility - Parse Open Source Server Research	Primary Responsibility - Develop backend solution and admin tool. Secondary Responsibility - Test backend solution and admin tool	Primary Responsibility - Develop backend solution and admin tool. Secondary Responsibility - Test backend solution and admin tool.
Name: Jiashun Zhou <i>iOS Developer</i>	Primary Responsibility - Develop iOS App Secondary Responsibility - Class Diagram Architecture	Primary Responsibility - Develop iOS App Secondary Responsibility - Class Diagram Architecture	Primary Responsibility - Develop iOS App Secondary Responsibility - Class Diagram Architecture
Name: Weili Zhang <i>Backend developer, iOS Developer</i>	Primary Responsibility - Learn iOS Development - Learn backend development Secondary Responsibility - Program Model	Primary Responsibility - Develop backend solution and admin tool. - Test backend solution and admin tool. Secondary Responsibility - Develop iOS App.	Primary Responsibility - Develop backend solution and admin tool. - Test backend solution and admin tool. Secondary Responsibility - Develop iOS App.

Name: Yao Wang <i>iOS Developer</i>	Primary Responsibility - Develop iOS App Secondary Responsibility - Life Cycle Plan, Team Roles and Responsibilities	Primary Responsibility - Develop iOS App Secondary Responsibility - Test iOS App	Primary Responsibility - Develop iOS App Secondary Responsibility - Test iOS App
Name: Yunwei Yuan <i>iOS Developer</i>	Primary Responsibility - Develop iOS App Secondary Responsibility - Risk Analysis, Traceability Matrix, 2 Metrics	Primary Responsibility - Develop iOS App Secondary Responsibility - Test iOS App	Primary Responsibility - Develop iOS App Secondary Responsibility - Test iOS App
Name: Nick Funk <i>Backend Developer, iOS Developer, QFP</i>	Primary Responsibility - Learn iOS development - Learn backend development Secondary Responsibility - Create Test Plan and Cases - Create Transition Plan	Primary Responsibility - Develop backend solution and admin tool. Secondary Responsibility - Develop iOS App - Test iOS App	Primary Responsibility - Develop backend solution and admin tool. Secondary Responsibility - Develop iOS App - Test iOS App
Name: Joseph Mouawad <i>Backend Developer, iOS Developer</i>	Primary Responsibility - Learn iOS Development Secondary Responsibility - Definition of Done - Technical Debt	Primary Responsibility - Learn backend development - Develop backend solution and admin tool. Secondary Responsibility - Develop iOS App	Primary Responsibility - Develop backend solution and admin tool. - Test backend solution and admin tool. Secondary Responsibility - Develop iOS App
Name: Rigo Garcia <i>Client</i>	Primary Responsibility - Provide overall product direction - Review product during milestones - Consultant for product questions	Primary Responsibility - Provide overall product direction - Review product during milestones - Consultant for product questions	Primary Responsibility - Provide overall product direction - Review product during milestones - Consultant for product questions

3.3 Skills

Table 5: Team Skills

Team members	Role	Skills
Logan Chang	Project Manager, iOS Developer	Current skills: <i>Swift, Objective-C, CocoaPods, Django (Python), MySQL, Play</i> Required skills: <i>Swift</i>
Yunwei Yuan	iOS Developer	Current skills: <i>Objective-C, Swift, MySQL, Play</i> Required skills: <i>Swift</i>
Jiashun Zhou	iOS Developer	Current skills: <i>Swift, Play</i> Required skills: <i>Swift</i>
Yao Wang	iOS Developer	Current skills: <i>Bootstrap, PHP, Ajax, MySQL, Oracle Database, Play, Swift</i> Required skills: <i>Swift</i>
Yew Hua Chua	Backend Developer	Current skills: <i>Play(Java)</i> Required skills: <i>Node.js, Javascript, Server Building Experience</i>
Weili Zhang	Backend Developer	Current skills: <i>PHP code Igniter, Swift</i> Required skills: <i>Node.js, Javascript, Server Building Experience</i>
Haihao Lu	Backend Developer	Current skills: <i>Python, JavaScript, Play(java), C#, PHP</i> Required skills: <i>Node.js, Javascript, Server Building Experience</i>
Nick Funk	Backend Developer, iOS Developer, QFP/Tester	Current skills: <i>Java/JavaFX, C/C++, MySQL, Python, HTML/CSS</i> Required skills: <i>Node.js, Swift, JavaScript</i>
Joseph Mouawad	Backend Developer, iOS Developer, Tester (TPR)	Current skills: <i>C, C++, C#, Java, Python, PHP, SQL, Html, Bootstrap</i> Required skills: <i>Node.js, Javascript, Swift</i>

