Use Case - Zapato

11.27.2017

CECS 491A Software Engineering Project I

Group D:

Agustin Barajas   Nhi Nguyen   Yixuan leng   Zhipeng Mei
Benjamin Castillo   An Jin Gang   Jimmy Nguyen

California State University Long Beach
Department of Computer Engineering and Computer Science

Table of Contents

1. System

1.1 New bid

Use case: New bid - push notification

Actor: Application

Goal in context: Notify user to take action.
Pre-Condition:
1. User puts the item on sale.
2. Zapato user has to have internet connected Android.
3. Zapato Android background service has to have not been killed by Android user (e.g. Samsung or HTC or Huawei smartphone having too much battery drain so user elects to use process killer to kill services that seem unnecessary to conserve power out of frustration (without knowing some apps need them e.g. Zapato) and turns out Zapato background service was also killed so the push service from Zapato server fails to communicate). (Because bidding is time critical user will start to complain if the push notification is not reliable, so give hint to user if they don't receive push notification for some reason for troubleshooting app problem)
4. Item has few hours or few minutes left before bid expiration (e.g. in physical bidding room the announcer says going once… going twice… sold!).
5. If bidding has low participation counts (so the profit for the seller might be too low), then check people interested in buying similar item and notify them even if they have not actively searched for the item.

Post-Condition: Alert message received in the system.
Android transmits acknowledged signal to server and server receives the signal and clears the alert.
Events:
1. Zapato Head Honcho Android push service is activated through google play services provided with Google developer accounts services activation code.
2. OAuth google account verification for Zapato app.

1.2 New message

Use case: New message - push notification

Actor: Application

Goal in context: Notify user to take action.
Pre-Condition: User's contact method is publicly displayed.
Similar to (New bid - push notification) Zapato Android background service must be on.
As soon as user Android connects to stable internet email like messages (not sms like messages) within the Zapato system prepares to sync up to last ~20 messages (sent/received)
Post-Condition:
1. Text message received in the system. Server side generate hash for current message counts and data size. Android side generate hash for current message counts and data size.
2. Android and server compare the hash (sub 20 Kbyte data transmission SHA256 or similar) to ensure the message (data packet parcel) are synced.
Events:
1. The user's device received messages from other devices.
2. System send the alert to display screen.
3. Zapato Head Honcho Android push service is activated through google play services provided with Google developer accounts services activation code.
4. OAuth google account verification for Zapato app.

2. Peer 2 Peer

2.1 Message Seller

Use case: Message Seller

Actor: User
Goal in context: To contact the Seller by sending messages in the app.
Pre-Condition: User login in, and User is in the view seller profile when user are in the product detail screen. User has intent to use the Zapato app and keep browsing information or purchase or user is just checking out Zapato app or user just wants to kill his/her time logging into random apps.
Post-Condition: User login in, and User is in the pop-up window when the user are in the Bid List and click on the bid product. User data syncs up. App internet communication protection is verified for purchase and sales.
Events:
    1. Click on "Message Seller" option from the seller profile screen
         a. If the user does not login in, it will pop out a login in box
         b.
    2. Message is written and message is temporarily saved.
    3. Message is sent.
    4. User sends the message
    5. User receives reply message about the message from the seller
    6. User receives reply from Zapato admins if seller does not reply for 3 or more days.

2.2 Message buyer

Use case: Message Buyer

Actor: User
Goal in context: To contact the Seller by sending messages in the app.
Pre-Condition: User login in, and User is in the pop-up window when the user are in the Bid List and click on the bid product. User has intent to send message about the question or the shipping (failed/successful) or inquire about negotiation of price adjustment.
Post-Condition: Original message gets copied to user's "sent" message box. Message reaches the seller that the user elected to send message to.
Events:
1. Seller has received messages about the item he/she/group sold
2. Seller has decided what to do for the inquiry from the buyer and writes the decision in the message
3. Message is written and message is temporarily saved.
4. Message is sent.
5. After user receives the decision and have made decision about the seller's decision (e.g. accept/decline/report to admins) user sends message back to seller. If no more issue or no more reply then can implicitly and by inspection imply case is resolved.
6. Back and forth message from buyer and seller until k'th (K=1,2,3,...) transmission where problem is resolved or seller is reported to Zapato admins.

