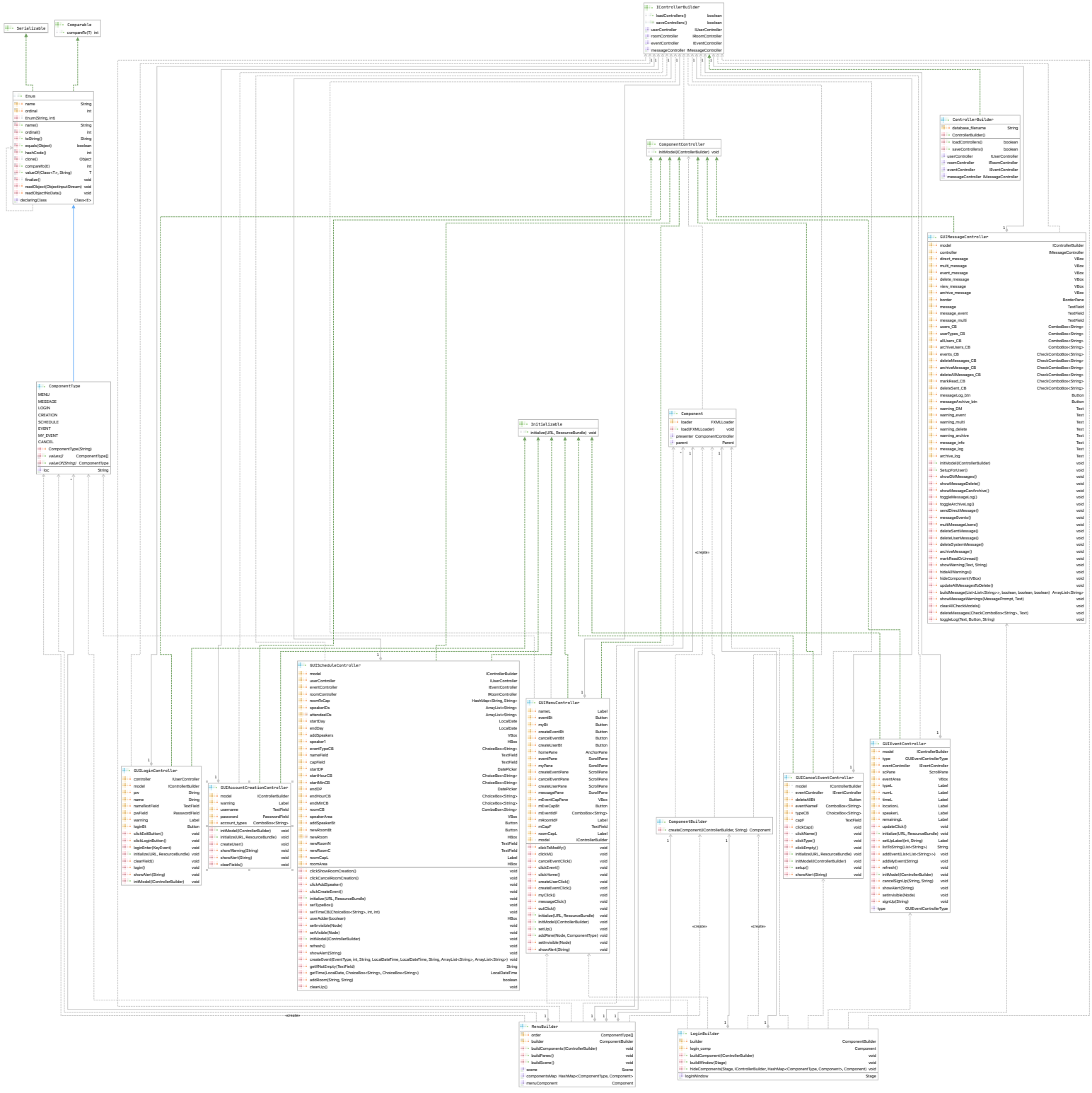
Serializable
Comparable
compareTo(T) int
Enum
name String
ordinal int
Enum(String, int)
name() String
ordinal() int
toString() String
equals(Object) boolean
hashCode() int
clone() Object
compareTo(E) int
valueOf(Class<T>, String) T
finalize() void
readObject(ObjectInputStream) void
readObjectNoData() void
declaringClass Class<E>
ComponentType
MENU
MESSAGE
LOGIN
CREATION
SCHEDULE
EVENT
MY_EVENT
CANCEL
ComponentType(String)
values() ComponentType[]
valueOf(String) ComponentType
loc String
IControllerBuilder
loadControllers() boolean
saveControllers() boolean
userController IUserController
roomController IRoomController
eventController IEventController
messageController IMessageController
ComponentController
initModel(IControllerBuilder) void
ControllerBuilder
database_filename String
ControllerBuilder()
loadControllers() boolean
saveControllers() boolean
userController IUserController
roomController IRoomController
eventController IEventController
messageController IMessageController
GUIMessageController
model IControllerBuilder
controller IMessageController
direct_message VBox
multi_message VBox
event_message VBox
delete_message VBox
view_message VBox
archive_message VBox
border BorderPane
message TextField
message_event TextField
message_multi TextField
users_CB ComboBox<String>
userTypes_CB ComboBox<String>
allUsers_CB ComboBox<String>
archiveUsers_CB ComboBox<String>
events_CB CheckComboBox<String>
deleteMessages_CB CheckComboBox<String>
archiveMessage_CB CheckComboBox<String>
deleteAllMessages_CB CheckComboBox<String>
markRead_CB CheckComboBox<String>
deleteSent_CB CheckComboBox<String>
messageLog_btn Button
messageArchive_btn Button
warning_DM Text
warning_event Text
warning_multi Text
warning_delete Text
warning_archive Text
message_info Text
message_log Text
archive_log Text
initModel(IControllerBuilder) void
SetupForUser() void
showDMMessages() void
showMessageDelete() void
showMessageCanArchive() void
toggleMessageLog() void
toggleArchiveLog() void
sendDirectMessage() void
messageEvents() void
multiMessageUsers() void
deleteSentMessage() void
deleteUserMessage() void
deleteSystemMessage() void
archiveMessage() void
markReadOrUnread() void
showWarning(Text, String) void
hideAllWarnings() void
hideComponent(VBox) void
updateAllMessagesToDelete() void
buildMessageList-List<String>-, boolean, boolean, boolean) ArrayList<String>
showMessageWarnings(MessagePrompt, Text) void
clearAllCheckModels() void
deleteMessages(CheckComboBox<String>, Text) void
toggleLog(Text, Button, String) void
Initializable
initialize(URL, ResourceBundle) void
Component
loader FXMLLoader
load(FXMLLoader) void
presenter ComponentController
parent Parent
<create>
GUIScheduleController
model IControllerBuilder
userController IUserController
eventController IEventController
roomController IRoomController
roomIDCap HashMap<String, String>
speakerIDs ArrayList<String>
attendeeIDs ArrayList<String>
startDay LocalDate
endDay LocalDate
addSpeakers VBox
speaker1 HBox
eventTypeCB ChoiceBox<String>
nameField TextField
capField TextField
startDP DatePicker
startHourCB ChoiceBox<String>
startMinCB ChoiceBox<String>
endDP DatePicker
endHourCB ChoiceBox<String>
endMinCB ChoiceBox<String>
roomCB ComboBox<String>
speakerArea VBox
addSpeakerBt Button
newRoomBt Button
newRoom HBox
newRoomN TextField
newRoomC TextField
roomCapL Label
roomArea HBox
clickShowRoomCreation() void
clickCancelRoomCreation() void
