Software Functional Specification

Software Development

Final Project - Baggage Recovery App

Wei Liu
Ziwei Dai
Lixin Chen

TABLE OF CONTENTS

Project Information

Project Name:	Baggage Recovery Application
Author:	Ziwei Dai, Wei Liu, Lixin Chen
Organization:	Rensselaer Polytechnic Institute
Project Version:	v1.0.0
Spec Version:	v1.0.0
Date of Current Spec:	Dec 8, 2018

1. Introduction

1.1 Objective

The objective of this Software Functional Specification (SFS) document is to provide a detailed description of the functionalities of the Baggage Recovery Application. This document will cover each of the app's intended features, as well as offer a preliminary glimpse of the software application's User Interface (UI). The document will also cover hardware, software, and various other technical dependencies.

1.2 Project Scope

The Baggage Recovery Application is composed of two main applications: a customer-side application, and a warehouse-side application. The application is designed to facilitate the process of baggage recovery. Potential scenarios include managing baggages, reporting missing baggages, matching baggages, customer verification, etc.

For more information about the project and its goals, see the Project Plan documentation.

1.3 Intended Audience

This document is intended for all individuals participating in and/or supervising the Baggage Recovery Application project. Readers interested in a brief overview of the product should focus on the rest of Part 1 (Introduction), as well as Part 2 of the document (Overall Description), which provide a brief overview of each aspect of the project as a whole. These readers may also be interested in Part 6 (Project Timeline) which lays out a concise timeline of the project.

Readers who wish to explore the features of Baggage Recovery Application in more detail should read on to Part 3 (System Features), which expands upon the information laid out in the main overview. Part 4 (Requirements) offers further technical details, including information on the user interface, the hardware and software platforms on which the application will run, performance, safety, security, as well as various other attributes that will be important to users.

Readers who are interested in the test process should check Part 5 (Test Plan), which includes any test information.

1.4 References

Other detailed installation guide of this application can be checked through the Readme file in the zipped code.

1.5 Document Revision History

Date	Version	Author	Desc of changes
10/01/2018	1.0.0	Ziwei Dai	Build the structure of func spec.
11/12/2018	1.0.1	Ziwei Dai	Modify all parts depends on the implementation.
12/08/2018	1.0.2	Ziwei Dai	Final revision

2. Overall Description

2.1 Product Perspective

The Baggage Recovery Application is a new, self-contained product intended for use for the Lost and Found in Airport. While the Baggage Recovery Application is a mobile-first web application, there is also a server-side component which will be responsible for database services. The scope of the project encompasses both warehouse- and customer-side functionalities, so both aspects are covered in detail within this document.

2.2 Product Features

The following list offers a brief outline and description of the main features and functionalities of the Baggage Recovery Application. The features are split into two major categories: customer features and warehouse features.

Customer Features

1) Customer login
    - Use the customer ID to login to the application.
2) Manage my baggage
    - Enable the customer to register a new bag (including bag image and bag name).
    - The customer can edit the registered bag name.
    - The customer can delete the saved bags.
3) Report missing baggages
    - Fill out a form to report the missing baggages. The report will have all the information about the baggage.
4) Check baggage status
    - The missing baggage status will show on the screen.
    - When the baggage is found by the warehouse employees, the customer can confirm their ownership by choosing the right bag and content images.

Warehouse Features

1) Warehouse login
    - Warehouse employees will log in to the application to process all the baggage and missing reports.
2) Register new baggages
    - Warehouse employees can register new baggage.

    - The saved information will be matched with the missing baggage reports to find the right match between the warehouse baggage and the missing report.
    - After registering a new baggage, the application will run the matching algorithm automatically and show the results.
3) Process missing baggage report
    - List all the missing baggage report.
    - Each report will show three system matches.
    - Warehouse employees can close the report manually. They can also confirm the right match.
4) Check resolved reports
    - For all resolved reports, staffs can contact the customer with the scheduled appointment time.
    - Return the bag with the address provided by the customer.
    - Close the report manually.

2.3 User Classes and Characteristics

The Baggage Recovery Application project is meant to offer a baggage recovery solution that is faster, easier, and more convenient than manually finding and matching missing baggages. Consequently, the application will have little or no learning curve, and the user interface will be as intuitive as possible. Thus, technical expertise and web application experience should not be an issue. Instead, anticipated users can be defined by how they will use the product in a particular situation. This application will be used by a variety of different users and includes both customers and employees. As such, the team had to accommodate all kinds of users and their different needs. For example, some users might have limited technical knowledge and would get frustrated by a complicated interface. Others might have disabilities that prevent them from accessing the application in certain ways.

