UML DIAGRAM

Team SAME | Phase 3
Principles of Data Management

<<enumeration>> Expediency

OVERNIGHT
TWODAY
REGULAR

<<enumeration>> PackageType

ENVELOPE_SMALL
ENVELOPE_LARGE
PACKAGE_SMALL
PACKAGE_MEDIUM
PACKAGE_LARGE

User

- firstName : String
- lastName : String
- email : String

+ getFirstName() : String
+ getLastName() : String
+ getEmail() : String
+ setFirstName(firstName : String) : void
+ setLastName(lastName : String) : void
+ setEmail(email : String) : void

Util

+ getCurrentUser(request : Request) :
Map<String, Object>
+ getAddressFromRequest(request :
Request) : HashMap<String, String>
+ getLabelDataFromRequest(request :
Request) : HashMap<String, String>

CmdLineUI

- PROMPT : String

Application

- webServer : WebServer

- Application(webServer : WebServer) : void
- initialize() : void

GetPasswordRecoverRoute

- templateEngine : TemplateEngine

+ handle(request : Request, response :
Response) : Object

WebServer

+ HOME_URL : String
+ SIGNIN_URL : String
+ SIGNUP_URL : String
+ SIGNOUT_URL : String
+ FORGOT_PASSWORD_URL : String
+ TRACKING_URL : String
+ ACCOUNT_URL : String
+ CREATE_LABEL_URL : String
+ PREPAY_PACKAGE_URL : String
+ GET_ADDRESS_DATA_URL : String
+ GET_CREDIT_CARD_DATA_URL :
String
+ GET_BANK_ACCOUNT_DATA_URL :
String
+ GET_TRACKING_DATA_URL : String
+ GET_PACKAGE_DATA_URL : String
+ ADD_CREDIT_CARD_URL : String
+ ADD_ADDRESS_URL : String
+ ADD_BANK_ACCOUNT_URL : String
- templateEngine : TemplateEngine

+ initialize() : void

PostSignInRoute

- templateEngine : TemplateEngine

+ handle(request : Request,
response : Response) : Object

PostSignOutRoute

+ handle(request : Request, response :
Response) : Object

GetHomeRoute

- templateEngine : TemplateEngine

+ handle(request : Request, response :
Response) : Object

GetAccountRoute

- templateEngine : TemplateEngine

+ handle(request : Request, response :
Response) : Object

GetSignInRoute

- templateEngine : TemplateEngine

+ handle(request : Request, response :
Response) : Object

DBLiason

- JDBC_DRIVER : String
- DB_URL : String
- USER : String
- PASS : String
- connection : Connection

+ setupDB() : void
- setupPackageTable() : void
- setupCustomerTable() : void
- setupTripTable() : void
- setupCarrierTable() : void
- setupWarehouseTable() : void
- setupSpecialInfoTable() : void
- setupCustomerPhoneTable() : void
- setupCustomerBankAccountTable() : void
- setupCustomerCreditCardTable() : void
- setupPackageSpecialInfoTable() : void
- setupTripPackageTable() : void
- populateTableFromCSV(tablename : String, filename : String, reformate : String) : void
- populateTableFromCSV(tablename : String, filename : String) : void
- escapeSingleQuotes(value : String) : String
- formatCommand(formatString : String, values : String ...)
+ executeArbitrarySQLQq(sl : String) : void
- currentMaxCustomerID() : int
- getCustomerByEmail(email : String) : int
- getCustomerByAddr(addr_line1 : String, addr_line2 : String, city : String, province : String, zipcode : String, country : String) :
int
- addCustomerByAddr(addr_line1 : String, addr_line2 : String, city : String, province : String, zipcode : String, country : String) :
int
- ensureAddressExists(addr_line1 : String, addr_line2 : String, city : String, province : String, zipcode : String, country :
String) : int
+ addCustomerByInfo(lastName : String, firstName : String, email : String, password : String) : int
+ linkAddress(email : String, addr_line1 : String, addr_line2 : String, city : String, province : String, zipcode : String, country :
String) : boolean
+ linkCreditCard(email : String, card_name : String, expiration : String, cvv : String) : boolean
+ linkBankAccount(email : String, acct_num : String, routing_num : String) : boolean
+ linkPhoneNumber(email : String, phone_num : String) : boolean
+ getAddressForCustomer(email : String) : HashMap<String, String>
+ getCreditCardsForCustomer(email : String) : ArrayList<HashMap<String, String>>
+ getBankAccountForCustomer(email : String) : HashMap<String, String>
+ getPhoneNumbersForCustomer(email : String) : ArrayList<String>
+ getNameForCustomer(email : String) : HashMap<String, String>
- checkPassword(email : String, password : String) : boolean
- currentMaxPackageID() : int
- computeDaysBetween(origin_customer_id : int, dest_customer_id : int) : int
- computePrice(type : PackageType, weight_in_grams : int, expediency : Expediency) : int
- createPackage(origin_customer_id : int, dest_customer_id : int, ship_timestamp : long, expediency : Expediency, type :
PackageType, weight_in_grams : int, receiver_pays : boolean, already_paid : boolean) : void
+ scanPackage(o_addr_line1 : String, o_addr_line2 : String, o_city : String, o_province : String, o_zipcode : String, o_country
: String, d_addr_line1 : String, d_addr_line2 : String, d_city : String, d_province : String, d_zipcode : String, d_country :
String, expediency : Expediency, type : PackageType, weight_in_grams : int, receiver_pays : boolean, already_paid :
boolean) : void
+ createLabel(email : String, address1 : String, address2 : String, city : String, state : String, zip : String, country : String,
expediency : String, packageType : String, weight : String) : boolean
+ getLatePackages() : ResultSet
- prettifyResultSet(format : String, rs : ResultSet) : String
- asLines(lines : ArrayList<String>) : String
+ prettyPackageList() : String
+ prettyCustomerList() : String
+ prettyCustomerAddressList() : String
+ prettyEmailPasswordList() : String