

## Object Dictionary for DISTRESS

**Name:** DISTRESS

**Purpose:** This is the main entry point for the simple Distributed Real Estate Searching Service (DISTRESS). Through this people may log into DISTRESS and access their accounts. It is also a first level barrier against unwarranted access and those whose credit does not successfully charge when using a service, such as registering or adding a listing.

### Attributes:

Type	Name	Description
String	pLoginName	Users login name.
String	pLoginPassword	Users password.
Date	pDateAndTime	Date and Time of login.

### Methods:

Method	Signature	Description
login()	boolean login(String pLoginName, String pLoginPassword)	returns true if user logged in successfully. Calls lookupPerson() to check if they are already in the system, if not, calls registerNewPerson()
lookupPerson()	Person lookupPerson(String pLoginName, String pLoginPassword)	returns reference to Person object if person already in the DISTRESS system, otherwise returns null
chargeUserAccount()	boolean chargeUserAccount(double mAmount, Person mUser)	Attempts to charge user's account mAmount, using credit card details found in their Person object.
main()	main(String args[])	main() method designates main entry point to DISTRESS
getUniquePropertyID()	int getUniquePropertyID()	Unique property ID returned for use identifying/branding Property objects. If a realtor lists the same property twice, the property will have two different identifiers.

### Associations:

- a) owns EventManager (composition), which is used to manage DISTRESS system
- b) aggregation with Person, to which control is passed to once created/retrieved
- c) calls on RegisterNewPerson to create Person instances

## Object Dictionary for DISTRESS

**Name:** EventManager

**Purpose:** Manages the assets of DISTRESS, specifically containing references to all Person, Property, Mortgage and Agent objects, interconnecting them and having them access to them. Updates all events and iterates Agents through object data.

### Attributes:

Type	Name	Description
ArrayList	objPeople	list of all People objects visible to the system. For use with event updates and search/browsing.
ArrayList	objProperties	list of all Property objects visible to the system. For use with event updates and search/browsing.
ArrayList	objMortgages	list of all Mortgage objects visible to the system. For use with event updates and search/browsing.
ArrayList	objAgents	list of all Agent objects (both PropertyAgent and MortgageAgent) visible to the system. For use with event updates and search/browsing.

### Methods:

Method	Signature	Description
updateAgents()	void updateAgents()	private method called by handleEvents() that ensures all Agent objects are uptodate. If an Agent has its criteria met then an email will be sent to the Person owner of that Agent notifying that a match has been found.
handleEvents()	void handleEvents()	Traverses through objects, updating all events. Invokes updateAgents().

### Associations:

a) Is owned by DISTRESS and has associations with: b) Person, c) Email, d) Property, e) Mortgage, f) PropertiesByCriteria, and g) MortgagesByCriteria, all to which it must list manage.

## Object Dictionary for DISTRESS

**Name:** RegisterNewPerson

**Purpose:** Used to register a new person to the system. Will collect details on the person, including credit card details, and pass these as parameters to construct a new Person object if their credit is good. New consumers don't have to pay, but Realtors and Lenders do.

### Attributes:

Type	Name	Description
String	pActorType	Type of actor this new user will be, "Realtor", "Lender" or "Consumer"

### Methods:

Method	Signature	Description
registerPerson(String)	Person registerPerson(String loginName, loginPassword, pActorType)	When registering a person of actor type = "Realtor" or "Lender" then chargeUserAccount() in DISTRESS is invoked and a fee charged. If DISTRESS fails to charge the fee then the Person is not instantiated. Persons of actor type = "Consumer" are not charged a fee and are instantiated for free. All actors must register. If failed to register user then null is returned, otherwise a reference to the new object

### Associations:

- a) with DISTRESS as RegisterNewPerson is called when a Person object cannot be found to match user login.
- b) Realizes a new Person instance when a new person is successfully registered.

## Object Dictionary for DISTRESS

**Name:** Person

**Purpose:** Holds common/generic attributes for all persons in the DISTRESS system. Is extended by three other specific actor classes, which contain functionality and methods for use within the system.

### Attributes:

Type	Name	Description
String	pActorType	The type of actor this person is, "Realtor", "Lender" or "Consumer"
String	pFullName	Persons full name
int	pPhoneNumber	Persons phone number
String	pEmailAddress	Persons email address
String	pAddress	Persons physical address
String	pPassword	Persons password, *encrypted
String	pCreditCardDetails	Persons credit card details, *encrypted

### Methods:

Method	Signature	Description
showPerson()	void showPerson()	used for displaying attribute data for this person.

### Associations:

- a) owned by DISTRESS but has a weak association as it exists as a registered instance outside this users session of DISTRESS
- b) EventManager passes to Person lists of instances of Property and Mortgage for browse or search

## Object Dictionary for DISTRESS

**Name:** Realtor

**Purpose:** Extends class Person for “Realtor” type actors. Realtors main usage of the DISTRESS system is to add Property listings. This class gives them the controls they need to add, edit, delete, and show/display their Property listings. All properties they have invoked are stored as a list of objects in the attribute objMyProperties

### Attributes:

Type	Name	Description
ArrayList	objMyProperties	List of all the Property objects that this realtor has created/listed.

### Methods:

Method	Signature	Description
addProperty()	Property addProperty()	Attempts to construct a new property invoking the Property constructor, charging the realtor in the process. Returns a reference to the Property object on success, otherwise null. New properties are appended to the objMyProperties list and passed to the EventManager.
editProperty()	void editProperty(Property mProperty)	Allows access to Property object mProperty for editing
deleteProperty()	void editProperty(Property mProperty)	Deletes a Property from both the objMyProperties attributes list here, and tells the EventManager to delete its reference in objProperties list
showProperties()	void showProperties()	Displays to realtor his Properties listed, as found in the attribute list objMyProperties.

### Associations:

[Same as for Person class it extends]

+) realizes new Property instances when Realtor adds a listing.

## Object Dictionary for DISTRESS

**Name:** Lender

**Purpose:** Extends class Person for “Lender” type actors

**Attributes:**

Type	Name	Description
ArrayList	objMyMortgages	List of all the Mortgage objects that this Lender has created/listed.
String	pLenderType	Specifies this lender as either a “Mortgage broker” or “Bank representative”

**Methods:**

Method	Signature	Description
addMortgage()	Mortgage addMortgage()	Attempts to construct a new Mortgage invoking the Mortgage constructor and charging the Lender in the process. Returns a reference to the Mortgage object on success, otherwise null. New Mortgages are appended to the objMyMortgages list and passed to the EventManager.
editMortgage()	void editMortgage(Mortgage mMortgage)	Allows access to Mortgage object mMortgage for editing
deleteMortgage()	void deleteMortgage(Mortgage mMortgage)	Deletes a Mortgage from both the objMyMortgages attributes list here, and tells the EventManager to delete its reference in objMortgages list
showMortgage()	void showMortgage()	Displays to lender his Mortgages listed, as found in the attribute list objMyMortgages.

**Associations:**

[Same as for Person class it extends]

+) realizes new Mortgage instances when Lender adds a Mortgage.

## Object Dictionary for DISTRESS

**Name:** Consumer

**Purpose:** Extends class Person for “Consumer” type actors.

### Attributes:

Type	Name	Description
ArrayList	objMyProfileAgents	List of all the Agent objects that this Consumer has created.

### Methods:

Method	Signature	Description
browseProperties()	void browseProperties()	Browse all Properties listed by Agents, as found in list EventManager.objProperties
browseMortgages()	void browseMortgages()	Browse all mortgage postings listed by Lenders as found in list EventManager.objMorgages
browseDailyRates()	void browseDailyRates()	Browse daily mortgage rates
addProfileAgent()	Agent addAgent()	Attempts to construct a new Agent invoking the Agent constructor and charging the Consumer. Returns a reference to the Agent object or null. New Agents are appended to the objMyProfileAgents list.
editProfileAgent()	void editProfileAgent(Agent mAgent)	Allows access to Agent object mAgent for editing
deleteProfileAgent()	void deleteProfileAgent(Age nt mAgent)	Deletes an Agent from both the objMyProfileAgents attributes list here, and tells the EventManager to delete its reference in objAgents list
showProfileAgents()	void showProfileAgents()	Displays to consumer his Agents listed, as found in the attribute list objMyProfileAgents.
emailRequest()	void emailRequest()	Invokes Email so the consumer can requestMoreInfo() or scheduleApptmt() with a Realtor via email.

### Associations:

[Same as for Person class it extends]

+) realizes new PropertiesByCriteria instances when Consumer searches/browses Properties and +) realizes new MortgagesByCriteria instances when Consumer searches/browses Mortgages.

## Object Dictionary for DISTRESS

**Name:** Property

**Purpose:** Holds common/generic attributes for all Properties in the DISTRESS system. Is extended by three other specific classes, which contain additional attributes for specific property types.

**Attributes:**

Type	Name	Description
double	pUniqueID	Unique Identifier for this property, set when listing is created
String	pType	is either = "Condo", "Single-family" or "Multi-family"
String	pDescription	String describing the property
String	pStyle	"Ranch", "Raised ranch", "Colonial", "Cape"
int	pNumRooms	Number of rooms in property
int	pNumBedrooms	Number of bedrooms in property
int	pNumBathrooms	Number of bathrooms in property
String	pTownOrCity	Name of the town or city property is in
double	pAskingPrice	Price being asked for
String	pPhoto	Pathname of photo of property
String	pAvailableSpace	Available space on property
int	pLotSize	Number of units in the lot (n)
int	pNumParkingSpaces	The number of parking spaces on property
boolean	pGarage	true = there is a garage
String	pSpecialFeatures	String describing special features of this property

**Methods:** None

**Associations:**

a) with EventManager as it needs to have its attributes available for browse/search



## Object Dictionary for DISTRESS

**Name:** Condo

**Purpose:** Extends class Property for “Condo” types. Condos are made up of condo units, each which can have its own fee price.

**Attributes:**

Type	Name	Description
int	pUnitsInComplex	Number of condo units in complex
double[ ]	pCondoFee	Array of fee specific to each condo.

**Methods:** None

**Associations:**

[Same as for Property class it extends]

## Object Dictionary for DISTRESS

**Name:** Single-family

**Purpose:** Extends class Property “Single-family” types. This type has no obvious attributes over-and-above what is found in the base case Property.

**Attributes:** None

**Methods:** None

**Associations:**

[Same as for Property class it extends]

## Object Dictionary for DISTRESS

**Name:** Multi-family

**Purpose:** Extends class Property for “Multi-family” types. Multi-familys are made up of units units, each which can have its number of rooms and varying rent potential.

**Attributes:**

Type	Name	Description
int	pNumUnits	Number of units on property
int[ ]	pRoomsEachUnit	Array for each unit containing the number of rooms each unit has.
double[ ]	pRentEachUnit	Array for each unit containing the potential rent for that unit.

**Methods:** None

**Associations:**

[Same as for Property class it extends]

## Object Dictionary for DISTRESS

**Name:** Mortgage

**Purpose:** Holds attributes for all Mortgage loans offered by Lenders in the DISTRESS system.

**Attributes:**

Type	Name	Description
String	pLoadType	Loans are of type "Fixed rate" or "Variable rate"
int	pInterest	Interest on the loan
int	pAmount	Gross amount
String	pGeneralInfo	String of other information added by lender
int	pDownPayment	The initial downpayment
int	pTermInMonths	Term of loan in months.

**Methods:** None

**Associations:**

a) with EventManager as it needs to have its attributes available for browse/search

## Object Dictionary for DISTRESS

**Name:** PropertiesByCriteria

**Purpose:** Interfaces with objProperties in the EventManager to allow Consumer to search listings by a set of criteria. Can be extended by an Agent that checks various criteria automatically for the Consumer.

