DC Package

Andrew Borba
Prototyper

Elisabeth Brooks
Project Manager

Jorge Go
Feasability Analyst

Katherine Hu
Lifecycle Planner

Nakul Joshi
Software Architect

Ian Malave
Requirements Engineer

Rishi Mukhopadyay
Operational Concept Engineer

December 6$^{\text{th}}$, 2013

Contents

1 Version History

Date	Author	Version	Changes Made	Rationale
08/20/2013	SK	1.0	Original for CS 477; Tailored from ICSM REQ Template	To fit CS 477 course content
09/14/2013	Team 1	1.1	First version	Project requirements
10/15/2013	Team 1	2.0	Shifted marketing focus from LA Metro to TAP	Reassesment of stakeholder values
11/01/2013	Team 1	2.1	Fixed various design flaws	Incorporated feedback from ARB
12/09/2013	Team 1	2.2	Re-evaluated risks and requirements	Incorporated feedback from DCR ARB

2 Project Plan

2.1 Schedule

Until we get the schedule for next semester with more specific dates, we are only able to identify some key phases and tasks that we will need to complete, but we cannot attach them to dates. Additionally, as we work more on development, this schedule may change as we face challenges, or finish tasks quicker than we estimated. Our current schedule overview and plan is as follows:

January - February Work with Teams

- Rebaseline prototype, prioritize requirements
- Plan for CS477b specifics, including transition strategy
- Participate in ARB review

February - May Scheduled Weekly Meetings to:

- Discuss status and plans
- Provide access to key transition people for strategy and readiness discussions

March 3–7 Iteration Assessment Reviews

March 26 Core Capability Drivethrough

April 14–18 Project Transition Readiness ARB Reviews

April 21 Installation and Transition

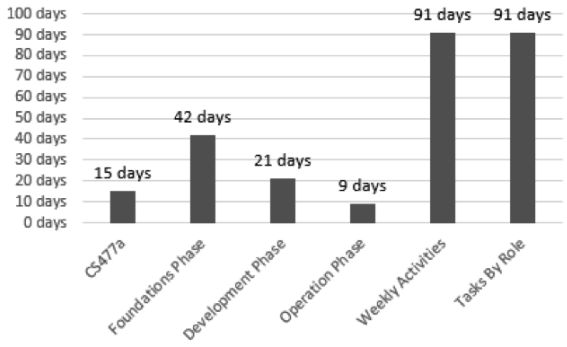
- Install Product
- Execute Transition Plan

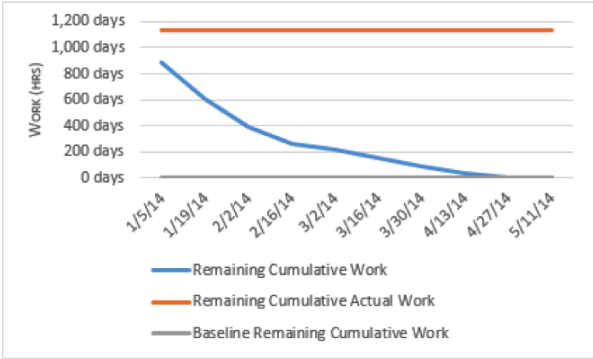
April 28 Operational Commitment Review for Initial Operational Capability

May 5 Client Evaluations

2.2 Project

We have created a project using Microsoft Project to track tasks, phases, and progress. These will be drastically updated as we are provided a more specific schedule of requirements, however sample charts can be seen below.





2.3 Phases and Milestones

Phases planned in the second semester include the end of the Foundations Phase, the Development Phase, which has two parts, construction and transition, and the Operations Phase. Milestones include the RDCR (Rebaselined Development Commitment Review), CCD (Core Capability Drivethrough), TRR (Transition Readiness Review) and OCR (Operational Commitment Review). Our tentative schedule of necessary phases and milestones is below:

Breakdown Element	Steps	Type
Re-Baselined Foundations Phase		Phase
Rebaseline Project Status		Activity
	Assess Feasibility Evidence	Task Descriptor
	Assess Life Cycle Content	Task Descriptor
	Assess Operational Concept	Task Descriptor
	Construct Traceability Matrix	Task Descriptor
	Assess System Architecture	Task Descriptor
	Assess requirements definition	Task Descriptor
	Assess Prototype and Components	Task Descriptor
Prepare for Development Phase		Activity
	Develop Transition Plan	Task Descriptor
	Identify Development Iteration	Task Descriptor
Plan for Testing		Activity
	Identify Test Plan	Task Descriptor
	Identify Test Cases	Task Descriptor
Plan and Manage Project		Activity
	Detail Project Plan	Task Descriptor
	Record Project Progress	Task Descriptor
	Re-Baselined Development Commitment Review	Milestone

Breakdown Element	Steps	Type
Development Phase		Phase
Construction Iteration		Iteration
	Detail Project Plan	Task Descriptor
	Record Project Progress	Task Descriptor
	Perform Core Capabilities Drive-Through	Task Descriptor
	Assess Development Iteration	Task Descriptor
Implement the System		Activity
	Develop Components	Task Descriptor
	Develop Glue Code	Task Descriptor
	Integrate Components	Task Descriptor
	Tailor Components	Task Descriptor
	Perform Testing	Task Descriptor
	Core Capability Drivethrough	Milestone
Construction Iteration		Iteration
	Detail Project Plan	Task Descriptor
	Record Project Progress	Task Descriptor
	Assess Development Iteration	Task Descriptor

