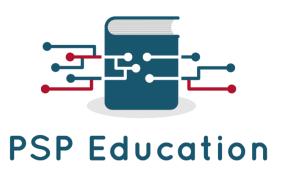
Design Document



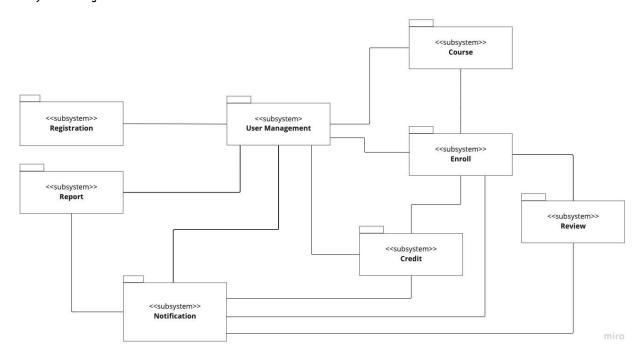
Project : WhereIsMyTutor (Group 16)

Authors and Contributions:

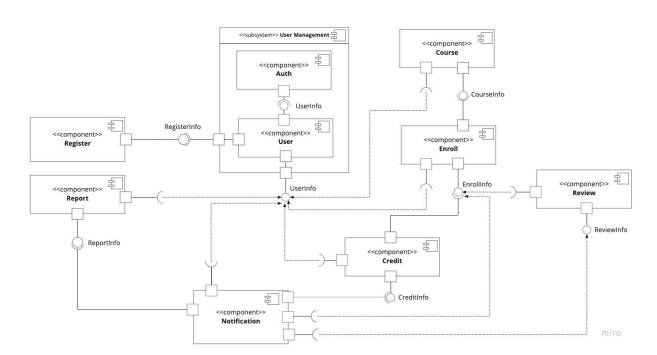
Authors	Contributions
	Deployment Diagram
6230563321	Class Diagram
Suchakree Maneepakorn	Class Diagram for Notification API
	Pseudocode
6231323221	Class Diagram
Dragorn Padungpattanodom	Class Diagram for Notification API
	Subsystem Diagram
	Component Diagram
6231325521	Class Diagram for Notification API
Thanapat Trachu	Swagger
	Subsystem Diagram
6231340921	Component Diagram
Pongsapak Pulthasthan	Class Diagram
(Leader)	Class Diagram for Notification API
6231343821	Subsystem Diagram
Peeranath Theerawatanachai	Component Diagram
6231356021	Pseudocode
Vorapon Khunakornkorbkij	Deployment Diagram
6231367921	Deployment Diagram
Sahutsawat Sivakosit	Subsystem Diagram
6231371321	Subsystem Diagram
Atibodee Champathes	Component Diagram

Date: 10/4/2022

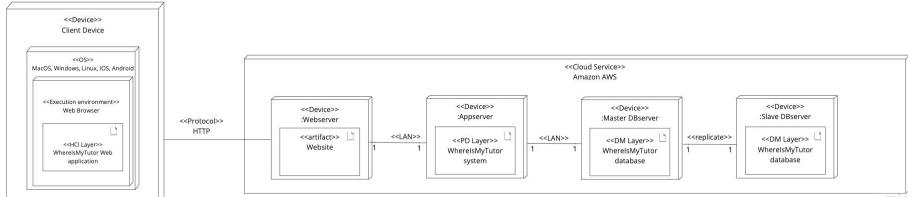
Subsystem Diagram



Component Diagram



Deployment Diagram



mirc

<<Control>> NotificationController - notificationService : NotificationService + getNotificationsByUserID(userID: string) : Notification[] + readNotification(notificationID: string): Notification <<Control>> **NotificationService** - notificationModel : Model<Notification> - userService: UserService + getNotificationsByUserID(userID: string) : Notification[] + readNotification(notificationID: string): Notification + createNotification(notificationDto: NotificationDto): Notification <<Entity>> Model<Notification> notifications: Set(Notification) 0..* <<Entity>> **Notification** - id : string - header : string - body : string - dateTimeCreated : Date - type : string - description : object - isRead : boolean - userId : string

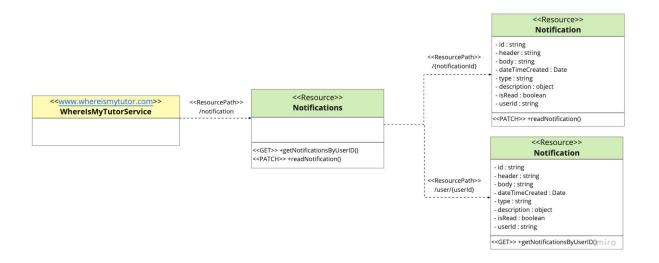
Pseudocode (NotificationService)

```
getNotificationsByUserID(userId: string) : Notification[]
         if (userService.findByld is not exist)
                   throw This userld does not exist
         notifications = notificationModel.find({'userID': userId})
         return notifications
readNotification(notificationID: string): Notification
         notification = notificationModel.find({'id': notificationID})
         notification.isRead = True
         return notification
createNotification(notificationDto: NotificationDto): Notification
         notification = new notificationModel()
         notification.header = notificationDto.header
         notification.body = notificationDto.body
         notification.type = notificationDto.type
         notification.description = notificationDto.description
         notification.userId = notificationDto.userId
         notification.isRead = False
         notification.dateTimeCreated = Date.now()
         return notification
```

Access Control Table (Notification System)

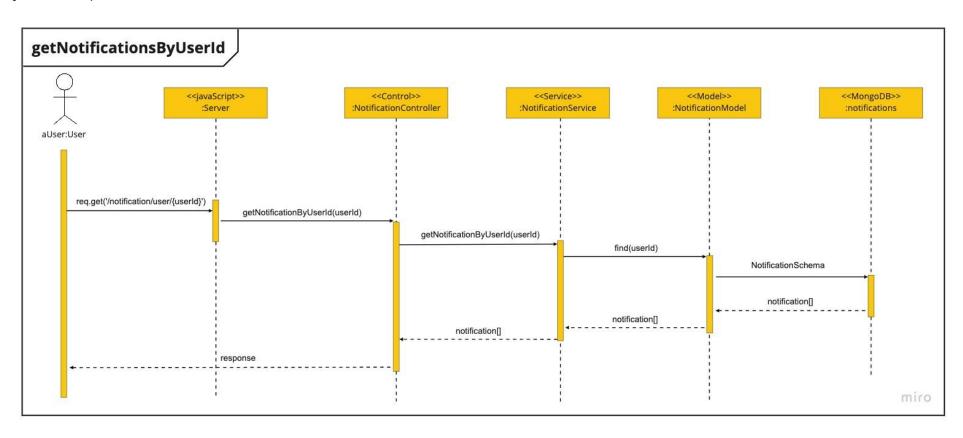
Objects Actors		Notification
User	Tutor	getNotificationsByUserID() readNotification()
	Student	
	Admin	getNotificationsByUserID() readNotification() createNotification()

Class Diagram for Notification Diagram

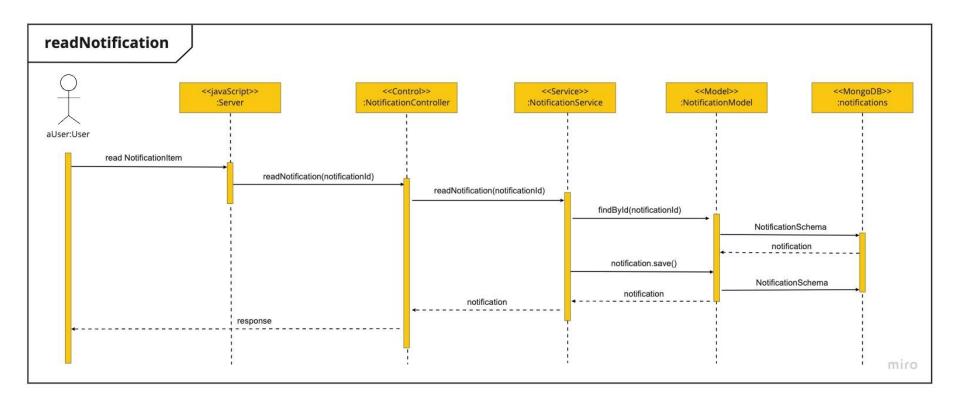


Sequence Diagram for Notification Diagram

getNotificationsByUserId



readNotifcation



Swagger