### Attributes:

Type	Name	Description
ArrayList	pSetOfTowns	Town or set of towns in which property listings are to be searched/browsed
int	pNumRooms	[optional] optimal number of rooms
int	pNumBedRooms	[optional] Minimum number of bedrooms being searched for
int	pNumBathRooms	[optional] Minimum number of bathrooms being searched for
int	pLotSize	[optional] The lot size consumer desires.
int	pAvailableSpace	[optional] The optimal available space.
double	pLowPrice	[optional] Price range, the lowest price consumer is looking for.
double	pHighPrice	[optional] Price range, the highest price consumer is looking for.

### Methods:

Method	Signature	Description
browseUsingCriteria()	void browseUsingCriteria()	Allows consumer to browse/search through Properties listed using specific criteria.

### Associations:

a) with EventManager as it needs to instigate browse/search

## Object Dictionary for DISTRESS

**Name:** Agent [Property]

**Purpose:** The Agent is a “robot” that contains a list of criteria specified by the Consumer. It searches for the criteria amongst the changing data. If the Agent finds a match then the Consumer is sent an alerting Email.

### Attributes:

Type	Name	Description
ArrayList	pCriteria	List of criteria attributes that agent is looking for amongst the Property objects in the EventManager

### Methods:

Method	Signature	Description
update()	void update()	Agent updates, checking properties for desired criteria.
fillOutProfile()	void fillOutProfile()	Consumer fills out a profile so to be notified via email when a new property is listed, or existing property changed, that matches criteria. Consumer is charged for this Agent [see Consumer.addProfileAgent() ].

### Associations:

[Same as for PropertiesByCriteria class it extends]

## Object Dictionary for DISTRESS

**Name:** MortgagesByCriteria

**Purpose:** Interfaces with objMortgages in the EventManager to allow Consumer to search loans by a set of criteria. Can be extended by an Agent that checks various criteria automatically for the Consumer.

### Attributes:

Type	Name	Description
ArrayList	pDailyRates	list of todays interest rates.
String	pLoanType	“Fixed” or “Variable”
double	pDownPayments	Down payment options being searched for
double	pInterestRate	pInterest rate being searched for
int	pTermInMonths	Term in months being looked for

### Methods:

Method	Signature	Description
viewDailyRates()	void viewDailyRates()	Allows consumer to browse/search through posted daily rates.
browseUsingCriteria()	void browseUsingCriteria()	Allows consumer to browse/search through Mortgages listed using specific criteria.

### Associations:

a) with EventManager as it needs to instigate browse/search

## Object Dictionary for DISTRESS

**Name:** Agent [Mortgage]

**Purpose:** The Agent is a “robot” that contains a list of criteria specified by the Consumer. It searches for the criteria amongst the changing data. If the Agent finds a match then the Consumer is sent an alerting Email.

### Attributes:

Type	Name	Description
ArrayList	pCriteria	List of criteria attributes that agent is looking for amongst the Mortgage objects in the EventManager

### Methods:

Method	Signature	Description
update()	void update()	Agent updates, checking mortgages for desired criteria.
fillOutProfile()	void fillOutProfile()	Consumer fills out a profile so to be notified via email when mortgage lenders post a mortgage, or an existing mortgage is changed, that matches criteria about interest rates, term (in months), type, and down payment options. Consumer is charged for this Agent [see Consumer.addProfileAgent() ].

### Associations:

[Same as for MortgagesByCriteria class it extends]



## Object Dictionary for DISTRESS

**Name:** Email

**Purpose:** Extends class Consumer

**Attributes:**

Type	Name	Description
ArrayList	pInBox	list of email message coming in
ArrayList	pOutBox	list of email messages going out

**Methods:**

Method	Signature	Description
requestMoreInfo()	void requestMoreInfo(Realtor mRealtor)	Send realtor email request for more information on a specific property.
scheduleApptmt()	void scheduleApptmt(Realtor mRealtor)	Schedule an appointment with realtor to see a property via email.
deleteEmail()	void deleteEmail(ArrayList mMail)	Delete an existing email from a mail box.

**Associations:**

[Same as for Consumer class it extends]

a) with EventManager as it needs to shuffle and send mail to other Persons