**High level requirements and specifications are as below**:

* Home page
  + Displays all public events as hyperlinks. There should be 2 columns, one for past events and one for future events.
  + Clicking hyperlink takes user to event details page.
  + Logged in users see additional header items: “My events”, “Events invited to”, “Create event”
* A book reading event has
  + Title of the book, date of the event, location and start time.
  + Optionally, the event organizer may also specify the duration, description and other details.
* The event can be marked as public or private.
* The event creator can add people to the event by specifying their email. Multiple people can be invited by specifying multiple, comma-separated emails.
* Users can register on the website to create their own events.
* Anonymous users can only view public events
* “Events invited to” (visible to logged in users) will list all events as hyperlinks where the current logged in user was invited to (by matching email). Even private events are shown here if the user was invited. Hyperlinks redirect to event details page.
* “My events” (visible to logged in users) shows all events created by user, sorted by newest event-start-date first. There should be an “Edit” link in front of each entry which allows to edit the event.
  + User can edit only the events that (s)he created!
* Create a user with a specific hard-coded email, for example “[myadmin@bookevents.com](mailto:myadmin@bookevents.com)”. This user will be treated as an administrator for the system (will have admin role).
  + Admin user can edit ANY event
  + The list of events will show ALL events for him, whether public/private
* Add a special url called “/customer-support” that simply redirects to helpdesk.nagarro.com

**Technical considerations:**

* Add a unit test project and write unit test cases for at least 5 business methods; any method(s).
* Validate at both server and client side (Use some fluent validation on server side).
* Use [Authenticate] and [Authorize] attributes for authentication and authorization
* Use sample framework
* Add interceptor and log all DB commands to console