clickAddSpeaker() void
clickCreateEvent() void
initialize(URL, ResourceBundle) void
setTypeBox() void
setTimeCB(ChoiceBox<String>, int, int) void
userAdder(boolean) void
setInvisible(Node) void
setVisible(Node) void
initModel(IControllerBuilder) void
refresh() void
showAlert(String) void
createEvent(EventType, Int, String, LocalDateTime, LocalDateTime, String, ArrayList<String>, ArrayList<String>) void
getNotEmpty(TextField) String
getTime(LocalDate, ChoiceBox<String>, ChoiceBox<String>) LocalDateTime
addRoom(String, String) boolean
cleanUp() void
GUIMenuController
nameL Label
eventBt Button
myBt Button
createEventBt Button
cancelEventBt Button
createUserBt Button
homePane AnchorPane
eventPane ScrollPane
myPane ScrollPane
createEventPane ScrollPane
cancelEventPane ScrollPane
createUserPane ScrollPane
messagePane ScrollPane
mEventCapPane VBox
mEwCapBt Button
mEventIdF ComboBox<String>
mRoomIdF Label
mCapF TextField
roomCapL Label
model IControllerBuilder
clickToMobility() void
clickM() void
cancelEventClick() void
clickEvent() void
clickHome() void
createUserClick() void
createEventClick() void
myClick() void
messageClick() void
outClick() void
initialize(URL, ResourceBundle) void
initModel(IControllerBuilder) void
setUp() void
addPane(Node, ComponentType) void
setInvisible(Node) void
showAlert(String) void
GUILoginController
controller IUserController
model IControllerBuilder
pw String
name String
nameTextField TextField
pwField PasswordField
warning Label
loginBt Button
clickExitButton() void
clickLoginButton() void
loginEnterKeyEvent() void
initialize(URL, ResourceBundle) void
clearField() void
login() void
showAlert(String) void
initModel(IControllerBuilder) void
GUIAccountCreationController
model IControllerBuilder
warning Label
username TextField
password PasswordField
account_types ComboBox<String>
initModel(IControllerBuilder) void
initialize(URL, ResourceBundle) void
createUser() void
showWarning(String) void
showAlert(String) void
clearFields() void
ComponentBuilder
createComponent(IControllerBuilder, String) Component
GUICancelEventController
model IControllerBuilder
eventController IEventController
deleteAllBt Button
eventNameF ComboBox<String>
typeCB ChoiceBox<String>
capF TextField
clickCap() void
clickName() void
clickType() void
clickEmpty() void
initialize(URL, ResourceBundle) void
initModel(IControllerBuilder) void
setup() void
showAlert(String) void
GUIEventController
model IControllerBuilder
type GUIEventControllerType
eventController IEventController
scPane ScrollPane
eventAreas VBox
typeL Label
numL Label
timeL Label
locationL Label
speakerL Label
remainingL Label
updateClick() void
initialize(URL, ResourceBundle) void
setUpLabel(Int, String) Label
listToStringList-String>) String
addEventLists-List<String>>) void
addMyEvent(String) void
refresh() void
initModel(IControllerBuilder) void
cancelSignUp(String, String) void
showAlert(String) void
setInvisible(Node) void
signUp(String) void
type GUIEventControllerType
<create>
MenuBuilder
order ComponentType[]
builder ComponentBuilder
buildComponents(IControllerBuilder) void
buildPanes() void
buildScene() void
scene Scene
componentsMap HashMap<ComponentType, Component>
menuComponent Component
LoginBuilder
builder ComponentBuilder
login_comp Component
buildComponent(IControllerBuilder) void
buildWindow(Stage) void
hideComponents(Stage, IControllerBuilder, HashMap<ComponentType, Component>, Component) void
loginWindow Stage