The following list categorizes the users and their corresponding characteristics in which the Baggage Recovery Application is expected to be utilized:

Baggage Recovery Application: Potential Users

1) Customer
    - Frequently flyer.
    - Many baggages to check-in.
    - Is used to use a web-app to process everything.
2) Warehouse or airport employees
    - Experienced using tablets or computers to register information.
    - Works for some airlines that have a warehouse-like location to store all the missing baggages.

- Experienced using phone or camera to take pictures.

2.4 Operating Environment

The main component of the Baggage Recovery Application project is the mobile-first web-application, which will be not be limited to any operating system. The application required that warehouse employees need hardware which has the capacity to take pictures and upload them to the application via a web browser or mobile application. This hardware can be a smartphone, tablet, or laptop. For the development of the application, the team used laptops and smartphones to develop and test the application.

2.5 Design and Implementation Constraints

The team primarily focused on mobile accessibility since the application is designed to be used through a phone. The team also made the user interface clear and intuitive to appeal to new and infrequent users as well as people with low literacy or non-fluency in English. This was done through thoughtful use of graphics and buttons to support any textual instructions. The team also strived to accommodate users with visual impairments like color blindness.

2.6 User Documentation

The primary goal of the Baggage Recovery Application is to facilitate the process of baggage recovery. Consequently, the application will be designed to be as simple to use as possible. Nonetheless, users may still require some supplementary information about each component of the Baggage Recovery Application. The application will have a documentation called system configuration to provide this.

2.7 Assumptions and Dependencies

Some of the features rely on hardware components. For instance, the camera will be used to record images of baggages for digital storage. Consequently, this feature is entirely reliant upon the ability to access the camera’s functionalities. The camera functionalities will be achieved using the API provided by the web browser.

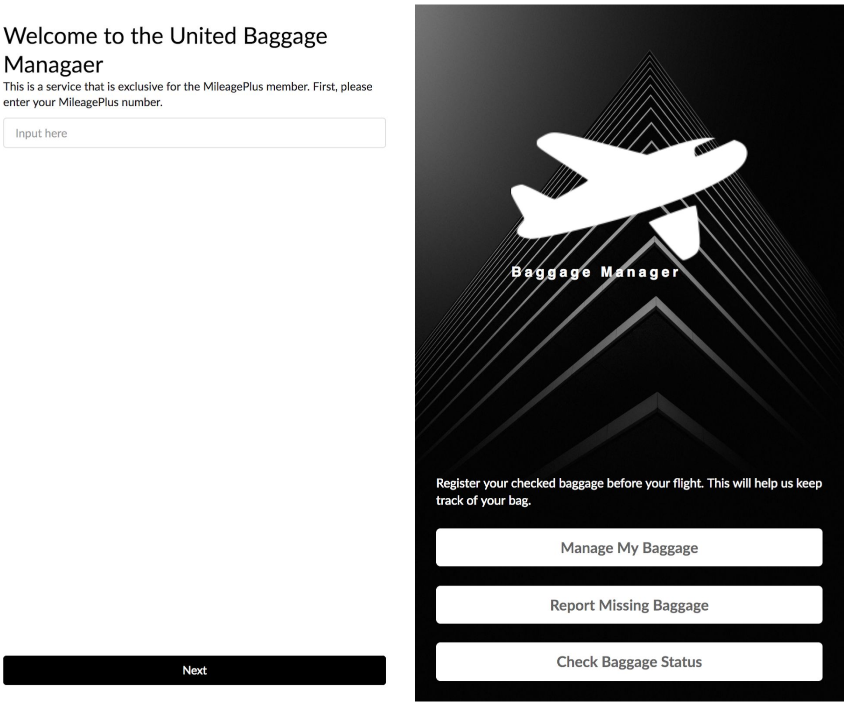
3. System Features

Baggage Recovery Application’s system features are divided into two main categories: customer features and warehouse features. Customer features include any features that are targeted to the passenger. Warehouse features include any features that are targeted to the warehouse employees.