4. Approach

4.1 Monitoring and Control

4.1.1 Closed Loop Feedback Control

The team uses Slack to provide feedback internally within the team throughout the course of the project. All documents that the team creates are uploaded to a shared Dropbox folder for review. Also, meeting notes and presentation slides are shared through Google Docs that are available for review. Lastly, the team provides feedback regularly in person during the weekly meetings.

4.1.2 Reviews

- The team meets once a week to plan sprint meetings and performs a sprint retrospective. During this meeting the team goes over the priorities and assigns work accordingly.
- The team also meets once a week after class for a quick sync-up to discuss the progress for the current week's sprint as well as any blockers.
- The team performs Code Reviews using Pull Requests through Github. After code is checked-in to the Github repository, the team member creates a Pull Request to merge it into the dev branch. This effectively shows any changes that the team member has performed

4.2 Methods, Tools and Facilities

Table 6: Tools

Tools	Usage	Provider
Jira	Used to create weekly sprints, maintain a backlog of bugs and features, and estimate and log work	Jira
Github	Used to version control source code and host it.	Github
Dropbox	Used to store shared team documents.	Dropbox
Google Docs	Used to store meeting minutes, presentation slides, and other documents subject to consistent change (requirement list, progress report, risk analysis, technical debt, etc.)	Google
Visual Paradigm	Used to create diagrams used for Resilient Agile process (Domain Model, Use Case Diagram, Robustness Diagram, Sequence Diagram, Class Diagram)	Visual Paradigm
Google Hangouts	Used for online team meetings as well as communication with DEN student	Google
Microsoft Word and Project Plan	Used to create formal documentation and project plan	Microsoft
Slack	Used as primary source for team communication	Slack

5. Resources

This section is not applicable to our project since our team used the Resilient Agile process during the first semester, and thus did not perform cost estimation.

6. Iteration Plan

6.1 Plan

The current iteration is the Re-Baselined Foundation Commitment Phase. This plan covers this phase as well as two additional phases, the Core Capability Drivethrough Phase and the Transition Readiness Phase. The team will use an agile approach to developing the capabilities of the product with weekly sprints.

6.1.1 Capabilities to be implemented

Table 7: Construction iteration capabilities to be implemented

ID	Capability	Description	Priority	Iteration
1	UC-01	Upload photo to event	Must Have	CCD
2	UC-02	Take photo for location	Must Have	RDCR
3	UC-03	Create public event	Must Have	RDCR
4	UC-04	Create private event	Must Have	RDCR
5	UC-05	Reserve a hashtag	Must Have	RDCR
6	UC-06	Set a password	Must Have	RDCR
7	UC-07	View photos of event	Must Have	CCD
8	UC-08	View photos of private event with password	Must Have	CCD
9	UC-09	View photos of location	Must Have	RDCR
10	UC-10	Search for event	Must Have	CCD
11	UC-11	Delete uploaded photos	Must Have	CCD
12	UC-12	Delete photos in private event if owner	Must Have	CCD
13	UC-13	Take photo	Must Have	RDCR
14	UC-14	Create account	Must Have	RDCR
15	UC-15	Create profile	Must Have	RDCR
16	UC-16	Login to account	Must Have	RDCR
17	UC-17	Delete expired events	Should Have	TRR
18	UC-18	Delete photos of expired events	Should Have	TRR
19	UC-19	Delete photos	Should Have	TRR
20	UC-20	Delete expired location photos	Should Have	TRR
21	UC-21	Download photo to device	Must Have	CCD
22	UC-22	Search for photos by user	Should Have	TRR

23	UC-23	Search for photos by event	Should Have	TRR
24	UC-24	Search event and upload photo	Must Have	CCD
25	UC-25	Logout of account	Must Have	RDCR
26	UC-26	View photos on device	Must Have	RDCR
27	UC-27	Navigate between screens	Must Have	RDCR
28	UC-28	View joined events	Must Have	CCD
29	UC-29	Join event	Must Have	CCD

6.1.2 Capabilities to be tested

Table 8: Construction iteration capabilities to be tested

ID	Capability	Description	Priority	Iteration
1	TC-01	Create Event	Must Have	CCD
2	TC-02	Search by Location	Must Have	CCD
3	TC-03	Upload Photo to Location	Must Have	CCD
4	TC-04	Delete Uploaded Photo	Must Have	CCD
5	TC-05	Download Photo for Event	Should Have	CCD
6	TC-06	User Interface Actions	Must Have	TRR
7	TC-07	Administrator Photo Management	Should Have	TRR
8	TC-08	Create Account	Must Have	CCD
9	TC-09	Search for Event	Must Have	CCD
10	TC-10	Join Event	Must Have	CCD
11	TC-11	Upload Photo to Event	Must Have	CCD
12	TC-12	Take Photo with Device Camera	Must Have	CCD
13	TC-13	Account Login and Logout	Must Have	CCD
14	TC-14	Automated System Functions	Should Have	TRR

6.1.3 Capabilities not to be tested

There are no capabilities that will not be tested

6.1.4 CCD Preparation Plans

The team will prepare the iOS app on multiple iPhone devices to be tested by the client and any other potential users. The team will also create a list of steps as a user manual that will act as a reference guide during the CCD. To prepare for the CCD, the team will seed the database with existing events and photos that will allow the users to simulate using the app in a real world scenario. In addition, the team will prepare by performing final regression tests as well as running all of the test cases to ensure that each feature is working as expected. Lastly, the team will provide forms in order to acquire feedback from the testers to improve the product

6.2 Iteration Assessment

6.2.1 CCD Capabilities Implemented, Tested, and Results

Table 9: Capabilities implemented, tested, and results

ID	Capability	Test Case	Test Results	If fail, why?
UC-01	Upload photo to event	TC-11	Fail	Could not select event joined by user to upload to
UC-02	Take photo for location	TC-03	Fail	No auto-refresh
UC-03	Create public event	TC-01	Fail	Duplicate events created
UC-04	Create private event	TC-01	Pass	
UC-05	Reserve a hashtag	TC-01	Pass	
UC-06	Set a password	TC-01	Pass	
UC-07	View photos of event	TC-01	Fail	Should be able to upload photos from this screen, missing delete buttons on photos
UC-08	View photos of private event with password	TC-01	Pass	
UC-09	View photos of location	TC-02	Fail	Could delete photos not owned by user
UC-10	Search for event	TC-09	Fail	Could not dismiss keyboard, negative time
UC-11	Delete uploaded photos	TC-04	Pass	
UC-12	Delete photos in private event if owner	TC-04	Pass	
UC-13	Take photo	TC-12	Pass	
UC-14	Create account	TC-08	Pass	
UC-15	Create profile	TC-08	Pass	
UC-16	Login to account	TC-13	Pass	
UC-21	Download photo to device	TC-05	Pass	
UC-24	Search event and upload photo	TC-11	Pass	
UC-25	Logout of account	TC-13	Pass	
UC-26	View photos on device	TC-03	Pass	
UC-27	Navigate between screens	TC-09	Fail	Icons images were incorrect
UC-28	View joined events	TC-10	Pass	
UC-29	Join event	TC-11	Fail	No error message for incorrect password, Users should be able to enter event by tapping on event

6.2.2 Core Capabilities Drive-Through Results

The results of the CCD were disastrous. The iOS application contained numerous bugs and did not have the final design implementation completed that led to a very poor user experience. Many of the acceptance tests were failed as a result and the team was behind schedule. There was an increased risk of not being able to deliver the project on time due to all of the bugs in the system as well as the uncertainty of completing the backend.

Along with fixing all of the bugs and completing the final designs, the client made suggestions for general improvement listed below:

- The client stated that the event name should be case insensitive.
- The client suggested that there should not be a create event option in search event screen.
- The client suggested that users should be able to enter the event after joining by tapping on the event from search results page.
- The client suggested that he should be able to upload photos when viewing his own event.
- The client suggested that there should be a pull-down to refresh feature.
- The client suggested that the “my photos screen” should be moved to profile screen.

