Project Group Members (Course Group 13):
Poojah Karunakaran (100618754)
Kevin Erskine (100762922)
Ivan Bisol (100701735)
Misha Larionov (100743093)

Step 2: Select Iteration Goal by Selecting Drivers

The primary use cases selected are:
UC-1
UC-2
UC-3
UC-4

Step 3: Choose One or More Elements to the System to Refine

The elements that will be refined in this iteration are the modules located in the different layers that are listed in iteration 1. The support of the functionality of the system requires the collaboration of components associated with the modules that are associated with modules that are located in the different layers.

Step 4: Choose One or More of Design Concepts That Satisfy the Selected Drivers

Design Decisions and Location	**Rationale and Assumptions**
Create a Domain Model for the Website	Before breaking down the different aspects of the website a Domain Model would be beneficial. Identifies the major entities along with their relationships.
Use the Angular framework to build the application front-end	Angular was chosen to support the front end as it is a widely used Javascript-based web and mobile application framework. Angular will provide a familiar environment for our web developers while delivering a performant web experience for our users that scales on all platforms, including mobile. This web application will make calls to the proprietary back-end API.
Use the Spring Boot framework to develop the	An API is used for the backend as this will allow

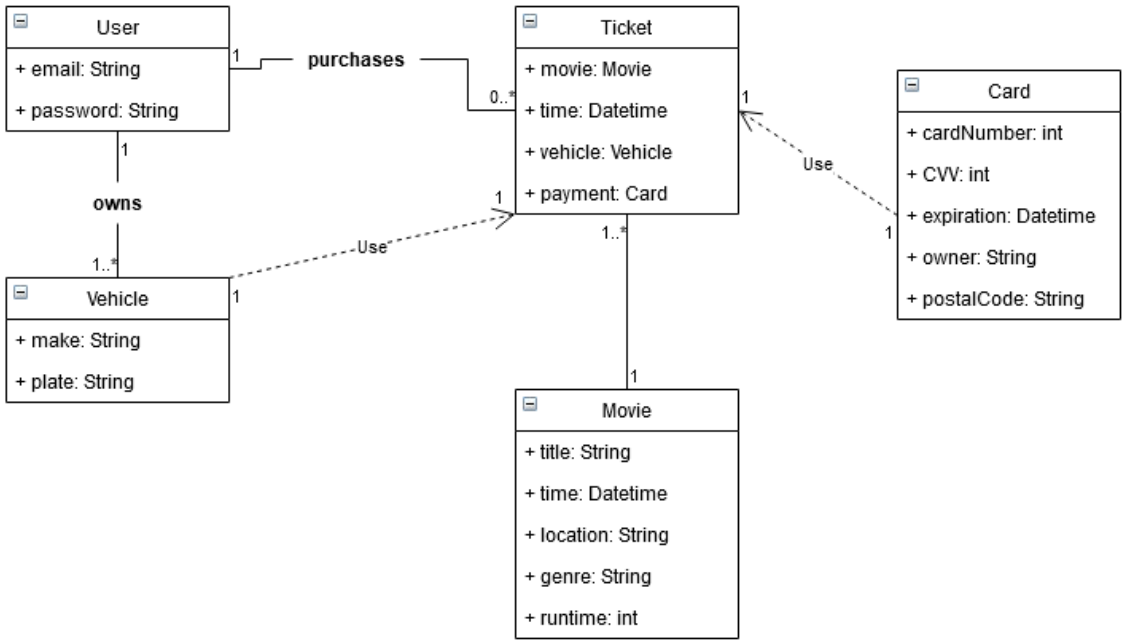
proprietary back-end API.	easier interoperability between front-end and back-end while also making maintaining individual applications easier. Spring Boot was chosen as it is a Java-based framework with support for Maven, and as such can be easily built to be modular and testable. This will make it easier to maintain the back-end in the long term.

Step 5: Instantiate Architectural Elements, Allocate Responsibilities, and Define Interfaces

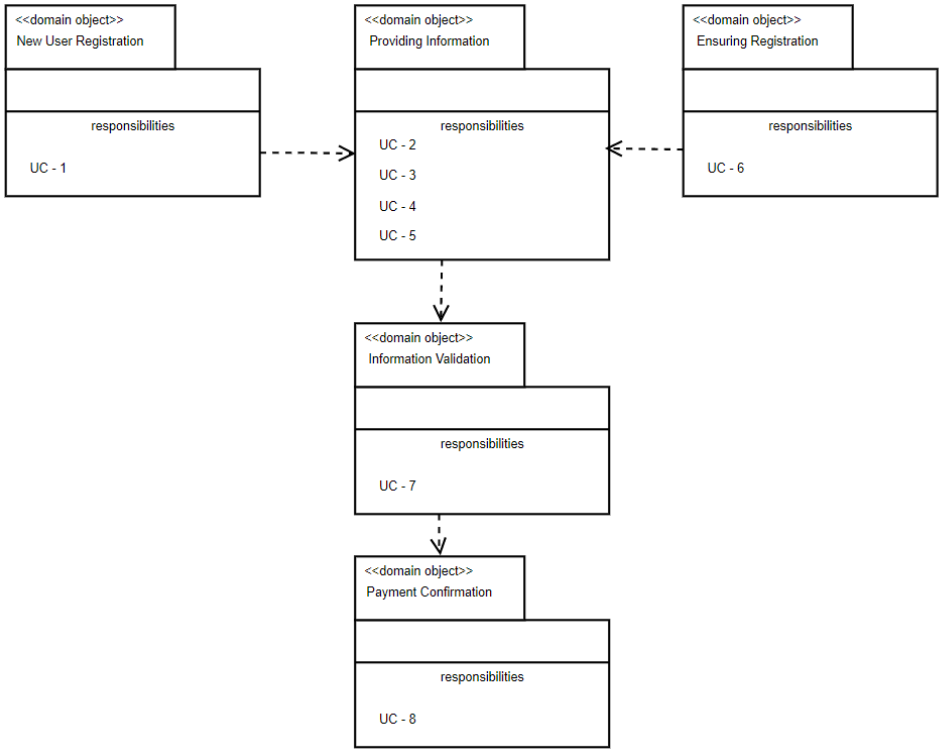
Design Decisions and Location	**Rationale**
Create only an initial domain model	Entities that participate in the primary use cases need to identified and modelled
Map the system use cases to domain objects	An initial identification of domain objects can be made by analyzing the system's various use cases.
Decompose the domain modules across layers	This allows for all the use cases to be accounted for and are supported correctly in the system. These are only done for the primary use cases

Step 6: Sketch Views and Record Design Decisions

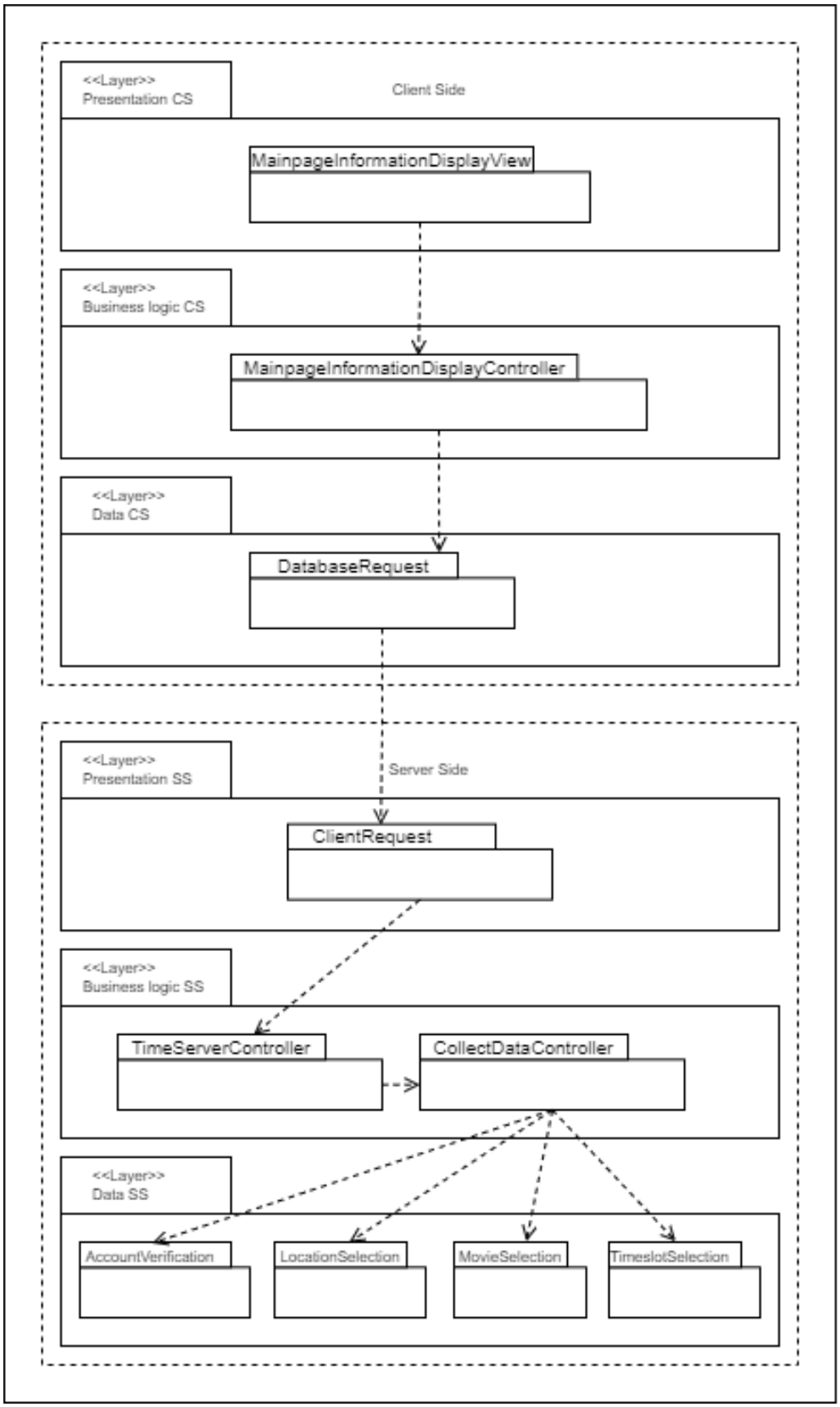
As a result of the various decisions made in Step 5, several diagrams are created.



2.1 Initial Domain Model



2.2 Domain Objects that are instantiated for the use cases



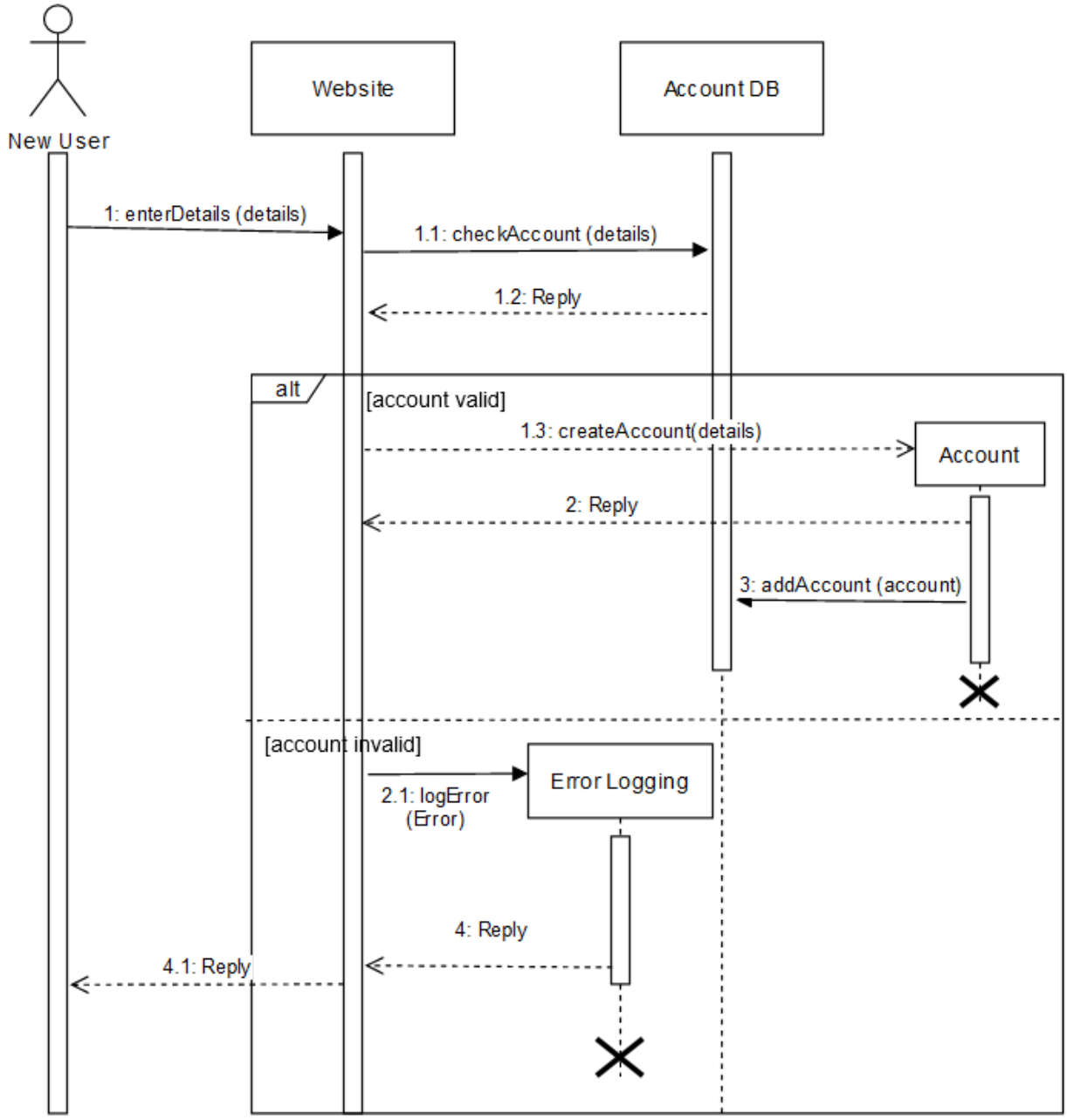
2.3 Modules that support the primary use cases

Element	**Responsibility**
MainPageInformationDisplayView	Displays the necessary pages to the client which includes creating/registering/logging into their account, selecting a location, movie and timeslot.
MainPageInformationController	Provides information to the presentation layer through network representation.
DatabaseRequest	Communicates with the database on the server side per clients request for information transaction
ClientRequest	Receives requests from the client-side
TimeServerController	Client requests are correlated to the time they have been requested for
CollectDataController	Contains the data submitted by user and retrieves information from the data layer accordingly
AccountVerification	Account information submitted by the user is verified and registration information is created in the database.
LocationSelection	Location selected by user is processed and checks if location is available for further transactions
MovieSelection	Movie selected by user is processed and checks if movie is available for further transactions
TimeslotSelection	Time slot selected by user is processed and checks if time is available for further transactions

UC-1: Account Creation

Below shows the initial sequence diagram for UC-1 (account creation). It shows how the user enters the appropriate information and how that information is then handled and sent to the appropriate element. Upon launch, the user enters the details into the website, that information is then checked and based on the response either creates a new account or displays an error and is declined. This diagram shows various interactions that take place between all the various elements.

Sequence diagram for UC-1



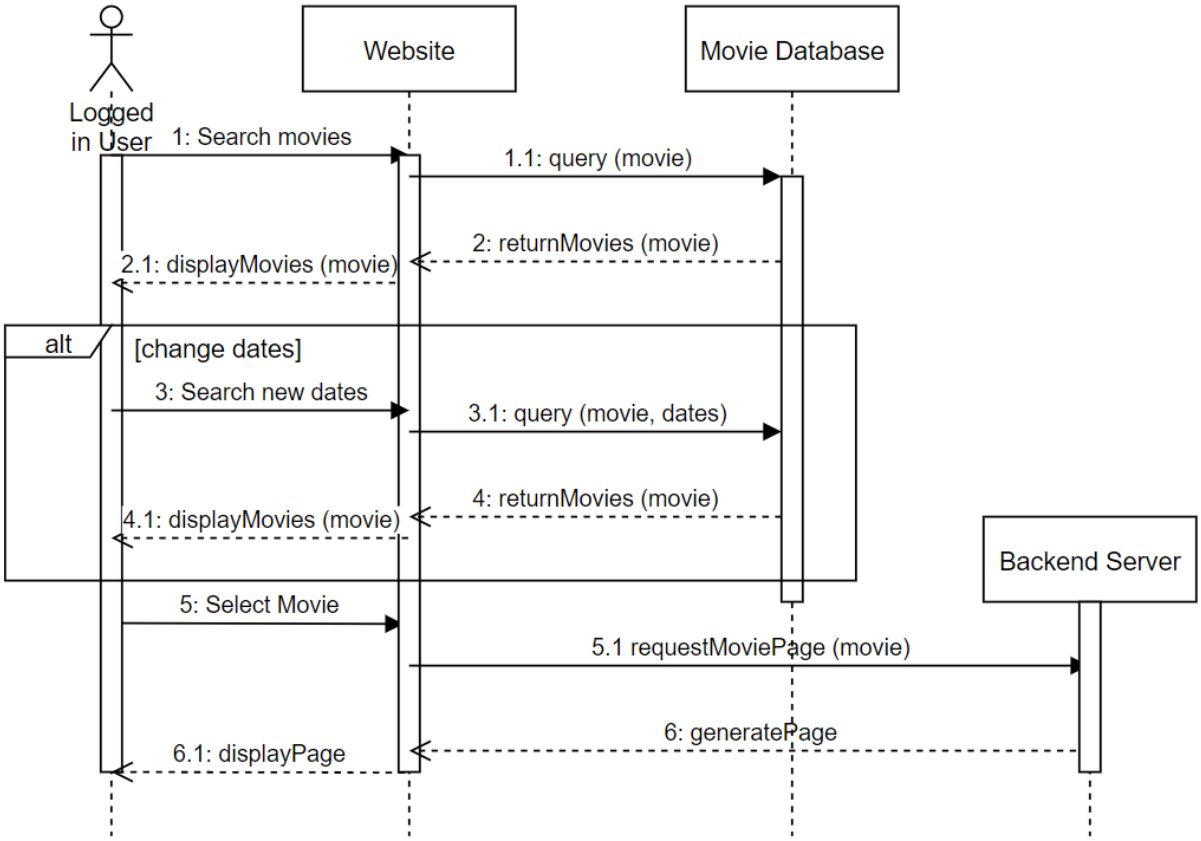
2.4 Sequence diagram for UC-1 (account creation)

All initial methods for the interfaces of the interacting elements are identified in the following table:

Method Name	**Description**
Element: Website	
checkAccount()	Checks if details entered by user match with database records
logError()	Records error if records of user do not match with database
createAccount()	Sends details of user to create a new account
Element: Account	
addAccount()	Details of user is sent to the database to be stored

UC-3: Movie Selection

Below shows the initial sequence diagram for UC-3 (movie selection). It shows how the user enters the appropriate information and how that information is then handled and sent to the appropriate element. Upon launch, searches for a movie on the website, that information is then sent as a query to the movie database and returns the appropriate response. This diagram shows various interactions that take place between all the various elements.



2.5 Sequence diagram for UC-3 (movie selection)

All initial methods for the interfaces of the interacting elements are identified in the following table:

Method Name	**Description**
Element: Website	
query()	Request the movie and or dates submitted by the user
requestMoviePage()	Request's for the movie page from the backend server

displayMovies()	Displays the requested movie on the user interface
displayPage()	Displays the requested page on the user interface
Element: Movie Database	
returnMovies()	Returns the requested movie from the database
Element: Backend Server	
generatePage()	Generates the page requested

Stuff to be done

Modules that support the primary use cases

Step 7: Perform Analysis of Current Design and Review Iteration

Not Addressed	Partially Addressed	Completely Addressed	Design Decisions Made During the Iteration
		UC-1	Necessary modules for account creation are to successfully implemented in this iteration
		UC-2	Necessary modules for location selection are to successfully implemented in this iteration
		UC-3	Necessary modules for movie selection are successfully implemented in this iteration
		UC-4	Necessary modules for time slot

			selection are successfully implemented in this iteration
UC-6			No relevant decisions made. As the requirements to meet in usability will be addressed in later iterations
UC-7			No relevant decisions made. As the requirements to meet in usability will be addressed in later iterations
	QA-2		Necessary modules to create a website with high usability are created but not implemented in this iteration
	QA-3		No relevant decisions made. As the requirements to meet in usability will be addressed in later iterations
	CON-2		No relevant decisions made. As the requirements to meet in usability will be addressed in later iterations
	CRN-3		Relevant tasks has been allocated among developers, more tasks to allocated for later iterations