3.1 Customer Features

3.1.1 Customer Login

For the login and landing pages, the user sees the possible options for the processes they will follow in a clear and minimal way. These landing pages also prominently display an airplane icon and utilize a black color scheme as a way to brand the application. These landing pages can be seen in Figure 1 below.



Welcome to the United Baggage Managaer

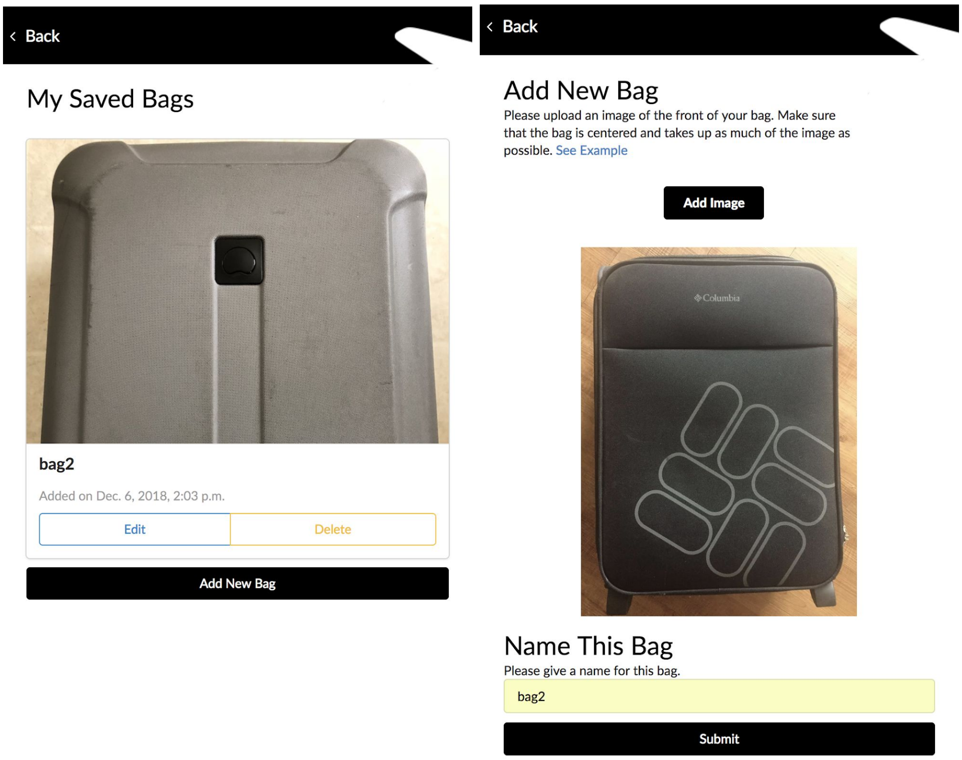
This is a service that is exclusive for the MileagePlus member. First, please enter your MileagePlus number.

Register your checked baggage before your flight. This will help us keep track of your bag.

Figure 1: Customer login (left) and Customer Landing Page(right) Interfaces

3.1.2 Manage My Baggage

One of the major functionalities on the customer side is the ability to add, edit, and delete images of stored bags. When the user clicks to manage their bags, they are redirected to a page containing all of their saved bags. From there, the user can select either to edit or delete an existing bag, add a new bag, or go back to the menu. Since it is import to notify users before doing a potentially irreversible action, the "delete" button is stylistically different and when clicked, it presents the user with a dialog asking the user to confirm the action. When a user decides to add a new bag, they are prompted with clear instructions for each step of the upload process. The user is also able to see the state of the upload with an animation to indicate the progress and upload completion. Some images from the baggage management interface are illustrated in Figure 2 below.



Please upload an image of the front of your bag. Make sure that the bag is centered and takes up as much of the image as possible. See Example