6.2.3 Re-CCD Capabilities Implemented, Tested, and Results

Table 10: Capabilities implemented, tested, and results

ID	Capability	Test Case	Test Results	If fail, why?
UC-01	Upload photo to event	TC-11	Pass	
UC-02	Take photo for location	TC-03	Fail	No auto-refresh
UC-03	Create public event	TC-01	Fail	Still able to create duplicate events
UC-04	Create private event	TC-01	Pass	
UC-05	Reserve a hashtag	TC-01	Pass	
UC-06	Set a password	TC-01	Pass	
UC-07	View photos of event	TC-01	Pass	
UC-08	View photos of private event with password	TC-01	Pass	
UC-09	View photos of location	TC-02	Pass	
UC-10	Search for event	TC-09	Fail	Negative time
UC-11	Delete uploaded photos	TC-04	Fail	Could not see delete button anymore for all photos
UC-12	Delete photos in private event if owner	TC-04	Fail	Delete button still missing
UC-13	Take photo	TC-12	Pass	
UC-14	Create account	TC-08	Pass	
UC-15	Create profile	TC-08	Pass	

UC-16	Login to account	TC-13	Pass	
UC-17	Delete expired events	TC-14	Fail	Negative time events showing
UC-18	Delete photos of expired events	TC-14	Fail	Photos of negative time events still showing
UC-19	Delete photos	TC-07	Fail	Delete button shows up when no photos to delete are selected
UC-20	Delete expired location photos	TC-14	Pass	
UC-21	Download photo to device	TC-05	Pass	
UC-22	Search for photos by user	TC-07	Fail	No feedback when no photos are found
UC-23	Search for photos by event	TC-07	Fail	No feedback when no photos are found
UC-24	Search event and upload photo	TC-11	Pass	
UC-25	Logout of account	TC-13	Pass	
UC-26	View photos on device	TC-03	Pass	
UC-27	Navigate between screens	TC-09	Pass	
UC-28	View joined events	TC-10	Pass	
UC-29	Join event	TC-11	Pass	

6.2.4 Retry Core Capabilities Drive-Through Results

The results of the Re-CCD were significantly improved over the first CCD. Many key important bugs were fixed on the iOS application and there was progress on the final design implementation. The admin tool was also completed and available for acceptance testing by the client. However, there were still some critical bugs and new ones discovered during the ReCCD. More acceptance tests were passed this time and the team was back on schedule to finish the project now that the backend was completed. The risk of not being able to deliver the project on time was significantly lowered as a result.

Along with fixing all of the bugs and completing the final designs, the client made suggestions for general improvement listed below:

- The client suggested disabling autocorrect for search fields.
- The client suggested having an automatic join feature for public events once a user had interacted with it.
- The client suggested adding a camera button to the event preview screen.
- The client suggested adding a progress indicator.
- The client suggested showing the navigation bar in multiple screens in order to prevent the user from having to press back multiple times.
- The client suggested moving the delete button on the admin tool.
- The client suggested allowing enter to perform a search.
- The client suggested clicking on a header field should effect perform a search for all users or all events or all photos.

6.3 Adherence to Plan

6.3.1 CCD Adherence

The development for the first CCD iteration was according to plan as multiple features were developed in parallel following the Resilient Agile process. However, there were numerous bugs and extremely difficult merge conflicts that occurred as a result causing significant delay in the project. The team also focused greatly on developing and finishing features over testing, which resulted in number minor bugs and defects along with the major ones caused by the parallel development. The team was struggling greatly to keep on schedule due to inexperience in iOS development as well as pressures from the Resilient Agile process. The custom backend was also incomplete with no confident estimate of a completion date. As a result, the team has shifted to focus more resources on the backend, allow developers to focus on areas more to their personal strengths, and also perform additional testing on the system.

6.3.2 Re-CCD Adherence

The development for the Re-CCD iteration was according to plan and our team was able to complete the custom backend as well as the admin tool for acceptance testing. Many significant bugs and defects were also fixed on the iOS application as well. As a result, the team performed much better during the Re-CCD compared to the CCD. While there is still much work remaining to be completed, the team now has much higher moral as well as a plan to complete all the work by the TRR. The team is now able to focus more resources on the iOS application and especially testing it significantly. After moving away from the Resilient Agile approach and following a more traditional agile approach by allowing the team to self-organize, the team is now back on schedule in terms of finishing the application on time. The team should continue to focus on testing, as that will ensure the quality of the code as well as the software in general.