2.3 Report a user

Use case: Report a user

Actor: User
Goal in context: To send a complaint against the person to the admins
Pre-Condition: User is in the profile of the person. Admins must not be corrupt to be willing to investigate and resolve Zapato service customer disputes. If Zapato service gets reputation for slow customer service user will likely not try to use this feature as time goes on.
Post-Condition: A complaint is sent about the person to the admins
Events:
   1. Scroll down from the person's profile
   2. Click on "Report"
   3. Click on the list of available offenses.
   4. Write optional sentences that happened to report this person
   5. Click "Submit"

Extensions: 2a. If not logged in 2b. Taken to login screen

3. Product Browsing

3.1 Bid a product

Use case: Bid Product

Actor: User
Goal in context: User want to place a bid for a product for a low price
Pre-Condition: User login in his/her account, and user are in the product detail screen. Or the user login in his/her account, and user are in the Bidlist screen (under ViewBid)
Post-Condition: User does not login in, and the user are in the product detail screen or Bidlist screen.
Events:
   1. From the product detail screen, click on the "Bid" button.
   2. Pop out a check Box of "Agree to payment obligation", Click "Yes" to continue.
   3. User enter a valid bid amount
   4. User Click "Submit" for finishing bid product

Extensions:

3.2 Buy product

Use case: Buy Product

Actor: User
Goal in context: User want to buy the product now.
Pre-Condition: User login in his/her account, and user are in the product detail screen.
Post-Condition: User does not login in, and user are in the product detail screen.
Events:
   1. From the product detail screen, click "Buy Now" button.
   2. User input his/her shipping address, or user uses the save address.
   3. user login in his/her Paypal account
   4. User choose a linked bank account, or user input the card information.
   5. User click "Confirm" to confirm purchases.

Extensions:

3.3 Sell product

Use case: Sell Product

Actor: User
Goal in context: User wants to sell a product
Pre-Condition: User has registered account and User has logged in.
Post-Condition: User creates a listing to sell their product
Events:
  1. From any screen, User taps and holds the home button on the lower left corner
  2. User swipes up to the camera icon
  3. User takes photo(s) or selects photo(s) of product
  4. User creates product listing title and product description
  5. User selects product condition
  6. User selects to put product for sale or bid
  7. User sets the price for sale/bid
  8. User inputs paypal email
  9. User selects to creates sell Product listing

Extensions:

3.4 Search product

Use case: Search Product

Actor: User
Goal in context: User has searched for a product they were looking for
Pre-Condition: User has to be at the main screen
Post-Condition: User is seeing their product that they searched
Events:
  1) From the main screen, user taps and drags the screen down.
  2) With their finger still on the phone, they drag their finger to the left and let go.
  3) This will show a search bar and a keyboard where the user can type what they want to search.
  4) After confirming their typed word(s), they tap enter and their product(s) will appear.

Extensions:

3.5 Filter products

Use case: Filter Products

Actor: User
Goal in context: User has filtered the products to their preference
Pre-Condition: User has to be at the main screen
Post-Condition: User is seeing their filtered products
Events:
   1) From the main screen, user taps and drags the screen down.
   2) With their finger still on the phone, they drag their finger to the right and let go.
   3) This will take them to a page with pre-determined categories.
   4) Tapping on one will show them products that only belong in that category

Extensions:

3.6 Add to favorites

Use case: Add to Favorites

Actor: User
Goal in context: User wants to add a product to their Favorites list
Pre-Condition: Product is not currently in Favorites list and User is at main screen
Post-Condition: User can see the product in the Favorites list
Events:
   1. User browse for a product
   2. Selects a product for detailed information
   3. User selects the add to Favorites icon
   4. Product is added to favorites list

Extensions:

4. Payment Processing

4.1 Payment method

Use case: Choose payment and confirm order - credit card ({VISA, Master}, {Paypal, Venmo})

Actor: User
Goal in context: Securely make money transfer to seller for the item one purchases.
Pre-Condition:
1. User has not chosen default payment method or first time purchasing with Zapato.
2. User has intent to change payment method for the purchase.
3. User has intent to make purchase and has moved into proceed to purchase.
4. User is debating whether to buy or not and checking out how the shipping fees and taxes will add up.
5. User wants to add and see the promo code they have just before purchase and debating whether to use it now or later.
6. User wants to check gift card balance and debate whether to use gift card or not.
7. User wants to add gift card during check out.

Post-Condition:
1. User presses confirm purchase detail and makes purchase
2. Seller-buyer payment is successful and the item is confirmed for shipping.
3. Payment and order details are invoiced with invoice identification number or code from Zapato and sent to user's Zapato purchase history and email to user's preferred contact method (if phone messaging service is to be added, send the purchase detail there too)
4. VISA, MasterCard confirmation gets emailed.
5. Paypal, Venmo, etc. third party payment processing confirmation gets emailed.
6. Shipping details are emailed.

Events:
1. Credit card input fields are encrypted for transmission to payment processing entities.
2. Credit card is successful or declined.
3. Third party payment method authentication is successful or declined.
4. User chooses to modify or cancel order.
5. Message is sent to buyer and seller for the order.

Extensions:
1. Modify order
2. Cancel order
3. Stop payment
4. Baby or toddler accidental smartphone purchase guard layer for unwanted purchases for people with children or babysitting children.

