**// User Class**

Attributes:

String Name;

String StudentID;

String Username;

String Password;

Double userLocation;

**createAccount**(Name,StudentID, Username,Password)

Create an account on the app requires name, student id, a username, and a password

**changeUserName**(UserName, Password)

Change the users username

**changePassword**(Password)

Changes old password to a new password

**deleteAccount**()

Deletes users account

**sendLocation**()

Send the users location to the map class to assist in navigation

**sendDestination**()

User sends desired destination to help map class in navigation

**getDirections**()

User receives directions from map class based on users location and desired destination

**getArrivalTIme**()

Map class sends user an estimated time of arrival based on users location and directions

**getNotifications**()

User receives notifications from the notification class

**//Map Class**

Attributes:

Int distance:

Int arrivalTime:

String directions:

**getUserLocation**()

Retrieves users current location from device

**getDestination**()

User enters desired destination

**sendDirections**()

App gives directions from users location to destination

**sendArrivalTime**()

App gives an estimated arrival time based on user location and destination

**//ParkingLot Class**

Attributes:

Int parkingLotNum;

Int lotCounter;

Int spots;

getParkingLotCount()

Parking lot count increases by one as car enters lot

Parking lot count decreases by one as car leaves parking lot

sendParkingLotFull()

If parking lot has a high percentage than send full message

sendParkingLotEmpty()

If parking lot has no cars in count send empty message

SendParkingLotPercentage()

If parking lot is not full or empty divide lotCounter by spots and send message with percentage

turnParkingLotOff()

User disabels notifications for certain parking lots

turnParkingLotOn()

User enables notifications for selected parking lots

setNotificationHours()

User selects hours to receive notifications for parking lots

**//General Information Class**

Attributes:

String facilitiesText:

String studentCenterText:

String libraryText:

String cohenCafeText:

String bursarText:

**expand**()

Expands drop down to show information such as a description and hours

**compress**()

After expanding stops displaying information and shrinks back to list

**//Notification Class**

Attributes:

String NotificationTitleText:

**SendNotification**()

App sends notification to the user

**turnOffNotification**()

Disables notifications and stops the sending of the selected notification type to the user

**turnOnNotification**()

Enables notifications and will allow the sending of the selected notification type to the user

**expand**()

Opens list of sub notification types

**compress**()

Closes list of sub notification types

**//Calendar Class**

Attributes:

Int timeSpan:

Int currentDate:

**getUpcomingInfo**()

Receive information about new events from events class

**getDateTime**()

Receive date and time about upcoming event from event class

**sendUpcomingEvents**()

Compare event date and time to current date and time

Sends notifications to user about upcoming events

**// Events Class**

Attributes:

Int eventDateTime:

Spring eventName:

**sendUpcomingInfo**()

Send information about upcoming events to the calendar class

**SendDateTIme**()

Send event date and time to calendar class