Deliverable 3: Elaboration

Spring 2018
CSCI 360 - Software Architecture
Semester Project

Team:
Team 6ix

Members:
Drew Moore
Megan Landau
Holden Lucas

Table of Contents

Vision

As a team, we are going to design, model, and develop software for a dual-alarm AM/FM clock radio. We are modelling our GUI after the 1970s style of the alarm clock where the AM/FM frequencies for the radio were chosen using a sliding bar.

Brief Use Cases

1. **Set time.** When the clock-radio is first turned on, the user chooses standard or military time and then sets the current time by clicking the *set time button*. This time will be displayed in large, bold numbers on the screen so they are able to be read at a distance. When the time is being set, the colon (:) in the middle of the time will blink so the user knows they are in *time-setting mode*. While the : is blinking, the user can use the *hour and minute* buttons on the side of the GUI to set the time to the current time.
2. **Set alarm times.** The user wishes to set an alarm for two different people (a *dual-alarm*). The user will choose which alarm they are setting (alarm 1 or 2) by sliding the *alarm-chooser toggle* from side to side. To set the first alarm for the first person, they will set the *alarm-chooser toggle* to the *alarm-1* position. They will then press the *set time button*, which is visible on the GUI. The : will begin to blink so the user knows they are in *time-setting mode*. Using the *hour and minute buttons*, they will adjust the time to the time they would like their alarm clock to go off. They will set the time for the alarm and then choose whether the alarm will go off or the radio will play. User also has the option to set no alarm at all by changing both the hour and minute digits to “00”. The user then moves the toggle to the *alarm-2* position to set the time for the second person’s alarm. They repeat the steps they completed to set the first alarm.
3. **Snooze the alarm.** The alarm goes off at the set time and the user has the option to either snooze the alarm or turn it off. The user presses the *snooze button* on the GUI, and the alarm pauses for 10 minutes before sounding again while the user falls back

asleep. When the alarm sounds again, the user presses the *alarm off button* which turns the alarm off until the following day, when it will ring again at the set time.

4. **Turn off alarm.** When the alarm goes off, the user can hit the *alarm off button* to make it stop sounding. Alarm turns off until the the next day, when it goes off at the set time.
5. **Turn radio on.** The user is awake and would like to play the radio. On the GUI, they change the *mode toggle* to *radio mode* which puts the alarm clock into *radio mode* and turns the radio on; whatever the radio was previously playing will begin playing to whatever volume it was at the last time it was playing.
6. **Change between AM/FM mode.** The user is currently in *AM mode* and wants to switch to *FM mode* so they flip the *AM/FM toggle switch* from AM to FM.
7. **Adjust radio frequency.** While the alarm clock radio is in *radio mode*, the user wants to switch from the FM station of 99.7 FM to their favourite station, 102.4 FM. Using the *radio frequency slider* on the GUI, they increase the frequency until they can see the *frequency bar* on the slider line up with “102.4” on the FM layer.
8. **Adjust volume of the radio.** The user wants to increase the volume of the music playing on the radio station while it is in *radio mode.* The user increases the volume by pressing the *+/- adjust radio volume buttons* on the GUI.

Fully Dressed Use Cases

Use Case UC1: Set Time

Use Case Section	**Comment**
Use Case Name	Set time
Scope	Dual-Alarm Clock AM/FM Radio Software
Level	User goal
Primary Actor	Clock user
Stakeholders and Interests	Clock user: - Wants their clock to align with real time
Preconditions	The radio must be powered on
Success Guarantee (Postconditions)	The clock’s internal time will be set to the user specified time
Main Success Scenario (Basic Flow)	1. User wants to set time 2. The display begins to flash ‘12:00’, signifying the time setting mode has been entered 3. User sets hours and minutes

	4. Desired time is entered 5. The clock is now in standard mode and set to the new time
Extensions	- The user can to increase by 10 a second if wanted - If the user sets the time to a previously set alarm time, the alarm will activate
Special Requirements	n/a
Technology and Data Variations List	n/a
Frequency of Occurrence	Once to rarely
Miscellaneous	n/a

Use Case UC2: Set Alarm Times

Use Case Section	**Comment**
Use Case Name	Set alarm times
Scope	Dual-Alarm Clock AM/FM Radio Software
Level	User goal
Primary Actor	Alarm clock user #1
Stakeholders and Interests	Alarm clock user #1 (ACU1): - Wants a fast, intuitive way to quickly set both alarms for themselves and also for alarm clock user #2. Wants both alarms to go off reliably. Alarm clock user #2 (ACU2): - Wants to be guaranteed that alarm will go off at the chosen time set for alarm #2 so they may wake up on time.
Preconditions	Alarm is powered on, initial time has been set, all necessary buttons are functioning properly, user #1 knows the correct time they would like to be woken up as well as alarm user #2's desired wake-up time. System must also have a way to emit sound - sound must be on.
Success Guarantee (Postconditions)	Both alarm #1 and #2 are saved after user #1 sets them. Both go off reliably at the times they are set to go off and both users wake

	up on time to get started with their days.
Main Success Scenario (Basic Flow)	1. Alarm Clock User 1 (ACU1) wishes to set an alarm for themselves and ACU2 (a *dual-alarm*). 2. ACU1 will choose to set alarm 1. 3. ACU1 will then choose to set the time. The : will begin to blink so ACU1 knows they are in *time-setting mode*. 4. ACU1 will adjust the time to the time they would like their alarm clock to go off the following day (5:30AM). 5. ACU1 sets the time for alarm #1 and then choose that the alarm will go off by playing the radio. 6. ACU1 saves the settings for alarm #1. 7. ACU1 then chooses to set alarm clock user 2's (ACU2) alarm. 8. ACU1 puts the alarm clock radio into *time-setting mode*. 9. ACU1 sets the time to ACU2's preferred wake up time (6:45AM). 10. ACU1 chooses "A" so that the alarm will sound, not the radio, when alarm #2 goes off. 11. ACU1 saves the settings for alarm #2.
Extensions	- At any time, ACU1 can choose to exit *time-setting mode.* - At any time if the system is shut down or unplugged, once it is turned on it will have the settings it had before it crashed, but will not be in *time-setting mode*. 2-11 a. ACU1 can choose which alarm they are setting - it does not have to be done in order. Alarm #1 can be set and alarm #2 does not have to be set. b. If the user does not wish to set an alarm, they can set the alarm to 00:00 using the hour and minute buttons, which effectively means "no alarm" for either alarm #1 or alarm #2.
Special Requirements	- GUI must have digits that display the time in large numbers that can be read - Sound emitted from system must be audible from 3 feet away
Technology and Data Variations List	- All input must be done through the GUI, there is no other way to use the system.
Frequency of Occurrence	Occasional.
Miscellaneous	n/a

Use Case UC3: Snooze the Alarm

Use Case Section	Comment
Use Case Name	Snooze the alarm
Scope	Dual-Alarm Clock AM/FM Radio Software
Level	User goal
Primary Actor	Alarm user
Stakeholders and Interests	Alarm user: - Wants to temporarily silence the alarm - Wants the alarm to play again in 5 minutes
Preconditions	- The alarm must be set - The alarm must be active
Success Guarantee (Postconditions)	- The alarm is disabled - The alarm is set to activate in 5 minutes
Main Success Scenario (Basic Flow)	1. The alarm activates 2. The user activates snooze 3. The alarm deactivates 4. The alarm is set to activate in 5 minutes
Extensions	- User uses Turn Off Alarm, snooze cycle is deactivated
Special Requirements	- Alarm 1 or Alarm 2 must be activated
Technology and Data Variations List	n/a
Frequency of Occurrence	Occasionally to regularly
Miscellaneous	n/a

Use Case UC4: Turn Off Alarm

Use Case Section	Comment
Use Case Name	Turn off alarm

Scope	Dual-Alarm Clock AM/FM Radio Software
Level	User goal
Primary Actor	Alarm user
Stakeholders and Interests	Alarm user: - Wants to disable the alarm.
Preconditions	System is powered on, alarm 1 or 2 has been set, alarm is active
Success Guarantee (Postconditions)	Alarm is no longer sounding.
Main Success Scenario (Basic Flow)	1. The alarm has activated and is sounding 2. The user chooses to turn off alarm 3. The alarm is now deactivated
Extensions	- Alarm will be disabled until the time specified in alarm 1 or alarm 2 is reached
Special Requirements	n/a
Technology and Data Variations List	n/a
Frequency of Occurrence	Regularly
Miscellaneous	n/a

Use Case UC5: Turn Radio On

Use Case Section	Comment
Use Case Name	Turn radio on
Scope	Dual-Alarm Clock AM/FM Radio Software
Level	User goal
Primary Actor	Radio user
Stakeholders and Interests	Radio user: - Wants a quick easy way to play the radio.
Preconditions	System is powered on, system volume is audible.

Success Guarantee (Postconditions)	Radio user is able to effectively turn the alarm clock into a radio.
Main Success Scenario (Basic Flow)	1. When system is in *time mode*, radio user turns the radio on by switching to *radio mode*. 2. Radio plays the station that it was set at previously - whatever frequency the frequency sliders are set at. 3. Radio user listens to the radio.
Extensions	- At any time, the radio user can choose to exit *radio mode* by switching back to *time mode.* 3a. If the user wants to change the volume that the radio is playing at, they can adjust the volume. 3b. If the user wants to change the radio to either *AM mode* or FM mode, they can set the radio mode to “AM” frequency or “FM” frequency, respectively.
Special Requirements	- At its lowest setting before 0, the volume of the radio must be audible from 1 foot away.
Technology and Data Variations List	n/a
Frequency of Occurrence	Occasional.
Miscellaneous	n/a

Use Case UC6: Change Between AM/FM Mode

Use Case Section	Comment
Use Case Name	Change between AM/FM mode
Scope	Dual-Alarm Clock AM/FM Radio Software
Level	User goal
Primary Actor	Radio user
Stakeholders and Interests	Radio user: - Wants to be able to play both AM and FM stations and be able to switch between them easily.
Preconditions	System is powered on, system volume is audible, system is in *radio mode* and radio is playing either an AM or FM station.

Success Guarantee (Postconditions)	Radio user is able to successfully change the frequency from AM to FM or vice versa, as desired.
Main Success Scenario (Basic Flow)	1. Radio user is listening to the radio and it is set on an FM station. The user wants to change it to an AM station. 2. Radio user switches to *AM mode* and adjusts the settings so that their desired station is playing.
Extensions	- At any time, the radio user can choose to exit *radio mode* by switching back to *time mode.* 2a. If the user wants to change the volume that the radio is playing at, they can adjust the volume. 2b. If the user wants to change the radio back to FM mode, they can set the radio mode to “FM” frequency.
Special Requirements	- At its lowest setting above 0, the volume of the radio must be audible from 1 foot away. - Switching between *AM mode* and *FM mode* will not affect the volume level.
Technology and Data Variations List	n/a
Frequency of Occurrence	Occasional.
Miscellaneous	n/a

Use Case UC7: Adjust Radio Frequency

Use Case Section	Comment
Use Case Name	Adjust radio frequency
Scope	Dual-Alarm Clock AM/FM Radio Software
Level	User goal
Primary Actor	Radio user
Stakeholders and Interests	Radio user: - Wants a quick, simple way to find their radio station of choice.
Preconditions	System is powered on, system volume must be audible, *radio mode* must be turned on.

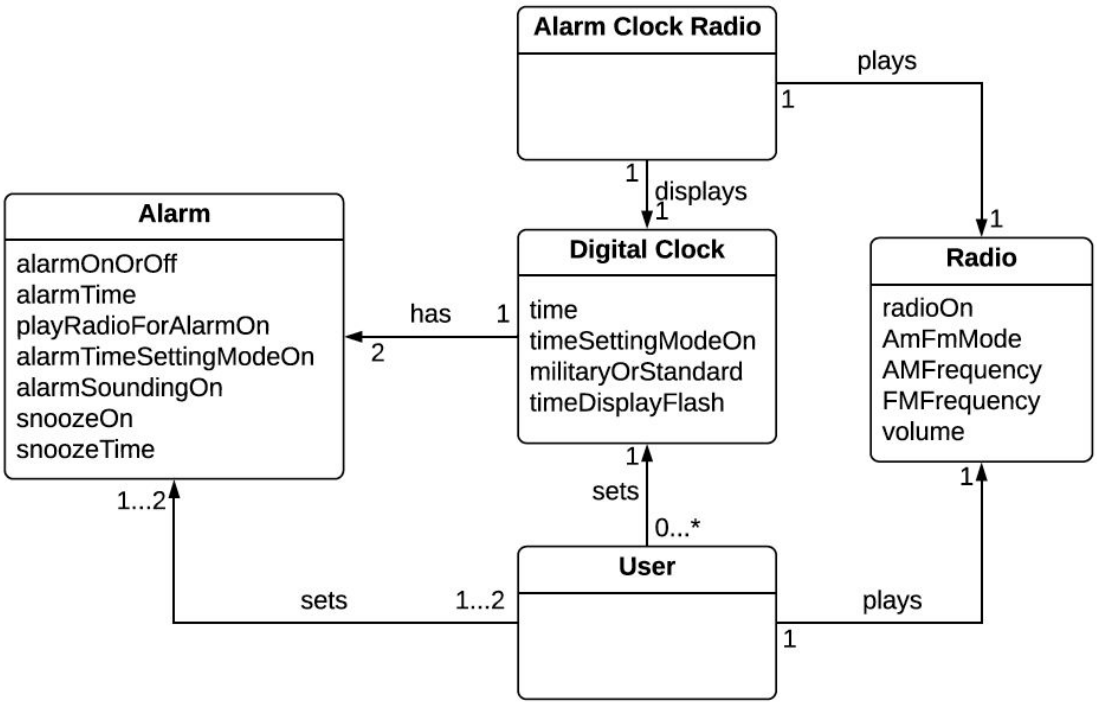
Success Guarantee (Postconditions)	Radio user is able to play the radio station of their choosing quickly and intuitively.
Main Success Scenario (Basic Flow)	1. User turns the radio on by switching to *radio mode*. Whatever station the radio is set on will play at whatever volume the volume has was set to previously. 2. The user increases or decreases the radio frequency until they come to the radio station that they are looking for.
Extensions	- At any time, ACU1 can choose to exit *radio mode* by switching back to *time mode.* - At any time if the system is shut down or otherwise interrupted, once it is turned on it will still be in *radio mode*. 1a. If the user wants to change the volume that the radio is playing at, they can adjust the volume. 2a. If the user wants to change the radio to *AM mode*, they can set the radio mode to “AM” frequency.
Special Requirements	- GUI must have large enough digits on display to read the *AM frequency* or *FM frequency panels* as well as be able to distinguish between tens-place decimal numbers such as 99.8 and 99.9.
Technology and Data Variations List	- All input must be done through the GUI, there is no other way to use the system.
Frequency of Occurrence	Occasional.
Miscellaneous	n/a

Use Case UC8: Adjust Volume of the Radio

Use Case Section	**Comment**
Use Case Name	Adjust volume of the radio
Scope	Dual-Alarm Clock AM/FM Radio Software
Level	User goal
Primary Actor	Radio user
Stakeholders and Interests	Radio User: - Wants to increase or decrease the volume of current or

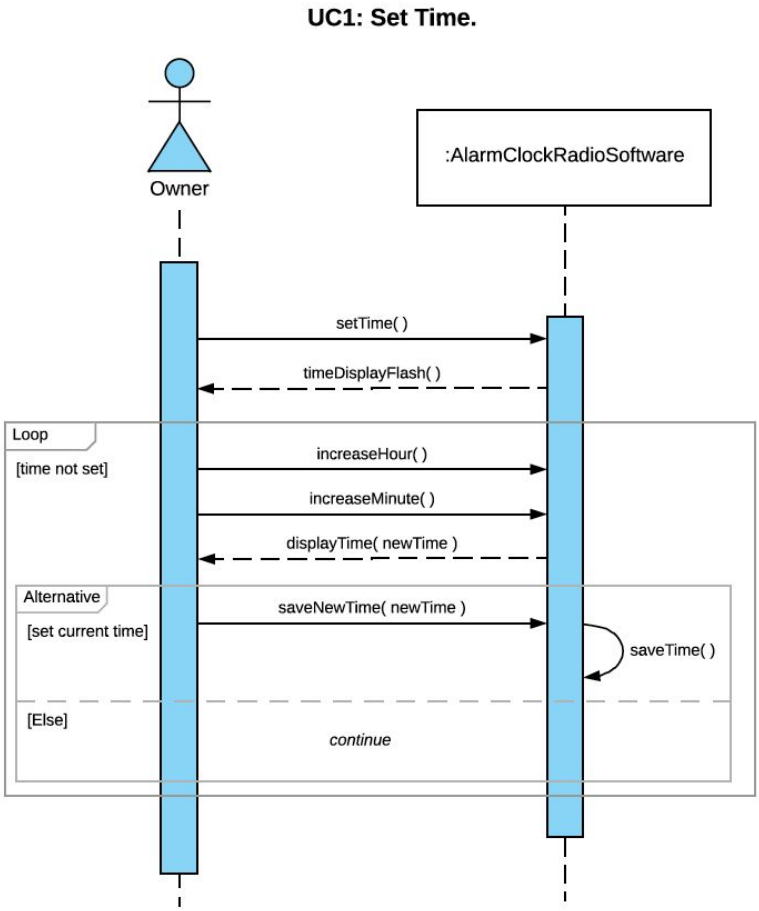
	future radio playback
Preconditions	Volume slider is functioning normally
Success Guarantee (Postconditions)	The volume of radio playback is increased/decreased
Main Success Scenario (Basic Flow)	1. The user sets volume between 0-100% 2. The internal volume of radio playback is changed accordingly
Extensions	- If the radio is playing, the volume is updated in real time - The volume is updated regardless of the device's power state
Special Requirements	n/a
Technology and Data Variations List	n/a
Frequency of Occurrence	Regular
Miscellaneous	n/a

Domain Model

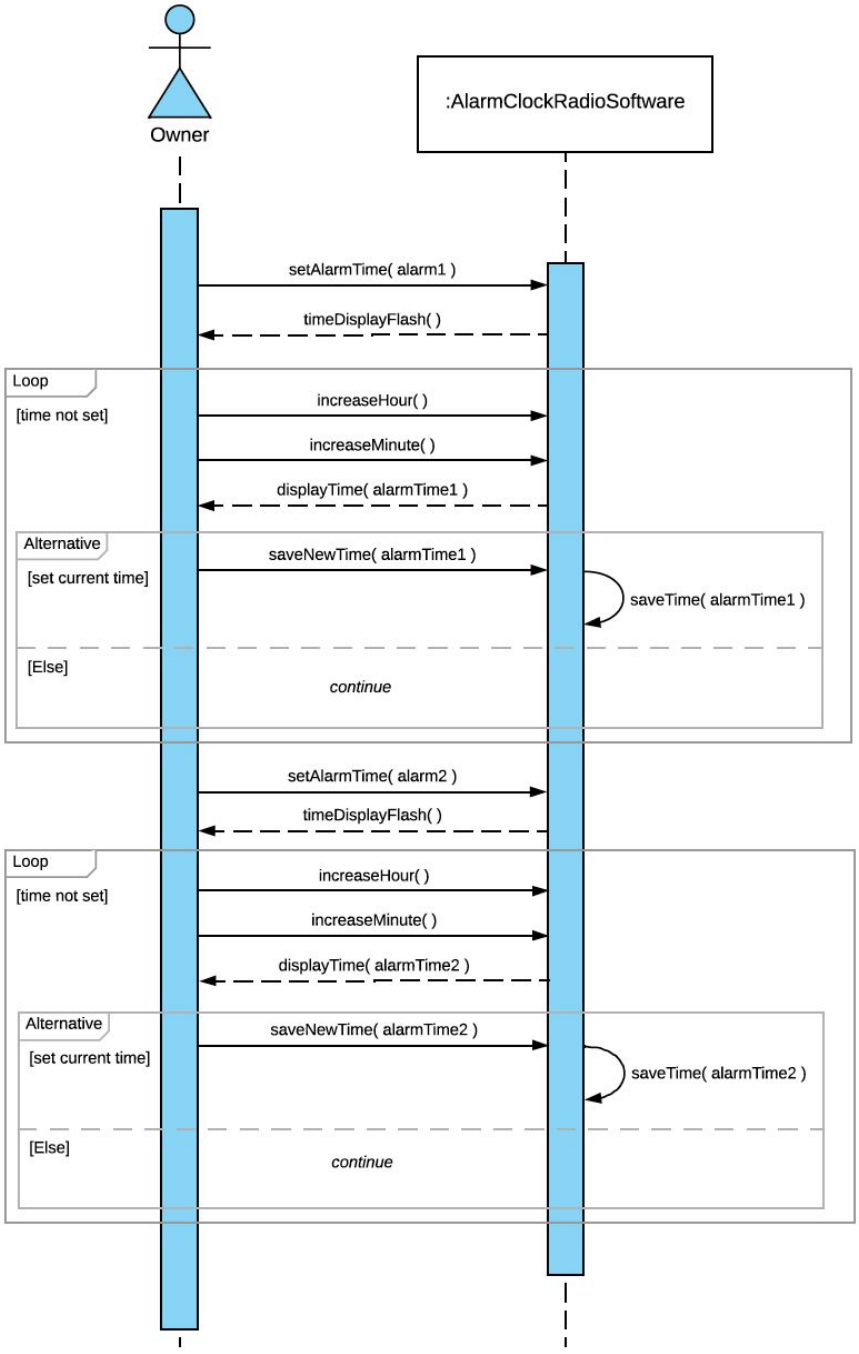


Initial UI Concept Design

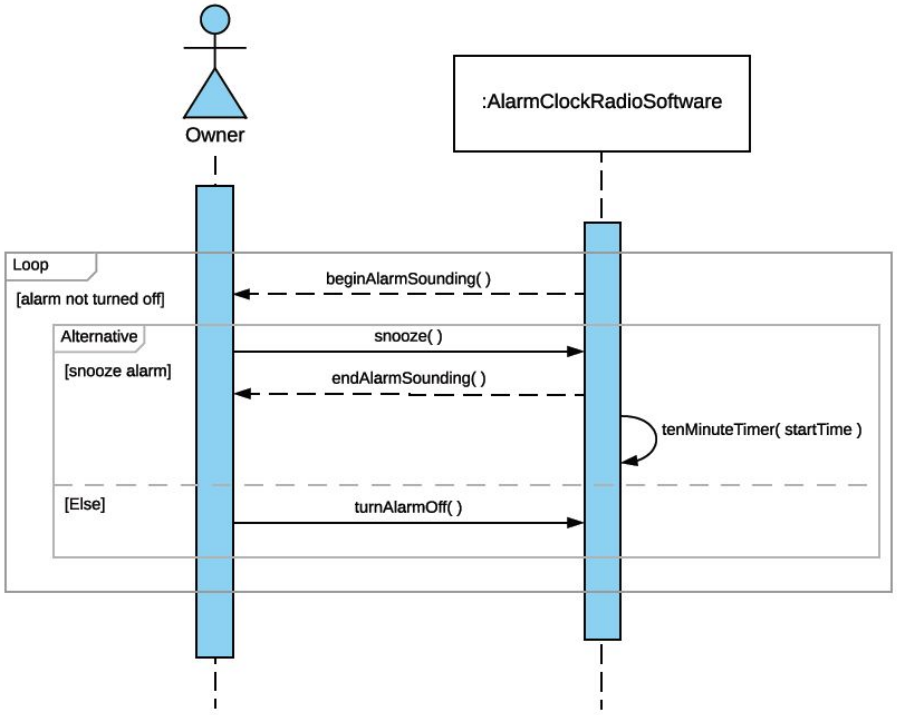
Sequence Diagrams



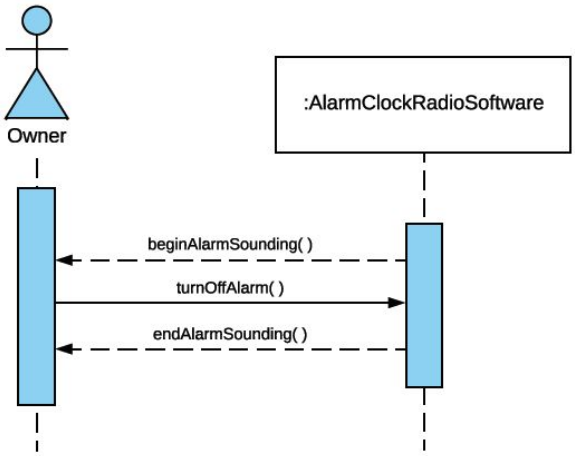
UC2: Set Alarm Times.



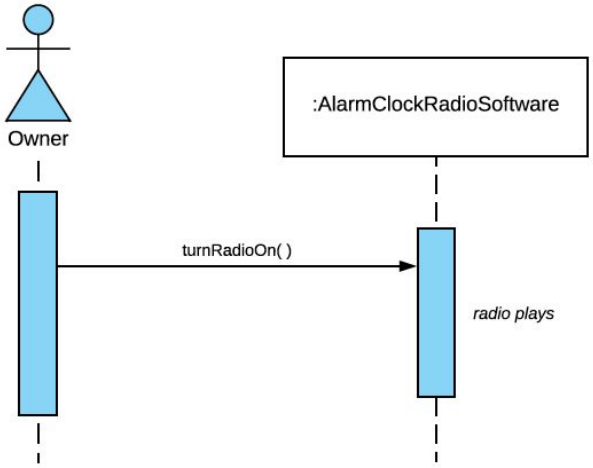
UC3: Snooze.



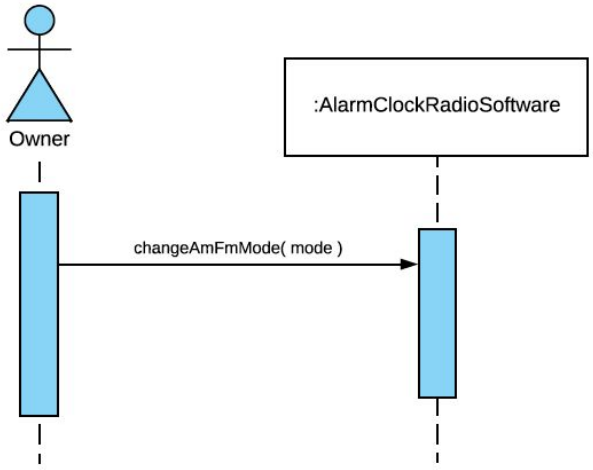
UC4: Turn Off Alarm.



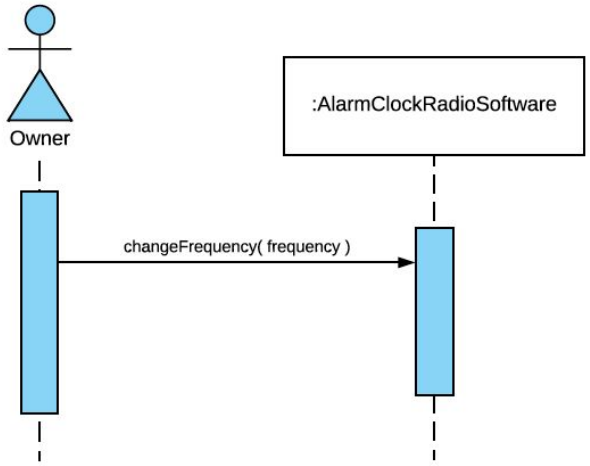
UC5: Turn Radio On.



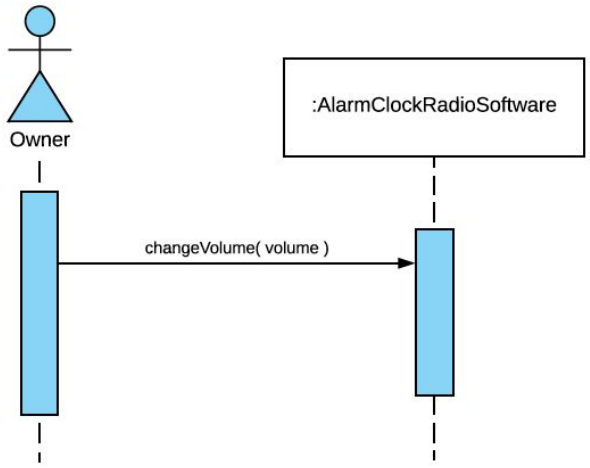
UC6: Change Between AM/FM Mode.



UC7: Adjust Radio Frequency.



UC8: Adjust Volume.



Operation Contracts

Contract CO1: setTime	
Operation:	setTime( )
Cross References:	UC1
Preconditions:	Time attribute is stored in the DigitalClock class. DigitalClock.timeSettingModeOn is set to False. DigitalClock.newTime is set to null.
Postconditions:	- newTime object is instantiated within the DigitalClock class. It is a copy of DigitalClock.time. - DigitalClock.timeSettingModeOn is set to True. - DigitalClock.timeDisplayFlash is set to True.

Contract CO2: timeDisplayFlash	
Operation:	timeDisplayFlash( )
Cross References:	UC1, UC2
Preconditions:	setTime() or setAlarmTime() has been called. DisplayClock.timeDisplayFlash is set to False. DigitalClock.timeSettingModeOn is set to True.
Postconditions:	- DisplayClock.timeDisplayFlash is True. - On the display, the “:” is blinking on and off.

Contract CO3: increaseHour	
Operation:	increaseHour( )
Cross References:	UC1, UC2
Preconditions:	setTime() or setAlarmTime() has been called. DisplayClock.timeDisplayFlash is set to True. DigitalClock.timeSettingModeOn is set to True.
Postconditions:	- DisplayClock.newTime.hour has been incremented.

Contract CO4: increaseMinute	
Operation:	increaseMinute( )
Cross References:	UC1, UC2
Preconditions:	setTime() or setAlarmTime() has been called. DisplayClock.timeDisplayFlash is set to True. DigitalClock.timeSettingModeOn is set to True.
Postconditions:	- DisplayClock.newTime.minute has been incremented.

Contract CO5: displayTime	
Operation:	displayTime(timeHour : int, timeMinute : int)
Cross References:	UC1, UC2
Preconditions:	setTime() or setAlarmTime() has been called. DisplayClock.timeDisplayFlash is set to True. DigitalClock.timeSettingModeOn is set to True.
Postconditions:	- DisplayClock.time is displayed on the GUI when the system is NOT in *time-setting mode*. When the user is in *time-setting mode*, the DisplayClock.newTime object's time is being displayed.

Contract CO6: saveNewTime	
Operation:	saveNewTime(newTime.hour : int, newTime.minute : int)
Cross References:	UC1, UC2
Preconditions:	setTime() or setAlarmTime() has been called. DisplayClock.timeDisplayFlash or Alarm.alarmTimeSettingModeOn is set to True, display is blinking the ":". DigitalClock.timeSettingModeOn is set to True.
Postconditions:	- DigitalClock.timeSettingModeOn or Alarm.alarmTimeSettingModeOn is set to False. - DisplayClock.timeDisplayFlash is set to False.

Contract CO7: saveTime	
Operation:	saveTime( )
Cross References:	UC1
Preconditions:	setTime() has been called. IncreaseHour() and increaseMinute() have possibly been called. DisplayClock.timeDisplayFlash is set to False. DigitalClock.timeSettingModeOn is set to False.
Postconditions:	- DigitalClock.time becomes DigitalClock.newTime. - DigitalClock.newTime is set to null.

Contract CO8: setAlarmTime	
Operation:	setAlarmTime(alarmClockNum: int)
Cross References:	UC2
Preconditions:	alarmTime attribute is stored in the Alarm class. Alarm.alarmTimeSettingModeOn is set to False. DisplayClock.timeDisplayFlash is set to False.
Postconditions:	- Alarm.alarmTimeSettingModeOn is set to True. - DisplayClock.timeDisplayFlash is set to True. - Alarm.newTime is instantiated.

Contract CO9: saveNewTime	
Operation:	saveAlarmTime( )
Cross References:	UC2
Preconditions:	setAlarmTime() has been called. Alarm.alarmTimeSettingModeOn is set to False. DisplayClock.timeDisplayFlash is set to False.
Postconditions:	- Alarm.alarmTime becomes Alarm.newTime. - Alarm.newTime is set to null.

Contract CO10: beginAlarmSounding	
Operation:	beginAlarmSounding()
Cross References:	UC3, UC4
Preconditions:	Alarm.alarmSoundingOn is set to False.
Postconditions:	- Alarm.alarmSoundingOn is set to True.

Contract CO11: snooze	
Operation:	snooze()
Cross References:	UC3, UC4
Preconditions:	Alarm.snoozeOn is set to False. Alarm.snoozeTime is set to null.
Postconditions:	- Alarm.snoozeOn is set to True. - tenMinuteTimer(DigitalClock.time) is called. Alarm.snoozeTime is set to a time ten minutes from then.

Contract CO12: endAlarmSounding	
Operation:	endAlarmSounding()
Cross References:	UC3, UC4
Preconditions:	Alarm.alarmSoundingOn is set to True.
Postconditions:	- Alarm.alarmSoundingOn is set to False.

Contract CO13: tenMinuteTimer	
Operation:	tenMinuteTimer(startTime.hour : int, startTime.minute : int)
Cross References:	UC3, UC4
Preconditions:	Alarm.snoozeOn is set to True.
Postconditions:	- Alarm.snoozeOn is set to False. (after 10 minutes) - Alarm.snoozeTime is equal to DigitalClock.time. - beginAlarmSounding( ) is called.

Contract CO14: turnAlarmOff	
Operation:	turnAlarmOff( )
Cross References:	UC3, UC4
Preconditions:	Alarm.snoozeOn is set to False. Alarm.alarmSoundingOn is True. beginAlarmSounding() has been called.
Postconditions:	- Alarm.alarmSoundingOn is False. - endAlarmSounding() has been called.

Contract CO15: turnRadioOn	
Operation:	turnRadioOn( )
Cross References:	UC5
Preconditions:	Radio.radioOn is set to False.
Postconditions:	- Radio.radioOn is set to True.

Contract CO16: changeAmFmMode	
Operation:	changeAmFmMode(mode : String)
Cross References:	UC6
Preconditions:	Radio.radioOn is set to True.
Postconditions:	- Radio.AmFmMode is set to the *mode* String argument that was passed in.

Contract CO17: changeFrequency	
Operation:	changeFrequency(frequency : int)
Cross References:	UC7
Preconditions:	Radio.radioOn is set to True.
Postconditions:	- Radio.frequency is set to the *frequency* int argument that was passed in.

Contract CO18: changeVolume	
Operation:	changeVolume(volume : int)
Cross References:	UC8
Preconditions:	Radio.radioOn is set to True.
Postconditions:	- Radio.volume is set to the *volume* int argument that was passed in.

Glossary

Term	Definition
A/R alarm-radio button	Located next to the *set time button*, this button allows the user to choose whether the alarm or the radio will sound when the alarm goes off. Users know their choice because there is a little “A” or “R” on the GUI next to the alarm time being set when they are in *time-setting mode*.
alarm	Defined as an interruptive sound that signals it is time to set about a task. Examples of useful alarms are when they sound to wake up sleepers, signalling it is time for the sleeper to get up and go to work.
alarm off button	Located next to the *snooze button*, this button turns the sound of the alarm off until it is time for it to ring at its set time the next day.
alarm-1 position	When the *alarm-chooser toggle* is set to this position, it means that, while the system in is *time-setting mode*, the user is setting the time for the person who corresponds with the first alarm.
alarm-2 position	When the *alarm-chooser toggle* is set to this position, it means that, while the system in is *time-setting mode*, the user is setting the time for the person who corresponds with the second alarm.
alarm-chooser toggle	Toggle that allows for the user to choose between setting the alarm for the first person using the *dual-alarm* system or the second person. Toggle can be switched from *alarm-1 position* to *alarm-2 position.*
AM frequency panel	The standard bandwidth for AM is 540-1600 kHz, allowing for a possible 106 bands or “stations” to be played. This panel displays this range in digits above the *radio frequency slider.* The position of the *radio frequency slider* tells the user which station the radio is playing by pointing to the number corresponding to that station if the radio is in *AM mode*.
AM mode	When the *AM/FM toggle switch* is in this position, the radio is playing stations corresponding to where the *radio frequency slider* is pointing on the *AM frequency panel*.
AM/FM toggle switch	Toggle that allows the user to choose between playing *AM mode* or *FM mode*.

dual-alarm	An alarm system that allows for two persons to have differing alarms. Each alarm goes off at its set time and when the *alarm off button* is pressed to stop the first alarm from ringing, it does not interfere with the workings of the second alarm.
dual-alarm AM/FM clock radio	The system we are implementing - this is a system that has both an alarm and a radio mode. The alarm is able to be set for 2 different times per day and the radio can play either AM or FM stations, depending on what the user wants to hear.
FM frequency panel	The standard bandwidth for FM is 88.1-108.1 MHz, allowing for a possible 100 bands or “stations” to be heard. This panel displays this range in digits above the *radio frequency slider.* The position of the *radio frequency slider* tells the user which station the radio is playing by pointing to the number corresponding to that station if the radio is in *FM mode*. This is located on the radio frequency
FM mode	When the *AM/FM toggle switch* is in this position, the radio is playing stations corresponding to where the *radio frequency slider* is pointing on the *FM frequency panel*.
Graphical User Interface (GUI)	The user interface for the system, this is how the user interacts with the system. It displays a graphical representation of the *dual-alarm AM/FM clock radio.*
hour button	Range 0-12 in *standard time* or 0-24 in *military time.* The button to set the hour time when the alarm clock is in *time-setting mode*. Numbers only increase and, when the digits reach either 12 in *standard time* or 24 in *military time*, the digits wrap around and begin at 0 again.
military time	Time is displayed in the range of 00:00 - 23:59. No AM or PM is necessary.
minute button	Range 0-59. The button to set the minute time when the alarm clock is in *time-setting mode*. Numbers only increase and, when the digits reach 60 in *standard time* or *military time*, the digits wrap around and begin at 0 again.
mode toggle	There are 3 positions this toggle can be in: Radio, Time, and Time-Set. When it is in the Radio position, the system is in *radio mode*. When it is in the Time position, the system in the *time mode*. When it is in the Time-Set position, the system is in *time-setting mode.*
radio	A device which plays AM or FM bands, referred to as “stations”. An example of a radio station is “102.5 FM”.

radio frequency slider	The *AM and FM frequency panels* are located above and below the slider. The slider has a vertical bar that can be dragged left and right using the *GUI* to change the digits that it points to and therefore the radio station that is playing.
radio mode	When the radio is on and the *mode toggle* is set to the “Radio” position.
set time button	When the user presses this button in *time-setting mode*, the : begins to blink on the time and the user is able to use the *hour button* and the *minute button* to set the desired time for their alarm. Once this button is pressed again, it saves the alarm time.
snooze button	Located next to the *alarm off button*, this button allows the user the option to snooze. When pressed, this button pauses the alarm for 10 minutes and then the alarm resumes ringing when the 10 minutes is complete.
standard time	Time is displayed in the range of 00:00 - 12:00 and morning is denoted by a small “AM” display while afternoon is denoted by “PM”.
time mode	When the system is simply displaying the time, not playing the radio, and is not in *time-setting mode.* The *mode toggle* is set to the “Time” position.
time-setting mode	When the user is changing the set time for either of the two alarms. The *mode toggle* must be set to the “Time-Set” position.
Volume analog slider	Adjusts the volume of radio.