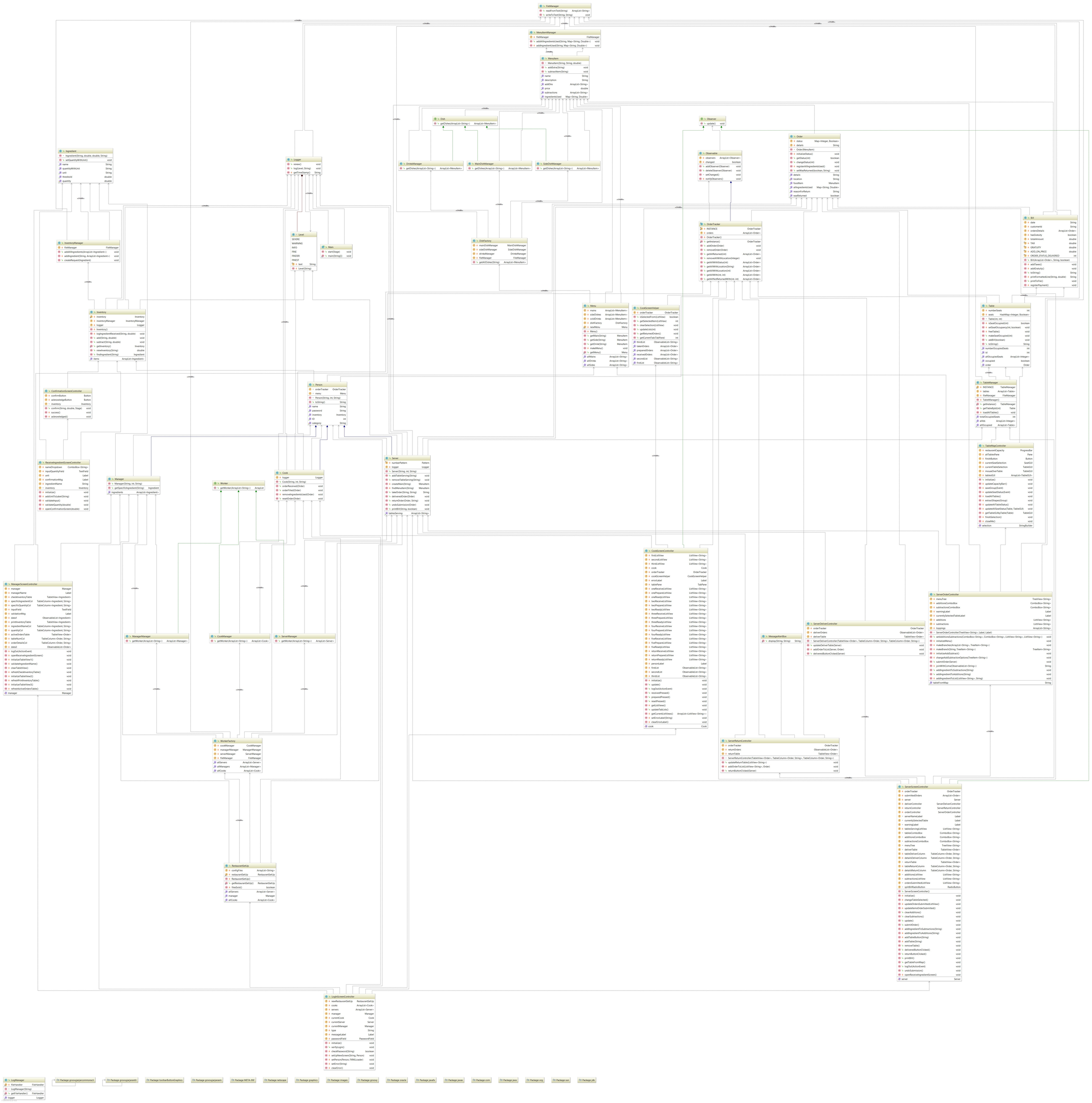
FileManager
readFromText(String) ArrayList<String>
writeToText(String, String) void
MenuItemManager
fileManager FileManager
addAllIngredientsUsed(String, Map<String, Double>) void
addIngredientUsed(String, Map<String, Double>) void
MenuItem
MenuItem(String, String, double)
addExtra(String) void
subtractItem(String) void
name String
description String
addOns ArrayList<String>
price double
subtractions ArrayList<String>
ingredientsUsed Map<String, Double>
Dish
getDishes(ArrayList<String>) ArrayList<MenuItem>
Observer
update() void
Observable
observers ArrayList<Observer>
changed boolean
addObserver(Observer) void
deleteObserver(Observer) void
setChanged() void
notifyObservers() void
Order
status Map<Integer, Boolean>
details String
Order(MenuItem)
initializeStatus() void
getStatus(int) boolean
changeStatus(int) void
registerAllIngredientsUsed() void
setWasReturned(boolean, String) void
details String
location String
foodItem MenuItem
allIngredientsUsed Map<String, Double>
reasonForReturn String
wasReturned boolean
Ingredient
Ingredient(String, double, double, String)
setQuantityWithUnit() void
name String
quantityWithUnit String
unit String
threshold double
quantity double
Logger
renew() void
log(Level, String) void
getTimeStamp() String
DrinksManager
getDishes(ArrayList<String>) ArrayList<MenuItem>
MainDishManager
getDishes(ArrayList<String>) ArrayList<MenuItem>
SideDishManager
getDishes(ArrayList<String>) ArrayList<MenuItem>
Level
SEVERE
WARNING
INFO
FINE
FINER
FINEST
text String
Level(String)
Main
start(Stage) void
main(String[]) void
DishFactory
mainDishManager MainDishManager
sideDishManager SideDishManager
drinksManager DrinksManager
fileManager FileManager
getAllDishes(String) ArrayList<MenuItem>
OrderTracker
INSTANCE OrderTracker
orders ArrayList<Order>
OrderTracker()
getInstance() OrderTracker
addOrder(Order) void
removeOrder(Order) void
getAllReturned(int) ArrayList<Order>
removeAllWithLocation(Integer) void
getAllWithStatus(int) ArrayList<Order>
getAllWithLocation(String) ArrayList<Order>
getAllWithLocation(int) ArrayList<Order>
getAllWith(int, int) ArrayList<Order>
getAllNotReturnedWith(int, int) ArrayList<Order>
Bill
date String
customerId String
orderDetails ArrayList<Order>
hasGratuity boolean
totalAmount double
TAX double
GRATUITY double
ADD_ON_PRICE double
ORDER_STATUS_DELIVERED int
Bill(ArrayList<Order>, String, boolean)
addTaxes() void
addGratuity() void
toString() String
printFormattedLine(String, double) String
printToFile() void
registerPayment() void
InventoryManager
fileManager FileManager
addAllIngredients(ArrayList<Ingredient>) void
addIngredient(String, ArrayList<Ingredient>) void
createRequest(Ingredient) void
Inventory
inventory Inventory
inventoryManager InventoryManager
logger Logger
Inventory()
logIngredientReceived(String, double) void
add(String, double) void
subtract(String, double) void
getInventory() Inventory
viewInventory(String) double
findIngredient(String) Ingredient
items ArrayList<Ingredient>
Menu
mains ArrayList<MenuItem>
sideDishes ArrayList<MenuItem>
coldDrinks ArrayList<MenuItem>
dishFactory DishFactory
newMenu Menu
Menu()
getMain(String) MenuItem
getSide(String) MenuItem
getDrink(String) MenuItem
makeMenu() void
getMenu() Menu
allMains ArrayList<String>
allDrinks ArrayList<String>
allSides ArrayList<String>
CookScreenHelper
orderTracker OrderTracker
isSelectedFromListView() boolean
getSelectedItem(ListView) int
clearSelection(ListView) void
updateLists(int) void
getReturnedOrders() void
getCurrentTabTabPane() int
thirdList ObservableList<String>
takenOrders ArrayList<Order>
preparedOrders ArrayList<Order>
receivedOrders ArrayList<Order>
secondList ObservableList<String>
firstList ObservableList<String>
Table
numberSeats int
seats HashMap<Integer, Boolean>
Table(int, int)
isSeatOccupied(int) boolean
setSeatOccupancy(int, boolean) void
freeTable() void
makeSeatOccupied(int) void
addBill(boolean) void
toString() String
numberOccupiedSeats int
id int
allOccupiedSeats ArrayList<Integer>
occupied boolean
order Order
TableManager
INSTANCE TableManager
tables ArrayList<Table>
fileManager FileManager
TableManager()
getInstance() TableManager
getTableById(int) Table
loadAllTables() void
totalOccupiedSeats int
allIds ArrayList<Integer>
allOccupied ArrayList<Table>
ConfirmationScreenController
confirmButton Button
acknowledgeButton Button
inventory Inventory
confirm(String, double, Stage) void
success() void
acknowledged() void
Person
orderTracker OrderTracker
menu Menu
Person(String, int, String)
toString() String
name String
password String
inventory Inventory
ID int
category String
Server
numberPattern Pattern
logger Logger
Server(String, int, String)
addTableServing(String) void
removeTableServing(String) void
createMain(String) MenuItem
findMenuItem(String) MenuItem
takeOrder(String, String) String
deliveredOrder(Order) void
returnOrder(Order, String) void
undoSubmission(Order) void
printBill(String, boolean) void
tablesServing ArrayList<String>
TableMapController
restaurantCapacity ProgressBar
allTablesPane Pane
finishButton Button
currentSeatSelection SeatGUI
currentTableSelection TableGUI
mouseOverTable TableGUI
tablesGUI ArrayList<TableGUI>
initialize() void
updateCapacityBarr() void
saveGroup(Event) void
updateAllStatus(Event) void
loadAllTables() void
extractShapes(Group) void
updateAllTableStatus() void
updateAllSeatStatus(Table, TableGUI) void
getTableGUIbyTable(Table) TableGUI
finishSelection() void
closeMe() void
selection StringBuilder
ReceiveIngredientScreenController
nameDropdown ComboBox<String>
inputQuantityField TextField
unit Label
confirmationMsg Label
ingredientName String
inventory Inventory
initialize() void
addUnitToLabel(String) void
validateInput() void
validateQuantity(double) void
openConfirmationScreen(double) void
Manager
Manager(String, int, String)
getSpecificIngredient(String) Ingredient
ingredients ArrayList<Ingredient>
Worker
getWorker(ArrayList<String>) ArrayList
Cook
logger Logger
Cook(String, int, String)
orderReceived(Order) void
orderFilled(Order) void
removeIngredientsUsed(Order) void
resetOrder(Order) void
CookScreenController
firstListView ListView<String>
secondListView ListView<String>
thirdListView ListView<String>
cook Cook
orderTracker OrderTracker
cookScreenHelper CookScreenHelper
errorLabel Label
tablePane TabPane
oneReceiveListView ListView<String>
onePrepareListView ListView<String>
oneReadyListView ListView<String>
twoReceiveListView ListView<String>
twoPrepareListView ListView<String>
twoReadyListView ListView<String>
threeReceiveListView ListView<String>
threePrepareListView ListView<String>
threeReadyListView ListView<String>
fourReceiveListView ListView<String>
fourPrepareListView ListView<String>
fourReadyListView ListView<String>
fiveReceiveListView ListView<String>
fivePrepareListView ListView<String>
fiveReadyListView ListView<String>
returnReceiveListView ListView<String>
returnPrepareListView ListView<String>
returnReadyListView ListView<String>
personLabel Label
firstList ObservableList<String>
secondList ObservableList<String>
thirdList ObservableList<String>
initialize() void
update() void
logOut(ActionEvent) void
receivedPressed() void
preparedPressed() void
resetPressed() void
getListViews() void
updateTabLists() void
getCurrentListViews() ArrayList<ListView<String>>
setErrorLabel(String) void
clearErrorLabel() void
cook Cook
ServerDeliverController
orderTracker OrderTracker
deliverOrders ObservableList<Order>
deliverTable TableView<Order>
ServerDeliverController(TableView<Order>, TableColumn<Order, String>, TableColumn<Order, String>)
updateDeliverTable(Server) void
addOrderToList(Server, Order) void
deliveredButtonClicked(Server) void
MessageAlertBox
display(String, String) String
ServerOrderController
menuTree TreeView<String>
additionsComboBox ComboBox<String>
subtractionsComboBox ComboBox<String>
warningLabel Label
currentlySelectedTableLabel Label
additions ListView<String>
subtractions ListView<String>
toppings ArrayList<String>
ServerOrderController(TreeView<String>, Label, Label)
setAdditionsSubtractions(ComboBox<String>, ComboBox<String>, ListView<String>, ListView<String>) void
initializeMenu() void
makeBranches(ArrayList<String>, TreeItem<String>) void
makeBranch(String, TreeItem<String>) TreeItem<String>
initializeAddSubtract() void
changeAddSubtractionOptions(TreeItem<String>) void
submitOrder(Server) void
joinWithComma(ObservableList<String>) String
addIngredientToSubtractions(String) void
addIngredientToAdditions(String) void
addIngredientToList(ListView<String>, String) void
tableFromMap String
ManagerScreenController
manager Manager
managerName Label
checkInventoryTable TableView<Ingredient>
specificIngredientCol TableColumn<Ingredient, String>
specificQuantityCol TableColumn<Ingredient, String>
inputField TextField
validationMsg Label
data1 ObservableList<Ingredient>
printInventoryTable TableView<Ingredient>
ingredientNameCol TableColumn<Ingredient, String>
quantityCol TableColumn<Ingredient, String>
activeOrdersTable TableView<Order>
tableNumCol TableColumn<Order, String>
orderDetailsCol TableColumn<Order, String>
data3 ObservableList<Order>
logOut(ActionEvent) void
openReceiveIngredientScreen() void
initializeTableView1() void
validateIngredientName() void
clearTableView() void
refreshCheckInventoryTable() void
initializeTableView2() void
refreshPrintInventoryTable() void
initializeTableView3() void
refreshActiveOrdersTable() void
manager Manager
ManagerManager
getWorker(ArrayList<String>) ArrayList<Manager>
CookManager
getWorker(ArrayList<String>) ArrayList<Cook>
ServerManager
getWorker(ArrayList<String>) ArrayList<Server>
WorkerFactory
cookManager CookManager
managerManager ManagerManager
serverManager ServerManager
fileManager FileManager
allServers ArrayList<Server>
allManagers ArrayList<Manager>
allCooks ArrayList<Cook>
ServerReturnController
orderTracker OrderTracker
returnOrders ObservableList<Order>
returnTable TableView<Order>
ServerReturnController(TableView<Order>, TableColumn<Order, String>, TableColumn<Order, String>)
updateReturnTable(ListView<String>) void
addOrderToList(ListView<String>, Order) void
returnButtonClicked(Server) void
ServerScreenController
orderTracker OrderTracker
submittedOrders ArrayList<Order>
server Server
deliverController ServerDeliverController
returnController ServerReturnController
orderController ServerOrderController
serverNameLabel Label
currentlySelectedTable Label
warningLabel Label
tablesServingListView ListView<String>
tablesComboBox ComboBox<String>
additionsComboBox ComboBox<String>
subtractionsComboBox ComboBox<String>
menuTree TreeView<String>
deliverTable TableView<Order>
tableDeliverColumn TableColumn<Order, String>
detailsDeliverColumn TableColumn<Order, String>
returnTable TableView<Order>
tableReturnColumn TableColumn<Order, String>
detailsReturnColumn TableColumn<Order, String>
additionsListView ListView<String>
subtractionsListView ListView<String>
ordersSubmittedListView ListView<String>
splitBillRadioButton RadioButton
ServerScreenController()
initialize() void
changeTableSelected() void
updateOrdersSubmittedListView() void
updateItemsOrderSubmitted() void
clearAdditions() void
clearSubtractions() void
update() void
submitOrder() void
addIngredientToSubtractions(String) void
addIngredientToAdditions(String) void
addTableButton(String) void
addTable(String) void
removeTable() void
deliveredButtonClicked() void
returnButtonClicked() void
printBill() void
getTableFromMap() void
logOutActionEvent() void
undoSubmission() void
openReceiveIngredientScreen() void
server Server
RestaurantSetUp
configFiles ArrayList<String>
restaurantSetUp RestaurantSetUp
RestaurantSetUp()
getRestaurantSetUp() RestaurantSetUp
filesExist() boolean
allServers ArrayList<Server>
manager Manager
allCooks ArrayList<Cook>
LoginScreenController
newRestaurantSetUp RestaurantSetUp
cooks ArrayList<Cook>
servers ArrayList<Server>
manager Manager
currentCook Cook
currentServer Server
currentManager Manager
type String
messageLabel Label
passwordField PasswordField
initialize() void
verifyLogin() void
checkPassword(String) boolean
setUpNewScreen(String, Person) void
setPerson(Person, FXMLLoader) void
setError(String) void
clearError() void
LogManager
fileHandler FileHandler
LogManager(String)
getFileHandler() FileHandler
logger Logger
Package groovyjarjarcommonscli
Package groovyjarjarantlr
Package toolbarButtonGraphics
Package groovyjarjarasm
Package META-INF
Package netscape
Package graphics
Package images
Package groovy
Package oracle
Package javafx
Package javax
Package com
Package java
Package org
Package sun
Package jdk