

LabelingMechanism <<abstract>>
- labelingMechanismTypeString
+ labelInstanceWithUser(UserInfo, Instance, List<ClassLabel>, Logger)

RandomLabeling
+ RandomLabeling()
+ labelInstanceWithUser(UserInfo, Instance, List<ClassLabel>, Logger): void

Instance
- id: int - instanceString - maxNumberOfLabelint - canLabelled:boolean - userLabels:List<LabeledInstance>
+ addLabel(UserInfo, List < ClassLabel>, Logger):void + getID(): int + setID(int): void + getInstance(): String + setInstance(String): void + getMaxNumberOfLabel(int):int + setMaxNumberOfLabel(int): void + isCanLabelled(): boolean + setCanLabelled(boolean):void + getUserLabels(): List<LabeledInstance> + setUserLabels(List<LabeledInstance>):void

ClassLabel
- labelID: int - labelText: String
+ getLabelID(): int + setLabelID(int): void + getLabelText(): String + setLabelText(String):void

Dataset
+ setLabels(List<Label>): void + getInstances(): List<Instance> + setInstances(List<Instance>): void
+ getDatasetID(): int + setDatasetID(int): void + getDatasetName(): String + setDatasetName(String): void + getInstanceType(): String + setInstanceType(String): void + getMaxNumberOfLabelsPerInstance(): int + setMaxNumberOfLabelsPerInstance(int): void + getLabels(): List<Label> + setLabels(List<Label>): void + getInstances(): List<Instance> + setInstances(List<Instance>): void

UserInfo
- userID: int - usernameString - userType: String - labeledInstances:List<LabeledInstance>
+ addLabeledInstance(LabeledInstance): void + getUserID(): int + setUserID(int):void + getUsername():String + setUsername(String):void + getUserType(): String + setUserType(String):void

User
- userInfos:List<User>
+ getUsers():List<User!> + setUser(List<User!>):void

LabeledInstance
- whoLabeledUserInfo - labels: List<ClassLabel> - date:LocalDateTime
+ LabeledInstance(UserInfo, List<ClassLabel>, LocalDateTime, Logger) + getWhoLabeled():UserInfo + setWhoLabeled(UserInfo):void + getLabels(): List<ClassLabel> + setLabels(List<ClassLabel>):void + getDate():LocalDateTime + setDate(LocalDateTime):void