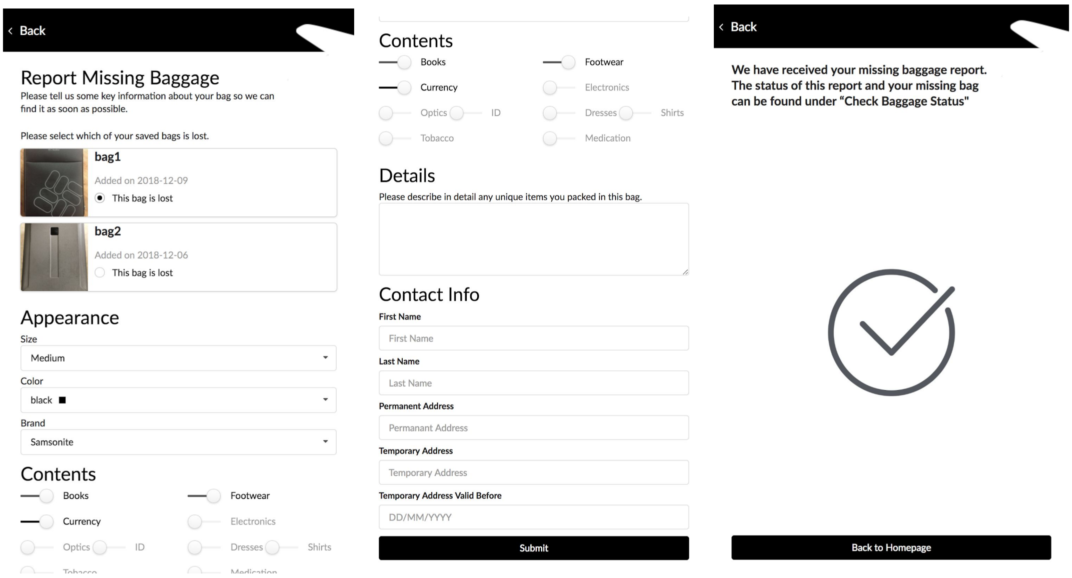
Please give a name for this bag.

bag2

Figure 2: Baggage Management (left) and Upload Bag Image (right) Interfaces

3.1.3 Report Missing Baggages

The main purpose of the application is to help improve the process of recovering missing bags, so reporting a missing bag serves as a crucial functionality. In this interface, it is important to guide users clearly through the information they must provide without overwhelming them. The interface thus provides a mix of dropdowns, checkboxes, and input fields for users to fill in the different kinds of information required when submitting a missing bag report. Figure 3 shows sections of the missing bag report.



Please tell us some key information about your bag so we can find it as soon as possible.

Please select which of your saved bags is lost.

bag1
Added on 2018-12-09
● This bag is lost

bag2
Added on 2018-12-06
○ This bag is lost

Size: Medium
Color: black ■
Brand: Samsonite

Please describe in detail any unique items you packed in this bag.

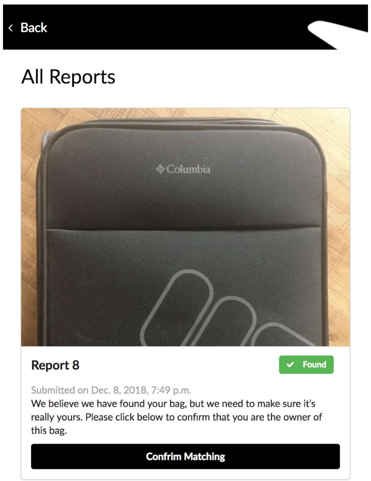
First Name: First Name
Last Name: Last Name
Permanent Address: Permanant Address
Temporary Address: Temporary Address
Temporary Address Valid Before: DD/MM/YYYY

We have received your missing baggage report. The status of this report and your missing bag can be found under “Check Baggage Status"

Figure 3: Sections of the Missing Bag Report

3.1.4 Check Baggage Status

When an employee matches a bag with a report, the customer is notified and requested that they confirm their ownership of the bag. This is done using a captcha-like system where the customer is presented with multiple images and they must select which images are of items that belong to them. They do not see how many items they got correct or incorrect which prevents malicious actors from exploiting the system and incorrectly confirming matches. Regardless of whether they selected the correct or incorrect pictures, the customer is required to provide their contact information. This way, if the employee determines that this customer is the proper owner, they will contact the customer and arrange for the bag to be returned. This functionality is shown in Figure 4.