Breakdown Element	Steps	Type
Implement the System		Activity
	Develop Components	Task Descriptor
	Develop Glue Code	Task Descriptor
	Integrate Components	Task Descriptor
	Tailor Components	Task Descriptor
	Perform Testing	Task Descriptor
	Transition Readiness Review	Milestone
Transition Iteration		Iteration
	Provide Training	Task Descriptor
	Transition The System	Task Descriptor
	Operation Commitment Review	Milestone
Operation Phase		Phase
	Feasibility Analyst	Role Descriptor

2.4 Team

For our team structure and roles, we will have each member coding, testing, and reviewing the work of someone else on the team. We plan to have peer coding,

where two people code together. While each member will be have multiple roles and responsibilities, we have broken the team up into front-end and back-end, testers and trainers, and then each member will be an implementer. As the semester of 477b progresses, we may update these roles as we feel necessary, depending on how we progress. The following charts lists the team structure and each team member, their positions for 477a/b, and their tentative roles and responsibilities. As we begin work, these may change due to what the team needs.

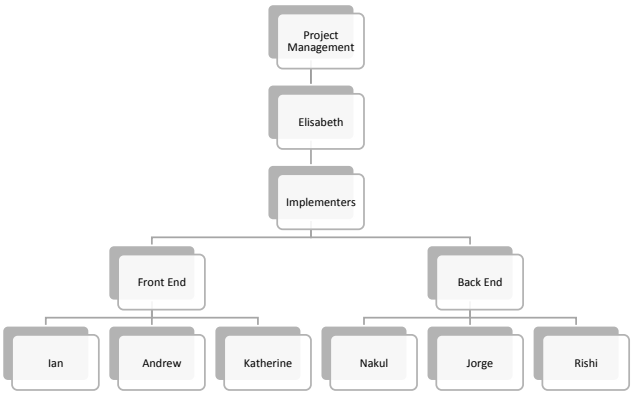


Figure 1: Team Structure: Implementation

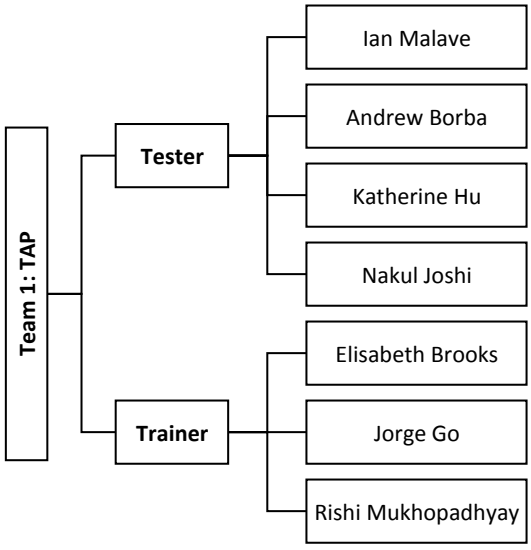


Figure 2: Team Roles: Testing and Training

Team Member	Role	Responsibility
Elisabeth Brooks	Project Manager	Create and manage schedules Maintain project management tool (Asana) Hold team members accountable
Ian Malave	Requirements Engineer Front End Developer	Develop front-end and prototype Update requirements to match prototype
Andrew Borba	Prototyper Front End Developer	Create application prototypes Develop the front-end
Katherine Hu	Life Cycle Planner Front End Developer	Validate application design Ensure user-friendliness and legibility
Nakul Joshi	Systems Architect Back End Developer	System & security design on high-level back-end development Manage LaTeX documentation
Jorge Go	Feasibility Analyst Back End Developer	Identify and analyze risks Plan risk mitigation strategies Run business case and cost-benefit analysis
Rishi Mukhopadyay	Operational Concept Engineer Back End Developer	Implement back-end Plan database structure

3 Operational Concept Description

3.1 Shared Vision

3.1.1 System Overview

Key Partners

1. TAP

Key Activities

1. Software Design and Development
2. Integration with Metro infrastructure
3. Marketing of application

Key Resources

1. Development Team
2. PhoneGap API
3. NFC technology
4. QR technology

Value Proposition

1. Convenience for customers to purchase and use metro tickets
2. Ticket elimination reducing cost and environmental impact
3. Technological advancement of public transportation system.

