Findings:

|  |  |
| --- | --- |
| Findings: | Fixes: |
| There’s no architectural structure design pattern | Implemented MVC design pattern |
| Names of view Controllers are very similar | This makes distinguishing between the two difficult as the only difference is the ‘s’ letter and to rectify that I renamed AlarmViewController to SettingAlarmViewController to make readability better. |
| when pressing ‘Delete’ action app was crashing | In **func** deleteAlarm(at indexPath: IndexPath)  Changed alarms.remove(at alarms.count)  To alarms.remove(at: indexPath.row) |
| There’s no access control implemented to protect private variables and functions. | Implemented access control as needed. |
| When adding new alarm the time datepicker isn’t presenting 8AM by default. | Set datepicker to 8 am if a new alarm is being added, or show the existing time if alarm is being amended. By correcting Alarm Model time property multiplication to the correct number of seconds which’s 3600 from the incorrect value of 360 |
| Alarms disappear after re-opening the app again | Implemented CoreData as a database to store alarms within the app. |
| After adding a new alarm, the new alarm is being appended at the end of the list,*and if edited alarms stay in the same position* | Implemented NS Fetched Result Controller, and used alarm time attribute as a sort descriptor to ascend alarms by time. |
| Tow constraints were conflicting in AlarmsViewController screen | Stack view probably |
| Code is not easy to read | Added comments for functions and variables, grouped and marked each relevant block of code to improve readability |
| Alarm always presents days even if it was a one-time alarm | In alarm’s **var** repeating:String {  **Return** caption.count = 1  Changed to **Return** caption.count > 1  So alarm will appear as a one-time alarm and also it will be a way to distinguish a one-time alarm when managing notifications in remove pending notifications function. |
|  |  |
|  |  |
|  |  |