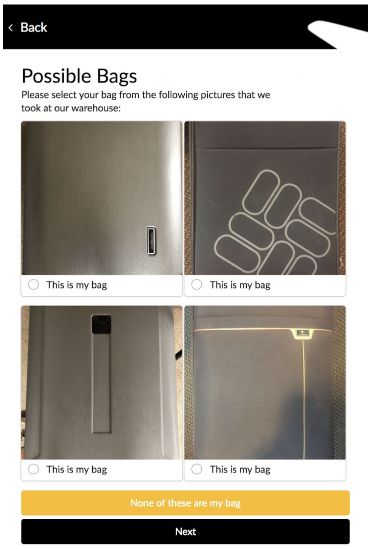


All Reports

Report 8

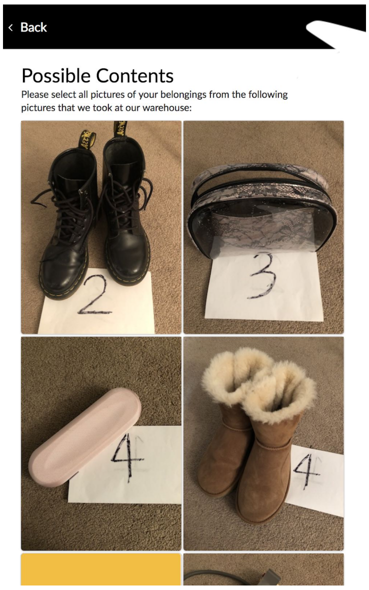
Submitted on Dec. 8, 2018, 7:49 p.m.

We believe we have found your bag, but we need to make sure it's really yours. Please click below to confirm that you are the owner of this bag.



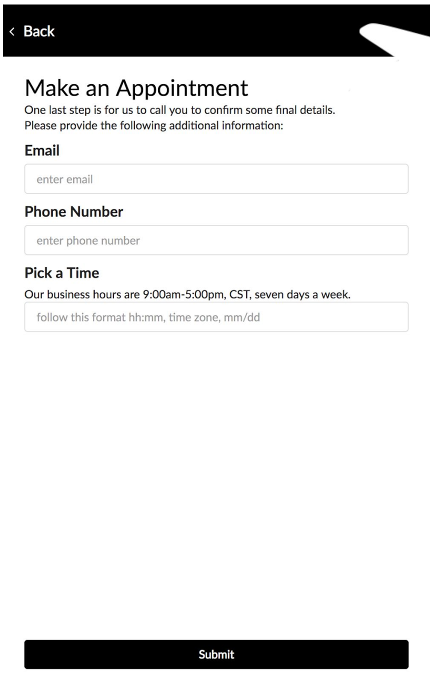
Possible Bags

Please select your bag from the following pictures that we took at our warehouse:



Possible Contents

Please select all pictures of your belongings from the following pictures that we took at our warehouse:



Make an Appointment

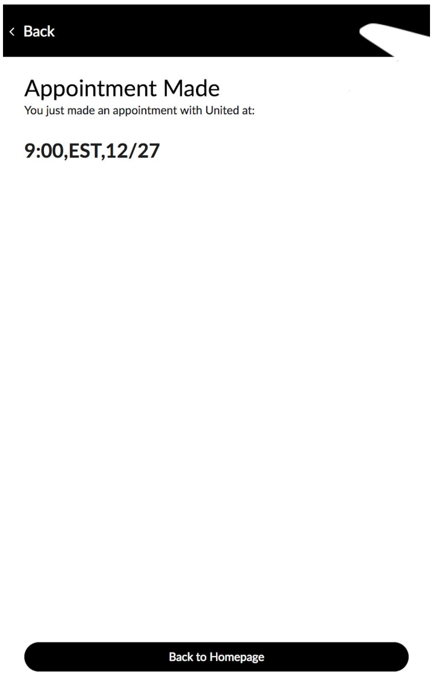
One last step is for us to call you to confirm some final details.
Please provide the following additional information:

Email

Phone Number

Pick a Time

Our business hours are 9:00am-5:00pm, CST, seven days a week.



Appointment Made

You just made an appointment with United at:

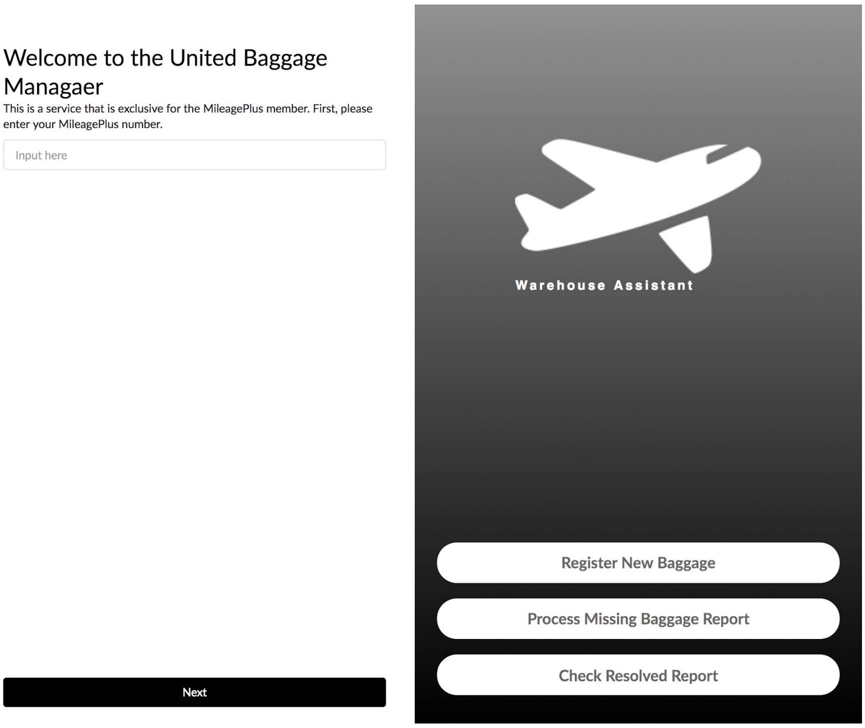
9:00,EST,12/27

Figure 4: Customer Baggage Confirmation interface

3.2 Warehouse Features

3.2.1 Warehouse Login

For the login and landing pages, the user sees the possible options for the processes they will follow in a clear and minimal way. These landing pages also prominently display an airplane icon and utilize a black color scheme as a way to brand the application. These landing pages can be seen in Figure 5 below.



Welcome to the United Baggage Managaer

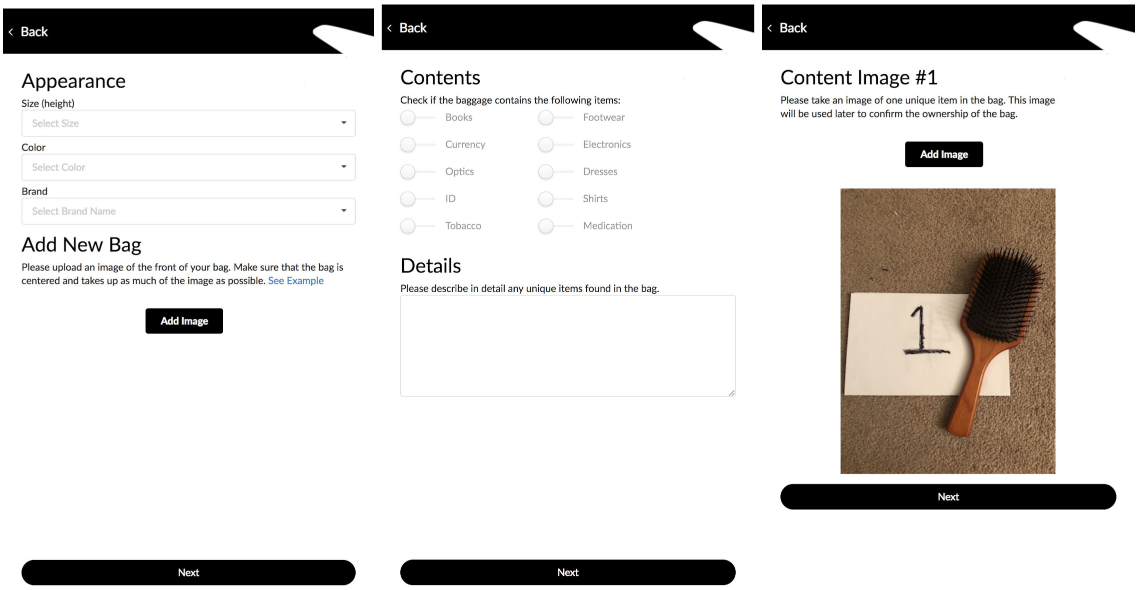
This is a service that is exclusive for the MileagePlus member. First, please enter your MileagePlus number.

Input here

Figure 5: Warehouse login (left) and Warehouse Landing Page(right) Interfaces

3.2.2 Register New Baggage

When a new bag arrives at the warehouse, it must be registered to the system by cataloging its appearance and contents. This component of the employee interface was designed similarly to the missing bag report component in the customer interface. Dropdowns are used for fields with limited values and the employees are prompted to upload images where necessary. Figure 6 depicts this component of the system.



Please upload an image of the front of your bag. Make sure that the bag is centered and takes up as much of the image as possible. See Example

Check if the baggage contains the following items:

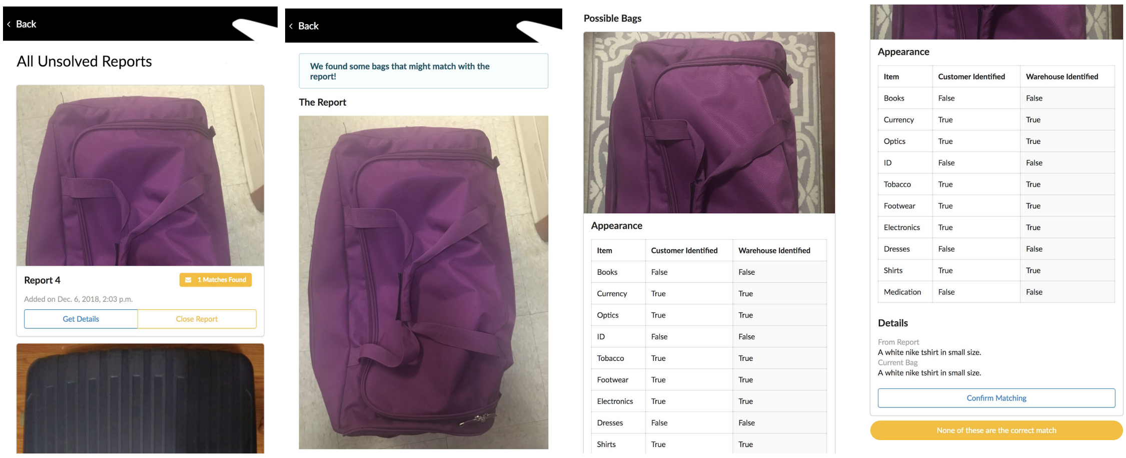
Please describe in detail any unique items found in the bag.

Please take an image of one unique item in the bag. This image will be used later to confirm the ownership of the bag.

Figure 6: Sections of the Baggage Registration

3.2.3 Process Missing Baggage Reports

On the employee end of the application, the most important function is to match bags to reports and vice versa. It was thus necessary to focus the design around the images themselves so that their employees can clearly see the possible matches to compare them. The employees are also shown a list of the content of the bag and its possible match in a way they can quickly scan and compare the contents of the two bags. Images of these interfaces are shown in Figure 7 below.



Added on Dec. 6, 2018, 2:03 p.m.

We found some bags that might match with the report!

Item	Customer Identified	Warehouse Identified
Books	False	False
Currency	True	True
Optics	True	True
ID	False	False
Tobacco	True	True
Footwear	True	True
Electronics	True	True
Dresses	False	False
Shirts	True	True

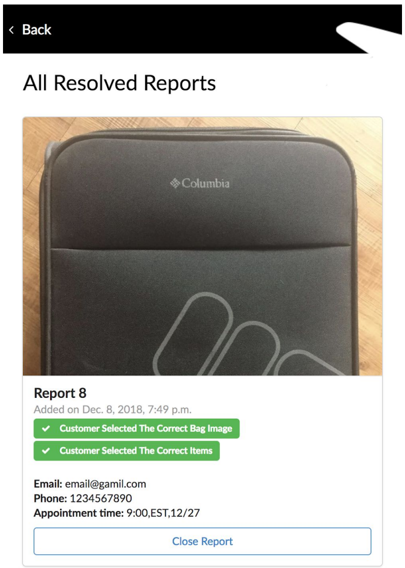
Item	Customer Identified	Warehouse Identified
Books	False	False
Currency	True	True
Optics	True	True
ID	False	False
Tobacco	True	True
Footwear	True	True
Electronics	True	True
Dresses	False	False
Shirts	True	True
Medication	False	False

From Report
A white nike tshirt in small size.
Current Bag
A white nike tshirt in small size.

Figure 7: Components of Employee Baggage Matching

3.2.4 Register New Baggage

A similar mechanism is used when an employee confirms the user's ownership of the bag. Here, the employee can see how many images the customer properly matched to confirm whether or not they are the true owner of the bag. Images of these interfaces are shown in Figure 8 below.



All Resolved Reports

Report 8

Added on Dec. 8, 2018, 7:49 p.m.

✓ Customer Selected The Correct Bag Image

✓ Customer Selected The Correct Items

Email: email@gamil.com
Phone: 1234567890
Appointment time: 9:00,EST,12/27

Figure 7: Employee Baggage Confrimation

4. Requirements

4.1 Hardware Requirements

Once the application is deployed, warehouse employees will need hardware which has the capacity to take pictures and upload them to the application via a web browser or mobile application. This hardware can be a smartphone, tablet, or laptop. For the development of the application, the team used laptops and smartphones to develop and test the application.

4.2 Software Requirements

The software requirements for the web application were established as follows:

- Language: Python 3.6
- Web application framework: Zepto, Javascript library
- DBMS: PostgreSQL
- OS: Mac OS, Windows
- Web Server: Django 2.0.3

The image matching technologies were implemented using a Python library called OpenCV. Since Python was being used for this functionality, the team decided to extend Python to the rest of the application and use a Django server to run it. The reasons behind the decision to use PostgreSQL are outlined section 4.3 Database Requirements.

4.3 Database Requirements

This application requires the storage of image addresses, customer details, and baggage details on the backend-side of the application. The following section further details these database requirements in terms of client and security and performance requirements.

Client Requirements

While the client did not directly specify what database the team should use, the choice of database was tied to the process requirements established by the client. Since the application needed to store information about customers and their bags, PostgreSQL--an object-relational database management system--was chosen for the database.

One of the advantages of PostgreSQL is that it is free and open-source, released under the terms of the PostgreSQL License. It is ACID-compliant, transactional, and provides updatable and materialized views, triggers, and foreign keys. It also supports stored procedures and functions, with emphasis on extensibility. Finally, PostgreSQL works with minimum hardware configurations and operating systems and is compatible with all major web browsers.

Security and Performance Requirements
Since the database stores customer information, the security of the database system is crucial because United is obligated to protect the privacy of each customer. Customers do not have any sort of direct access to the database through the application, mitigating the concern of information theft from the database. Furthermore, the database was built using the Django web-framework which is considered to be one of the most reliable and safe server frameworks.

For each customer, at least four images are stored: the customer image, the warehouse image, and two content images. Since this application is intended to be of widespread use for United customers, the number of images being stored will be in the thousands. As such, the application needed to keep performance in mind. Going through and trying to match each customer image with every possible warehouse image would be computationally intense and take a lot of time, causing any process advantages of the application to be lost. To combat this, the application also stores keyword information about each bag so that the images are narrowed down by things like color, brand, and size before any image matching even takes place. This made it so the image matching itself was more efficient.

4.4 Performance Requirements

Performance should not be an issue because all of our server queries involve small pieces of data. Changing screens will require very little computation and thus will occur very quickly. Server updates should only take a few seconds as long as the phone can maintain a steady signal. The baggage matching algorithms used in server side will be kind of cost time, however, it runs at background so the user will not experience all the time cost.

4.5 Security Requirements

Baggage Recovery Application will not affect data stored outside of its servers nor will it affect any other applications installed on the user's phone. It cannot cause any damage to the phone or its internal components.

5. Test Plan

The application is tested by matching several kinds of baggages. Like carry-on, small and medium-size baggages with different color and different size. This is to test the accuracy of the image matching algorithm.

6. Project Timeline

The red tasks indicate software management documentation.

Task	Deadline	Sprint
System Requirements	10/15	10/7-10/21
Customer-End Interface Design	10/21	10/7-10/21
Database Analysis	10/22	10/22-11/7
Project Plan	10/26	10/22-11/7
Database & Server Setup/Implementation	10/28	10/22-11/7
Warehouse-End Interface Design	11/7	10/22-11/7
Image Matching Functionality	11/7	10/22-11/7
Software Engineering	11/8	11/8-11/21
Client-Side Application (front and back-end)	11/21	11/8-11/21
Image Matching between Bags	11/21	11/8-11/21
Draft of Marketing Plan	11/21	11/8-11/21
Warehouse-side Application (front and back-end)	12/1	11/22-12/1
Final Development	12/8	11/22-12/8
Project Turnover	12/8	12/1-12/8