4.2 Shipping address

Use case: Choose Shipping Address

Actor: User
Goal in context: Confirm the shipping destination
Pre-Condition:
1. User has not chosen a default shipping address.
2. User intended to change shipping address
3. User has intent to add more shipping address

Post-Condition:
1. New shipping address is successfully added to user's account
2. The chosen shipping address carries throughout the current purchase session.

Events:
1. Address is added
2. Address is deleted
3. Address is modified

Extensions:
1. APO military address instructions and agreements and statements whether seller ships to APO address.

5. Profile

5.1 Login

Use case: Login to account from Start Screen

Actor: User
Goal in context: Login to registered Zapato account
Pre-Condition: User has a Zapato account, is not logged in yet, and is at the Start Screen.
Post-Condition: User has logged into account, and is at Main Screen
Events:
1. Click on "Login" option from Start Screen
2. Enter email and password
3. Confirm login information
4. Click "OK" to continue
 	
Extensions:
    1. Registration verification (e.g. click emailed verification link).
    2. Registration captcha to guard robot registration.

5.2 Log out

Use case: Log out
Actor: User **Goal in context:** User logs out from current logged in account **Pre-Condition:** User has a Zapato account is logged in, and is at the Main Screen. **Post-Condition:** User has logged out is at the Main Screen. **Events:** 1. Click Profile 2. Click Log Out 3. Log Out prompt appears 4. Click Yes to log out **Extensions:**

5.3 Change password

Use case: Change password
Actor: User **Goal in context:** Change user password to new password **Pre-Condition:** User is logged in and at the Main Screen **Post-Condition:** User password is changed **Events:** 1. Click Profile icon 2. Click "edit profile" option 3. Click "change password" option 4. Receive 6-digit verification code through email 5. Enter verification code into specified field 6. Enter and confirm new password 7. Save Changes

Extensions
1a. User is not logged in
    1. Send user to login screen and have them log in
    2. Continue use case at step 2
5a. Incorrect/outdated verification code entered
    1. Display error message "Incorrect code entered"
    2. Click "Send new code"
    3. Continue use case at step 4

5.4 Update username

Use case: Update username

Actor: User
Goal in context: Change username associated with account
Pre-Condition: User is in Main Screen
Post-Condition: Username is changed
Events:
1. Click Profile icon
2. Click "Edit Profile"
3. Click "Change Username"
4. Enter New Username
5. Save Changes
Extensions:
   1. Can make username to be the email address (e.g. Amazon account, Ebay account, Blizzard account, etc.)
   2. If "Extensions" item 1 is met above then user account can be transferrable. Account based on user email address verification when user wants to transfer user account for prolonged account usage without creation of new account (e.g. user doesn't like the account name (for some reason) so user wants to change the account name).

5.5 Update delivery address

Use case: Update Delivery Address

Actor: User
Goal in context: Change the delivery address of the user
Pre-Condition: User is in Main Screen
Post-Condition: User's delivery address is changed
Events:
    1. Click Profile icon
    2. Click "Edit Profile"
    3. Click "Change Delivery Address"
    4. Enter New Delivery Address
    5. Save Changes
 	
Extensions:
    1. User can add multiple addresses to quickly choose when purchasing.
    2. User can choose individual address to ship to individual item to individual locations in a single transaction (given that per transaction can contain multiple items (e.g. multiple count of same item but want to send to different address for this app which has no cart)

5.6 Update email address

Use case: Update email address

Actor: User
Goal in context: Change email address of the user
Pre-Condition: User is in Main Screen
Post-Condition: User's email is changed
Events:
    1. Click Profile icon
    2. Click "Edit Profile"
    3. Click "Change Email Address"
    4. Enter new email address
    5. Save changes
Extensions:
    1. Buyer email and seller email account can be different.

5.7 Verify email address

Use case: Verify email address

Actor: User
Goal in context: Verify user input email address
Pre-Condition: User is in sign-up screen
Post-Condition: User is in Main Screen with a registered account
Events:
    1. User has inputted all of their information necessary to make an account
    2. Click "Submit"
    3. "Verify Email" screen, Text box is displayed
    4. User has to verify email with a code sent to their address
    5. User writes the code within text box
    6. Click "Submit"
    7. Email has been verified

Extensions:
3a. If User exits verify email screen
    1. User is in Main Screen
    2. Click Profile Icon
    3. Click "Verify Email" on top

5.8 View selling list

Use case: View selling list

Actor: User
Goal in context: Switch screen to view list of all products User is selling
Pre-Condition: User is in Main Screen and logged in.
Post-Condition: Favorite items screen is the current displaying screen
Events:
    1. Click profile icon
    2. Click Listing icon
    3. Click Sell List

Extensions:

5.9 Remove product from selling list

Use case: Remove sell listing

Actor: User
Goal in context: User wants to remove listing of their product for sale
Pre-Condition: User has placed product on sale and the item has not been sold yet. The User is signed in.
Post-Condition: The product put on sale has been removed from the sale listing
Events:
  1. Click profile icon
  2. Click Listing icon
  3. Click Sell List
  4. Click Remove
  5. Click Product to remove
  6. Confirmation prompt appears
  7. Click Yes to confirm removal

Extensions:

5.10 Create product in sell listing

Use case: Create sell listing

Actor: User
Goal in context: User want to create a sell listing of his/her product for sale
Pre-Condition: User login in, and User is in the Main Screen
Post-Condition: Add listing screen is the current displaying screen
Events:
  1. Click profile icon
  2. Click Listing icon
  3. Click Sell list
  4. Click Add Listing icon
  5. Input the product title
  6. Upload the product picture
  7. Input the product description
  8. Input the product detail(condition, color, brand, etc)
  9. Select sell now or bid
  10. Input the Paypal account
  11. Click submit button for finishing Create sell listing

Extensions:

5.11 View favorite list

Use case: View favorite list

Actor: User
Goal in context: User wants to switch screen to view all favorite items
Pre-Condition: User is in Main Screen. The User is logged in.
Post-Condition: Favorite items screen is the current displaying screen
Events:
1. Click Profile icon
2. Click Listing icon
3. Click Favorites List

Extensions:

5.12 Remove favorites

Use case: Remove Favorites

Actor: User
Goal in context: User wants to remove an item in their Favorites list
Pre-Condition: User is in Main Screen. The user is logged in. The user has items in the Favorites list
Post-Condition: An item is removed from the User's Favorite List
Events:
1. Click Profile
2. Click Listing
3. Click Favorites
4. Click Remove
5. Click Product to remove
6. Confirmation prompt appears
7. Click Yes to confirm removal

Extensions:

6. Settings

All push notifications are enabled by default.

6.1 Enable message notifications

> **Use case:** Enable message notifications
>
> ---
>
> **Actor:** User
> **Goal in context:** Change notification settings to turn on message notifications
> **Pre-Condition:** User in Main Screen and message notification is disabled
> **Post-Condition:** Message notifications are enabled for User
> **Events:**
> 1. Click Settings
> 2. Click Notifications
> **Extensions:**
>   1. Notifications for advertisements in the settings (e.g. subscriptions, favorite category, recommendation)

6.2 Disable message notifications

> **Use case:** Disable message notifications
>
> ---
>
> **Actor:** User
> **Goal in context:** Change notification settings to turn off message notification
> **Pre-Condition:** User in Main Screen
> **Post-Condition:** Message notifications are disabled for User
> **Events:**
> 1. Click Settings
> 2. Click Notifications
> 3. Slide Message Notifications from On to Off
>
> **Extensions:**
>   1. When message notification is switched off from setting kill Zapato Android background services from the apps system state to conserve unnecessary usage of battery in the background that will not be used after notification is chosen by user to not receive.

6.3 Change profile image

Use case: Change Profile Image

Actor: User
Goal in context: Replace the current profile image with another image
Pre-Condition: User in Main Screen
Post-Condition: Previous profile image is being replaced with another image
Events:
    1. Click Profile icon
    2. Click "Edit Profile"
    3. Click "Change Profile Image"
    4. Upload new image from device
    5. Save changes
 	
Extensions:
    6. Image size is too large, user needs to upload another one.

6.4 View help

Use case: View Help

Actor: User
Goal in context: User will see the help section of the app
Pre-Condition: User will be in the Main Screen
Post-Condition: User will be looking at the help section in the app
Events:
    1) User has to be at the Main Screen
    2) User taps and drags home icon to "profile"
    3) They click on the gear icon to head to settings.
    4) Click on "Help"

Extensions:

6.5 View terms of service / Privacy policy

Use case: View Terms of Service/Privacy Policy

Actor: User
Goal in context: User views the Terms of Service/Privacy Policy
Pre-Condition: User in Main Screen
Post-Condition: The Terms of Service/Privacy Policy is displayed
Events:
    1. Click on profile
    2. Click on setting
    3. Click on View Terms of Service/Privacy policy

Extensions:

6.6 View application information

Use case: View Application Information

Actor: User
Goal in context: User views the information of the application
Pre-Condition: User are in the Main Screen
Post-Condition: The application information is current display screen
Events:
    1. Click Profile
    2. Click Setting
    3. Click View application information

Extensions:

6.7 Send feedback

Use case: Send Feedback

Actor: User
Goal in context: User wants to send feedback for the Application
Pre-Condition: User is in the Main Screen.
Post-Condition: Feedback is sent to the application developers
Events:
    1. Click Profile
    2. Click Settings
    3. Click Feedback
    4. Prompt to type feedback appears
    5. Type in feedback and select category of feedback
    6. Click Send

Extensions: