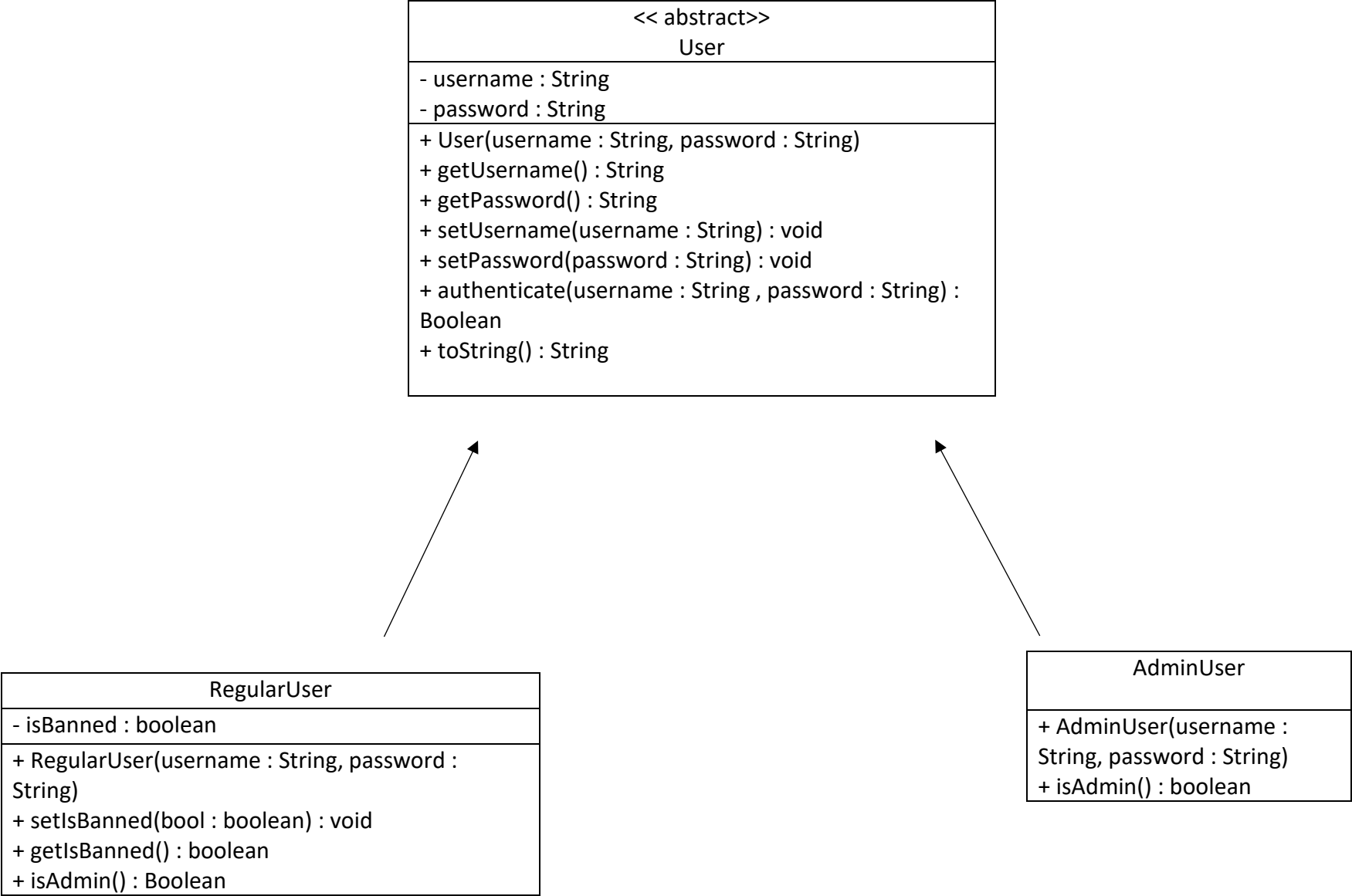


UML Model

Entities:



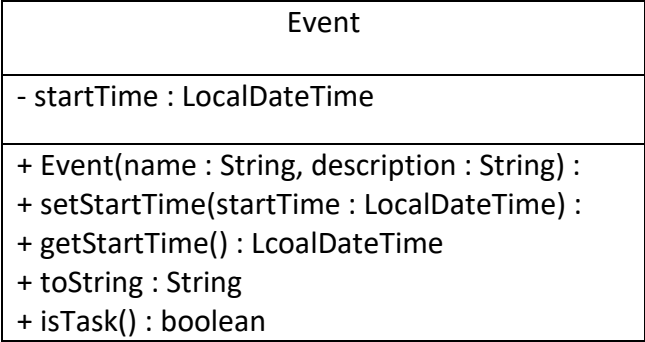
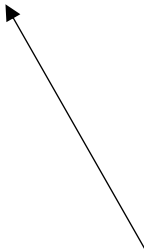
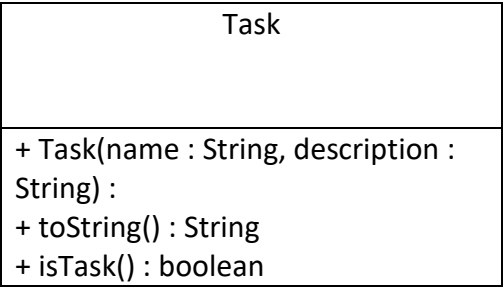
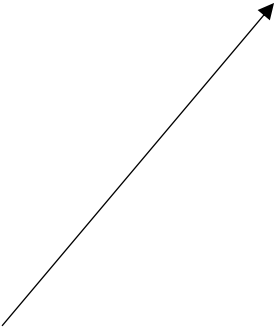
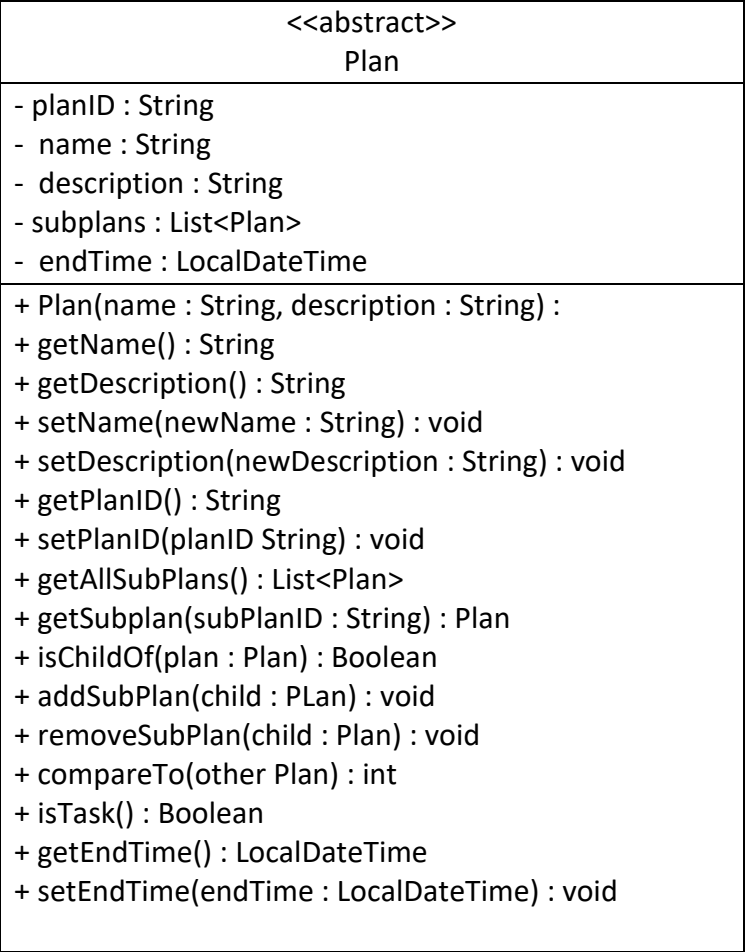
History
- action : Action(enum) - username : String - time : LocalDateTime
+ History(username : String, action : Action(enum)) + getAction() : Action(enum) + getUsername() : String + getTime() : LocalDateTime + toString() : String

Label
- name : String - isThematic : boolean
+ Label(name : String, isThematic : boolean): + getName() : String + setName(name : String) : void + isThematic() : booleam + toString() : String

Calendar
- calendarID : String - name : String - description : String
+ Calendar(name : String, description : String) : + setName(name : String) : void + setDescription(description String) : void + getName() : String + getDescription() : String + getCalendarID () : String + setCalendarID(CalendarID : String) : void + toString() : String

TimeParser
+ parseTime(time : String) : LocalDateTime - YYYYMMDDHHMM(time : String) : LocalDateTime - YYDDHHMM(time : String) : LocalDateTime - DDHHMM(time : String) : LocalDateTime - YYYYMMDDHH(time : String) : LocalDateTime - YYYYMMDD(time : String) : LocalDateTime - MMDD(time : String) : LocalDateTime - DD(time : String) : LocalDateTime - intSlice(text : String, i : int, j : int) : int - tomorrow() : LocalDateTime - nextDat(time : String) : LocalDateTime

NameDuplicatable
+ assignID(name : String, ids : Iterable<String>, username : String)



Use Case Class:

UserManager
- users : Map<String, User>
+ getUser(username : String) : User + authenticateUser(username : String, password : String) : User + createUser(username : String, password : String) : String + createAdminUser(username : String, password : String) : String + deleteUser(username : String, deleteSelf : boolean) : String + banUser(username : String) : String + unbanUser(username : String) : String - isBanned(username : String) : Boolean - addUser(newUser : User) : void

LoggingManager
- currUser : User
+ getCurrUsername() : String + userLoggedIn() : Boolean + adminUserLoggedIn() : Boolean + login(user : User) : String + logout() : String

HistoryManager
- histories : Map<LocalDateTime, History>
+ HistoryManager() + cleanHistories() : void + addHistory(username : String, action : History.Action) + getHistoriesByUser(username : String) : String + clearHistoriesByUser(username) : void

CalendarManager
- calendars : Map<String, Calendar> - currCalendarID : String
+ getCalendar(calendarID : String) : Calendar + createCalendar(name : String, description : String) : String + deleteCalendar(calendarID : String) : String + setCurrCalendarID(currCalendarID : String) : boolean + getCurrCalendarID() : String - addCalendar(c : Calendar) : void

PlanManager

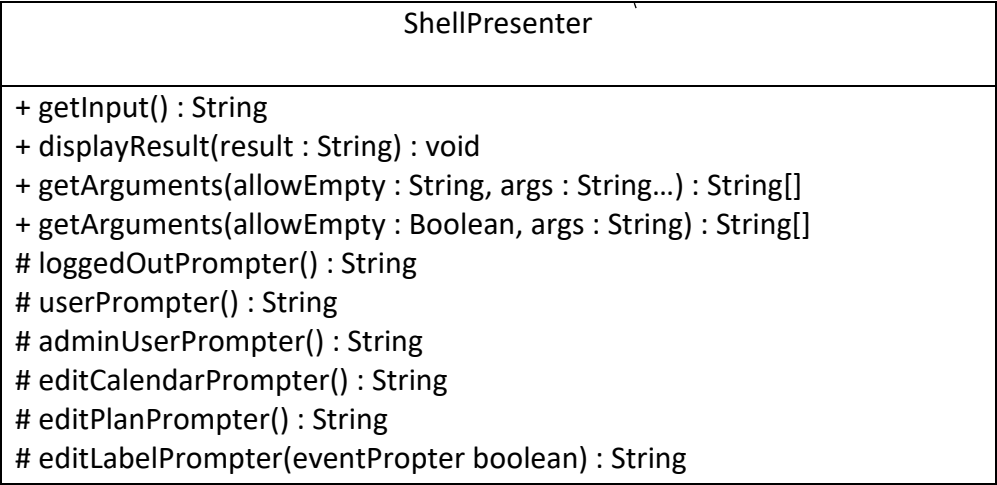
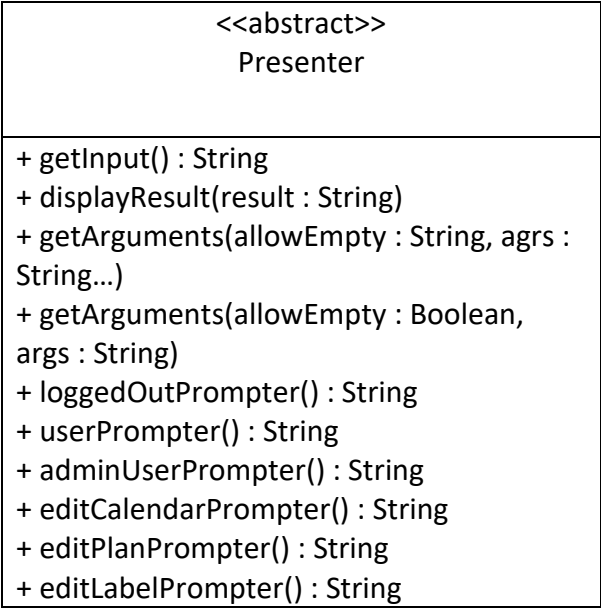
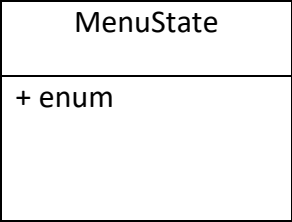
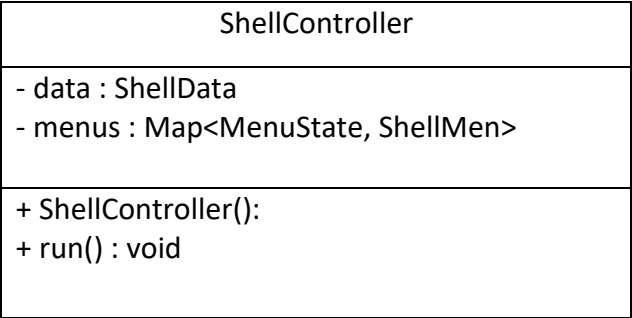
- allPlans : Map<String, List<Plan>>
- timeParser : TimeParser
- currPlanID : String

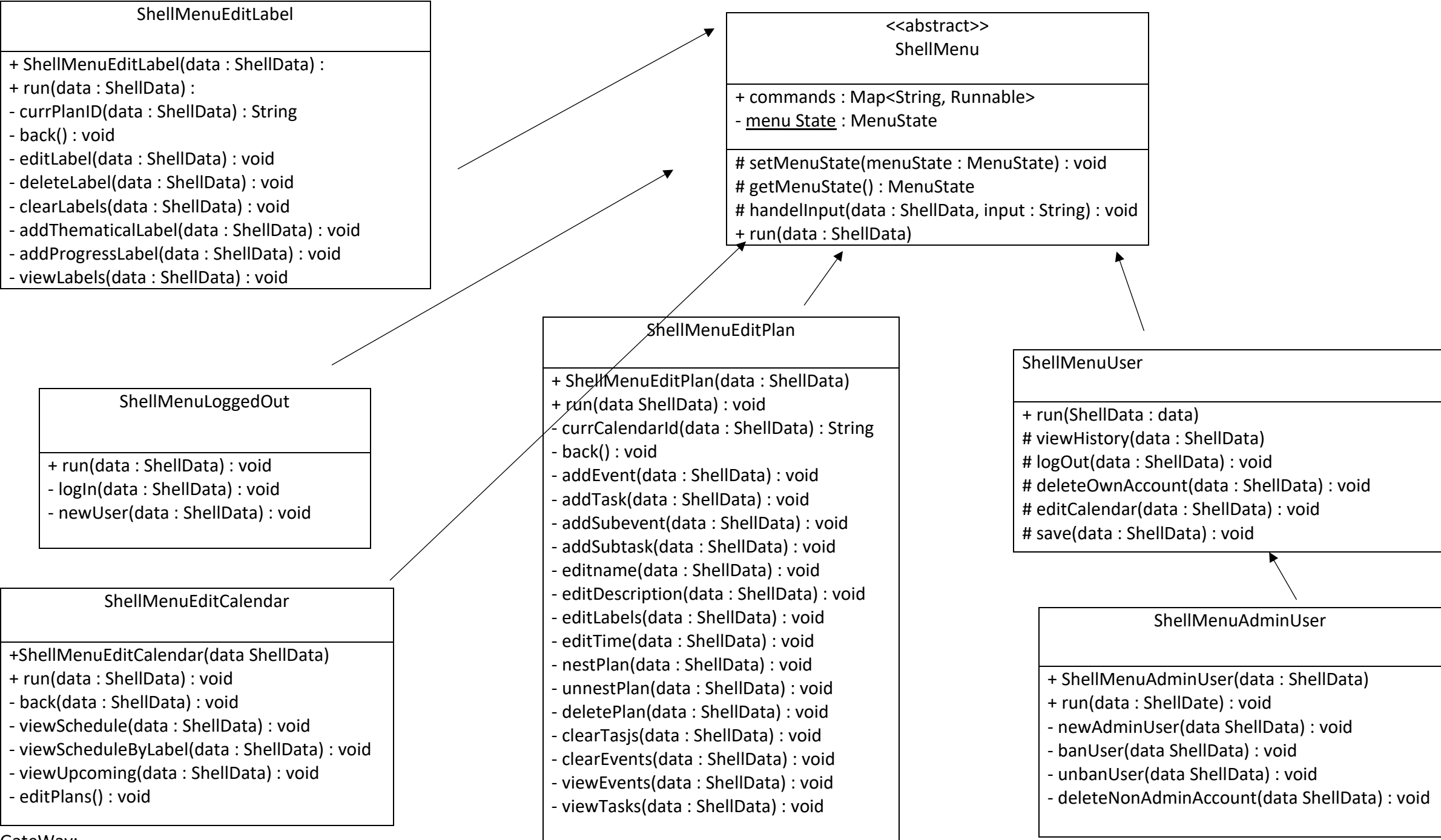
+ PlanManager() :
+ getAllPlanIDs(CalendarID : String) : List<String>
+ handlecreateCalendar(calendarID : String) : void
+ createTask(name : String, description : String, CalendarID : String) : String[]
+ createEvent(name : String, description : String, CalendarID : String) : String[]
+ getAllPlans(CalendarID String) : List<Plan>
+ getPlan(planID : String, CalendarID : String) : Plan
+ removePlan(planID : String, CalendarId : String) : String
+ changePlanName(name : String, planID : String, CalendarID : String) : String[]
+ changePlanDescription(description : String, planID : String, CalendarID : String) : String
+ nest(parentPlanID : String, subPlanID : String, CalendarID : String) : String
+ unnest(parentID : String, subPlanID : String, CalendarID : String) : String
+ clearTasks(CalendarID : String) : String
+ clearEvents(CalendarID : String) : String
+ handleclearCalendar(calendarID : String) : void
+ setCUrrPlanId(planID : String, CalendarID : String) : void
+ getCurrPlanID() : String
+ isEvent(planID : String, CalendarID : String) : Boolean
+ setTime(eventID : String, startTime : String, endTime : String, calendarID : String) : String
+ setDeadline(taskID : String, deadline : String, calendarID : String) :String
+ viewSchedule(allPlans : List<Plan>) : String
+ viewScheduleByLabel(calendarID : String, labelMap : Map<String, Map<String, Label>>, labelName : List<String>) : String
+ viewUpcoming(allPlans : List<Plan>) : String
+ viewEvents(currCalendarID : String) : String
+ viewTasks(currCalendarID : String) : String

LabelManager
- labels : Map<String, Map<String, Label>>
+ getLabels() : Map<String, Map<String, Label>> + handleNewPlan(planID : String) : void + handleDeletePlan(planID : String) : void + handleClearUserPlans(planIDs : List<String>) : void + handleSetPlanName(oldID : String, newID : String) : void + containsPlanID(planID : String) : boolean + addThematicLabel(labelName : String, planID : String) : String + addProgressLabel(labelName : String, planID : String) : String + setLabel(labelName : String, labelName2 : String, planID : String) : String + deleteLabel(labelName : String, planID : String) : String + clearPlans(planIDs : List<String>) : void + clearLabels(planID : String) : String + toString(planID : String) : String

ShellData
- users : UserManager - history : HistoryManager - logging : LoggingManager - calendar : CalendarManager - plans : PlanManager - labels : LabelManager - presenter : ShellPresenter - saver : Saver
+ ShellDate() : + getUsers() : UserManager + getHistory() : HistoryManager + getLogging() : LoggingManager + getPresenter() : ShellPresenter + getCalendar() : CalendarManager + getLabels() : LabelManager + getPlans() : PlanManager + getSaver() : Saver

Controller:





GateWay:

