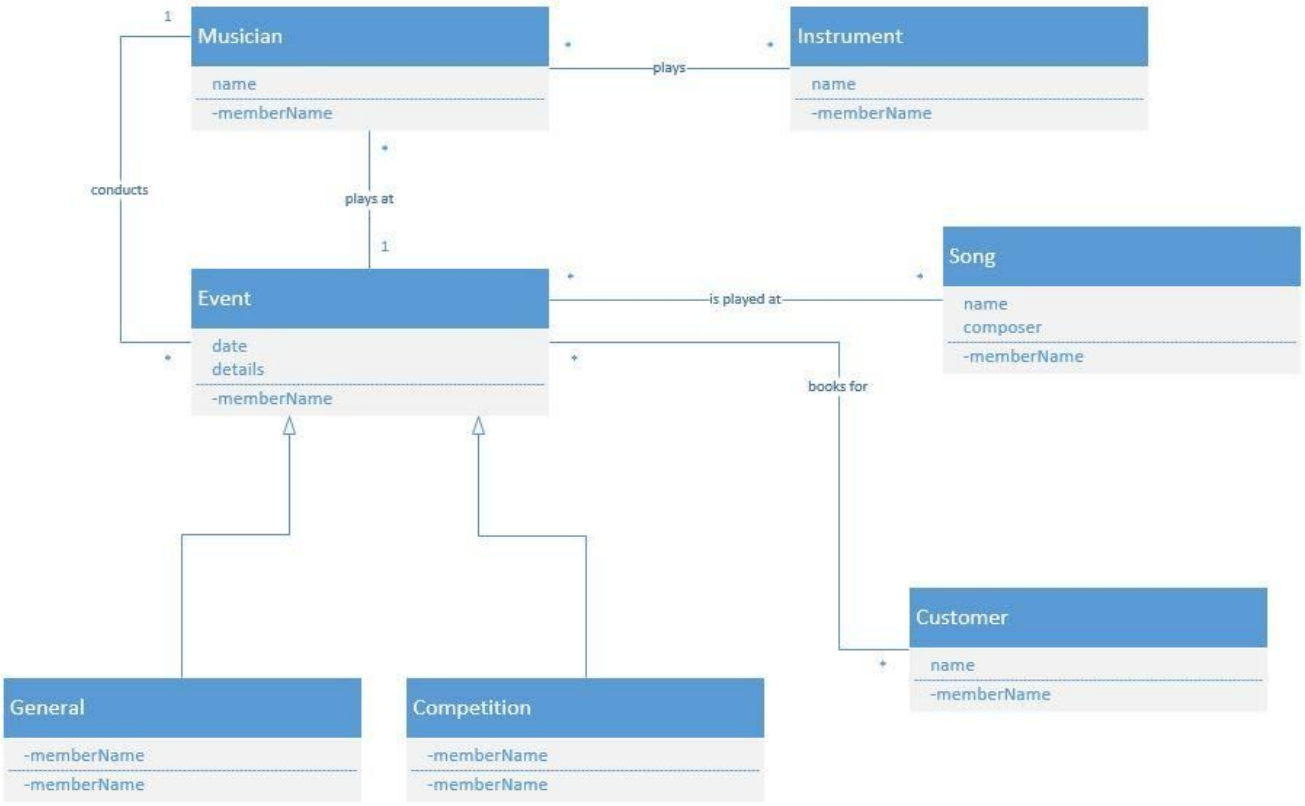
Orchestra Project

#1 UML Deliverables

Group C

Cédric CLIGNIEZ
Siqi LEI
Antoine MALINET

Class diagram

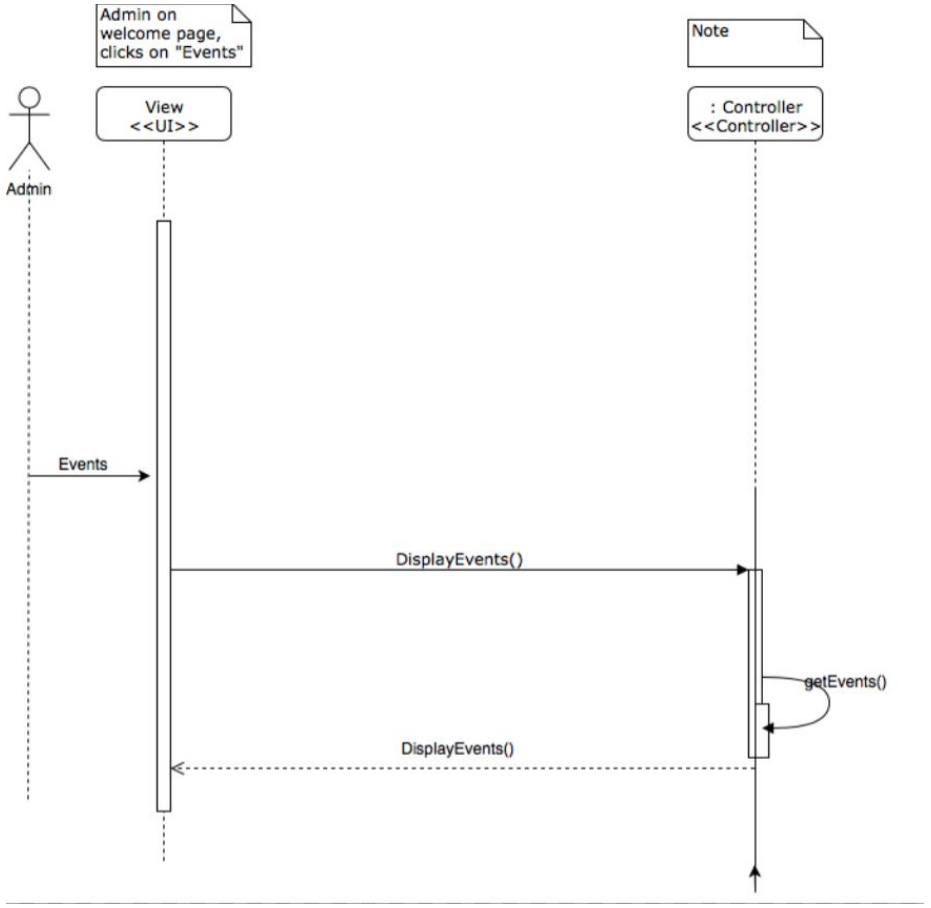


The Customer class is not handled (yet), but without Visio we could not modify the class diagram. The corresponding use cases are therefore not in the use case diagram and in the mock-up.

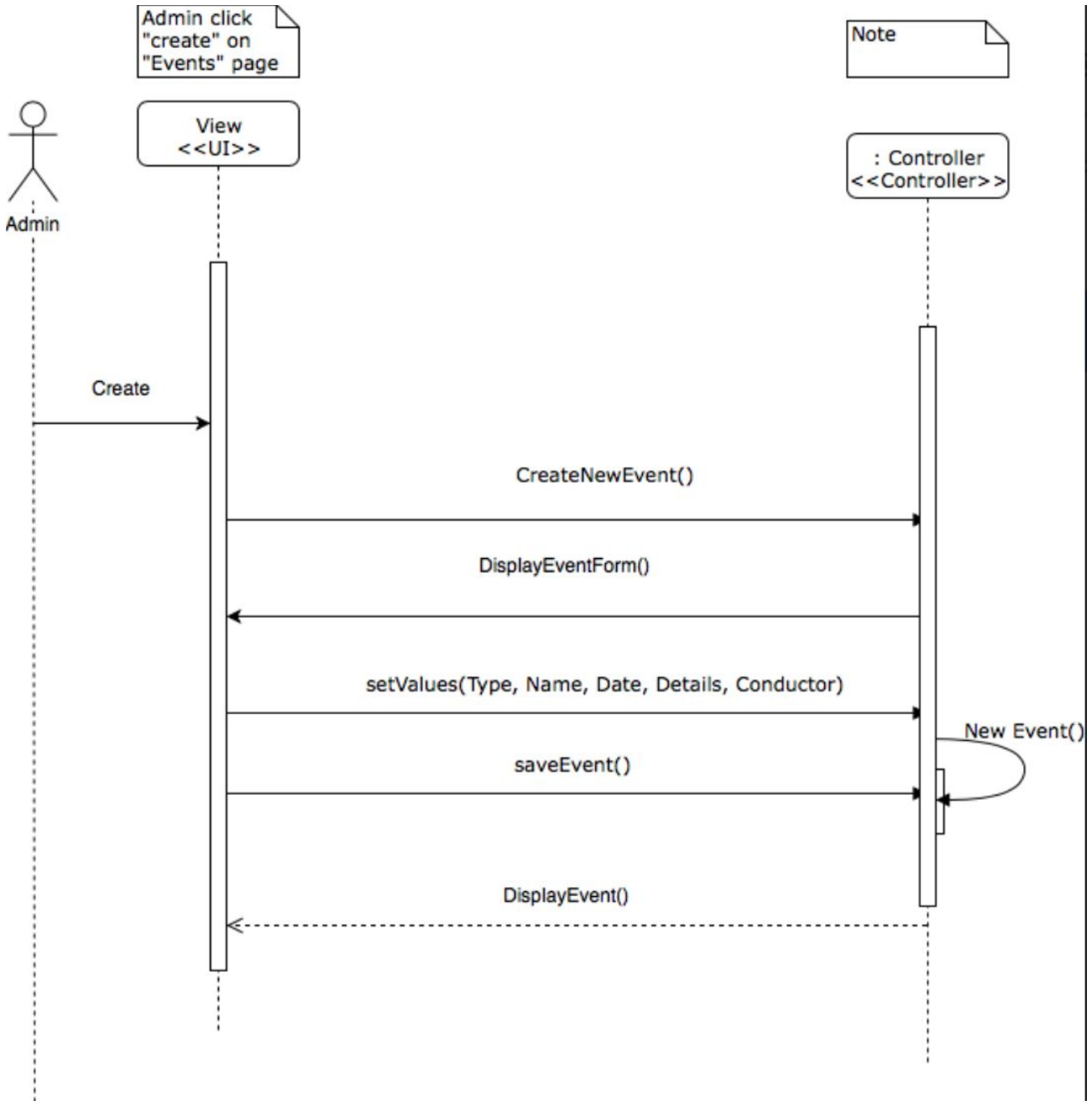
Use case diagram

Scenarios and Sequence diagrams

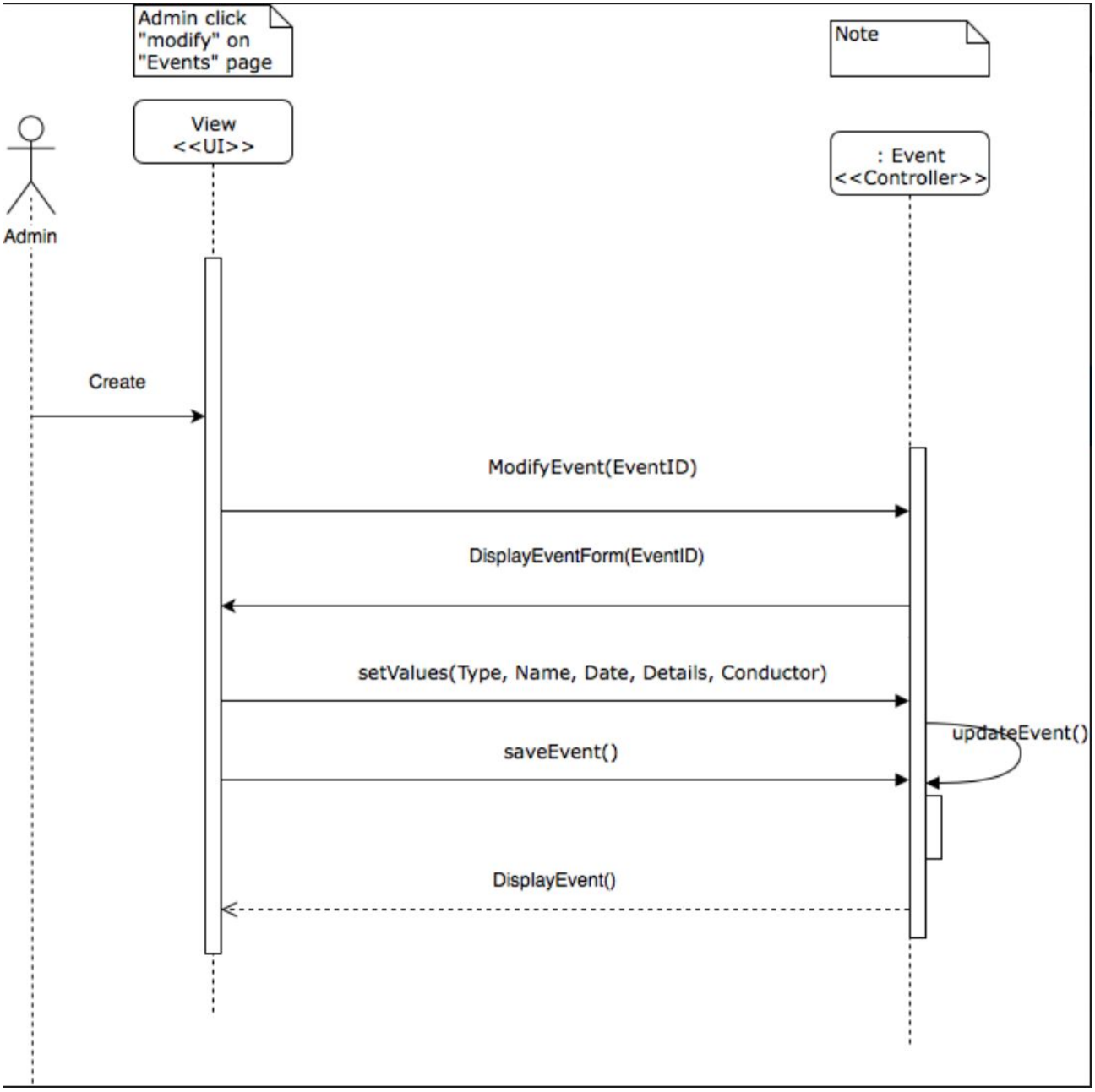
Number	1	
Name	See Events	
Summary	Displays the Events page	
Preconditions	Being on the homepage	
Postconditions	Being on the Events page	
Primary actor(s)	User	
Secondary actor(s)	Events Class	
Main Scenario	**Step**	**Action**
	1	System displays the Events page



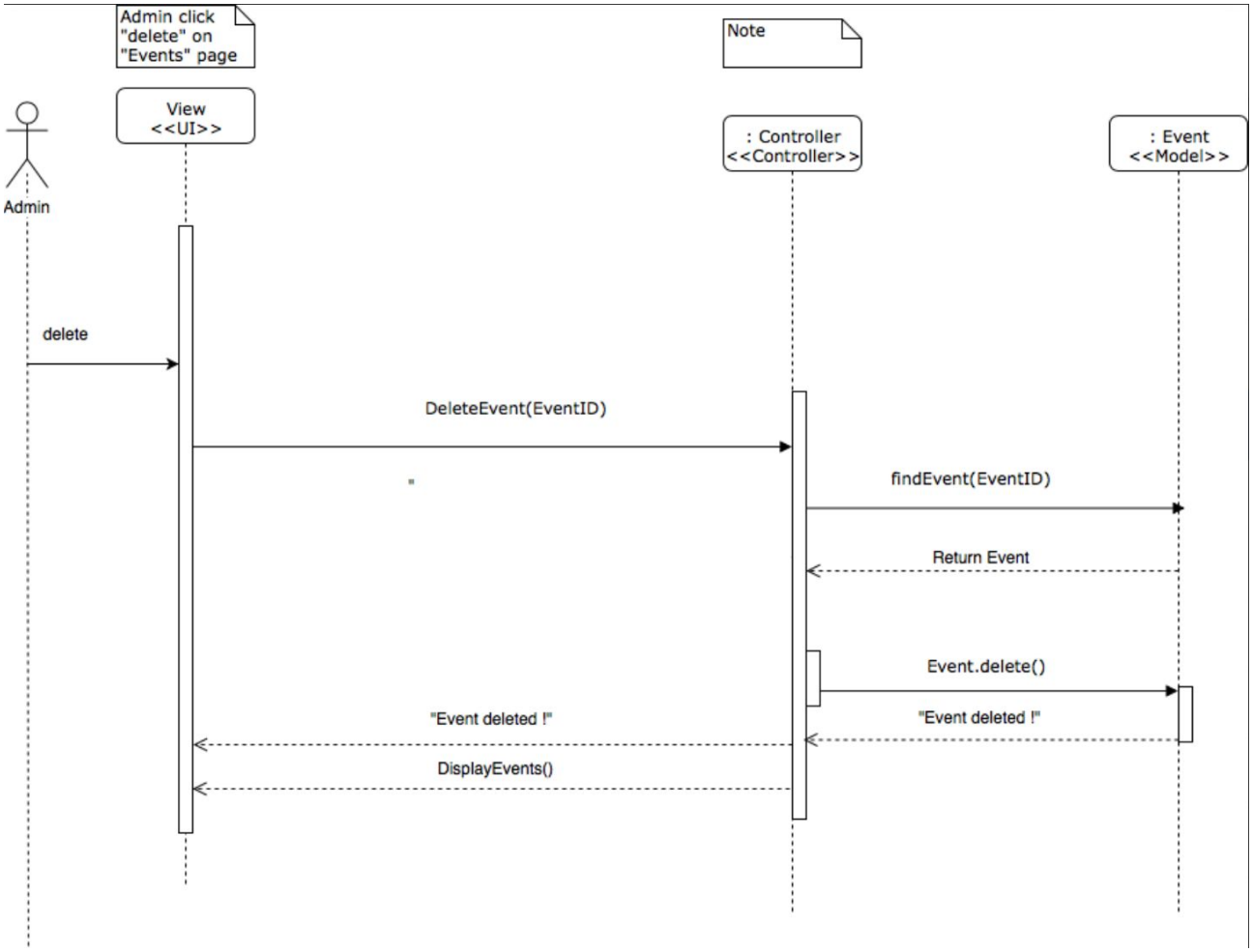
Number	2	
Name	Create Event	
Summary	The admin enters the necessary information to add an event	
Preconditions	Being on the events page, having logged in as admin	
Postconditions	An event has been added	
Primary actor(s)	Admin	
Secondary actor(s)	Events Class	
Main Scenario	**Step**	**Action**
	1	System displays form
	2	User fills form
	3	User presses Save button
	4	System saves the event
	5	System displays event page
Extensions	**Step**	**Branching Action**
	2a	User presses Cancel button
	2b	System displays event page
	3a	If the form has not been correctly filled, display error


Admin click "create" on "Events" page
Note
View <<UI>>
: Controller <<Controller>>
Admin
Create
CreateNewEvent()
DisplayEventForm()
setValues(Type, Name, Date, Details, Conductor)
New Event()
saveEvent()
DisplayEvent()

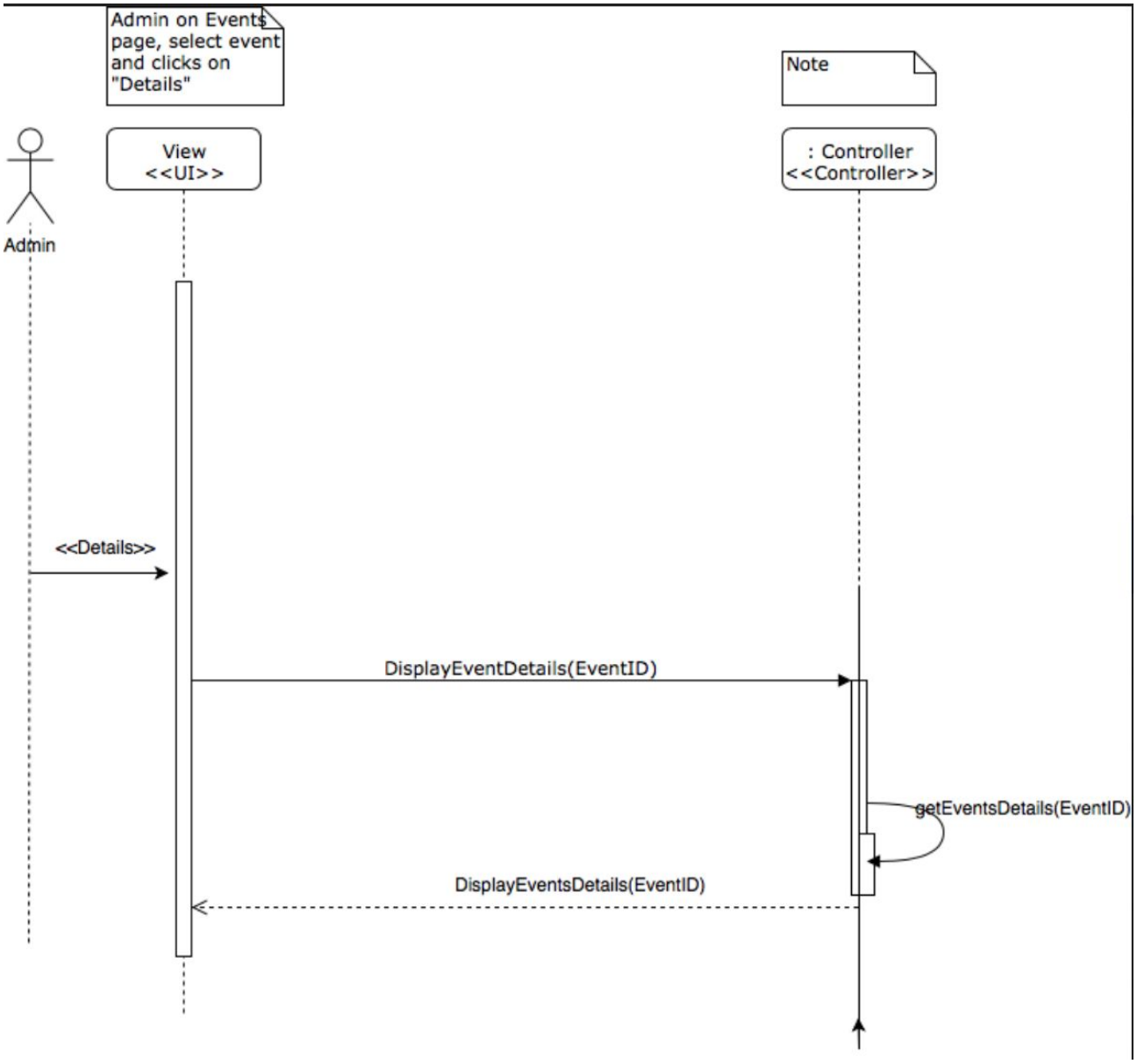
Number	3	
Name	Modify Event	
Summary	The admin enters the modified information of an event	
Preconditions	Being on the events page, having logged in as admin	
Postconditions	The event has been modified	
Primary actor(s)	Admin	
Secondary actor(s)	Events Class	
Main Scenario	**Step**	**Action**
	1	System displays form
	2	User fills form
	3	User presses Save button
	4	System saves the event
	5	System displays event page
Extensions	**Step**	**Branching Action**
	2a	User presses Cancel button
	2b	System displays event page
	3a	If the form has not been correctly filled, display error


Admin click "modify" on "Events" page
Note
View <<UI>>
: Event <<Controller>>
Admin
Create
ModifyEvent(EventID)
DisplayEventForm(EventID)
setValues(Type, Name, Date, Details, Conductor)
updateEvent()
saveEvent()
DisplayEvent()

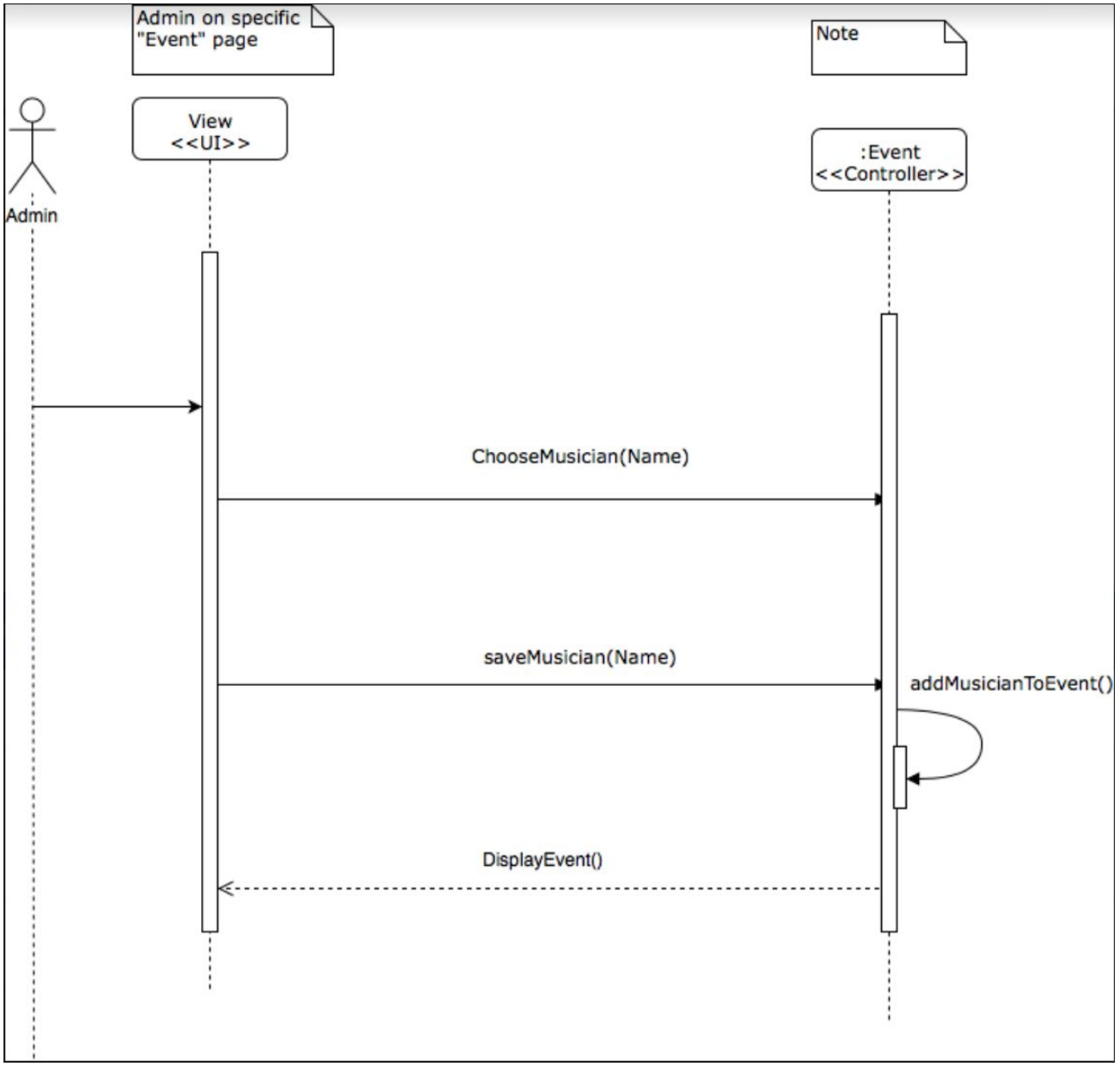
Number	4	
Name	Delete Event	
Summary	Remove an event from the events list	
Preconditions	Being on the events page, having logged in as admin	
Postconditions	The event has been deleted	
Primary actor(s)	Admin	
Secondary actor(s)	Events Class	
Main Scenario	**Step**	**Action**
	1	System displays confirmation window
	2	User presses OK button
	3	System deletes the event
	4	System displays event page
Extensions	**Step**	**Branching Action**
	2a	User presses Cancel button
	2b	System displays event page


Admin click "delete" on "Events" page
Note
View <<UI>>
: Controller <<Controller>>
: Event <<Model>>
Admin
delete
DeleteEvent(EventID)
findEvent(EventID)
Return Event
Event.delete()
"Event deleted !"
"Event deleted !"
DisplayEvents()

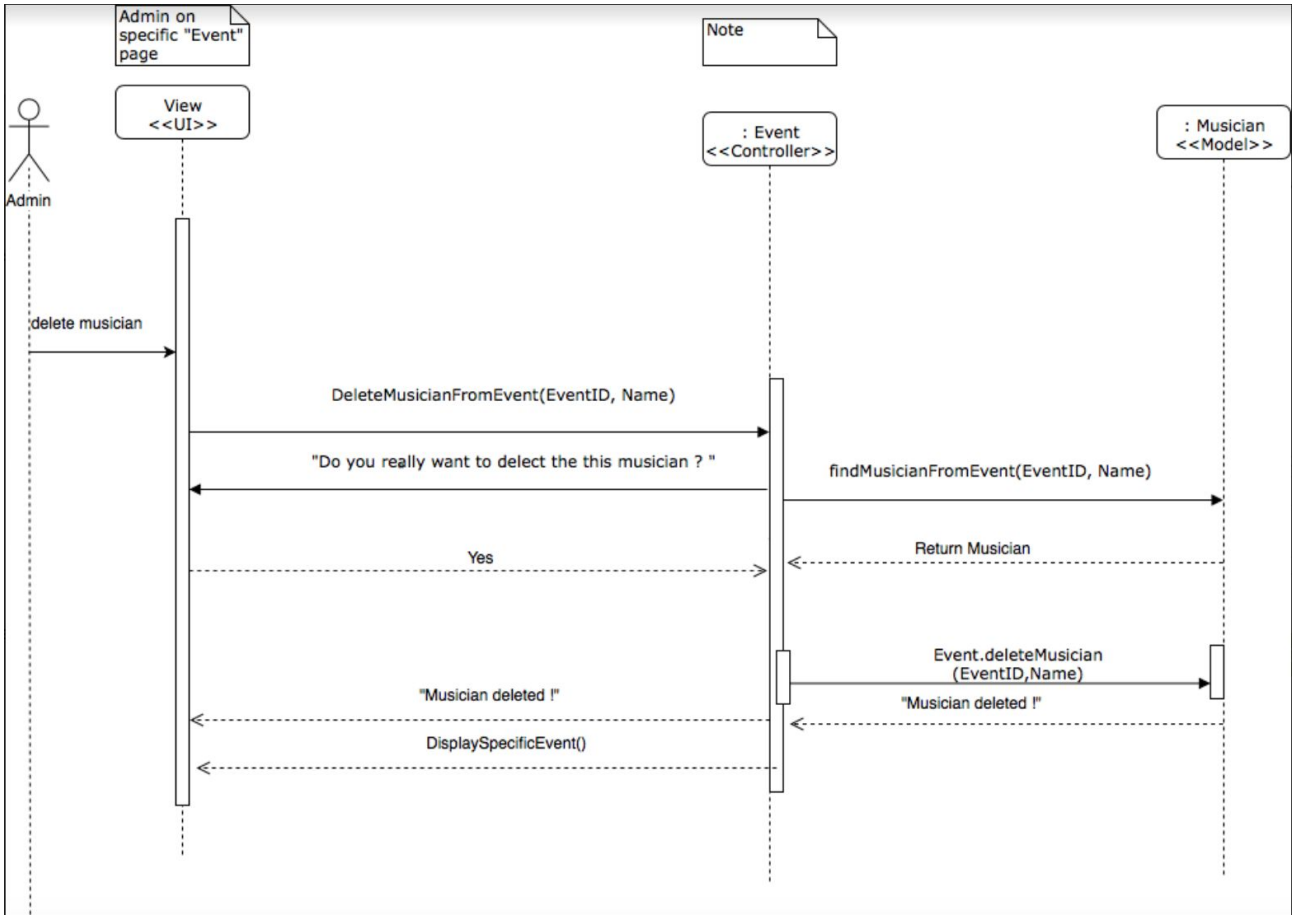
Number	5	
Name	Check Event Details	
Summary	Displays the details of an event	
Preconditions	Being on the events page	
Postconditions	Being on the details page of the event	
Primary actor(s)	Admin	
Secondary actor(s)	Events Class	
Main Scenario	**Step**	**Action**
	1	System displays the details page


Admin on Events page, select event and clicks on "Details"
Note
View <<UI>>
: Controller <<Controller>>
Admin
<<Details>>
DisplayEventDetails(EventID)
getEventsDetails(EventID)
DisplayEventsDetails(EventID)

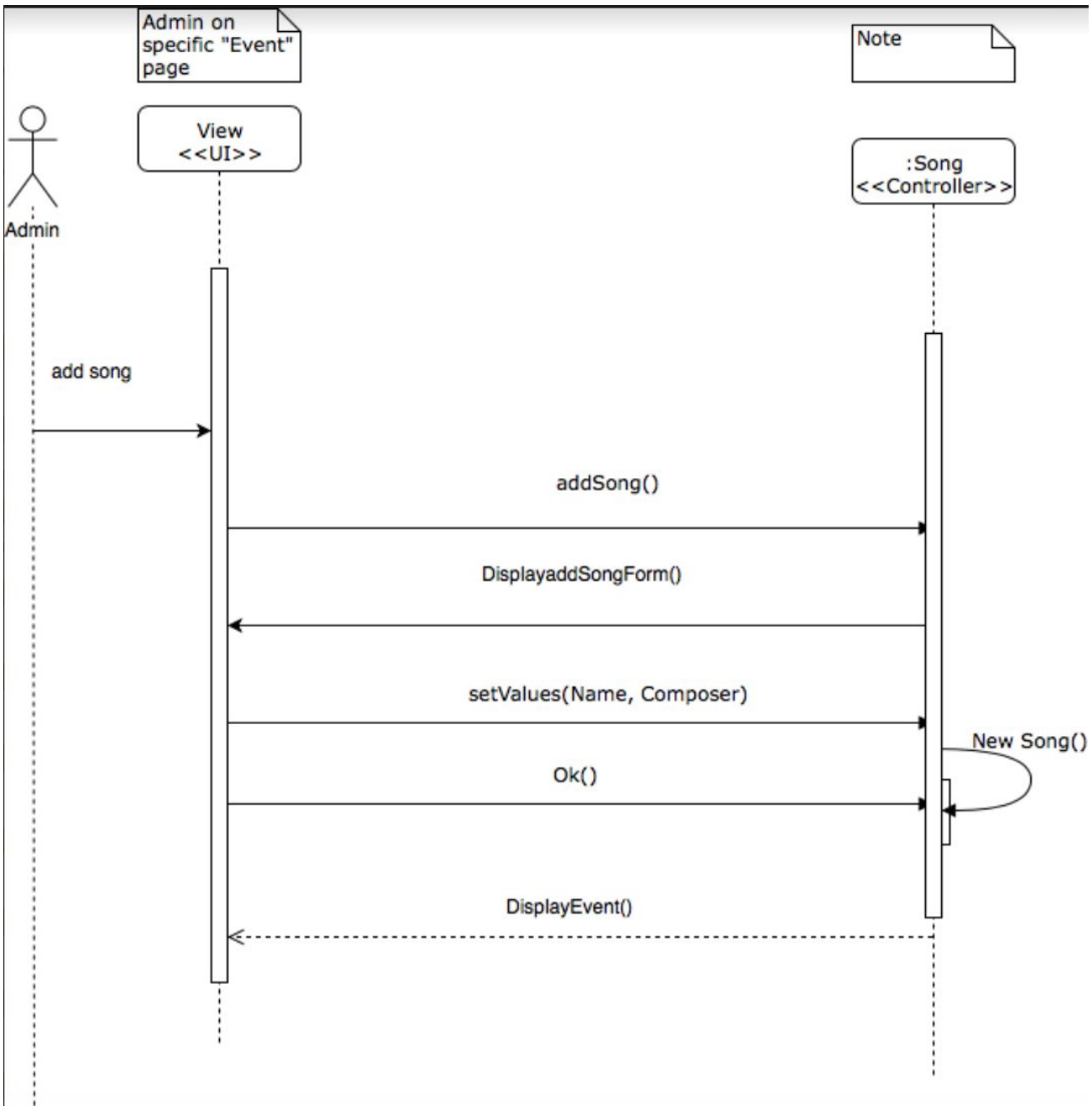
Number	6	
Name	Add Musician To Event	
Summary	Adds a musician to an event	
Preconditions	Being on the details page of an event, being logged in as an admin	
Postconditions	The musician has been added	
Primary actor(s)	Admin	
Secondary actor(s)	Event Class, Musician Class	
Main Scenario	**Step**	**Action**
	1	The user chooses an musician in the scrolling menu
	2	The user presses the Add musician button
	3	The system adds the musician to the event
Extensions	**Step**	**Branching Action**
	2a	The chosen musician is already participating to that event
	2b	Display « The chosen musician is already participating to that event»


Admin on specific "Event" page
Note
View <<UI>>
:Event <<Controller>>
Admin
ChooseMusician(Name)
saveMusician(Name)
addMusicianToEvent()
DisplayEvent()

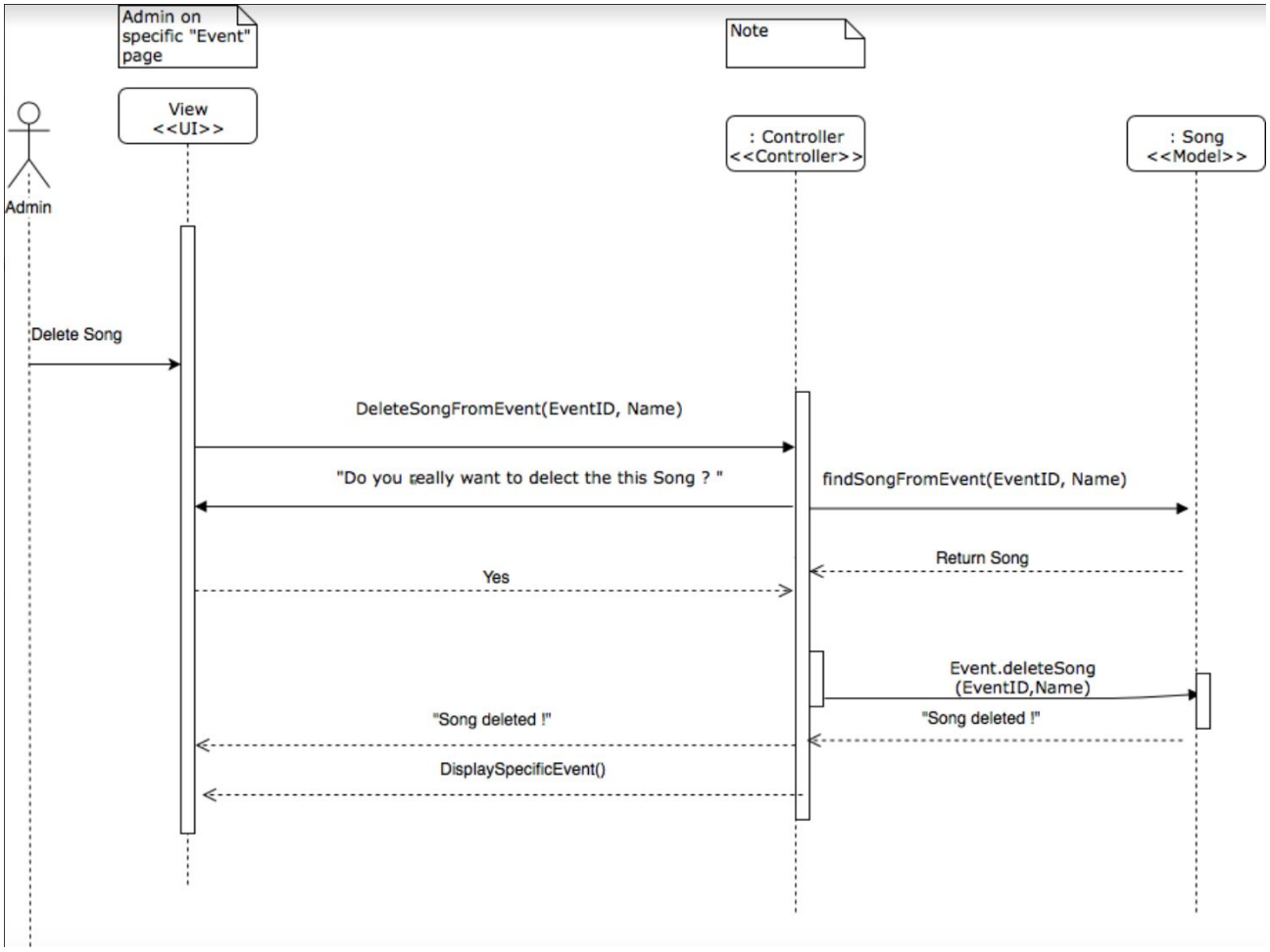
Number	7	
Name	Delete Musician From Event	
Summary	Remove a selected musician from an event	
Preconditions	Being on the details page of an event, having logged in as admin, having selected a musician in the list	
Postconditions	The musician has been removed from the event	
Primary actor(s)	Admin	
Secondary actor(s)	Events Class	
Main Scenario	**Step**	**Action**
	1	System displays confirmation window
	2	User presses OK button
	3	System deletes the musician
	4	System displays details page
Extensions	**Step**	**Branching Action**
	2a	User presses Cancel button
	2b	System displays details page


Admin on specific "Event" page
Note
View <<UI>>
: Event <<Controller>>
: Musician <<Model>>
Admin
delete musician
DeleteMusicianFromEvent(EventID, Name)
"Do you really want to delect the this musician ? "
findMusicianFromEvent(EventID, Name)
Return Musician
Yes
Event.deleteMusician (EventID,Name)
"Musician deleted !"
"Musician deleted !"
DisplaySpecificEvent()

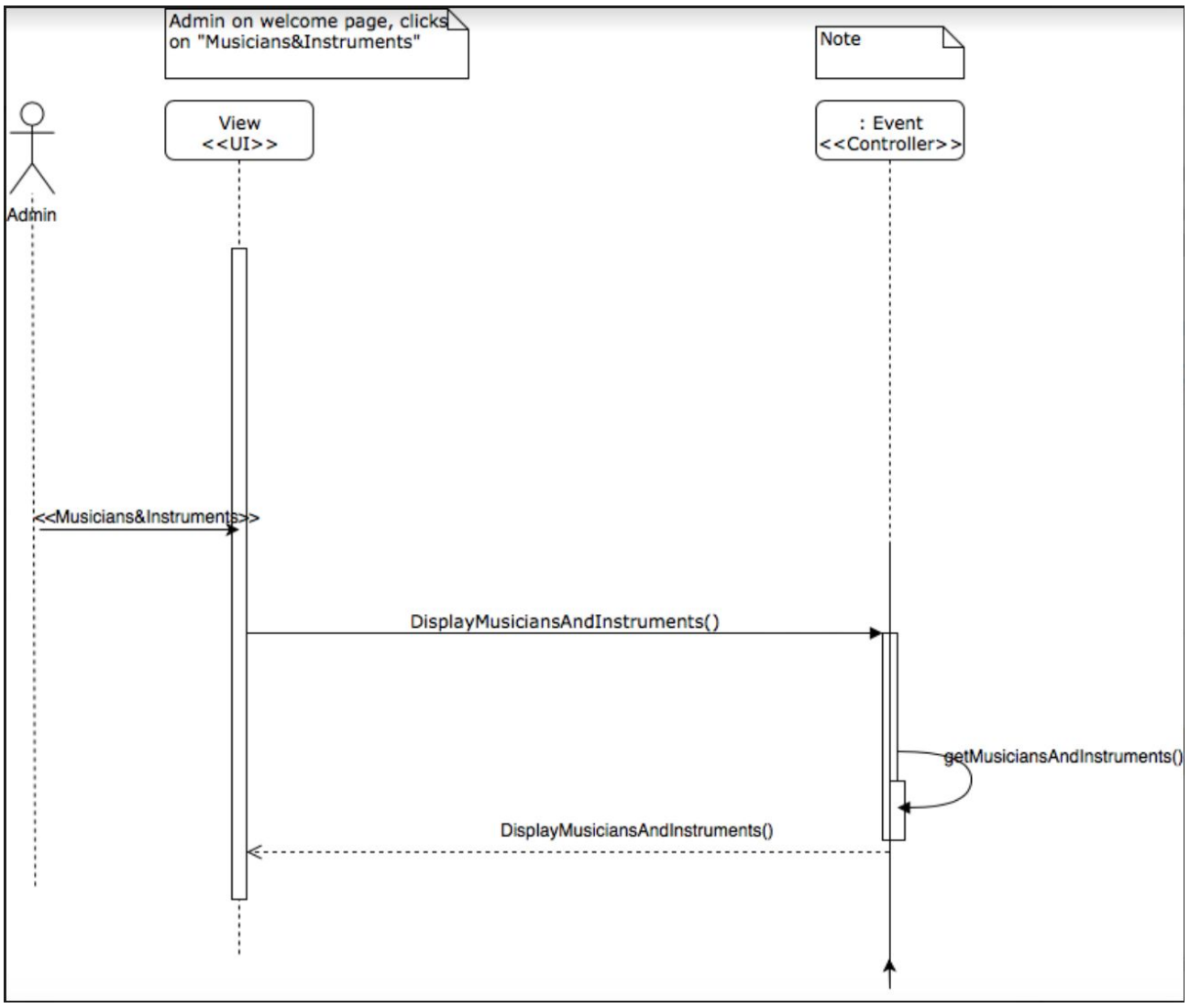
Number	8	
Name	Add Song To Event	
Summary	The admin enters the necessary information to add a song to an event	
Preconditions	Being on the details page of an event, having logged in as admin	
Postconditions	A song has been added	
Primary actor(s)	Admin	
Secondary actor(s)	Events Class	
Main Scenario	**Step**	**Action**
	1	System displays form
	2	User fills form
	3	User presses OK button
	4	System saves the song
	5	System displays details page of the event
Extensions	**Step**	**Branching Action**
	2a	User presses Cancel button
	2b	System displays details page of the event
	3a	If the form has not been correctly filled, display error


Admin on specific "Event" page
Note
View <<UI>>
:Song <<Controller>>
Admin
add song
addSong()
DisplayaddSongForm()
setValues(Name, Composer)
New Song()
Ok()
DisplayEvent()

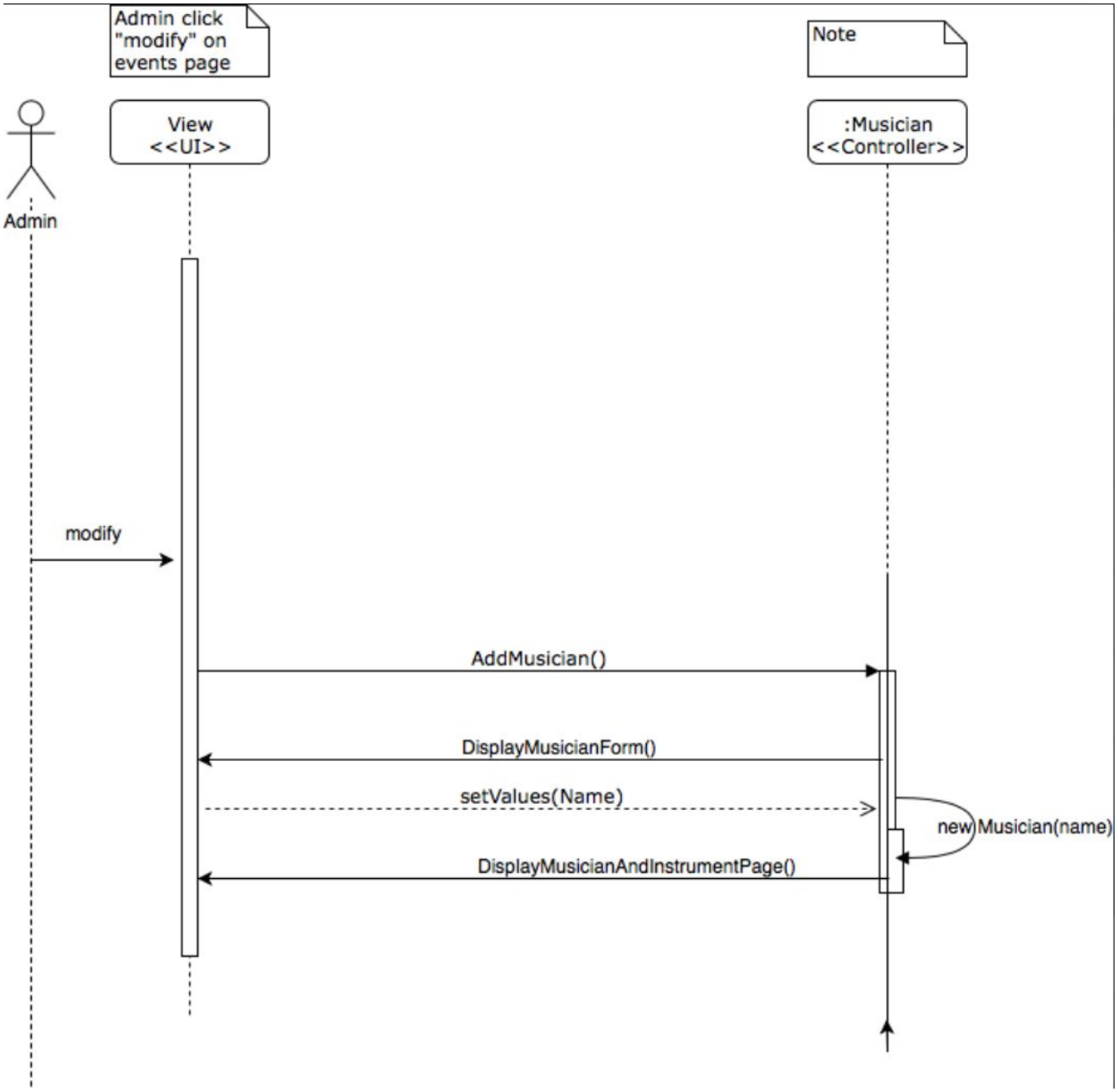
Number	9	
Name	Delete Song From Event	
Summary	Remove a selected song from an event	
Preconditions	Being on the details page of an event, having logged in as admin, having selected a song in the list	
Postconditions	The song has been removed from the event	
Primary actor(s)	Admin	
Secondary actor(s)	Events Class	
Main Scenario	**Step**	**Action**
	1	System displays confirmation window
	2	User presses OK button
	3	System deletes the song
	4	System displays details page
Extensions	**Step**	**Branching Action**
	2a	User presses Cancel button
	2b	System displays details page


Admin on specific "Event" page
Note
View <<UI>>
: Controller <<Controller>>
: Song <<Model>>
Admin
Delete Song
DeleteSongFromEvent(EventID, Name)
"Do you really want to delect the this Song ? "
findSongFromEvent(EventID, Name)
Return Song
Yes
Event.deleteSong (EventID,Name)
"Song deleted !"
"Song deleted !"
DisplaySpecificEvent()

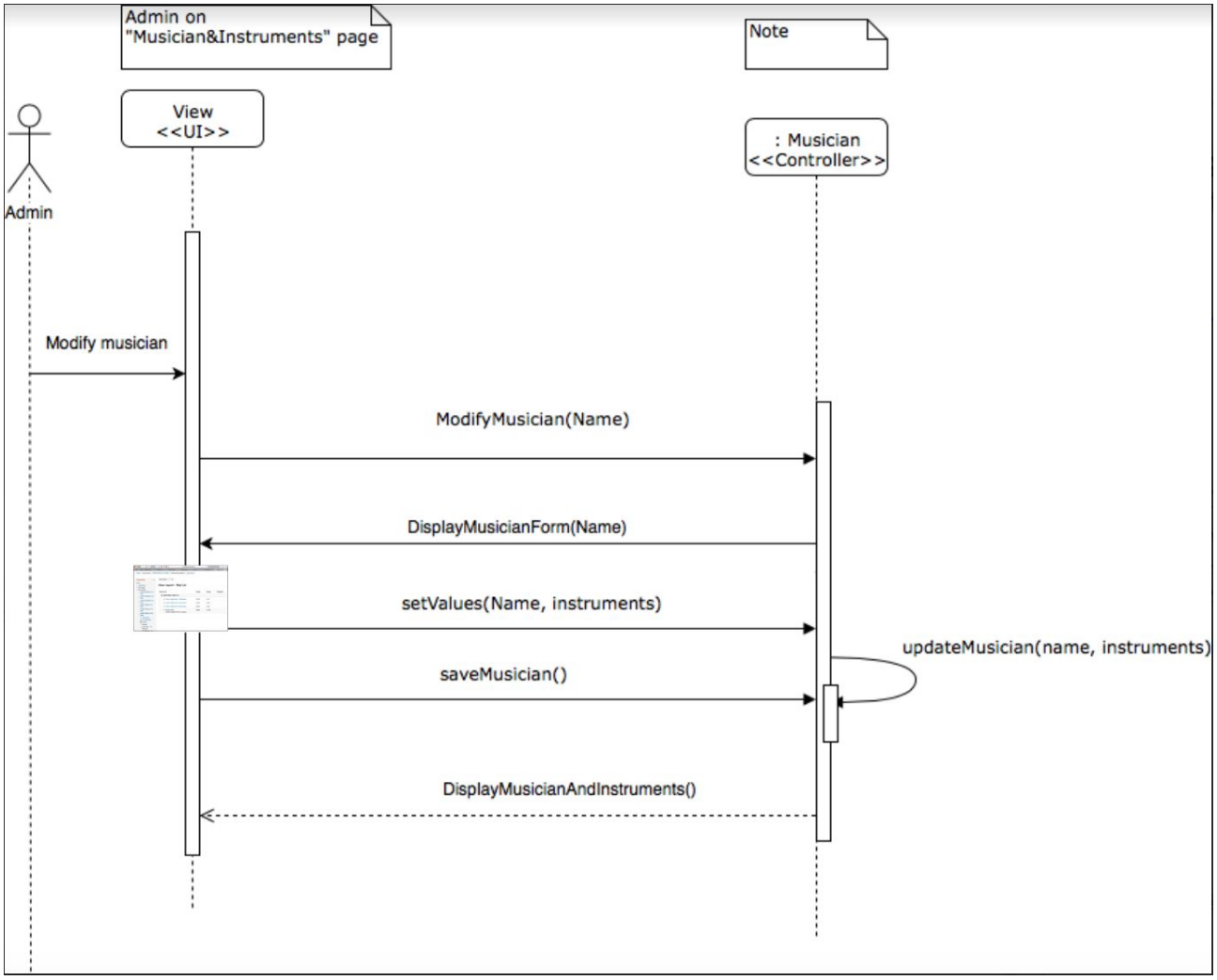
Number	10	
Name	See Musicians And Instruments	
Summary	Displays the Musicians&Instruments page	
Preconditions	Being on the homepage	
Postconditions	Being on the Musicians&Instruments page	
Primary actor(s)	User	
Secondary actor(s)	Musician Class	
Main Scenario	**Step**	**Action**
	1	System displays the Musicians&Instruments page


Admin on welcome page, clicks on "Musicians&Instruments"
Note
View
<<UI>>
: Event
<<Controller>>
Admin
<<Musicians&Instruments>>
DisplayMusiciansAndInstruments()
getMusiciansAndInstruments()
DisplayMusiciansAndInstruments()

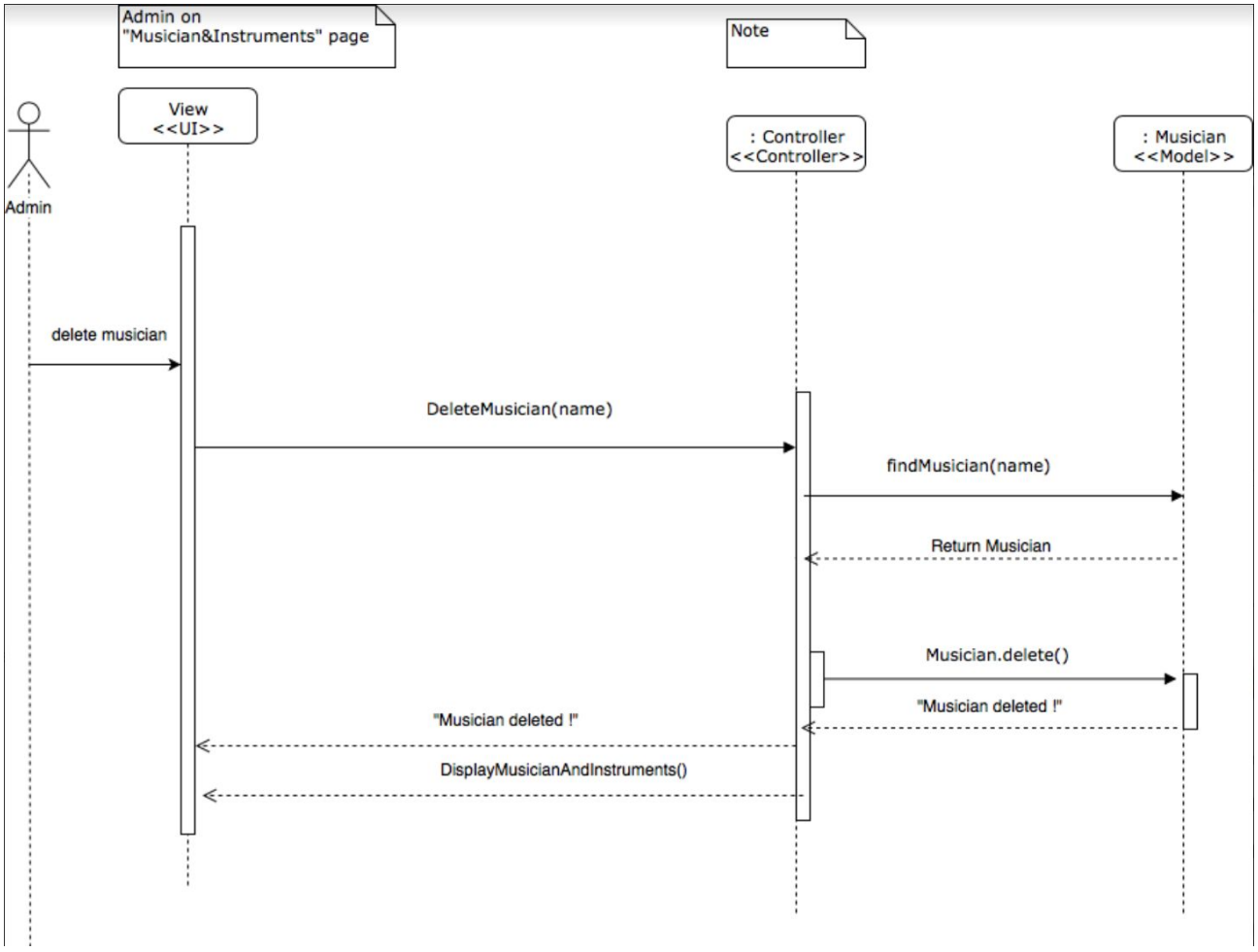
Number	11	
Name	Add Musician	
Summary	The admin enters the necessary information to add a musician	
Preconditions	Being on the Musicians&Instruments page, having logged in as admin	
Postconditions	A musician has been added	
Primary actor(s)	Admin	
Secondary actor(s)	Musician Class	
Main Scenario	**Step**	**Action**
	1	System displays form
	2	User fills form
	3	User presses OK button
	4	System saves the musician
	5	System displays Musician&Instruments page
Extensions	**Step**	**Branching Action**
	2a	User presses Cancel button
	2b	System displays Musicians&Instruments page
	3a	If the form has not been correctly filled, display error


Admin click "modify" on events page
Note
View <<UI>>
:Musician <<Controller>>
Admin
modify
AddMusician()
DisplayMusicianForm()
setValues(Name)
new Musician(name)
DisplayMusicianAndInstrumentPage()

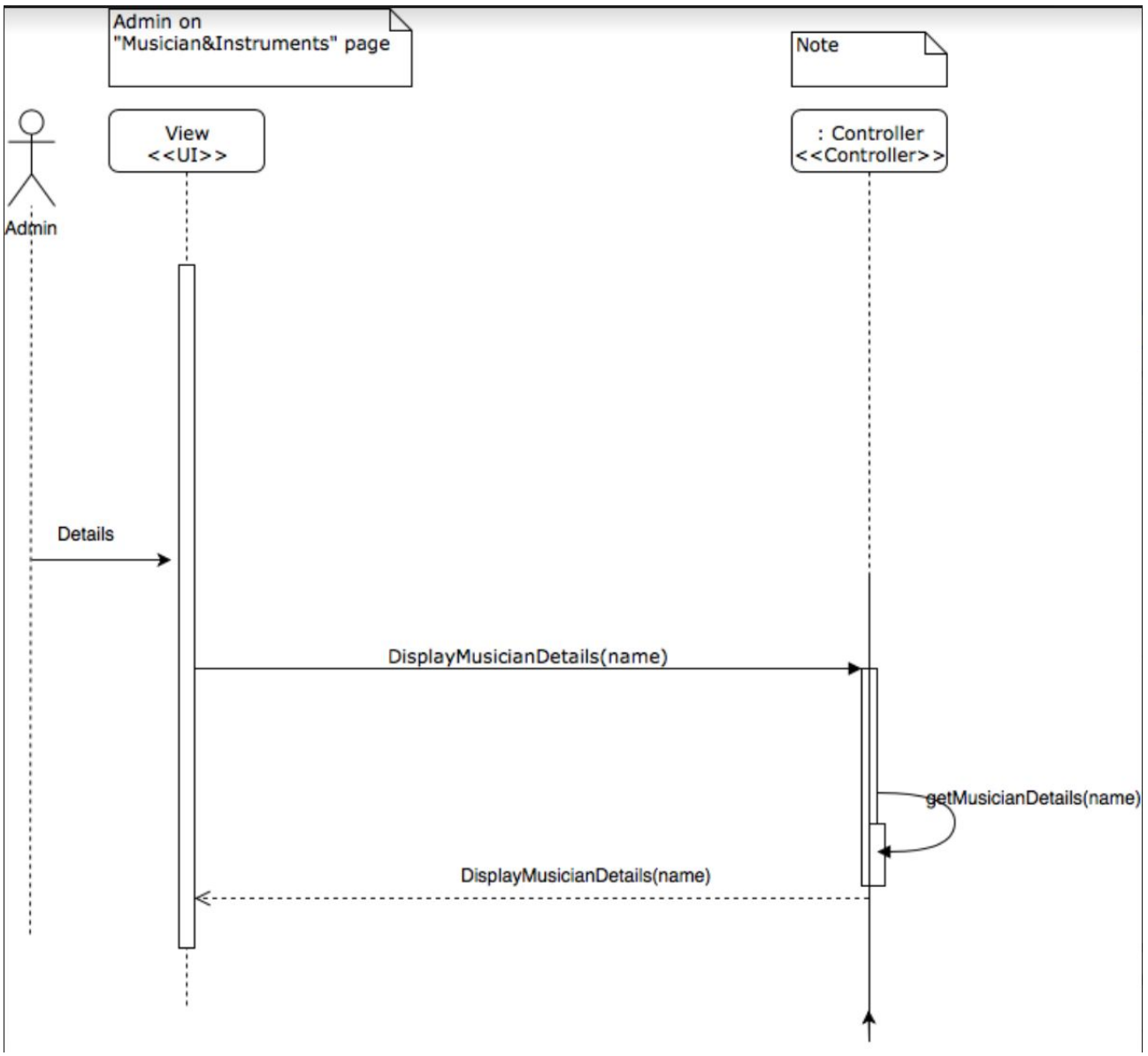
Number	12	
Name	Modify Musician	
Summary	The admin enters the modified information of a musician	
Preconditions	Being on the Musicians&Instruments page, having logged in as admin	
Postconditions	The musician has been modified	
Primary actor(s)	Admin	
Secondary actor(s)	Musician Class	
Main Scenario	**Step**	**Action**
	1	System displays form
	2	User fills form
	3	User presses Save button
	4	System saves the musician
	5	System displays Musician&Instruments page
Extensions	**Step**	**Branching Action**
	2a	User presses Cancel button
	2b	System displays Musicians&Instruments page
	3a	If the form has not been correctly filled, display error


Admin on
"Musician&Instruments" page
Note
View
<<UI>>
: Musician
<<Controller>>
Admin
Modify musician
ModifyMusician(Name)
DisplayMusicianForm(Name)
setValues(Name, instruments)
updateMusician(name, instruments)
saveMusician()
DisplayMusicianAndInstruments()

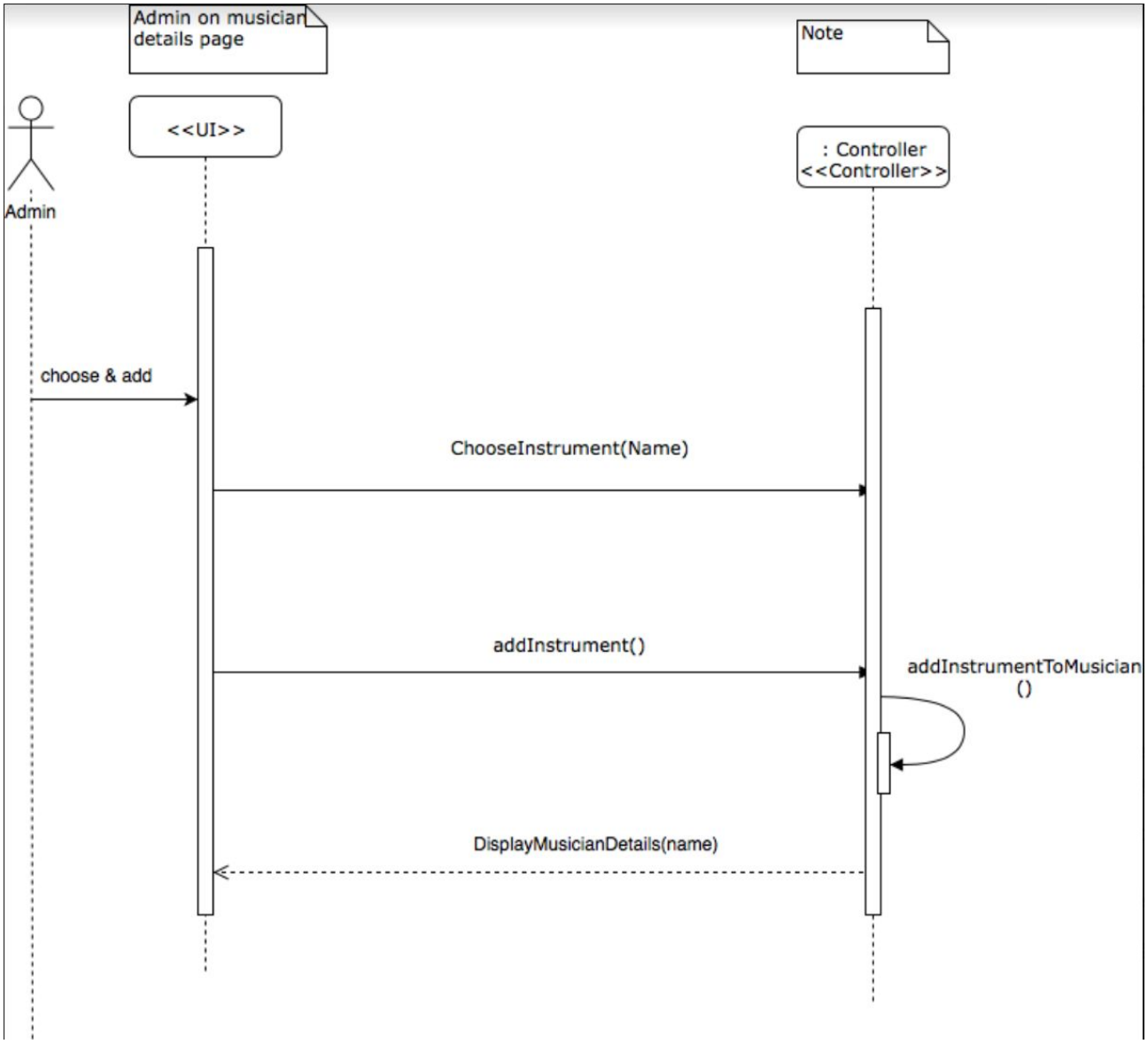
Number	13	
Name	Delete Musician	
Summary	Remove a musician from the musicians list	
Preconditions	Being on the Musicians&Intruments page, having logged in as admin, having selected a musician	
Postconditions	The musician has been deleted	
Primary actor(s)	Admin	
Secondary actor(s)	Musician Class	
Main Scenario	**Step**	**Action**
	1	System displays confirmation window
	2	User presses OK button
	3	System deletes the musician
	4	System displays Musicians&Intruments page
Extensions	**Step**	**Branching Action**
	2a	User presses Cancel button
	2b	System displays Musicians&Intruments page


Admin on
"Musician&Instruments" page
Note
View
<<UI>>
: Controller
<<Controller>>
: Musician
<<Model>>
Admin
delete musician
DeleteMusician(name)
findMusician(name)
Return Musician
Musician.delete()
"Musician deleted !"
"Musician deleted !"
DisplayMusicianAndInstruments()

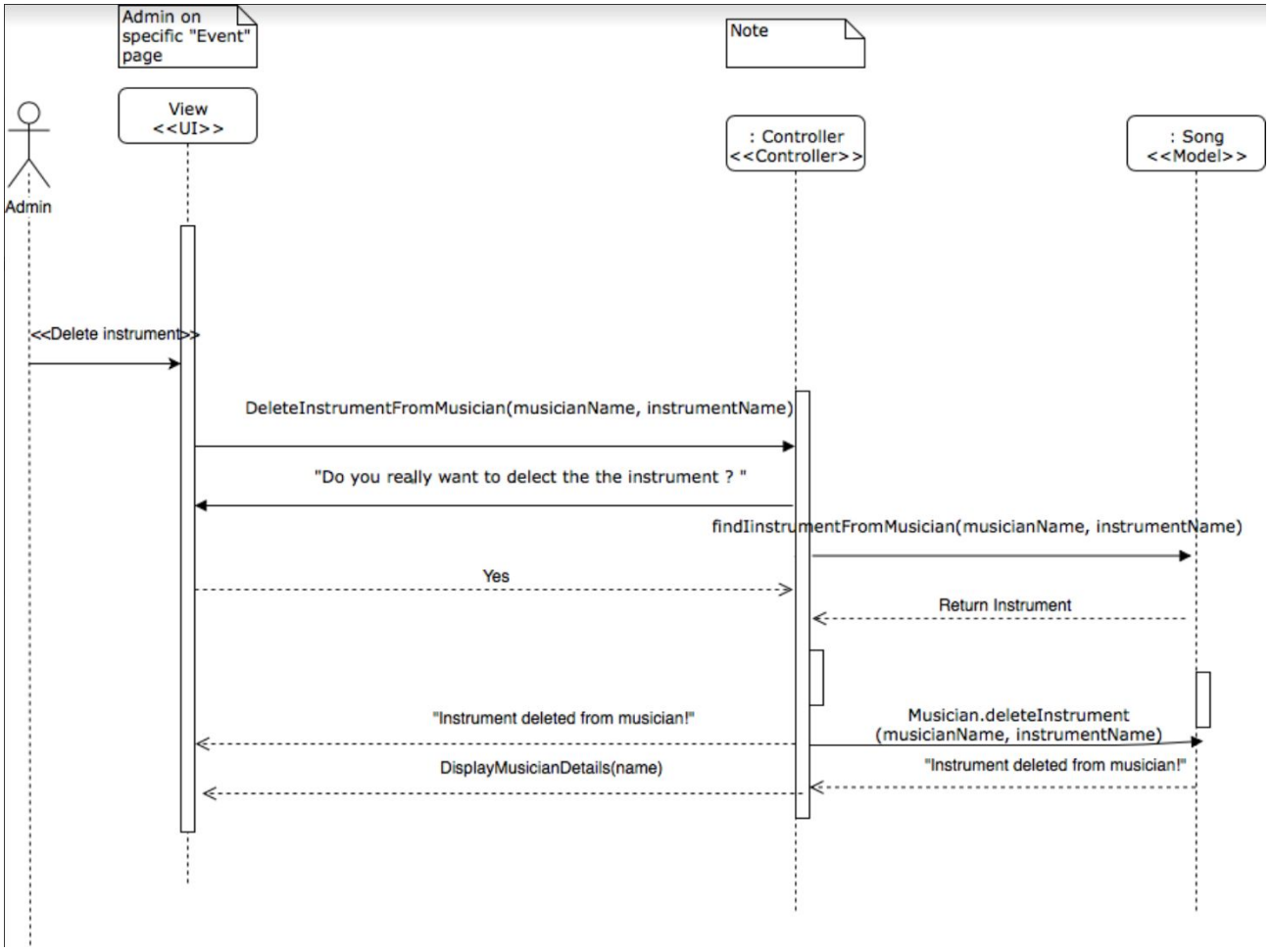
Number	14	
Name	Check Musician Details	
Summary	Displays the details page of a musician	
Preconditions	Being on the Musicians&Instruments page, having logged in as an admin, having selected a musician	
Postconditions	Being on the details page of a musician	
Primary actor(s)	Admin	
Secondary actor(s)	Musician Class	
Main Scenario	**Step**	**Action**
	1	System displays the details page of the selected musician


Admin on
"Musician&Instruments" page
Note
View
<<UI>>
: Controller
<<Controller>>
Admin
Details
DisplayMusicianDetails(name)
getMusicianDetails(name)
DisplayMusicianDetails(name)

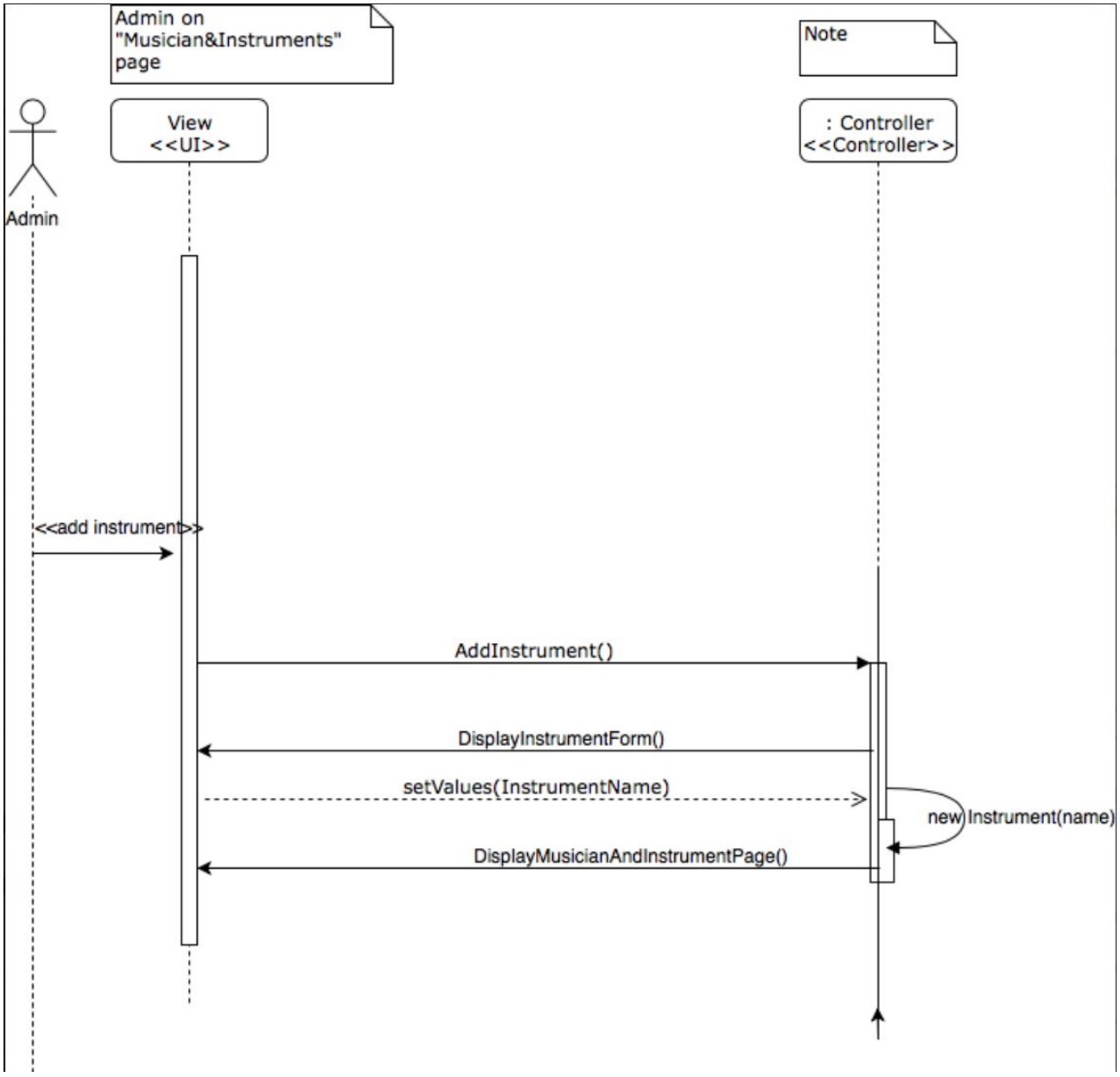
Number	15	
Name	Add Instrument To Musician	
Summary	Adds an instrument to a musician	
Preconditions	Being on the details page of a musician, being logged in as an admin	
Postconditions	The instrument has been added	
Primary actor(s)	Admin	
Secondary actor(s)	Musician Class, Instrument Class	
Main Scenario	**Step**	**Action**
	1	The user chooses an instrument in the scrolling menu
	2	The user presses the Add instrument button
	3	The system adds the instrument to the musician
Extensions	**Step**	**Branching Action**
	2a	The chosen instrument is already in the list of instruments for that musician
	2b	Display « The chosen instrument is already in the list of instruments for that musician»


Admin on musician details page
Note
<<UI>>
: Controller
<<Controller>>
Admin
choose & add
ChooseInstrument(Name)
addInstrument()
addInstrumentToMusician()
DisplayMusicianDetails(name)

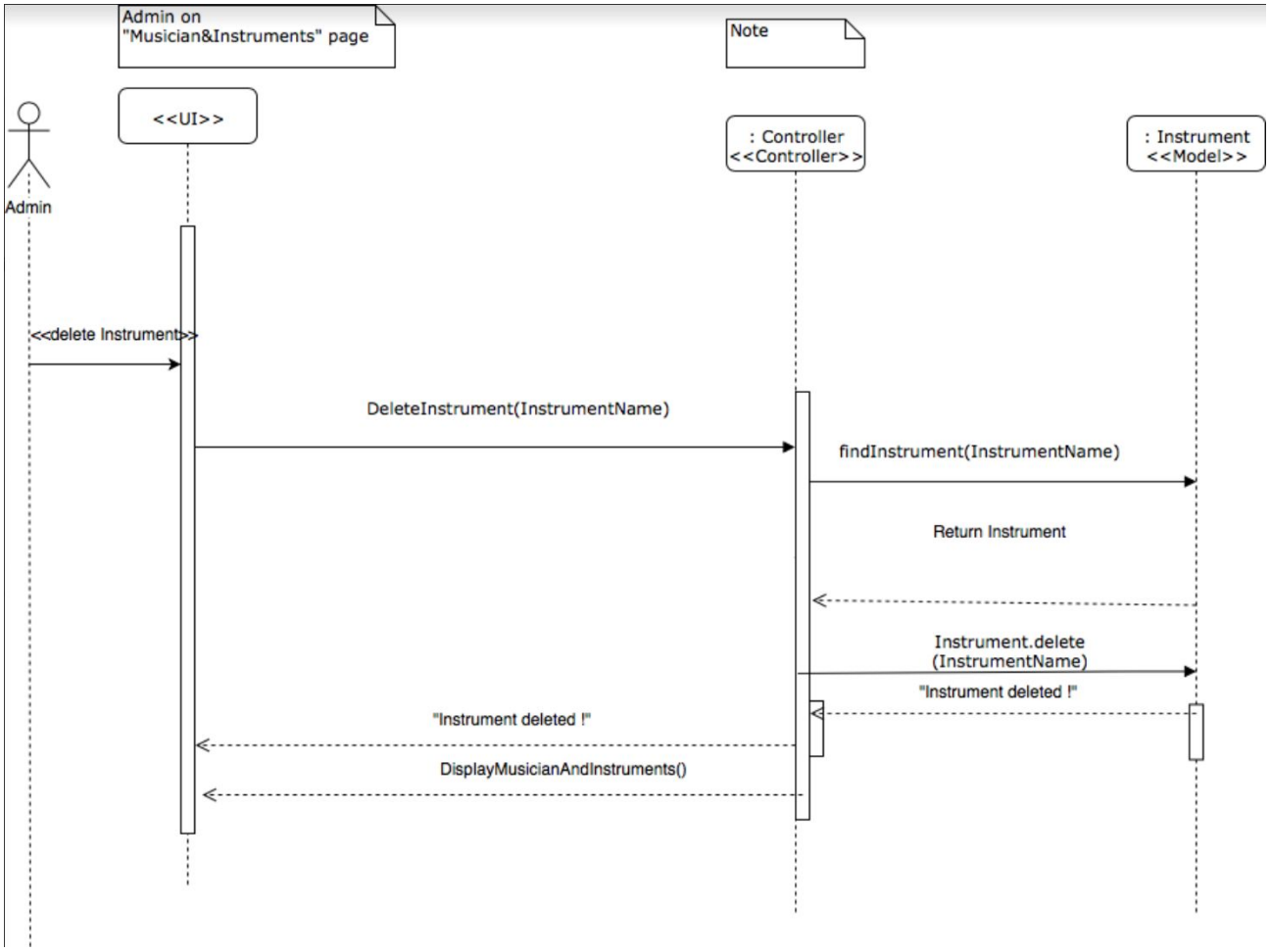
Number	16	
Name	Delete Instrument From Musician	
Summary	Remove a selected instrument from a musician	
Preconditions	Being on the details page of a musician, having logged in as admin, having selected an instrument in the list	
Postconditions	The instrument has been removed from the musician	
Primary actor(s)	Admin	
Secondary actor(s)	Musician Class	
Main Scenario	**Step**	**Action**
	1	System displays confirmation window
	2	User presses OK button
	3	System deletes the instrument
	4	System displays details page
Extensions	**Step**	**Branching Action**
	2a	User presses Cancel button
	2b	System displays details page


Admin on specific "Event" page
Note
View <<UI>>
: Controller <<Controller>>
: Song <<Model>>
Admin
<<Delete instrument>>
DeleteInstrumentFromMusician(musicianName, instrumentName)
"Do you really want to delect the the instrument ? "
findIinstrumentFromMusician(musicianName, instrumentName)
Yes
Return Instrument
"Instrument deleted from musician!"
Musician.deleteInstrument (musicianName, instrumentName)
DisplayMusicianDetails(name)
"Instrument deleted from musician!"

Number	17	
Name	Add Instrument	
Summary	The admin enters the necessary information to add an instrument	
Preconditions	Being on the Musicians&Instruments page, having logged in as admin	
Postconditions	An instrument has been added	
Primary actor(s)	Admin	
Secondary actor(s)	Instrument Class	
Main Scenario	**Step**	**Action**
	1	System displays form
	2	User fills form
	3	User presses Save button
	4	System saves the instrument
	5	System displays Musician&Instruments page
Extensions	**Step**	**Branching Action**
	2a	User presses Cancel button
	2b	System displays Musicians&Instruments page
	3a	If the form has not been correctly filled, display error

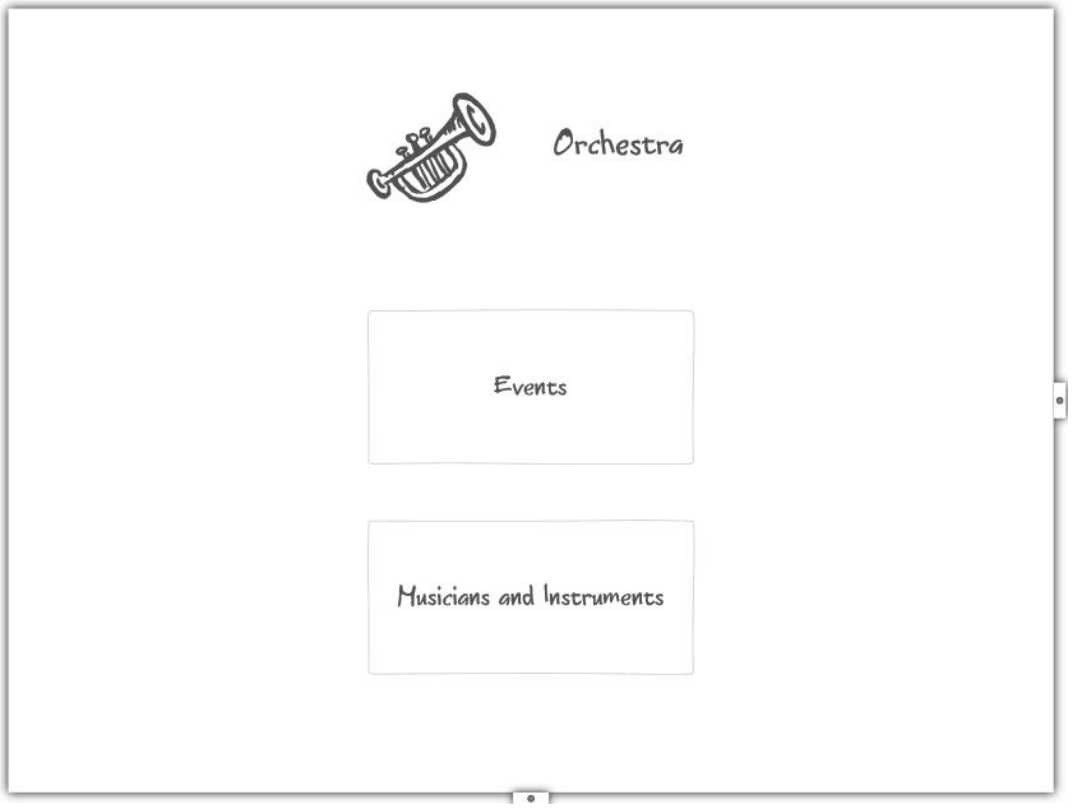

Admin on
"Musician&Instruments"
page
Note
View
<<UI>>
: Controller
<<Controller>>
Admin
<<add instrument>>
AddInstrument()
DisplayInstrumentForm()
setValues(InstrumentName)
new Instrument(name)
DisplayMusicianAndInstrumentPage()

Number	18	
Name	Delete Instrument	
Summary	Remove an instrument from the instruments list	
Preconditions	Being on the Musicians&Intruments page, having logged in as admin, having selected an instrument	
Postconditions	The instrument has been deleted	
Primary actor(s)	Admin	
Secondary actor(s)	Instrument Class	
Main Scenario	**Step**	**Action**
	1	System displays confirmation window
	2	User presses OK button
	3	System deletes the instrument
	4	System displays Musicians&Intruments page
Extensions	**Step**	**Branching Action**
	2a	User presses Cancel button
	2b	System displays Musicians&Intruments page


Admin on
"Musician&Instruments" page
Note
<<UI>>
: Controller
<<Controller>>
: Instrument
<<Model>>
Admin
<<delete Instrument>>
DeleteInstrument(InstrumentName)
findInstrument(InstrumentName)
Return Instrument
Instrument.delete
(InstrumentName)
"Instrument deleted !"
"Instrument deleted !"
DisplayMusicianAndInstruments()

Mock-up

Homepage

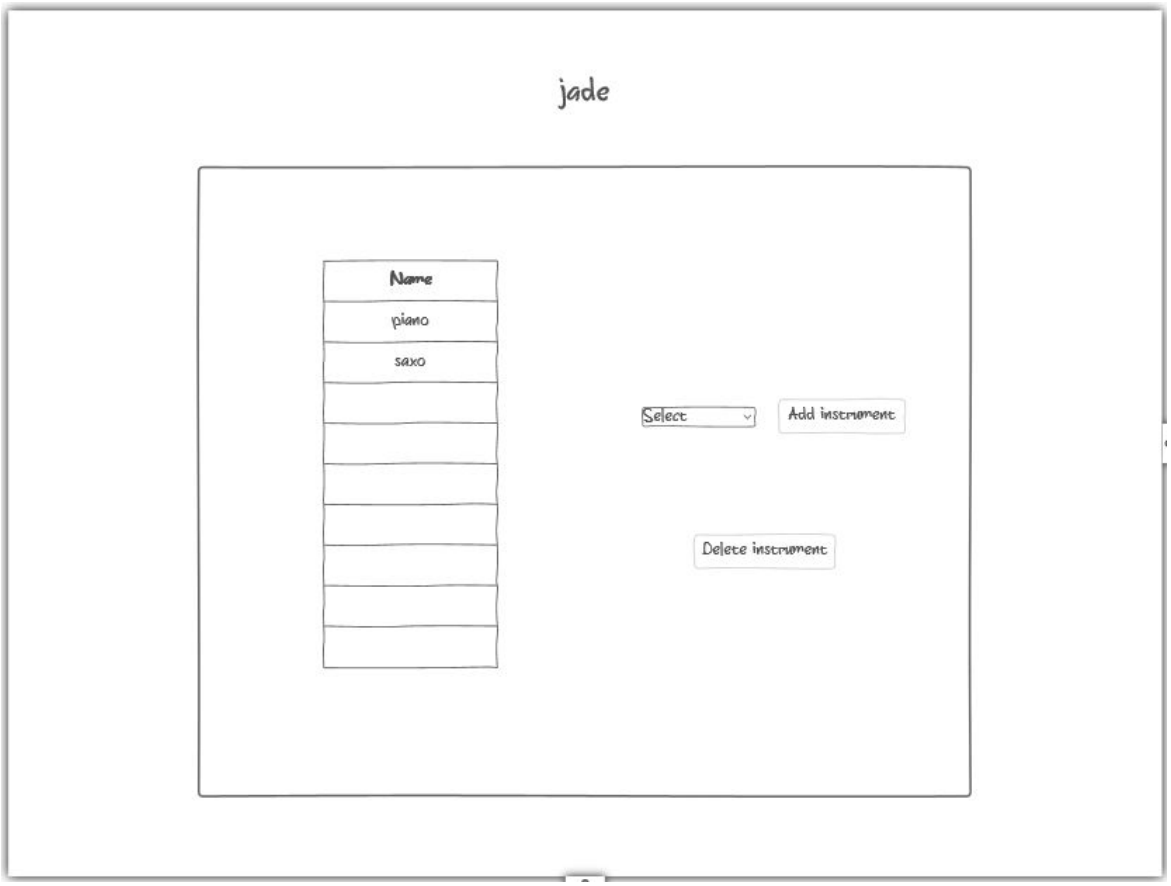


Musicians and Instruments

New/modify musician

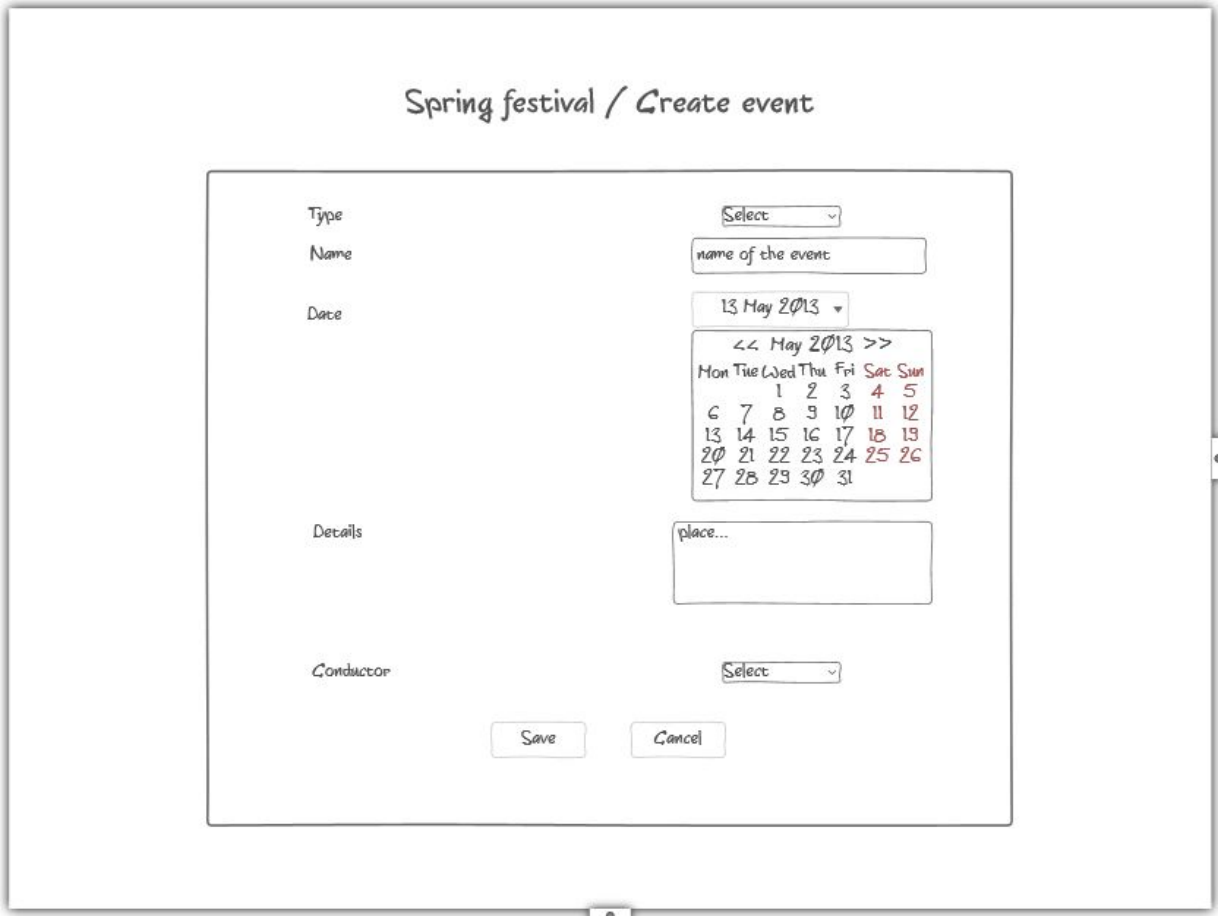
New instrument

Musician details

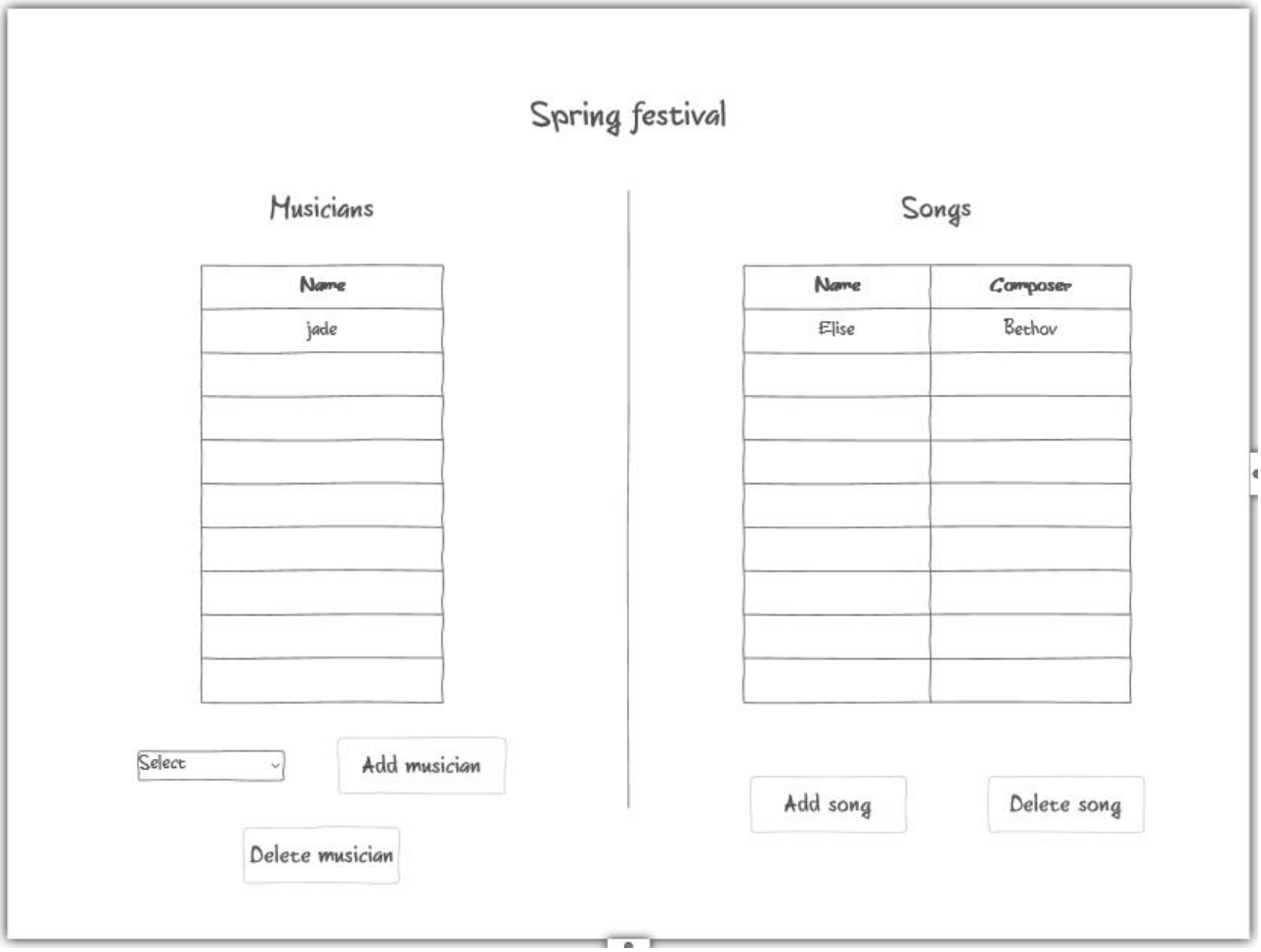


Events

Create/modify event



Event details



Add song