Customer Relation

1. LA Metro
2. Apple Appstore
3. Android Store
4. Windows Phone Marketplace

Channels

1. Application stores
2. LA Metro website
3. Posters and billboards at stations.

Customer Segments

1. Transportation Companies

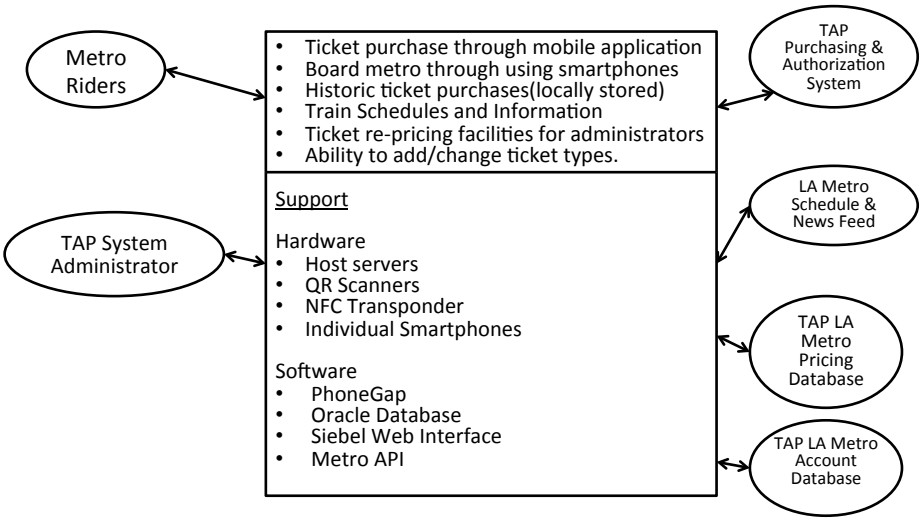
Cost Structure

1. Development Team
2. Back-end System Administrator

Revenue Streams

1. Flat fee for project implementation
2. Recurring fee per ticket sale through application

3.1.2 System Boundary and Environment



3.2 System Transformation

3.2.1 System Objectives, Constraints, and Priorities

Capability Goals	Priority Level
OC-1 Cross-platform Compatible: The application is compatible with iOS, Android and Windows Phone	Must have
OC-2 Account Creation: The application is able to create new rider accounts, update information, and log in users using existing information.	Must have
OC-3 Usage: The application allows metro riders to board trains via NFC or QR code technology.	Must have
OC-4 Payments: The application allows metro riders to pay for tickets using a secure payment gateway and will allow metro riders to store credit card information.	Must have
OC-5 Schedules: The application allows metro riders to view train schedules.	Should have
OC-6 Map: The application allows metro riders to view a static map of metro stations.	Should Have
OC-7 Updates: The application allows metro riders to receive updates about train station information and irregular service interruptions.	Should Have
OC-8 Pricing: The administration application will allow metro administrators to change ticket pricing.	Should Have
OC-9 Ticket Types: The application will support linking of additional tap accounts for the purpose of allowing dependants to be charged with one QR scan.	Could Have
OC-10 Ticket Display: The application will use GPS services to automatically display tickets for the closest station.	Should Have

Organizational Goals

OG-1 Increase convenience for ticket buyers.

OG-2 Decrease cost for LA Metro and ticket buyers.

OG-3 Increase efficiency of public transit system by advancing technologically.

Level of Service Goals	Priority Level
Reliability of application	Must have
Usability	Must have
Security	Must have
Performance of system	Should have
Inter-operability	Should have
Maintainability	Should have

Constraints

CO-1 Align with Current Infrastructure: The new application must complement the existing tap card system, and be implemented with minimal changes to existing infrastructure.

CO-2 Cross Platform Compatibility: The new application must be compatible with major smartphone operating systems(iOS, Android and Windows)

CO-3 Phone Hardware: The application must be compatible with the existing hardware in major smartphones.

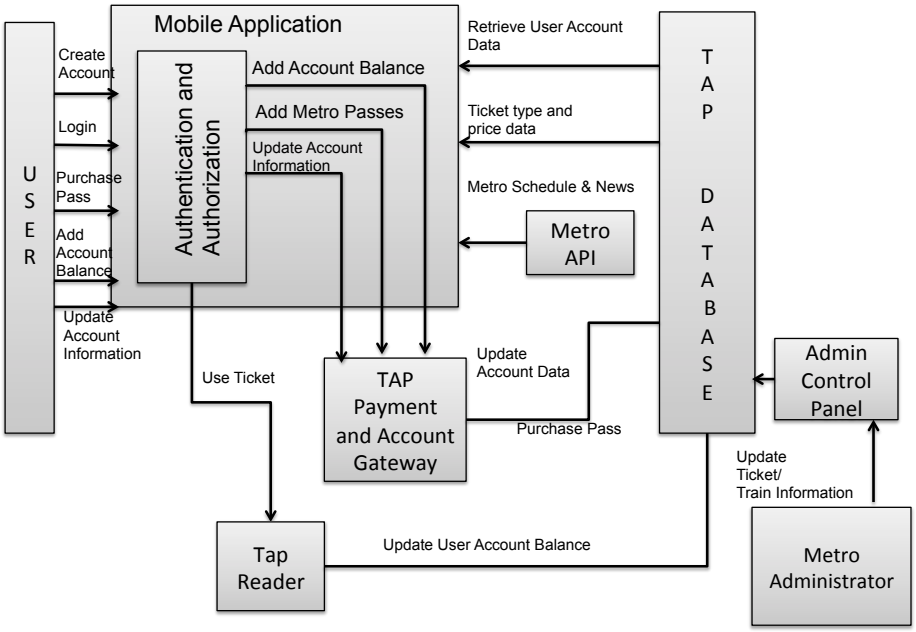
3.2.2 Proposed Operational Concept

The application will allow metro riders to use their mobile devices to purchase tickets and board trains eliminating the need for physical TAP cards. The new system will act as a complement to the TAP card system; allowing riders to use existing metro facilities if they wish.

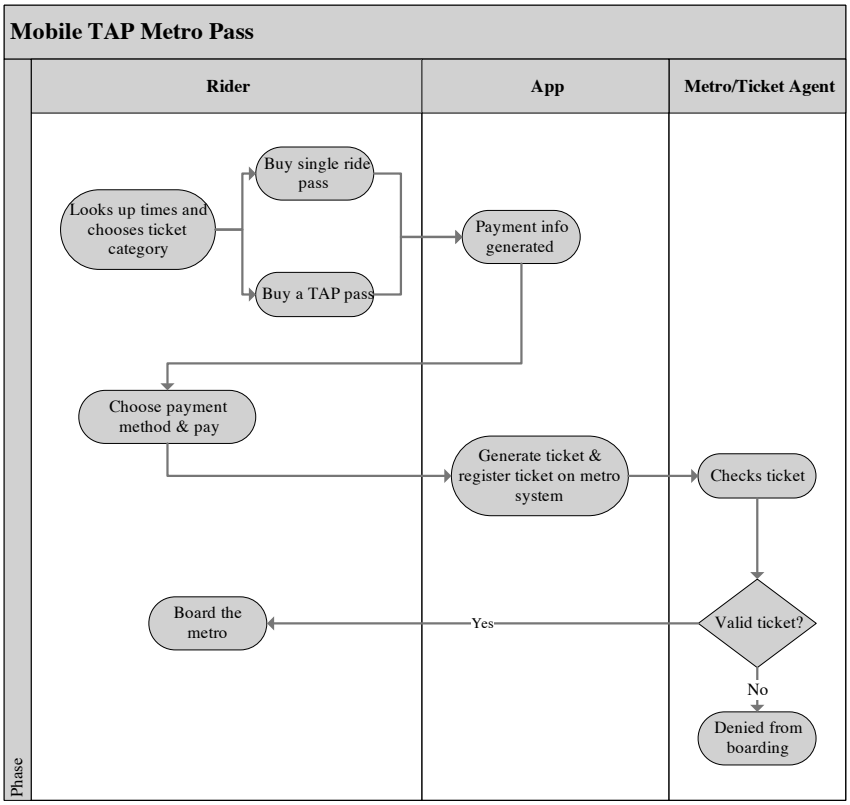
For those smartphones enabled with NFC, the rider will be able to tap their phones instead of their tap cards. For those smartphones not enable with NFC, QR readers will be installed at stations that have turnstiles. For stations that do not have turnstiles, Metro agents will be given a QR reader to verify tickets on the train.

The application will allow users to view train schedules along with a map of the stations in the users metro system. Users will also receive news updates in case of unusual delays or train cancellations.

3.2.3 Proposed Element Relationship



3.2.4 Proposed Workflow



4 Requirements

4.1 Capability Requirements

4.1.1 Platform

1. The application shall be compatible with the iOS mobile platform.
2. The application shall be compatible with the Android mobile platform.

4.1.2 User Accounts

1. If it is a users first-time opening the application, the application shall prompt the user to enter an email address, password, credit card information, and to create or link to a TAP account.
   (a) If the account is created successfully, the application shall send a confirmation to the users stored email address.
   (b) If the account creation is not successful, the application shall display an error message that prompts the user to reenter their information and will not be logged in.
2. If a returning user opens the application, it shall prompt the user to enter their log-in information.
   (a) If the log-in attempt fails, the application will display an error message that prompts the user to reenter their log-in information.
   (b) If the log-in attempt succeeds, the user will be shown a transit management screen.
3. The user shall be able to allow the application to automatically sign them in.
4. The user shall be able to sign out of their account.

4.1.3 Usage

1. If the application loses connectivity, the application shall display an error message.
2. The application shall retrieve the users location at intervals of 120 ± 10 seconds.
3. The application shall allow the user to manually refresh their location.
4. Upon updating the users location, the application shall update information for nearby Metro transportation.
5. The user shall be able to select which bus or train line is displayed first when the user is nearby a particular Metro station.

6. The application shall allow the user to change their email address.
7. The application shall allow the user to change their password.
8. The application shall allow the user to delete their account.
9. The user shall be able to link up to 5 dependents (name & birthdate) to their account.
10. The user shall be able to link up to 5 dependents TAP accounts to their account.

4.1.4 Payments

1. The application shall allow users to purchase non-exclusive TAP passes.
2. If the user selects to purchase a ticket, the application shall check to see if the user has an applicable TAP pass.
   (a) If the user has an applicable TAP pass, the application shall generate the selected virtual ticket.
   (b) If the user does not have an applicable TAP pass, the application shall confirm with the user it will charge the stored credit card.
   (c) If the card does not process, an error message shall be displayed and the order shall not be accepted.
   (d) If the transaction succeeds, the application shall create a virtual ticket.
3. If the user has more than one TAP unused pass, the application shall prompt the user to select which to use first.
4. The user shall be able to select up to 2 dependents accounts to be attached to a virtual ticket.
   (a) If the dependent is over 5 years old, the application shall calculate the additional ticket fare.
5. If the user selects to use their ticket, the application shall signal the gate up to a maximum of 3 attempts at 10 ± 5 second intervals to accept.
   (a) If the application is unsuccessful on the 3rd attempt, the application shall display an error message.
   (b) Otherwise, the gate is unlocked, permitting the user to pass through.
6. The application shall store unused virtual tickets for at least 1 year.
7. The application shall store used virtual tickets for at least 30 days.
8. The application shall allow the user to change their credit card information.

9. The user shall be able to set up the application to generate a default ticket that auto-charges their stored credit card when used.
10. The application shall support remote adjustment of ticket prices (e.g., senior discount) by TAP id through a web interface.
11. If the user has an active TAP pass, the application shall display how much time is remaining on the pass.

4.2 Level of Service Requirements

LOS Requirements	Desired Level	Accepted Level
LOS-1: Concurrent Users	150000	75000
LOS-2: Start-up and user location time	7 s	15 s
LOS-3: Ticket Purchase Transaction Time	5 s	20 s
LOS-4: Update Account Information Time	10 s	30 s
LOS-5: Tickets stored per User	1500	500
LOS-6: % first-time users able to purchase ticket without outside help	80	75
LOS-7: % first-time users able to use ticket without outside help	85	80
LOS-8: % users that ride metro at least once per week that would rate ease of use at 3 out of 5 or higher	90	80
LOS-9: Average Time for User to Create an Account	45 s	60 s
LOS-10: Failed Ticket Purchases per 1000	0.25	1
LOS-11: Failed Ticket Uses per 1000	0.25	1
LOS-12: Hours per day that app shall purchase tickets	20	19.5
LOS-13: Hours per day that app shall allow use of tickets	20	19.5
LOS-14: # iOS generations app shall support	3	2
LOS-15: # Android generations app shall support	3	2
LOS-16: # versions of app that Metro system shall support	3	2

5 Risk Lists

5.1 Major Risks

Network Risk *Current Rank:* 1 *Previous Rank:* N/A

Risk Description Since our app will require a live connection to TAP/LA Metro infrastructure to access ticket prices, and send, receive, and verify data, a weak internet signal can cripple the app's functionality. This can result in lost revenue from users who cannot use the app because of weak network connectivity. This is largely an external risk beyond our control.

Risk Exposure As presented in the groups ARB session, below is a risk exposure calculation to estimate revenue loss per month from network connectivity risks.

	Item	Exposure		
		High +20%	Normative	Low -20%
A	Avg. number of passengers/day	162,333	135,278	108,222
B	Avg. ticket price		$2.00	
C	Estimated % of passengers affected	2.4%	2.0%	1.6%
D	Estimated number of passengers affected (A×C)	3,896	2,706	1,732
E	Estimated revenue loss/day (B×D)	$7,792.0	$5411.1	3,463.1
F	Days/month		30	
G	**Total expected loss/month (E×F)**	**$233,760**	**$162,333**	**$103,893**
H	Probability of network issues	96%	80%	64%
	Total monthly risk exposure (G×H)	**$224,410**	**$129,867**	**$66,492**

Mitigation Action Items The app will inform the user of connectivity errors and show a streamlined version of the app that does not require connectivity. For example, it will show ticket price quotes from the last update, main route availabilities, and transaction history. If possible, the group will work with LA Metro to develop ticket checking systems and protocol on the app without using internet, or via alternative methods such as Bluetooth.

Technology Risk *Current Rank:* 2 *Previous Rank:* 3

Risk Description A technical malfunction of our platform might cause boarding delays for passengers and cause inefficiencies in LA Metro operations. For example, if the app fails to update prices or routes, passengers may seek alternative transportation, which translates to lost revenue for the LA Metro.

Risk Exposure As presented in the groups ARB session, below is a risk exposure calculation to estimate revenue loss for an instance of realized technology risk.

	Item	Exposure		
		High Case + 20%	Normative	Low Case - 20%
A	Avg. Number of Passengers/ Day	162,333	135,278	108,222
B	Avg. Ticket Price		$2.0	
C	Estimated % of Passengers affected	60.0%	50.0%	40.0%
D	Estimated Number of Passengers affected (A*C)	97,400	67,639	43,289
E	Estimated Revenue Loss/ Day (B*D)	$194,800.0	$135,277.8	$86,577.8
F	Estimated number of days to fix problem	4.8	4	3.2
G	**Total Expected Loss (E*F)**	**$935,040**	**$541,111**	**$277,049**
H	Probablity of Technology malfuction	6.0%	5.0%	4.0%
	Total Risk Exposure (G*H)	**$56,102**	**$27,056**	**$11,082**

Mitigation Action Items The group will implement non-normative functionality for certain big cases of technology risk. For example, if ticket price update functions are inhibited, the app will charge the user a pre-set fare that is lower than the average rate to compensate for the technology malfunction. Also, there will be a disclaimer advising time-sensitive passengers to have a back-up physical TAP card.

Security Risk *Current Rank:* 3 *Previous Rank:* 1

Risk Description An unsecure platform can be exploited (e.g. loopholes to avoid fees, reusing tickets, etc.). An unsecure payment process might compromise user accounts, user information, or LA Metro information. These problems can cut into LA Metro revenue.

Risk Exposure As presented in the groups ARB session, below is a risk exposure calculation to estimate revenue loss for an instance of realized security risk.

	Item	Exposure		
		High Case + 20%	Normative	Low Case - 20%
A	Avg. Number of Passengers/ Day	162,333	135,278	108,222
B	Avg. Ticket Price		$2.0	
C	Estimated % of Passengers involved	24.0%	20.0%	16.0%
D	Estimated Number of Passengers involved (A*C)	38,960	27,056	17,316
E	Estimated Revenue Loss/ Day (B*D)	$77,920.0	$54,111.1	$34,631.1
F	Estimated number of days to fix problem	12	10	8
G	**Total Expected Loss (E*F)**	**$935,040**	**$541,111**	**$277,049**
H	Probablity of Security issue	1.2%	1.0%	0.8%
	Total Risk Exposure (G*H)	**$11,220**	**$5,411**	**$2,216**

Mitigation Action Items Our groups architect will choose an existing encryption system for our application that will be tested on our app until it meets the clients requirements and security standards.

Platform Compatibility Risk *Current Rank:* 4 *Previous Rank:* 8

Risk Description The group will be using PhoneGap to implement the mobile application with standard web technologies such as HTML/CSS/JavaScript which makes the application subject to device-specific mobile browser display discrepancies. Performance inconsistencies may require additional costs to fix.

Risk Exposure As presented in the groups ARB session, below is a risk exposure calculation to estimate the cost to LA Metro/TAP to fix an escalated instance of realized platform risk.

	Item	Exposure		
		High Case + 20%	Normative	Low Case - 20%
A	Estmated Hourly Rate for IT contractor	$42.0	$35.0	$28.0
B	Estimated Team Size	4	3	2
C	Estimated Fees for 8 hour day (8*A*B)	$1,209.6	$840.0	$537.6
D	Estimated number of days to tweak compatibility	10	8	6
E	**Total Expected Loss (C*D)**	**$11,612**	**$6,720**	**$3,441**
F	Probablity of compatibility issues	12.0%	10.0%	8.0%
	Total Risk Exposure (E*F)	**$1,393**	**$672**	**$275**

Mitigation Action Items The app will be tested on a wide variety of devices (different iPhone versions, windows phone and various Android phones) to ensure the application is rendered properly on all screen sizes and OS versions.

The group has researched the option of developing natively for each platform but it is not feasible given time, costs, and expertise constraints. The group has tested simpler PhoneGap apps on various phones, which ran better and more smoothly than expected, resulting in lower Platform compatibility risk exposure (from first ARB).

Changing Requirements Risk *Current Rank:* 5 *Previous Rank:* 6

Risk Description After the platform has been implemented and integrated, the LA Metro/TAP requirements might change, or the technologies being used

in relevant processes might change. Such scenarios may require additional costs to modify the app.

Risk Exposure As presented in the groups ARB session, below is a risk exposure calculation to estimate the cost to LA Metro/TAP to fix an instance of realized changing requirements risk.

	Item	Exposure		
		High Case + 20%	Normative	Low Case - 20%
A	Estmated Hourly Rate for IT contractor	$35.0	$35.0	$35.0
B	Estimated Team Size	5	4	3
C	Estimated Fees for 8 hour day (8*A*B)	$1,344.0	$1,120.0	$896.0
D	Estimated number of days to add requirements	10	10	10
E	**Total Expected Loss (C*D)**	**$13,440**	**$11,200**	**$8,960**
F	Probablity of additional Requirements	5.0%	5.0%	5.0%
	Total Risk Exposure (E*F)	**$672**	**$560**	**$448**

Mitigation Action Items If the team is under a contract/agreement, the team will develop updates/patches that will adapt to new requirements. Otherwise, we will give the LA Metro a copy of our comprehensive documentation for the software so that necessary changes can be handled in-house or outsourced to another contractor.

The group has conducted a survey to evaluate currently prototyped features. Good feedback was received from a representative sample size of LA Metro users, which should lower requirements risk exposure (relative to first ARB).

5.2 Minor Risks

Partnership Risks *Current Rank:* 6 *Previous Rank:* 6

Risk Description LA Metro might not be willing to implement our app with their system, or allow us to test compatibility with their system.

Mitigation Action Items Our project manager is currently in talks with the LA Metro authorities, but we could also potentially do a proof of concept using similar hardware/systems. TAP could also be considered a potential partner, as LA Metro outsources projects such as these to TAP.

Shifted targeting focus to TAP instead of LA Metro. E-mailed TAP to discuss plans moving forward.

Estimation Risks *Current Rank:* 7 *Previous Rank:* 5

Risk Description The estimated mount of work might not be completed in two semesters, or another developer might implement the system first.

Mitigation Action Items We will develop a running list of unexpected risks as they come up and develop strategies to approach them. For example, we would explicitly ask LA Metro for their timeline and ask for an exclusive partnership.

Utilized Asana as project planning tool to plan advanced deadlines, deliverables, and responsibilities.

Expertise Risks *Current Rank:* 8 *Previous Rank:* 7

Risk Description Team members might lack necessary expertise in certain development requirements.

Mitigation Action Items Identify team member skills and project requirements so we can individually prepare ourselves for technologies that we will be implementing next semester

Researched and found software tools such as PhoneGap that can help us create a functional app more easily.

Organizational Risks *Current Rank:* 9 *Previous Rank:* 9

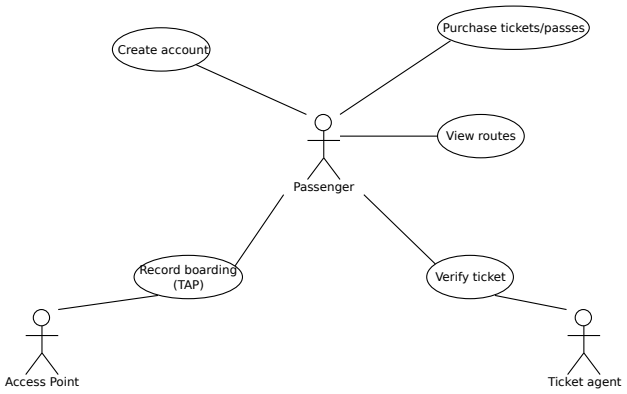
Risk Description Statically defined team roles might not fit specific project requirements.

Mitigation Action Items Dynamically reassign team roles as the project and its requirements mature. Also the team shall be aware that responsibilities will be very flexible and we may need to step outside the scope of our assigned position.

Assigned responsibilities dynamically via Asana and scheduled more frequent group meetings.

6 Architecture

6.1 Use Cases



The basic use scenarios are:

Account creation The user can create accounts from within the app.

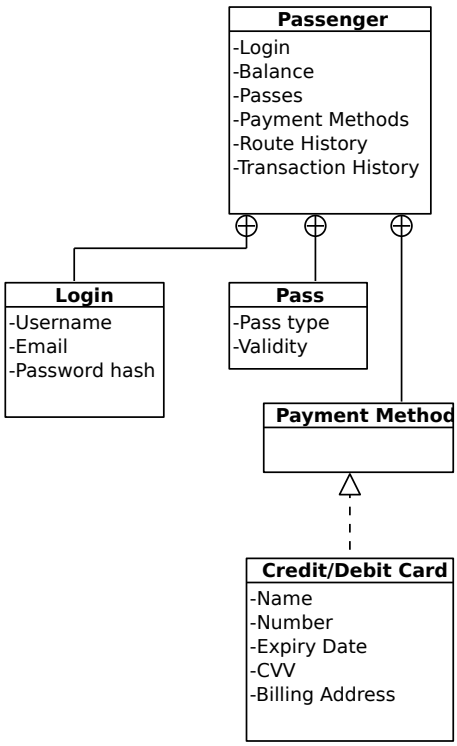
View routes Upon locating the user via GPS, the app presents a list of nearby routes.

Purchase of tickets/passes The user may purchase a pass, or simply add money to their TAP funds.

Record boarding Upon boarding the train/bus, the user must record this by presenting the QR code to the turnstile/bus driver.

Ticket verification A ticket agent aboard a train/bus may ask passengers to present proof of payment at any time. This is again verified by a QR code.

6.2 Class Hierarchy



The important classes to be maintained are:

Passenger Maintains passenger data such as login info and payment history.

Login Holds a username paired with a password that is hashed for security.

Passes Maintains the list of fare payment methods available to the passenger. Also selects the most appropriate one for each journey (Eg. If a passenger has a day pass then fare will not be deducted from TAP funds.)

Payment Methods An abstraction for various payment methods. This layer allows addition of functionality in the future, eg. Google Wallet and Paypal.

6.3 Software Choices

7 Prototype

7.1 Survey for Feedback

Before we updated our prototype, we sent out a survey to a range of participants to get feedback on our original starting prototype. Survey participants included both frequent, and non-frequent Metro riders, people who were both tech-savvy and those who didnt use smartphones much, males and females of a range of ages, and colorblind individuals. Based on this feedback, we came up with updates to the design, which were then implemented and presented at our DCR ARB Presentation and Review. Our survey asked the following questions:

- Personal Information
  - Name
  - Age
  - Occupation
  - Email Address
- Background
  - How often do you take the Metro? If not at all, why not?
  - Have you ever used a TAP card or similar pass? (Y/N)
  - Do you have a car? (Y/N)
  - Do you regularly use services on your smartphone? (Y/N)
  - Would you use this application on your smartphone (instead of or in addition to a TAP card? (Y/N)
- Feedback
  - What did you find difficult within the app?
  - Was it aesthetically pleasing? (1-5)
  - What was the most challenging thing about navigating the app?
  - What did you like about the app?
  - What additional features, if any, would you like to see?

7.2 Results

Features Participants said that the location setting was confusing, and that the times for nearby transportation are important to have. They also felt that tickets were hard to purchase and then find/use after they were purchased. These were major updates that were made to the new prototype.

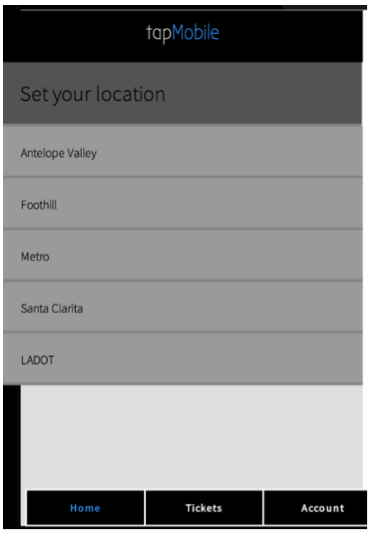
Aesthetics Survey participants liked the overall look of the app. They enjoyed the color scheme, and didnt find it hard to read. We also made sure we had identified color-blind individuals test the app, to make sure that it was readable by the majority of potential users.

7.3 Significant Changes in Capabilities

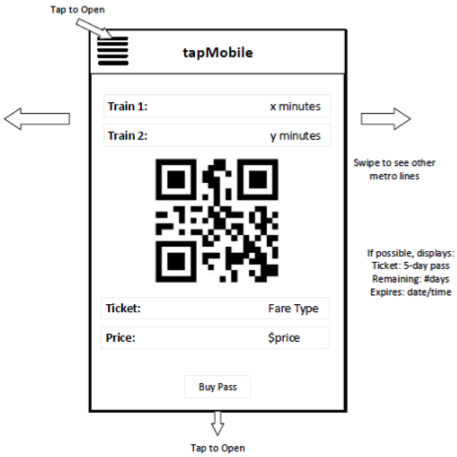
Changes in prototype capabilities are listed below:

- Got rid of the tabbed layout in favor of a more focused main home page with less important screens being accessible from a side panel
- Support for linking additional tap accounts for the purpose of allowing dependents to be charged with one QR scan
  Eg: a mom would create tap accounts for her kids, links those accounts to her account through the app and then can use her phone to scan the kids in
- App will use your location (updated every 2 minutes) to find the nearest metro line to location and will automatically display your ticket on the home screen
- If GPS is off or you're using the bus, ability to manually choose a ticket from list as well
- The home screen will display arrival times of the nearest train line to currently location along with destination of train
- Visually display any special statuses such as senior citizen, child or any discount passes

7.4 Prototype Updates



(a) Version 1



(b) Version 2

Figure 3: Home Page

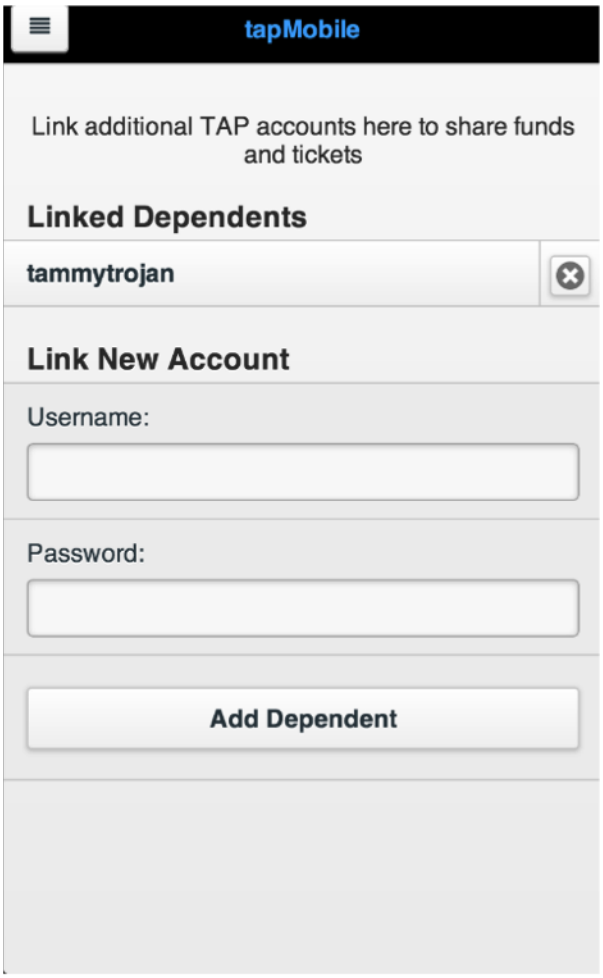


Figure 4: Dependents Page

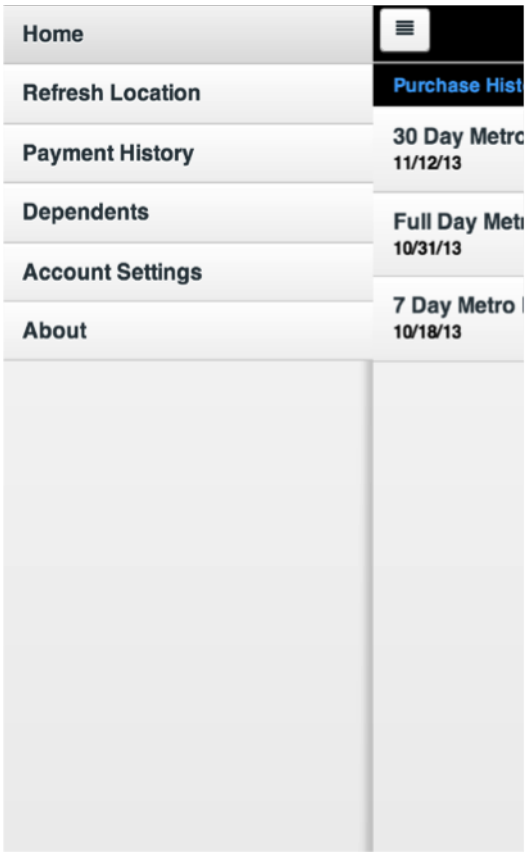


Figure 5: Side Panel